Содержание

[Введение 3](#__RefHeading__556_1027853445)

[1 Описание предметной области 4](#__RefHeading__558_1027853445)

[1.1 Обзор литературы 4](#__RefHeading__589_1698352192)

[1.2 Постановка задачи 5](#__RefHeading___Toc31277_289182750)

[1.3 Обоснование актуальности задачи 6](#__RefHeading___Toc2267_646782902)

[1.4 Обоснование используемых технологий 6](#__RefHeading___Toc2267_646782902)

[2 Проектирование архитектуры программно-информационной системы 8](#__RefHeading__1746_2075059964)

[2.1 Установление требований 8](#__RefHeading__1890_2075059964)

[2.2 Составление спецификации требований 13](#__RefHeading__872_1456780723)

[3 Реализация программно-информационной системы 18](#__RefHeading__1892_2075059964)

[4 Результаты 20](#__RefHeading___Toc25676_289182750)

[Заключение 21](#__RefHeading__1083_1027853445)

[Список используемой литературы 22](#__RefHeading___Toc26650_289182750)

[Приложение 1 ― Код программных модулей 23](#__RefHeading__1453_2075059964)

Введение

В настоящее время актуальным становится повышение производительности труда и снижение на него затрат за счёт компьютеризации и автоматизации производственных процессов. Компьютеризация – это основа технического перевооружения производства, необходимое условие повышения его эффективности. На базе ЭВМ и микропроцессоров создаются технологические комплексы, машины и оборудование, измерительные, регулирующие и информационные системы, ведутся проектно-конструкторские работы и научные исследования, осуществляется информационное обслуживание, обучение и многое другое, что обеспечивает повышение общественной и индивидуальной производительности труда, создание условий для всестороннего и гармоничного развития личности.

Целью данной работы является разработка архитектуры программно-информационной системы «каталог товаров магазинов частных объявлений». В частности, в этой работе будет представлен модуль «Частные объявления».

Необходимо создать прототип веб-приложения, реализующего функциональность доски объявлений.

Результатом разработки должно стать приложение, позволяющее размещать частные объявления, а также совершать поиск по уже имеющимся объявлениям. Полный функционал системы описан далее в данной работе. Разработанная система должна корректно работать в большинстве современных веб-браузеров.

Описание предметной области

* 1. Обзор литературы

Электронная коммерция — это сфера экономики, которая включает в себя все финансовые и торговые транзакции, осуществляемые при помощи компьютерных сетей, и бизнес-процессы, связанные с проведением таких транзакций. Существует несколько общепризнанных категорий, на которые подразделяется электронная коммерция. Как правило, такое размежевание проводится по целевой группе потребителей.

C2C (от англ. Consumer-to-consumer, буквально — «Потребитель для Потребителя») — термин, обозначающий схему электронной торговли конечного потребителя с конечным потребителем, при которой покупатель и продавец не являются предпринимателями в юридическом смысле этого слова. Такой способ осуществления электронной коммерции предполагает совершение сделок между двумя потребителями, ни один из которых не является предпринимателем в юридическом смысле слова. Интернет-площадки для подобной торговли являются чем-то средним между рынком-толкучкой и колонкой объявлений в газете. Как правило, коммерция по схеме С2С осуществляется на сайтах Интернет-аукционов, приобретающих все большую популярность в наше время. Для клиентов таких систем основное удобство заключается в более низкой цене товара, по сравнению с его стоимостью в магазинах. Обычно, в таких коммерческих взаимоотношениях участвует третья сторона — посредник, который организует торговую площадку, например, интернет-аукцион, сайт-объявлений о купле/продаже и пр. Также, посредник может являться гарантом и/или исполнителем проведения платежа, Посредник не является гарантом получения товара, но в некоторых случаях может влиять на разрешение спорных ситуаций. Также посредник не участвует в продвижении товара, этим занимается продавец самостоятельно. К особенностям схемы С2С можно отнести низкие транзакционные издержки, более низкую цену за товар.

Преимущества электронной коммерции:

### Для потребителей:

* Повсеместность
* Анонимность
* Большой выбор товаров и услуг
* Персонализация
* Более дешевые продукты и услуги
* Оперативная доставка
* Электронная социализация

### Для общества

* Широкий перечень предоставляемых услуг (например, образование, здравоохранение, коммунальное обслуживание)
* Повышение уровня жизни
* Повышение национальной безопасности
* Уменьшение «цифрового» разрыва
* Онлайн-продажа/заказ товаров/услуг уменьшает автомобильный трафик и снижает загрязнение окружающей среды

Рекламное объявление - это информационное сообщение в любом печатном издании, оплачиваемое и предназначенное для обращения к массовой аудитории с целью убеждения, информирования, содействия, побуждения или изменения отношения к характеру потребления, поддержки или одобрения конкретных товаров, услуг, идей или организаций.

Товар — любая вещь, которая участвует в свободном обмене на другие вещи. Предметы, произведённые для личного потребления производителем, в экономическом смысле товарами не являются.

Услуга — результат, по меньшей мере, одного действия, обязательно осуществлённого при взаимодействии поставщика и потребителя, и, как правило, нематериальна.

Источником, регулирующим все правовые нормы в отношении размещения рекламы, являются законодательные акты Республики Беларусь. В них описываются все правила размещения рекламы (в том числе рекламных объявлений), полномочия отдельных лиц в этой области, а также указаны основные органы, занимающиеся урегулированием в данном направлении.

При разработке архитектуры данного приложения были использованы различные источники, в том числе книга Грейди Буча, Джеймса Рамбо и Айвара Джекобсона «Язык UML».

Основными источниками, использованными для написания серверной части приложения, были следующие интернет-ресурсы: документация по веб-API ASP.NET: <http://www.asp.net/web-api>, документация по ASP.NET MVC: <http://www.asp.net/mvc>.

Для разработки моделей базы данных был использован интернет-ресурс, посвященный технологии Entity Framework: <http://www.asp.net/entity-framework>.

Основными источником, используемым для построения front-end части был интернет-ресурс, посвященный фреймворку Angular-Js: <http://angular.ru>.

* 1. Постановка задачи

Задачей этого курсового проекта является проектирование и разработка веб-приложения, реализующего функционал доски объявлений.

Данная система предназначена для исполнения следующих процессов:

* Создания объявлений;
* Удаление и редактирование уже имеющихся объявлений;
* Поиск по объявлениям;
* Просмотр информации о конкретном объявлении;
* Оценка объявлений;
* Расчёт статистики для объявлений;
* Комментирование объявлений;
  1. Обоснование актуальности задачи

Виртуальная доска объявлений – именно тот случай, когда глобальная виртуализация мира и практически повсеместная доступность интернета делают онлайновый вариант более приемлемым, чем привычный бумажный. Да и очевидных преимуществ такого решения несравнимо больше.

