**2 ПРОЕКТИРОВАНИЕ**

## 2.1 Проектирование структуры системы

Общая структура системы изображена на рисунке 2.1



Рисунок 2.1 – Схема системы

Я выбрал архитектуру MVVM для разработки программы. Так как MVVM используется для разделения модели и её представления, что необходимо для изменения их отдельно друг от друга. Например, разработчик задает логику работы с данными, а дизайнер соответственно работает с пользовательским интерфейсом.

Теоретические сведение об архитектуре:

Шаблон Model-View-ViewModel (MVVM) — применяется при проектировании архитектуры приложения. Первоначально был представлен сообществу Джоном Госсманом (John Gossman) в 2005 году как модификация шаблона Presentation Model. MVVM ориентирован на современные платформы разработки, такие как Windows Presentation Foundation, Silverlight от компании Microsoft, ZK framework.

Шаблон MVVM делится на три части:

1) Модель (англ. Model), так же, как в классической MVC, Модель представляет собой логику работы с данными и описание фундаментальных данных, необходимых для работы приложения.

2) Представление (англ. View) — это графический интерфейс, то есть окно, кнопки и т. п. Представление является подписчиком на событие изменения значений свойств или команд, предоставляемых Моделью Представления. В случае, если в Модели Представления изменилось какое-либо свойство, то она оповещает всех подписчиков об этом, и Представление, в свою очередь, запрашивает обновленное значение свойства из Модели Представления. В случае, если пользователь воздействует на какой-либо элемент интерфейса, Представление вызывает соответствующую команду, предоставленную Моделью Представления.

3) Модель Представления (англ. ViewModel) является, с одной стороны, абстракцией Представления, а с другой, предоставляет обёртку данных из Модели, которые подлежат связыванию. То есть, она содержит Модель, которая преобразована к Представлению, а также содержит в себе команды, которыми может пользоваться Представление, чтобы влиять на Модель.

Общая схема архитектуры MVVM изображена на рис. 2.2

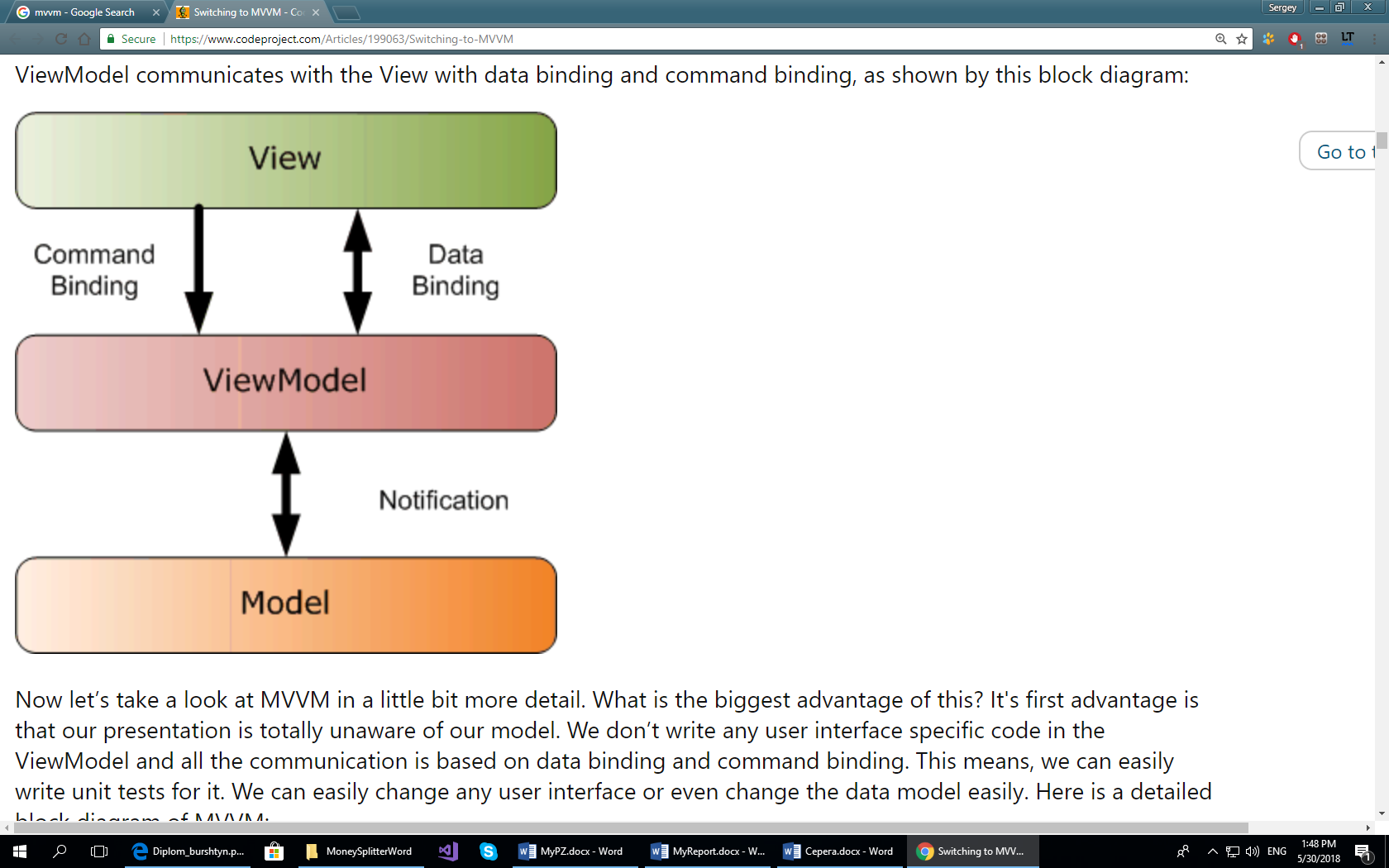


Рисунок 2.2 – Схема архитектуры MVVM.

## 2.2 Проектирование программного обеспечения системы

Организация корневого каталога проекта. Корневой каталог исходного кода системы представлен на рисунке 2.3.

