

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Образовательная программа бакалавриата «Программная инженерия»

СОГЛАСОВАНО
Научный руководитель,
Научный руководитель,
преподаватель факультета Компьютерных
наук Департамента «Программная
инженерия»

УТВЕРЖДАЮ
Академический руководитель
образовательной программы
«Программная инженерия»
профессор департамента программной
инженерии, канд. техн. наук

« ____ » _____ 2020 г. Н.К.Чуйкин

« ____ » _____ 2020 г. В.В. Шилов

Агрегатор магазинов с элементами социальной сети

Пояснительная записка

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.04.01-01 81 01-1-ЛУ

Исполнитель
студент группы БПИ194

« ____ » _____ 2020 г. / Е.В.Аникеев /

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

Москва 2020

УТВЕРЖДЕН
RU.17701729.04.01-01 81 01-1-ЛУ

Агрегатор магазинов с элементами социальной сети

Пояснительная записка

RU.17701729.04.01-01 81 01-1

Листов 44

<i>Подп. и дата</i>	
<i>Инв. № дубл.</i>	
<i>Взам. инв. №</i>	
<i>Подп. и дата</i>	
<i>Инв. № подл</i>	

Москва 2020

АННОТАЦИЯ

В данном программном документе приведена пояснительная записка к программе «ShopsAggregator.apk» («Агрегатор магазинов с элементами социальной сети»), предназначенной для применения пользователями в повседневной жизни для облегчения выбора товаров магазинов..

В разделе «Введение» указано наименование программы, краткое наименование программы и документы, на основании которых ведется разработка.

В разделе «Назначение и область применения» указано функциональное назначение программы, эксплуатационное назначение программы и краткая характеристика области применения программы.

В разделе «Технические характеристики» содержатся следующие подразделы:

- постановка задачи на разработку программы;
- описание алгоритма и функционирования программы с обоснованием выбора схемы алгоритма решения задачи;
- описание и обоснование выбора метода организации входных и выходных данных;
- описание и обоснование выбора состава технических и программных средств.

В разделе «Ожидаемые технико-экономические показатели» указана предполагаемая потребность и экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами

Настоящий документ разработан в соответствии с требованиями:

- 1) ГОСТ 19.101-77 Виды программ и программных документов [1];
- 2) ГОСТ 19.102-77 Стадии разработки [2];
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов [3];
- 4) ГОСТ 19.104-78 Основные надписи [4];
- 5) ГОСТ 19.105-78 Общие требования к программным документам [5];
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];
- 7) ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению [7].

Изменения к Пояснительной записке оформляются согласно ГОСТ 19.603-78 [8], ГОСТ 19.604-78 [9].

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ.....	3
1.1. Наименование программы	3
1.2. Документы, на основании которых ведется разработка	3
2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ	4
2.1. Назначение программы.....	4
2.2. Характеристика области применения.....	4
3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	5
3.1. Постановка задачи на разработку программы.....	5
3.2. Описание алгоритма и функционирования программы.....	5
3.3. Обоснование выбора архитектуры приложения	6
3.4. Описание и обоснование выбора технических и программных средств.....	6
4. ОЖИДАЕМЫЕ ТЕХНИКО-ТЕХНИЧЕСКИЕ ПОКАЗАТЕЛИ	8
4.1. Предполагаемая потребность.....	8
4.2. Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами.....	8
5. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	9
ПРИЛОЖЕНИЕ 1 ТЕРМИНОЛОГИЯ.....	10
ПРИЛОЖЕНИЕ 2 ДИАГРАММА КЛАССОВ КЛИЕНТА	11
ПРИЛОЖЕНИЕ 3 ДИАГРАММА КЛАССОВ СЕРВЕРА.....	12
ПРИЛОЖЕНИЕ 4 ПРЕДСТАВЛЕНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ.....	13
ПРИЛОЖЕНИЕ 5 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ КЛИЕНТА.....	14
ПРИЛОЖЕНИЕ 6 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ МЕТОДОВ И СВОЙСТВ КЛИЕНТА.....	15
ПРИЛОЖЕНИЕ 7 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ СЕРВЕРА.....	32
ПРИЛОЖЕНИЕ 8 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ МЕТОДОВ И СВОЙСТВ СЕРВЕРА.....	33
6 ЛИСТ ИЗМЕНЕНИЙ.....	44

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

1. ВВЕДЕНИЕ

1.1. Наименование программы

Наименование программы - «Агрегатор магазинов с элементами социальной сети»

Наименование программы на английском языке - «Providers Aggregator with Social Network Elements»

Краткое наименование программы - «ShopYou»

1.2. Документы, на основании которых ведется разработка

Разработка ведется на основании приказа Национального исследовательского университета "Высшая школа экономики" № 2.3-02/1112-04 от 11.12.2019.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

2.1. Назначение программы

2.1.1. Функциональное назначение

Программа позволяет пользователю регистрировать личный аккаунт, изменять пароль и почту, указанные при регистрации, просматривать товары на страницах магазинов, подписываться на обновления магазинов, оставлять заявку на приобретение товара. Для пользователей типа магазин программа должна позволять регистрировать аккаунт магазина, добавлять товар с комментарием, возможность отвечать на заявки других пользователей.

2.1.2. Эксплуатационное назначение

Программа объединяет множество интернет-магазинов, чтобы пользователь мог быстрее находить необходимые товары в магазинах своих любимых брендов и компаний.

2.2. Характеристика области применения

Программа предназначена для удобного взаимодействия пользователей, которые продают какие-либо товары, с пользователями, которые заинтересованы в покупке этого товара, другими словами с целевой аудиторией этого пользователя-продавца путем публикации товаров первыми, а также возможностью заказывать их вторыми.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

3.1. Постановка задачи на разработку программы

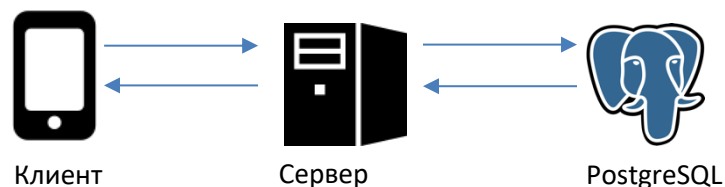
Разрабатываемая программа должна:

- 1) Иметь возможность регистрировать и авторизовывать 2 типа пользователей: пользователь-покупатель и пользователь-продавец
- 2) Пользователь-покупатель должен иметь возможность просматривать товары пользователей-продавцов, добавлять их в список понравившихся товаров и удалять их из него, создавать новые заказы на основе записей и удалять их.
- 3) Пользователь-продавец должен иметь возможность публиковать товары, просматривать заказы пользователей и изменять их статус.

3.2. Описание алгоритма и функционирования программы

3.2.1. Архитектура приложения

Приложение использует клиент-серверную архитектуру с сохранением всех данных в базу данных PostgreSQL.