Большинство интернет-ресурсов, посвященных данной теме имеют следующие недостатки:

* При большинстве действий на сайте браузеру приходится выполнять запрос на сервер для получения новой страницы, содержащей дополнительные данные, которые могут составлять лишь небольшую часть исходной страницы. Это приводит к лишней потере трафика, также частая перезагрузка страницы раздражает пользователей. Решением данной проблемы будет использование для front-end приложения архитектуры SPA(Single page application), благодаря этому решению приложение будет подгружать лишь необходимую часть страницы, без её полной перезагрузки, а также уменьшит потери трафика.
* Сайты, имеющие функцию поиска по объявлениям, позволяют пользователю получать результаты поиска лишь по одному запросу, то есть если пользователю нужны одновременно результаты несколько поисков- ему приходится открывать столько вкладок сайта, и в каждом искать нужную ему информацию. Решением данной проблемы будет размещение для каждого запроса поиска отдельного блока, на который можно переключиться для просмотра результатов поиска по данному запросу.

1.4 Обоснование выбранных технологий

Используемые технологии:

* AngularJs. Выбрана из-за того, что позволяет легко создать приложение с архитектурой SPA, а также позволяет легко связывать данные.
* Entity Framework. Выбрана потому что предоставляет возможность работы с базами данных через объектно-ориентированный код C#. Этот подход предоставляет ряд существенных преимуществ: нам не нужно беспокоиться о коде доступа к данным, нам не нужно знать деталей работы СУБД SQL Server и синтаксиса языка запросов T-SQL, вместо этого мы работаем с таблицами базы данных как с классами C#, с полями этих таблиц - как со свойствами классов, а синтаксис SQL-запросов, который в ADO.NET раньше нужно было вставлять в код C# в виде команд, заменен на более удобный подход с LINQ. Entity Framework берет на себя обязанности по преобразованию кода C# в SQL-инструкции.
* Web Api. Выбрана потому что позволяет создать свои RESTfull сервисы, для взаимодействия с сервером, в проекте используется для отсылки данных.

Проектирование архитектуры программно-информационной системы

* 1. Установление требований
     1. Основные замечания по проекту

Цели и рамки проекта

Целью данного проекта является разработка информационной системы для размещения информации о товарах магазинов и частных объявлений. Название системы - ИС «Каталог товаров магазинов и частных объявлений».

Деловой контекст

Мы хотим создать площадку, которая будет максимально полезна магазинам и покупателям. Как показывает официальная статистика, основные нарушения в интернет магазинах связаны с отсутствием полной информации о продавце и о самом товаре. Нередко предоставляется недостоверная информация о продавце и о товаре. Веб-портал даст возможность для каждого продавца размещать информацию о существующих товарах, что поможет увеличить продажи, количество звонков в офис, привлечь новых посетителей на сайт и сформировать постоянную аудиторию, также каждый пользователь сможет размещать частные объявления.

Участники проекта

Заказчик : Овсянников К.В.

Разработчики: Малахов В.В., Лавицкий Е.Д., Мисников А.Г.

Идеи в отношении решений

Программа должна быть реализована в виде настольного приложения для операционных систем семейств MS Windows и UNIX. Программное обеспечение должно поставляться в виде архива с разработанным модулем и необходимыми для корректной работы библиотеками.

Обзор документа

В разделе «Системные сервисы» описывается требования к разрабатываемой информационной системе. В разделе «Системные ограничения» определяются критерии и ограничения разрабатываемой системы при обслуживания и дальнейшей поддержке.

В разделе «Проектные вопросы» освещаются прочие проектные вопросы.

* + 1. Системные сервисы

Рамки системы можно моделировать с помощью диаграммы контекста

представленной на рисунке 1.

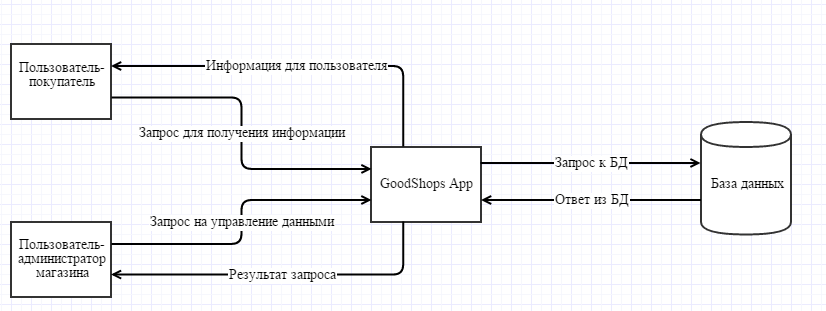


Рисунок 1 - Контекстная диаграмма «GoodShops App»

Приложение может взаимодействовать с двумя видами пользователей. Обычным пользователям-покупателям оно позволяет получить информацию о каких-либо объявлениях, товарах или магазинах. Пользователь-администратор магазина имеет возможность изменять данные о магазине. Приложение получает данные из базы данных.

Функциональные требования

ИС должна обеспечивать следующие функциональные возможности:

* Создания объявлений;
* Удаление и редактирование уже имеющихся объявлений;
* Поиск по объявлениям;
* Просмотр информации о конкретном объявлении;
* Оценка объявлений;
* Расчёт статистики для объявлений;
* Комментирование объявлений;

Требования к данным

ИС должна хранить свои данные в специализированной базе.

* + 1. Системные ограничения

Требования к интерфейсу

ИС должна иметь интуитивно-понятный интерфейс, не требующий долгого изучения и разработанный для удобства выполнения сложных операций. Интерфейс приложения должен иметь частичный альтернативный способ взаимодействия (через горячие клавиши) с элементами управления. Также приложение должно иметь адаптивный дизайн элементов управления.

Требования к производительности

Приложение должно корректно работать на системах с поддержкой IIS7 и старше имеющей 4гб оперативной памяти и 2гб видеопамяти. Время ответа от сервера не должно превышать 30 секунд.

Требования к безопасности

ИС не должна раскрывать личных данных пользователей.

Эксплуатационные требования

ИС должна функционировать на ОС Windows 7, ОС Windows 8, ОС Windows 8.1, ОС Windows 10. ИС требует 4гб оперативной памяти и 2гб видеопамяти, процессор не ниже Intel i5 с частотой не ниже 2.2Ггц и 2 физических ядра.

Политические и юридические требования

Нет.

Другие ограничения

Не выявлены.

* + 1. Проектные вопросы

Список открытых вопросов:

* Интерфейс приложения
* Сторонние библиотеки для использования

Предварительный план-график

1.09.2015 – 1.10.2015 – Анализ и установление требований к ИС

1.10.2015 – 1.11.2015 – Спецификация требований к ИС

1.11.2015 – 1.12.2015 – Кодирование ИС

1.12.2015 – 31.12.2015 – Тестовая эксплуатация ИС

11.01.2016 – 13.01.2016 – Ввод в эксплуатацию

Предварительный бюджет

Предварительный бюджет отсутствует.

