МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «УФИМСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» (УГНТУ)

Кафедра «Цифровые технологии и моделирование»

**КУРСОВАЯ РАБОТА**

по дисциплине «Проектирование программного обеспечения»

на тему

«Проектирование программного обеспечения подсистема управления банком»

Выполнил:

студент гр. БИЦ-20-01 Багаутдинов М.Р.

Проверила:

к.п.н., доцент, Старцева О.Г.

Уфа 2023

Содержание

**Введение3**

**Глава 1.Теоретические основы4**

Описание предметной области4

Модель бизнес-процессов(IDEF0)5

Предложения для автоматизации бизнес-процессов6

**Глава 2.Практическая реализация3**

Система требований и атрибуты качества7

UML диаграммы требований9

UML диаграммы требований иерархии классов10

Описание работы и размещение ПО11

Тестирование13

**Заключение17**

**Список литературы18**

**Введение:**

Курсовая работа посвящена теме проектирования программного обеспечения (ПО). Проектирование ПО играет ключевую роль в разработке современных информационных систем и приложений, поскольку это этап, на котором определяется архитектура, структура и функциональность программного продукта. Качественное проектирование ПО является фундаментальным аспектом успешного разработки, обеспечивая его эффективность, надежность и расширяемость.

В данной курсовой работе будут рассмотрены основные аспекты и методологии проектирования ПО. В частности, будут изучены различные подходы к проектированию, моделирование бизнес-процессов и требований, а также различные виды диаграмм UML для визуализации и описания архитектуры и компонентов ПО.

Целью работы является изучение и анализ процесса проектирования ПО, а также разработка конкретного проекта с применением полученных знаний. В результате работы будет создана модель проектирования программного продукта, включающая диаграммы классов, последовательности, состояний, активностей, компонентов и развертывания. Кроме того, будет разработана система требований к ПО, а также система тестов для проверки реализации требований и соответствия атрибутам качества.

Данная курсовая работа имеет практическую значимость, поскольку позволяет применить полученные знания и навыки в процессе разработки реального программного продукта. Также она способствует более глубокому пониманию процесса проектирования ПО и его важности в разработке современных информационных систем.

В ходе работы будут использованы различные методы и инструменты, включая UML-диаграммы, моделирование бизнес-процессов, требования и тестирование. Каждый этап проектирования будет подробно рассмотрен и документирован, чтобы обеспечить ясность и полноту представленной информации.

Итак, данная курсовая работа посвящена проектированию программного обеспечения, включая изучение основных концепций, методологий и инструментов, а также разработку конкретного проекта с использованием полученных знаний. Она имеет целью показать важность проектирования в разработке ПО и предоставить практические навыки для успешного выполнения про

**ГЛАВА 1. Теоретические основы**

**Описание предметной области**

Подсистема управления банком - это комплексное программное решение, разработанное для автоматизации и управления операциями в банковской сфере. Она обеспечивает эффективное функционирование банка, обработку финансовых транзакций, управление счетами клиентов и обеспечение безопасности финансовых данных.

Одной из основных функций подсистемы управления банком является обработка операций по снятию и пополнению счетов клиентов. Клиенты имеют возможность осуществлять снятие наличных со своих счетов через банкоматы, а также пополнять счета путем внесения наличных или осуществления банковских переводов.

Приложение также обеспечивает функцию перевода денежных средств между счетами. Клиенты могут осуществлять переводы на другие счета внутри банка или на счета клиентов других банков. Система управления банком обрабатывает эти запросы на перевод, проверяет доступные средства на счете отправителя, осуществляет перевод и обновляет информацию о балансах счетов.

Подсистема управления банком также включает функциональность регистрации и входа для клиентов. Новые клиенты могут зарегистрироваться в системе, предоставив необходимую информацию и создавая учетную запись. Зарегистрированные клиенты могут войти в систему с использованием своих учетных данных, таких как логин и пароль, и получить доступ к своим счетам, балансам, истории операций и другим банковским услугам.

Подсистема управления банком также обеспечивает важные функции, такие как управление кредитами, обработка платежей, генерация отчетов о финансовых операциях, аналитика и контроль финансовой активности.

