

TUGAS BESAR 2
ALJABAR LINIER DAN GEOMETRI IF2123
IMAGE RETRIEVAL DAN MUSIC INFORMATION
RETRIEVAL MENGGUNAKAN PCA DAN VEKTOR



Kelompok 17
Institut Gambling Bandung

Lutfi Hakim Yusra	13523084
David Bakti Lodianto	13523083
Nadhif Al Rozin	13523076

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
JL. GANESA 10, BANDUNG 40132

2024

DAFTAR ISI

DAFTAR ISI.....	1
BAB I	
DESKRIPSI MASALAH.....	2
BAB II	
TEORI SINGKAT.....	3
A. Image Retrieval.....	3
a. Principal Component Analysis.....	3
b. Database Preparation.....	3
c. Querying Image.....	5
B. Music Information Retrieval.....	5
a. Music Feature Vector and Processing.....	5
b. Querying Music.....	6
BAB III	
ARSITEKTUR PROGRAM.....	8
BAB IV	
EKSPERIMEN.....	11
BAB V	
KESIMPULAN DAN SARAN.....	13
LAMPIRAN.....	14
DAFTAR REFERENSI.....	15

BAB I

DESKRIPSI MASALAH

Information Retrieval adalah konsep meminta informasi dari sebuah data dengan memasukkan data tertentu. Pada tugas besar ini, anda akan berlutik dengan 2 jenis Information Retrieval. Image Retrieval dan Music Information Retrieval. Image Retrieval adalah konsep untuk memasukkan sebuah input gambar dan berharap mendapatkan gambar yang ada di data sesuai dengan informasi dan perhitungan yang diinginkan. Sedangkan Music Information Retrieval (MIR) adalah konsep untuk memasukkan sebuah input audio dan berharap mendapatkan audio yang ada di data sesuai dengan informasi dan perhitungan yang diinginkan. Pada tugas besar kali ini, kalian akan mengimplementasikan Image Retrieval dengan menggunakan Principal Component Analysis dan Music Information Retrieval dengan menggunakan humming.

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 diminta untuk 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, anda akan 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 tur-tur yang dimilikinya. Tidak hanya itu, Kami juga akan menggunakan konsep Principal Component Analysis (PCA) untuk mencari kumpulan audio melalui deteksi wajah berbagai orang (anggap saja mereka sebagai seorang penyanyi).

BAB II TEORI SINGKAT

A. Image Retrieval

a. Principal Component Analysis

Principal Component Analysis (PCA) adalah teknik statistik yang digunakan untuk mereduksi dimensi data dengan tetap mempertahankan sebanyak mungkin informasi yang ada. PCA mengubah data berdimensi tinggi menjadi beberapa dimensi yang lebih kecil, tanpa kehilangan pola utama dalam data tersebut, yang disebut *principal components*.

Principal components dihasilkan dari eigenspace (basis ruang eigen vektor) matriks kovarians dari sebuah dataset. Matriks kovarians merepresentasikan hubungan satu-ke-satu antara data pada dataset, dan eigenspace darinya akan merepresentasikan arah-arah yang mewakili nilai varians, yang terkandung di dalam nilai eigennya. Jika nilai eigen dan vektor eigen diurutkan dari yang terbesar, kita dapat mewakili sebuah data hanya dengan fitur-fiturnya yang paling penting. *Principal components* dapat didapat dari penguraian SVD (Singular Value Decomposition), dari nilai vektor-vektor singular kanan (v). Matriks v merepresentasikan vektor-vektor eigen dari ($A^T \times A$), yang merupakan rumus kovarians.

$$\begin{array}{c}
 \begin{array}{ccccccc}
 & d & & v & & v & d \\
 x \left[\begin{array}{|c|} \hline \\ \hline \end{array} \right] & = & x \left[\begin{array}{|c|} \hline \frac{x_1 v_1}{\sigma_1} \\ \hline \end{array} \right] & \bullet & v \left[\begin{array}{|c|} \hline \sigma_1 \\ \hline \end{array} \right] & \bullet & v \left[\begin{array}{|c|} \hline \\ \hline \end{array} \right] \\
 & & & & & & \\
 \mathbf{A} & = & \mathbf{U} & \bullet & \mathbf{\Sigma} & \bullet & \mathbf{V}^T \\
 \uparrow & & \uparrow & & \uparrow & & \uparrow \\
 \text{dataset} & & \text{"length" of samples in dataset on V} & & \text{magnitude of directions} & & \text{directions of greatest variance}
 \end{array}
 \end{array}$$