3.2.2. Организация хранения данных в базе данных postgresql

Для удобного хранения данных база данных под названием «shopsagggregator» была разделена на 4 таблицы:

- 1) buyers - таблица с информацией о пользователях-покупателях. Столбцы таблицы:
 - a. id - уникальный номер пользователя-покупателя в таблице;
 - b. username - имя пользователя, которое было введено при регистрации;
 - c. password - пароль пользователя, введенный при регистрации;
 - d. iconpath - имя файла с иконкой пользователя на сервере;
 - e. subscribed - id пользователей-продавцов, на которых подписан пользователь;
 - f. likedposts - id записей, которые понравились пользователю;
 - g. newposts - id записей, которые были добавлены пользователями-продавцами, на которых подписан данный пользователь;
 - h. info - информация о пользователе, которую пользователь указал в своем аккаунте;
 - i. orders - id заказов, которые создал пользователь;
- 2) sellers - таблица с информацией о пользователях-продавцах. Столбцы таблицы:
 - a. id - уникальный номер пользователя-покупателя в таблице;
 - b. username - имя пользователя, которое было введено при регистрации;
 - c. password - пароль пользователя, введенный при регистрации;
 - d. iconpath - имя файла с иконкой пользователя на сервере;
 - e. info - информация о пользователе, которую пользователь указал в своем аккаунте;

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

- f. orders - id заказов, которые создали пользователи-покупатели;
- g. subscribers - id пользователей-покупателей, которые подписались на данного пользователя-продавца;
- h. items - id товаров, которые создал данный пользователь;
- 3) posts - таблица с информацией о записях, которые создали пользователи-продавцы. Столбцы таблицы:
 - a. id - уникальный номер записи;
 - b. info - информация о товаре;
 - c. creatorid - id пользователя-продавца, который создал запись;
 - d. likers - id пользователей-покупателей, которые добавили запись в список понравившихся записей;
- 4) orders - таблица с информацией о заказах, которые создали пользователи-покупатели. Столбцы таблицы:
 - a. id - уникальный номер заказа;
 - b. postid - id записи товара, который заказал пользователь-покупатель;
 - c. customerid - id пользователя-покупателя, который создал заказ;
 - d. issucceeded - подтвердил ли пользователь-продавец этот заказ;
 - e. iscanceled - отклонил ли пользователь-продавец этот заказ;

3.2.3. Получение данных с сервера

Для получения данных с сервера используются классы RestClient и RestResponse из сторонней библиотеки RestSharp/106.10.2-alpha.0.8. Код запроса:

```
RetsClient client = new RestClient(адрес сервера);
```

```
client.Timeout = 1000(устанавливается в зависимости от запаса);
```

```
RestRequest request = new RestRequest(метод передачи данных);
```

```
IRestResponse response = client.Execute(request); отправка запроса на сервер и  
получение результата
```

Далее происходит обработка результата

3.2.4. Обработка запросов на сервере

Запросы приходят на строго прописанную маршрутизацию и обрабатываются сервисом работы с базой данных. Для взаимодействия с postgresql используется сторонняя библиотека EntityFramework, а данные из таблиц обрабатываются при помощи LINQ.

3.3. Обоснование выбора архитектуры приложения

3.3.1. Достоинства клиент-серверной архитектуры

Клиент-серверная архитектура была выбрана из-за ряда достоинств:

- 1) Надежность хранения данных - возможность просмотреть информацию из базы данных имеет только человек, имеющий пароль от базы данных.
- 2) Отказоустойчивость - работоспособность системы полностью зависит от работы сервера и базы данных, сбой работы клиента никак не повлияет на работу программы у других пользователей.
- 3) Масштабируемость - при больших объемах данных систему можно легко улучшить, чтобы увеличить ее скорость работы

3.4. Описание и обоснование выбора технических и программных средств

3.4.1. Состав технических и программных средств

Для работы программы потребуется:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

1. Смартфон с установленной операционной системой Android версии не ниже 5 или iOS версии не ниже 13;
2. Процессор поколения не ниже Apple A11 Bionic или Qualcomm Snapdragon 439.
3. 100 мб системной памяти на устройстве;
4. Оперативная память не менее 2048 мб;
5. Стабильное подключение к сети интернет;
6. Дисплей с разрешением не ниже 1333x750;

3.4.2. Обоснование выбора технических и программных средств

Postgresql является реляционной базой данных, что позволяет хранить изначально заданные структуры данных и объединять их по какой-либо заданной логике. База данных была разделена на 4 таблицы, чтобы организовать различное отношение логически отделенных друг от друга объектов.

RestSharp является хорошо реализованной и удобной для получения и отправки запросов библиотекой, в которой нашлись все необходимые реализации для разработки.

EntityFramwork позволяет гибко взаимодействовать с базой данных.

Для разработки клиента использовался фреймворк от Microsoft Xamarin.Forms для кроссплатформенной разработки.

Для разработки сервера использовалась платформа для разработки веб приложений от Microsoft ASP.Net.

Остальные используемые в программе пространства имен имеются в реализации Microsoft .Net Core 3.1

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

4. ОЖИДАЕМЫЕ ТЕХНИКО-ТЕХНИЧЕСКИЕ ПОКАЗАТЕЛИ

4.1. Предполагаемая потребность

Возможность пользователя следить за новыми товарами, скидками и акциями своих любимых брендов. Магазины смогут более четко ориентироваться на своих потенциальных клиентов и быстрее получать от них обратную связь о товарах и заказах.

4.2. Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами

На зарубежном рынке существуют приложения «Instagram», «Facebook» обладающие подобным функционалом: <https://www.instagram.com/>, <https://www.facebook.com/>

На отечественном рынке существуют интернет ресурсы «Вконтакте», «Avito», «Lamoda», «Беру», «Юла» с подобным функционалом: <https://vk.com/>, <https://www.avito.ru/>, <https://www.lamoda.ru/>, <https://beru.ru/>, <https://youla.ru/>

Приложение обладает преимуществами:

1. Строго ориентировано на распространение и поиск товаров различных магазинов.
2. Магазины могут правильно узнавать и анализировать свою целевую аудитория.
3. Магазины могут продавать свои товары без внедрения посредников в виде интернет ресурсов.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

5. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

ГОСТ 15150-69 Машины, приборы и другие технические изделия. Исполнения для различных климатических районов. Категории, условия эксплуатации, хранения и транспортирования в части воздействия климатических факторов внешней среды. – М.: Изд-во стандартов, 1997.

ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

ГОСТ 19.401-78. ЕСПД. Текст программы. Требования к содержанию и оформлению. – М.: ИПК Издательство стандартов, 2001.

ГОСТ 19.404-79. ЕСПД. Пояснительная записка. Требования к содержанию и оформлению. – М.: ИПК Издательство стандартов, 2001.

ГОСТ 19.505-79. ЕСПД. Руководство оператора. Требования к содержанию и оформлению. . – М.: ИПК Издательство стандартов, 2001.

Документация по .Net [Электронный ресурс]// URL: <https://docs.microsoft.com/ru-ru/dotnet/> (Дата обращения: 13.05.2020, режим доступа: свободный)

Документация по Xamarin [Электронный ресурс]// URL: <https://docs.microsoft.com/ru-ru/xamarin/> (Дата обращения: 13.05.2020, режим доступа: свободный)

Документация по Asp.Net [Электронный ресурс]// URL: <https://docs.microsoft.com/ru-ru/aspnet/> (Дата обращения: 13.05.2020, режим доступа: свободный)

Документация по Postgresql [Электронный ресурс]// URL: <https://postgresql.org/docs/> (Дата обращения: 13.05.2020, режим доступа: свободный)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 1 ТЕРМИНОЛОГИЯ

Http - протокол передачи данных.

Авторизация - предоставление определённому лицу или группе лиц определённых прав.

Архитектура программы - организация взаимодействия программных модулей.

База данных - организованная структура, предназначенная для хранения, изменения и обработки взаимосвязанной информации

Пользователь - человек, который использует данную программу.

Пользователь-покупатель - тип пользователя, который имеет возможность публиковать записи и взаимодействовать с заказами пользователей-покупателей.

Пользователь-продавец - тип пользователя, который имеет возможность взаимодействовать с записями и аккаунтом пользователя-покупателя.

Сервер - компонент вычислительной системы, выполняющий функции по запросу клиента, предоставляя ему доступ к определённым ресурсам.

Клиент - программа, работающая на устройстве пользователя и позволяющая получать и отправлять информацию на сервера и предоставлять ее пользователю в интерактивном виде.