* + 1. Глоссарий и список используемых документов

Глоссарий

ПО – программной обеспечение

ИС – информационная система

ОС – операционная система

MVC - Model View Controller (шаблон проектирования разделяющий бизнес-логику, данные и их представление )

Деловые документы и формы

ГОСТ 2,105.95 — Оформление текстовых документов

* + 1. Раздел предварительных замечаний

Сравнение программы с аналогами.

Основные отличия решения от аналогов:

* Использование для front-end приложения архитектуры SPA(Single page application). Данная архитектура позволяет веб приложению выболнять все свои функции без полной перезагрузки страницы. Данная архитектура реализована с помощью фреймворка AngularJs
* Реализация функции мультипоиска, позволяющей на одной странице одновременно получать результаты по нескольким запросам поиска.
  1. Составление спецификации требований

Диаграмма иерархии классов, взаимодействующих на сервере представлена на рисунке 2

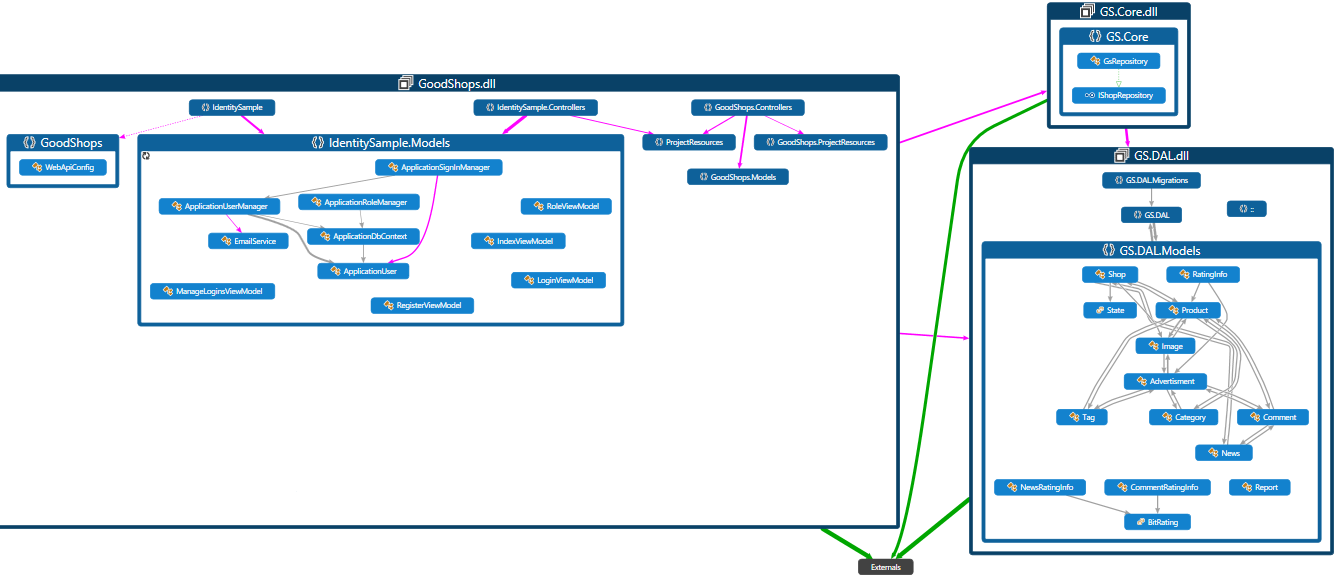


Рисунок 2 - Диаграмма иерархии классов

Таблица 1.1 - Определение варианта использования «Добавить объявление»

|  |  |
| --- | --- |
| Идентификатор | UC-01 |
| Наименование | Добавить объявление |
| Первичное действующее лицо | Зарегистрированный пользователь |
| Другие действующие лица | Нет |
| Описание | Зарегистрированный пользователь может добавлять своё частное объявление, в котором он может указать заголовок, описание, стоимость, а также добавить картинки и теги. |
| Предварительные условия | Аутентификация пользователя. Есть соединение с Базой Данных. Пользователь находится на странице создания объявления. |
| Выходные условия | Изменения зафиксированы в базе данных. |
| Нормальный поток развития | UC-01.0 Добавить объявление   1. Аутентифицироваться в системе. 2. Перейти на страницу добавления объявления. 3. Заполнить данные. 4. Нажать на кнопку “Сохранить” |
| Альтернативный поток развития | Нет |
| Исключения | UC-01.Е.1 Ошибка заполнения данных:  Если пользователь не заполнить все нужные поля, или же не заполнит их правильно – то кнопка сохранить будет недоступна.  UC-01.Е.2 Ошибка соединения с базой данных:  Наличие такой ошибки не позволит зафиксировать изменения в базе данных. |
| Включение | Нет |
| Частота использования | Используется каждый раз при добавлении объявления. |
| Ссылки | Нет |
| Предположения | Нет |
| Замечания и вопросы | Нет |

Таблица 1.2 - Определение варианта использования «Изменить объявление»

|  |  |
| --- | --- |
| Идентификатор | UC-02 |
| Наименование | Изменить объявление |
| Первичное действующее лицо | Зарегистрированный пользователь |
| Другие действующие лица | Нет |
| Описание | Зарегистрированный пользователь может изменить информацию о своём объявлении. |
| Предварительные условия | Аутентификация пользователя. Есть соединение с Базой Данных. Пользователь находится на странице изменения объявления. |
| Выходные условия | Изменения зафиксированы в базе данных. |
| Нормальный поток развития | UC-02.0 Изменить объявление   1. Аутентифицироваться в системе. 2. Перейти на страницу изменения объявления. 3. Заполнить данные. 4. Нажать на кнопку “Сохранить” |
| Альтернативный поток развития | Нет |
| Исключения | UC-02.Е.1 Ошибка заполнения данных:  Если пользователь не заполнить все нужные поля, или же не заполнит их правильно – то кнопка сохранить будет недоступна.  UC-02.Е.2 Ошибка соединения с базой данных:  Наличие такой ошибки не позволит зафиксировать изменения в базе данных. |
| Включение | Нет |
| Частота использования | Используется каждый раз при изменении объявления. |
| Ссылки | Нет |
| Предположения | Нет |
| Замечания и вопросы | Нет |

