

IF2123 Aljabar Linear dan Geometri

# **IMAGE RETRIEVAL DAN MUSIC INFORMATION RETRIEVAL**

## **MENGGUNAKAN PCA DAN VEKTOR**

### **Laporan Tugas Besar 2**

Disusun untuk memenuhi tugas mata kuliah Aljabar Linear dan Geometri pada Semester 1 (satu) Tahun Akademik 2024/2025



**Oleh**

**Jonathan Kenan Budianto 13523139**

**Mahesa Fadhillah Andre 13523140**

**Jovandra Otniel P. S. 13523141**

**Kelompok popmie**

**PROGRAM STUDI TEKNIK INFORMATIKA  
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG**

**2024**

## **DAFTAR ISI**

<b>DAFTAR ISI</b>	<b>2</b>
<b>BAB 1: Deskripsi Masalah</b>	<b>3</b>
Gambar 1. Shazam sebagai aplikasi audio retrieval system	3
<b>BAB 2: Teori Singkat</b>	<b>5</b>
I. Album Picture Finder – Principal Component Analysis	5
1. Image Processing dan Loading	5
2. Data Centering (Standardization)	6
3. PCA Computation Using Singular Value Decomposition (SVD)	7
4. Retrieval and Output	8
II. Music Information Retrieval – Query by Humming	8
1. Pemrosesan Audio	8
2. Ekstraksi Fitur	8
a. Distribusi Tone	8
b. Normalisasi	8
3. Perhitungan Similaritas	8
<b>BAB 3: Arsitektur Website</b>	<b>9</b>
<b>Pembuatan</b>	<b>9</b>
<b>BAB 4: Eksperimen</b>	<b>10</b>
<b>BAB 5: Kesimpulan</b>	<b>11</b>

## BAB 1: Deskripsi Masalah

Suara selalu menjadi hal yang paling penting dalam kehidupan manusia. Manusia berbicara mengeluarkan suara dan mendengarkan suatu suara untuk diresap ke otak dan mencari informasi dari suara tersebut. Suara juga bisa dijadikan orang-orang di dunia ini sebuah media untuk membuat karya seni. Contohnya adalah alat mendeteksi lagu. Manusia bisa mendeteksi suara dengan menggunakan indera pendengar dan memberikan kesimpulan akan apa jenis suara tersebut melalui respon dari otak. Sama seperti manusia, teknologi juga bisa mendeteksi suara dan memberikan jawaban mereka melalui algoritma-algoritma yang beragam bahkan bisa melebihi kapabilitas manusia. Dengan menggunakan algoritma apapun, konsep dari pendeksi dan interpretasi suara itu bisa juga disebut dengan sistem temu balik suara atau bisa disebut juga dengan *audio retrieval system*. Banyak aplikasi yang menggunakan konsep sistem temu balik contohnya adalah Shazam.



Gambar 1. Shazam sebagai aplikasi *audio retrieval system*

Selain suara, manusia juga memiliki penglihatan sebagai salah satu inderanya dan bisa melihat warna dan gambar yang bermacam-macam. Teknologi komputasi juga memiliki kapabilitas yang sama dan bisa melihat gambar sama seperti kita, tetapi teknologi seperti ini juga bisa merepresentasikan gambar tersebut sebagai beragam-ragam angka yang bisa disebut juga fitur. Tahun ke tahun, *image processing* selalu menjadi fokus utama dari tugas besar 2 Algeo. Algoritma yang digunakan adalah Eigenvalue, Cosine Similarity, Euclidean Distance, dll.

Kelompok kami sudah melewati Tugas Besar 1 yaitu tentang matriks dan implementasi terhadap berbagai hal. Matriks adalah salah satu komponen yang penting dalam aplikasi aljabar vektor. Di dalam Tugas Besar 2 ini, kami membuat semacam aplikasi Shazam yaitu sebuah aplikasi yang meminta input lagu dan aplikasi tersebut mendeteksi apa nama dari lagu tersebut dan beberapa detail lainnya. Pada tugas besar ini, kami menggunakan aljabar vektor untuk mencari perbandingan antar satu audio dengan audio yang lain. Kami menggunakan konsep yang bernama *Music Information Retrieval* atau MIR untuk mencari dan mengidentifikasi suara berdasarkan fitur-fitur yang dimilikinya. Tidak hanya itu, kami juga menggunakan konsep Principal Component Analysis (PCA) untuk mencari kumpulan audio melalui deteksi wajah berbagai orang (anggap saja mereka sebagai seorang penyanyi).

## **BAB 2: Teori Singkat**

Information retrieval merupakan proses mencari informasi dari sebuah kumpulan data berdasarkan input tertentu. Terdapat beberapa jenis information retrieval. Namun, pada tugas besar ini, information retrieval yang diaplikasikan adalah *image retrieval* dan *music information retrieval*. Kedua jenis information retrieval tersebut akan diimplementasikan dengan menggunakan Principal Component Analysis dan Music Information Retrieval menggunakan humming.

### **I. Album Picture Finder – Principal Component Analysis**

Album picture finder mencari informasi sebuah lagu dengan mencari gambar cover album dari lagu tersebut. Untuk menemukan informasi dari lagu tersebut, album picture finder menggunakan teknik PCA untuk mencocokkan input gambar album dengan gambar album yang ada pada sebuah database.

Principal Component Analysis merupakan teknik dalam ilmu statistika untuk mereduksi dimensi sebuah data dengan mempertahankan informasi sebanyak mungkin. Dimensi data yang telah tereduksi disebut principal components. Hasil data yang didapatkan dari proses tersebut merupakan eigenvector dan proyeksi data.

Album Picture Finder menggunakan PCA dilakukan dengan langkah-langkah berikut:

#### **1. Image Processing dan Loading**

Seluruh dataset diproses menjadi bentuk grayscale untuk membuat fokus pencarian berdasarkan intensitas gelap terang gambar saja (proses ini mengurangi kompleksitas gambar). Pemrosesan grayscale menggunakan rumus berikut:

$$I(x,y) = 0.2989 \cdot R(x,y) + 0.5870 \cdot G(x,y) + 0.1140 \cdot B(x,y)$$

Vektor grayscale yang didapatkan diubah menjadi 1 dimensi untuk dilakukan pemrosesan data. Apabila data memiliki dimensi  $M \times N$ , maka hasilnya adalah vektor dengan panjang  $M \cdot N$ :

$$I = [I_1, I_2, \dots, I_{M \cdot N}]$$

Terkait dengan transformasi linier dalam aljabar linier, di mana komponen warna RGB dipetakan ke intensitas grayscale. Konversi ini bisa dianggap sebagai operasi perkalian matriks pada vektor RGB, yang kemudian diubah menjadi vektor satu dimensi melalui vektorisasi untuk mempermudah pemrosesan data. Proses ini juga berhubungan dengan reduksi dimensi, mengurangi kompleksitas gambar untuk analisis lebih lanjut dalam pengolahan citra.

## 2. Data Centering (Standardization)

Setelah data diproses jadi grayscale, data akan distandarisasi pada sekitar nilai 0 menggunakan rumus berikut:

$$\mu_j = \frac{1}{N} \sum_{i=1}^N x_{ij}$$

dengan:

- $x_{ij}$ : Nilai piksel ke- $j$  gambar ke- $i$ ,
- $N$ : Jumlah total gambar dalam dataset.

Setelah itu, piksel dikurangi dengan rata-rata yang didapatkan dari standarisasi.

$$x'_{ij} = x_{ij} - \mu_j$$

Data centering meminimalkan bias pemusatan dan mempermudah analisis lanjutan seperti normalisasi atau reduksi dimensi.

### **3. PCA Computation Using Singular Value Decomposition (SVD)**

Semua gambar yang telah distandarisasi akan dilakukan perhitungan SVD. Dibuat matriks kovarians dengan rumus berikut:

$$C = \frac{1}{N} X'^T X'$$

dengan:

- $X'$ : matriks data yang sudah distandarisasi.

Lalu dilakukan dekomposisi nilai singular untuk mendapatkan komponen utama:

$$\mathbf{C} = \mathbf{U}\Sigma\mathbf{U}^T$$

Proses ini memanfaatkan ilmu teori aljabar linear dan statistika, khususnya dalam perhitungan matriks kovarians dan dekomposisi nilai singular (Singular Value Decomposition/SVD).

Singular Value Decomposition atau SVD merupakan teknik dalam aljabar linier dan geometri untuk memecah sebuah matriks A menjadi 3 komponen utama.

### **4. Retrieval and Output**

Hal pertama yang harus dilakukan yaitu pengguna mengunggah database untuk gambar, suara, dan mapper. Kumpulkan gambar-gambar yang serupa dengan permintaan masukan dengan cara membatasi jumlah atau menetapkan batas jarak Euclidean. Hasil pencarian gambar bisa digabungkan dengan hasil pencarian suara, atau dijalankan secara terpisah. Pencarian gambar akan menampilkan kumpulan gambar yang serupa disertakan link audio yang bersangkutan dengan gambar, dan sebaliknya. Hasil akan tertampak di sebelah kanan tampilan aplikasi.

## **II. Music Information Retrieval – Query by Humming**

Query by humming merupakan proses pencarian informasi dari sebuah data set audio menggunakan query audio berupa "humming" bagian dari sebuah audio/lagu oleh user. Query juga dapat berupa potongan dari sebuah audio/lagu. Jenis file audio yang dipakai dalam query by humming adalah file .MID atau MIDI file.

Musical Instrument Digital Interface merupakan sebuah protokol standar yang digunakan untuk menghubungkan alat musik dengan sebuah komputer atau hardware audio lainnya untuk dikendalikan. MIDI file atau file .MID merupakan file yang berisi instruksi alat digital untuk memainkan audio dari komputer untuk menghasilkan musik.

File midi terdiri dari beberapa komponen:

**1. Nada (Note)**

Komponen nada menyimpan instruksi untuk memainkan note (C, D, E,...).

Komponen nada juga menyimpan oktaf (interval nada pada tingkat tertentu, misalnya C pada oktaf 4 suaranya lebih "tinggi" dari C pada oktaf 5).

**2. Instrumen**

Komponen instrumen menyimpan instruksi untuk memainkan alat musik yang terdapat (dan juga dimainkan melalui) General MIDI Instrument Set.

**3. Tempo dan Waktu**

Tempo dan waktu menyimpan instruksi yang menentukan kecepatan lagu (tempo) dan juga struktur ritme.

**4. Volume dan Dinamika**

Komponen ini mengatur intensitas (keras atau lembut) nada yang dimainkan.

## 5. Kontrol lain

Terdapat pitch bend atau perubahan nada dan juga panning atau posisi stereo.

Pemrosesan query by humming dilakukan dalam beberapa langkah:

### 1. Pemrosesan Audio

MIDI file diproses dengan fokus pada track melodi utama (umumnya pada Channel 1). Setiap file MIDI akan melalui proses windowing (membagi MIDI file menjadi beberapa "window") dengan ukuran 1 window 20–40 beat. Setiap window akan memiliki slide window sebesar 4–8 beat (ditentukan oleh pembuat software). Windowing akan memudahkan pencocokan audio query yang berbentuk potongan dari sebuah MIDI file

Proses windowing juga disertai normalisasi note pada setiap window dengan rumus berikut:

$$NP(note) = \frac{(note - \mu)}{\sigma}$$

dengan  $\mu$  adalah rata-rata dari pitch dan  $\sigma$  adalah standar deviasi setiap pitch.

### 2. Ekstraksi Fitur

#### a. Distribusi Tone

Distribusi Tone merupakan proses pengolahan audio yang merujuk kepada suatu karakteristik tertentu dari audio. Distribusi Tone yang dipakai pada proses ini adalah Absolute Tone Based (ATB), Relative Tone Based (RTB), dan First Tone Based (FTB).

### **1. Absolute Tone Based (ATB)**

Analisis pada ATB didasarkan oleh nilai absolut intensitas tertentu tanpa mengkhawatirkan hubungan relatif dengan nilai lain. Pada proses ini, ATB menghitung frekuensi kemunculan setiap nada berdasarkan skala MIDI (0-127). Histogram yang dihasilkan akan memberikan gambaran distribusi absolut nada dalam data untuk menangkap karakteristik statis melodi dalam sinyal audio.

### **2. Relative Tone Based (RTB)**

RTB menganalisis perubahan pada setiap nada secara konsekutif. Histogram yang dihasilkan terdiri dari nilai -127 hingga 127. RTB berguna untuk menganalisis pola interval melodi yang lebih relevan dalam proses pencocokan humming dengan dataset (tanpa bergantung pada pitch absolut).

Pada proses ini dibuat histogram dengan total bin 255 (dengan range -127 – +128) Kemudian dihitung selisih antara nada dalam data.

### **3. First Tone Based (FTB)**

FTB menganalisis perubahan setiap nada dengan nada pertama. Histogram yang dihasilkan terdiri dari nilai -127 hingga 127. FTB berguna untuk menganalisis hubungan relatif terhadap titik referensi awal untuk menangkap struktur relatif nada yang lebih stabil terhadap variasi pitch pengguna.

### **b. Normalisasi**

Pada proses ini, setiap histogram dan nilai dalam histogram dipastikan dalam skala probabilitas. Digunakan formula umum normalisasi sebagai berikut:

$$H_{norm} = \frac{H[d]}{\sum_d^{127} H[d]}$$

Dengan H adalah Histogram dan d adalah bin dari histogram.

### **3. Perhitungan Similaritas**

Perhitungan similaritas pada proses ini menggunakan cosine similarity. Cosine similarity merupakan konsep dalam Aljabar Linier dan Geometri untuk mencari tingkat kesamaan antara dua vektor dalam ruang berdimensi tinggi dengan menghitung sudut cosinus yang dibentuk antara kedua vektor (range nilai 0–1). Semakin dekat dengan nilai 1, tingkat kemiripan antara dua vektor semakin tinggi.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

## **BAB 3: Arsitektur Website**

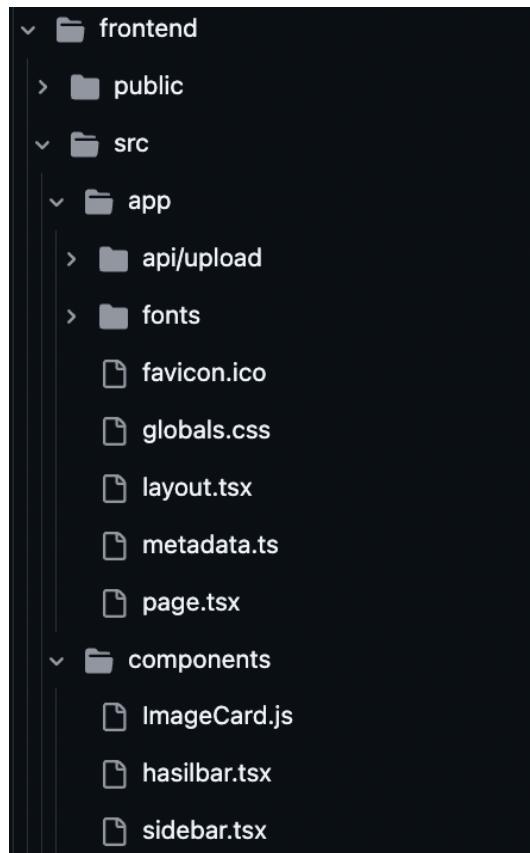
Pembuatan struktur dan pengaplikasian website menggunakan Nextjs. Struktur directory yang digunakan yaitu meliputi layout yang terdiri dari page, bar untuk hasil, dan sidebar. Di bagian page, hanya terdapat main background atau background utama yang akan digunakan. Di bagian sidebar terdiri dari logo yang berada di paling atas sidebar. Di bawah logo terdapat 7 "button" atau tombol. Terdapat tombol untuk mengunggah gambar dan audio yang ingin dicari kemiripannya, beserta tombol "search" untuk gambar dan audio. Kemudian ada 3 buah tombol untuk mengunggah database yang ingin digunakan, yaitu File Picture Zip, File Audio Zip, dan juga File Mapper Zip. Di bawah semua, ditampilkan nama file zip yang telah diunggah. Di bagian bar hasil atau "hasilbar", terdiri dari hasil pencarian gambar dan audio yang ditemukan setelah pengguna menekan tombol "search". Penampilan gambar dan audio berupa grid. Metode penampilan gambar dan audio berupa page yang menggunakan fitur pagination.

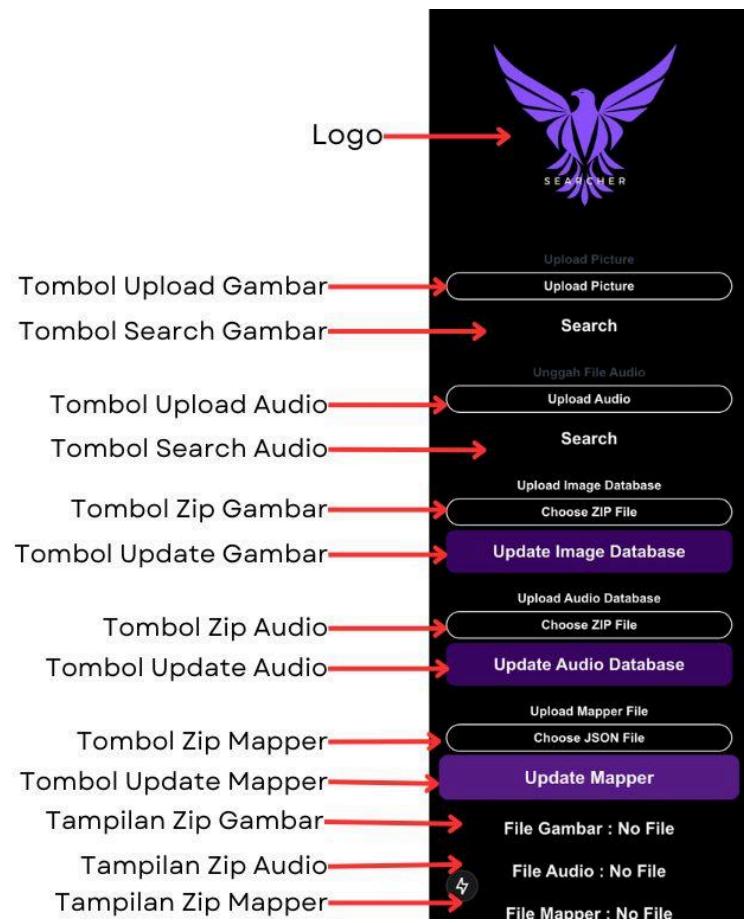
Website ini menggunakan framework Python Flask sebagai server backend. Pada kode finder.py dari folder src/backend, terdapat 5 fungsi yang menjadi fitur utama website ini. Kelima fungsi tersebut menerima, menyimpan, memproses, dan mengirim audio (MIDI), gambar (JPG), dan mapper (JSON) dari request website frontend (Next.js).

Pertama, handle\_image\_upload mencari gambar serupa dari database dengan algoritma PCA dan menghasilkan daftar gambar serupa dan informasi audio yang dihubungkan mapper (file MIDI, judul, penyanyi, album, tahun). Kedua, handle\_midi\_upload mencari audio serupa dari database dengan algoritma MIR dan menghasilkan audio serupa beserta informasi yang dikaitkan mapper (file gambar, judul, penyanyi, album, tahun).

Ketiga, handle\_image\_zip\_upload menerima kumpulan gambar dalam bentuk folder terkompres (ZIP), mengekstraknya, dan menyimpannya dalam bentuk file utuh (JPG) dan data vektor (NPZ). Keempat, handle\_midi\_zip\_upload menerima kumpulan audio dalam bentuk ZIP, mengekstraknya, dan menyimpannya dalam bentuk file utuh (MID) dan data vektor (NPZ). Terakhir, handle\_mapper\_upload menerima dan menyimpan file mapper dalam bentuk JSON.

NPZ dipakai untuk mempercepat penghitungan vektor terdekat antar file, JSON dipakai untuk *readability* pemetaan gambar, audio, beserta informasi yang terkait. ZIP diterima sebagai format database gambar/audio karena bersifat universal (hampir semua komputer mempunyai algoritma untuk membukanya).

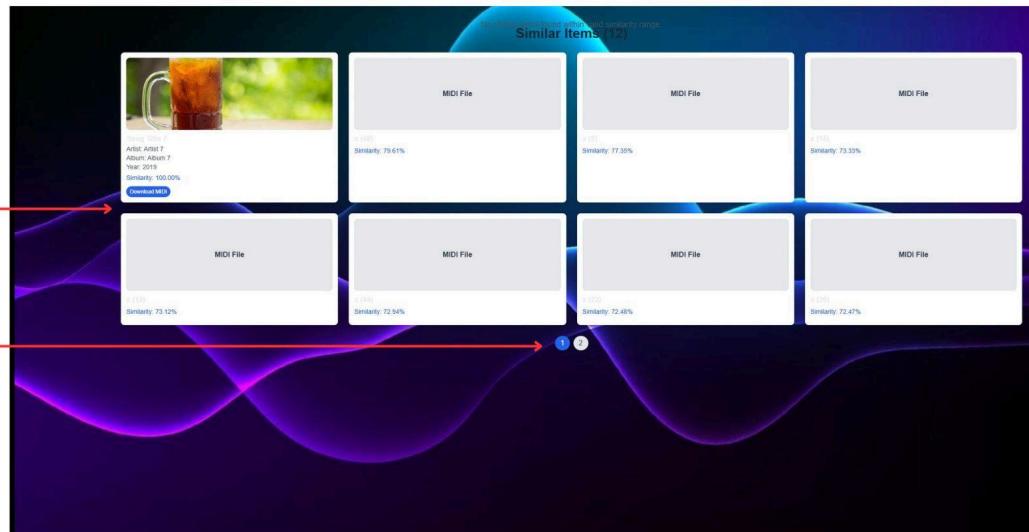




Hasil :

- 1.Gambar
- 2.Artis
- 3.Album
- 4.Year
- 5.Similarity
- 6.Suara

Pagination



## BAB 4: Eksperimen

### 1. Setup Dataset MIDI, Images, Mapper.

The image consists of two screenshots. The top screenshot shows a web application interface at [localhost:3000](http://localhost:3000). The page has a dark background with a large, abstract blue and purple wavy graphic on the right. On the left, there is a sidebar with various upload and search buttons:

- Upload Picture
- Search
- Upload File Audio
- Search
- Upload Image Database  
Choose ZIP file  
Update Image Database
- Upload Audio Database  
Choose ZIP file  
Update Audio Database
- Upload Mapper File  
Choose JSON file  
Update Mapper

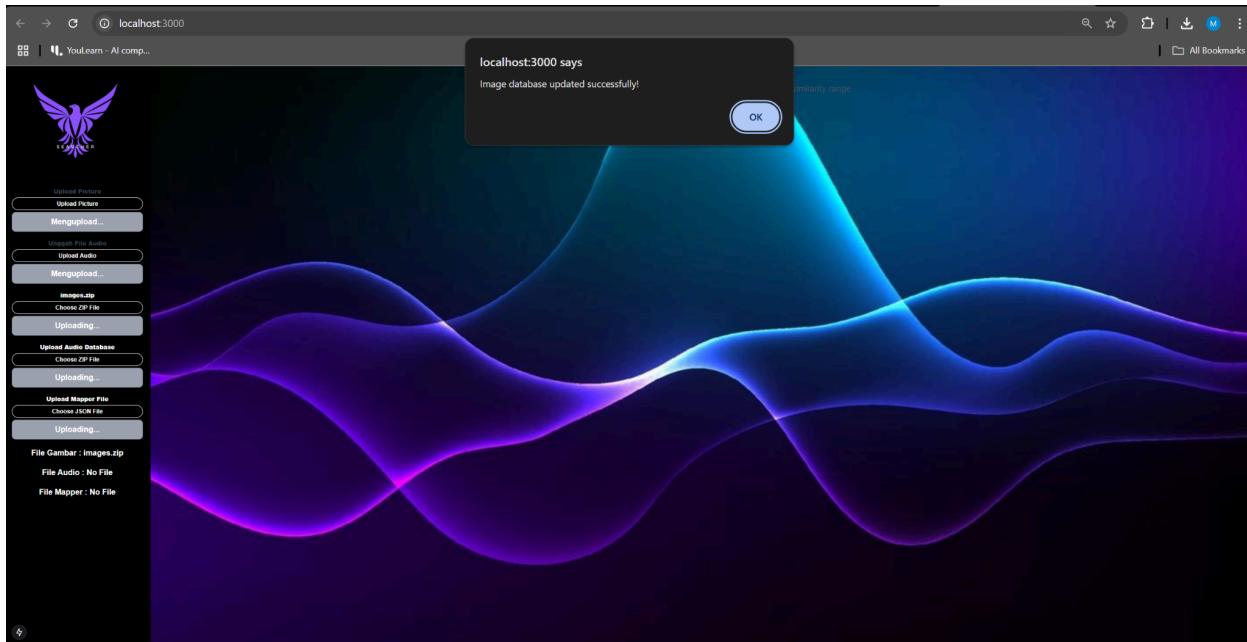
Below the sidebar, there are three status messages:

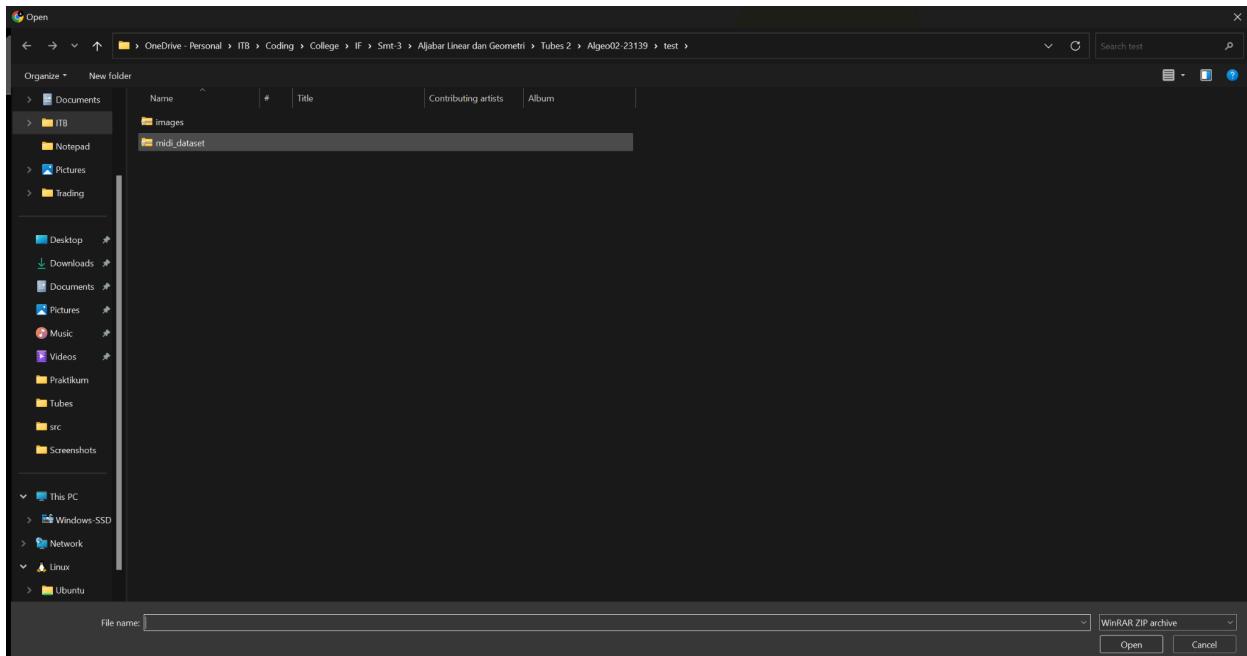
- File Gambar : No File
- File Audio : No File
- File Mapper : No File

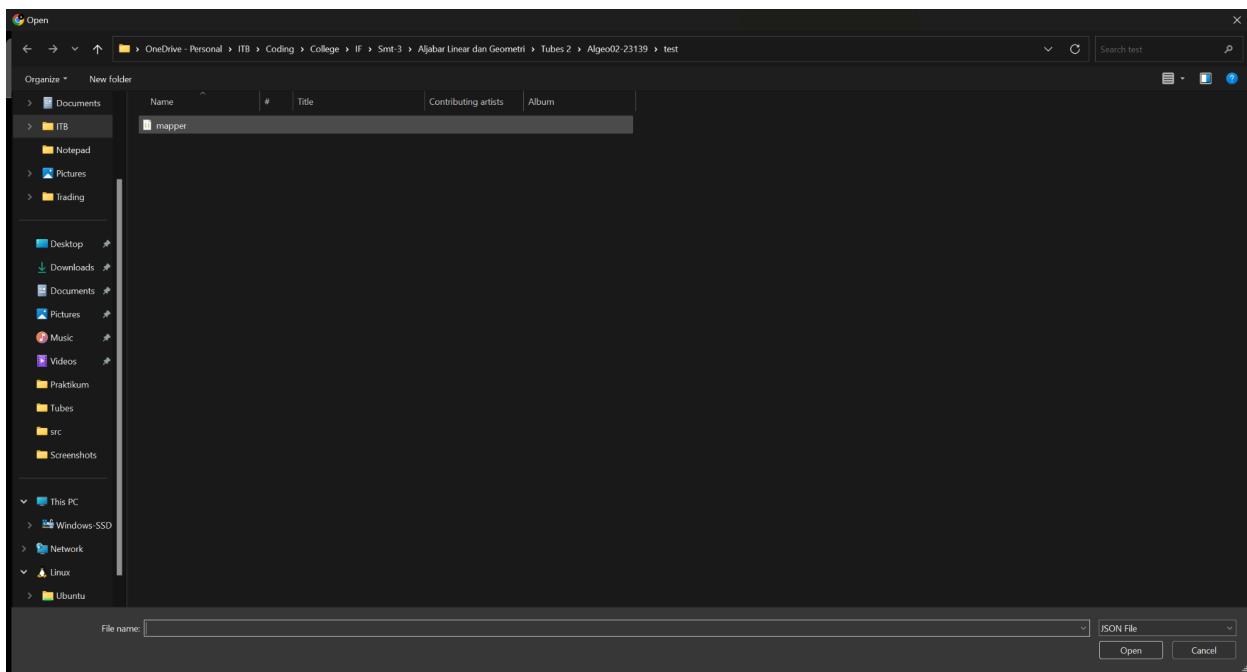
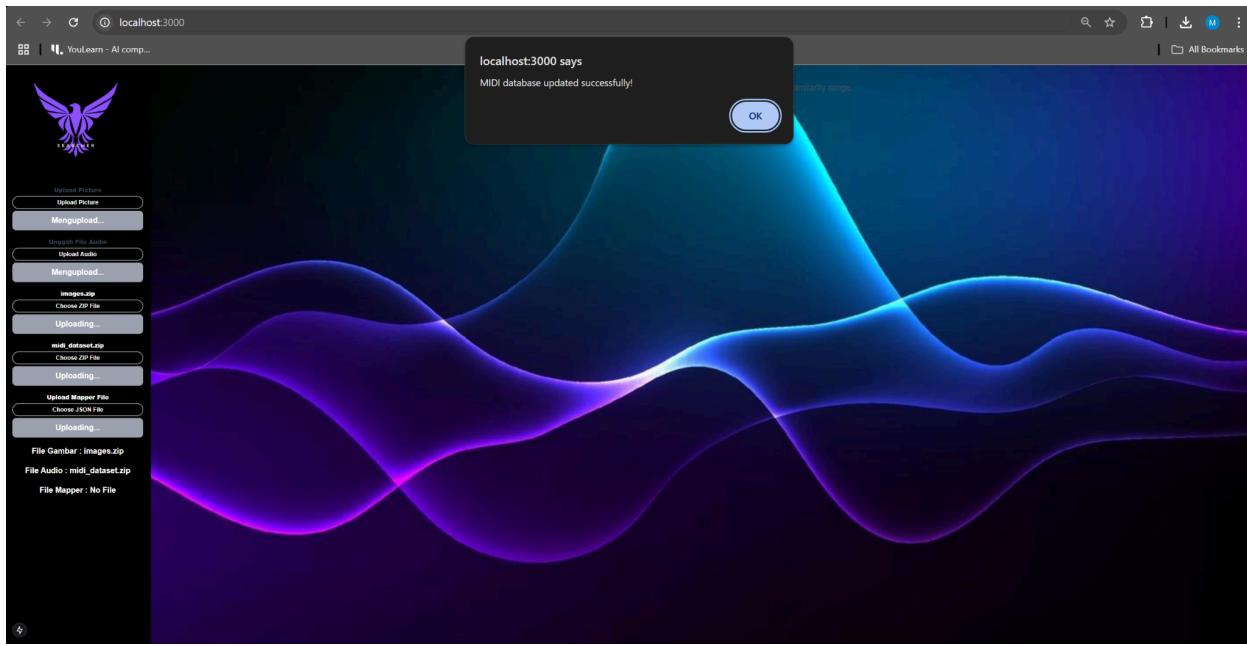
The bottom screenshot shows a Windows file explorer window titled "Open". The path in the address bar is: OneDrive - Personal > ITB > Coding > College > IF > Smt-3 > Aljabar Linear dan Geometri > Tubes 2 > Algeo02-23139 > test >. The "images" folder is selected. The file list shows:

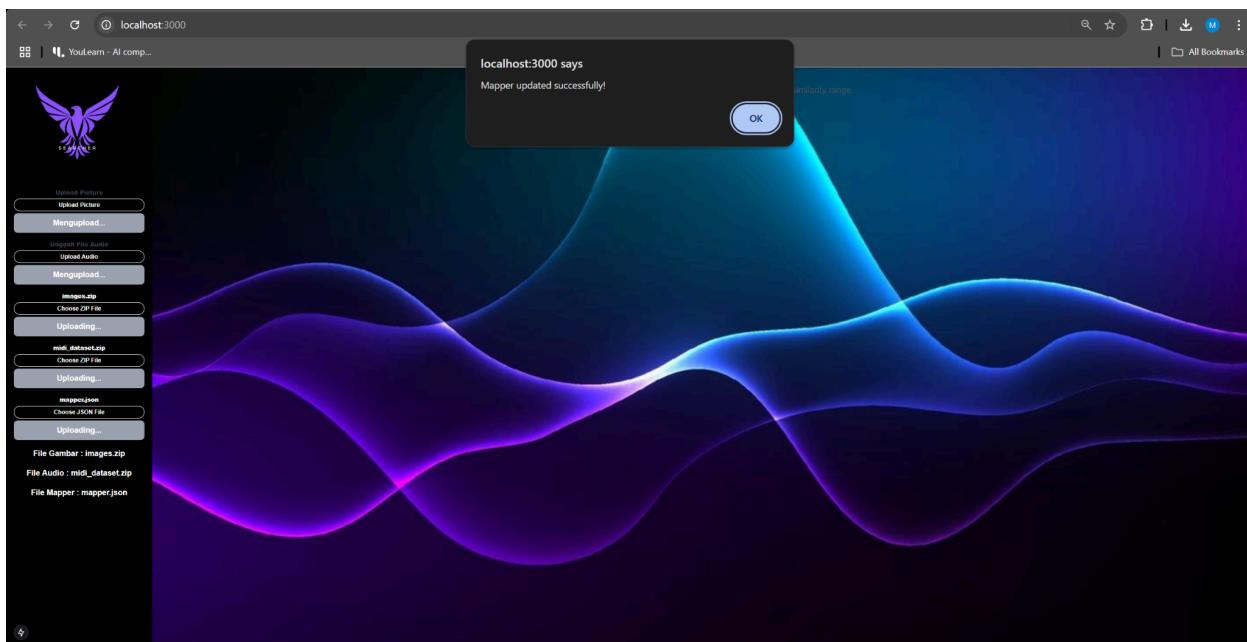
- images
- midi\_dataset

The file explorer interface includes standard buttons for Open, Cancel, and WinRAR ZIP archive.

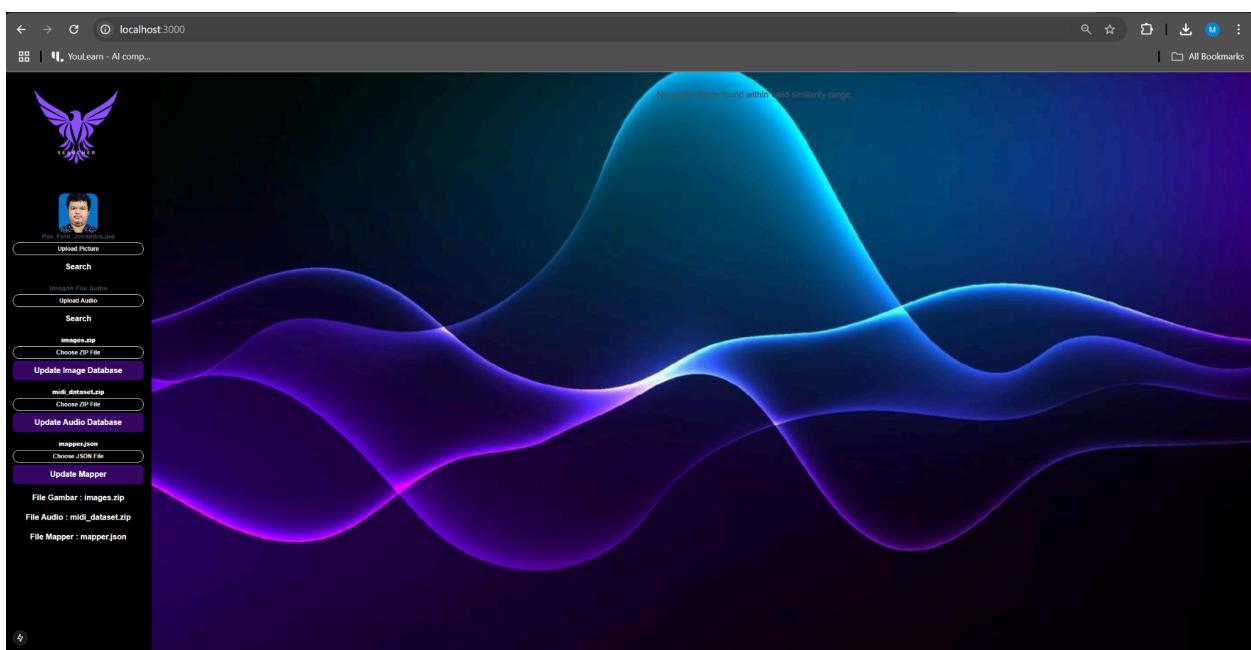
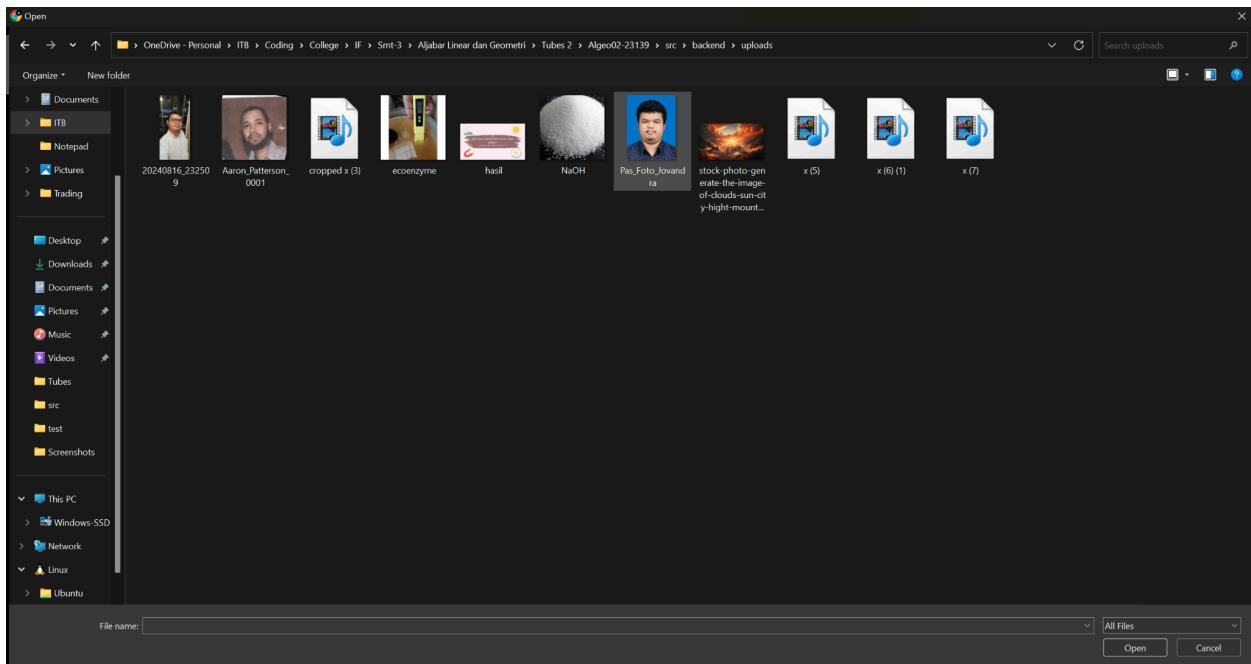


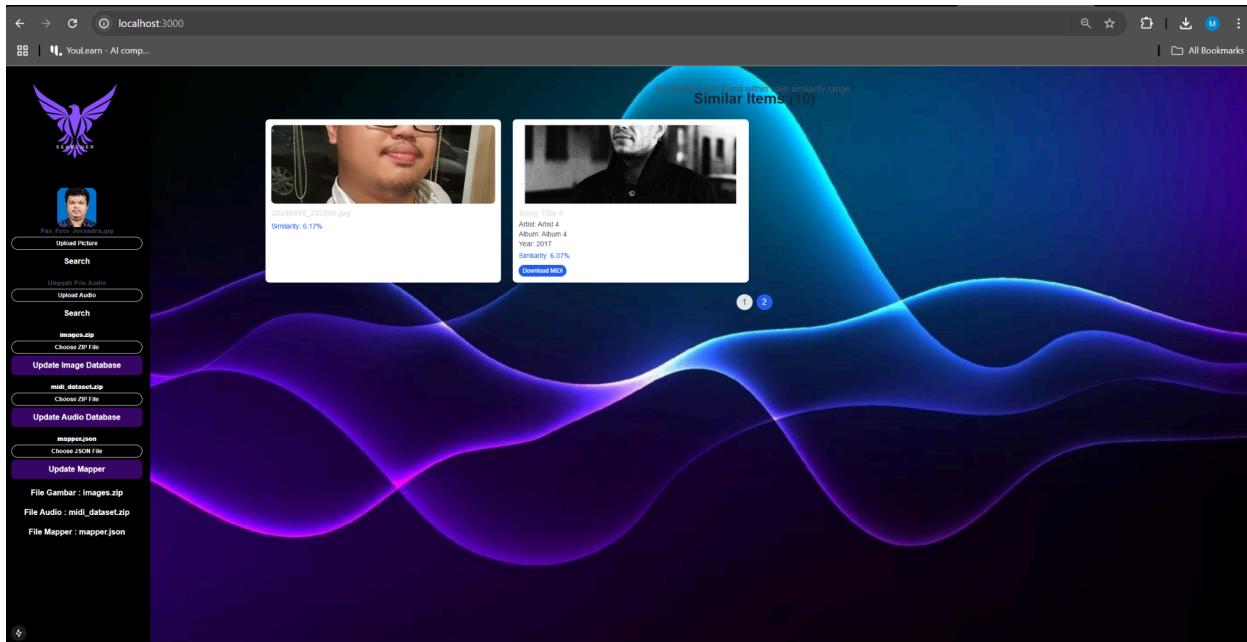
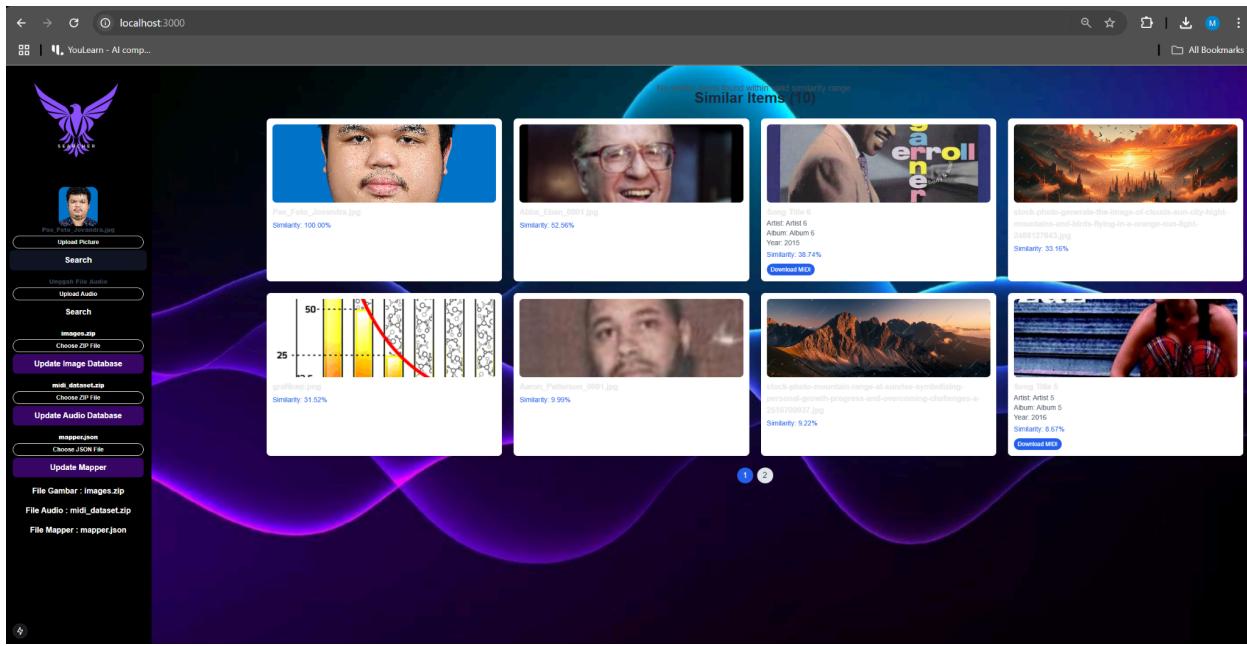




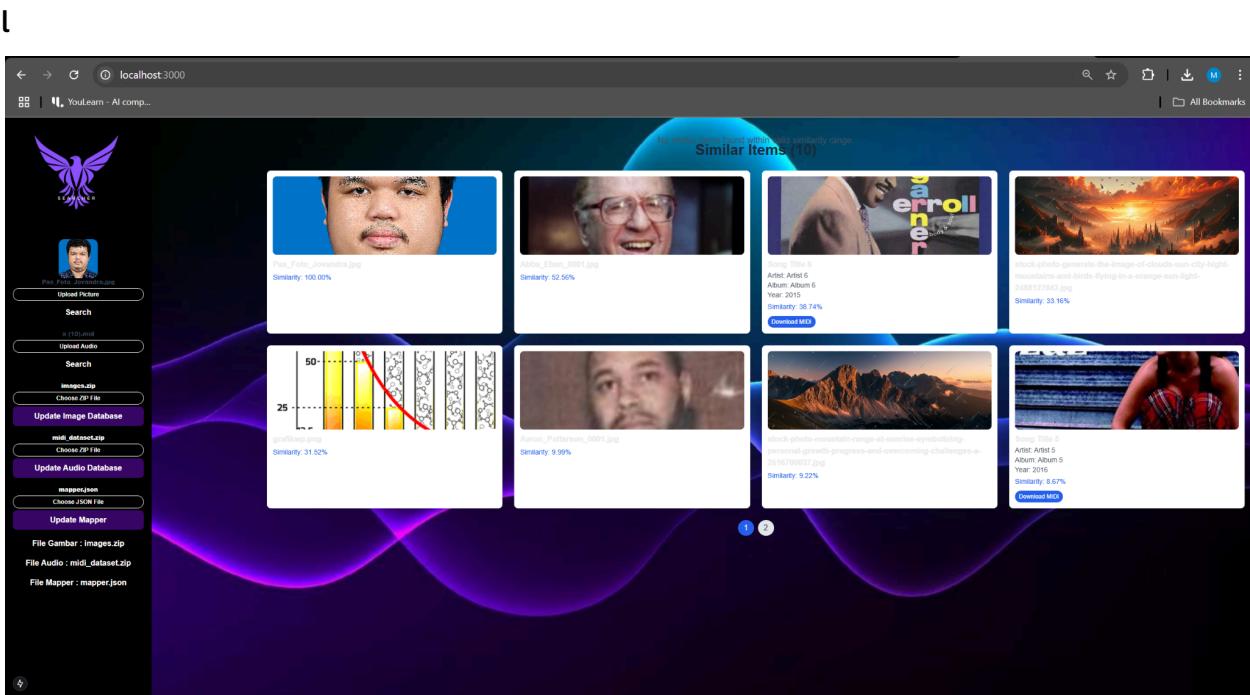
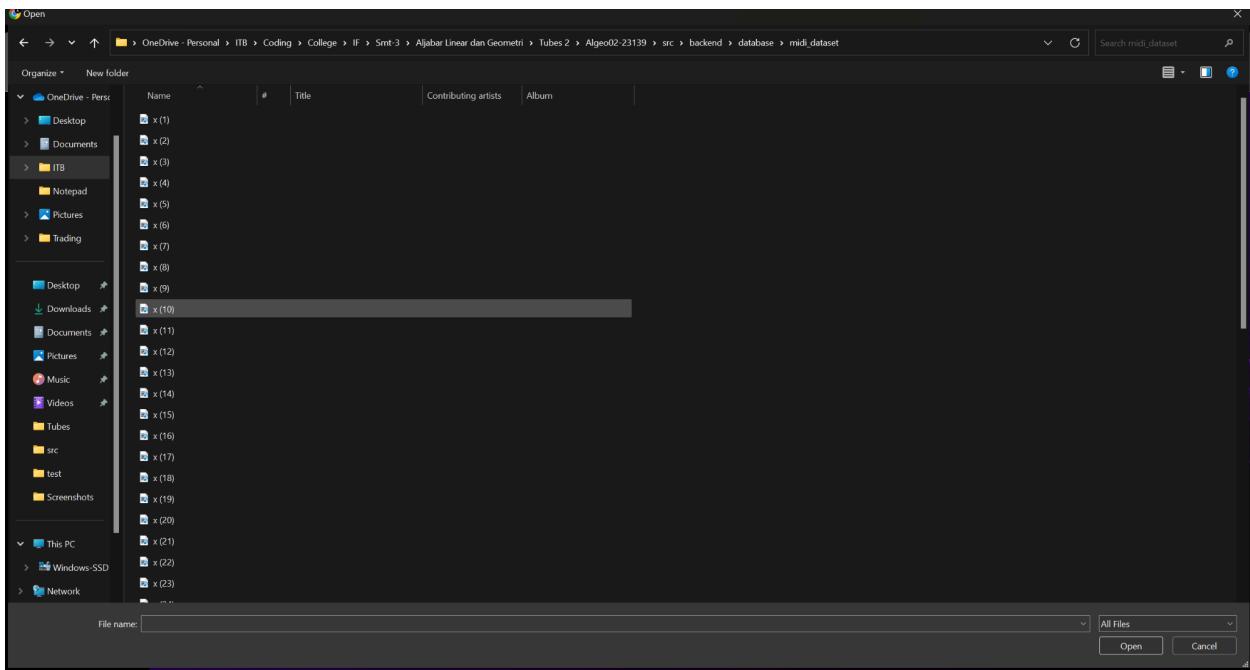


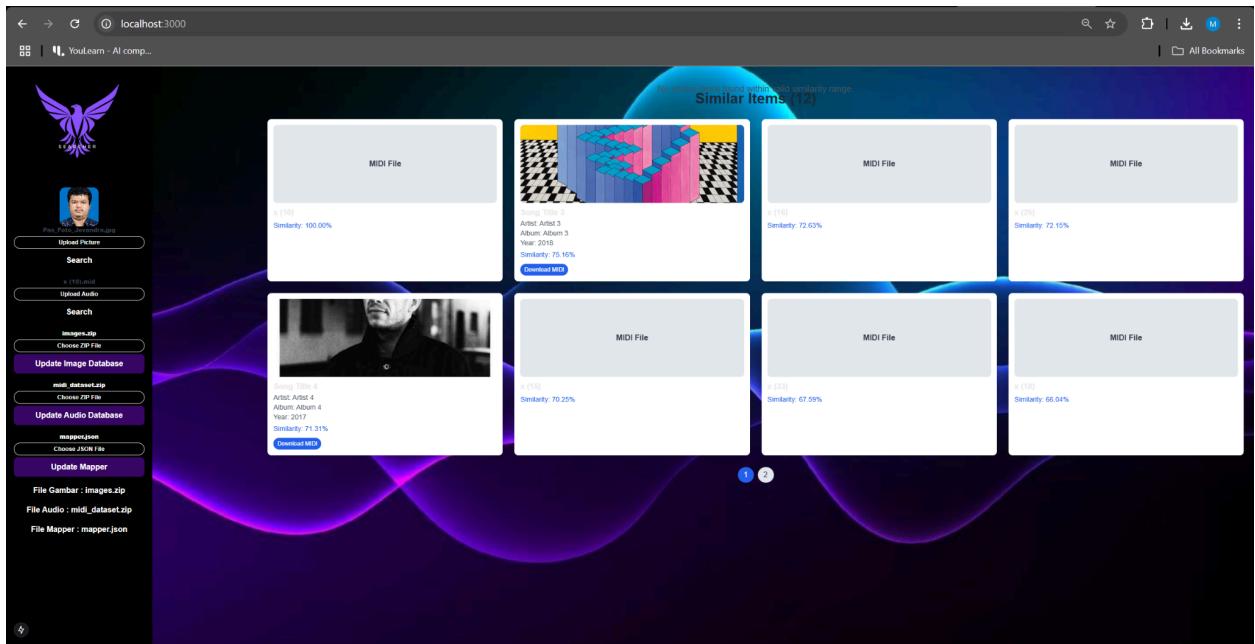
## 2. Testing Image Retrieval



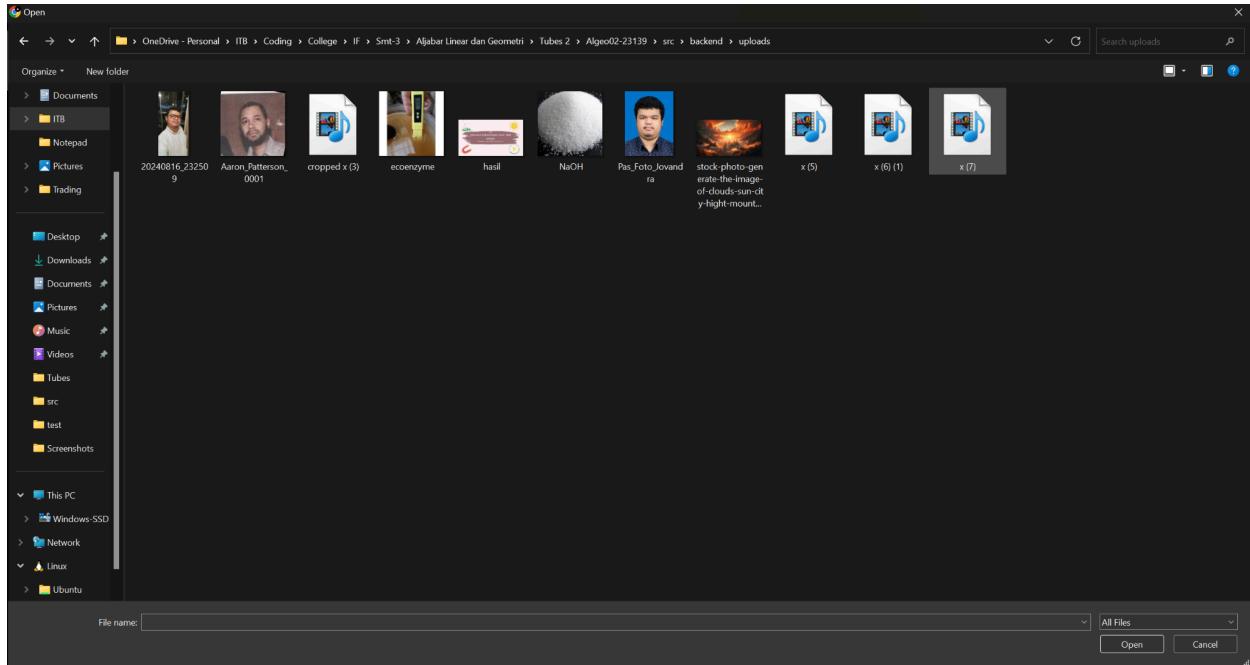


### 3. Testing music information retrieval

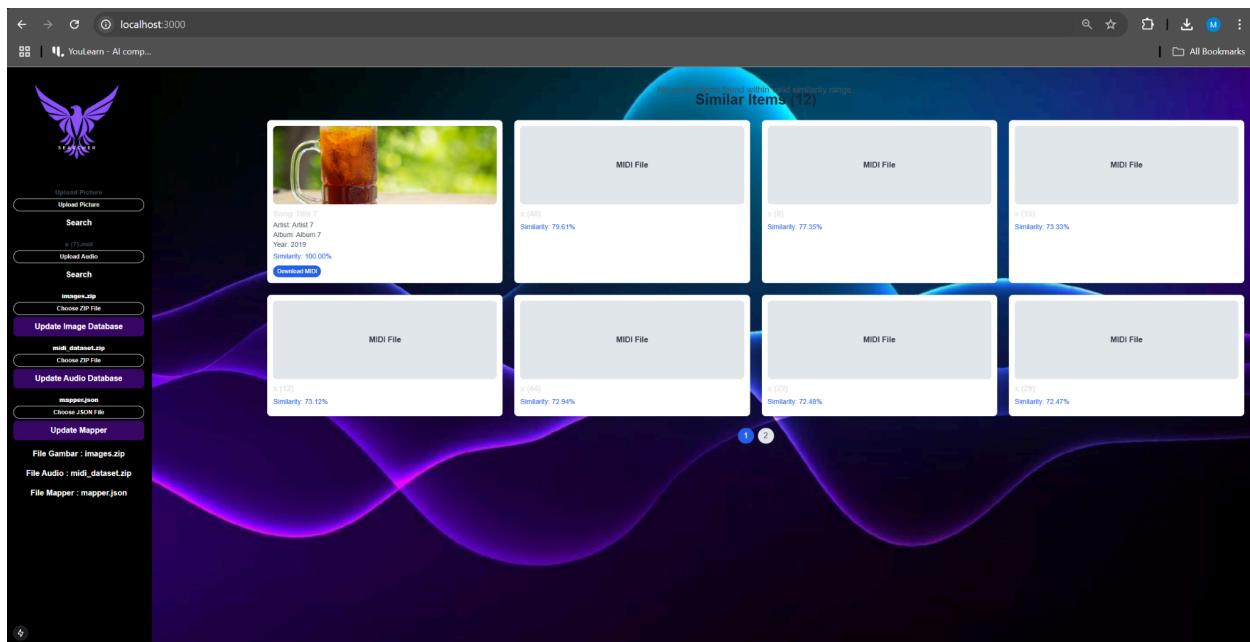




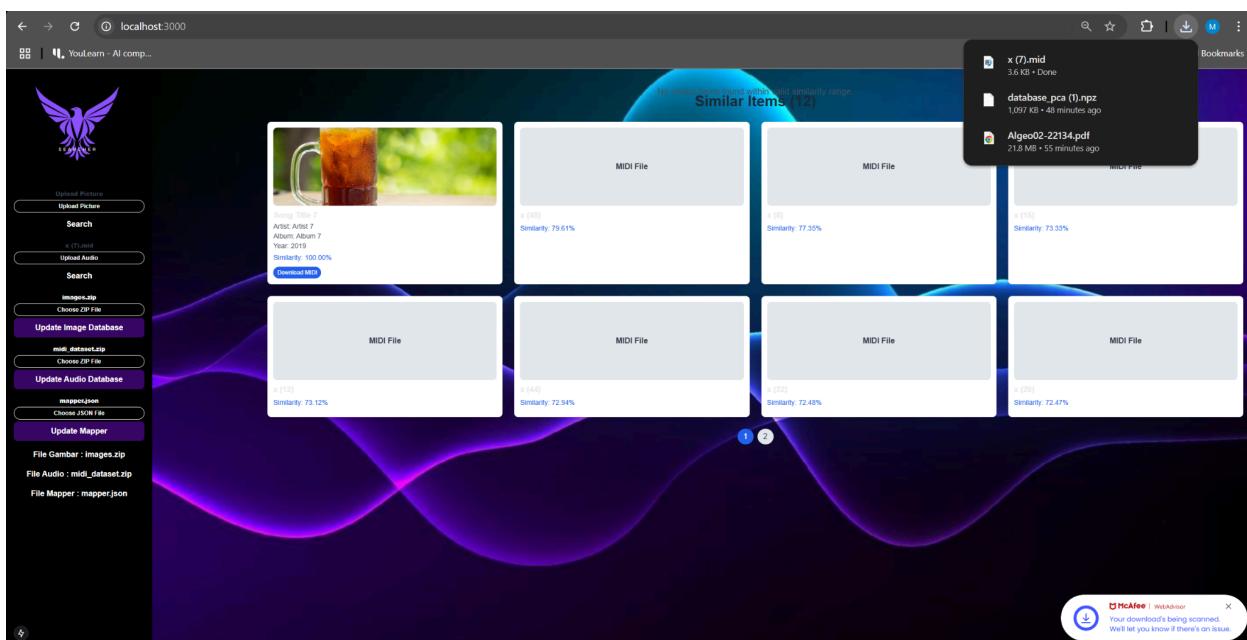
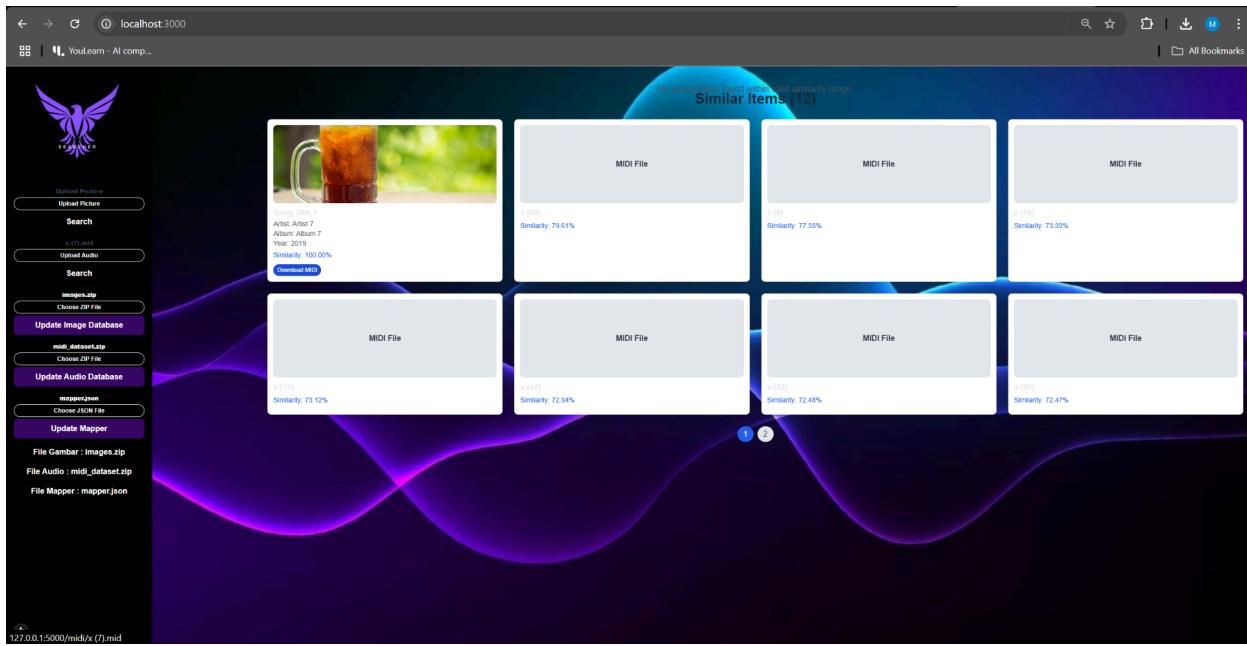
#### 4. Testing Download MIDI file dari search



A screenshot of a Windows File Explorer window. The path is OneDrive - Personal > ITB > Coding > College > IF > Smt-3 > Aljabar Linear dan Geometri > Tubes 2 > Algeo02-23139 > src > backend > uploads. The 'uploads' folder contains several files: 20240816\_232509, Aaron\_Patterson\_0001, cropped x (3), ecoenzyme, hasil, NaOH, Pas\_Foto\_Jowandra, stock-photo-gen-erale-the-image-of-clouds-sun-at-night-mount..., x (5), x (6) (1), and x (7). The left sidebar shows standard Windows navigation options like Desktop, Downloads, and This PC.

A screenshot of a web application at localhost:3000. The interface includes a sidebar with various upload and update buttons for images and audio. The main area displays a search result for "Amy Winehouse - Rehab" with a similarity score of 100.00%. Below this, there are eight cards labeled "Similar Items (12)", each showing a thumbnail, file type ("MIDI File"), file name (e.g., "x (1)", "x (2)", etc.), and similarity percentage (e.g., 79.61%, 77.35%, 73.33%, 73.12%, 72.94%, 72.48%, 72.47%). A purple decorative wave graphic is visible at the bottom of the page.



## **BAB 5: Kesimpulan**

### **I. Kesimpulan**

Pada tugas besar ini, kelompok kami berhasil mengimplementasikan Album Picture Finder dan Music Information Retrieval serta fungsionalitas keduanya pada sebuah website. Pada Album Picture Finder, kami berhasil mengimplementasikan seluruh proses mulai dari image processing, data centering, serta PCA computation. Pada Music Information Retrieval, kami berhasil mengimplementasikan pemrosesan audio dengan windowing yang ternormalisasi, ekstraksi fitur dari distribusi tone FTB, ATB, dan RTB yang dinormalisasikan, dan juga perhitungan similaritas menggunakan cosine similarity. Website yang kami buat juga berhasil dalam proses upload dataset MIDI, dataset images, mapper MIDI dan images, upload query MIDI, upload query image, dan juga unduh MIDI file.