Учетная запись – данные о пользователе, по которым можно его идентифицировать и предоставить доступ к личной информации и возможностям в используемом сервисе.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

The diagram illustrates the architecture of a mobile application, showing various classes and their relationships. The classes are organized into a hierarchy, with some classes inheriting from others. The classes and their methods are as follows:

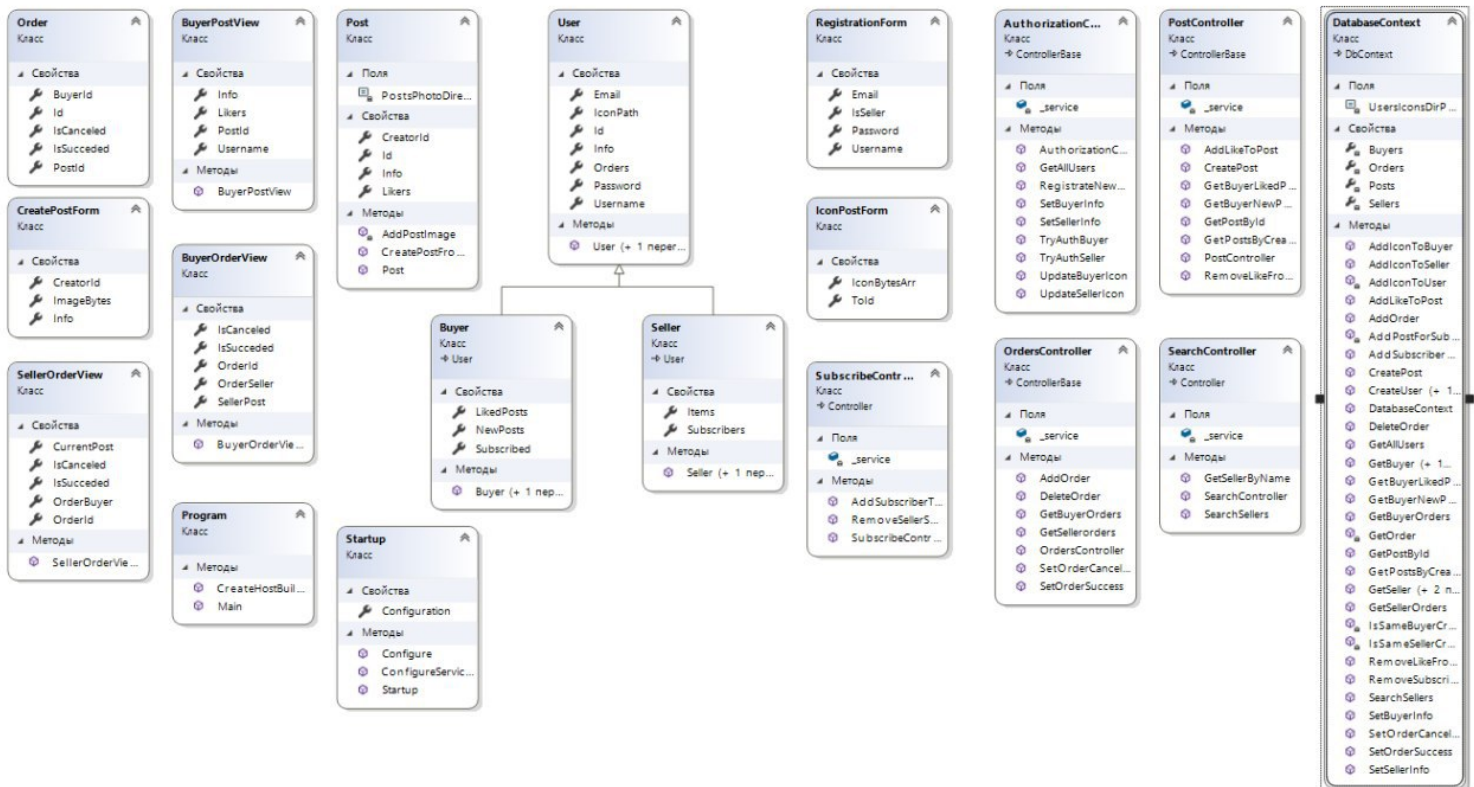
- RegistrationForm** (Klacc):
 - Методы: Email, IsSeller, Password, Username
- SellerOrderPage** (Klacc):
 - Методы: ContentPage
- Authentication** (Klacc):
 - Методы: ContentPage
- SellerOrderView** (Klacc):
 - Методы: ContentPage
- User** (Klacc):
 - Методы: ContentPage
- BuyerMainPage** (Klacc):
 - Методы: ContentPage
- RegistrationPage** (Klacc):
 - Методы: ContentPage
- SellerMainPage** (Klacc):
 - Методы: ContentPage
- AllPostsPage** (Klacc):
 - Методы: ContentPage
- IconPostForm** (Klacc):
 - Методы: ContentPage
- SearchPage** (Klacc):
 - Методы: ContentPage
- CreatePostForm** (Klacc):
 - Методы: ContentPage
- BuyerPostView** (Klacc):
 - Методы: ContentPage
- WatchSellerPage** (Klacc):
 - Методы: ContentPage
- CreatePostPage** (Klacc):
 - Методы: ContentPage

The diagram also shows relationships between these classes, including inheritance (indicated by solid lines with hollow triangle heads) and associations (indicated by solid lines). For example, the **User** class is associated with **Seller** and **Buyer** classes. The **Seller** class is associated with **Buyer** and **Post** classes. The **Buyer** class is associated with **Post** and **PostItem** classes. The **Post** class is associated with **PostItem** and **PostList** classes. The **PostItem** class is associated with **PostList** and **PostDetail** classes. The **PostList** class is associated with **PostDetail** and **PostForm** classes. The **PostDetail** class is associated with **PostForm** and **PostView** classes. The **PostForm** class is associated with **PostView** and **PostEdit** classes. The **PostView** class is associated with **PostEdit** and **PostDelete** classes. The **PostEdit** class is associated with **PostDelete** and **PostCreate** classes. The **PostDelete** class is associated with **PostCreate** and **PostUpdate** classes. The **PostCreate** class is associated with **PostUpdate** and **PostCancel** classes. The **PostUpdate** class is associated with **PostCancel** and **PostConfirm** classes. The **PostCancel** class is associated with **PostConfirm** and **PostReject** classes. The **PostConfirm** class is associated with **PostReject** and **PostAccept** classes. The **PostReject** class is associated with **PostAccept** and **PostDeny** classes. The **PostAccept** class is associated with **PostDeny** and **PostApprove** classes. The **PostDeny** class is associated with **PostApprove** and **PostDisapprove** classes. The **PostApprove** class is associated with **PostDisapprove** and **PostRevoke** classes. The **PostDisapprove** class is associated with **PostRevoke** and **PostSuspend** classes. The **PostRevoke** class is associated with **PostSuspend** and **PostUnsuspend** classes. The **PostSuspend** class is associated with **PostUnsuspend** and **PostReactivate** classes. The **PostUnsuspend** class is associated with **PostReactivate** and **PostDeactivate** classes. The **PostReactivate** class is associated with **PostDeactivate** and **PostArchive** classes. The **PostDeactivate** class is associated with **PostArchive** and **PostUnarchive** classes. The **PostArchive** class is associated with **PostUnarchive** and **PostRestore** classes. The **PostUnarchive** class is associated with **PostRestore** and **PostRecover** classes. The **PostRestore** class is associated with **PostRecover** and **PostRebuild** classes. The **PostRecover** class is associated with **PostRebuild** and **PostRecreate** classes. The **PostRebuild** class is associated with **PostRecreate** and **PostReinitialize** classes. The **PostRecreate** class is associated with **PostReinitialize** and **PostReformat** classes. The **PostReinitialize** class is associated with **PostReformat** and **PostReinstall** classes. The **PostReformat** class is associated with **PostReinstall** and **PostReconfigure** classes. The **PostReinstall** class is associated with **PostReconfigure** and **PostReoptimize** classes. The **PostReconfigure** class is associated with **PostReoptimize** and **PostRecompress** classes. The **PostReoptimize** class is associated with **PostRecompress** and **PostReencrypt** classes. The **PostRecompress** class is associated with **PostReencrypt** and **PostReauthenticate** classes. The **PostReencrypt** class is associated with **PostReauthenticate** and **PostReauthorize** classes. The **PostReauthenticate** class is associated with **PostReauthorize** and **PostRevalidate** classes. The **PostReauthorize** class is associated with **PostRevalidate** and **PostReverify** classes. The **PostRevalidate** class is associated with **PostReverify** and **PostReconfirm** classes. The **PostReverify** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept** class is associated with **PostReconfirm** and **PostReapprove** classes. The **PostReconfirm** class is associated with **PostReapprove** and **PostReaccept** classes. The **PostReapprove** class is associated with **PostReaccept** and **PostReconfirm** classes. The **PostReaccept</**