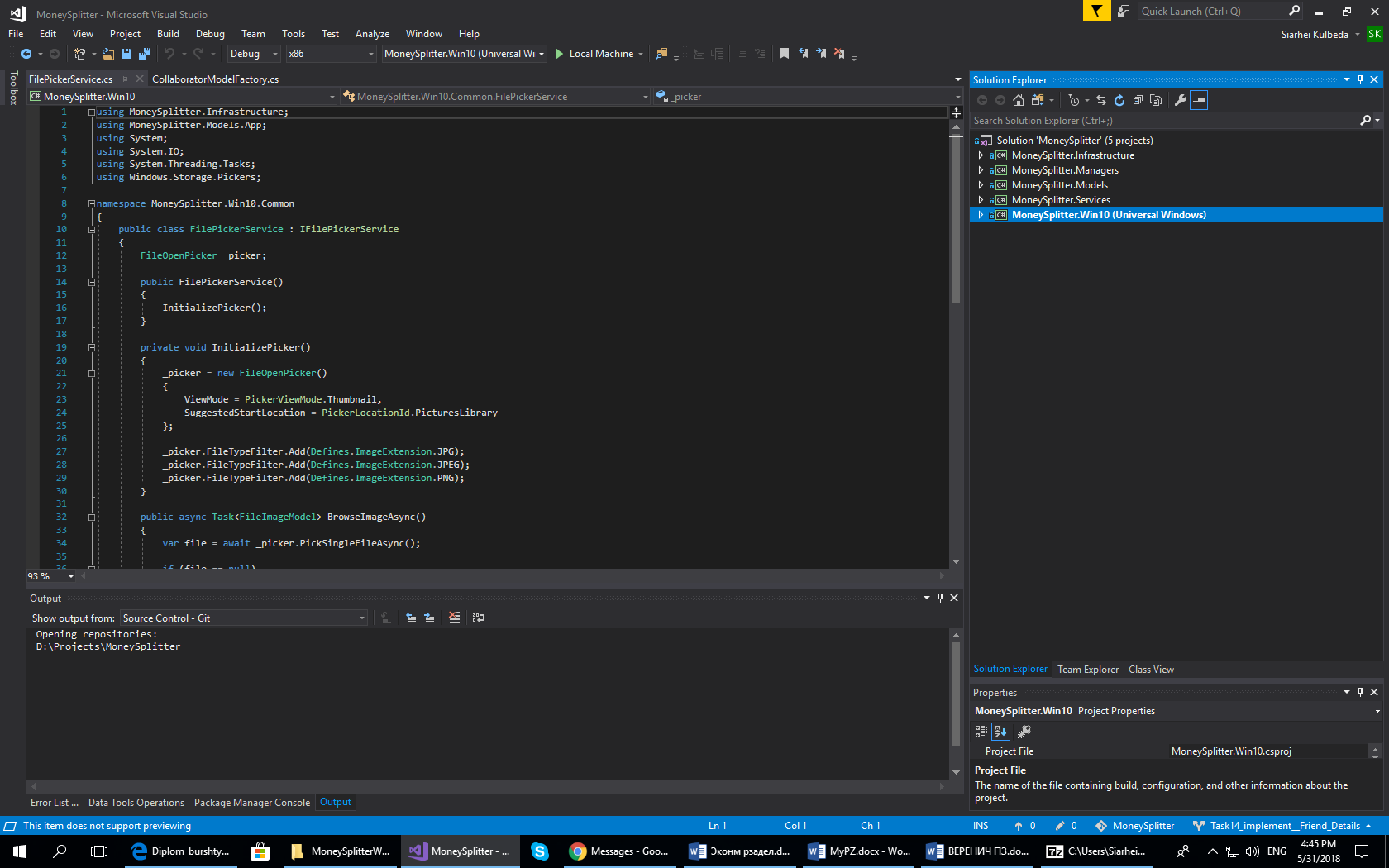


Рисунок 2.3 - Корневой каталог проекта

Проект будет состоять из 5 модулей. Каждый модель он не зависим и в последующим его можно заменить другим.

Рассмотрим назначение модулей:

Infrastrucure ‑ модуль в котором хранятся все интерфейсы программы. Интерфейсы позволяют определить некоторый функционал для класса, не имеющий конкретной реализации. Класс, который подписан на конкретный интерфейс, обязан реализовать весь функционал, прописанный в интерфейсе. На один интерфейс могут порисоваться множество классов. И интерфейс в программе может служить, как обобщённый тип. Но работать мы будем с классами реализующие этот интерфейс.

Managers ­ модуль, где реализуются менеджеры. Менеджеры — это такие классы способны управлять и манипулировать данными. Также они умеют обрабатывать входные данные.

Models ­ каталог, содержащий пользовательские классы, используемые в приложении, все они описывают сущности.

Services ­ модуль, где реализуются сервисы. Сервисы ­ это классы, которые предоставляют данные. Они никак не манипулируют с данными, а про сто предоставляют ее. Есть сервисы, которые запрашивают данные у веб-сервера, и предоставляют её в удобном виде. Такие сервисы, называются api-сервисы.

Win10 ­ это модуль отвечающий за клиентскую часть приложения. Все остальные модули ­ вспомогательные. Они кроссплатформенные, ни как как не зависят от клиентской части. В теории, мы можем по-разному реализовать клиент. И это не скажется на всех остальных модулях.

На рис. 2.4 предоставлен каталог Win10.

