МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИИ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ**

**«МИСиС»**

КАФЕДРА ИНЖЕНЕРНОЙ КИБЕРНЕТИКИ

**КУРСОВАЯ РАБОТА**

**ПРОГРАММА ПО УЧЕТУ СЕМЕЙНОГО БЮДЖЕТА «ВАШ БАНКИРЪ»**

по курсу «Технологии программирования»

Группа: БПИ-19-3

Студент: Рыбалко Д.А.

Преподаватель: Широков А.И.

Москва,

2020г

Оглавление

[1. Техническое задание 3](#_Toc61605673)

[1.1. Введение 3](#_Toc61605674)

[1.2. Основание для разработки 3](#_Toc61605675)

[1.3. Назначение разработки 3](#_Toc61605676)

[1.4. Требования к программе 3](#_Toc61605677)

[1.4.1. Функциональные характеристики. 3](#_Toc61605678)

[1.4.2. Надежность 5](#_Toc61605679)

[1.4.3. Эксплуатация 5](#_Toc61605680)

[1.4.4. Состав и параметры технических средств 5](#_Toc61605681)

[1.4.5. Информационная программная совместимость 5](#_Toc61605682)

[1.5. Требования к документации 5](#_Toc61605683)

[1.6. Стадии и этапы разработки 5](#_Toc61605684)

[1.7. Порядок контроля и приема 6](#_Toc61605689)

[2. Эскизный проект 6](#_Toc61605690)

[2.1. Предварительная разработка структуры данных 6](#_Toc61605691)

[2.2. Уточнение методов решения задачи 6](#_Toc61605692)

[2.3. Разработка общего описания алгоритма решения задачи 6](#_Toc61605693)

[3. Технический проект 6](#_Toc61605694)

[3.1. Уточнение структуры входных и выходных данных 6](#_Toc61605695)

[3.2. Разработка алгоритма решения задачи 7](#_Toc61605696)

[3.3. Определение формы представления входных и выходных данных 8](#_Toc61605697)

[3.4. Определение семантики и синтаксиса языка 8](#_Toc61605698)

[3.5. Разработка структуры программы. 9](#_Toc61605699)

[3.6. Окончательное определение конфигурации технических средств. 10](#_Toc61605700)

[Информационная программная совместимость 10](#_Toc61605701)

[4. Рабочий проект 10](#_Toc61605702)

[4.1 Программирование и отладка программы 10](#_Toc61605703)

[4.2 Корректировка программы и программной документации по итогам испытаний 11](#_Toc61605704)

[5. Внедрение 12](#_Toc61605705)

[5.1. Руководство по установке 12](#_Toc61605706)

[5.2. Инструкция по эксплуатации 12](#_Toc61605707)

[6. Приложение 19](#_Toc61605708)

# 1. Техническое задание

## 1.1. Введение

Я выбрал тему под номером 12, звучит она так: «Программа учета и анализа доходов и расходов семьи. Хранятся данные о не менее 6 членов семьи и не менее трех поколений. Старшее поколение получает пенсию (один раз в месяц). Среднее поколение получает зарплату два раза в месяц. Деньги поступают в общий фонд, из которого совершаются траты.»

Цель данной курсовой работы – приобретение практических навыков в области отладки интерактивных систем программного обеспечения, а также создание приложения анализа финансов «Ваш БанкирЪ». Приложение позволяет вести учет финансов семьи из 6 человек, анализировать доходы и расходы, а также добавлять финансовые цели. Программа имеет графический интерфейс, основанный на базе платформы UWP. Ввод данных и навигация происходит посредством клавиатуры и мыши.

## 1.2. Основание для разработки

Выполнение курсовой работы по предмету «Технологии программирования».

## 1.3. Назначение разработки

Назначением программы является предоставление широкому кругу пользователей возможности вести учет своих финансов, а также финансов своих родственников.

## 1.4. Требования к программе

### 1.4.1. Функциональные характеристики.

При запуске программы пользователь увидит окно входа в приложение, где необходимо будет ввести заранее заданный логин и пароль. Начальные значения логина и пароля заданы, но их возможно будет сменить в дальнейшем. Каждый логин и пароль соответствуют заданному аккаунту, который уникален. После входа в приложение пользователь увидит меню выбора функции.

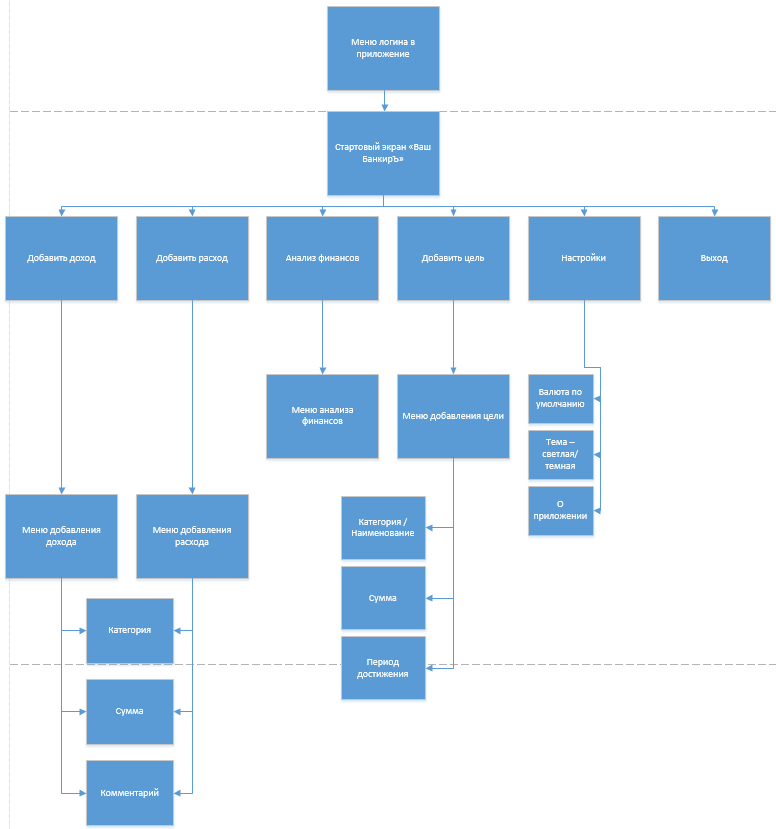


Рис. 1 – неокончательная схема программы

Каждая из функций реализуется индивидуально для каждого пользователя. Пользователь может добавлять доходы и расходы только для себя (от своего имени). Статистику и анализ финансов можно смотреть для всей семьи, а также для своего поколения и для конкретного человека. Можно добавить коллективную или индивидуальную цель, что поможет пользователю экономить и копить на то, что он давно хочет. В настройках можно поменять свой пароль, а также найти информацию о программе и разработчике – версию приложения, разработчика и ссылку на открытый проект на платформе GitHub

### 1.4.2. Надежность

Программа автоматически сохраняет все изменения в память, при экстренном выключении компьютера все сохраненные изменения остаются.

### 1.4.3. Эксплуатация

Требования не предъявляются

### 1.4.4. Состав и параметры технических средств

В состав технических средств должен входить персональный компьютер, включающий в себя:

* Процессор: не менее 1 ГГц
* ОЗУ: не менее 1 Гбайт
* Свободное место на накопителе: не менее 1 Гбайт
* Видеоадаптер: DirectX 9 или более поздняя версия с драйвером WDDM 1.0
* Экран: не меньше 800 x 600
* ОС: Windows 10 с обновлением Creators Update и выше

### 1.4.5. Информационная программная совместимость

• Система должна работать под управлением ОС Windows 10 с обновлением Creators Update и выше.

• Среда программирования Visual Studio 2017

• Язык программирования – С#, XAML

## 1.5. Требования к документации

Состав программной документации должен включать в себя:

1) техническое задание;

2) спецификация;

3) текст программы;

4) описание программы;

5) программу и методики испытаний;

6) руководство пользователя

## 1.6. Стадии и этапы разработки

Разработка должна быть проведена в 5 этапов:

## 1) выбор темы и разработка плана;

## 2) изучение литературы по теме и проведение анализа предметной области;

## 3) проектирование и разработка программного решения;

## 4) написание основной части курсовой работы;

5) оформление.

# 1.7. Порядок контроля и приема

* Испытания программы будут проводиться студентами ИТАСУ согласно разработанной методике испытаний.
* Ход проведения испытаний фиксируется в соответствующем отчете.
* После проведения испытаний в полном объеме проводится защита работы с последующей демонстрацией работы программы.

# 2. Эскизный проект

# 2.1. Предварительная разработка структуры данных

Входными данными в этом приложении будут данные, вводимые пользователем с помощью клавиатуры или мыши: различные суммы, выбор категории, даты, выбор раздела меню, добавление комментариев к расходам, доходам и целям, добавление самих расходов, доходов и целей, и др.

Выходными данными в данной программе будут являться данные, выводимые на экран пользователя в процессе использования приложения.

# 2.2. Уточнение методов решения задачи

В данной курсовой работе я буду использовать C#, язык разметки XAML и платформу Universal Windows Platform.

# 2.3. Разработка общего описания алгоритма решения задачи

В программе решаются несколько подзадач:

- Предоставление многопользовательского использования  
- Добавление расходов и доходов  
- Обработка и структурирование данных о финансах семьи и отдельных личностей  
- Добавление и редактирование финансовых целей  
- Настройка приложения

# 3. Технический проект

# 3.1. Уточнение структуры входных и выходных данных

**Входные данные:**

*- На странице добавления доходов/расходов:*

Сумма дохода/расхода, категория дохода/расхода, комментарий. Дата и имя пользователя задаются программой автоматически.

Добавить доход/расход «задним числом» нельзя.

*- На странице добавления целей:*

Наименование цели, сумма для накопления, комментарий (описание). Имя пользователя и дата задаются автоматически.

Накопленная сумма по умолчанию равна 0.

*- На странице просмотра активных целей:*

Редактирование существующих целей, путем изменения уже накопленной суммы.

Также возможно удаление существующих целей с помощью графического интерфейса.

***Все входные данные вводятся с помощью клавиатуры и мыши.***

**Выходные данные:**

*- На странице добавления доходов/расходов:*

Последние добавленные доходы/расходы (название, дата добавления и сумма, доп. информация по нажатию на кнопку)

*- На странице просмотра активных целей:*

Активные цели (название и доп. информация по нажатию на кнопку), прогресс достижения суммы для накопления.

# 3.2. Разработка алгоритма решения задачи

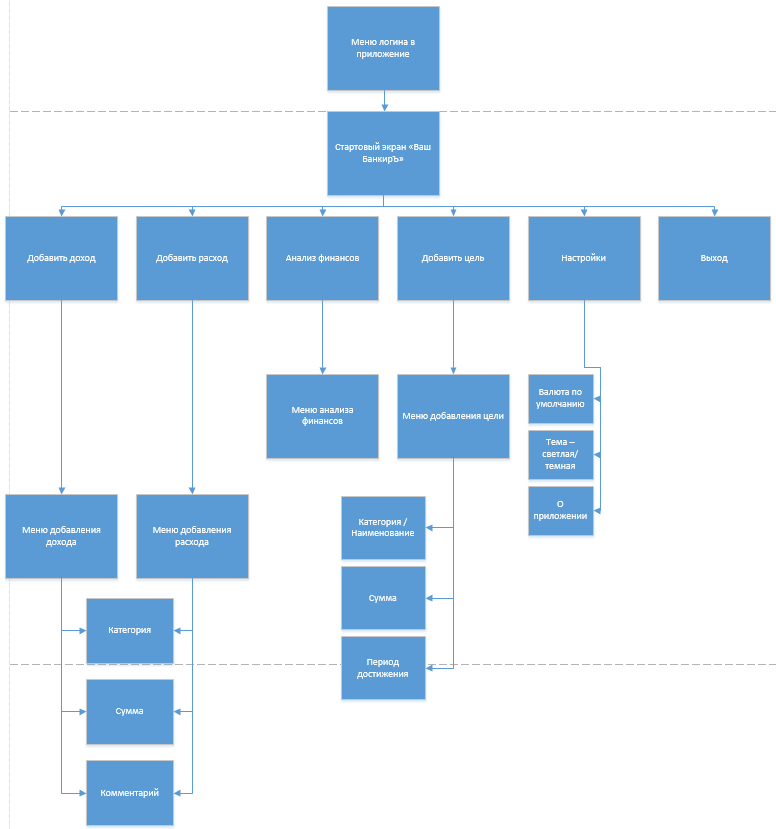


Рис. 2 – неокончательная схема программы

На схеме представлен обобщенный алгоритм решения различных задач. Схема содержит окна, которые будут отображаться в самом приложении:

- Окно входа в приложение  
- Окно добавления дохода  
- Окно добавления расхода  
- Окно анализа финансов  
- Окно добавления и редактирования цели  
- Окно настроек  
- а также кнопка выхода из приложения

# 3.3. Определение формы представления входных и выходных данных

Программа будет выполнена с использованием графического интерфейса => входные и выходные данные будут в графическом интерфейсе

# 3.4. Определение семантики и синтаксиса языка

Курсовая работа будет выполнена на языке C# с использованием языка разметки XAML.

# 3.5. Разработка структуры программы.

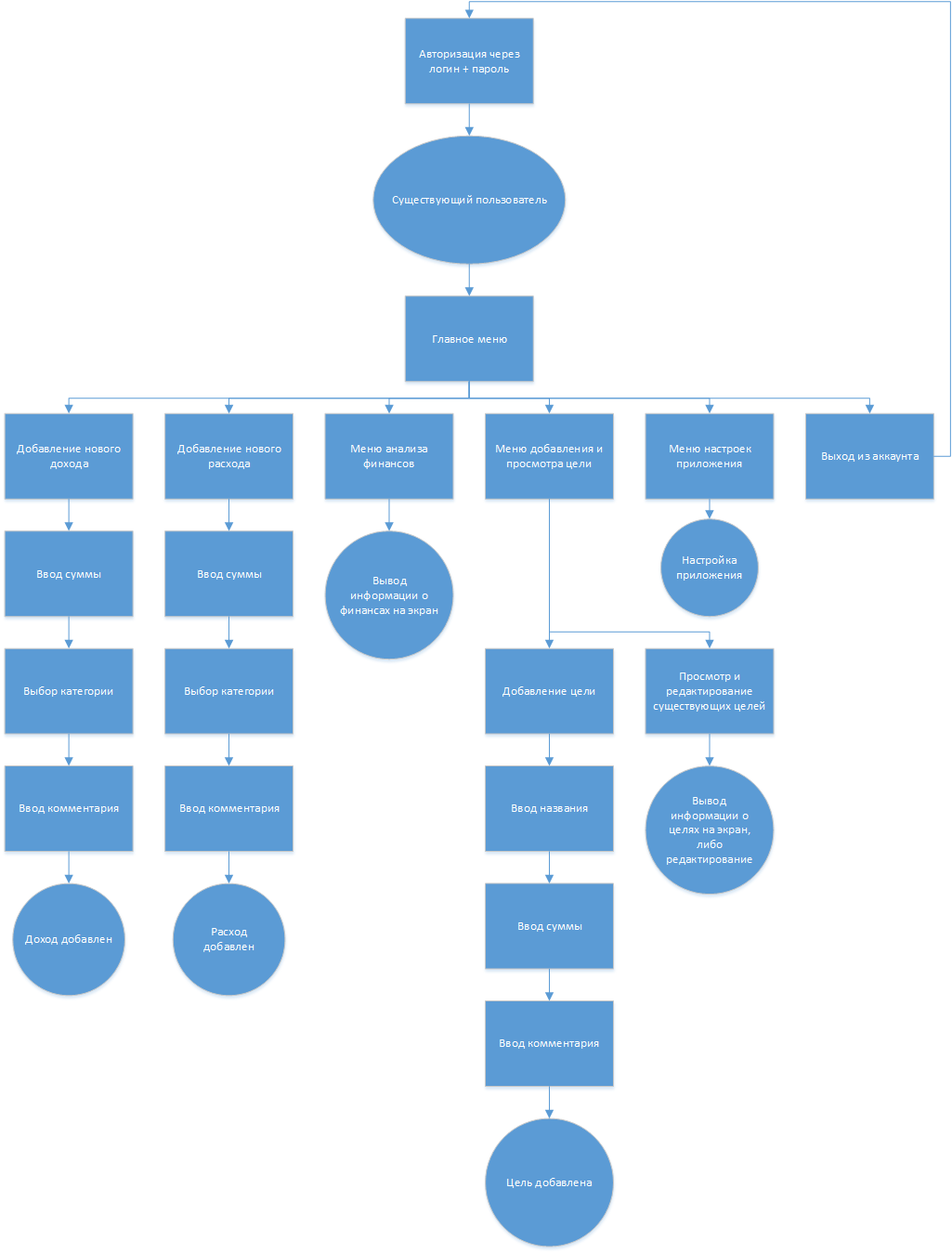


Рис.3 – Структура работы программы

На схеме представлена структура программы в виде дерева, где главное меню является головным модулем. Сначала проектируется главное меню, и уже потом проектируются дополнительные модули, такие как «добавление нового дохода» и так далее по нисходящей. Переход и навигация между страницами происходит только если ниже стоящий модуль спроектирован и запрограммирована логика работы с ним.

# 3.6. Окончательное определение конфигурации технических средств.

* Процессор: не менее 1 ГГц
* ОЗУ: не менее 1 Гбайт
* Свободное место на накопителе: не менее 1 Гбайт
* Видеоадаптер: DirectX 9 или более поздняя версия с драйвером WDDM 1.0
* Экран: не меньше 800 x 600
* ОС: Windows 10 с обновлением Creators Update и выше

### Информационная программная совместимость

- Система должна работать под управлением ОС Windows 10.

- Среда программирования Visual Studio 2017

- Язык программирования – С#, XAML

# 4. Рабочий проект

# 4.1 Программирование и отладка программы

В пункте 1.4 были уточнены требования для проекта. Основываясь на поставленных целях и требованиях, я создал графический интерфейс приложения на базе языка разметки XAML. Было использовано множество базовых форм, таких как:

- ComboBox – выпадающий список

- TextBox – поле для ввода текста

- Button – многофункциональная кнопка

- TextBlock – поле для вывода текста

- и другие

Изменение стилей и оформления происходит с помощью встроенных средств UWP и библиотек Windows.UI.

Разработка началась с разметки и программирования окна входа в приложение – LoginPage.xaml и LoginPage.xaml.cs. На ней пользователю предлагается ввести пароль и логин для входа в приложение. Также на ней есть доп. информация о версии приложения и разработчике. Далее была начата разработка MainMenuPage.xaml и MainMenuPage.xaml.cs – это главное меню всего приложение и преимущественно отсюда происходит вся навигация. Далее были разработаны остальные страницы, которые несут в себе функционал, такие как AddTargetPage.xaml (.cs), AddIncomePage.xaml (.cs) и т.д.

Весь проект очень большой и представлен в виде [публичного репозитория на GitHub](https://github.com/RMarK0/Your_Bankir/tree/debugging). Также по ссылке можно найти [wiki проекта](https://github.com/RMarK0/Your_Bankir/wiki), в котором будут полезные ссылки, например, история версий и известные недоработки и баги, которые необходимо решить.

Стек технологий сугубо встроенный – XAML + C# + UWP. Никаких иных библиотек я не использовал, но в планах есть внедрение библиотеки логирования NLog.

# 4.2 Корректировка программы и программной документации по итогам испытаний

Первым делом я начал проверять на всех текстовых полях для ввода правильность ввода данных. Пришлось переделывать множество кусков кода, вводить валидацию.

Итак, при логине в приложение происходит проверка логина и хеша(!) пароля. Хеш пароля необходим для того, чтобы нельзя было просто зайти в XML файл данных и просмотреть пароль.

Далее, я ввел ограничение на сумму доходов и расходов, а также финансовых целей – не более 8 символов в длину. Категория выбирается только через ComboBox, так что что-то не то ввести не получится. При отсутствии выбора категории программа выдаст ошибку и попросит изменить категорию.

Я проводил множество отладочных действий, и, к сожалению, не документировал абсолютно все из них. Лучше спрашивать индивидуально по каждому случаю.

# 5. Внедрение

# 5.1. Руководство по установке

Переведите вашу операционную систему в режим разработчика. Для этого в Настройках раздел «Обновления и Безопасность», затем войдите в категорию «Для разработчиков» и найдите переключатель «Режим разработчика». Переведите этот переключатель в положение Включено.

Для установки программы откройте файл сертификата «Ваш БанкирЪ v.x.x.x.x.cer», установите его для локального компьютера с автоматическими настройками. Затем, откройте файл установки «Ваш БанкирЪ v.x.x.x.x.appx» и следуйте инструкциям.

# 5.2. Инструкция по эксплуатации

После установки нужно запустить файл «Ваш БанкирЪ.exe». Откроется окно входа в приложение LoginPage.



Рис. 4 – страница входа в приложение

Здесь необходимо ввести правильный логин и пароль для входа в приложение. В учебном проекте вместе с файлом лежат вшитые логины и пароли, в дальнейшем будет добавлена возможность создания новых пользователей.

После успешного входа в приложение вы увидите перед собой меню выбора функции MainMenuPage.



Рис. 5 – главное меню приложения

Снизу выведена информация о пользователе, однако в будущем под это будет сделано отдельное меню. Чтобы выбрать какой-либо раздел, нужно нажать по соответствующей кнопке левой кнопкой мыши.

У меню добавления расхода и дохода один и тот же интерфейс, и отличаются они совсем незначительно, поэтому приведу пример только для одной страницы. Итак, при выборе функции добавления дохода у нас открывается страница AddIncomePage.

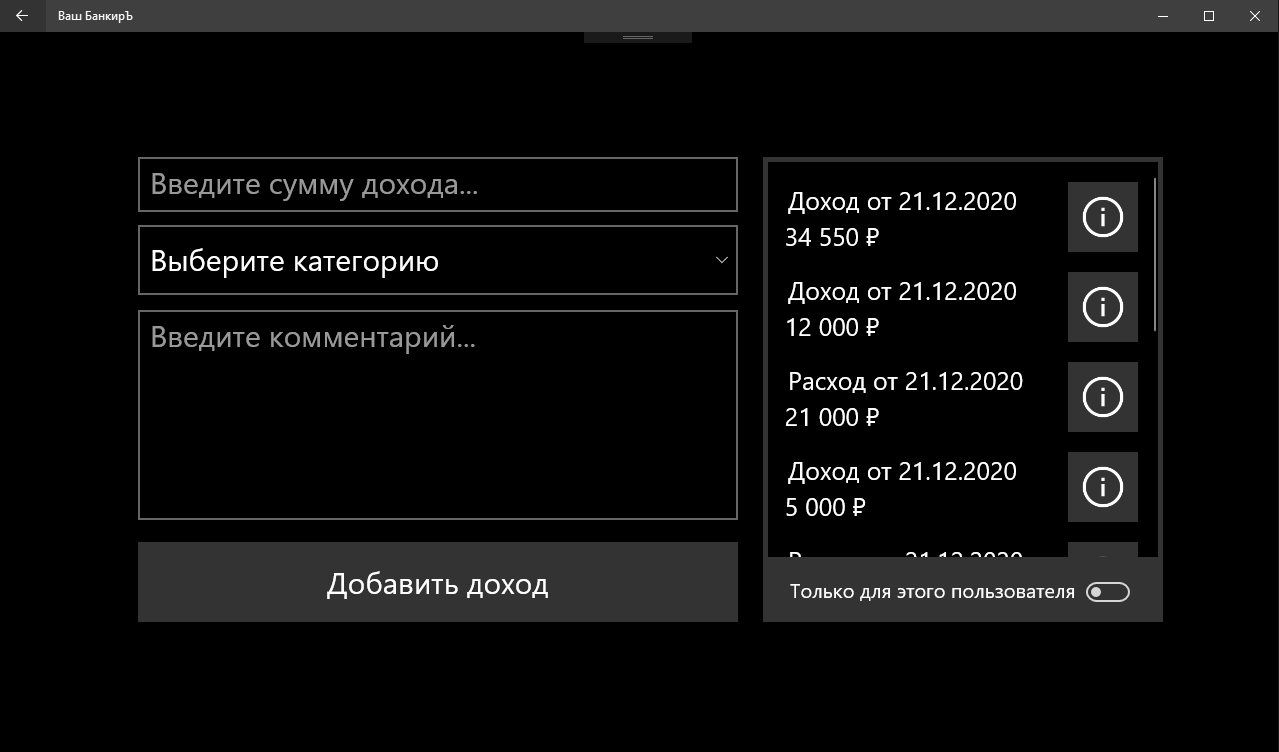


Рис. 6 – страница добавления дохода

Здесь, чтобы добавить доход, нам необходимо ввести сумму дохода и выбрать категорию. Опционально можно написать комментарий. Если комментарий не ввести, то в истории будет отображаться, что комментарий отсутствует.

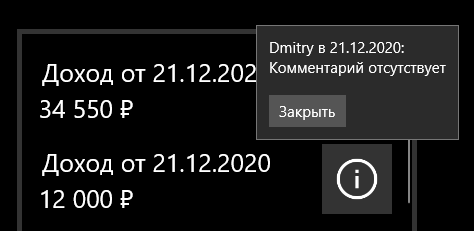


Рис. 7 – демонстрация работы вывода информации

Как мы можем видеть, при нажатии на кнопку (i) мы попадаем в панель информации конкретного дохода или расхода. Тут мы можем посмотреть кто добавил доход или расход, когда, и какой он/она оставили комментарий.

Доход/расход мы добавили. Теперь выйдем обратно и попробуем добавить финансовую цель. Для этого нажимаем в меню на раздел «Финансовые цели». Тут увидим другое меню, меню выбора разделов – для просмотра целей нам нужно нажать на «Просмотр активных целей», а для добавления – на «Добавить цель». Добавим цель.

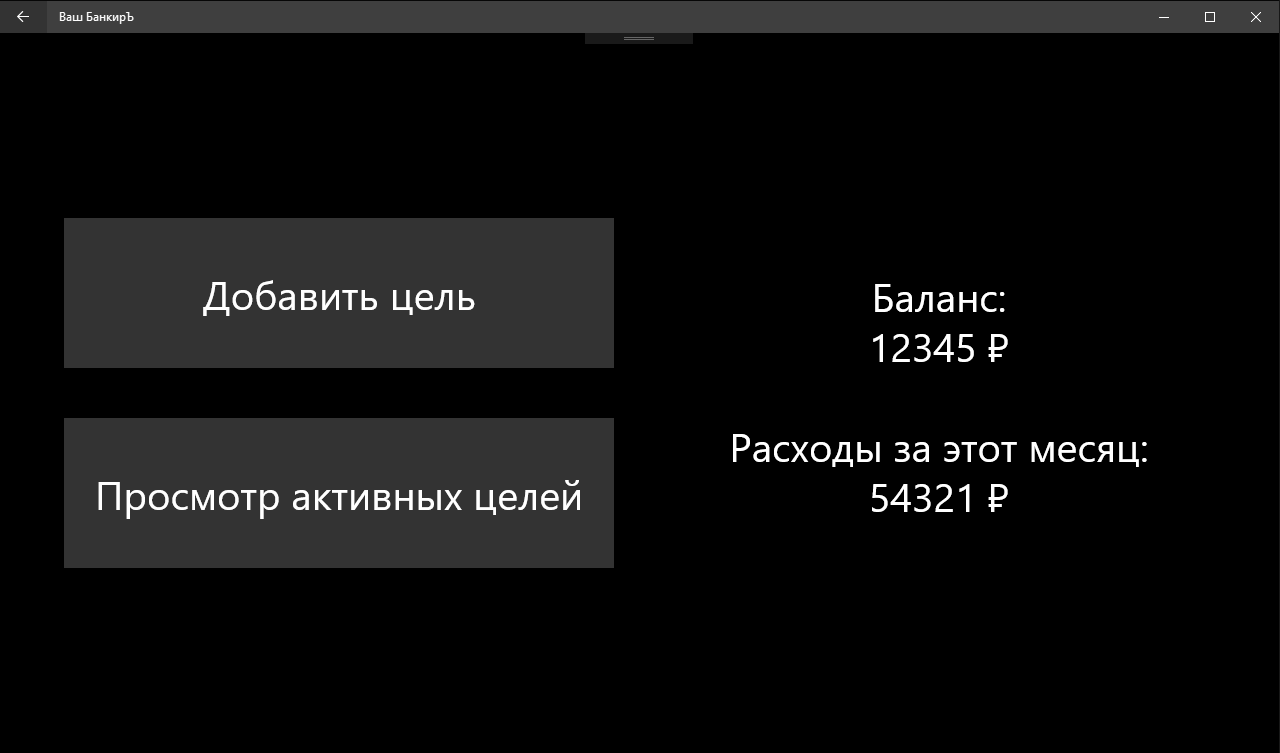


Рис. 8 – страница выбора функции

После нажатия на «Добавить цель» мы увидим страницу, похожую на добавление дохода/расхода.

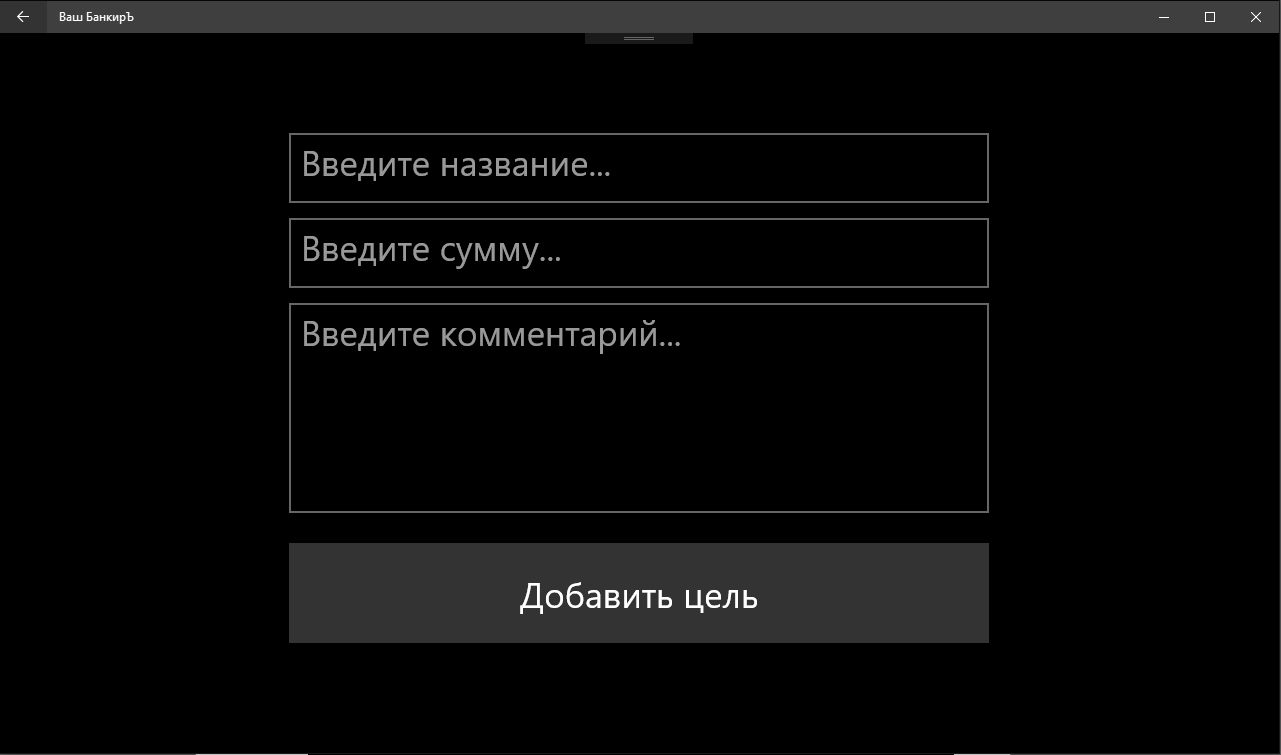


Рис. 9 – страница добавления финансовой цели

Здесь нам нужно ввести название цели и сумму. Также мы можем оставить комментарий, но это опционально. Вернемся обратно и посмотрим на добавленную цель. Для этого заходим в «Просмотр активных целей».

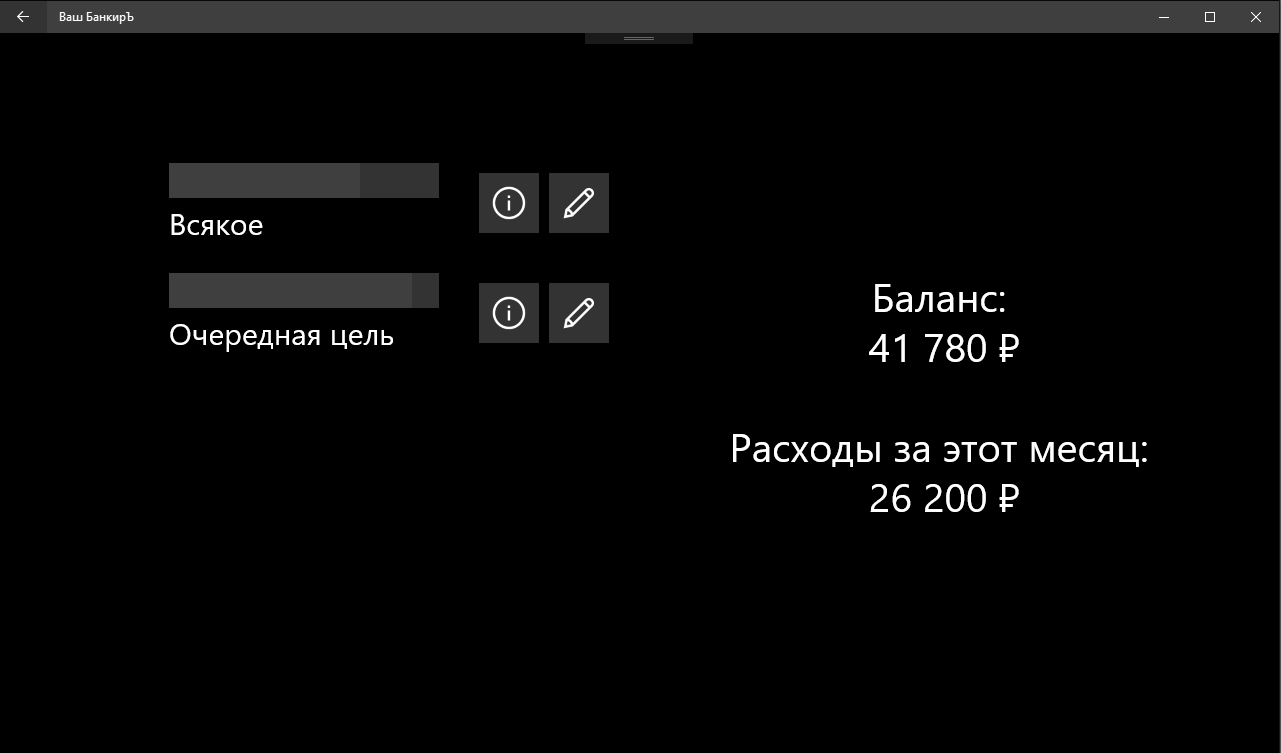


Рис. 10 – страница просмотра целей

Тут видим наши активные цели. Мы можем смотреть информацию о них и редактировать их сумму, а также можем их совсем удалить.

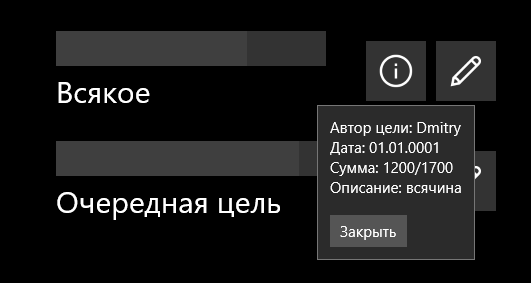


Рис. 11 – демонстрация работы кнопки (i)

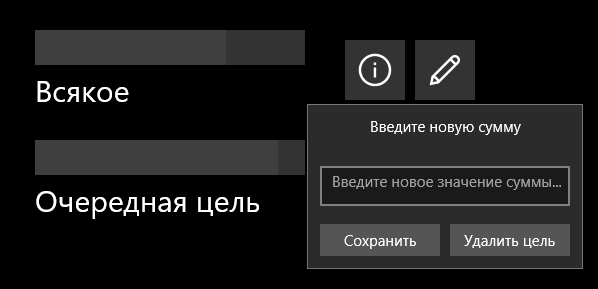


Рис.12 – демонстрация редактирования цели

Ничего тут трогать не будем, перейдем обратно в меню и увидим там настройки. Зайдем в этот раздел и увидим перед собой небольшое меню.

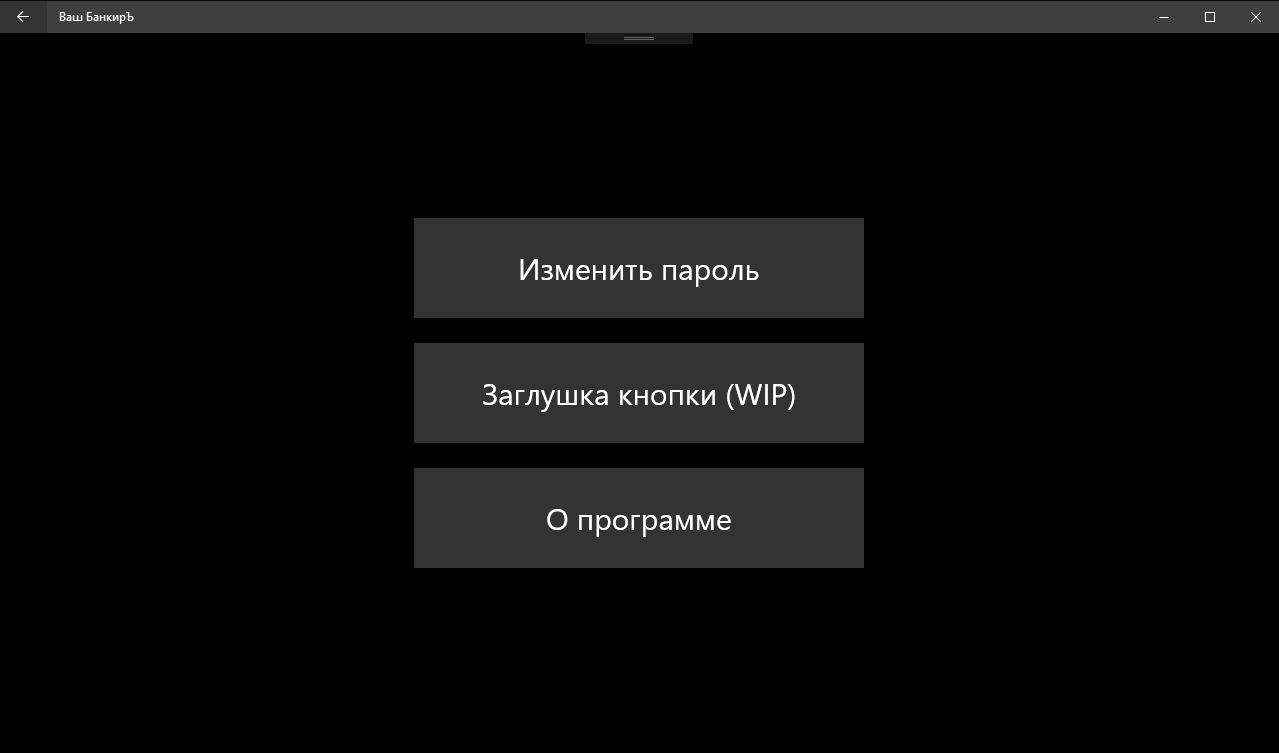


Рис. 13 – страница настроек

При изменении пароля нам открывается окно изменения пароля для активного пользователя.

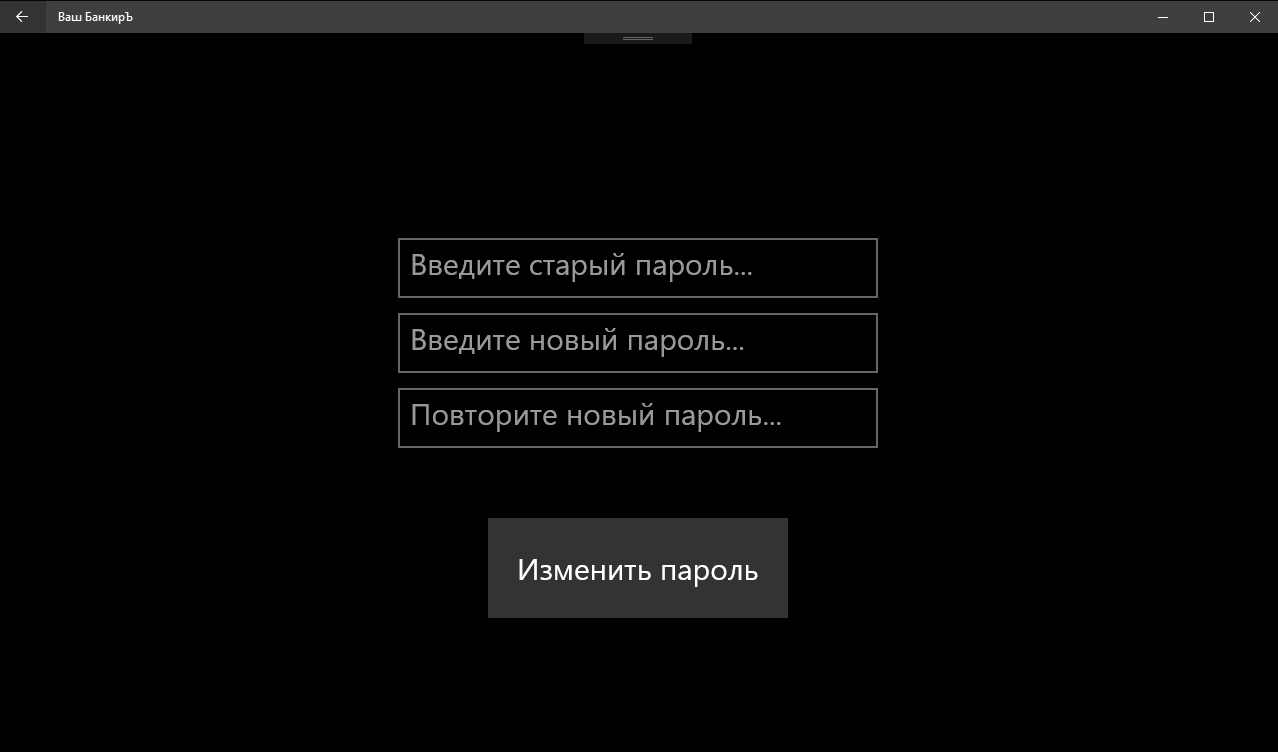


Рис. 14 – страница смены пароля

Выйдем отсюда и зайдем в раздел «О программе». Тут увидим информацию о разработчике, о программе и ссылку на репозиторий GitHub.

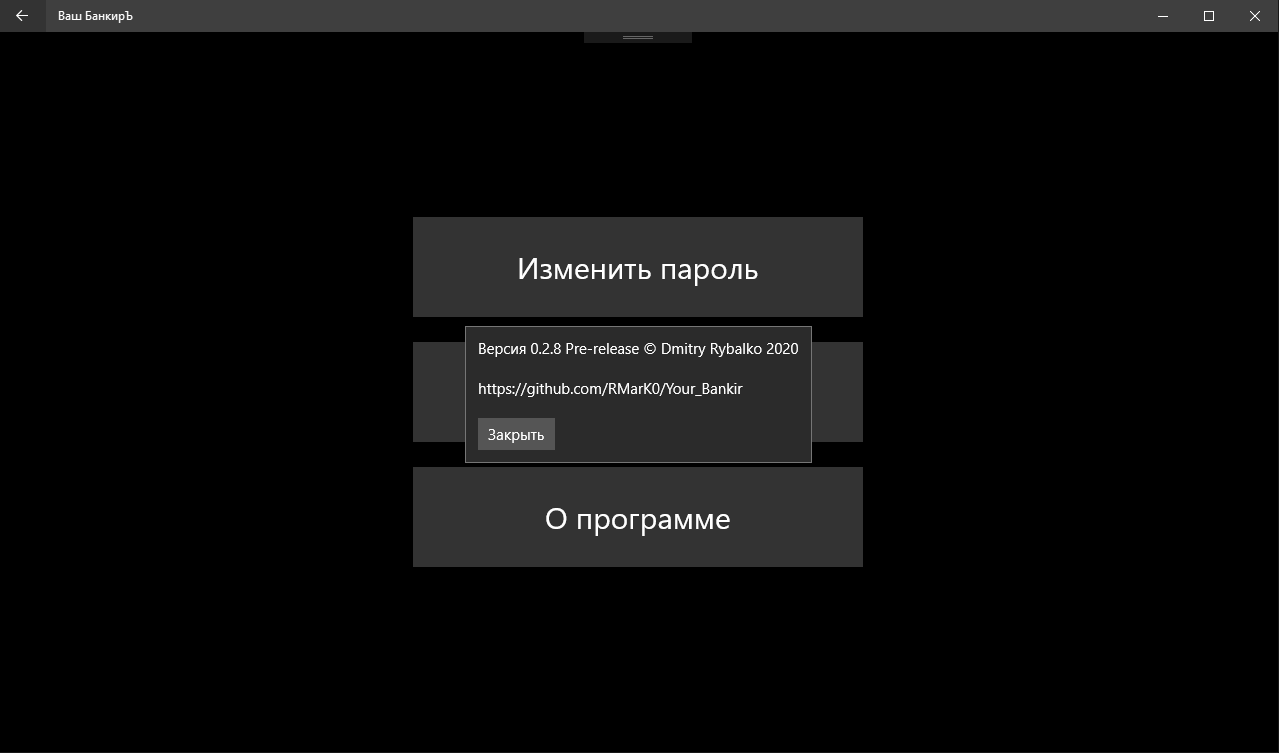


Рис. 15 – демонстрация работы кнопки «О программе»

Кнопка «Выход» осуществляет выход из аккаунта. Это нужно, если мама или бабушка пришла, и вы хотите, чтобы они добавили доход из своего аккаунта не закрывая программу.

# 6. Приложение

Весь код можно найти на GitHub - <https://github.com/RMarK0/Your_Bankir/tree/debugging>