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 3

ДИАГРАММА КЛАССОВ СЕРВЕРА



Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 4 ПРЕДСТАВЛЕНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ

Название таблицы	Представление таблицы
buyers	 <pre> id integer = nextval('buyers_id. username varchar(50) email varchar(350) password varchar(50) iconpath varchar(100) subscribed integer[] likedposts integer[] newposts integer[] info text orders integer[] </pre>
sellers	 <pre> id integer = nextval('sellers_i...) username varchar(50) email varchar(350) password varchar(100) iconpath varchar(100) subscribers integer[] items integer[] info text orders integer[] </pre>
posts	 <pre> id integer = nextval('posts_id_...) info text creatorid integer likers integer[] </pre>
orders	 <pre> id integer = nextval('orders_id...) postid integer customerid integer issucceded boolean iscanceled boolean </pre>

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 5 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ КЛИЕНТА

Описание и функциональное назначение классов

Класс	Назначение
User	Общий тип пользователей.
Buyer	Тип пользователя-покупателя.
Seller	Тип пользователя-продавца.
BuyerOrderView	Модель данных для заказа пользователя-покупателя.
BuyerPostView	Модель данных записи для пользователя-покупателя.
CreatePostForm	Модель для создания записи, которая отправляется на сервер.
IconPostForm	Экземпляр типа установки иконки пользователю.
Post	Тип записи, которую создает пользователь-продавец
RegistrationForm	Форма для регистрации пользователя в сервисе
SellerOrderView	Модель заказа для пользователя-покупателя.
AllPostsPage	Код для страницы всех записей.
Authentification	Код страницы авторизации пользователя.
BuyerMainPage	Код страницы пользователя-покупателя.
BuyerOrdersPage	Код страницы заказов пользователя.
CreatePostPage	Код страницы создания записи.
RegistrationPage	Код страницы регистрации пользователя.
SearchPage	Код страницы поиска пользователей-продавцов.
SellerOrdersPage	Код страницы заказы пользователя-продавца.
WatchSellerPage	Код страницы просмотра страницы пользователя-продавца
App	Основная информация о приложении хранится в этом классе.
ImageResourceExtension	Расширение функционала Image view.
MainTabbedPage	Код для страницы с множеством вкладок.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 6 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ МЕТОДОВ И СВОЙСТВ КЛИЕНТА

Описание полей методов и свойств класса User.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
iconPath	private	String	Поле значения иконки пользователя.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
User	Public	Конструктор	-	Пустой конструктор.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Id	public	Int32	get;set;	Id пользователя.
Username	public	String	get;set;	Имя пользователя.
Email	public	String	get;set;	Email пользователя.
Password	public	String	get;set;	Пароль пользователя.
Info	public	String	get;set;	Информация о пользователе.
IconPath	public	String	get;set;	Устанавливает имя иконки пользователя на сервер и возвращает ссылку на него.

Описание полей, методов и свойств класса Buyer.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Buyer	public	Конструктор	-	Пустой конструктор.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
LikedPosts	public	List<Int32>	get;set	Id понравившихся записей в базе данных.
Subscribed	public	List<Int32>	get;set;	Id пользователей-продавцов, на которых подписан пользователь.
NewPosts	public	List<Int32>	get;set;	Id записей, которые добавили пользователи-продавцы, на которых подписан данный пользователь.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Описание полей, методов и свойств класса Seller.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Seller	public	конструктор	-	Пустой конструктор.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Items	public	List<Int32>	get;set;	Id записей, созданных пользователем.
Subscribers	public	List<Int32>	get;set;	Id подписчиков, которые подписались на данного пользователя.

Описание полей, методов и свойств класса BuyerOrderView.cs

Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
OrderId	public	Int32	get;set;	Id заказа.
OrderSeller	public	Seller	get;set;	Экземпляр пользователя-покупателя, который создал запись.
SellerPost	public	Post	get;set;	Запись, которую опубликовал пользователь-продавец.
IsSucceded	public	Boolean	get;set;	Одобрил ли пользователь-продавец заказ.
IsCanceled	public	Boolean	get;set;	Отклонил ли пользователь-продавец заказ.
OrderStatus	public	String	get;	Возвращает текст со статусом заказа.

Описание полей, методов и свойств класса BuyerPostView.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
BuyerPostView	public	Конструктор с параметрами	Post post	Конструктор модели данных в зависимости от экземпляра записи.
BuyerPostView	public	Конструктор	-	Пустой конструктор.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
BuyerId	public	Int32	get;set;	Id пользователя-покупателя.
Username	public	String	get;set;	Имя пользователя-продавца, который опубликовал запись.
PostId	public	Int32	get;set;	Id записи.
Info	public	String	get;set;	Информация о записи.
IsUserLikePost	public	Boolean	get;	Понравилась ли запись данному пользователю-покупателю.
Likers	public	List<Int32>	get;set;	Пользователи-покупатели, которым понравилась запись.
IconPath	public	String	get;	Ссылка на фотографию к записи.
SellerIconPath	public	String	get;	Ссылка на иконку пользователя-продавца, опубликовавшего эту запись.
IsUserDontLikePost	public	Boolean	get;	Не ставил ли пользователь-покупатель лайк.

Описание полей, методов и свойств класса CreatePostForm.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
CreatePostForm	public	Конструктор с параметрами	Int32 id	Создает экземпляр с id пользователя-продавца.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
CreatorId	public	Int32	get;set;	Id пользователя-продавца, который создает запись.
Info	public	String	get;set;	Информация о записи.
ImageBytes	public	List<Int32>	get;set;	Массив байтов картинки записи.

Описание полей, методов и свойств класса IconPostForm.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
IconPostForm	public	Конструктор	-	Пустой конструктор.
IconPostForm	public	Конструктор с параметрами	Int32 to id	Конструктор, который создает экземпляр с id пользователя.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
IconBytesArr	public	List<Int32>	get;set;	Байты фотографии, выбранной пользователем.
ToId	public	Int32	get;set;	Id пользователя, который изменяет иконку.

Описание полей, методов и свойств класса Post.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Поля				
Имя	Модификатор доступа	Тип	Назначение	
info	private	String	Строка информации о записи.	
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Id	public	Int32	get;set;	Id записи.
CreatorId	public	Int32	get;set;	Id создателя записи - пользователя-продавца.
Info	public	String	get;set;	Устанавливает значение полю _info и в зависимости от информации в этом поле возвращает данные о записи.
ImagePath	public	String	get;	Возвращает ссылку на картинку к записи на сервере.
Likers	public	List<Int32>	get;set;	Id пользователей-покупателей, которым понравилась эта запись.
LikesCount	public	Int32	get;	Количество пользователей-покупателей, которым понравилась эта запись.