В целом, подсистема управления банком предоставляет банку инструменты для эффективного и безопасного управления банковскими операциями, обеспечивая удобство и удовлетворение потребностей клиенто

**Модель бизнес-процессов(IDEF0)**

Управление банком

1. Обработка операций

1.1. Снятие со счета

1.2.Пополнение счета

1.3. Перевод средств другому человеку

1. Регистрация и вход

2.1. Регистрация нового пользователя

2.2. Вход в систему

1. Управление клиентскими счетами

3.1. Создание нового счета

3.2. Закрытие счета

3.3. Обновление информации о счете

1. Генерация отчетов

4.1. Формирование отчета о балансе счета

**Предложения для автоматизации бизнес-процессов**

Для автоматизации бизнес-процессов в подсистеме управления банком, можно предложить использование программного продукта, такого как "Bank Management System" - комплексное программное решение, специально разработанное для автоматизации и управления операциями в банковской сфере. Этот программный продукт предлагает широкий набор функциональности, которая поможет оптимизировать бизнес-процессы и улучшить эффективность работы банка.

Вот некоторые особенности и возможности, которые предоставляет "Bank Management System":

1. Управление счетами клиентов: Приложение позволяет банку легко создавать, обновлять и закрывать счета клиентов. Клиенты могут просматривать баланс своих счетов, историю транзакций и выполнять различные операции, связанные с их счетами.
2. Обработка операций: Программный продукт предоставляет возможность обрабатывать операции по снятию денег со счетов, пополнению счетов и переводам между счетами. Банк может легко проводить эти операции с помощью встроенных механизмов проверки баланса и подтверждения операций.
3. Управление кредитами: Приложение позволяет банку автоматизировать процессы управления кредитами, включая одобрение заявок, выдачу кредитов и контроль за погашением задолженности. Это помогает банку эффективно управлять кредитным портфелем и минимизировать риски.
4. Обработка платежей: Программный продукт облегчает процесс приема платежей от клиентов и осуществления выплат. Он интегрируется с различными системами платежных шлюзов, что обеспечивает безопасную и надежную обработку платежей.
5. Генерация отчетов: "Bank Management System" предлагает возможность генерации разнообразных отчетов, таких как отчеты о финансовых операциях, отчеты о балансе счетов, отчеты о кредитах и другие. Эти отчеты помогают банку анализировать финансовые данные, контролировать процессы и принимать информированные решения.
6. Регистрация и вход: Приложение предоставляет механизмы регистрации новых клиентов и аутентификации уже зарегистрированных пользователей. Это позволяет клиентам получить доступ к своим счетам и использовать функциональность системы управления банком.

**ГЛАВА 2. Практическая реализация**

**Система требований и атрибуты качества**

a. Бизнес-требования:

* Программный продукт должен обеспечивать автоматизацию банковских операций и управление счетами клиентов.
* Приложение должно быть легким в использовании и интуитивно понятным для клиентов и сотрудников банка.
* Программный продукт должен быть надежным и обеспечивать безопасность финансовых данных клиентов.
* Приложение должно поддерживать масштабируемость, чтобы справляться с увеличением числа клиентов и объемом транзакций.
* Программный продукт должен соответствовать законодательству и нормативным требованиям в области банковской деятельности.

b. Функциональные требования:

* Система должна позволять клиентам снимать деньги со своих счетов, пополнять счета и осуществлять переводы другим людям.
* Приложение должно предоставлять возможность регистрации новых пользователей и аутентификации уже зарегистрированных пользователей.
* Система должна обеспечивать управление клиентскими счетами, включая создание новых счетов, закрытие счетов и обновление информации о счетах.
* Программный продукт должен позволять банку одобрять заявки на кредит, выдавать кредиты и отслеживать их погашение.
* Система должна обеспечивать обработку платежей, включая прием платежей от клиентов и осуществление выплат клиентам.

c. Требования к интерфейсу:

* Интерфейс приложения должен быть интуитивно понятным, простым в использовании и приятным для пользователя.
* Пользовательский интерфейс должен быть отзывчивым и обеспечивать плавную навигацию между различными функциональными модулями.
* Интерфейс должен быть адаптивным и поддерживать различные устройства, включая компьютеры, планшеты и смартфоны.

d. Требования к надежности работы:

* Программный продукт должен быть стабильным и надежным, обеспечивая непрерывную работу системы даже при высокой нагрузке.
* Система должна иметь механизмы резервного копирования данных и восстановления после сбоев.
* Программный продукт должен обеспечивать защиту от несанкционированного доступа и вредоносных атак.

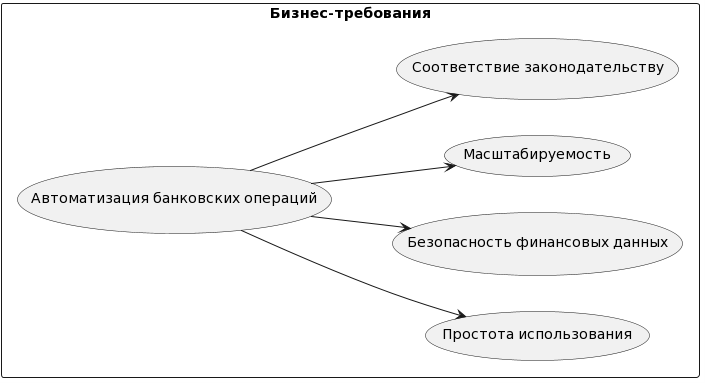
e. Системные требования и требования к совместимости:

* Программный продукт должен быть разработан для определенных операционных систем, таких как Windows, macOS или Linux, с указанием требуемой версии.
* Система должна быть совместима с используемыми в банке базами данных, сетевыми протоколами и другими программными средствами.
* Программный продукт должен быть легко интегрируемым с существующими банковскими системами, такими как системы учета и аналитики.

Эти требования к создаваемому программному продукту и атрибуты качества помогут обеспечить разработку и внедрение эффективной и надежной системы управления банком.

**UML диаграммы требований**

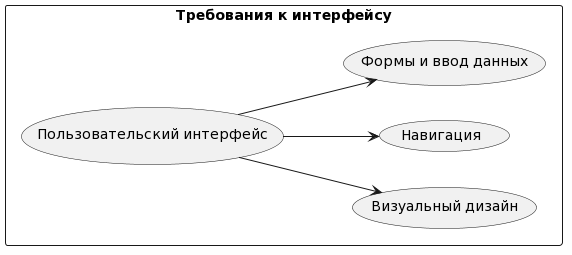
Бизнес-требования:



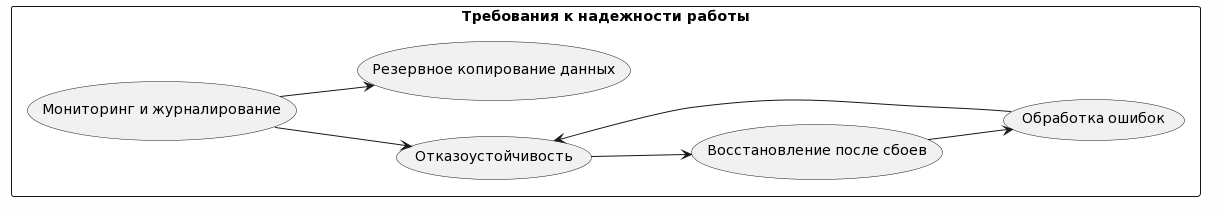
Функциональные требования:



Требования к интерфейсу:



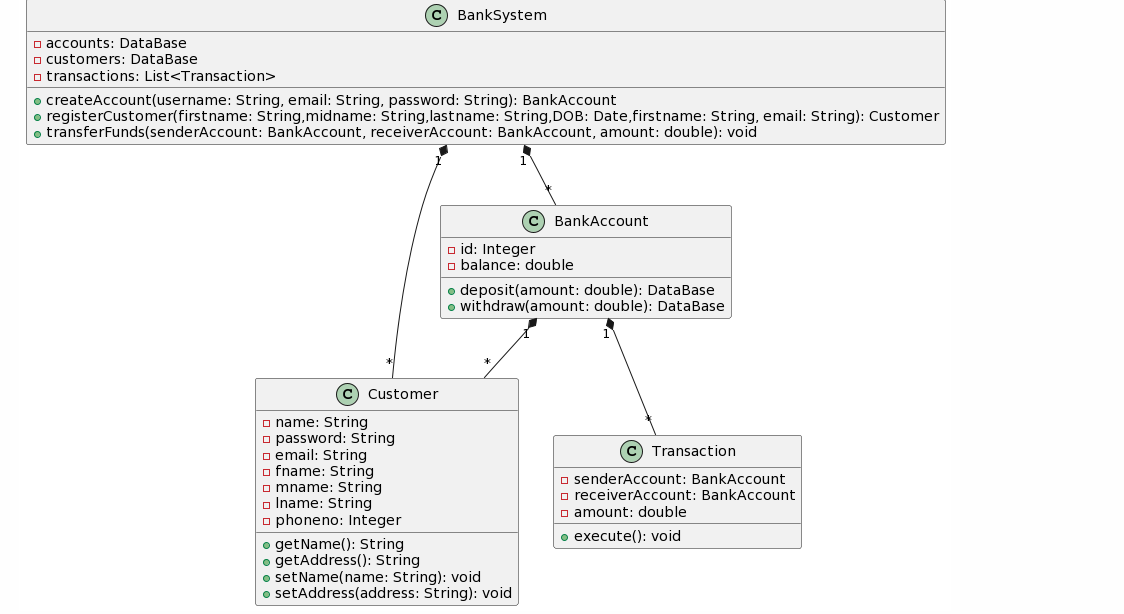
Требования к надежности работы:



Системные требования и требования к совместимости:



**UML диаграмма иерархии классов**



**Описание работы и размещение ПО (UML-диаграммы):**

диаграмма последовательности (Sequence Diagram);

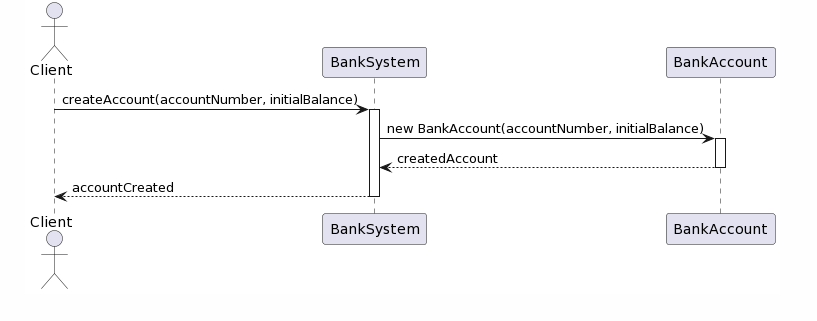


диаграмма состояний (State Diagram);

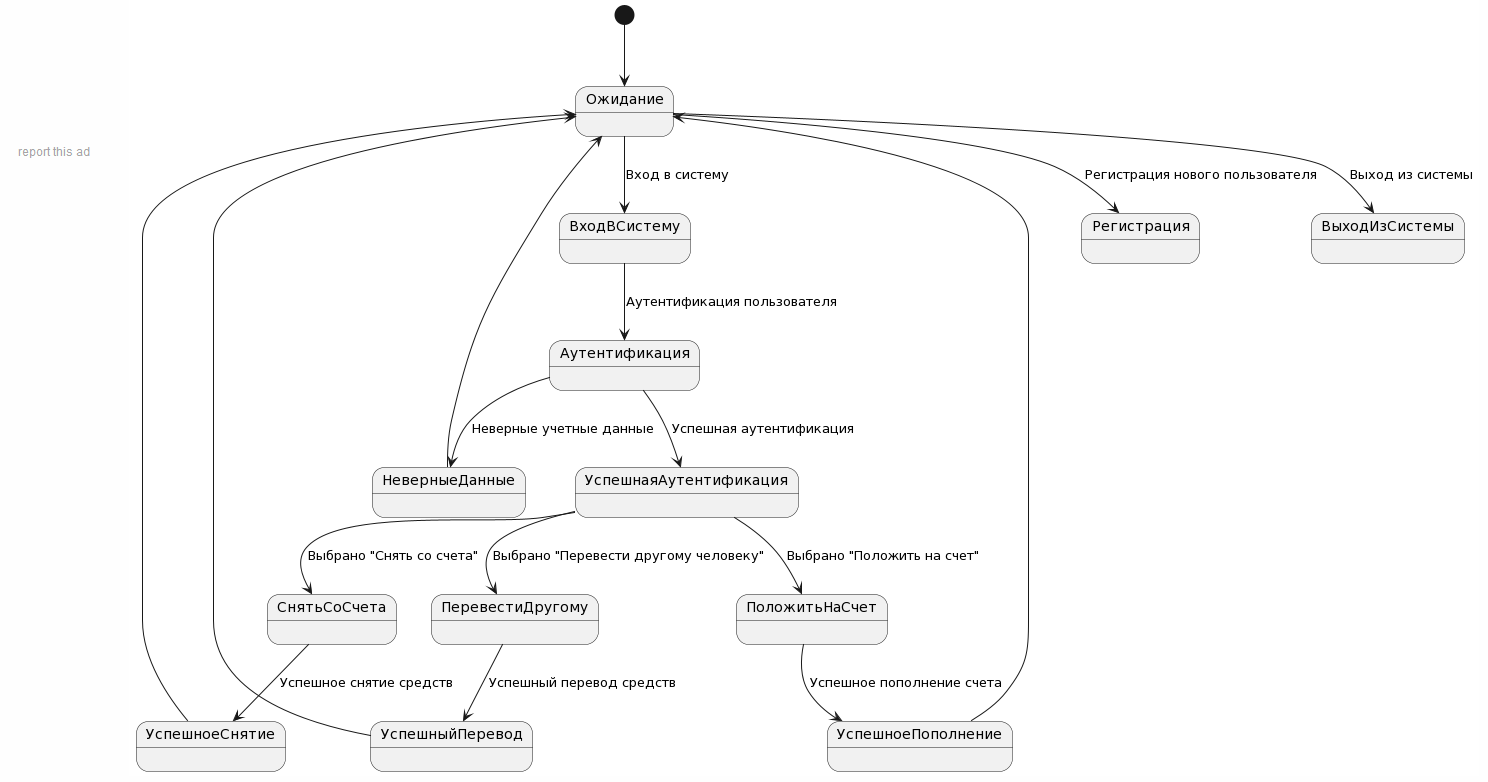


диаграмма активности (Activity Diagram);

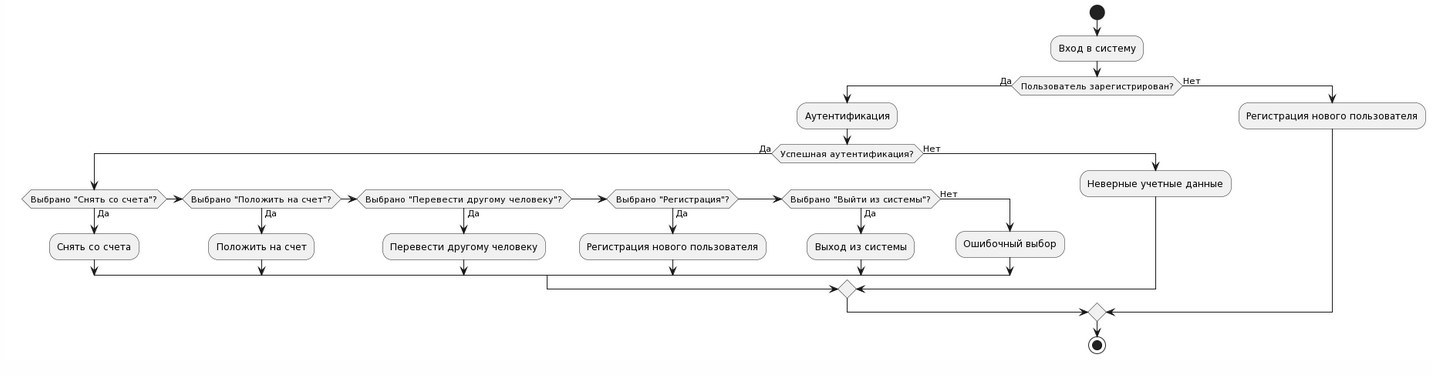


диаграмма взаимодействия (Cooperation Diagram);

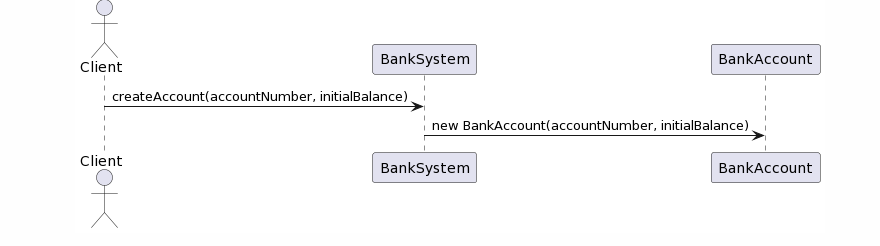
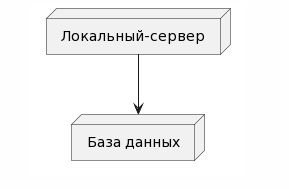


диаграмма компонентов (Component Diagram);



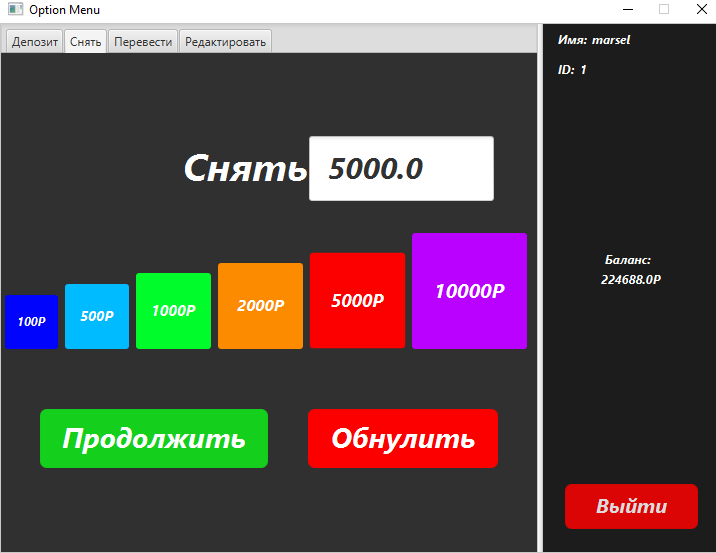
диаграмма развертывания (Deployment Вiagram).



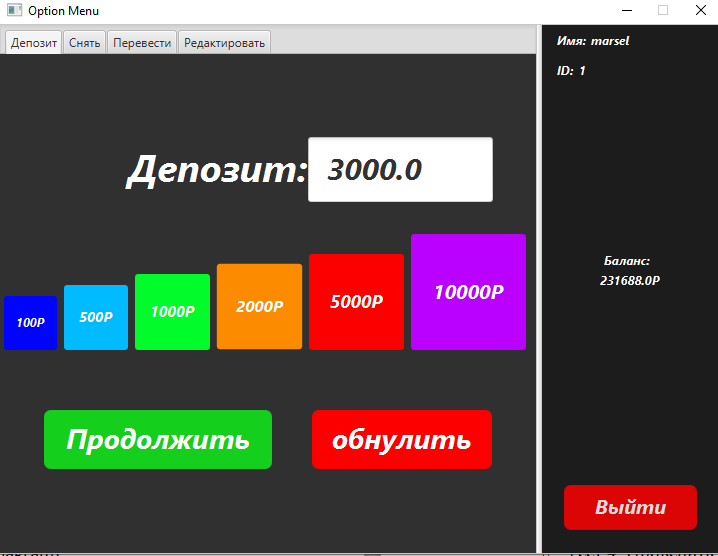
**Тестирование**

1. Бизнес-требование: Система должна обеспечивать возможность снятия и внесения средств на счет.

Тест 1: Проверить, что пользователь может успешно снять средства со счета.

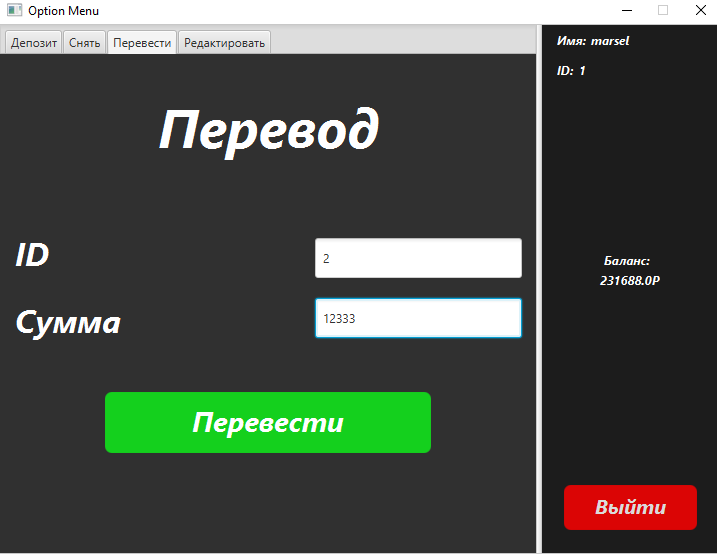


Тест 2: Проверить, что пользователь может успешно внести средства на счет.

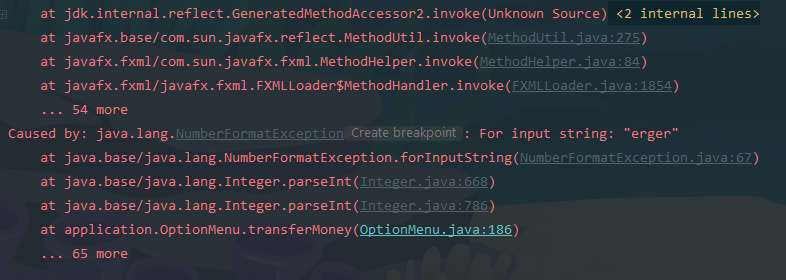


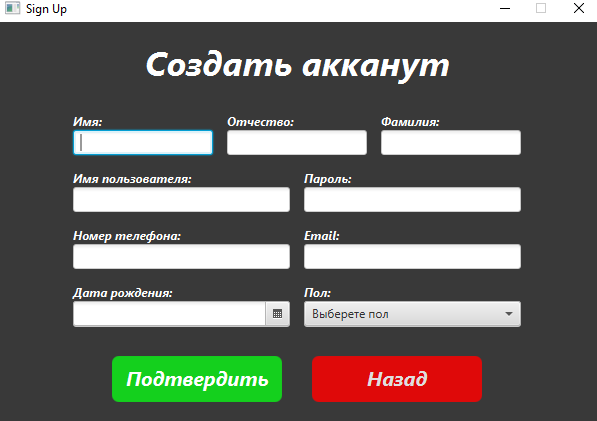
1. Функциональное требование: Система должна обеспечивать возможность выполнения перевода средств на другой счет.

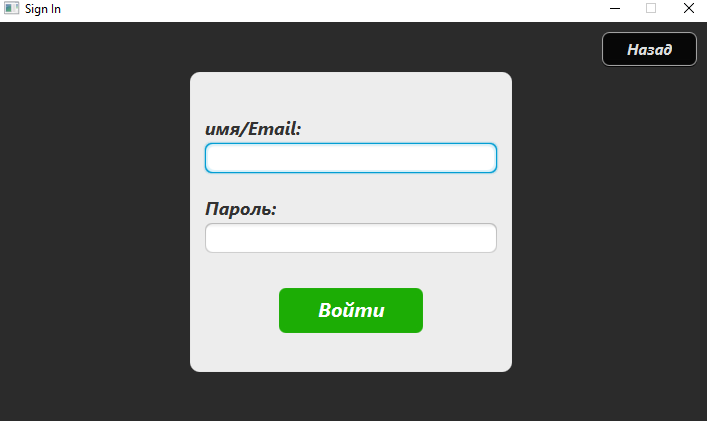
Тест 3: Проверить, что пользователь может успешно выполнить перевод средств на другой счет с правильными данными.



Тест 4: Проверить, что система обрабатывает ошибки при некорректных данных перевода.



Тест 5:Проверить, что система может регистрировать новых пользователей. 

Тест 6:Проверить, что система может проверять данные и входить в аккаунт. 

1. Требование к интерфейсу: Интерфейс должен быть интуитивно понятным и удобным в использовании.

Тест 7: Проверить, что основные элементы интерфейса (кнопки, поля ввода) отображаются корректно и реагируют на действия пользователя.

Проверил основные элементы интерфейса (кнопки, поля ввода),они отображаются корректно и реагируют на действия пользователя.

Тест 8: Проверить, что пользователь может легко выполнить основные операции (снятие средств, внесение средств, перевод) через интерфейс.

Проверил,что пользователь может легко выполнить основные операции (снятие средств, внесение средств, перевод) через интерфейс.

1. Требование к надежности работы: Система должна быть отказоустойчивой и обеспечивать восстановление после сбоев.

Тест 9: Провести нагрузочное тестирование для проверки стабильности и производительности системы при большом количестве одновременных пользователей.

Приложение не ресурсоемкое, поэтому запуститься и будет работать на любом устройстве со слабыми комплектующими

Тест 10: Проверить, что система корректно восстанавливается после возможных сбоев и ошибок.

Для проверки возможных сбоев необходимо в поле для цифровых значений ввести любую буквенную комбинацию.

Приложение спокойно восстановиться так как за это отвечает большое количество обработчиков событий и сам компилятор

1. Требования к совместимости: Система должна быть совместима с различными операционными системами и браузерами.

Тест 11: Проверить, что система работает корректно на различных операционных системах (Windows, macOS, Linux).

Так как ПО написано полностью на Java, то данное приложение запуститься на любых системах и дистрибутивах.

**Заключение:**

В ходе выполнения курсовой работы были рассмотрены основные аспекты и методологии проектирования программного обеспечения (ПО). Были изучены различные подходы к проектированию, включая принципы SOLID, моделирование бизнес-процессов и требований, а также использование диаграмм UML для визуализации и описания архитектуры ПО.

Целью работы было изучение процесса проектирования ПО и его важности в разработке современных информационных систем. В результате была разработана модель проектирования программного продукта, включающая диаграммы классов, последовательности, состояний, активностей, компонентов и развертывания. Также была создана система требований к ПО и система тестов для проверки реализации требований и соответствия атрибутам качества.

Курсовая работа имеет практическую значимость, поскольку позволяет применить полученные знания и навыки в разработке реального программного продукта. В процессе выполнения работы были использованы различные методы и инструменты, включая UML-диаграммы, моделирование бизнес-процессов, требования и тестирование.

Процесс проектирования ПО является фундаментальным этапом в разработке, поскольку он определяет архитектуру, структуру и функциональность программного продукта. Качественное проектирование ПО обеспечивает его эффективность, надежность и расширяемость. От правильного проектирования зависит успешность всего проекта.

В заключение, выполнение данной курсовой работы позволило углубить знания в области проектирования ПО и применить их на практике. Полученные результаты и разработанные модели и системы могут быть использованы в дальнейшей разработке программного продукта, обеспечивая его качество и соответствие требованиям.

**Список использованной литературы**

1. Куликов А.А., Колесова О.В. Программная инженерия: Методы и подходы. – СПб.: Питер, 2021.

2. Соммервиль И., Инженерия программного обеспечения. - М.: Техносфера, 2022.

3. Бехтерев М.В. Проектирование и разработка программных систем. – М.: ДМК Пресс, 2023.

4. Макконнелл С. Совершенный код: Мастер-класс. – М.: ДМК Пресс, 2018.

5. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Д. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – СПб.: Питер, 2022.

6. Бейзер Б. Тестирование программного обеспечения. – М.: Издательский дом «Вильямс», 2023.

7. Лукин А.В., Лебедева Л.С. Методы тестирования программного обеспечения. – М.: Финансы и статистика, 2021.

8. Болдырев А.А., Болдырев Д.А. "Проектирование программных систем управления предприятием в сфере общественного питания". Издательство Самарского государственного технического университета, 2021.

10. Склярова Н.В. "Основы проектирования информационных систем управления предприятиями общественного питания". Издательство Южного федерального университета, 2022.

# 11. Игорь Голдовский: “Банковские микропроцессорные карты”.