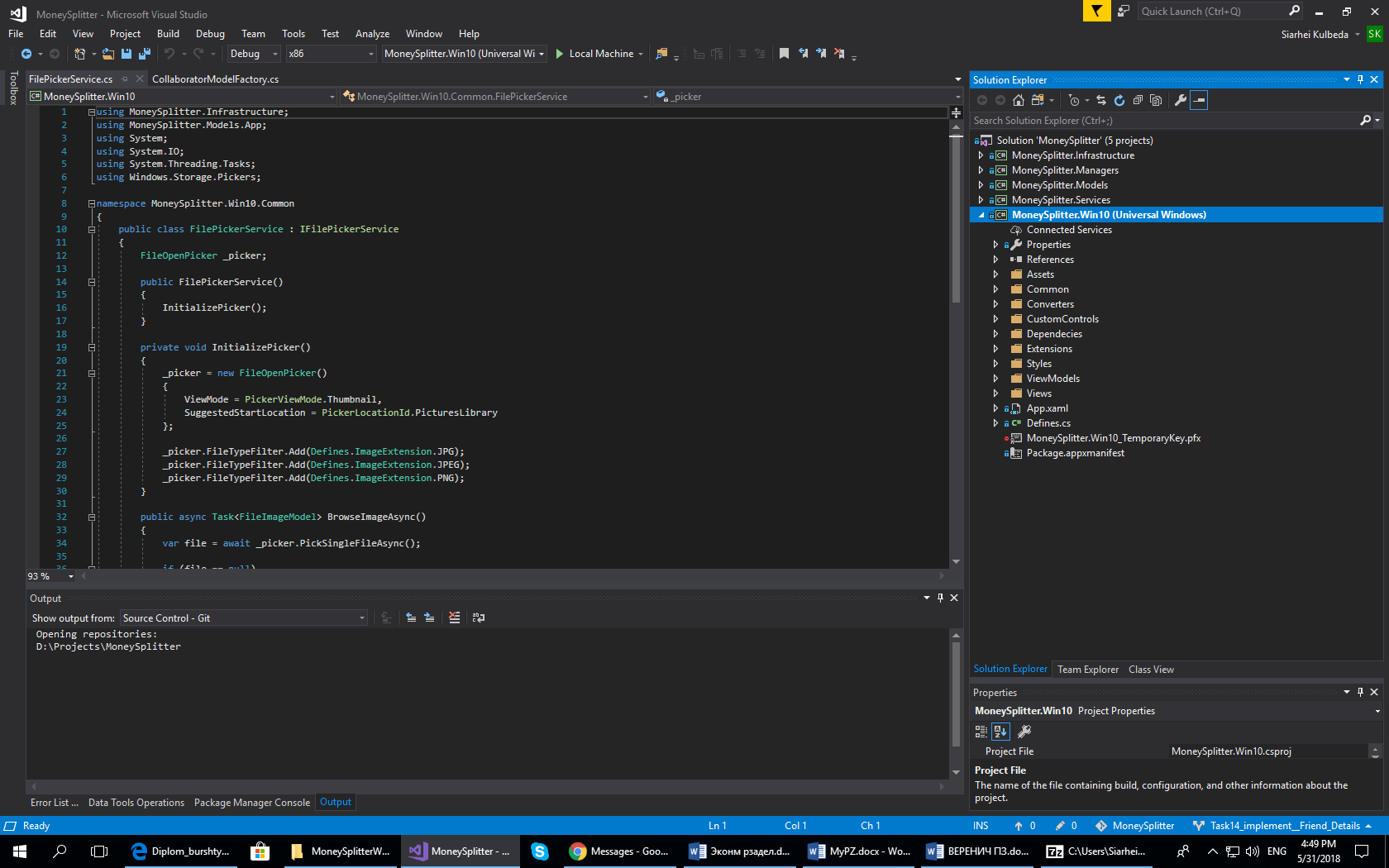


Рисунок 2.4 – Содержание каталога Win10

В данном части приложение осуществляется работа с пользовательским интерфейсом. Рисуются представления, разрабатываются стили. И создаются функционал для вывода необходимой информации на экран.

## 2.3 Структура информационного обеспечения

Информационное обеспечение приложения будет представлять собой базу данных. База данных будет содержать семь сущностей (таблиц). В логическую модель базы данных включены такие таблицы, как:

* пользователь (User);
* транзакция (Transaction);
* друга транзакция (FriendTransaction);
* событие транзакции (TransactionEvent);
* участник транзакции (CollaboratorTransaction);
* участник, который отдал деньги, (CollaboratorInFinished);
* участник, который собирается отдать деньги (CollaboratorInProgress).

Логическая модель системы изображена на рисунке 2.5.

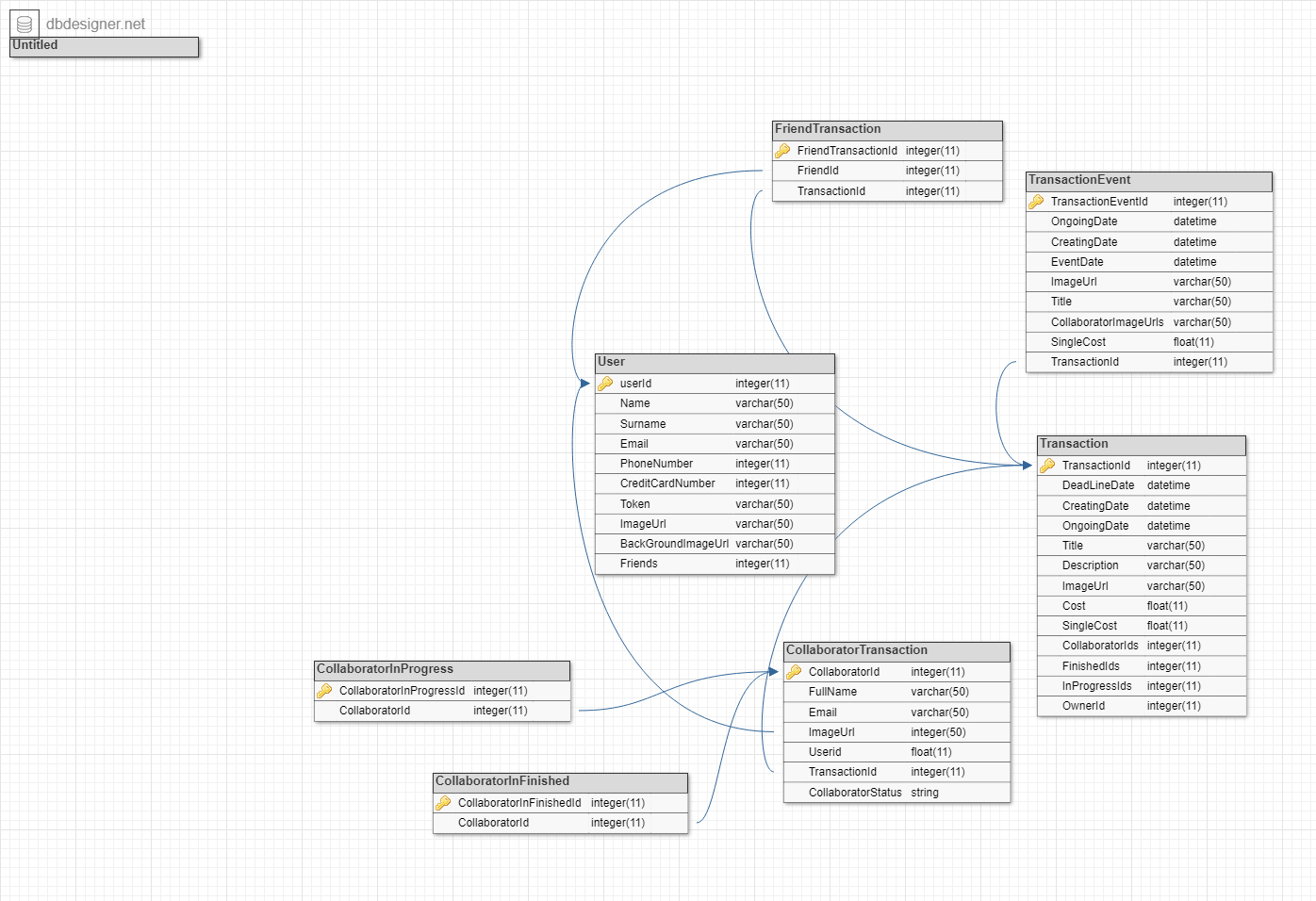


Рисунок 2.5 – Логическая модель системы

Описание сущности user представлено в таблице 2.1.

Таблица 2.1 – Описание сущности user

| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
| --- | --- | --- | --- |
| user | Объект, содержащий информацию о пользователе: id, имя, фамилию, электронную почту, номер телефона, номер банковской карты, ссылку на аватарку, и друзей. | Пользователь приложения | Используется для хранения и предоставления данных о пользователе. |

Свойства сущности выступают в роли атрибутов в базе данных, они представлены в таблице описания атрибутов по сущности administrators (таблица 2.2).

Таблица 2.2 – Описание атрибутов по сущности administrators

|  |  |  |  |
| --- | --- | --- | --- |
| Имя сущности | Имя атрибута | Назначение атрибута | Тип данных |
| user | UserId | Уникальный идентификатор | int |
| Name | Имя | varchar(50) |
| Surname | Фамилия | varchar(50) |
| Email | Электронная почта | varchar(50) |
| PhoneNumber | Номер телефона | int |
| CreditCardNumber | Номер банковской карты | int |
| Token | Идентификатор доступа | varchar(50) |
| ImageUrl | Ссылка на автатарку | varchar(50) |
| BackGroundUrl | Ссылка на картинку фона | varchar(50) |
| Friends | Друзья | varchar(50) |

Описание сущности transaction представлено в таблице 2.3.

Таблица 2.3 – Описание сущности transaction

| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
| --- | --- | --- | --- |
| transaction | Объект, содержащий информацию о транзакциях: id, название, описание, стоимость участники и т.д. | Транзакция | Используется для хранения и предоставления данных о транзакции. |

Свойства сущности выступают в роли атрибутов в базе данных, они представлены в таблице описания атрибутов по сущности transaction (таблица 2.4).

Таблица 2.4 – Описание атрибутов по сущности exams

|  |  |  |  |
| --- | --- | --- | --- |
| Имя сущности | Имя атрибута | Назначение атрибута | Тип данных |
| transaction | TransactionId | Уникальный идентификатор | int |
|  | Title | Название | varchar(50) |
| Description | Описание | varchar(50) |
| OwnerId | Владелец транзакции | int |
|  | ImageUrl | Изображение транзакции | varchar(50) |
| Cost | Стоимость | float |
| DeadLineDate | - | date |
| CreatingDate | Дата создания | date |
| OngoingDate | Дата события | date |
| CollaboratorIds | Участники | int |
| Продолжение таблицы 2.4 | | | |
| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
|  | InProgressIds | - | int |
| FinishedIds | - | int |
| SingleCost | - | float |

Описание сущности collaboratorTransaction представлено в таблице 2.5.

Таблица 2.5 – Описание сущности collaboratorTransaction

| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
| --- | --- | --- | --- |
| CollaboratorTransaction | Объект, содержащий информацию о участнике: id, полное имя, транзакция, статус участника, электронная почта и.т.д. | Участник транзакции | Используется для хранения и предоставления данных о участниках транзакции |

Свойства сущности выступают в роли атрибутов в базе данных, они представлены в таблице описания атрибутов по сущности collaboratorTransaction (таблица 2.6).

Таблица 2.6 – Описание атрибутов по сущности exam\_category

|  |  |  |  |
| --- | --- | --- | --- |
| Имя сущности | Имя атрибута | Назначение атрибута | Тип данных |
| CollaboratorTra-nsaction | collaboratorId | Уникальный идентификатор | int(11) |
| FullName | Полное имя | varchar(50) |
| Email | Электронная почта | varchar(50) |
| Cost | Стоимость | varchar(50) |
| ImageUrl | Изображение участника | varchar(50) |
| TransactionId | Транзакция | int |
| CollaboratorStatus | Участника статус | varchar(50) |
|  | UserId | Пользователь | int |

Описание сущности TransactionEvent представлено в таблице 2.7.

Таблица 2.7 – Описание сущности TransactionEvent

| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
| --- | --- | --- | --- |
| TransactionEvent | Объект, содержащий информацию о событии транзакции: id, название, стоимость дата события, изображение и.т.д. | Событие транзакции | Используется для хранения и предоставления данных о событии транзакции |

Свойства сущности выступают в роли атрибутов в базе данных, они представлены в таблице описания атрибутов по сущности TransactionEvent (таблица 2.8).

Таблица 2.8 – Описание атрибутов по сущности questions

|  |  |  |  |
| --- | --- | --- | --- |
| Имя сущности | Имя атрибута | Назначение атрибута | Тип данных |
| TransactionEvent | TransactionEventId | Уникальный идентификатор | int(20) |
| Title | Название | varchar(50) |
| ImageUrl | Изображение транзакции | varchar(50) |
| EventDate | Дата события | date |
| CreatingDate | Дата создания | date |
| OngoingDate | Дата события | date |
| CollaboratorImage | Изображение участников | blob |
| TransactionId | Транзакция | int |
| SingleCost | - | float |

Описание сущности FriendTransaction представлено в таблице 2.9.

Таблица 2.9 – Описание сущности FriendTransaction

| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
| --- | --- | --- | --- |
| FriendTransaction | Объект, содержащий информацию транзакциях друга:id, друг, транзакция. | Транзакция друга | Используется для хранения и предоставления данных о транзакции друга. |

Свойства сущности выступают в роли атрибутов в базе данных, они представлены в таблице описания атрибутов по сущности FriendTransaction (таблица 2.10).

Таблица 2.10 – Описание атрибутов по сущности FriendTransaction

|  |  |  |  |
| --- | --- | --- | --- |
| Имя сущности | Имя атрибута | Назначение атрибута | Тип данных |
| FriendTransaction | FriendTransactionId | Уникальный идентификатор | int(20) |
| TransactionId | Транзакция | int |
| FriendId | Друг | int |

Описание сущности CollaboratorInProgress представлено в таблице 2.11.

Таблица 2.11 – Описание сущности CollaboratorInProgress

| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
| --- | --- | --- | --- |
| CollaboratorInProgress | Объект, содержащий информацию о участниках, которые еще не завершили транзакцию, id, участник транзакции. | Участники, которые еще не завершили транзакцию | Используется для хранения и предоставления данных о участниках, которые еще не завершили транзакцию. |

Свойства сущности выступают в роли атрибутов в базе данных, они представлены в таблице описания атрибутов по сущности CollaboratorInProgress (таблица 2.12).

Таблица 2.12 – Описание атрибутов по сущности CollaboratorInProgress

|  |  |  |  |
| --- | --- | --- | --- |
| Имя сущности | Имя атрибута | Назначение атрибута | Тип данных |
| CollaboratorInProgress | CollaboratorInProgressId | Уникальный идентификатор | int |
| CollaboratorId | Участник транзакции | int |

Описание сущности CollaboratorInFinished в таблице 2.13.

Таблица 2.13 – Описание сущности CollaboratorInFinished

| Имя сущности | Описание сущности | Псевдонимы | Особенности использования |
| --- | --- | --- | --- |
| CollaboratorInFinished | Объект, содержащий информацию о участниках, которые завершили транзакцию, id, участник транзакции. | Участники, которые завершили транзакцию | Используется для хранения и предоставления данных о участниках, которые завершили транзакцию. |

Свойства сущности выступают в роли атрибутов в базе данных, они представлены в таблице описания атрибутов по сущности CollaboratorInFinished (таблица 2.14).

Таблица 2.14 – Описание атрибутов по сущности users

|  |  |  |  |
| --- | --- | --- | --- |
| Имя сущности | Имя атрибута | Назначение атрибута | Тип данных |
| CollaboratorInFinished | CollaboratorInFinishedId | Уникальный идентификатор | int |
| CollaboratorId | Участник транзакции | int |

Описание ключей представлено в таблице 2.15.

Таблица 2.15 – Описание ключей

|  |  |  |
| --- | --- | --- |
| Имя сущности | Первичный ключ | Альтернативный ключ |
| пользователь | userId | - |
| транзакция | transactionId | userId |
| друга транзакция | friendTransactionId | userId, transactionId |
| событие транзакции | transactionEventId | transactionId |
| участник транзакции | CollaboratorId | userId, transactionId |
| участник, который отдал деньги, | CollaboratorInFinishedId | CollaboratorId |
| участник, который собирается отдать деньги | CollaboratorInProgressId | CollaboratorId |

## 2.3 Структура пользовательского интерфейса

В пользовательской части приложения предполагается наличие пользовательского интерфейса, посредством которого пользователи будут взаимодействовать с системой.

При открытии приложения будет открываться страница входа/регистрации. На данной странице расположены две кнопки: «Вход» и «Регистрация». По умолчанию будет доступна форма для входа. На которой присутствуют поля: «Email», «Пароль» и кнопка «Войти». Прототип этой страницы изображен на рисунке 2.4.

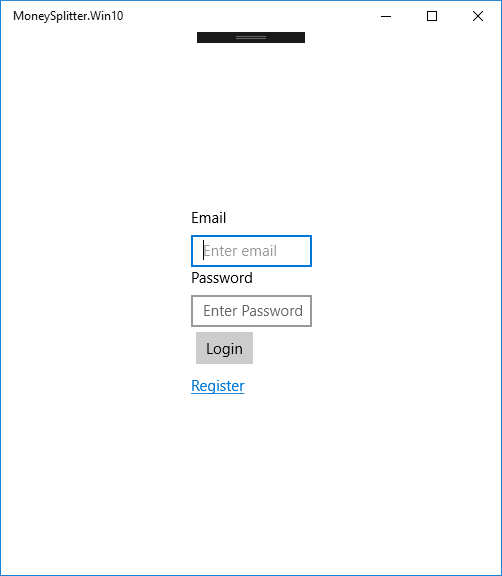


Рисунок 2.6 – Форма входа

При нажатии на кнопку «Регистрация» форма входа будет заменятся на форму регистрации. На которой присутствуют поля: «Email», «Password», «Confirm password», «Name», «Surname», «Phone number», «Browse avatar image» , «Browse background image» и кнопка «Registered». Прототип этой страницы изображен на рисунке 2.7.

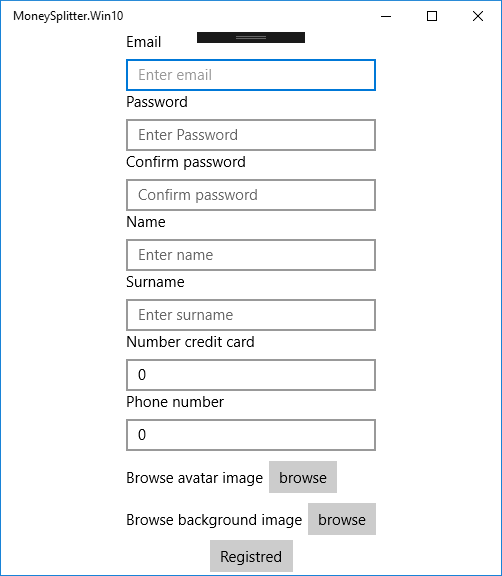


Рисунок 2.7 – Форма регистрации

После входа в систему пользователь попадает домашнею страницу, где есть возможность просмотреть ближайшее событие, а также краткую сводку, сколько вам должны, вы должны. Домашняя страница пользователя изображена рисунке 2.8.

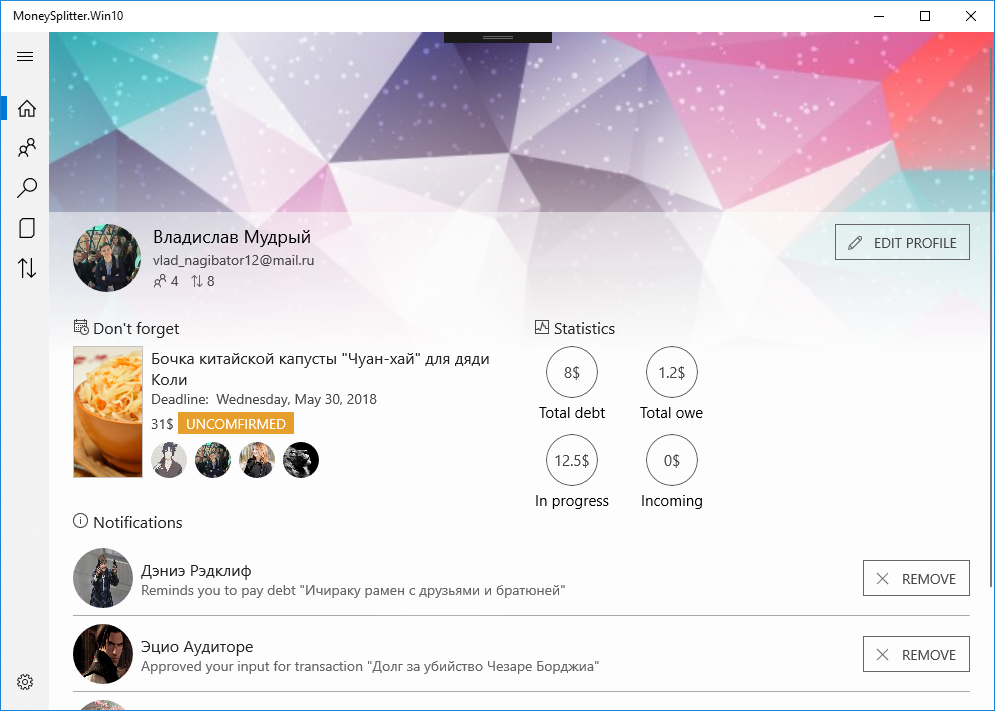


Рисунок 2.8 – Домашняя страница пользователя

Слева будет находится меню, которое имеет три режима, полностью свернутое (для мобильных телефонов), частично свернутое (для планшетов) и полностью развернутое (для компьютеров). При выборе пункта «Friends» пользователь попадает на страницу, где представлены его друзья. Можно узнать подробную информацию друга, щелкнув на иконку друга. Также можно удалить друга, нажав на кнопку «Remove». Прототип этой страницы изображен на рисунке 2.9.

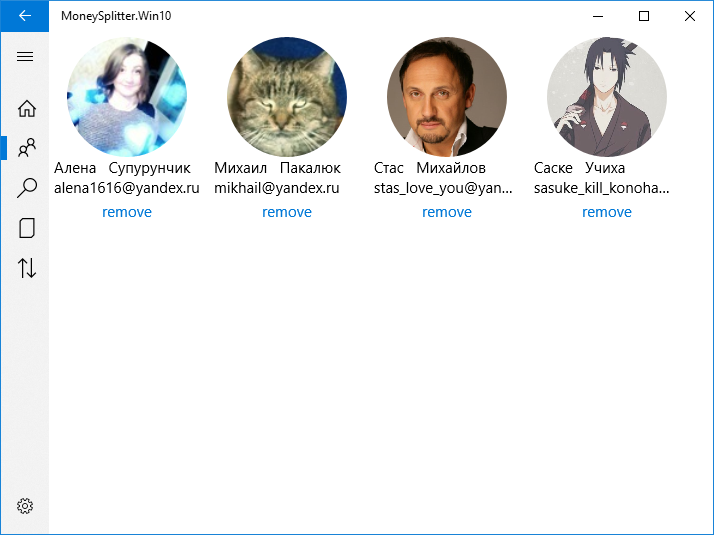


Рисунок 2.9 – Друзья пользователя

Когда пользователь нажимает на и иконку друга ­ он переходит на страницу детали друга. Там он увидит необходимую информацию, чтобы связаться с данным пользователем, это телефон и адрес электронной почты. А также имеется номер банковской карты, куда можно скинуть денежные средства. Справа подсчитана разница сумм всех долгов и займов, и можно увидеть сколько ты должен денег или сколько тебе должен друг. Пользователь имеет право погасить все сделки, если друг в жизни отдал эту сумму. Для этого нужно нажать на кнопку «Approve all». Внизу страницы показаны следующие транзакции по отношению к пользователю, который зашел на данную страницу: долги друга, займы друга, транзакции в которых вы с другом участвуете и друг другу ничего не должны. Этим с данными транзакциями можно манипулировать использую кнопки справа от транзакций. Прототип этой страницы изображен на рисунке 2.10.

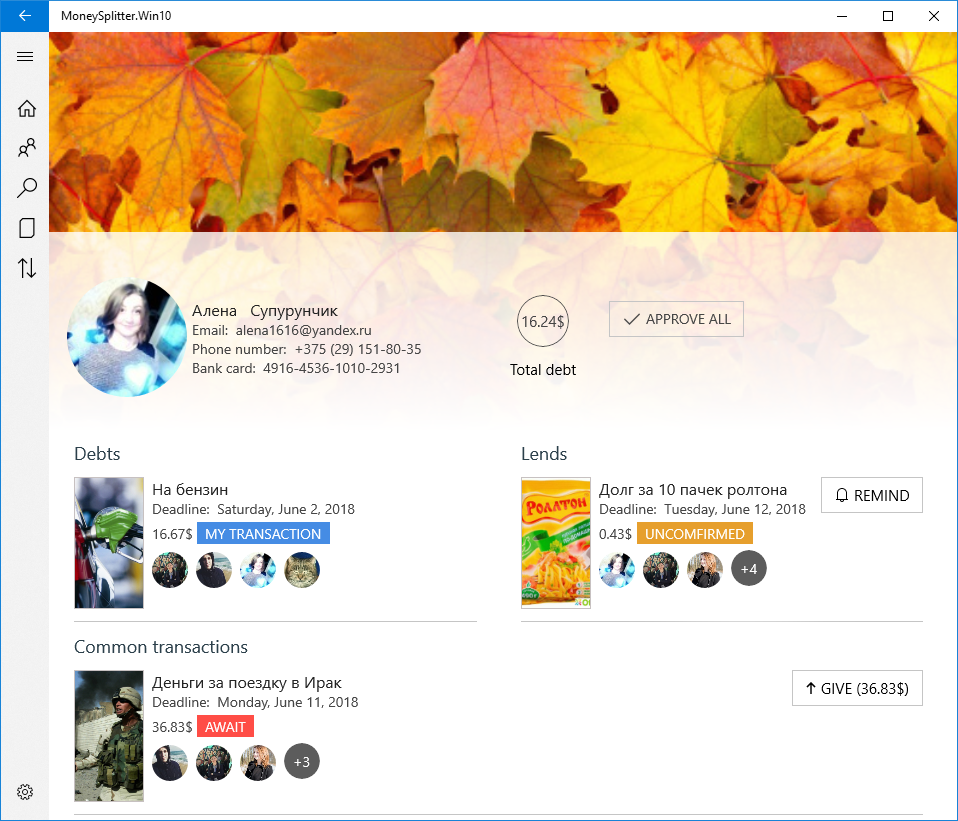


Рисунок 2.10– Детали друга

После нажатия на кнопку «Search» в меню, пользователь попадет непосредственно на страницу поиска пользователей. На странице будет располагаться форма для поискового запроса. Вводя строку, мгновенна будут отображаться найденные пользователи. Далее их можно добывать в друзья. Прототип этой страницы изображен на рисунке 2.11.

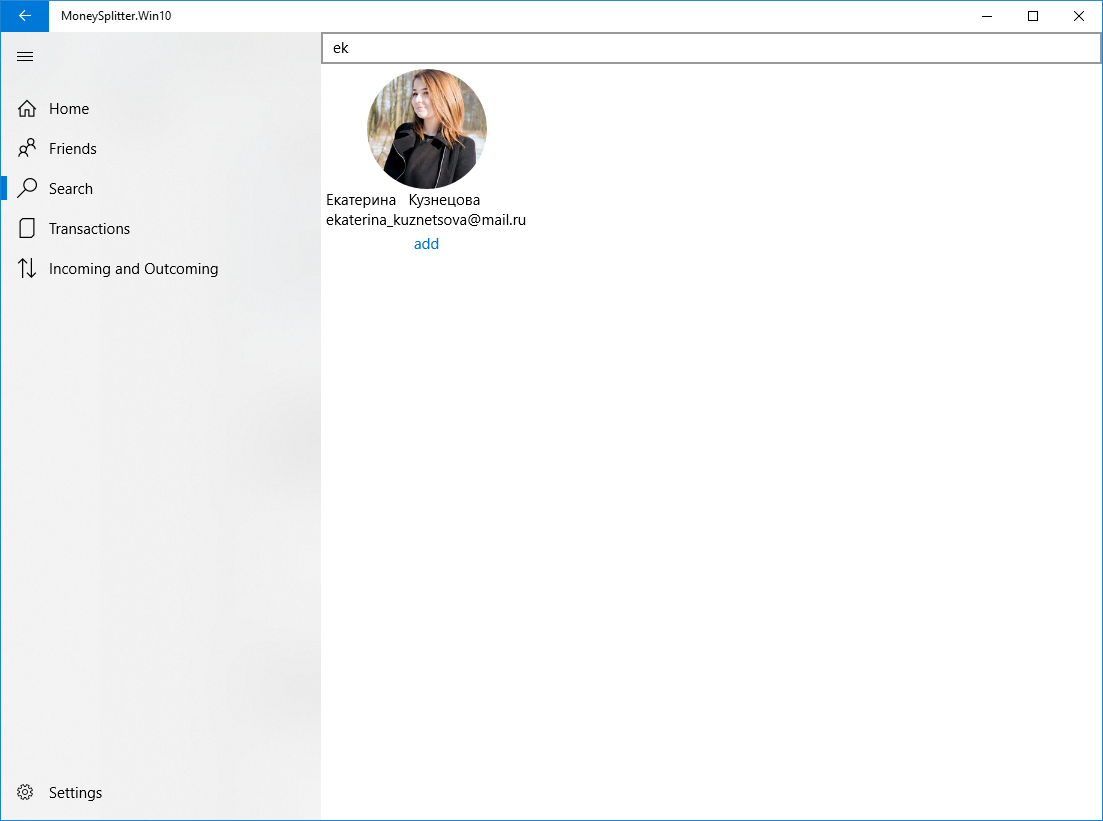


Рисунок 2.11 – Поиск пользователей

После нажатия на кнопку «Transactions» в меню, пользователь попадет на страницу транзакций пользователей. На странице будет располагаться список всех транзакций пользователя. У каждой транзакция хранит следующую информацию. Название, дедлайн или дата события, стоимость, и метка определяющую роль пользователя в этой транзакции. А также некоторые транзакции имеет действии: отдать, напомнить. Вверху страницы есть кнопка «Add», при нажатии переходим на страницу добавлении транзакции. Также есть кнопка для сортировки по категории. Прототип этой страницы изображен на рисунке 2. 12.

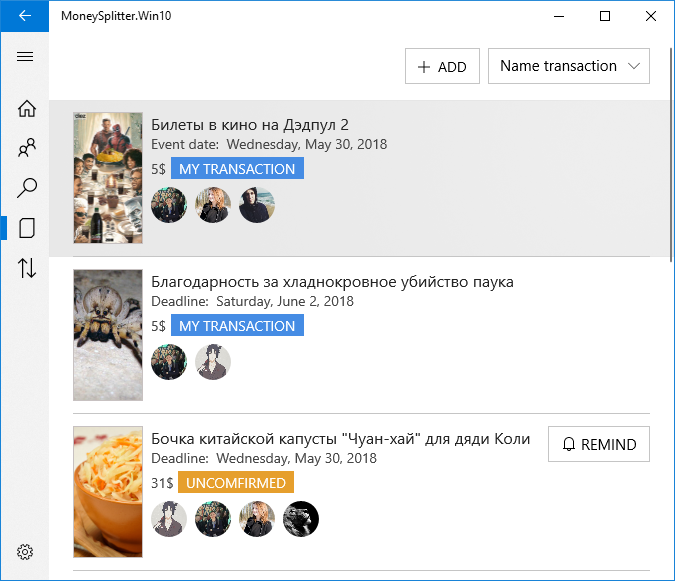


Рисунок 2.12 – Транзакции пользователя

При выборе транзакции открывается детали транзакции. Да данной странице располагается подробная информация о транзакции. Небольшая информация об участниках транзакции, также имеется информация о владельце и прогрессе пользователей в данной сделке. Прототип этой страницы изображен на рисунке 2. 13.

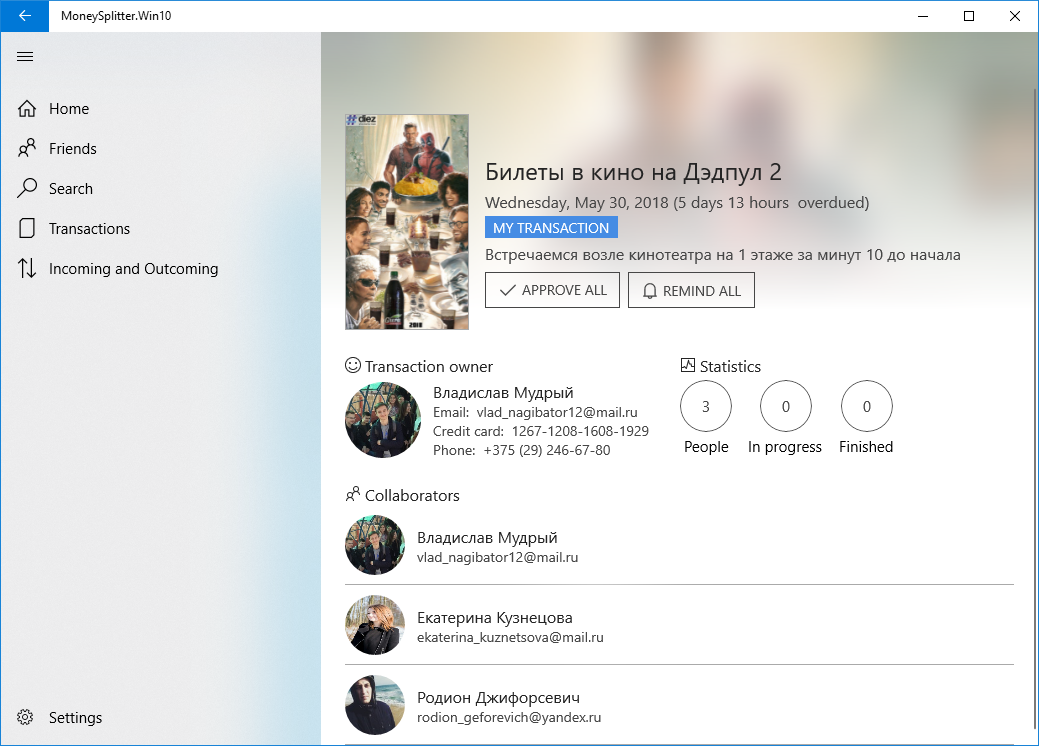


Рисунок 2.13 – Детали транзакции

При выборе пункта «Add» пользователь попадает на страницу для создание новой транзакции: имеются поля название, описание цена, дедлайн, изображение, и возможность выбора участников из друзей. Прототип этой страницы изображен на рисунке 2. 14.

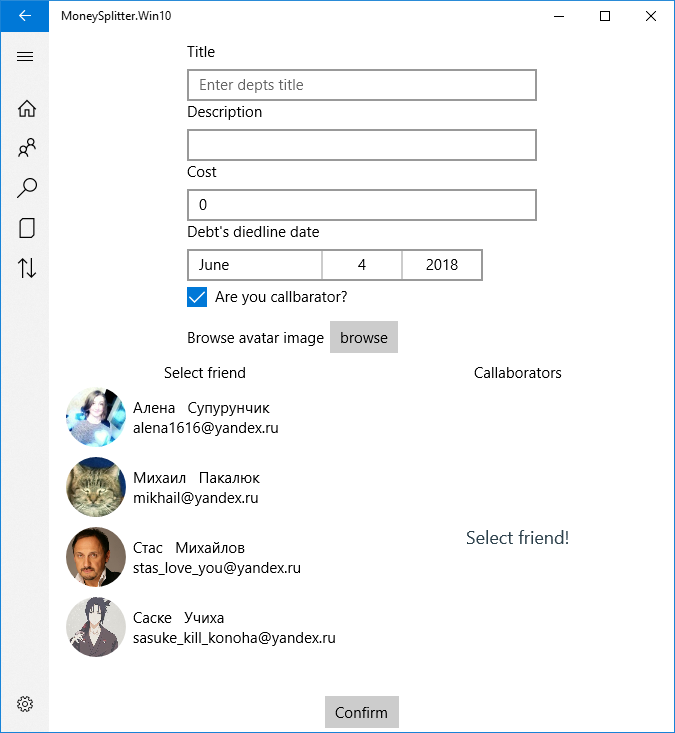


Рисунок 2.14 – Добавление транзакции

После нажатия на кнопку «IncomingAndOutgoing» в меню, пользователь попадет на страницу c должниками и люди, одолжившие пользователю. Показывает кто пользователю должен сколько, нажав на запись, происходит переход на детали пользователя. Прототип этой страницы изображен на рисунке 2. 15

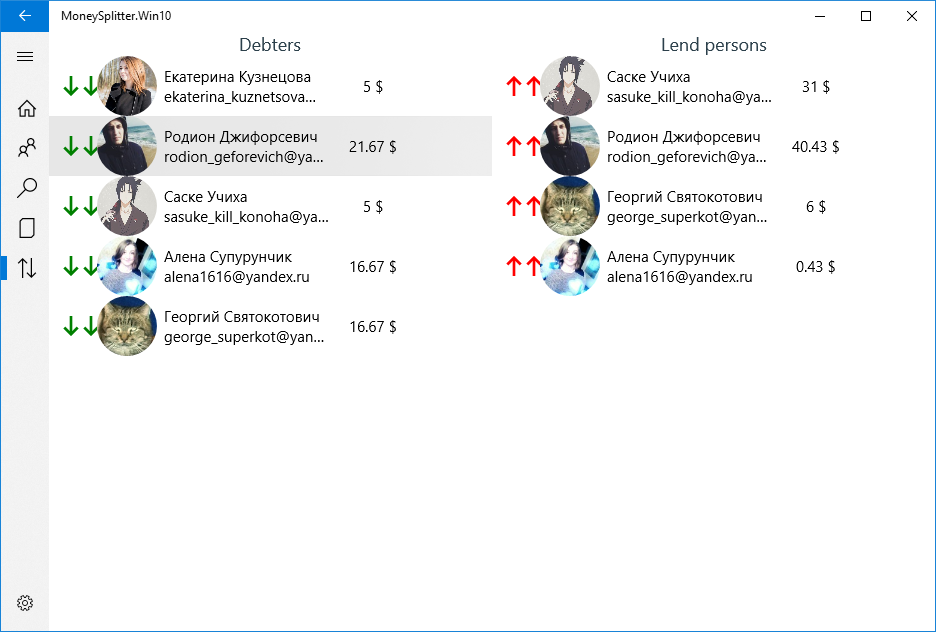


Рисунок 2.15 – Должники и одолжившие