Описание полей, методов и свойств класса RegistrationForm.cs

Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Username	public	String	get;set;	Имя пользователя.
Email	public	String	get;set;	Email пользователя.
Password	public	String	get;set;	Пароль пользователя.
IsSeller	public	Boolean	get;set;	Является ли пользователь пользователем-продавцом.

Описание полей, методов и свойств класса SellerOrderView.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
OrderId	public	Int32	get;set;	Id заказа.
OrderBuyer	public	Buyer	get;set;	Экземпляр типа пользователя, который сделал заказ.
CurrentPost	public	Post	get;set;	Экземпляр типа записи, которую заказал пользователь.
IsSucceded	public	Boolean	get;set;	Одобрил ли пользователь-продавец заказ.
IsCanceled	public	Boolean	get;set;	Отклонил ли пользователь-продавец заказ.
OrderStatus	public	String	get;	Возвращает текст со статусом заказа.

Описание полей, методов и свойств класса AllPotsPage.xaml.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Поля				
Имя	Модификатор доступа	Тип	Назначение	
_buyer	private	Buyer	Экземпляр типа пользователя-покупателя.	
newPosts	private	List<BuyerPostView>	Новые записи, доступные пользователю-покупателю.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
AllPostsPage	public	Конструктор параметрами	Buyer buyer	Конструктор страницы.
RefreshViewOnRefreshing	private	void	object sender, EventArgs e	Обновляет данные страницы по запросу пользователя.
OnAppearing	protected	void	-	Получает данные о новых записях и вызывает базовое поведение метода.
GetNewPostsAsync	private	void	-	Получает новые записи, доступные пользователю.
OnLikeImageTapped	private	void	object sender, EventArgs e	Событие нажатия пользователем на картинку "понравилась запись(лайк)".
OnDislikeImageTapped	private	void	object sender, EventArgs e	Событие нажатия пользователем на картинку "убрать лайк".
SendLikeTapped	private	void	String param, BuyerPostView post	Отправляет на сервер PUT запрос, чтобы добавить в или убрать картинку из понравившееся у пользователя.
Posts_OnItemTapped	private	void	object sender, ItemTappdeEventArgs e	Отменяет событие нажатия на элемент из ListView.
OnPostHeaderTapped	private	void	object sender, EventArfs e	Обрабатывает событие нажатия

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

				на иконку или имя пользователя-продавца, который опубликовал запись.
DeliverButton_OnClicked	private	void	object sender, EventArgs e	Событие добавления пользователем записи к заказам.

Описание полей, методов и свойств класса Authentication.xaml.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
IncorrectUsernameInputErrorMessage	private	String	Текст сообщения о том, что поле имени пользователя пустое.	
IncorrectPasswordInputErrorMessage	private	String	Текст сообщения о том, что поле пароля пустое.	
BadSignInTryAlertTitle	private	String	Заголовок сообщения о неудачном входе.	
BadSignInTryAlertCancelText	private	String	Текст сообщения о неудачном входе.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Authentication	public	Конструктор	-	Конструктор страницы.
OnAppearing	protected	void	-	Устанавливает цвет границ кнопок авторизации и вызывает базовое поведение метода.
OnBuyerSignInButtonClicked	private	void	object sender, EventArgs e	Событие попытки входа в аккаунт пользователя-покупателя.
OnSellerSignInButtonClicked	private	void	object sender, EventArgs e	Событие попытки входа в аккаунт

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

				пользователя-продавца.
CheckIsBoxesAndConnectionCorrect	private	Boolean	-	Проверяет есть ли соединение с интернетом и все ли поля заполнены верно.
TrySignIn	private	T	String login, String password, String authType	Отправляет запрос на сервер для входа в аккаунт
MessageAboutIncorrectBox	private	void	Entry view, String errorMsg	Вызывает анимацию встряски View элемента и выводит сообщение о том что поле заполнено неверно.
OnEntryTextChanged	private	void	object sender, TextChangedEventArgs e	Убирает сообщение о статусе входа при изменении полей входа.
OnRegistrationButtonClick	private	void	object sender, EventArgs e	Перенаправляет пользователя на страницу регистрации.

Описание полей, методов и свойств класса BuyerMainPage.xaml.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
iconPath	private	String	Ссылка на иконку пользователя.	
_buyer	private	Buyer	Экземпляр типа пользователя-покупателя.	
FailedConnectionToServerAlertTitle	private	String	Заголовок сообщения об ошибке подключения к серверу.	
likedPosts	private	List<Int32>	Список записей, которые понравились пользователю.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
BuyerMainPage	public	Конструктор	Buyer buyer	Конструктор страницы.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

OnAppearing	protected	void	-	Устанавливает ширину ListView Posts, получает понравившиеся записи и вызывает базовое поведение метода.
GetBuyerLikedPosts	private	void	-	Получает с сервера записи, которые понравились пользователю.
SendUpdateUserPut	private	void	IconPostForm form	Отправляет запрос с обновлением фотографии пользователя.
GetPhoto	private	void	object sender, EventArgs e	При нажатии пользователя на иконку предоставляет пользователю выбрать фотографию.
GetPhotoBytes	private	void	String path, IconPostForm form	Получает байты фотографии, которую выбрал пользователь.
RefreshView_OnRefreshing	private	void	object sender, EventArgs e	Обновляет информацию о пользователе.
OnExitToolbarItemTapped	private	void	object sender, EventArgs e	Обработчик события нажатия на кнопку выхода.
OnPostHeaderTapped	private	void	object sender, EventArgs e	Открывает страницу пользователя, который опубликовал запись.
SendDislikeImageTapped	private	void	object sender, EventArgs e	Отправляет запрос на сервер, чтобы убрать запись из понравившегося.
VisualElement_OnUnfocused	private	void	object sender, FocusEventArgs e	Отправляет запрос, чтобы изменить информацию о пользователе.

Описание полей, методов и свойств класса BuyerOrdersPage.xaml.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Поля				
Имя	Модификатор доступа	Тип	Назначение	
buyer	private	Buyer	Экземпляр типа пользователя.	
buyerOrders	private	List<BuyerOrderView>	Список заказов пользователя.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
BuyerOrdersPage	public	Конструктор	Buyer buyer	Конструктор страницы.
OnAppearing	protected	void	-	Получает заказы пользователя и вызывает базовое поведение метода.
OnPostHeaderTapped	private	void	object sender, EventArgs e	Открывает страницу пользователя, который опубликовал запись.
OnOrdersUpdate	private	void	object sender, EventArgs e	Событие обновления заказов.
UpdateOrdersAsync	private	void	-	Отправляет на сервер запрос, чтобы получить заказы пользователя.
OnOrdersItemTapped	private	void	object sender, ItemTappedEventArgs e	Отменяет нажатие на элемент ListView.
DeleteButtonClickedAsync	private	void	object sender, EventArgs e	Обработчик события удаления заказа.