Таблица 1.3 - Определение варианта использования «Удалить объявление»

|  |  |
| --- | --- |
| Идентификатор | UC-03 |
| Наименование | Удалить объявление |
| Первичное действующее лицо | Зарегистрированный пользователь |
| Другие действующие лица | Нет |
| Описание | Зарегистрированный пользователь может удалить своё объявление. |
| Предварительные условия | Аутентификация пользователя. Есть соединение с Базой Данных. Пользователь находится на странице изменения объявления. |
| Выходные условия | Изменения зафиксированы в базе данных. |
| Нормальный поток развития | UC-03.0 Удалить объявление   1. Аутентифицироваться в системе. 2. Перейти на страницу изменение объявления. 3. Нажать на кнопку “Удалить” |
| Альтернативный поток развития | Нет |
| Исключения | UC-03.Е.1 Ошибка соединения с базой данных:  Наличие такой ошибки не позволит зафиксировать изменения в базе данных. |
| Включение | Нет |
| Частота использования | Используется каждый раз при необходимости удалить объявление. |
| Ссылки | Нет |
| Предположения | Нет |
| Замечания и вопросы | Нет |

Таблица 1.4 - Определение варианта использования «Поиск по объявлениям»

|  |  |
| --- | --- |
| Идентификатор | UC-04 |
| Наименование | Поиск по объявлениям |
| Первичное действующее лицо | Пользователь |
| Другие действующие лица | Нет |
| Описание | Пользователь может совершать поиск по частным объявлениям |
| Предварительные условия | Есть соединение с Базой Данных. Пользователь находится на сайте. |
| Выходные условия | На странице отобразится список результатов поиска. |
| Нормальный поток развития | UC-04.0 Поиск по объявлениям   1. Ввести в строку поиска нужный запрос. 2. Если нужно – выбрать категорию, в которой должен вестись поиск объявления. 3. Нажать на кнопку “Search” |
| Альтернативный поток развития | Нет |
| Исключения | UC-04.Е.1 Ошибка соединения с базой данных:  Наличие такой ошибки не позволит зафиксировать изменения в базе данных.  UC-04.Е.1 Ошибка заполнения данных:  Если пользователь не введет в строку поиска запрос, или же введет некорректный запрос- кнопка “Search” станет неактивна. |
| Включение | Нет |
| Частота использования | Используется каждый раз при необходимости поиска по объявлениям. |
| Ссылки | Нет |
| Предположения | Нет |
| Замечания и вопросы | Нет |

Таблица 1.5 - Определение варианта использования «Просмотр конкретного объявления»

|  |  |
| --- | --- |
| Идентификатор | UC-05 |
| Наименование | Просмотр конкретного объявления |
| Первичное действующее лицо | Пользователь |
| Другие действующие лица | Нет |
| Описание | Пользователь может просмотреть информацию о конкретном объявлении |
| Предварительные условия | Есть соединение с Базой Данных. Пользователь находится на сайте. |
| Выходные условия | Пользователь окажется на странице конкретного объявления, содержащей всю информацию о нём. |
| Нормальный поток развития | UC-05.0 Просмотр конкретного объявления   1. Пользователь нажимает на заголовок-ссылку нужного объявления. |
| Альтернативный поток развития | Нет |
| Исключения | UC-05.Е.1 Ошибка соединения с базой данных:  Наличие такой ошибки не позволит зафиксировать изменения в базе данных. |
| Включение | Нет |
| Частота использования | Используется каждый раз при необходимости просмотра полной информации о объявлении объявления. |
| Ссылки | Нет |
| Предположения | Нет |
| Замечания и вопросы | Нет |

Таблица 1.6 - Определение варианта использования «Оставить комментарий к объявлению»

|  |  |
| --- | --- |
| Идентификатор | UC-06 |
| Наименование | Оставить комментарий к объявлению |
| Первичное действующее лицо | Зарегистрированный пользователь |
| Другие действующие лица | Нет |
| Описание | Зарегистрированный пользователь может оставлять комментарии к существующим объявлениям. |
| Предварительные условия | Аутентификация пользователя. Есть соединение с Базой Данных. Пользователь находится на странице конкретного объявления, к которому хочет оставить комментарий. |
| Выходные условия | Изменения зафиксированы в базе данных, комментарий появился у пользователя на странице. |
| Нормальный поток развития | UC-06.0 Оставить комментарий к объявлению   1. Перейти на страницу нужного объявления. 2. Заполнить текст для комментария. 3. Нажать на кнопку “Отправить” |
| Альтернативный поток развития | Нет |
| Исключения | UC-06.Е.1 Ошибка заполнения данных:  Если пользователь не заполнить все нужные поля, или же не заполнит их правильно – то кнопка отправить будет недоступна.  UC-06.Е.2 Ошибка соединения с базой данных:  Наличие такой ошибки не позволит зафиксировать изменения в базе данных. |
| Включение | Нет |
| Частота использования | Часто. |
| Ссылки | Нет |
| Предположения | Нет |
| Замечания и вопросы | Нет |

Таблица 1.7 - Определение варианта использования «Оценить объявление»

|  |  |
| --- | --- |
| Идентификатор | UC-07 |
| Наименование | Оценить объявление |
| Первичное действующее лицо | Зарегистрированный пользователь |
| Другие действующие лица | Нет |
| Описание | Зарегистрированный пользователь может оценивать существующие объявления. |
| Предварительные условия | Аутентификация пользователя. Есть соединение с Базой Данных. Пользователь находится на странице конкретного объявления, к которому хочет оставить комментарий. |
| Выходные условия | Изменения зафиксированы в базе данных, комментарий появился у пользователя на странице. |
| Нормальный поток развития | UC-07.0 Оценить объявление   1. Перейти на страницу нужного объявления. 2. Выбрать оценку. 3. Нажать на кнопку оценивания. |
| Альтернативный поток развития | Нет |
| Исключения | UC-07.Е.1 Ошибка заполнения данных:  Если пользователь не заполнить все нужные поля, или же не заполнит их правильно – то кнопка отправить будет недоступна.  UC-07.Е.2 Ошибка соединения с базой данных:  Наличие такой ошибки не позволит зафиксировать изменения в базе данных.  UC-07.Е.3 Ошибка повторной оценки:  Пользователь попытался поставить оценку объявлению, которое уже оценивал до этого.  UC-07.Е.4 Ошибка оценки своего объявления:  Пользователь попытался оценить собственное объявление. |
| Включение | Нет |
| Частота использования | Часто. |
| Ссылки | Нет |
| Предположения | Нет |
| Замечания и вопросы | Нет |

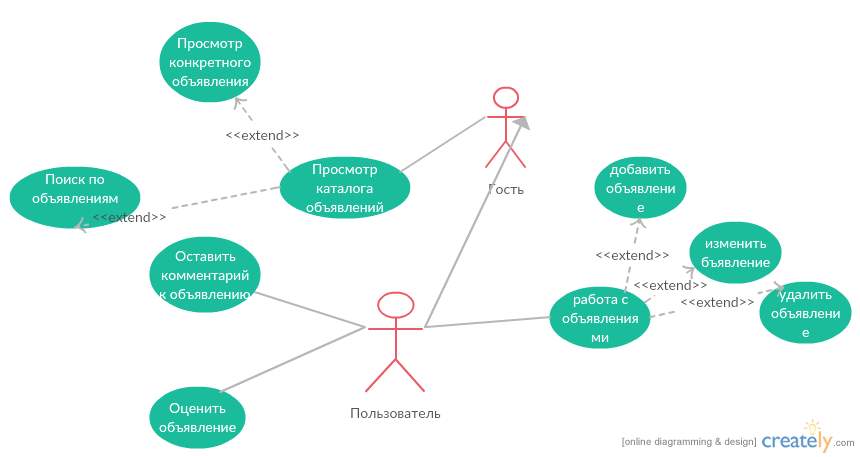


Рисунок 2 – Диаграмма вариантов использования системы

На рисунке 3 представлена диаграмма последовательности варианта использования “Добавить объявление”.



