Министерство науки и высшего образования РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
институт
Программная инженерия
кафедра

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ

Проектирование базы данных и ее реализация в среде СУБД PostgreSQL
тема

Преподаватель _____ А. Д. Вожжов
подпись, дата — инициалы, фамилия

Студент КИ23-17/1б, 032320521 _____ А. С. Лысаковский
номер группы, зачётной книжки — подпись, дата — инициалы, фамилия

Красноярск 2025

1 ВВЕДЕНИЕ

1.1 Цель работы

Изучить теоретический материал по теме «Проектирование базы данных и ее реализация в среде СУБД PostgreSQL». Выполнить задания.

1.2 Задачи

В рамках данной практической работы необходимо выполнить следующие задачи:

1 изучить теоретический материал по предложенной теме;
2 выполнить задание;
3 предоставить отчёт преподавателю.

1.3 Задание

Задание данной практической работы состоит из следующих частей:

1 Изучить материал лекций 6-8;
2 Выбрать предметную область, которая вам интересна и в которой вы разбираетесь;
3 Спроектировать базу данных для выбранной предметной области с учётом требований;
4 Ввести небольшое количество записей в таблицы базы данных, чтобы можно было продемонстрировать типичные запросы к базе данных;
5 Подготовить несколько типичных запросов к базе данных и сохранить их в отдельных текстовых файлах. Для демонстрации этих запросов их можно вызывать как извне утилиты «psql», так и изнутри нее.

Для каждого отношения (таблицы) необходимо указать номер нормальной формы, в которой это отношение находится, и кратко обосновать, из чего это следует. Если какое-либо отношение не находится хотя бы в 3НФ, необходимо обосновать, почему принято такое проектное решение.

2 ХОД РАБОТЫ

2.1 Предметная область

В качестве предметной области была выбрана система выдачи и хранения книг.

2.2 Концептуальная модель

Концептуальная модель представлена на рисунке 1.

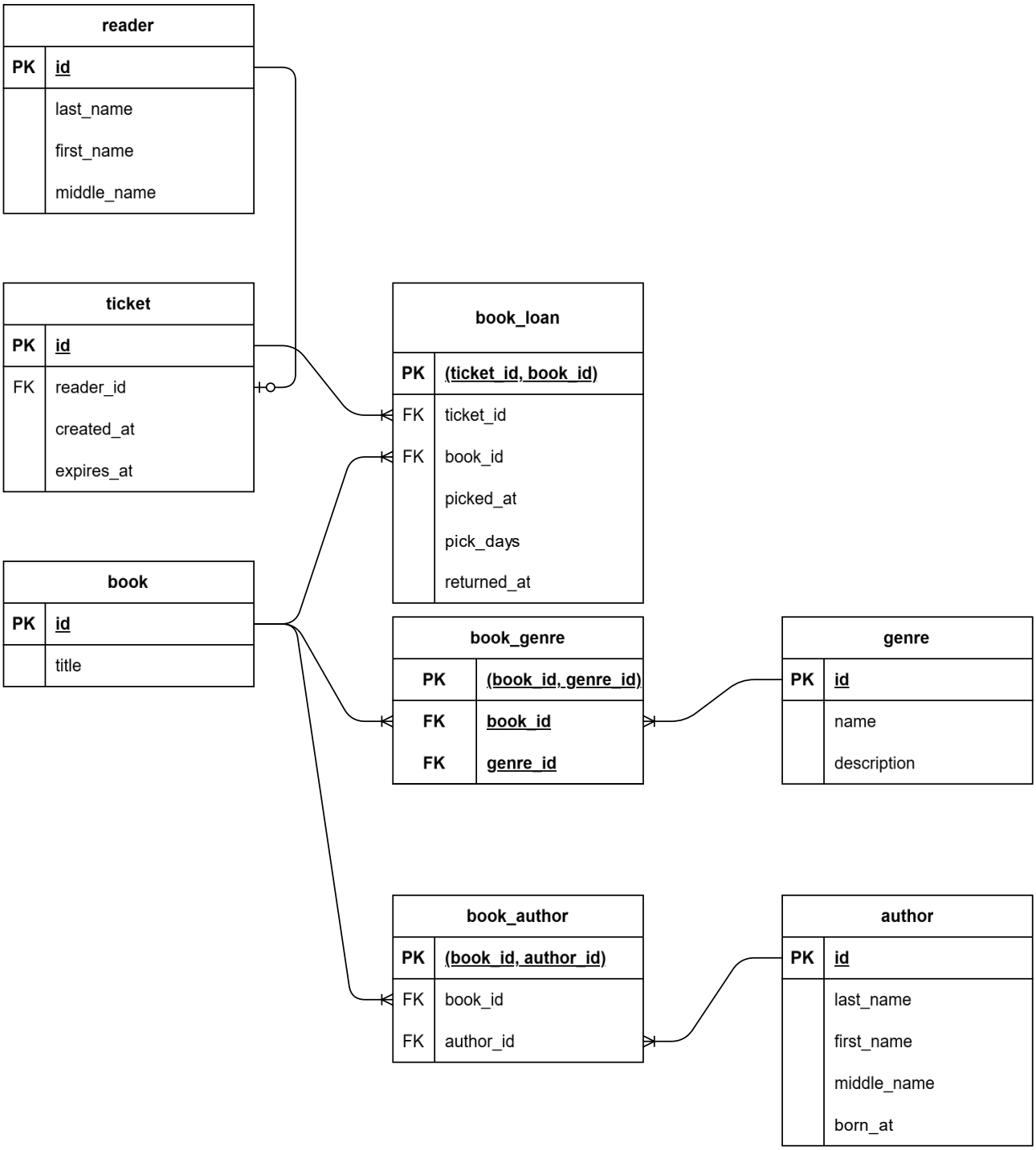

Рисунок 1 – Концептуальная модель

2.3 Логическая модель

Логическая модель представлена на рисунке 2.

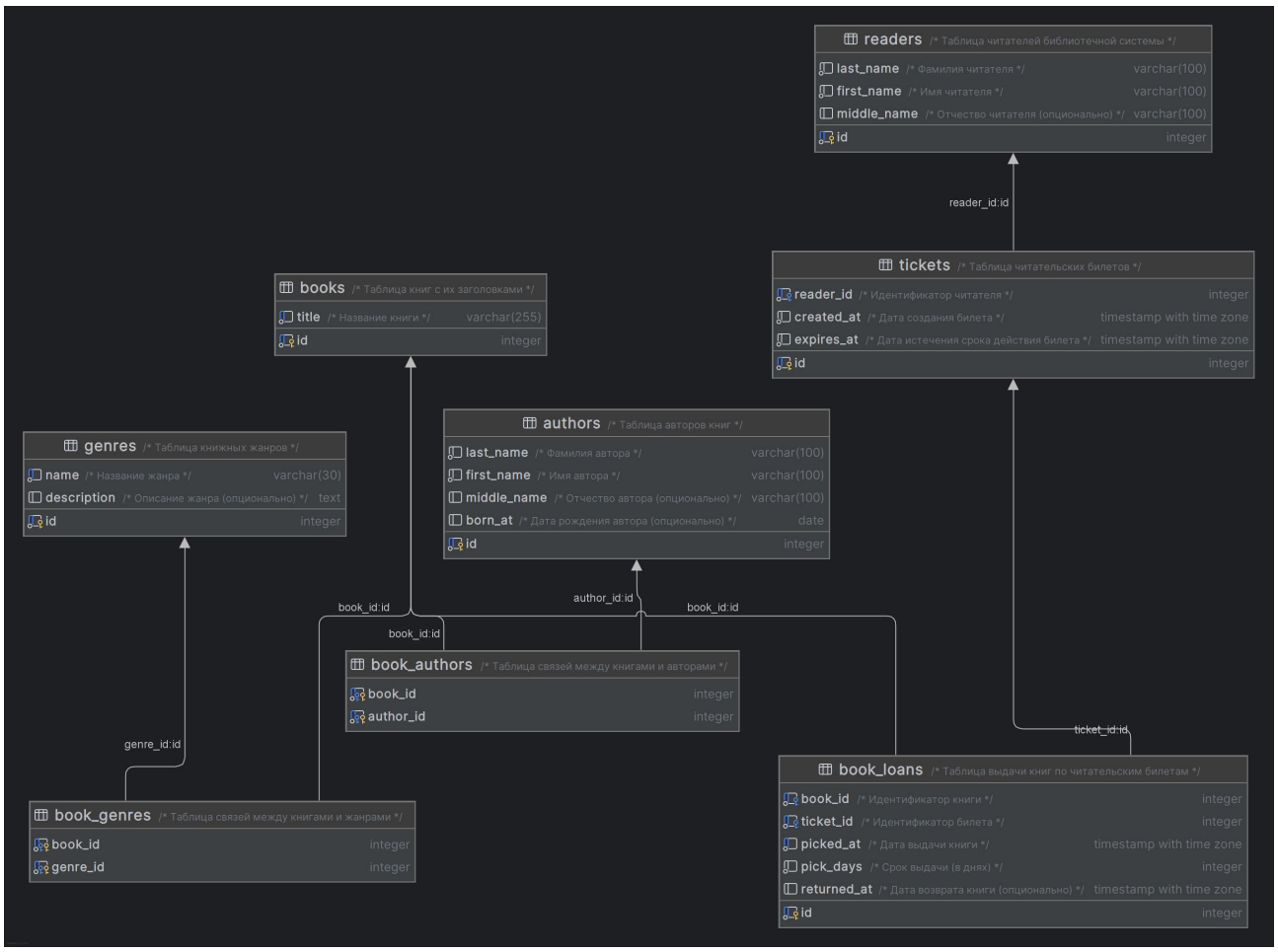

Рисунок 2 – Логическая модель

2.4 Физическая модель

Физическая модель представлена ниже.

```
CREATE SCHEMA library;

CREATE TABLE library.readers (
    id SERIAL PRIMARY KEY,
    last_name VARCHAR(100) NOT NULL,
    first_name VARCHAR(100) NOT NULL,
    middle_name VARCHAR(100),
    CONSTRAINT check_last_name CHECK (TRIM(last_name) <> ''),
    CONSTRAINT check_first_name CHECK (TRIM(first_name) <> ''),
    CONSTRAINT check_middle_name CHECK (middle_name IS NULL OR TRIM(middle_name) <> '')
);
COMMENT ON TABLE library.readers IS 'Таблица читателей библиотечной системы';
COMMENT ON COLUMN library.readers.last_name IS 'Фамилия читателя';
COMMENT ON COLUMN library.readers.first_name IS 'Имя читателя';
```

```
COMMENT ON COLUMN library.readers.middle_name IS 'Отчество читателя (опционально)';

CREATE TABLE library.tickets (
    id SERIAL PRIMARY KEY,
    reader_id INTEGER NOT NULL,
    created_at TIMESTAMPTZ NOT NULL DEFAULT CURRENT_TIMESTAMP,
    expires_at TIMESTAMPTZ NOT NULL,
    FOREIGN KEY (reader_id) REFERENCES library.readers(id) ON DELETE CASCADE,
    CONSTRAINT check_created_at CHECK (created_at <= expires_at)
);
COMMENT ON TABLE library.tickets IS 'Таблица читательских билетов';
COMMENT ON COLUMN library.tickets.reader_id IS 'Идентификатор читателя';
COMMENT ON COLUMN library.tickets.created_at IS 'Дата создания билета';
COMMENT ON COLUMN library.tickets.expires_at IS 'Дата истечения срока действия билета';

CREATE TABLE library.books (
    id SERIAL PRIMARY KEY,
    title VARCHAR(255) NOT NULL UNIQUE,
    CONSTRAINT check_title CHECK (TRIM(title) <> '')
);
COMMENT ON TABLE library.books IS 'Таблица книг с их заголовками';
COMMENT ON COLUMN library.books.title IS 'Название книги';

CREATE TABLE library.genres (
    id SERIAL PRIMARY KEY,
    name VARCHAR(30) NOT NULL UNIQUE,
    description TEXT,
    CONSTRAINT check_name CHECK (TRIM(name) <> ''),
    CONSTRAINT check_description CHECK (description IS NULL OR TRIM(description) <> '')
);
COMMENT ON TABLE library.genres IS 'Таблица книжных жанров';
COMMENT ON COLUMN library.genres.name IS 'Название жанра';
COMMENT ON COLUMN library.genres.description IS 'Описание жанра (опционально)';

CREATE TABLE library.book_genres (
    book_id INTEGER NOT NULL,
```

```
    genre_id INTEGER NOT NULL,
    PRIMARY KEY (book_id, genre_id),
    FOREIGN KEY (book_id) REFERENCES library.books(id) ON DELETE
CASCADE,
    FOREIGN KEY (genre_id) REFERENCES library.genres(id) ON DELETE
CASCADE
);
COMMENT ON TABLE library.book_genres IS 'Таблица связей между
книгами и жанрами';

CREATE TABLE library.authors (
    id SERIAL PRIMARY KEY,
    last_name VARCHAR(100) NOT NULL,
    first_name VARCHAR(100) NOT NULL,
    middle_name VARCHAR(100),
    born_at DATE,
    CONSTRAINT check_last_name CHECK (TRIM(last_name) <> ''),
    CONSTRAINT check_first_name CHECK (TRIM(first_name) <> ''),
    CONSTRAINT check_middle_name CHECK (middle_name IS NULL OR
TRIM(middle_name) <> ''),
    CONSTRAINT check_born_at CHECK (born_at IS NULL OR born_at <=
CURRENT_DATE)
);
COMMENT ON TABLE library.authors IS 'Таблица авторов книг';
COMMENT ON COLUMN library.authors.last_name IS 'Фамилия автора';
COMMENT ON COLUMN library.authors.first_name IS 'Имя автора';
COMMENT ON COLUMN library.authors.middle_name IS 'Отчество автора
(опционально)';
COMMENT ON COLUMN library.authors.born_at IS 'Дата рождения автора
(опционально)';

CREATE TABLE library.book_authors (
    book_id INTEGER NOT NULL,
    author_id INTEGER NOT NULL,
    PRIMARY KEY (book_id, author_id),
    FOREIGN KEY (book_id) REFERENCES library.books(id) ON DELETE
CASCADE,
    FOREIGN KEY (author_id) REFERENCES library.authors(id) ON DELETE
CASCADE
);
COMMENT ON TABLE library.book_authors IS 'Таблица связей между
книгами и авторами';

CREATE TABLE library.book_loans (
```

```
    id SERIAL PRIMARY KEY,
    book_id INTEGER NOT NULL,
    ticket_id INTEGER NOT NULL,
    picked_at TIMESTAMPTZ NOT NULL,
    pick_days INT NOT NULL,
    returned_at TIMESTAMPTZ,
    UNIQUE (book_id, ticket_id, picked_at),
    FOREIGN KEY (book_id) REFERENCES library.books(id) ON DELETE CASCADE,
    FOREIGN KEY (ticket_id) REFERENCES library.tickets(id) ON DELETE CASCADE,
    CONSTRAINT check_picked_at CHECK (picked_at <= CURRENT_TIMESTAMP),
    CONSTRAINT check_pick_days CHECK (pick_days > 0),
    CONSTRAINT check_returned_at CHECK (returned_at IS NULL OR returned_at >= picked_at)
);
COMMENT ON TABLE library.book_loans IS 'Таблица выдачи книг по читательским билетам';
COMMENT ON COLUMN library.book_loans.book_id IS 'Идентификатор книги';
COMMENT ON COLUMN library.book_loans.ticket_id IS 'Идентификатор билета';
COMMENT ON COLUMN library.book_loans.picked_at IS 'Дата выдачи книги';
COMMENT ON COLUMN library.book_loans.pick_days IS 'Срок выдачи (в днях)';
COMMENT ON COLUMN library.book_loans.returned_at IS 'Дата возврата книги (опционально)';

-- Индексы для оптимизации
CREATE INDEX idx_tickets_reader_id ON library.tickets(reader_id);
CREATE INDEX idx_book_genres_book_id ON library.book_genres(book_id);
CREATE INDEX idx_book_authors_book_id ON library.book_authors(book_id);
CREATE INDEX idx_book_loans_ticket_id ON library.book_loans(ticket_id);
CREATE INDEX idx_book_loans_book_id ON library.book_loans(book_id);

-- Триггер для проверки выдачи книги на срок, превышающий срок действия читательского билета
CREATE FUNCTION check_loan_expiry() RETURNS TRIGGER AS $$
BEGIN
  IF NEW.picked_at + NEW.pick_days * INTERVAL '1 day' > (
```

```
        SELECT expires_at FROM library.tickets WHERE id = NEW.ticket_id
    ) THEN
        RAISE EXCEPTION 'Период выдачи книги превышает срок действия читательского билета';
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER trigger_check_loan_expiry
BEFORE INSERT OR UPDATE ON library.book_loans
FOR EACH ROW EXECUTE FUNCTION check_loan_expiry();
```

2.5 Нормальные формы

Нормальные формы для таблиц представлены в таблице 1.

Таблица 1 – нормальные формы таблиц базы данных

Таблица	Нормальные формы таблицы
readers	1, 2, 3
tickets	1, 2, 3
books	1, 2, 3
genres	1, 2, 3
book_genres	1, 2, 3
authors	1, 2, 3
book_authors	1, 2, 3
book_loans	1, 2, 3

3 ЗАКЛЮЧЕНИЕ

По результатам работы был изучен теоретический материал по теме «Проектирование базы данных и ее реализация в среде СУБД PostgreSQL». Все поставленные цели и задачи были выполнены.