

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

Отчет к лабораторной работе №2

на тему

**МОДЕЛИРОВАНИЕ БАЗЫ ДАННЫХ И ПРОЕКТИРОВАНИЕ
ПРОГРАММНОГО СРЕДСТВА**

Студент

В.О. Горожанкин

Руководитель

Е.В. Тушинская

Минск 2023

СОДЕРЖАНИЕ

Введение.....	3
1 Моделирование базы данных.....	4
1.1 MongoDB.....	4
1.2 Разработка инфологической модели базы данных.....	5
2 Проектирование программного средства	6
Заключение	9

ВВЕДЕНИЕ

В современном мире интернет-банкинг приобретает нарастающую популярность как средство управления финансами. Клиенты банков весьма удобно могут получать доступ к своим счетам, осуществлять финансовые операции, включая переводы и оплату счетов, не выходя из уютных стен своего дома. Тем не менее, для обеспечения бесперебойной функциональности данной системы, необходима надежная и эффективная база данных, которая служит основой для хранения критически важной информации о клиентах, их счетах, транзакциях и прочих ключевых данных.

Целью данной лабораторной работы является разработка структуры базы данных, предназначенной для использования в интернет-банкинге, а также создание диаграмм последовательностей и компонентов. Эти диаграммы предоставят возможность визуализировать алгоритмы функционирования определенных банковских операций и компонентов системы, что, в свою очередь, обеспечит более подробное исследование и описание взаимодействия различных элементов данной системы.

1 МОДЕЛИРОВАНИЕ БАЗЫ ДАННЫХ

1.1 MongoDB

MongoDB – это документоориентированная система управления базами данных, пользующаяся популярностью в мире разработки программного обеспечения. Эта система баз данных также имеет открытый исходный код и предоставляет альтернативу коммерческим реляционным и NoSQL базам данных.

Иногда MongoDB сравнивают с бесплатным аналогом системы управления базами данных Oracle. Обе системы предназначены для обработки больших объемов данных и обеспечения высокой производительности, но у них есть существенные различия. MongoDB отличается своей документоориентированной структурой данных и предоставляет разработчикам мощные инструменты для хранения и обработки информации. Она позволяет хранить данные в формате BSON (Binary JSON), что делает ее особенно удобной для хранения и обработки документов с разной структурой.

1.2 Разработка инфологической модели базы данных

Исходя из необходимости использования в проектируемом приложении базы данных, разработаем ее инфологическую модель. Для создания данной модели возьмем за основу предметную область проекта.

Предметная область разрабатываемого программного средства включает в себя следующие сущности и их атрибуты:

- а) пользователь:
 - имя;
 - хешированный пароль;
 - электронная почта;
 - роль;
 - список карт;
 - список общих счетов;
 - список кредитных счетов;
- б) общий счет:
 - список участников;
 - номер счета
 - баланс;
 - тип счета;

в) кредит:

- сумма кредита
- процентная ставка;
- количество платежей;
- ежемесячный платеж;
- расчетная сумма;
- статус

г) транзакция:

- отправитель;
- получатель;
- тип;
- сумма;
- дата
- описание.

2 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО СРЕДСТВА

Исходя из выдвинутых функциональных требований, проектируемое программное обеспечение предполагает совершение транзакций между пользователями, создание общих счетов и возможность оформления кредитов.

Для описания логики сценария использования данных функциональных требований были использованы диаграммы последовательностей, представленные на рисунках 2.1, 2.2 и 2.3.

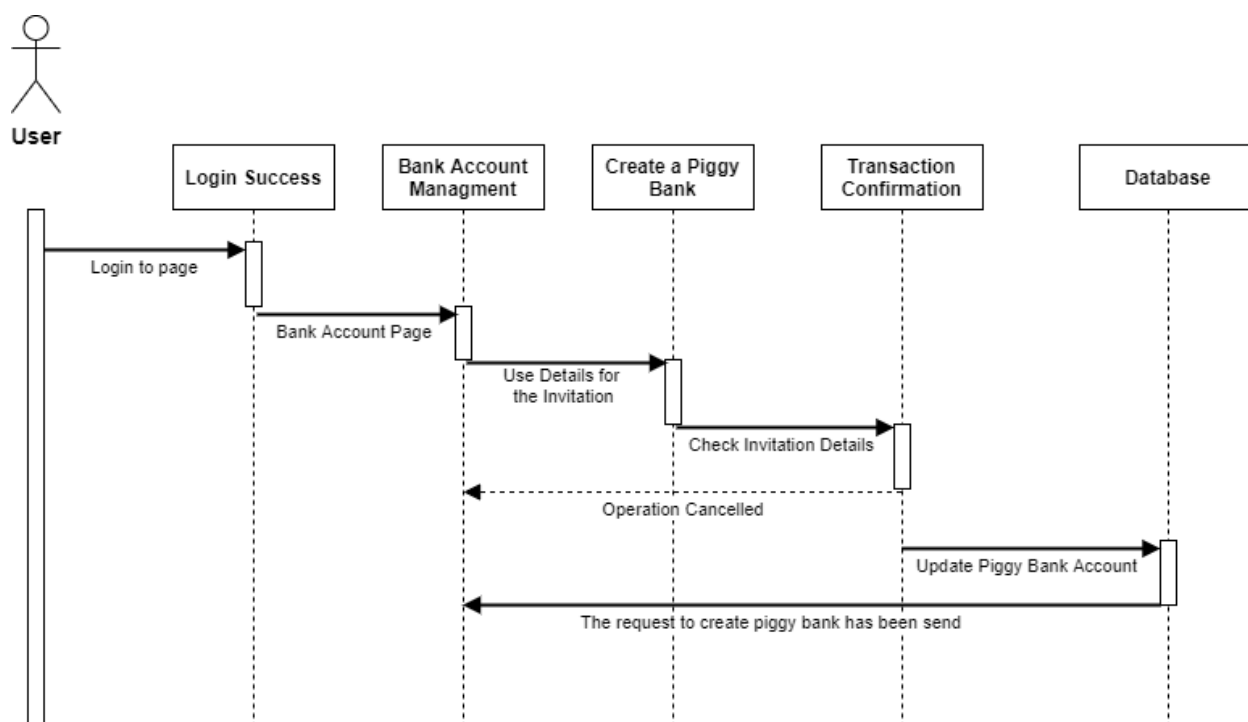


Рисунок 2.1 – Диаграмма последовательности совершения транзакции

Данная диаграмма описывает процесс транзакции пользователя в системе управления транзакциями. Включает основные этапы:

1 Пользователь: Это начальная точка, где пользователь начинает свою работу с системой.

2 Успешный вход: Пользователь входит в систему. Если вход успешен, пользователь переходит на следующий этап.

3 Страница управления транзакциями: После успешного входа пользователь попадает на страницу управления транзакциями.

4 Совершение транзакции: На этом этапе пользователь начинает процесс транзакции.

5 Подтверждение транзакции: После совершения транзакции пользователь переходит на страницу подтверждения транзакции, где он может проверить детали транзакции. Если детали верны, пользователь подтверждает транзакцию; если нет – отменяет её.

6 База данных: Подтвержденная транзакция обновляет банковский счет в базе данных. Если транзакция была отменена, этот шаг пропускается.