Рисунок 3 - Диаграмма последовательности варианта использования «Добавить объявление»

На рисунке 4 представлена диаграмма последовательности варианта использования “Изменить объявление”.

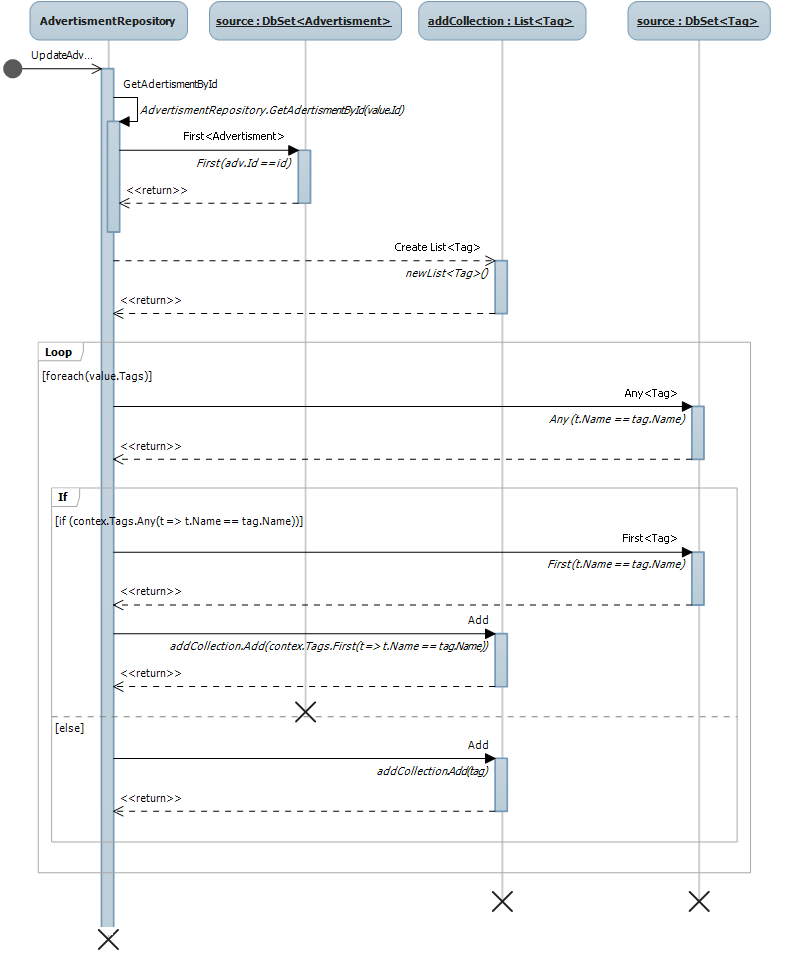


Рисунок 4 - Диаграмма последовательности варианта использования «Изменить объявление»

На рисунке 5 представлена диаграмма последовательности варианта использования “Удалить объявление”.



Рисунок 5 - Диаграмма последовательности варианта использования «Удалить объявление»

На рисунке 6 представлена диаграмма последовательности варианта использования “Поиск по объявлениям”.



Рисунок 6 - Диаграмма последовательности варианта использования «Поиск по объявлениям»

На рисунке 7 представлена диаграмма последовательности варианта использования “Просмотр конкретного объявления”.



Рисунок 7 - Диаграмма последовательности варианта использования «Просмотр конкретного объявления»

На рисунке 8 представлена диаграмма последовательности варианта использования “Оставить комментарий к объявлению”.

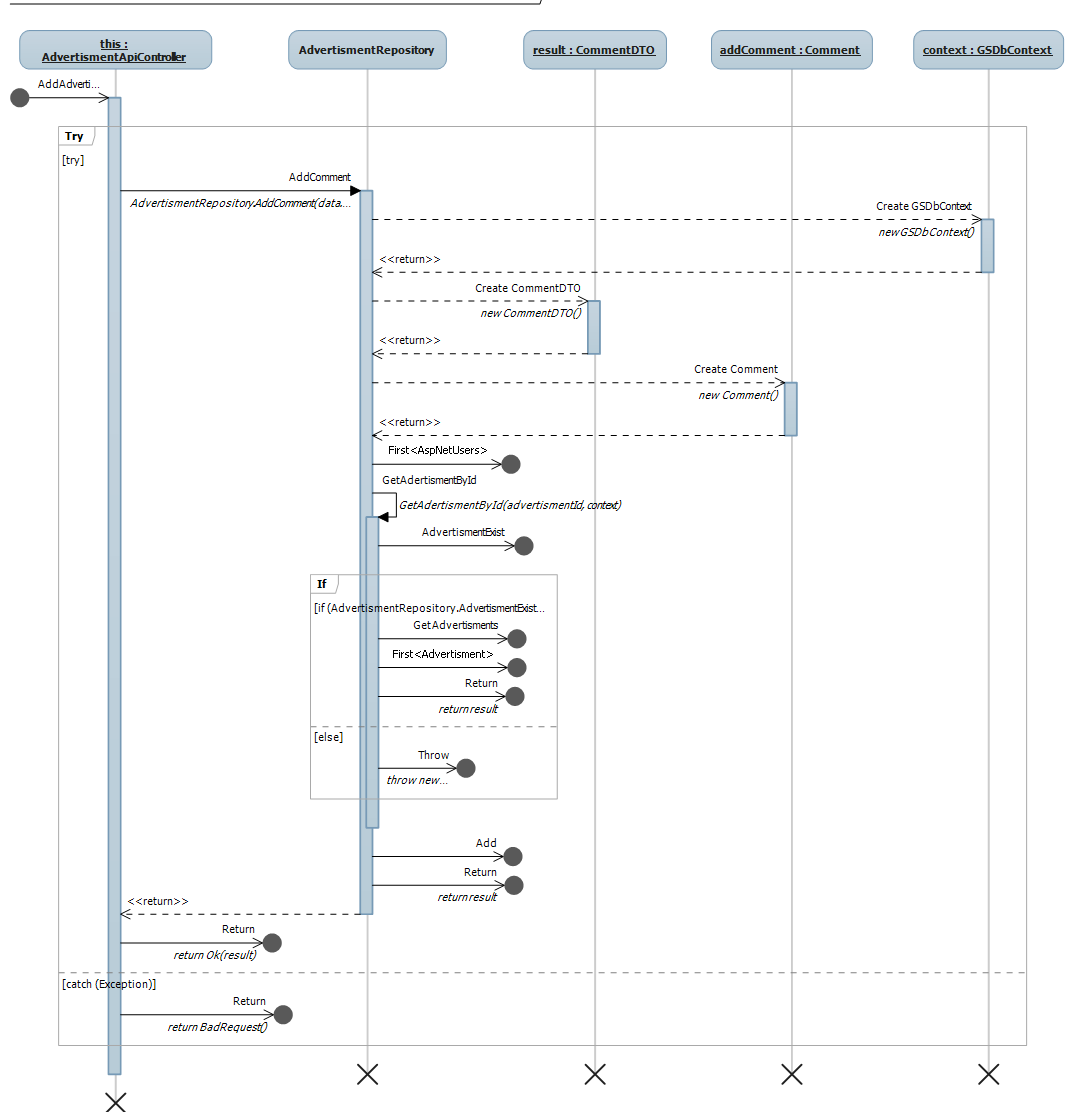


Рисунок 8 - Диаграмма последовательности варианта использования «Оставить комментарий к объявлению»

На рисунке 9 представлена диаграмма последовательности варианта использования “Оценить объявление”.



Рисунок 9 - Диаграмма последовательности варианта использования «Оценить объявление»

* 1. Проектирование структуры слоя доступа к данным

Структура базы данных.

На рисунке 10 представлена структура таблицы “Advertisments”.

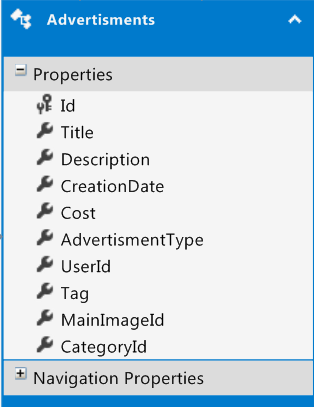


Рисунок 10 - Cтруктура таблицы “Advertisments”

Сущность “Advertisment” представляет конкретное частное объявление, она содержит следующие поля:

* Id (уникальный номер);
* Title (заголовок объявления);
* Cost (стоимость объявления);
* Description (описание объявления);
* CreationDate (дата создания объявления);
* CategoryId (внешний ключ к таблице категорий, объявление может принадлежать к некой категории, связь один-к-одному);
* MainImageId (внешний ключ к таблице Images, хранящей url адреса картинок, объявление может именить главную картинку, связь один-к-одному);
* User\_Id (внешний ключ к таблице пользователей, у каждого объявления есть владелец, связь один-ко-многим так как у одного пользователя может быть много объявлений, а конкретное объявление может принадлежать только одному пользователю).
* Tag (внешний ключ к таблице тегов, у каждого объявления может быть несколько тегов, связь один ко многим так как у одного тега может быть несколько объявлений.

На рисунке 11 представлена структура таблицы “Categories”.

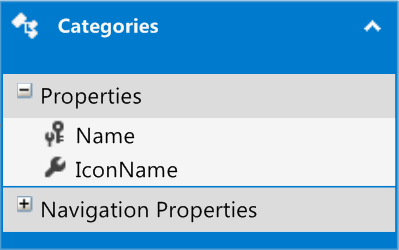


Рисунок 11 - Cтруктура таблицы “Categories”

Сущность “Category“ представляет категорию, к которой могут относиться товары и объявления, она содержит следующие поля:

* Name (первичный ключ а также имя категории);
* IconName (имя файла, являющегося иконкой данной категории).

На рисунке 12 представлена структура таблицы “Images”.

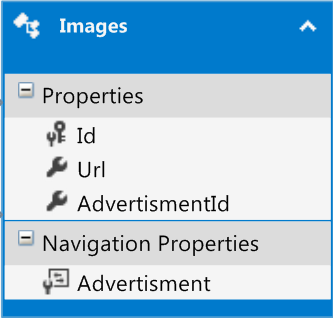


Рисунок 12 - Cтруктура таблицы “Images”

Сущность “Image” представляет картинку, которая может использоваться в объявлении или товаре, она содержит следующие поля:

* Id (уникальный идентификатор);
* Url (url адрес картинки);
* Advertisment\_Id (внешний ключ к таблице объявлений, данное значение содержит id объявления, которому принадлежит картинка, связь один-к-одному)

На рисунке 13 представлена структура таблицы “Comments”.

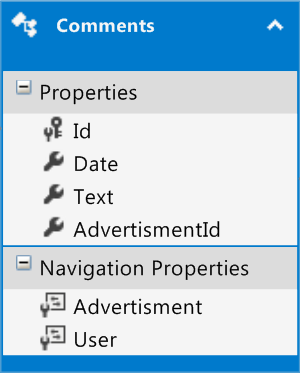


Рисунок 13 - Cтруктура таблицы “Comments”

Сущность “Comment” представляет комментарий, который может оставлен пользователем к существующему объявлению:

* Id (уникальный идентификатор);
* Date (дата создания комментария);
* Text (текст комментария);
* Advertisment\_Id (внешний ключ к таблице объявлений, данное значение содержит id объявления, которому принадлежит картинка, связь один-к-одному)

На рисунке 14 представлена структура таблицы “Tags”.

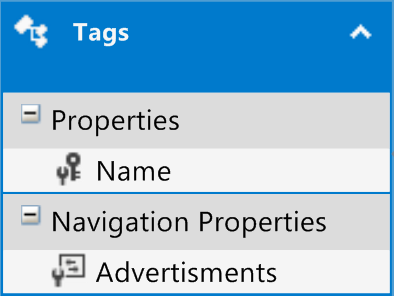


Рисунок 14 - Cтруктура таблицы “Tags”

Сущность “Tag” представляет комментарий, который может оставлен пользователем к существующему объявлению:

* Name (уникальный идентификатор, имя тега);

На рисунке 15 представлена структура таблицы “AdvertismentStatistics”.

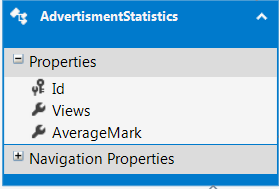


Рисунок 15 - Cтруктура таблицы “AdvertismentStatistics”

Сущность “AdvertismentStatistics” описывает статистику конкретного объявления:

* Id (уникальный идентификатор);
* Views (количество просмотров);
* AverageMark (средняя оценка объявления);

На рисунке 16 представлена структура таблицы “AdvertismentMarks”.