Описание полей, методов и свойств класса CreatePostPage.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
seller	private	Seller	Экземпляр типа пользователя-покупателя.	
form	private	CreatePostForm	Форма создания записи.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
CreatePostPage	public	Конструктор	Seller seller	Конструктор страницы.
OnGetPostPhotoClicked	private	void	object sender, EventArgs e	Позволяет пользователю выбрать фотографию после нажатия на кнопку.
GetImageBytesFromPath	private	void	String path, CreatePostForm form	Получает байты изображения, выбранного пользователем.
CreatePostButtonClicked	private	void	object sender, EventArgs e	Обработчик события нажатия на кнопку создания записи.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Описание полей, методов и свойств класса RegistrationPage.xaml.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
IncorrectEmailMessage	private	String	Текст сообщения о неверном Email.	
IncorrectUsernameMessage	private	String	Текст сообщения о неверном Username.	
IncorrectPasswordMessage	private	String	Текст сообщения о неверном пароле.	
IncorrectPasswordAgainMessage	private	String	Текст сообщения о неверном повторном пароле.	
BadAccountCreateAlertTitle	private	String	Заголовок сообщения о неудачном создании аккаунта.	
BadAccountCreateAlertCancelText	private	String	Текст кнопки сообщения о неудачном создании аккаунта.	
BadRequestAlertMessage	private	String	Текст сообщения о неудачном создании аккаунта.	
SuccessAccountCreateTitle	private	String	Текст кнопки сообщения об удачном создании аккаунта.	
SuccessAccountCreateMessage	private	String	Текст сообщения об удачном создании аккаунта.	
SuccessAccountCreateCancel	private	String	Заголовок сообщения об удачном создании аккаунта.	
form	private	RegistrationForm	Модель регистрации пользователя.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
RegistrationPage	public	Конструктор	-	Конструктор страницы.
OnRegistrationButtonCLicked	private	void	object sender, EventArgs e	Проверяет все ли поля заполнены верно и вызывает метод отправки формы на сервер.
SendRegistrationPost	private	Boolean	RegistrationForm user	Отправляет на сервер запрос создания аккаунта.
CheckBoxChanged	private	void	object sender, EventArgs e	Изменяет цвет CheckBox и его текста при изменении полей ввода данных.
IsBoxesCorrect	private	Boolean	-	Проверяет все ли поля для заполнения заполнены верно.
MessageAboutIncorrectBox	private	void	Entry view, String errorMsg	Встряхивает неверное поле с

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

				данными и выводит соответствующее сообщение.
OnEntryTextChanged	private	void	object sender, TextChangedEventArgs	Событие об изменении Entry.
IsTextboxCorrect	private	Boolean	Entry box, Func<Char, Boolean> func	Проверяет введенные данные.
IsTextboxCorrect	private	Boolean	Entry box, Func<String, Boolean> func	Проверяет введенные данные.
ValidateEmail	private	Boolean	String email	Проверяет, является ли введенный email верным.
ValidatePassword	private	Boolean	String input	Проверяет является ли введенный пароль допустимым.

Описание полей, методов и свойств класса SearchPage.xaml.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
connectionAlertTitle	private	String	Заголовок сообщения о неудачном запросе к серверу.	
connectionAlertContent	private	String	Текст сообщения об отсутствии интернета.	
alertCancel	private	String	Текст кнопки сообщения.	
getResponseAlertTitle	private	String	Заголовок сообщения о неудачном запросе к серверу.	
getResponseAlertContent	private	String	Текст сообщения о неудачном получении данных.	
searchedSellers;	private	List<String>	Список найденных пользователей.	
buyer	private	Buyer	Экземпляр типа пользователя-покупателя.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SearchPage	public	Конструктор	Buyer buyer	Конструктор страницы.
OnSearchButtonPressed	private	void	object sender, EventArgs e	Обработчик события поиска пользователя по имени.
Search	private	void	String searchText	Вызывает метод с запросом на сервер, получает результат работы сервера и обрабатывает их.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

TryGetSearchResult	private	Task<String>	String searchLine	Отправляет запрос на сервер для поиска пользователя.
OnSearchItemSelected	private	void	object sender, SelectedItem ChangedEventArgs e	Обработчик события нажатия на элемент списка найденных пользователей.

Описание полей, методов и свойств класса SellerOrdersPage.xaml.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
_seller	private	Seller	Экземпляр типа пользователя-продавца.	
sellerOrders	private	List<SellerOrderView>	Список заказов пользователя-продавца.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SellerOrdersPage	public	Конструктор	Seller seller	Конструктор страницы.
OnAppearing	protected	void	-	Вызывает метод получения заказов пользователя и вызывает базовое поведение метода.
OnOrdersUpdate	private	void	object sender, EventArgs e	Обработчик события обновления заказов.
UpdateOrdersAsync	private	void	-	Обновляет заказы пользователя.
OnOrdersItemTapped	private	void	object sender, ItemTappedEventArgs e	Отменяет событие нажатия на элемент из ListView.
SetOrderSuccess	private	void	object sender, EventArgs e	Ставит статус заказа одобренным и отправляет запрос на сервер.
SetOrderCanceledStatus	private	void	object sender, EventArgs e	Ставит статус заказа отклоненным и отправляет запрос на сервер.
SendOrderStatusAsync	private	async	String reqStatus, Int32 orderId	Отправляет запрос на сервер чтобы обновить статус заказа.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Описание полей, методов и свойств класса WatchSellerPage.xaml.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
_seller	private	Seller	Экземпляр типа пользователя-продавца.	
_sellerName	private	String	Имя пользователя-продавца.	
_sellerPosts	private	List<BuyerPostView>	Список записей пользователя-продавца.	
_buyer	private	Buyer	Экземпляр типа пользователя-покупателя.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
WatchSellerPage	public	Конструктор	String sellerName, Buyer buyer	Конструктор страницы. Получает экземпляр пользователя-продавца с сервера.
SetSubscribeButton	private	void	-	Устанавливает кнопке статус подписки.
SetUnsubscribeButton	private	void	-	Устанавливает кнопке статус отписки.
SubscribeToSeller	private	void	object sender, EventArgs e	Обработчик события подписки на пользователя-продавца.
UnsubscribeFromSeller	private	void	object sender, EventArgs e	Обработчик события отписки от пользователя-продавца.
RefreshViewOnRefreshing	private	void	object sender, EventArgs e	Обработчик события обновления информации о пользователе-продавце.
GetSellerByNameAsync	private	void	String sellerName	Получает с сервера пользователя-продавца по имени.
GetSellerPostsAsync	private	void	String sellerName	Получает записи пользователя-продавца с сервера.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Posts_OnItemSelected	private	void	object sender, SelectedItem ChangedEventArgs e	Отменяет событие нажатия на элемент из ListView.
OnLikeImageTapped	private	void	object sender, EventArgs e	Событие нажатия пользователем на картинку "понравилась запись(лайк)".
OnDislikeImageTapped	private	void	object sender, EventArgs e	Событие нажатия пользователем на картинку "убрать лайк".
SendLikeTapped	private	void	string param, BuyerPostView post	Отправляет на сервер PUT запрос, чтобы добавить в или убрать картинку из понравившееся у пользователя.

Описание полей, методов и свойств класса App.xaml.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
ServerUrl	public	String	Ссылка сервера на контроллер users.	
BaseUrl	public	String	Базовый адрес сервера.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
App	public	Конструктор	-	Конструктор. Устанавливает язык приложения и интерфейса.
Shake	public	void	View element	Анимировывает встряхивание элемента.
IsConnected	public	Boolean	-	Проверяет, есть ли у приложения подключение к интернету.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Описание полей, методов и свойств класса ImageResourceExtension.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
ProvideValue	public	object	IServiceProvider serviceProvider	Получает значения из view и возвращает изображение.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Source	public	String	get;set;	Путь к файлу с изображением.

