**Государственное бюджетное профессиональное образовательное учреждение**

**Астраханской области «Астраханский колледж вычислительной техники»**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **К защите допустить:** | | | | | | | |
| **Старший методист** | | | | | | | |
|  |  | | | **Л.О. Земцова** | | | |
| **«** |  | **»** |  | | **20** |  | **г.** |

**курсовая работа**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| по МДК | 02.02 | Технология разработки и защиты баз данных | | | | | |
|  | | | | | | | |
|  | | 6 семестр | | | | | |
| Тема курсовой работы | | Разработка базы данных | | | | | |
|  | | «Кредитный отдел банка» | | | | | |
|  | |  | | | | | |
| Аквт.09.02.07. КР00.14ПЗ | | | | | | | |
| Листов: | | | 21 |  | | | |
|  | |  | | | | | |
|  | |  | | | | | |
| Разработчик | |  | | | Студент гр. | | ПБ – 32 |
|  | |  | | | Д.Е.Максимов | | |
|  | |  | | |  |  | |
| Руководитель | |  | | | Преподаватель | | |
|  | |  | | | И.А. Храмцовский | | |

**2023**

|  |
| --- |
| УТВЕРЖДЕНО |
| Цикловой комиссией |
| специальности 09.02.07 |
| ПЦК \_\_\_\_\_\_\_\_\_ Ю.С. Андрианова |
| «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. |
| Специальность | *09.02.07 Информационные системы и программирование* | | |
| МДК 04.01 | *Технология разработки и защиты баз данных* | | |
| Группа | *ПБ– 32* | | |
| **ЗАДАНИЕ** | | | |
| **на курсовое проектирование** | | | |
| ***Максимов Данила Евгеньевич*** | | | |
| (фамилия, имя, отчество студента) | | | |
| Тема курсового проекта | | ***Разработка БД*** | |
| ***«Кредитный отдел банка»*** | | | |

Содержание пояснительной записки курсового проекта

Содержание

Введение

1. Проектирование базы данных
2. Анализ и описание предметной области информационной системы
3. Проектирование концептуальной модели
4. Построение логической модели, схемы базы данных
5. Выбор СУБД
6. Реализация базы данных
7. Физическое проектирование
8. Написание исходного кода БД
9. Написание запросов, функций и хранимых процедур

Заключение

Список использованных источников

Приложения

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Дата выдачи задания | « |  | » |  | |  | 20 |  | г. |
| Руководитель |  | | | | И.А. Храмцовский | | | | |
| Студент |  | | | | Максимов Д. Е. | | | | |

**Аннотация**

Данная курсовая работа посвящена анализу баз данных Кредитного отдела банка. В работе рассмотрен анализ заявок на кредит и принятием решения по кредиту. Также рассмотрены основные задачи, структура и функционирование баз данных Кредитного отдела банка. В работе проведен анализ возможных проблем в работе баз данных Кредитного отдела банка, а также предложены рекомендации для оптимизации и улучшения их работы. Результаты работы позволяют более эффективно организовать работу с клиентами и повысить уровень обслуживания приложения банка.

**Содержание**

[Введение 5](#_Toc135224632)

[1 Проектирование базы данных 6](#_Toc135224633)

[1.1 Анализ и описание предметной области информационной системы 7](#_Toc135224634)

[1.2 Проектирование концептуальной модели 8](#_Toc135224635)

[1.3 Построение логической модели, схемы базы данных 9](#_Toc135224636)

[1.4 Выбор СУБД 10](#_Toc135224637)

[2 Реализация базы данных 11](#_Toc135224638)

[2.1 Физическое проектирование 12](#_Toc135224639)

[2.2 Написание исходного кода БД 13](#_Toc135224640)

[2.3 Написание запросов, функций и хранимых процедур 14](#_Toc135224641)

[Заключение 15](#_Toc135224642)

[Список использованных источников 16](#_Toc135224643)

[Приложение 17](#_Toc135224644)

**Введение**

В наш век важной характеристикой для любого сервиса является его скорость и доступность для человека. Переносить информацию на внешних носителях невероятно трудоемкий и не эффективный способ, а если взять в расчет что это нужно многим людям, то это может занимать часы что отразиться на эффективности работы. Для таких случаев в помощь были разработаны базы данных (БД) и также системы их управления (СУБД).

Использование баз данных повышает производительность труда, упрощают аналитику и использования различных средств представления информации. Создание БД является очень важным событием ведь нам постоянно нужно иметь дело с большими объемом информации.

Реализация поставленной задачи будет проводится в PostgreSQL это бесплатная объективно-реляционная СУБД, располагающая широкими возможностями для создания баз данных и удобным интерфейсом с простотой настроек, быстрым созданием таблиц и запросов.

База данных «Кредитный отдел банка» предназначенная для простоты работы с клиентской базой и упрощение аналитических отчетов.

Объектом исследования является база данных «Кредитного отдела банка» которая должна автоматизировать систему ввода и вывода данных.

Предметом исследования является процесс разработки объектно-реляционной базы данных

Теоретическая значимость работы состоит в обосновании реляционного подхода к проектированию и разработке базы данных санаторно-курортного комплекса.

Практическая значимость работы состоит в разработке базы данных информационной системы санаторного комплекса.

# **Проектирование базы данных**

## **Анализ и описание предметной области информационной системы**

База данных создается для кредитного отдела банка с целью отслеживания динамики его работы. В данной работе будет реализована база данных для автоматизации сортировки заявок и одобренных кредитов. Задачей менеджеров и работников принимать на рассмотрение заявки. Клиенты могу узнавать о статусе своей заявки и знать на каком этапе она находится. У кредитного отдела есть информация о ОКВЕД, которая берется с официального сайта “Центр Регистрации и Сопровождения Бизнеса”.

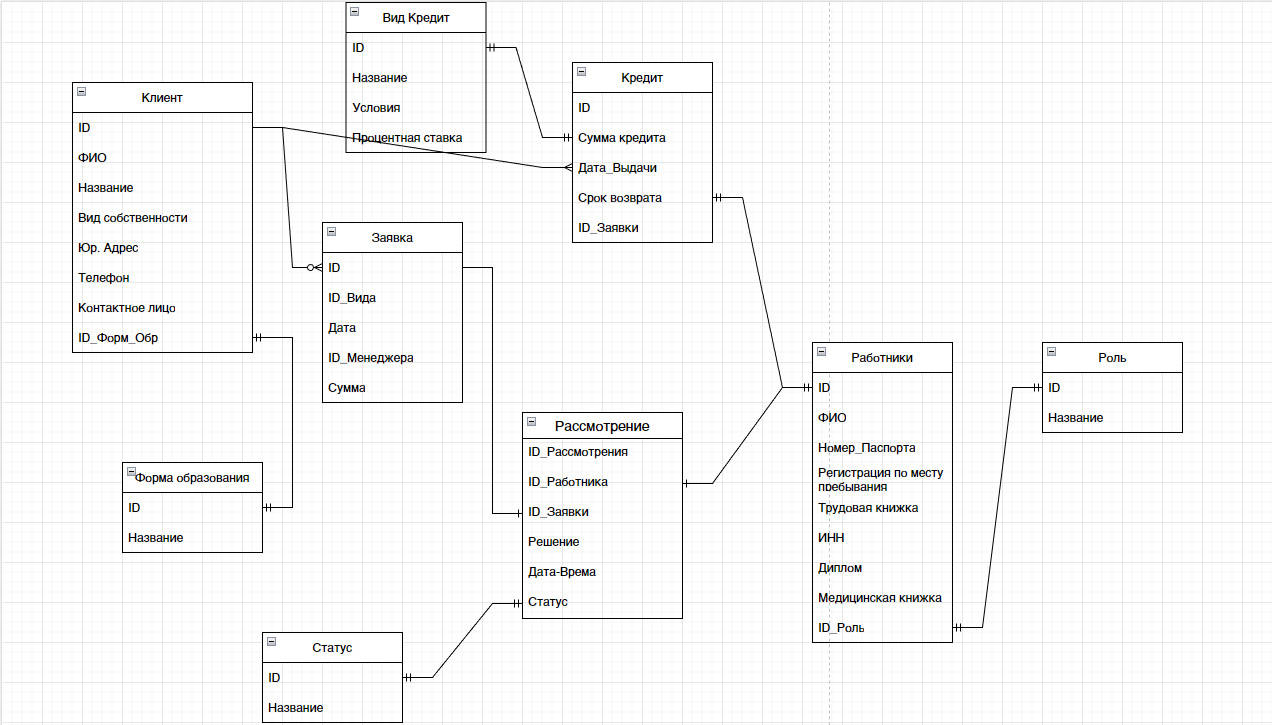
Выходные документы данного процесса отдаются клиенту в электронном виде.

Менеджеры могут смотреть информацию об открытых кредитов и какой из сотрудников открыл данный кредит.

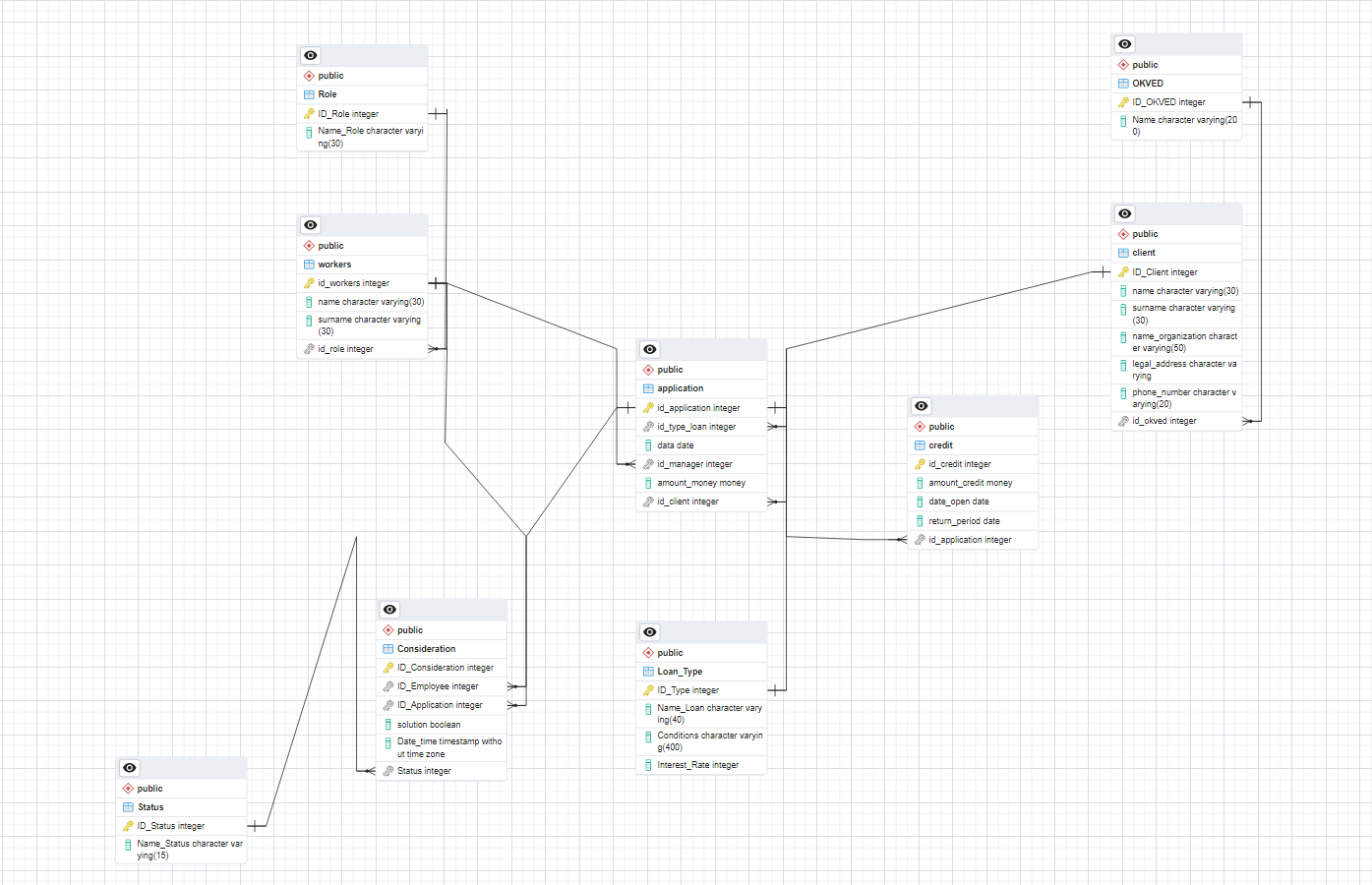
Таблица 1 – Бизнес-правила

|  |  |  |
| --- | --- | --- |
| № | Определение правила | Тип правила |
| 1 | Слежка динамики кредитного отдела | Факт |
| 2 | Клиент может отправить электронную заявку | Факт |
| 3 | Заявку имеет статус «Рассмотрение» в случае, если заявка отправился на рассмотрение к работникам | Вывод |
| 4 | Заявку имеет статус «Одобрено» в случае, если заявку приняли | Вывод |
| 5 | Заявку имеет статус «Отказано» в случае, если заявку отклонили | Вывод |

## **Проектирование концептуальной модели**



## **Построение логической модели, схемы базы данных**



## **Выбор СУБД**

PostgreSQL — это мощная и гибкая система управления реляционными базами данных, которая обладает множеством преимуществ. Вот некоторые из них:

1. Бесплатность: PostgreSQL распространяется как свободное программное обеспечение и доступен всем пользователям без каких-либо дополнительных затрат.
2. Высокая производительность: PostgreSQL обеспечивает высокую производительность в обработке больших объемов данных. Он обладает мощными функциями оптимизации запросов и способен обрабатывать многопоточные операции, что увеличивает скорость работы.
3. Безопасность: PostgreSQL обеспечивает высокий уровень безопасности своих баз данных. Он поддерживает шифрование данных, аутентификацию пользователей и механизмы доступа на основе ролей и разрешений.
4. Масштабируемость: PostgreSQL имеет возможность горизонтального масштабирования, что позволяет увеличивать мощность и производительность базы данных при необходимости.
5. Поддержка стандартов: PostgreSQL активно поддерживает множество стандартов языка SQL, что делает возможным перенос баз данных между различными системами управления базами данных.
6. Модульность: PostgreSQL имеет модульную архитектуру, что позволяет добавлять и изменять функциональность базы данных в соответствии с требованиями приложений.

В целом, PostgreSQL обеспечивает высокую надежность, скорость и безопасность работы с данными, что делает его одним из наиболее популярных и эффективных СУБД на рынке.

# **Реализация базы данных**

## **Физическое проектирование**

-----Вид кредита-----

ID - id вида кредита

Название - название вида кредита

Условия - Условия при которых этот кредит можно выдать

Процентная ставка - процент который платит получатель

-----Кредит-------

ID - id открытого кредита

Сумма кредита - Сумма взятого кредита

ID клиента - ID клиента взявшего кредита

Дата выдачи - дата когда кредит был выдан

ID Работника - ID рабоника одобряющего кредит

Срок возврата- Дата возврата кредита

ID\_Заявки - ID Заявки

----Работники-----

ID - ID Работников

ФИО - Фио работника

Роль - Роль сотрудника в системе

----Расмотрение----

ID Рассмотрения - ID Рассмотрения кредита

ID Работника - ID Работника который рассматривает

ID Заявки - ID Заявки на кредит

Решение - Решенеие кредиту да/нет

Дата-Время - Принятого на рассмотрение

Статус - На каком этапе находиться рассмотрение

---Заявка----

ID - ID Заявка на кредит

ID\_Вида - ID вида кредита

Дата - дата когда отправленна заявка

ID\_Менеджера - ID\_Менеджера занимающегося зявкой

Сумма - Сумма указаная в заявке

ID\_Клиента - ID клиента отправившего заявку

-----Клиент----

ID- Id клиента

Фамилия - Фамилия пользователя

Имя - Имя пользователя

Отчество - Отчество пользователя

Названия - Названия компании

Вид собственности - вид вобственности клиента

Юр. Адрес - Адрес по которому находиться собственность

Телефон - Номер телофона клиента

Контактное лицо - Лицо ответственное

ID\_Форм\_Обр - ID Формы образовавния

---Форма Образования---

ID - ID формы образования

Названия - Название формы образования

---Статус---

ID - ID статуса

Названия - Название статуса

---Роль

ID - ID роли

Названия - Название Роли

## **Написание исходного кода БД**

--

-- PostgreSQL database dump

--

-- Dumped from database version 15.2 (Debian 15.2-1.pgdg110+1)

-- Dumped by pg\_dump version 15.2

-- Started on 2023-06-09 13:52:26

SET statement\_timeout = 0;

SET lock\_timeout = 0;

SET idle\_in\_transaction\_session\_timeout = 0;

SET client\_encoding = 'UTF8';

SET standard\_conforming\_strings = on;

SELECT pg\_catalog.set\_config('search\_path', '', false);

SET check\_function\_bodies = false;

SET xmloption = content;

SET client\_min\_messages = warning;

SET row\_security = off;

SET default\_tablespace = '';

SET default\_table\_access\_method = heap;

--

-- TOC entry 215 (class 1259 OID 17283)

-- Name: client; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public.client (

"ID\_Client" integer NOT NULL,

name character varying(30) NOT NULL,

surname character varying(30) NOT NULL,

name\_organization character varying(50) NOT NULL,

legal\_address character varying NOT NULL,

phone\_number character varying(20) NOT NULL,

id\_okved integer NOT NULL

);

ALTER TABLE public.client OWNER TO akvt;

--

-- TOC entry 214 (class 1259 OID 17282)

-- Name: Client\_ID\_Client\_seq; Type: SEQUENCE; Schema: public; Owner: akvt

--

ALTER TABLE public.client ALTER COLUMN "ID\_Client" ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."Client\_ID\_Client\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

--

-- TOC entry 219 (class 1259 OID 17301)

-- Name: Consideration; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public."Consideration" (

"ID\_Consideration" integer NOT NULL,

"ID\_Employee" integer NOT NULL,

"ID\_Application" integer NOT NULL,

solution boolean,

"Date\_time" timestamp without time zone NOT NULL,

"Status" integer NOT NULL

);

ALTER TABLE public."Consideration" OWNER TO akvt;

--

-- TOC entry 228 (class 1259 OID 17966)

-- Name: Consideration\_ID\_Consideration\_seq; Type: SEQUENCE; Schema: public; Owner: akvt

--

ALTER TABLE public."Consideration" ALTER COLUMN "ID\_Consideration" ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."Consideration\_ID\_Consideration\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

--

-- TOC entry 225 (class 1259 OID 17323)

-- Name: credit; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public.credit (

id\_credit integer NOT NULL,

amount\_credit money NOT NULL,

date\_open date NOT NULL,

return\_period date NOT NULL,

id\_application integer NOT NULL

);

ALTER TABLE public.credit OWNER TO akvt;

--

-- TOC entry 224 (class 1259 OID 17322)

-- Name: Credit\_ID\_Credit\_seq; Type: SEQUENCE; Schema: public; Owner: akvt

--

ALTER TABLE public.credit ALTER COLUMN id\_credit ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."Credit\_ID\_Credit\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

--

-- TOC entry 226 (class 1259 OID 17329)

-- Name: Loan\_Type; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public."Loan\_Type" (

"ID\_Type" integer NOT NULL,

"Name\_Loan" character varying(40) NOT NULL,

"Conditions" character varying(400) NOT NULL,

"Interest\_Rate" integer NOT NULL

);