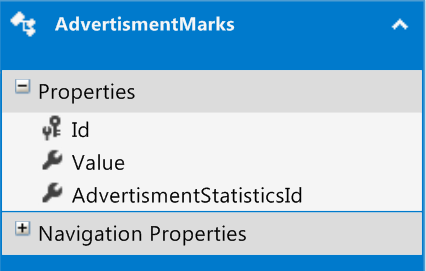


Рисунок 16 - Cтруктура таблицы “AdvertismentMarks”

Сущность “AdvertismentMark” описывает оценку, поставленную неким пользователем некому объявлению:

* Id (уникальный идентификатор);
* Views (сама оценка, варьируются от 1 до 5);
* AdvertismentStatisticsId (внешний ключ к таблице AdertismentStatistics, содержит Id объекта статистики, связанного с объявлением, которому поставили оценку);

На рисунке 17 представлена структура таблицы “AdvertismentRatingPercentage”.

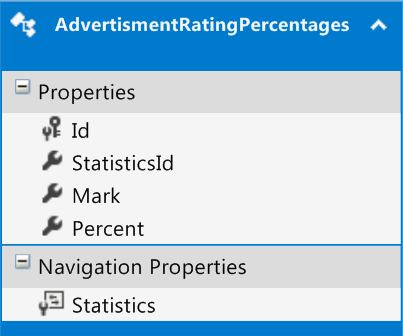


Рисунок 17 - Cтруктура таблицы “AdvertismentRatingPercentage”

Сущность “AdvertismentRatingPercentage” описывает процентное соотношение данной оценки:

* Id (уникальный идентификатор);
* Mark (оценка, процент которой содержится);
* Percent (процент выставления этой оценки этому объявлению);
* StatisticsId (внешний ключ к таблице AdertismentStatistics, содержит Id объекта статистики, связанного с объявлением, которому поставили оценку);

На рисунке 18 представлена общая структура базы данных.

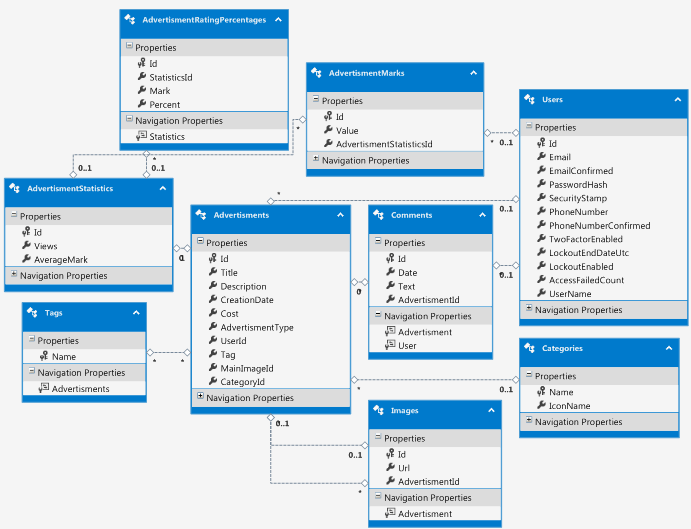


Рисунок 18- общая структура базы данных

Реализация программно-информационной системы

3.1. Описание бизнес логики

Класс AdvertismentRepository предоставляет интерфейс для удобного взаимодействия с всеми данными, связанными с частными объявлениями.

На рисунке 19 представлена диаграмма класса “AdvertismentRepository”.

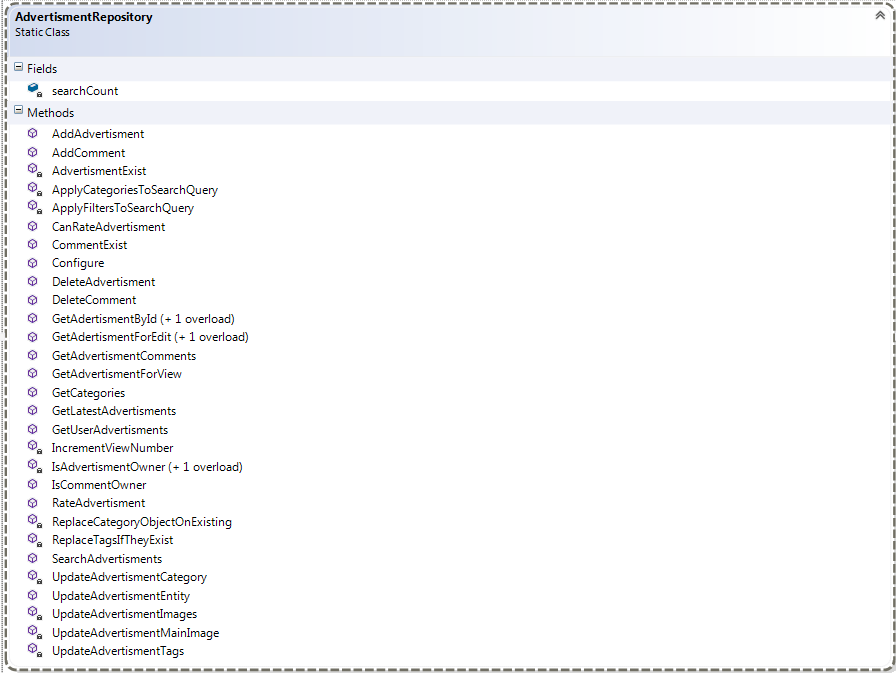


Рисунок 19- диаграмма класса “AdvertismentRepository”

Описание полей и методов класса:

* searchCount (поле, указывающее размер одной страницы поиска);
* AddAdvertisment (метод, добавляющий новое объявление в базу);
* AddComment (метод, добавляющий новый комментарий к объявлению в базу);
* AdvertismentExist(метод, который проверяет существует ли объявление с указанным id);
* ApplyCategoriesToSearchQuery (метод, который налаживает условие на выборку объявлений по указанным категориям);
* ApplyCategoriesToSearchQuery (метод, который налаживает условие на выборку объявлений по указанным фильтрам);
* CanRateAdvertisment (метод, который проверяет может ли определенный пользователь оценивать определенное объявление);
* CommentExist (метод, который проверяет существует ли комментарий с заданным id);
* Configure(метод, который конфигурирует AutoMapeper для конвертирования сущностей во ViewModel);
* DeleteAdvertisment (метод, который удаляет оъявление);
* DeleteComment (метод, который удаляет комментарий);
* GetAdvertismentById (метод, который возвращает объявление с указанным id);
* GetAdvertismentForEdit (метод, который возвращает ViewModel объявления для изменения);
* GetAdvertismentComments (метод, который возвращает комментарии определенного объявления);
* GetAdvertismentForView(метод, который возвращает ViewModel объявления для просмотра);
* GetCategories (метод, который возвращает список категорий);
* GetLatestAdvertisments (метод,который возвращает последние объявления);
* GetUserAdvertisments (метод, который возвращает объявления, принадлежащие определенному пользователю);
* IncrementViewNumber (метод, который увеличивает колличество просмотров объявления на единицу);
* IsAdvertismentOwner (метод, который проверяет, является определенный пользователь владельцем определенного объявления);
* RateAdvertisment (метод, который добавляет оценку определенному объявлению);
* ReplaceCategoriesObjectOnExisting (метод, который заменяет пришедший с клиента объект категории на уже существующий в базе);
* ReplaceTagsIfTheyExist (метод, который заменяет пришедшие с сервера теги на уже существующие в базе теги, если они существуют);
* SearchAdvertisments (метод, который ищет объявления по запросу);
* UpdateAdvertismentategory (метод, который метод, который изменяет категорию объявления);
* UpdateAdvertismentEntity (метод, который изменяет данные объявления);
* UpdateAdvertismentImages (метод, который изменяет список картинок объявления);
* UpdateAdvertismentMainImage (метод, который изменяет главную картинку объявления);
* UpdateAdvertismentTags(метод, который изменяет теги объявления);