Pada *Image Retrieval*, sebuah foto diwakilkan dengan puluhan ribuan pixel yang membuat perbandingan antara foto sangat intensif jika dilakukan secara langsung. Dengan mendapatkan *principal components* sebuah dataset, sebuah data foto dapat direpresentasikan dengan komponen-komponen paling penting darinya terhadap dataset yang akan di-*query*.

b. Database Preparation

1. Image Processing and Loading

Semua foto ukurannya di-*resize* menjadi sebuah ukuran yang sama, lalu diubah grayscale, menggunakan rumus NTSC untuk mendapatkan tingkat intensitas dari setiap *pixel*. Setiap *pixel* direpresentasikan dalam sebuah array 1D berukuran $M \cdot N$ untuk sebuah gambar dengan ukuran $M \times N$. Semua foto dikompilasikan dalam bentuk sebuah matriks, yang merupakan kumpulan array foto yang telah diproses.



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

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

2. Data Centering

Kita memastikan dataset memiliki rata-rata kolom sama dengan nol, agar analisis yang dilakukan murni dari varians. Ini dapat dilakukan dengan menghitung rata-rata dari setiap gambar untuk suatu piksel di array $M \cdot N$, lalu dikurangi setiap elemennya sesuai rata-rata kolom.

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

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

3. PCA Computation

Setelah melakukan pemrosesan sebelumnya, kita memiliki matriks dataset X . Dengan melakukan penguraian SVD, kita akan mendapatkan *principal components* dari V . Setelah mendapatkan itu, kita memilih k -komponen teratas untuk hanya mengambil komponen-komponen yang paling penting. Dari itu, kita proyeksikan dataset X kita terhadap PC untuk mendapatkan dataset Z , yang merupakan kumpulan foto-foto dalam bentuk komponen-komponen yang terpenting (Vektor).

$$A = U \cdot \Sigma \cdot V^T$$

$$Z = X'U_k$$

c. Querying Image

1. Process Query

Dengan foto yang ingin dicari, kita melakukan Image Processing dan Data Centering yang sama dengan dataset, lalu proyeksikan terhadap PCA untuk mendapatkan vektor q .

2. Similarity Computation

Setelah mendapatkan q , kita membandingkan vektor tersebut dengan seluruh vektor yang terdapat di Z dengan menggunakan Euclidean Distance. Setelah mendapatkan jarak-jarak relatif dari setiap vektor di dataset terhadap vektor query, kita mengambil foto-foto yang memiliki jarak terdekat.

$$d(\mathbf{q}, \mathbf{z}_i) = \sqrt{\sum_{j=1}^k (q_j - z_{ij})^2}$$

B. Music Information Retrieval

a. Music Feature Vector and Processing

Sebuah file MIDI terdiri dari note-note pada *channel* (instrumen) yang berbeda-beda. Dengan asumsi channel 1 merupakan instrumen melodi, pengolahan hanya dilakukan pada channel tersebut. Sebuah note terdiri dari waktu mulai, akhir, velocity (tidak digunakan), dan pitch.

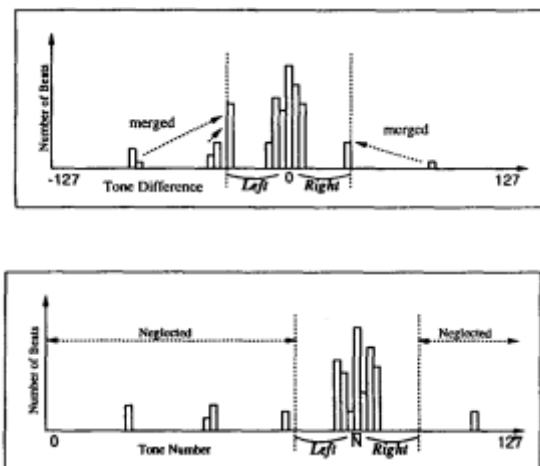
Sebuah musik dipotong-potong menjadi *window* dengan tumpuan 20 beat dengan lompatan 4 beat yang terdiri dari pitch. Kemudian, tiap *window* diproses untuk menghasilkan tiga histogram: Absolute Tone Based, yang merepresentasikan nilai kemunculan pitch pada window; Relative Tone Based, yang merepresentasikan nilai kemunculan lompatan antar pitch; First Tone

Based, yang merepresentasikan nilai kemunculan lompatan setiap pitch terhadap pitch pertama. Setiap histogram ini direpresentasikan sebagai sebuah vektor.