ALTER TABLE public."Loan\_Type" OWNER TO akvt;

--

-- TOC entry 217 (class 1259 OID 17291)

-- Name: OKVED; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public."OKVED" (

"ID\_OKVED" integer NOT NULL,

"Name" character varying(200) NOT NULL

);

ALTER TABLE public."OKVED" OWNER TO akvt;

--

-- TOC entry 216 (class 1259 OID 17290)

-- Name: OKVED\_ID\_OKVED\_seq; Type: SEQUENCE; Schema: public; Owner: akvt

--

ALTER TABLE public."OKVED" ALTER COLUMN "ID\_OKVED" ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."OKVED\_ID\_OKVED\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

--

-- TOC entry 223 (class 1259 OID 17317)

-- Name: Role; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public."Role" (

"ID\_Role" integer NOT NULL,

"Name\_Role" character varying(30) NOT NULL

);

ALTER TABLE public."Role" OWNER TO akvt;

--

-- TOC entry 220 (class 1259 OID 17306)

-- Name: Status; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public."Status" (

"ID\_Status" integer NOT NULL,

"Name\_Status" character varying(15) NOT NULL

);

ALTER TABLE public."Status" OWNER TO akvt;

--

-- TOC entry 222 (class 1259 OID 17312)

-- Name: workers; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public.workers (

id\_workers integer NOT NULL,

name character varying(30) NOT NULL,

surname character varying(30) NOT NULL,

id\_role integer NOT NULL

);

ALTER TABLE public.workers OWNER TO akvt;

--

-- TOC entry 221 (class 1259 OID 17311)

-- Name: Workers\_ID\_Workers\_seq; Type: SEQUENCE; Schema: public; Owner: akvt

--

ALTER TABLE public.workers ALTER COLUMN id\_workers ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."Workers\_ID\_Workers\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

--

-- TOC entry 218 (class 1259 OID 17296)

-- Name: application; Type: TABLE; Schema: public; Owner: akvt

--

CREATE TABLE public.application (

id\_application integer NOT NULL,

id\_type\_loan integer NOT NULL,

data date NOT NULL,

id\_manager integer NOT NULL,

amount\_money money NOT NULL,

id\_client integer NOT NULL

);

ALTER TABLE public.application OWNER TO akvt;

--

-- TOC entry 227 (class 1259 OID 17617)

-- Name: application\_id\_application\_seq; Type: SEQUENCE; Schema: public; Owner: akvt

--

ALTER TABLE public.application ALTER COLUMN id\_application ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public.application\_id\_application\_seq

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

## **Написание запросов, функций и хранимых процедур**

1.Select \* from public."Consideration"

2.Select client.name, application.id\_application,application.amount\_money

from public."client"

inner join application

on client."ID\_Client" = application."id\_client"

3.Select workers."name", role."Name\_Role"

from workers

inner join role

on workers."id\_role" = role."ID\_Role"

**Заключение**

В работе были проанализированы существующие базы данных в кредитном отделе банка и проведены исследования по оптимизации их использования. Были выявлены недостатки в существующих системах, такие как длительное время на обработку информации и недостаточная конфиденциальность данных клиентов.

Решением этих проблем могут стать интеграция облачных технологий, использование системы шифрования данных и автоматизация процессов, что позволит ускорить работу кредитного отдела и повысить уровень безопасности информации.

Также были предложены рекомендации по улучшению процесса принятия решений в кредитном отделе банка, например, создание единой базы данных с информацией о клиентах, и упрощение процесса выдачи кредитов.

В целом, данная курсовая работа может послужить полезным руководством для улучшения работы кредитного отдела банка и повышения уровня сервиса для клиентов.

**Список использованных источников**

[**https://www.postgresql.org/docs/current/tutorial.html**](https://www.postgresql.org/docs/current/tutorial.html)

**https://alfabank.ru**

**https://www.regfile.ru/okved2.html**

**Приложение**

БД — база данных

СУБД - Система управления базами данных

PostgreSQL — свободная объектно-реляционная система управления базами данных.