### **II. Saran dan Komentar**

Pada tugas besar ini, kami memiliki beberapa saran dan komentar yang kami harapkan dapat membantu pembaca jika ingin menjalankan tugas yang serupa kedepannya:

- 1. Pahami bahasa pemrograman yang dipakai**

Dalam pengerjaan tugas besar ini, kami mengalami beberapa kendala yang dialami karena kurangnya menguasai bahasa pemrograman yang dipakai. Banyak waktu yang habis dipakai untuk memahami sintaks dan penggunaan bahasa pemrograman pada tengah pengerjaan. Menurut kami, proses pengerjaan akan berjalan lebih efisien apabila sebelum mulai mengerjakan proses implementasi, disediakan waktu untuk memahami bahasa yang dipakai terlebih dahulu.

- 2. Memiliki pengalaman integrasi frontend dan backend**

Pada tugas besar ini, kami memanfaatkan frontend Next JS dan backend flask. Proses integrasi yang kami lalui memakan waktu yang lama karena sumber daya manusia yang kami miliki masih minim pengalaman dalam menggabungkan bagian frontend dan backend website.

### 3. Memahami library yang digunakan

Pemahaman kami terkait library yang digunakan pada pengerjaan tugas besar ini belum maksimal. Sehingga, proses pengerjaan tidak berjalan seefisien mungkin karena sumber daya manusia yang kami miliki masih sering butuh waktu untuk memahami library yang digunakan saat pengerjaan.

## III. Refleksi

Pengerjaan tugas besar II mata kuliah Aljabar Linier dan Geometri kali ini sangat berkesan. Awalnya kami sangat bingung bagaimana cara mengimplementasikan spesifikasi yang diberi. Kami sebelumnya belum pernah menggunakan flask sebagai backend dengan Next JS sebagai frontend untuk sebuah website, dan kami juga belum pernah menggunakan bahasa python sebagai bahasa utama pada backend. Namun, seiring berjalannya waktu, kami belajar mencari library dan sumber belajar yang sesuai dengan kebutuhan software kami.

Tugas besar ini sangat membantu kami dalam mendapat bayangan untuk tugas-tugas yang akan kami dapatkan sebagai seorang software engineer, karena itu kami berterima kasih atas kesempatan belajar ini yang diberi kepada kami.