Dari setiap histogram, kita melakukan pembuatan *fuzzy histograms*, yang intinya menambah daerah sekitar bin yang memiliki value. Dengan menggunakan membership function berbasis Gauss.

$$\mu_{A_i}(x) = \exp\left(-\frac{(c_i - x)^2}{2\sigma_i^2}\right)$$

Agar histogram dapat direpresentasikan dengan adil, kita melakukan normalisasi dengan membagi semua bin dengan jumlah setiap bin. Setelah itu, dilakukan shrinking dengan mencari pitch rata-rata yang akan dijadikan inti, setelah itu mengambil bin dengan range (left-center, center+right). Untuk ATB, semua bin yang diluar range akan dijumlahkan dan diletakkan di bin paling kiri histogram yang telah dikecilkan, dan lakukan sama untuk di kanan. Untuk RTB dan FTB, tidak perlu diproses untuk bin di luar range. Ketiga histogramnya append satu-satu untuk menjadi feature vector. Semua feature vector ke dalam dataset.



b. Querying Music

Musik dipisah-pisah menjadi window seperti proses di atas, lalu membentuk feature vector musik, yang akan dibandingkan antara vektor-vektor di database dan vektor query. Perbandingan melakukan Cosine Similarity, yang merepresentasikan kemiripan antara dua vektor (0 paling beda, 1 sama persis). Vektor-vektor yang diatas 83% akan dimasukkan ke dalam list musik yang mendekati.

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

BAB III

ARSITEKTUR PROGRAM

A. Arsitektur Website (Front End)

1. **Next.js**

Framework React untuk SSR (Server-Side Rendering) dan SSG (Static Site Generation). Framework ini digunakan karena cepat dan mendukung routing dinamis serta integrasi API.

2. **React**

Library untuk membangun UI yang interaktif dan modular. React membuat arsitektur yang berbasis komponen, sehingga efisien dalam rendering untuk pengalaman pengguna yang responsif.

3. **Shadcn.ui**

Koleksi komponen UI berbasis Radix UI yang dapat disesuaikan. Shadcn menyediakan komponen-komponen siap pakai seperti tombol dan modal, mudah diubah sesuai kebutuhan desain.

Tech stack ini memastikan aplikasi pencarian gambar dan lagu memiliki antarmuka yang cepat, intuitif, dan modern.

a. Component

1. Button

Komponen ini untuk memberi user pilihan aksi untuk melakukan sesuatu

2. Navbar

Komponen ini bertujuan agar user dapat me-navigasi website dengan mudah

3. Upload

Komponen ini untuk mengunggah file gambar, audio, atau file mapping. Untuk file gambar dan audio, file ini digunakan langsung untuk query.

4. Song

Komponen ini digunakan untuk menampilkan *card* Song dengan keterangan sudah di-mapping sesuai dengan file mapping.

5. Humming

Komponen Humming digunakan untuk merekam *humming* user yang dapat digunakan sebagai query audio.

6. NotFound

Komponen ini untuk menampilkan tampilan jika belum ada dataset yang diunggah.

7. Pagination

Komponen ini digunakan untuk mengatur pagination pada preview dataset di homepage.