Описание полей, методов и свойств класса MainTabbedPage.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
MainTabbedPage	public	Конструктор	-	Общий конструктор. Устанавливает заголовок страницы.
MainTabbedPage	public	Конструктор	Buyer buyer	Добавляет страницы, если пользователем является пользователь-покупатель.
MainTabbedPage	public	Конструктор	Seller seller	Добавляет страницы, если пользователем является пользователь-продавец.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 7
ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ СЕРВЕРА

Класс	Назначение
User	Общий тип пользователей.
Buyer	Тип пользователя-покупателя.
Seller	Тип пользователя-продавца
BuyerOrderView	Модель данных для заказа пользователя-покупателя.
BuyerPostView	Модель данных записи для пользователя-покупателя.
CreatePostForm	Модель для создания записи, которая отправляется на сервер.
IconPostForm	Модель типа установки иконки пользователю.
Orders	Модель типа заказ.
Post	Тип записи, которую создает пользователь-продавец
RegistrationForm	Форма для регистрации пользователя в сервисе.
SellerOrderView	Модель заказа для пользователя-покупателя.
AuthorizationContoller	Контроллер взаимодействия с пользователями.
OrdersController	Контроллер взаимодействия с заказами.
PostController	Контроллер взаимодействия с записями.
SearchController	Контроллер взаимодействия с поиском.
SubscribeController	Контроллер взаимодействия с подписками.
DatabaseContext	Сервис работы с базой данных.
Program	Класс с Main.
Startup	Класс создания сервиса.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 8 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ МЕТОДОВ И СВОЙСТВ СЕРВЕРА

Описание полей, методов и свойств класса User.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
User	public	Конструктор	-	Пустой конструктор.
User	public	Конструктор	RegistrationForm data data	Конструктор от формы создания пользователя.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Id	public	Int32	get;set;	Id пользователя.
Username	public	String	get;set;	Имя пользователя.
Email	public	String	get;set;	Email пользователя.
Password	public	String	get;set;	Пароль пользователя.
Info	public	String	get;set;	Информация о пользователе.
IconPath	public	String	get;set;	Устанавливает имя иконки пользователя на сервер и возвращает ссылку на него.
Orders	public	List<Int32>	get;set;	Id заказов пользователя.

Описание полей, методов и свойств класса Buyer.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Buyer	public	Конструктор	-	Пустой конструктор.
Buyer	public	Конструктор	RegistrationForm data	Конструктор от формы создания пользователя.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
LikedPosts	public	List<Int32>	get;set	Id понравившихся записей в базе данных.
Subscribed	public	List<Int32>	get;set;	Id пользователей-продавцов, на которых подписан пользователь.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

NewPosts	public	List<Int32>	get;set;	Id записей, которые добавили пользователи-продавцы, на которых подписан данный пользователь.
----------	--------	-------------	----------	--

Описание полей, методов и свойств класса Seller.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Seller	public	конструктор	-	Пустой конструктор.
Seller	public	Конструктор	RegistrationForm data	Конструктор от формы создания пользователя.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Items	public	List<Int32>	get;set;	Id записей, созданных пользователем.
Subscribers	public	List<Int32>	get;set;	Id подписчиков, которые подписались на данного пользователя.

Описание полей, методов и свойств класса BuyerOrderView.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
BuyerOrderView	public	Конструктор	-	Пустой конструктор.
BuyerOrderView	public	Конструктор	Order order, Post post, Seller seller	Конструктор от записи, заказа и пользователя-продавца.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
OrderId	public	Int32	get;set;	Id заказа.
OrderSeller	public	Seller	get;set;	Экземпляр пользователя-покупателя, который создал запись.
SellerPost	public	Post	get;set;	Запись, которую опубликовал пользователь-продавец.
IsSucceded	public	Boolean	get;set;	Одобрил ли пользователь-продавец заказ.
IsCanceled	public	Boolean	get;set;	Отклонил ли пользователь-продавец заказ.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Описание полей, методов и свойств класса BuyerPostView.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
BuyerPostView	public	Конструктор с параметрами	Post post, Seller creator	Конструктор от записи и пользователя-продавца.
BuyerPostView	public	Конструктор	-	Пустой конструктор.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Username	public	String	get;set;	Имя пользователя-продавца, который опубликовал запись.
PostId	public	Int32	get;set;	Id записи.
Info	public	String	get;set;	Информация о записи.
Likers	public	List<Int32>	get;set;	Пользователи-покупатели, которым понравилась запись.

Описание полей, методов и свойств класса CreatePostForm.cs

Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
CreatorId	public	Int32	get;set;	Id пользователя-продавца, который создает запись.
Info	public	String	get;set;	Информация о записи.
ImageBytes	public	List<Int32>	get;set;	Массив байтов картинки записи.

Описание полей, методов и свойств класса IconPostForm.cs

Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
IconBytesArr	public	List<Int32>	get;set;	Байты фотографии, выбранной пользователем.
ToId	public	Int32	get;set;	Id пользователя, который изменяет иконку.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Описание полей, методов и свойств класса Order.cs

Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Id	public	Int32	get;set;	Id заказа.
PostId	public	Int32	get;set;	Id записи, на которую создали запись.
BuyerId	public	Int32	get;set;	Id пользователя-покупателя, который создал заказ.
IsSucceded	public	Boolean	get;set;	Подтвердил ли пользователь-продавец заказ.
IsCanceled	public	Boolean	get;set;	Отклонил ли пользователь-продавец заказ.

Описание полей, методов и свойств класса Post.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
PostsPhotoDirectory	private	String	Путь для сохранения картинок.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Post	public	Конструктор	-	Пустой конструктор.
CreatePostFromForm	public	void	CreatePostForm form	Ставит полям значения из формы и вызывает метод добавления фотографии.
AddPostImage	private	void	CreatePostForm form	Добавляет фотографию для записи из формы.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Id	public	Int32	get;set;	Id записи.
CreatorId	public	Int32	get;set;	Id создателя записи - пользователя-продавца.
Info	public	String	get;set;	Информация о записи
Likers	public	List<Int32>	get;set;	Id пользователей-покупателей, которым понравилась эта запись.

Описание полей, методов и свойств класса RegistrationForm.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Username	public	String	get;set;	Имя пользователя.
Email	public	String	get;set;	Email пользователя.
Password	public	String	get;set;	Пароль пользователя.
IsSeller	public	Boolean	get;set;	Является ли пользователь пользователем-продавцом.

Описание полей, методов и свойств класса SellerOrderView.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SellerOrderView	public	Конструктор	-	Пустой конструктор.
SellerOrderView	public	Конструктор	Order order, Post post, Buyer buyer	Конструктор от заказа, записи и пользователя-покупателя.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
OrderId	public	Int32	get;set;	Id заказа.
OrderBuyer	public	Buyer	get;set;	Экземпляр типа пользователя, который сделал заказ.
CurrentPost	public	Post	get;set;	Экземпляр типа записи, которую заказал пользователь.
IsSucceded	public	Boolean	get;set;	Одобрил ли пользователь-продавец заказ.
IsCanceled	public	Boolean	get;set;	Отклонил ли пользователь-продавец заказ.

Описание полей, методов и свойств класса AuthorizationController.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
context	private	DatabaseContext	Сервис базы данных.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
AuthorizationContoller	public	Конструктор	DatabaseContext service	Конструктор контроллера.
GetAllUsers	public	ActionResult<List<User>>	-	Метод получения всех пользователей.
RegistrateNewUser	public	ActionResult	RegistrationForm user	Регистрирует нового пользователя.
TryAuthBuyer	public	ActionResult<User>	String login, String password	Аутентификация пользователя-покупателя.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

TryAuthSeller	public	ActionResult<User>	String login, String password	Аутентификация пользователя- продавца.
UpdateSellerIcon	public	IActionResult	IconPostForm icon	Добавление иконки пользователю- продавцу.
UpdateBuyerIcon	public	IActionResult	IconPostForm icon	Добавление иконки пользователю- покупателю.
SetBuyerInfo	public	IActionResult	Int32 buyerId, String info	Обновляет информацию пользователя- покупателя.
SetSellerInfo	public	IActionResult	Int32 buyerId, String info	Обновляют информацию пользователя- продавца.