На рисунках 20-25 представлены диаграммы последовательности основных методов класса.

3.2. Пример использования программно-информационной системы

Страница с объявлениями, принадлежащими пользователю, представлена на рисунке 8

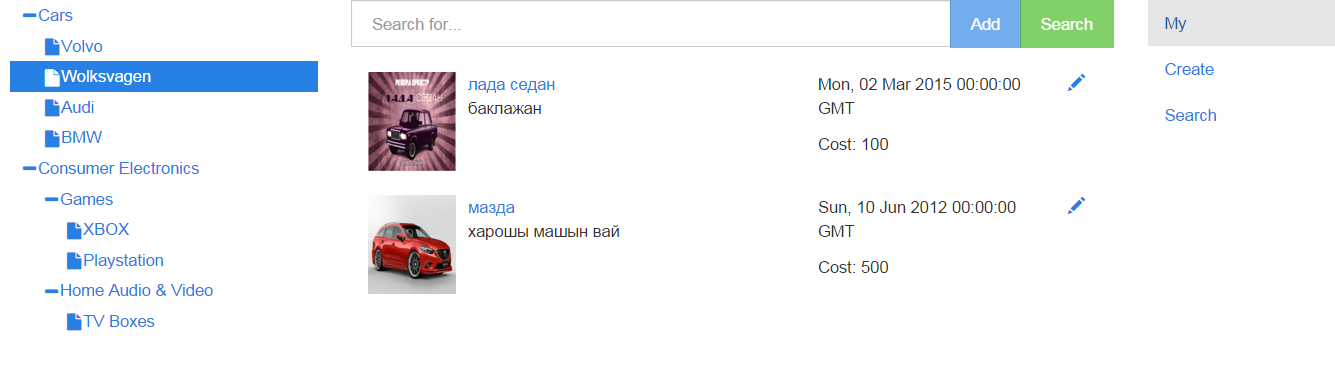


Рисунок 8 - Страница с объявлениями, принадлежащими пользователю

Страница редактирования объявления представлена на рисунке 9

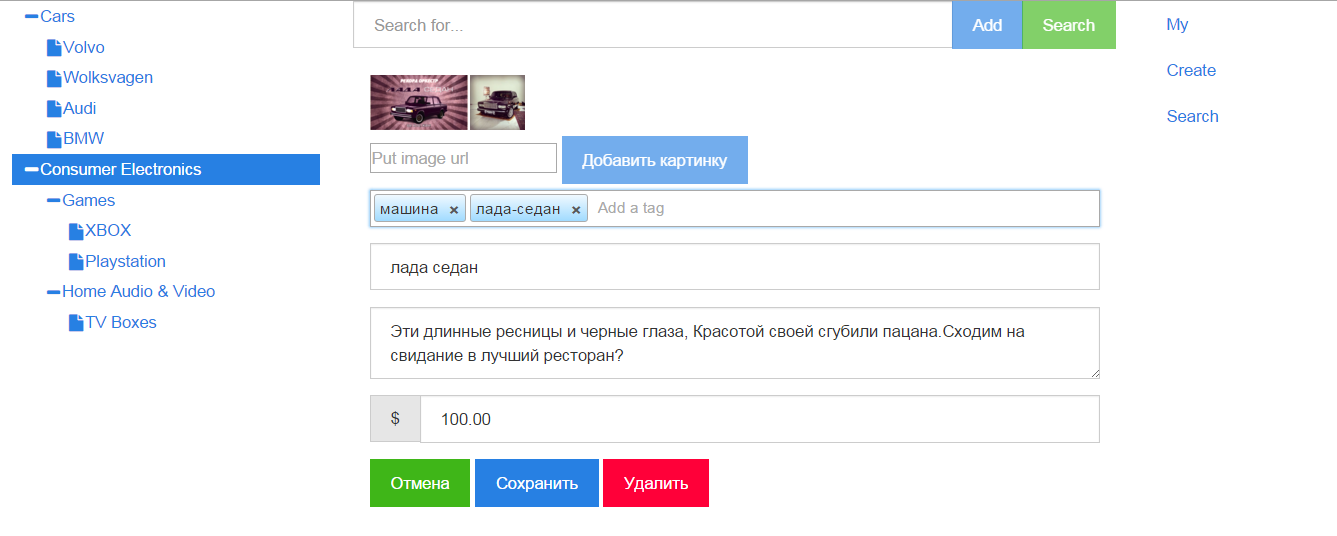


Рисунок 9 - Страница редактирования объявления

Результаты

Выполнен основной этап разработки данной ИС. Установлены новые требования к её дальнейшему развитию. Удалось добиться стабильной работы приложения при разных типах входных данных. Получены необходимые знания и расчёты для оптимизации и «облегчения» частично реализованной системы. Разработана первичная техническая документация в виде данной пояснительной записки. Проведено тестирование программных модулей.

Заключение

Разработан первый модуль информационной системы. Продумана базовая архитектура приложения и реализовано 50% намеченного в ходе планирования функционала. Реализовано централизованное хранилище и логика по хранению динамичных и статичных данных ИС.

Список используемой литературы

1. Биллинг, В. А. Основы программирования на C#. / В. А. Биллинг. – М: Бином, 2012 – 488 с.
2. Васильев, А. Н. C#. Объектно-ориентированное программирование. Учебный курс. / А. Н. Васильев. — СПб. : Питер. 2012.—320 с: ил.
3. Дейтел, Харви. Как программировать на Visual C# 2012/ / Харви Дейтел. — СПб. : Питер. 2014.— 864 с:
4. Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. 2-е изд.: Пер. с англ.Мухин Н. –М.: ДМК Пресс. – 496 с.: ил.

5 О рекламе: Закон Республики Беларусь от 10 мая 2007 г. № 225-З