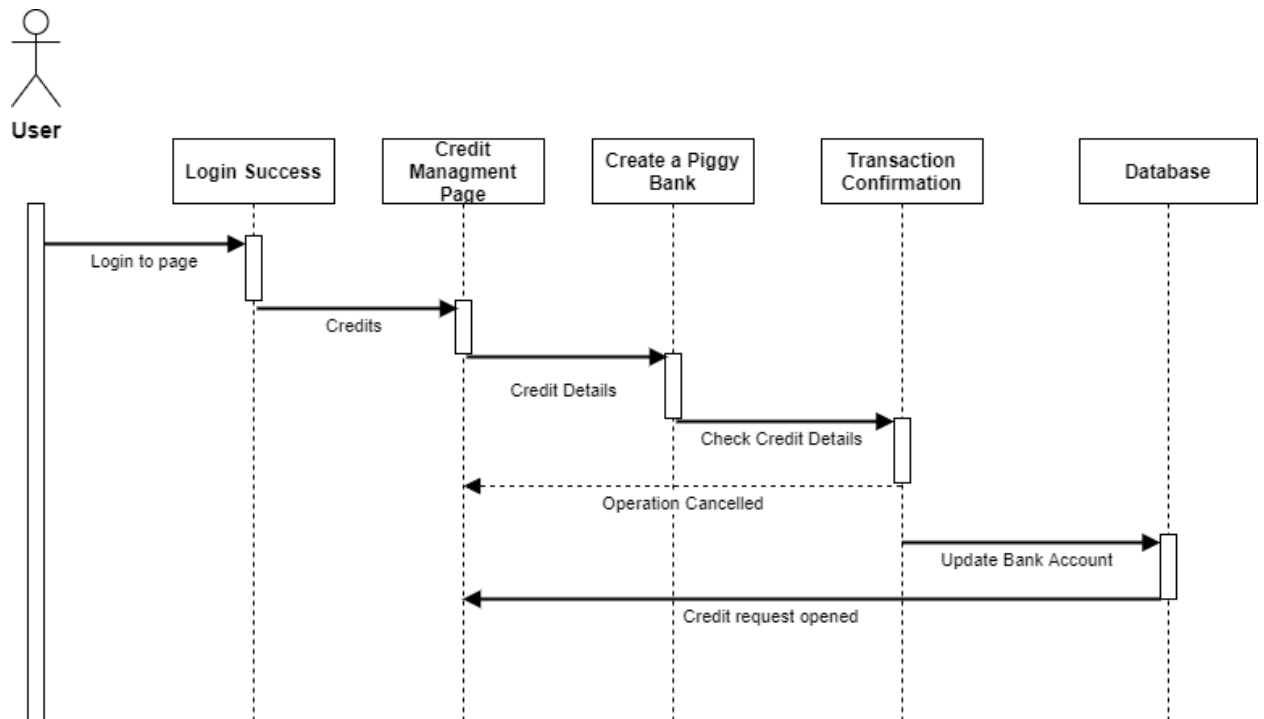


Рисунок 2.2 – Диаграмма последовательности создания коллективной копилки

Данная диаграмма описывает процесс создания коллективной копилки пользователем. Включает основные этапы:

1 Пользователь: Это начальная точка, где пользователь начинает свою работу с системой.

2 Успешный вход: Пользователь входит в систему. Если вход успешен, пользователь переходит на следующий этап.

3 Страница управления банковским счетом: После успешного входа пользователь попадает на страницу управления банковским счетом.

4 Создание совместной коллективной копилки: На этом этапе пользователь начинает процесс создания совместной коллективной копилки, вводя данные пользователя для приглашения.

5 Подтверждение: Здесь проверяются детали приглашения. Если детали верны, пользователь подтверждает приглашение; если нет – отменяет его.

6 База данных: Если приглашение подтверждено, запрос на создание коллективной копилки отправляется в базу данных, которая обновляется информацией о новой коллективной копилки.

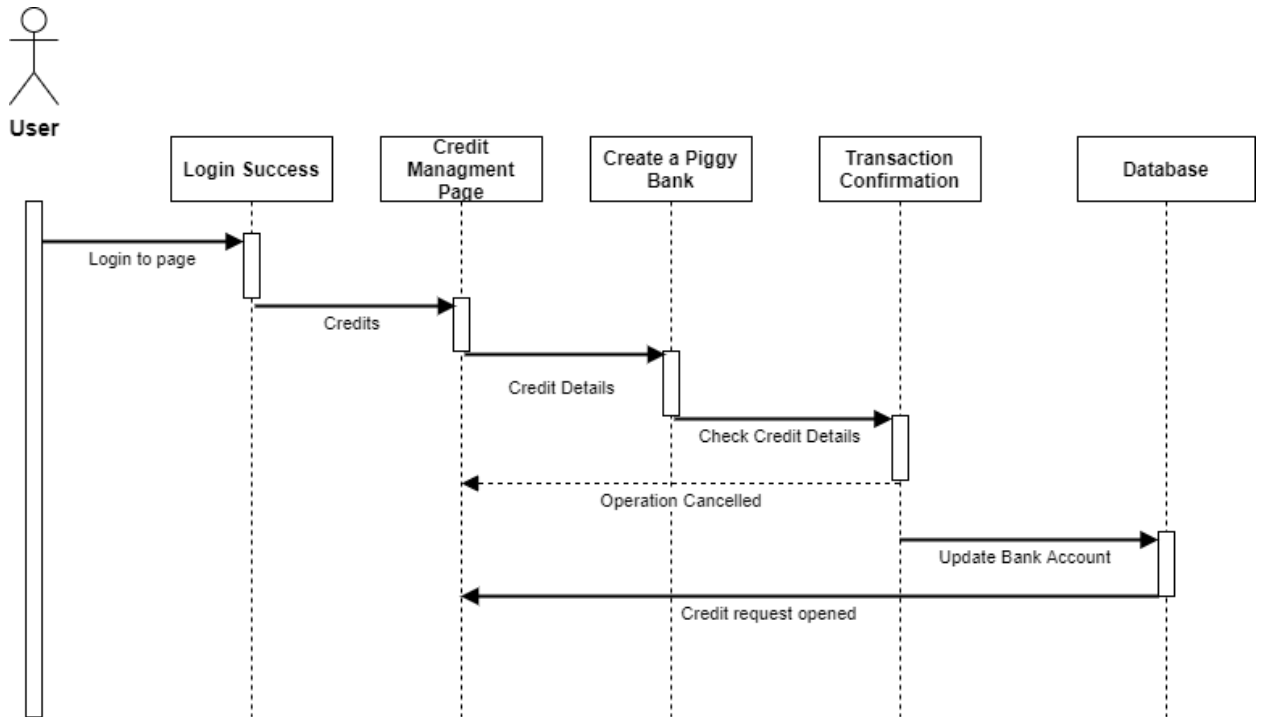


Рисунок 2.3 – Диаграмма последовательности оформления кредита

Данная диаграмма описывает процесс запроса кредита пользователем на веб-сайте. Включает основные этапы:

1 Пользователь: Это начальная точка, где пользователь начинает свою работу с системой.

2 Успешный вход: Пользователь входит в систему. Если вход успешен, пользователь переходит на следующий этап.

3 Страница управления кредитами: После успешного входа пользователь попадает на страницу управления кредитами.

4 Просмотр кредитов: На этой странице пользователь может просмотреть детали своих кредитов или отменить кредит.

5 Создание запроса на кредит: Пользователь может также начать процесс создания нового запроса на кредит.

6 Проверка деталей кредита: После создания запроса на кредит, детали кредита проверяются.

7 Подтверждение кредита: Если детали верны, пользователь подтверждает кредит; если нет – отменяет его.

8 База данных: Если кредит подтвержден, информация обновляется в базе данных банковского счета.

Для отображения системы в виде структурных компонентов и связей между ними была разработана диаграмма компонентов, представленная на рисунке 2.4.

Данная диаграмма представляет архитектуру банковского веб-приложения. Включает основные этапы:

1 Банковское веб-приложение: Центральный компонент, который взаимодействует с различными модулями.

2 Компоненты доступа к данным: Это модули, которые обрабатывают и управляют различными аспектами банковских услуг.

а) Кредит: Модуль, отвечающий за обработку и управление кредитами.

б) Транзакции: Модуль для обработки транзакций.

в) Общие банковские счета: Модуль для управления совместными банковскими счетами.

г) Карты: Модуль для управления банковскими картами.

д) Банковский счет: Модуль для управления индивидуальными банковскими счетами.

3 Контроль доступа и шифрование: Каждый модуль данных имеет механизмы контроля доступа и шифрования для обеспечения безопасности данных.

4 Компоненты безопасности: Отдельные модули, такие как “Безопасность” и “Постоянство”, обеспечивают дополнительные слои защиты данных и функциональности приложения.

5 База данных: Все данные хранятся в базе данных, которая подключается через компонент “Соединитель базы данных”.

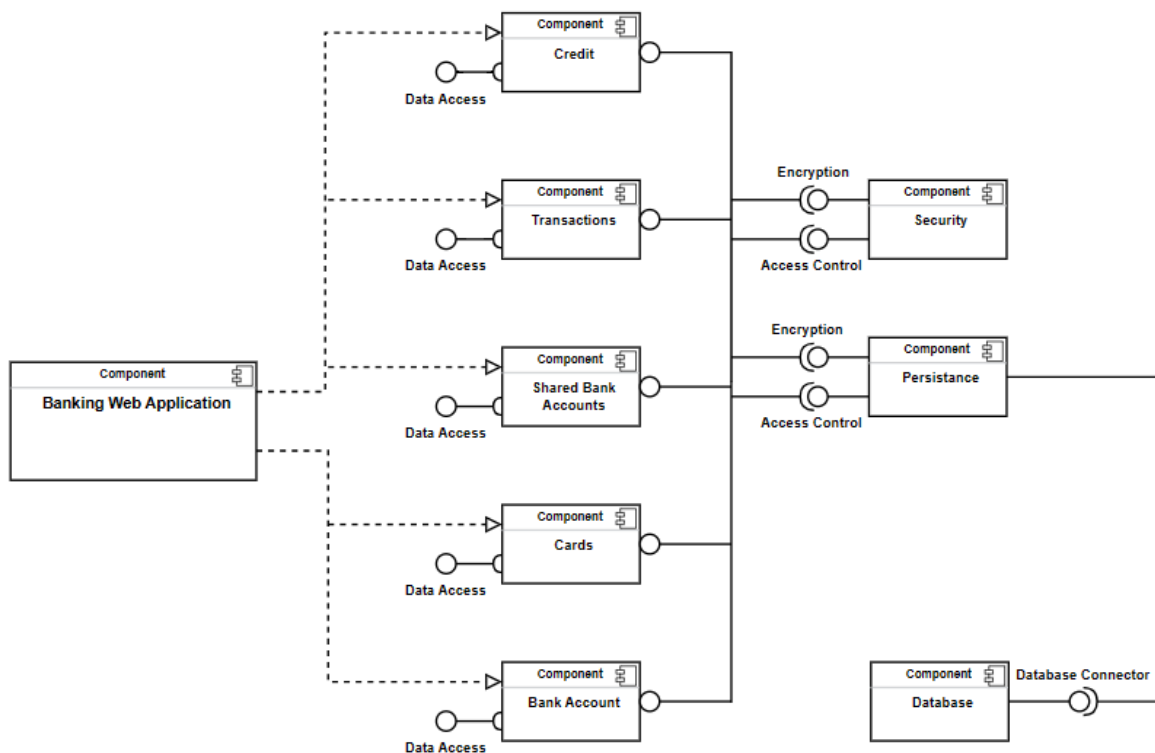


Рисунок 2.4 – Диаграмма компонентов программного средства

ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы, посвященной моделированию базы данных и проектированию программного продукта с использованием диаграмм последовательностей и компонентов, была успешно достигнута главная цель – создание надежной и эффективной системы. Эта система призвана обеспечивать клиентов удобством и безопасностью при управлении своими финансами.

Созданные диаграммы последовательностей значительно усовершенствовали понимание взаимодействия компонентов в системе и последовательности выполнения операций. Это, в свою очередь, обеспечивает глубокий анализ процессов и позволяет проводить оптимизацию. Диаграммы компонентов дают четкое представление о структуре системы, а также о том, как распределена функциональность между компонентами.