

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Южно-Уральский государственный университет (национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра системного программирования

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ГЕНЕРАЦИИ МУЗЫКИ

Рецензент:
Начальник отдела
суперкомпьютерного
моделирования НИУ ВШЭ,
к.ф.-м.н., доцент
П.С. Костенецкий

Научный руководитель:
доцент кафедры СП, к.ф.-м.н.
Г.И. Радченко

Автор:
студент группы КЭ-229
Д.В. Кутюшкин

Челябинск, 2024

АКТУАЛЬНОСТЬ

- Активное развитие применения нейронных сетей в области синтетических медиа.
- Сложность и затраты времени на придумывание интересной и уникальной музыки.
- Потенциал для творческого экспериментирования

ЦЕЛЬ И ЗАДАЧИ

Цель:

разработка веб-приложения для генерации музыки.

Задачи:

1. Провести обзор аналогов и литературы
2. Подготовить набор данных
3. Выбрать архитектуру нейронной сети
4. Спроектировать веб-приложение
5. Реализовать модель и провести ее обучение
6. Реализовать и протестировать веб-приложение для генерации музыки

ОБЗОР АНАЛОГОВ

Проект	Архитектура	Достоинства	Недостатки
Jukebox	Генеративно-состязательные сети	Высокий уровень детализации, включая возможность генерации вокала	Отсутствие контроля над низкоуровневыми характеристиками мелодии
MuseNet	Трансформеры	Высокий уровень пользовательского контроля над генерацией	Отсутствие контроля над низкоуровневыми характеристиками мелодии

НАБОР ДАННЫХ

- Данные извлечены из набора Lakh MIDI Dataset, содержащего множество уникальных MIDI файлов.
- MIDI файлы были предобработаны путем извлечения нескольких дорожек, содержащих массив значений на временных шагах.
- Значение на временном шаге представляет собой ноту, звучащую в текущий момент времени для соответствующей дорожки.
- Итоговое число образцов для обучения – 1000
- Размер тестовой выборки – 100
- Размер обучающей выборки – 900

СРЕДСТВА РЕАЛИЗАЦИИ НЕЙРОННОЙ СЕТИ

Язык программирования:

- Python 3.11.9

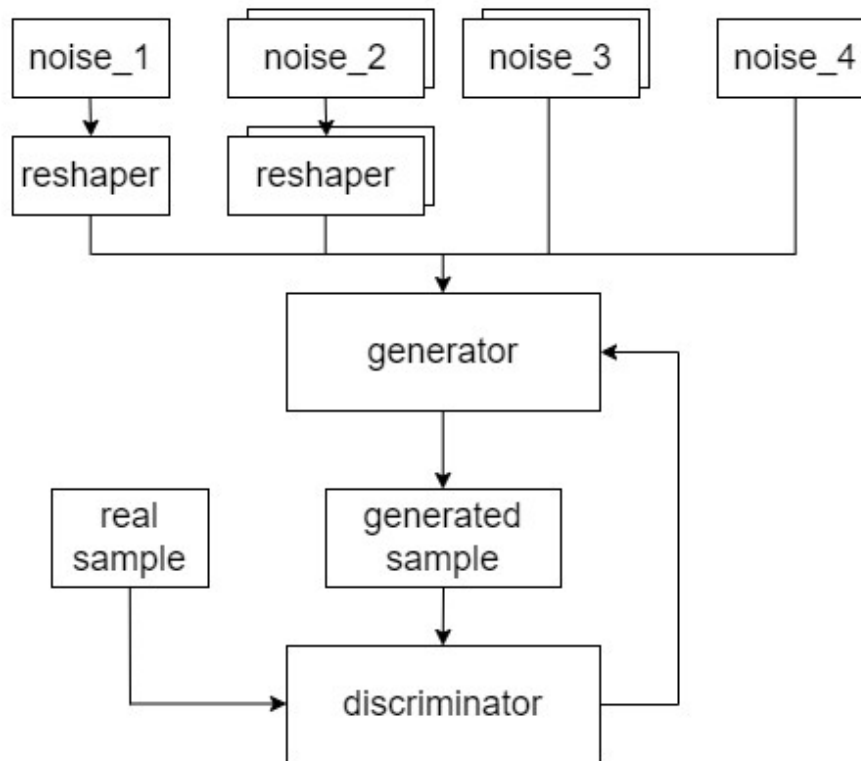
Среда разработки:

- PyCharm Community Edition 2023

Библиотеки:

- Tensorflow 2.15.0
- Music21 9.1.0

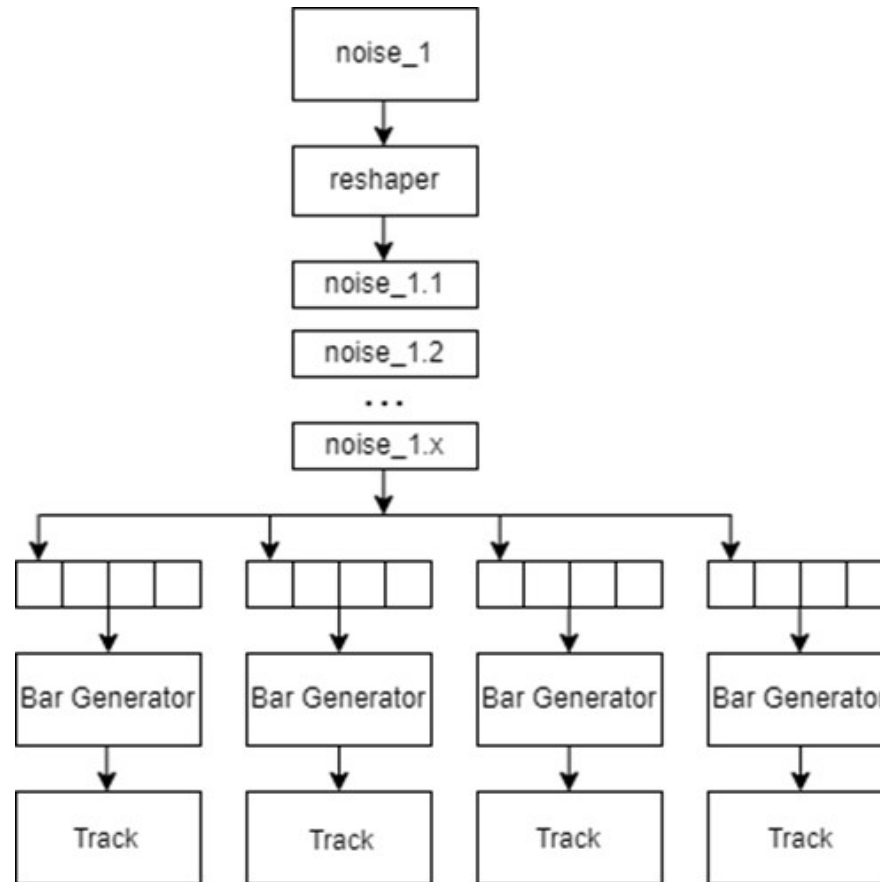
АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ



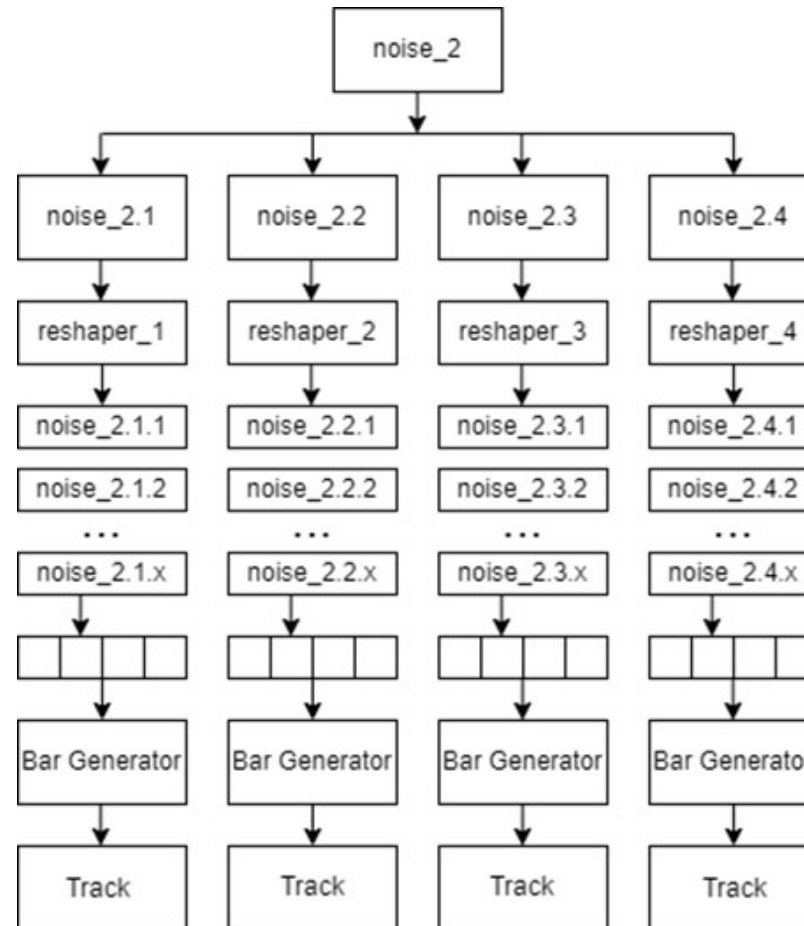
* Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, Yi-Hsuan Yang. MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment [Электронный ресурс] // arXiv.org. 2017.
Дата обновления: 24.11.2017 г. URL: <https://arxiv.org/abs/1709.06298>

Исходные коды: <https://github.com/CootKin/GraduateWork>

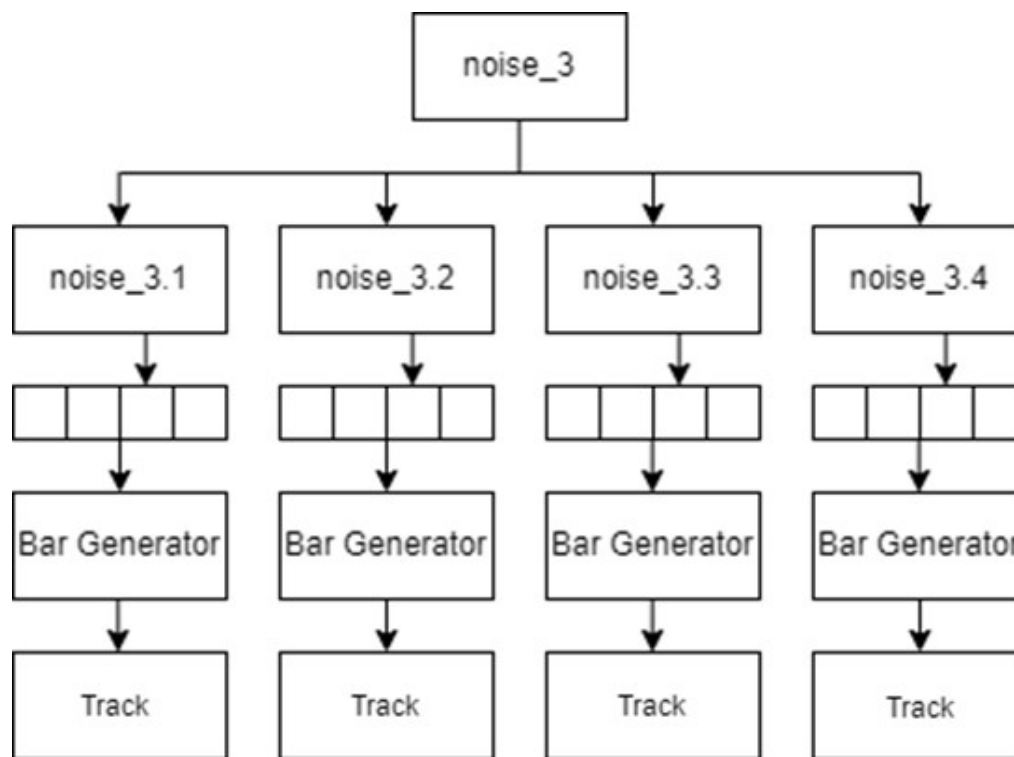
ВЕКТОРА ШУМОВ



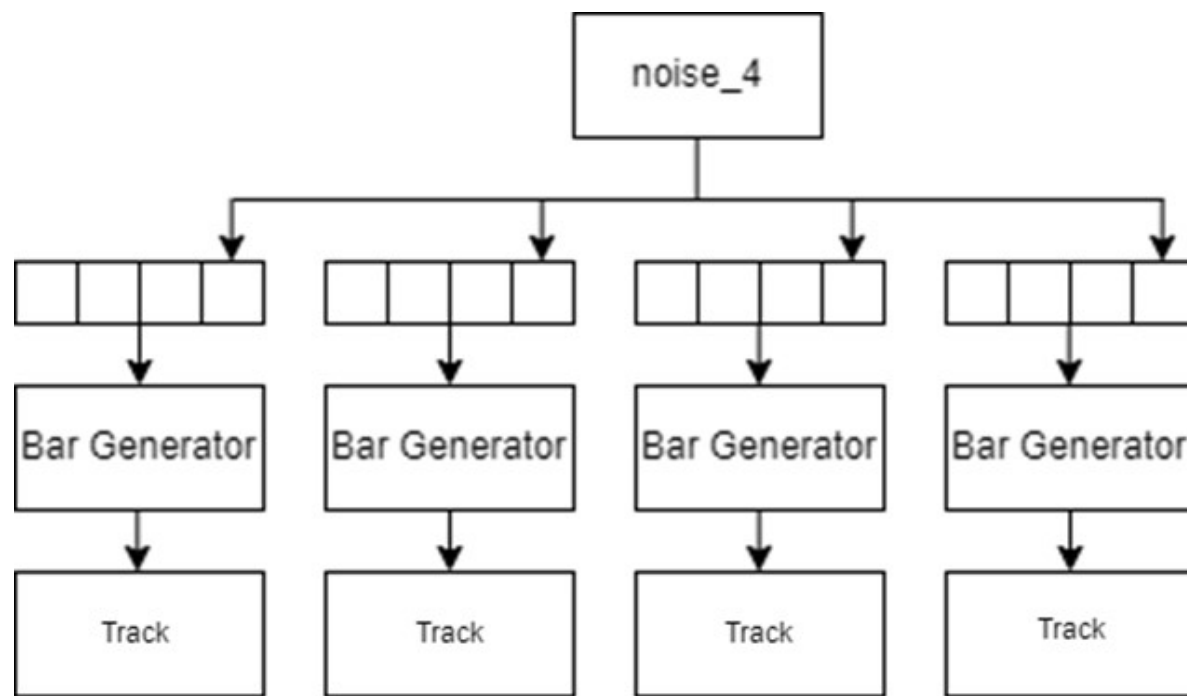
ВЕКТОРА ШУМОВ



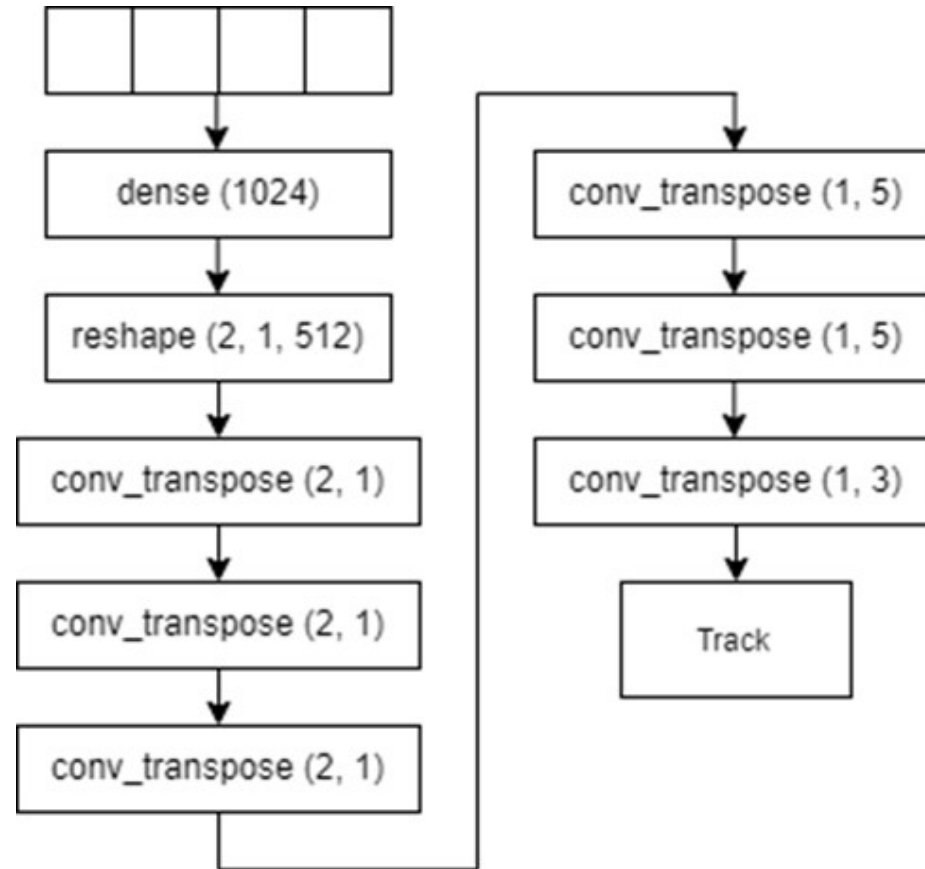
ВЕКТОРА ШУМОВ



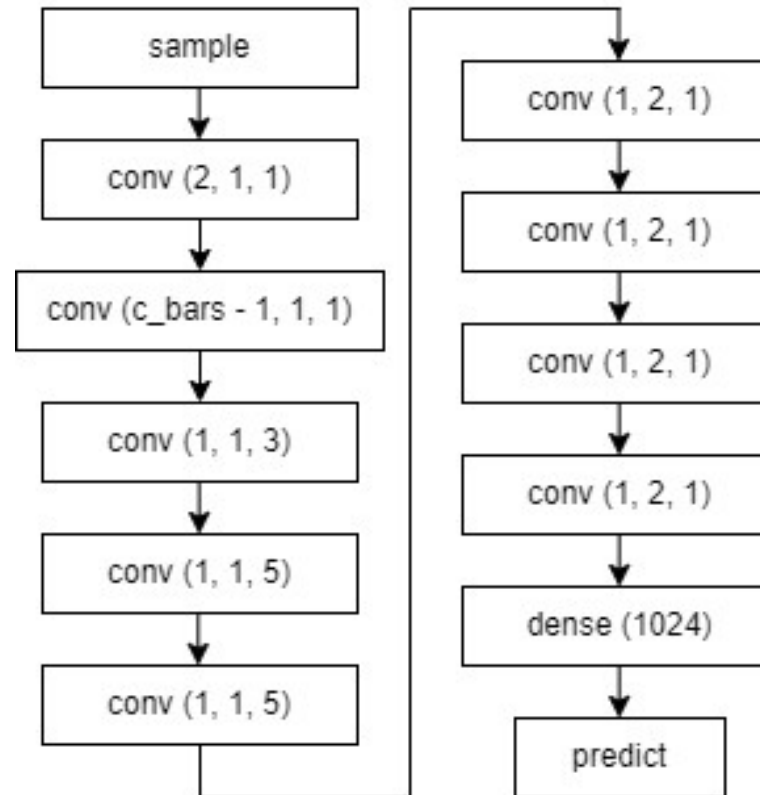
ВЕКТОРА ШУМОВ



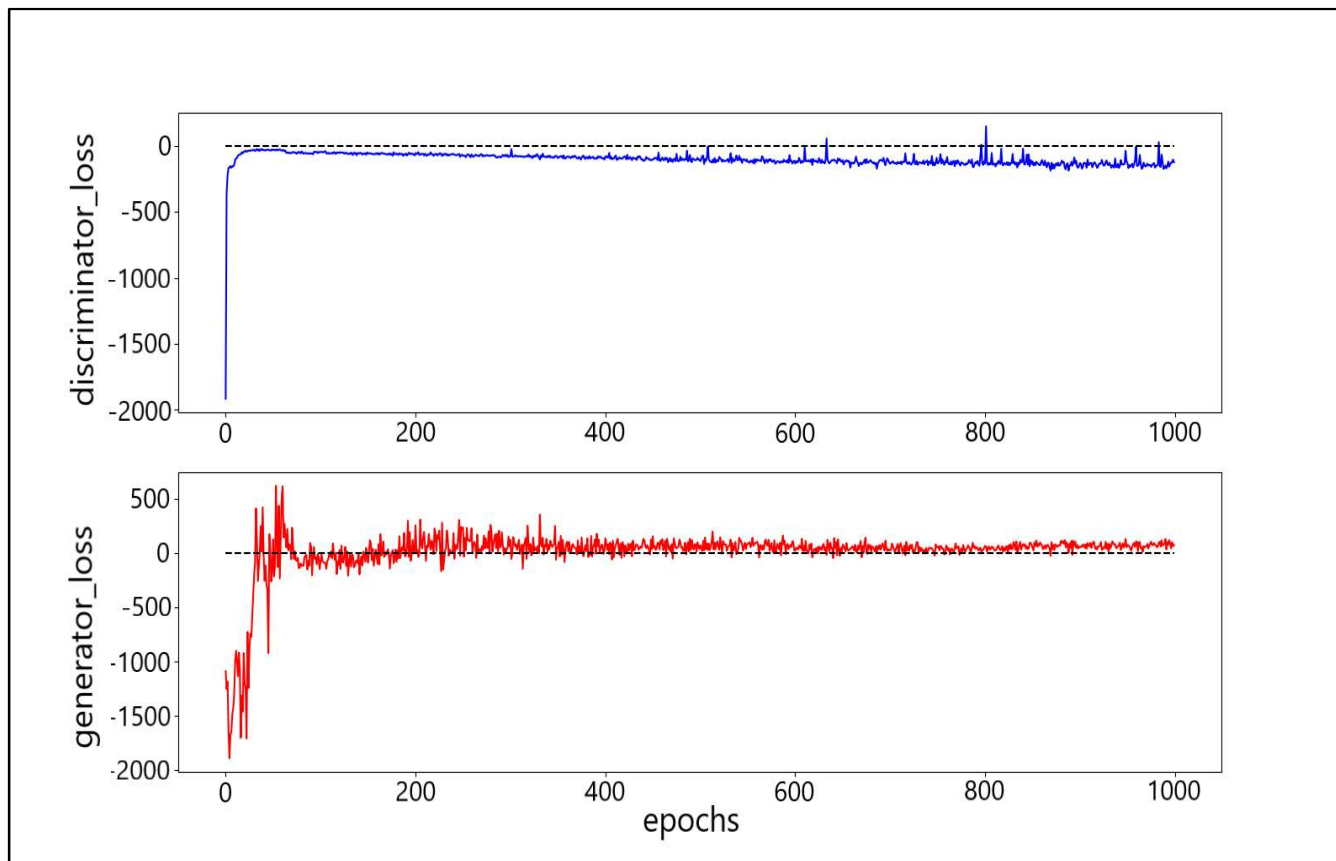
ГЕНЕРАТОР ТАКТА



ДИСКРИМИНАТОР



ГРАФИКИ ПОТЕРЬ



Исходные коды: <https://github.com/CootKin/GraduateWork>

РЕЗУЛЬТАТЫ

Евклидово расстояние между тестовой выборкой и 10 сгенерированными партитурами

- Среднее – 53.0
- Максимальное – 62.18
- Максимальное – 31.18

Евклидово расстояние между тестовой выборкой и 10 партитурами из тренировочной

- Среднее – 59.75
- Максимальное – 64.31
- Минимальное – 52.0

ДИАГРАММА ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ

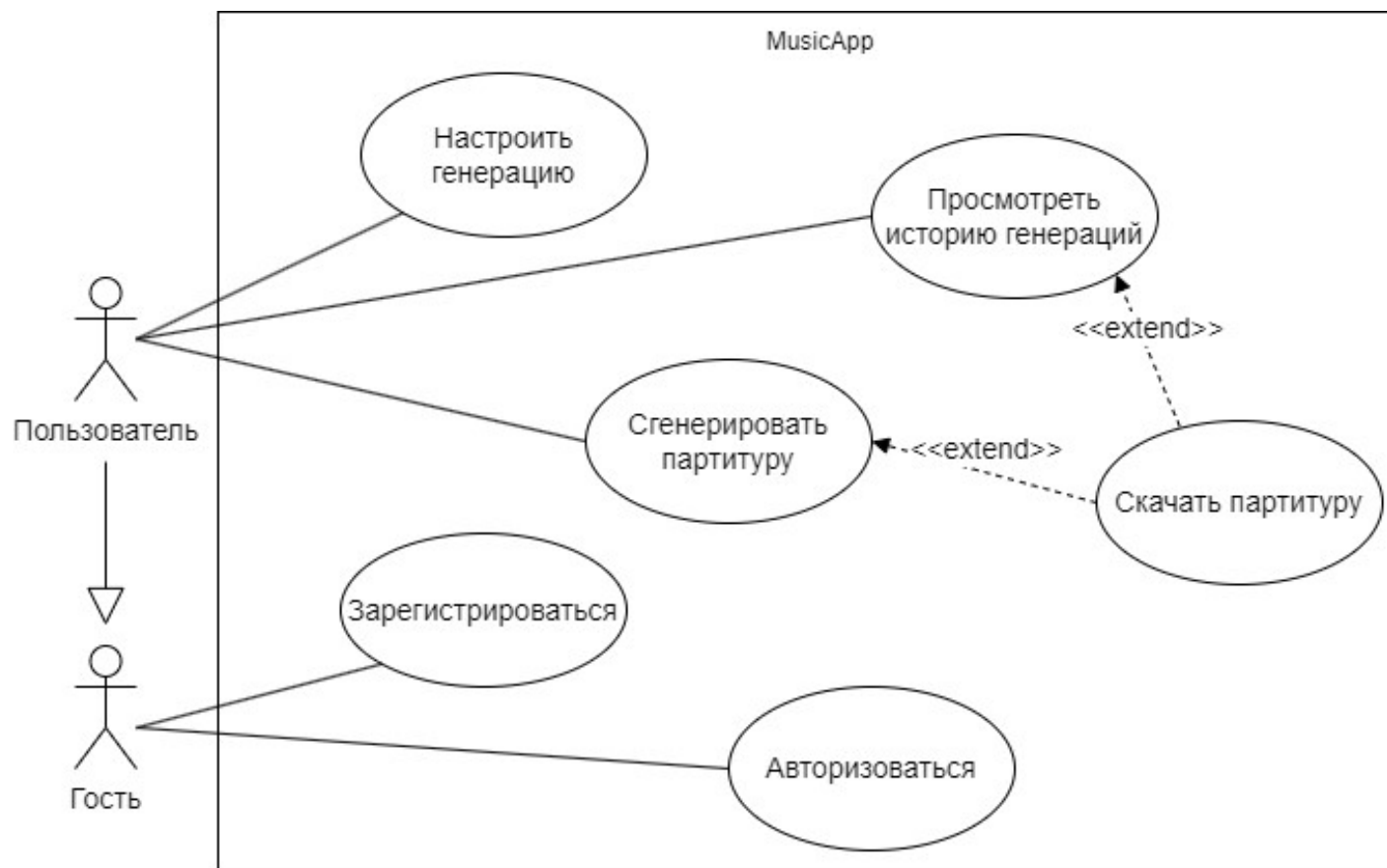


ДИАГРАММА КОМПОНЕНТОВ

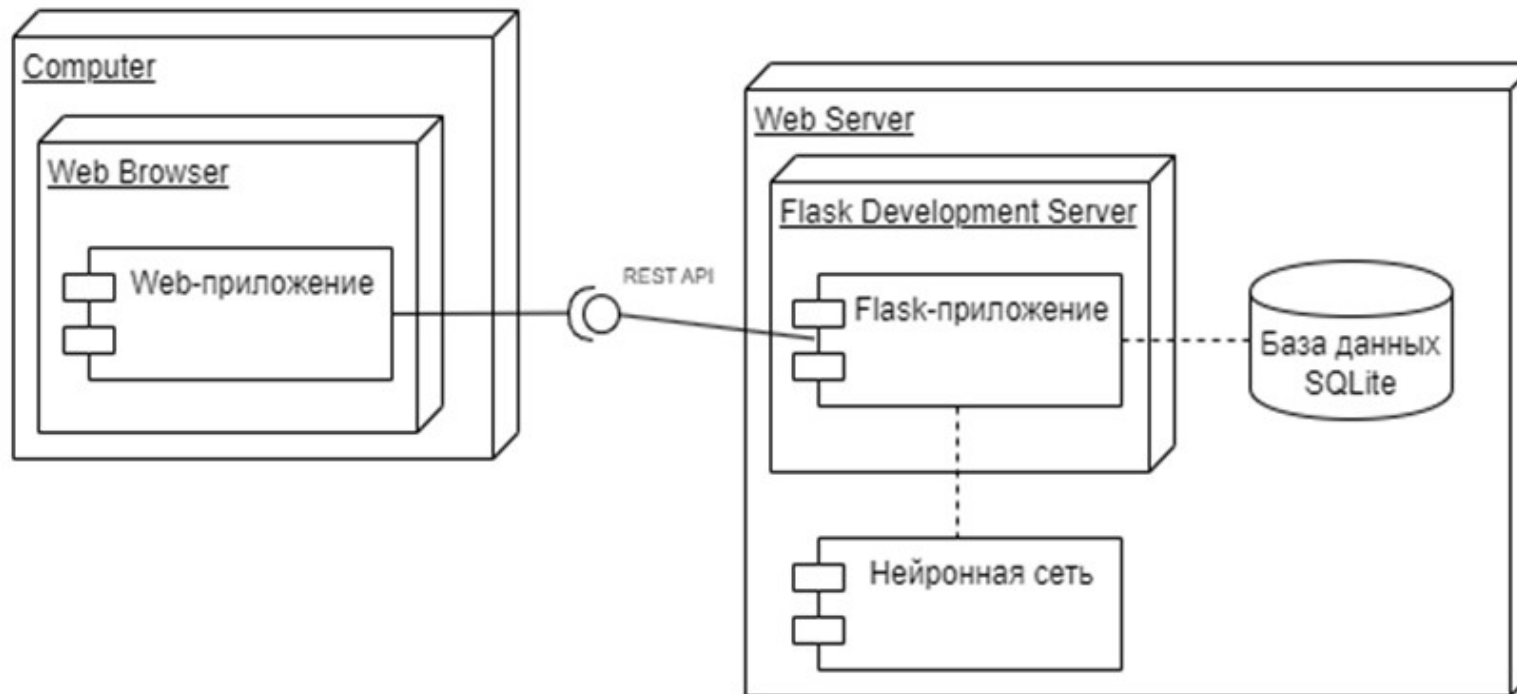
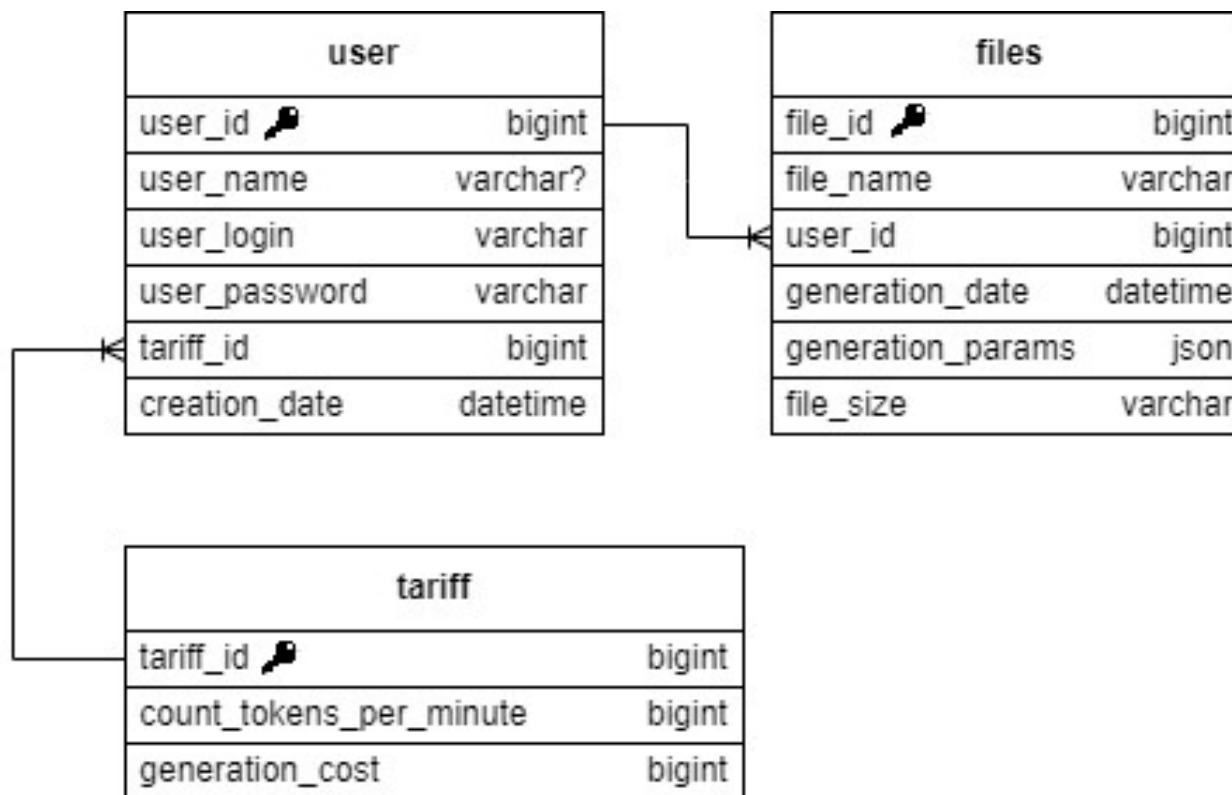


СХЕМА БАЗЫ ДАННЫХ



СРЕДСТВА РЕАЛИЗАЦИИ ВЕБ-ПРИЛОЖЕНИЯ

Среда разработки:

- PyCharm Community Edition 2023

Сервер:

- Язык программирования – Python 3.11.9
- Flask 3.0.3
- SQLAlchemy 2.0.30
- СУБД – SQLite

Клиент:

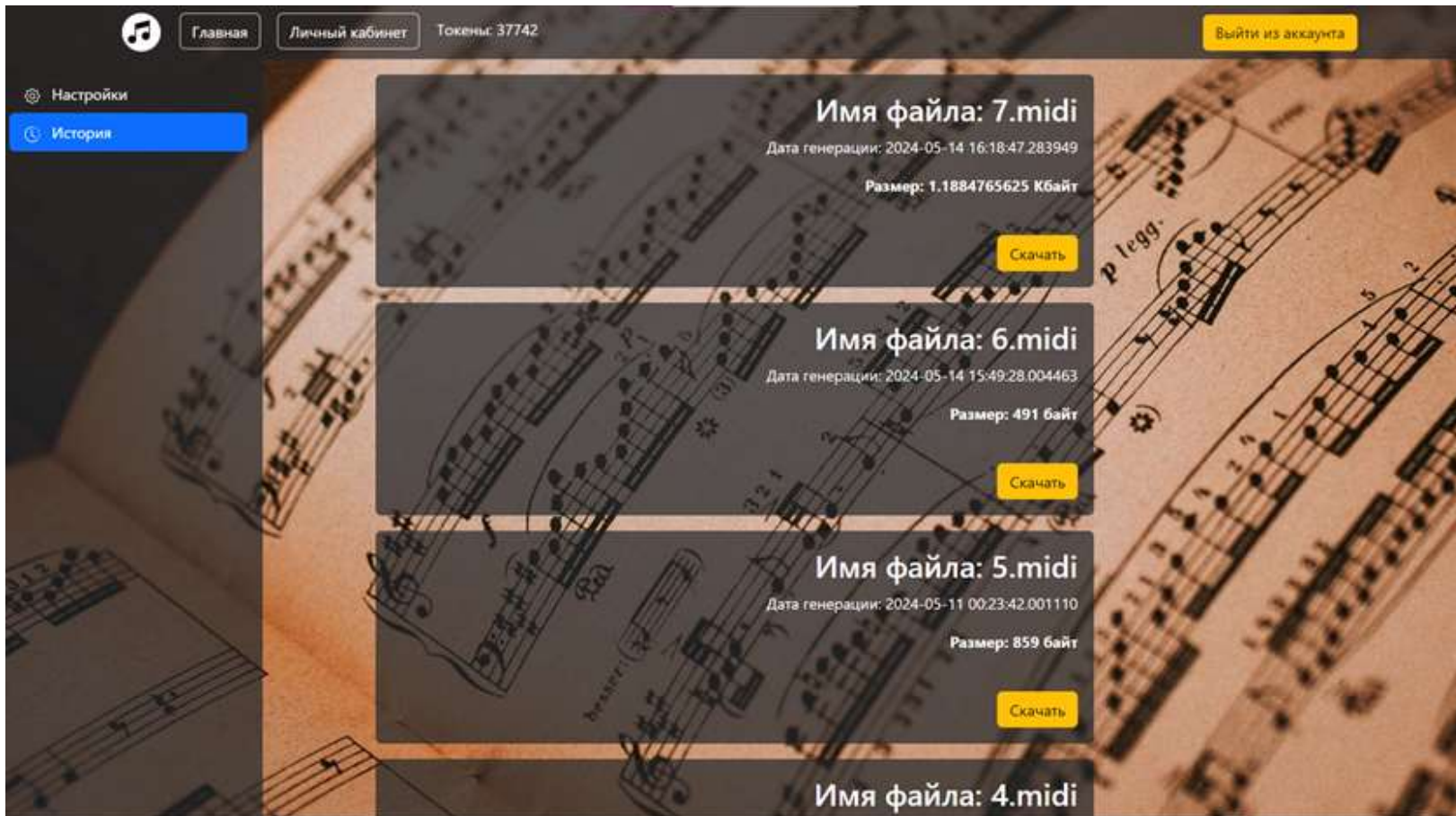
- Bootstrap 5
- JavaScript

ГЕНЕРАЦИЯ



Исходные коды: <https://github.com/CootKin/GraduateWork>

ИСТОРИЯ ГЕНЕРАЦИЙ



Исходные коды: <https://github.com/CootKin/GraduateWork>

ФУНКЦИОНАЛЬНОЕ ТЕСТИРОВАНИЕ

В ходе функционального тестирования был разработан набор тестов и приложение было протестировано на этом наборе.

Суммарно было проведено 18 тестов, все тесты были успешно пройдены.

НАПРАВЛЕНИЕ ДАЛЬНЕЙШЕЙ РАБОТЫ

1. Улучшение качества данных
2. Увеличение количества данных
3. Разделение серверной части на Docker контейнеры

ОСНОВНЫЕ РЕЗУЛЬТАТЫ

1. Проведен обзор аналогов и литературы
2. Подготовлен набор данных
3. Выбрана архитектура модели
4. Спроектировано веб-приложение
5. Реализована модель и проведено ее обучение
6. Реализовано и протестировано веб-приложение для генерации музыки

Разработанное приложение полностью соответствует поставленным требованиям.

ПРИМЕРЫ ГЕНЕРАЦИИ

1

2

3

4