8. HomePage

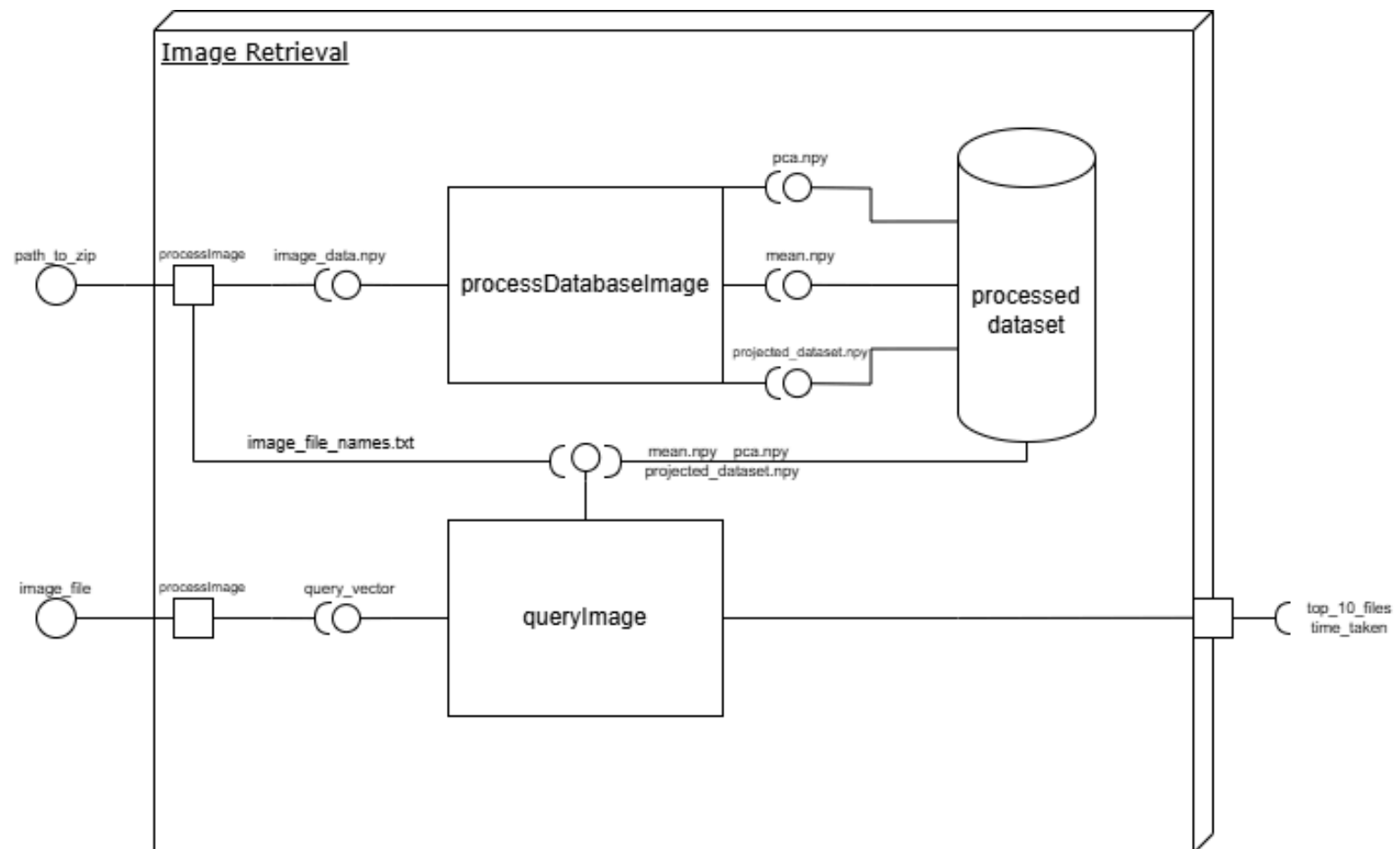
Komponen ini berisi tampilan untuk ditampilkan di homepage

9. DataPage

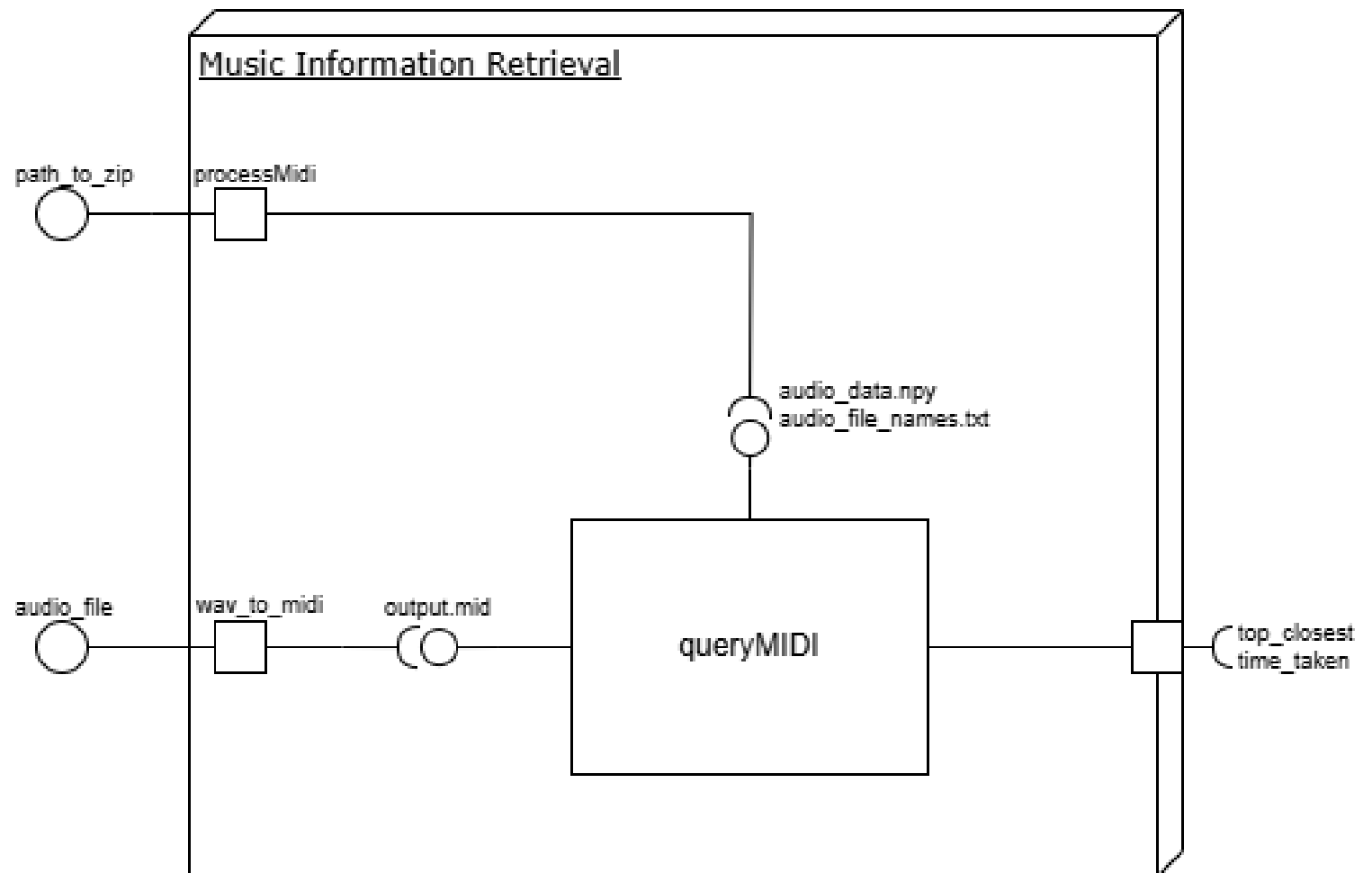
Komponen ini berisi tampilan untuk ditampilkan, yang berisi data hasil query ataupun data preview dataset.

B. Arsitektur Program Information Retrieval (Back End)

a. Image Retrieval

























b. Music Information Retrieval



BAB IV EKSPERIMEN

Dataset

 four	12/16/2024 8:47 PM	PNG File	289 KB
 fourteen	12/15/2024 2:26 PM	PNG File	1,091 KB
 funf.mid	12/16/2024 8:47 PM	VLC.mid	7 KB
 funfzehn.mid	12/16/2024 1:59 PM	VLC.mid	5 KB
 neun.mid	12/16/2024 8:47 PM	VLC.mid	9 KB
 nine	12/16/2024 8:47 PM	PNG File	276 KB
 one	12/16/2024 8:47 PM	PNG File	183 KB
 sechs.mid	12/16/2024 8:47 PM	VLC.mid	5 KB
 seven	12/16/2024 8:47 PM	PNG File	277 KB
 sieben.mid	12/16/2024 8:47 PM	VLC.mid	9 KB
 six	12/16/2024 8:47 PM	PNG File	255 KB
 ten	12/16/2024 8:47 PM	PNG File	270 KB
 thirteen	12/15/2024 2:26 PM	PNG File	351 KB
 three	12/16/2024 8:47 PM	PNG File	289 KB
 twelve	12/15/2024 2:26 PM	PNG File	327 KB
 two	12/16/2024 8:47 PM	PNG File	292 KB
 vier.mid	12/16/2024 8:47 PM	VLC.mid	55 KB
 vierzehn.mid	12/16/2024 1:59 PM	VLC.mid	4 KB
 zehn.mid	12/16/2024 8:47 PM	VLC.mid	11 KB
 zwei.mid	12/16/2024 8:47 PM	VLC.mid	4 KB
 zwolf.mid	12/16/2024 1:59 PM	VLC.mid	7 KB
 elven	12/16/2024 8:51 PM	WinRAR ZIP archive	5,068 KB

Query Image

The interface features a top navigation bar with a logo, a 'START' button, and a 'Data Set' dropdown. On the left, a sidebar contains buttons for 'Song', 'Image', and 'Mapping', along with a 'Preview' window showing a thumbnail and 'Save'/'Cancel' buttons. Below the sidebar, a 'Current Page: 1' indicator is present. A red error banner at the bottom left displays '1 error' and a 'Search' button. The main area is a grid of 5 thumbnails, each with a play button and a title:

- 1: Symphony No. 9
- 2: Symphony No. 7
- 3: Symphony No. 8
- 4: Symphony No. 6
- 5: Symphony No. 11 (with a pop-up showing 'Artist: mozart' and 'File Name: acht.mid')

Query Audio

The interface is similar to the Query Image section, with a top navigation bar, a sidebar with 'Song', 'Image', and 'Mapping' buttons, and a 'Preview' window. A red error banner at the bottom left displays '1 error'. The main area is a grid of 8 thumbnails, each with a play button and a title:

- 1: Symphony No. 14
- 2: Symphony No. 12
- 3: Symphony No. 15
- 4: Symphony No. 13
- 5: Symphony No. 9 (with a pop-up showing 'Artist: mozart' and 'File Name: dreizehn.mid')
- 6: Symphony No. 16
- 7: Eine kleine Nachtmusi
- 8: Symphony No. 11

BAB V

KESIMPULAN DAN SARAN

Program ini merupakan program yang dapat melakukan Image Retrieval dan Music Information Retrieval dengan proses yang meliputi Eigenvalue, Cosine Similarity, Euclidean Distance, dengan prinsip PCA dan analisis Feature Vector.

Setelah menyelesaikan tugas ini, kami menyadari bahwa sistem kerja terstruktur itu penting, agar pekerjaan yang dilakukan itu efektif dan efisien. Perencanaan yang matang juga penting dalam pengerjaan dan pembagian tugas, agar tidak ada pekerjaan yang sia-sia. Untuk tugas-tugas selanjutnya, sebaiknya diperbanyak pertemuan-pertemuan *online* maupun *offline* agar komunikasinya lancar antar anggota dan juga memantau progres tim. Selain itu, kita bisa membangun kerangka programnya dulu sebelum langsung terjun ke pengembangan aplikasi.

LAMPIRAN

Video:

https://drive.google.com/drive/folders/1B9ufHcRHHqCpXESrVNTM3uA_3msTffAg?usp=s_haring

Link Repository Github: <https://github.com/koinen/Algeo02-23076/tree/main>

DAFTAR REFERENSI

<https://docs.google.com/document/d/13qCaJJtmTyNFSrAvUT6BRVtRclOYFUHCVGsXvg3DU00/edit?tab=t.0>
<https://storrs.io/svd/>
<https://www.ijrte.org/wp-content/uploads/papers/v8i5/F2543037619.pdf>
<https://ieeexplore.ieee.org/document/799561/>
<https://pemrogramanmatlab.com/2023/08/10/pengenalan-wajah-menggunakan-metode-principal-component-analysis-pca/>
<https://asmp-eurasipjournals.springeropen.com/articles/10.1155/2010/179303>