Описание полей, методов и свойств класса OrdersController.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
context	private	DatabaseContext	Сервис базы данных.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
OrdersController	public	Конструктор	DatabaseContext service	Конструктор контроллера.
AddOrder	public	IActionResult	Int32 buyerId, Int32 postId	Добавляет новую запись.
GetBuyerOrders	public	List<BuyerOrderView>	Int32 buyerId	Получает список заказов пользователя-покупателя.
GetSellerorders	public	List<SellerOrderView>	Int32 sellerId	Получает список заказов пользователя-продавца.
SetOrderSuccess	public	IActionResult	Int32 orderId	Ставит одобренный статус заказу.
SetOrderCanceled	public	IActionResult	Int32 orderId	Ставит отклоненный статус заказу.
DeleteOrder	public	IActionResult	Int32 orderId	Удаляет заказ.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Описание полей, методов и свойств класса PostController.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
service	private	DatabaseContext	Сервис базы данных.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
PostController	public	Конструктор	DatabaseContext service	Конструктор контроллера.
CreatePost	public	ActionResult	CreatePostForm postForm	Создает новую запись.
GetPostById	public	Post	Int32 id	Получает запись по id.
GetPostsByCreatorId	public	List<Post>	Int32 sellerName	Получает записи, опубликованные пользователем-продавцом.
GetBuyerNewPosts	public	List<BuyerPostView>	Int32 buyerId	Получает новые записи пользователя-покупателя.
AddLikeToPost	public	ActionResult	Int32 likerId, Int32 postId	Добавляет лайк пользователя-покупателя к записи.
RemoveLikeFromPost	public	ActionResult	Int32 likerId, Int32 postId	Удаляет лайк пользователя-покупателя у записи.
GetBuyerLikedPosts	public	List<BuyerPostView>	Int32 buyerId	Получает записи, которые понравились пользователю-покупателю.

Описание полей, методов и свойств класса SearchController.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
_service	public	DbContext	Сервис базы данных.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SearchController	public	Конструктор	DbContext service	Конструктор контроллера.
SearchSellers	public	ActionResult<List<String>>	String q	Получает пользователей-продавцов, которые содержат введенную строку в имени.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

GetSellerByName	public	Seller	String sellerName	Получает пользователя- продавца по имени.
-----------------	--------	--------	----------------------	---

Описание полей, методов и свойств класса SubscribeController.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
service	public	DbContext	Сервис базы данных.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
SubscribeController	public	Конструктор	DbContext service	Конструктор контроллера.
AddSubscriberToSeller	public	ActionResult	Int32 buyerId, String sellerName	Добавляет подписчика пользователя-покупателя к пользователю-продавцу.
RemoveSellerSubscriber	public	ActionResult	Int32 buyerId, String sellerName	Удаляет из подписчиков пользователя-покупателя у пользователю-продавцу.

Описание полей, методов и свойств класса DatabaseContext.cs

Поля				
Имя	Модификатор доступа	Тип	Назначение	
UsersIconsDirPath	private	String	Путь для сохранения картинок.	
Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
DatabaseContext	public	Конструктор	DbContextOptions<DatabaseContext> options	Конструктор.
GetAllUsers	public	List<User>	-	Получает всех пользователей, зарегистрированных в системе.
CreateUser	public	User	Buyet user	Создает нового пользователя-покупателя.
CreateUser	public	User	Seller user	Создает нового пользователя-продавца.
IsSameSellerCreated	private	Boolean	User user	Проверяет, создан ли пользователь-

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

				продавец с таким же email или username.
IsSameBuyerCreated	private	Boolean	User user	Проверяет создан ли пользователь-покупатель с таким же email или username.
GetBuyer	public	Buyer	(String login, String password)	Получает пользователя по логину и паролю.
GetBuyer	private	Buyer	Int32 id	Получает пользователя-покупателя по его id.
GetSeller	public	Seller	string login, string password	Получает пользователя по логину и паролю.
GetSeller	private	Seller	Int32 id	Получает пользователя-продавца по его id.
GetSeller	public	Seller	String sellerName	Получает пользователя по имени.
AddIconToSeller	public	Boolean	IconPostForm icon, Int32 sellerId	Добавляет иконку пользователю-продавцу.
AddIconToBuyer	public	Boolean	IconPostForm icon, Int32 buyerId	Добавляет иконку пользователю-покупателю.
AddIconToUser	private	void	IconPostForm icon, User toUser	Добавляет иконку пользователю.
SearchSellers	public	List<String>	String s	Ищет пользователя-покупателя, который содержит в username поисковую строку.
AddSubscriber	public	Boolean	Int32 buyerId, String sellerName	Добавляет подписчика.
AddSubscriber	private	Boolean	Int32 buyerId, Int32 sellerId	Добавляет подписчика.
RemoveSubscriber	public	Boolean	Int32 buyerId, String sellerName	Удаляет подписчика.
RemoveSubscriber	private	Boolean	Int32 buyerId, String sellerName	Удаляет подписчика.
CreatePost	public	Boolean	CreatePostForm postForm	Создает новую запись и добавляет ее в базу данных.
AddPostForSubscribers	private	void	Seller creator, Int32 postId	Добавляет новую запись всем

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

				подписчикам пользователя- продавца.
GetPostById	public	Post	Int32 id	Получает экземпляры записи по id.
GetPostsByCreatorName	public	List<Post>	String sellerName	Получаем записи пользователя- продавца по имени.
GetBuyerNewPosts	public	List<BuyerPostView>	Int32 buyerId	Получает новые записи пользователя- покупателя по его id.
AddLikeToPost	public	void	Int32 likerId, Int32 postId	Добавляет лайк пользователя- покупателя к записи.
RemoveLikeFromPost	public	void	Int32 likerId, Int32 postId	Удаляет лайк пользователя- покупателя к записи.
GetBuyerLikedPosts	public	List<BuyerPostView>	Int32 buyerId	Получает записи, понравившиеся пользователю- покупателю.
AddOrder	public	void	Int32 buyerId, Int32 postId	Создает новый заказ и добавляет его в базу данных.
GetBuyerOrders	public	List<BuyerOrder View>	Int32 buyerId	Получает заказы пользователя- покупателя.
GetSellerOrders	public	List<SellerOrder View>	Int32 sellerId	Получает заказы пользователя- продавца.
GetOrder	private	Order	Int32 orderId	Получает заказ по Id.
SetOrderSuccess	public	void	Int32 orderId	Устанавливает одобренный статус заказа.
SetOrderCanceled	public	void	Int32 orderId	Устанавливает отклоненный статус заказа.
DeleteOrder	public	void	Int32 orderId	Удаляет заказ.
SetBuyerInfo	public	void	Int32 buyerId, String info	Изменяет информацию о пользователе- покупателе.
SetSellerInfo	public	void	Int32 sellerId, String info	Изменяет информацию о

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

				пользователь-продавец.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Sellers	private	DbSet<Seller>	get;set;	База данных пользователей-продавцов.
Buyers	private	DbSet<Buyer>	get;set;	База данных пользователькей-покупателей.
Posts	private	DbSet<Post>	get;set;	База данных записей пользователей-продавцов.
Orders	private	DbSet<Order>	get;set;	База данных заказов пользователей-покупателей.

Описание полей, методов и свойств класса Program.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Main	public	void	string[] args	Main метод.
CreateHostBuilder	public	IHostBuilder	string[] args	Создает хост.

Описание полей, методов и свойств класса Startup.cs

Методы				
Имя	Модификатор доступа	Тип	Аргументы	Назначение
Startup	public	Конструктор	IConfiguration configuration	Конструктор.
ConfigureServices	public	void	IServiceCollection services	Добавляет различные сервисы.
Configure	public	void	IApplicationBuilder app, IWebHostEnvironment env	Создает HTTP клиент.
Свойства				
Имя	Модификатор доступа	Тип	Доступ	Назначение
Configuration	public	IConfiguration	get;	Конфигурация сервера.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

[illegible]

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата