MoneyCalc - депозитный и кредитный калькуляторы

Студенты ФИТ НГУ

группа

Версия 1.0.4

Содержание

[1. Введение 3](#_Toc52317765)

[1.1 Цель 3](#_Toc52317766)

[1.2 Область действия 3](#_Toc52317767)

[1.3 Определения и сокращения 3](#_Toc52317768)

[1.4 Ссылки 3](#_Toc52317769)

[1.5 Краткое описание 3](#_Toc52317770)

[2. Предметная область проекта 4](#_Toc52317771)

[2.1 Существующие проблемы 4](#_Toc52317772)

[2.2 Предполагаемое решение 4](#_Toc52317773)

[3. Требования к программному решению 4](#_Toc52317774)

[3.1 Роли 4](#_Toc52317775)

[3.2 Функциональные требования для роли User 4](#_Toc52317776)

[3.2.1 Выбор кредитного калькулятора 4](#_Toc52317777)

[3.2.2 Выбор депозитного калькулятора 5](#_Toc52317778)

[3.2.3 Вывод результатов расчета депозита 5](#_Toc52317779)

[3.2.4 Вывод результатов расчета кредита 5](#_Toc52317780)

[3.3 Нефункциональные требования 6](#_Toc52317781)

[4. Обзор архитектуры 6](#_Toc52317782)

[4.1.1 Компонентная модель системы 6](#_Toc52317783)

[4.1.1.1 Компонент 1 6](#_Toc52317784)

[4.1.1.2 Компонент 2 6](#_Toc52317785)

[4.1.2 Компоненты сторонних производителей 6](#_Toc52317786)

[4.1.3 Схема развертывания приложения 6](#_Toc52317787)

[5. Допущения и ограничения 6](#_Toc52317788)

[6. Известные проблемы 8](#_Toc52317789)

[6.1 Невысокая производительность приложения 8](#_Toc52317790)

# Введение

## Цель

Данный документ представляет собой техническое описание проекта *MoneyCalc - депозитный и кредитный калькуляторы* и содержит основные требования к разрабатываемой в рамках проекта программной системе и описание архитектуры программного решения.

## Область действия

Документ разработан в рамках проекта MoneyCalc - депозитный и кредитный калькуляторы на основе стандартного шаблона *Inteks SEP* и предназначен для использования студентами ФИТ и преподавателями курса ООАД.

## Определения и сокращения

Таблица 1: Определения и сокращения

|  |  |
| --- | --- |
| Термин | Описание |
| Кредит | Денежная сумма, предоставляемая кредитором на определённый срок и на определённых условиях. |
| Депозит | Сумма денег, переданная лицом [кредитному учреждению](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%B5%D0%B4%D0%B8%D1%82%D0%BD%D0%BE%D0%B5_%D1%83%D1%87%D1%80%D0%B5%D0%B6%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5) с целью получить доход в виде [процентов](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D0%BD%D1%82%D0%BD%D1%8B%D0%B9_%D0%B4%D0%BE%D1%85%D0%BE%D0%B4), образующихся в ходе финансовых операций с вкладом. |
| Аннуитетная схема погашения кредита | Схема ежемесячных выплат, при которой сумма (тело) *кредита* остается неизменной на протяжении всего срока погашения долга, но меняется ее структура. |
| Дифференцированная схема погашения кредита | Схема ежемесячных выплат, при которой размер ежемесячного платежа по погашению кредита постепенно уменьшается к концу периода кредитования. |
| Капитализация процентов | Причисление процентов к сумме вклада, позволяет в дальнейшем осуществлять начисление процентов на проценты. |

## Ссылки

В тексте содержатся ссылки на следующие документы:

Ссылки приводятся в виде [N], где N – номер документа в вышеприведенном списке.

## Краткое описание

Содержание данного документа построено таким образом, чтобы дать ответ на следующие вопросы:

* Какие проблемы предметной области должен решать будущий программный продукт
* Посредством какой функциональности системы будут достигнуто решение проблем предметной области
* Какова архитектура программного решения

Описание предметной области и проблем, для решения которых предназначен будущий программный продукт, приведены в разделе 2.

Раздел 3 содержит описание требований к программному решению, раздел 4 – описание архитектуры выбранного решения.

# Предметная область проекта

Цель нашего проекта – облегчить трудоемкие расчеты платежей как *кредитов*, так и *депозитов*. Согласитесь, это в жизни каждого человека довольно актуально.

## Существующие проблемы

Данный проект решает следующие проблемы:

1. Высокая сложность расчетов.

* Большое количество нюансов предметной области, в которые обычный человек может быть не посвящён.
* Предложить пользователям простое приложение, за которым скрывается сложная реализация расчетов.

2. Недооценка переплат по *кредитам*.

* За процентами люди часто не видят реальных сумм, которые они должны будут кредитору.
* Реализация подсчетов переплат и вывод этой информации в удобном формате

3. Переоценка выгоды *депозитов*.

* Часто люди идут в банк, думая, что это лучший способ вложить деньги, но подсчет реальной выгоды, может направить их в другую сферу инвестиций.
* Реализация подсчетов полученных процентов и вывод этой информации в удобном формате

## Предполагаемое решение

Создание простого и удобного приложения, с помощью которого каждый человек сможет детальную информацию о том или ином *кредите* или *депозите*. Для удобства пользователя вся информация будет представлена в виде таблицы в приложении с возможностью ее сохранения в файл формата CSV.

# Требования к программному решению

Данный раздел описывает требования к программной системе, разрабатываемой в рамках проекта MoneyCalc - депозитный и кредитный калькуляторы.

## Роли

Роль - это что-то (например: другая система) или кто-то (например: человек) вне системы, которые взаимодействуют с ней. В предлагаемой к разработке системе идентифицированы следующие роли:

1. User – пользователь приложения

## Функциональные требования для роли User

### Выбор кредитного калькулятора

При выборе кредитного калькулятора будут доступны следующие поля:

* Сумма (с возможностью выбора валюты)
* Срок (количество месяцев или лет)
* Процентная ставка
* Дата взятия кредита
* Схема выплат (аннуитентная или дифференцированная)
* Опция вывода результатов в csv файл

### Выбор депозитного калькулятора

При выборе депозитного калькулятора будут доступны следующие поля:

* Сумма (с возможностью выбора валюты)
* Срок (количество месяцев или лет)
* Процентная ставка
* Дата открытия депозита
* Периодичность выплат:
* Ежедневно
* Еженедельно
* Раз в месяц в день вклада
* Раз в 3 месяца в день вклада
* Раз в полгода в день вклада
* Раз в год в день вклада
* В первый день месяца
* В последний день месяца
* В последний день года
* В конце срока
* Включение/отключение капитализации процентов
* Опция вывода результатов в csv файл

### Вывод результатов расчета депозита

Вывод результатов идет в таблицу в приложении, а также в csv файл если User выбрал данную опцию.

В таблице имеются следующие колонки:

* + - Номер выплаты
    - Дата выплаты
    - Сумма начисленных процентов
    - Удержанный налог
    - Сумма с учетом налога
    - Баланс вклада

Так же идет обобщения результатов таких как:

* + - Баланс вклада на конец срока
    - Сумма начислений
    - Налог
    - Сумма начислений с учетом налога

### Вывод результатов расчета кредита

Вывод результатов идет в таблицу в приложении, а также в csv файл если User выбрал данную опцию.

В таблице имеются следующие колонки:

* + - Номер выплаты
    - Дата выплаты
    - Сумма
    - Часть основного долга
    - Часть погашения процентов
    - Остаток основного долга

Так же идет обобщения результатов таких как:

* + - Общая сумма выплат
    - Переплата

## Нефункциональные требования

* Точность расчетов
* Unit-тестирование расчетов

# Обзор архитектуры

Этот раздел описывает архитектуру системы.

### Компонентная модель системы

В приложении используются следующие пакеты: calculator.deposit, calculator.credit, ui, csv

Все расчеты производятся в классах пакетов calculator.deposit и calculator.credit.

Для вывода результатов в файл используются классы пакета csv.

Для взаимодействия с пользователем, используется графический интерфейс из пакета ui, написанный на javaFX. Взаимодействие с пакетом csv идет через классы пакета ui.

#### Ui

В пакете ui собраны классы реализующие графический интерфейс:

* ResultOutputController – интерфейс вывода результатов вычислений
* CreditResultOutputController – класс, реализующий интерфейс ResultOutputController и заполняющий экран вывода результатов расчета кредита в приложении. Контроллер связан с FXML файлом
* DepositResultOutputController – класс, реализующий интерфейс ResultOutputController и заполняющий экран вывода результатов расчета депозита в приложении. Контроллер связан с FXML файлом
* CreditController – класс, управляющий формой ввода информации о кредите. Контроллер связан с FXML файлом
* DepositController - класс, управляющий формой ввода информации о депозите. Контроллер связан с FXML файлом

#### csv

В пакете csv собраны классы для работы с CSV файлами:

* CSVWriter – класс, предназначенный для вывода строк в csv файл
* CreditCSVFormatter – класс, служащий для форматирования информации по кредиту в формат CSV
* DepositCSVFormatter - класс, служащий для форматирования информации по депозиту в формат CSV

#### Calculator.deposit

В пакете calculator.deposit собраны класса необходимые для расчетов депозитов, такие как:

* FrequencyOfPayment – класс перечислений, которые задают периодичность выплат. Используется для вычисления следующей даты выплаты
* DepositInfo – класс, включающий предоставленную пользователем информацию о депозите
* DepositPayment – класс, содержащий информацию об отдельной выплате по депозиту
* Deposit – класс, который получает DepositInfo и сразу при создании вычисляет список всех DepositPayment, а в дальнейшем служит для получение информации об отдельной выплате на протяжении всего срока депозита

#### Calculator.credit

В пакете calculator.credit собраны класса необходимые для расчетов кредитов, такие как:

* CreditType – класс перечислений, содержащий ANNUITY, DIFFERENTIATED
* CreditInfo – класс, включающий предоставленную пользователем информацию о кредите
* CreditPayment – класс, содержащий информацию об отдельном платеже по кредиту
* Credit – интерфейс представляющий кредит
* AnnuityCredit – класс реализует интерфейс Credit и получает CreditInfo, и сразу при создании вычисляет список всех CreditPayment при аннуитетной схеме погашения, а в дальнейшем служит для получение информации об отдельном платеже на протяжении всего срока кредитования
* DifferentiatedCredit - класс реализует интерфейс Credit и получает CreditInfo, и сразу при создании вычисляет список всех CreditPayment при дифференцированной схеме погашения, а в дальнейшем служит для получение информации об отдельном платеже на протяжении всего срока кредитования

### Компоненты сторонних производителей

* JavaFX – библиотека для GUI приложения
* JUnit 5 – библиотека для тестирования
* Logback – библиотека для логирования

### Схема развертывания приложения

Приложение является десктопным и запускается на одной машине. Приложение собирается при помощи системы сборки – Gradle. Так же есть пакет с тестами, которой не попадает в файлы исходного кода приложения

# Допущения и ограничения

При разработке проекта принято допущение, что вывод результатов вычислений с вещественными числами ограничивается до двух знаков после запятой. Также приложение не сохраняет результаты предыдущих вычислений не записанных в csv файл.

# Известные проблемы

Ниже приводятся известные на данный момент проблемы и недоработки выработанного программного решения, а также возможные пути их устранения в последующих итерациях проекта.

## Невысокая производительность приложения

|  |  |
| --- | --- |
| Проблема | Небольшая погрешность вычислений из-за сложных расчетов с применением чисел с плавающей точкой |
| Ранг | 3 (низкий) |
| Влияние на проект | Невозможность провести расчеты с точностью до копейки |
| Пути решения | Использовать BigDecimal |

Лист регистрации изменений

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Версия | Описание | Автор |
| 06.09.2020 | 1.0.1 | Добавлен пункт 2 (Project Vision), внесение 5 терминов в пункт 1.3 | Гайдамака А.В, Ланчуковская К.C. |
| 30.09.2020 | 1.0.2 | Добавлен пункт 3 (Требования к программному решению), а также изменено название проекта. | Гайдамака А.В, Ланчуковская К.C. |
| 22.11.2020 | 1.0.3 | Добавлены пункты 4 и 5 | Гайдамака А.В, Ланчуковская К.C. |
| 09.12.2020 | 1.0.4 | Добавлен пункт 6 | Гайдамака А.В, Ланчуковская К.C. |
|  |  |  |  |

Лист регистрации проверок

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Версия | Описание | Автор |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |