Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет   
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина «Технологии обработки финансовой информации»

**ОТЧЕТ**

к лабораторной работе

на тему:

**«Анализ предметной области. Выработка требований»**

|  |  |
| --- | --- |
|  | Выполнил:  Слуцкий Никита Сергеевич,  студент группы 053505 |
|  | Проверила: Тушинская Екатерина Вадимовна, ассистент каф. Информатики |

Минск 2023

СОДЕРЖАНИЕ

[Введение 3](#_Toc148021458)

[1 Анализ существующих областей рынка 4](#_Toc148021459)

[1.1 Белинвестбанк 4](#_Toc148021460)

[2 Выработка требований 5](#_Toc148021461)

[2.1 Регистрация и авторизация 5](#_Toc148021462)

[2.2 Профиль пользователя 5](#_Toc148021463)

[2.3 Информация по операциям 5](#_Toc148021464)

[2.4 Проведение платежей 6](#_Toc148021465)

[2.5 Установка лимитов операций 6](#_Toc148021466)

[2.6 Система кредитования 6](#_Toc148021467)

[3 Технологии, выбранные для разработки 7](#_Toc148021468)

[3.1 NextJS 7](#_Toc148021469)

[3.2 TailwindCSS 7](#_Toc148021470)

[3.3 База данных 7](#_Toc148021471)

[Список используемых источников 8](#_Toc148021472)

# Введение

Для разработки банковской системы необходимо учитывать множество требований, которые могут быть как общими для всех банковских систем, так и специфичными для конкретного проекта.

Онлайн банковские системы должны соответствовать некоторым базовым требованиям:

– безопасность;

– надежность;

– возможность совершать базовые банковские операции;

– удобство использования;

– хорошая доступность как с мобильных, так и с десктопных платформ;

– некоторые другие требования, которые, возможно, будут затронуты ниже в настоящем отчёте.

Они должны иметь возможность обеспечивать конфиденциальность данных клиентов, а также защищать их от взлома и кражи. Кроме того, они должны быть доступными для использования в любое время суток и на любом устройстве с доступом в Интернет.

Базовая простая банковская система должна иметь возможность обрабатывать основные операции, такие как переводы денежных средств, проверку баланса и т.д. Она должна быть легко понятной и удобной в использовании для клиентов.

Целью данной лабораторной работы является разработать комплексную базовую банковскую систему с виртуальными деньгами и пользователями. Это должно быть любого рода приложение, позволяющее совершать платежи, оформлять виртуальные кредиты, переводить деньги со счёта на счёт, просматривать курсы валют, а также имеющее внутри себя относительно уникальную фичу по сравнению с традиционным банковским функционалом.

Разработать полноценное решение, особенно для банка, – это долго. Соответственно, при разработке учебной системы в рамках данного курса многие подзадачи будут сильно упрощены.

# 1 Анализ существующих областей рынка

Перед началом разработки любого продукта необходимо хотя бы минимально проанализировать уже существующий аналог или аналоги, подумать, что можно было бы позаимствовать, что можно было бы улучшить, чтобы в конечном счёте разработать по крайней мере относительно конкурентноспособное приложение, а также чтобы понимать, в каком направлении двигаться на начальных этапах разработки, после выделения приоритетной последовательности функциональности.

## 1.1 Белинвестбанк

Интуитивно понятное веб-решение которое позволяет совершать платежи переводы открывать счета и вклады оформлять карты и кредиты. Также отличается приятным внешним видом, интуитивными расположением элементов, адаптивным дизайном. На рисунке 1 представлен скриншот из веб-версии онлайн-банкинга Белинвестбанка.

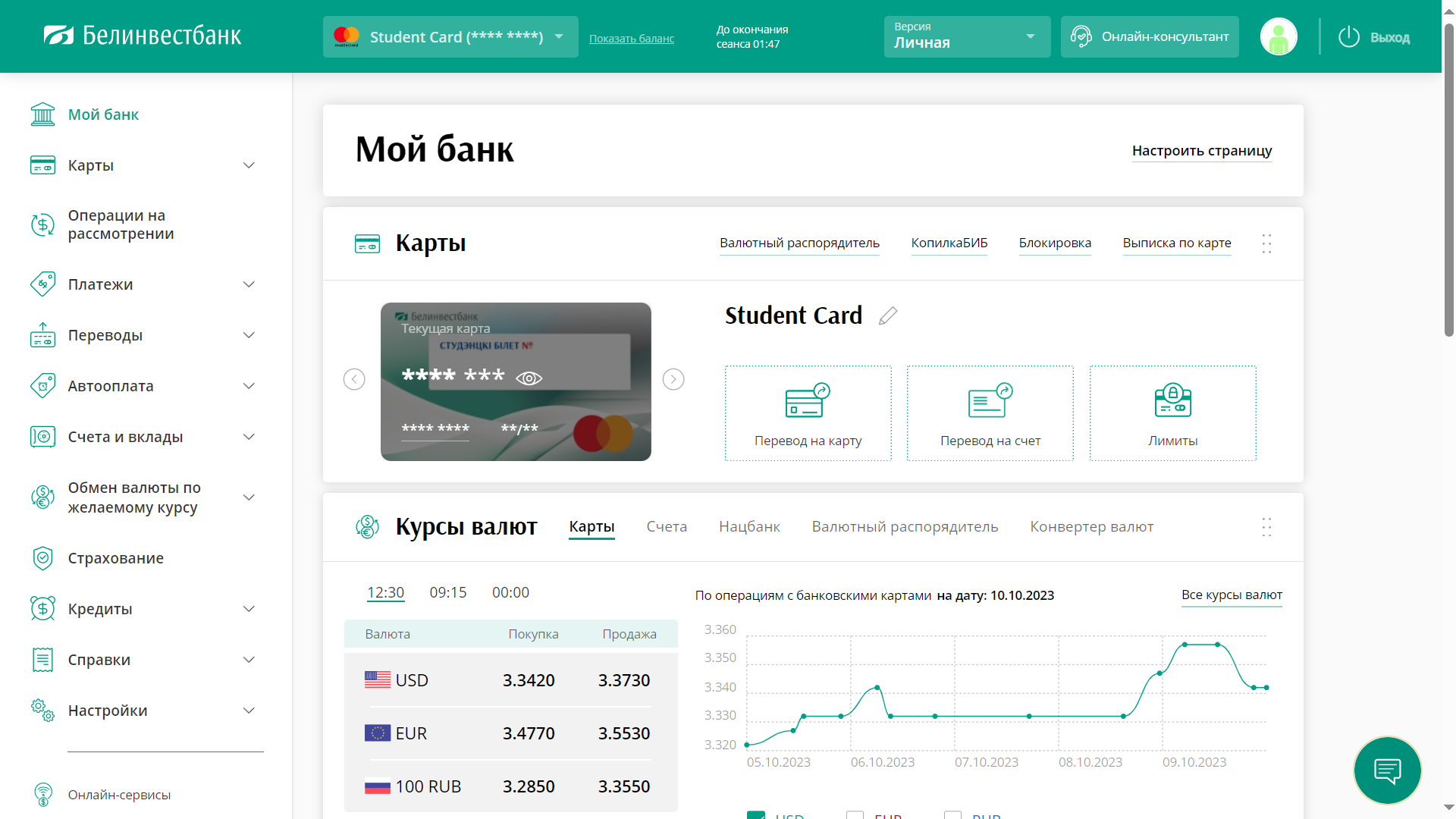


Рисунок 1 – Скриншот из интернет-банкинга

Соответственно, в рамках разработки программного продукта будет создана упрощённая версия подобной системы банкинга.

# 2 Выработка требований

Ставится задача разработки веб-решения с основными необходимыми функциями, которые пользователи уже привыкли видеть в банковских приложениях:

– регистрация и авторизация;

– профиль владельца;

– информация по операциям;

– проведение платежей;

– установка лимитов по операциям;

– оформление кредитов.

Ниже будет рассмотрена каждая из описанных категорий.

## 2.1 Регистрация и авторизация

Для регистрации пользователю не нужно идти в офис, а можно заполнить данные на сайте: заполнить данные об имени, фамилии, придумать пароль, ввести адрес электронной почты, прикрепить фотографию паспорта, селфи и зарегистрироваться. В ходе расширения системы в дальнейшем необходим будет функционал по верификации пользователей, для чего необходима база сотрудников, которые будут этим заниматься. После регистрации пользователь автоматически становится клиентом банка.

Для авторизации пользователю достаточно будет ввести адрес электронной почты и через код, посланный сервером на указанный адрес, залогиниться в приложении, разумеется, при условии, что этот адрес уже зарегистрирован.

## 2.2 Профиль пользователя

Профиль пользователя содержит информацию о клиенте, сюда же можно добавить почту и адрес для заказа физической карты. Тут же можно заказать выписки по карте. Можно наблюдать состояние текущего счёта и так далее.

## 2.3 Информация по операциям

В данном разделе в виде списка будет представлена последовательность транзакций, связанных с аккаунтом пользователя. При наличии времени будет реализована функциональность фильтрации и сортировки списка транзакций, а также возможность поиска по транзакциям.

## 2.4 Проведение платежей

В данном разделе должна быть возможность переводов средств между клиентами банка, проведением платежей за какие-либо услуги.

В рамках данного раздела одной из целей ставится реализовать уникальный функционал для разрабатываемой системы. Это будет возможность проведения переводов денег между пользователями с использованием QR-кода. Клиент-получатель формирует QR-код, клиент-отправитель для быстрой идентификации получателя может просканировать этот код. В традиционном же случае необходимо ввести полный номер аккаунта клиента-получателя.

## 2.5 Установка лимитов операций

Для операций перевода средств можно установить лимит. В данном разделе приложения можно поставить лимит на перевод в рамках одной транзакции или суммарный лимит на перевод на такой период времени, как сутки. Все изменения в лимитах сопровождаются подтверждением пароля. Также можно задать лимит на максимальную сумму перевода без запрашивания пароля пользователя.

## 2.6 Система кредитования

Банковская система предоставляет возможность системы кредитования. При оформлении кредита в разрабатываемой системе имеют важность такие параметры, как процентная ставка, срок кредита, сумма кредита. Соответственно, после оформления деньги переходят на счёт аккаунта, в профиле клиента отображается вкладка кредиты, где можно увидеть кредитную историю и текущий долг.

## 2.7 Диаграмма вариантов использования

Диаграмма вариантов использования – диаграмма, описывающая, какой функционал разрабатываемой программной системы доступен каждой группе пользователей.

# 3 Технологии, выбранные для разработки

Разрабатываемая система – это современное веб-приложение, разработанное в окружении NodeJS. Исходя из этого и строится процесс выбора конкретных технологий.

## 3.1 NextJS

Next.js – фреймворк на основе React, для улучшенной производительности и пользовательского опыта. Предварительный рендеринг, включая SSR и SSG, создает улучшенные веб-приложения.

Как учебная платфома, CEO-оптимизации разрабатываемое решение не требует. Однако последние годы команда React рекомендует не создавать чистые React-приложения, а создавать их, базируясь на каких-либо фреймворках. NextJS помимо оптимизации CEO диктует более чёткую структуру при разработке, предлагает много удобных инструментов и позволяет писать бэк-енд прямо в своей экосистеме. Ввиду того, что на данный момент это самое мощное и популярное решение, оно и будет стоять в центре используемых технологий для разработки программного средства.

## 3.2 TailwindCSS

TailwindCSS – это CSS-фреймворк, позволяющий вносить изменения в оформление сайтов и приложений, не покидая HTML-разметку. Внутри Tailwind присутствуют тысячи сгенерированных классов, которые позволяют максимально гибко и быстро писать стили для веб-приложения.

В разрабатываемом приложении интерфейс играет важное, но не главное значение, поэтому во избежание больших временных затрат на разработку визуальной составляющей будет использоваться этот фреймворк, который упростит разработку страниц.

## 3.3 База данных

В рамках выполнения лабораторной работы №2 и №3 будет более детально проведён анализ и выбор итоговой СУБД для разработки. Теоретическими вариантами для использования являются Supabase или MongoDB.

Supabase – это набор low-code инструментов для создания приложений. В список фич входят база данных на PostgreSQL, механизм аутентификации пользователей, API, хранение данных и управление подписками. Все – с открытым кодом, то есть с возможностью развернуть проект на собственном сервере.

MongoDB – система управления базами данных, которая работает с документоориентированной моделью данных. В отличие от реляционных СУБД, MongoDB не требуются таблицы, схемы или отдельный язык запросов. Информация хранится в виде документов либо коллекций.

Разработчики позиционируют продукт как промежуточное звено между классическими СУБД и NoSQL. MongoDB не использует схемы, как это делают реляционные базы данных, что повышает производительность всей системы.

# Список используемых источников

[1] 3D-планировщик [Электронный ресурс]. – Режим доступа: https://cedreo.com/faq/what-is-a-3d-floor-plan-rendering. – Дата доступа: 20.09.2023

[2] Платформа Roomle [Электронный ресурс]. – Режим доступа: https://www.roomle.com/en. – Дата доступа: 20.09.2023

[3] Платформа RoomToDo [Электронный ресурс]. – Режим доступа: https://roomtodo.com/en/. – Дата доступа: 20.09.2023

[4] Семантически корректный сайт [Электронный ресурс]. – Режим доступа: https://web.dev/learn/html/semantic-html. – Дата доступа: 21.09.2023

[5] Доступный интерфейс [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/en-US/docs/Learn/Accessibility/HTML/. – Дата доступа: 21.09.2023

[6] Фреймворк NextJS [Электронный ресурс]. – Режим доступа: https://nextjs.org/. – Дата доступа: 22.09.2023

[7] Язык программирования TypeScript [Электронный ресурс]. – Режим доступа: https://www.typescriptlang.org/. – Дата доступа: 22.09.2023

[8] Платформа NodeJS [Электронный ресурс]. – Режим доступа: https://nodejs.org/en. – Дата доступа: 22.09.2023

[9] Библиотека для разработки интерфейсов ReactJS [Электронный ресурс]. – Режим доступа: https://react.dev/. – Дата доступа: 22.09.2023

[10] Chrome V8 [Электронный ресурс]. – Режим доступа: https://v8.dev/. – Дата доступа: 23.09.2023

[11] СУБД MongoDB [Электронный ресурс]. – Режим доступа: https://www.mongodb.com/. – Дата доступа: 23.09.2023

[12] СУБД PostgreSQL [Электронный ресурс]. – Режим доступа: https://www.postgresql.org/. – Дата доступа: 23.09.2023

[13] IDE JetBrains WebStorm [Электронный ресурс]. – Режим доступа: https://www.jetbrains.com/webstorm/. – Дата доступа: 24.09.2023

[14] Редактор кода Microsoft Visual Studio Code [Электронный ресурс]. – Режим доступа: https://code.visualstudio.com/. – Дата доступа: 24.09.2023

[15] WebGL API [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/en-US/docs/Web/API/WebGL\_API. – Дата доступа: 24.09.2023

[16] Система контроля версий GIT [Электронный ресурс]. – Режим доступа: https://git-scm.com/. – Дата доступа: 24.09.2023