

# **LAPORAN TUGAS BESAR**

## **IF2111 Algoritma dan Struktur Data STI**


### **WayangWave**

Dipersiapkan oleh:

Kelompok 01 - brot

18221136	Michael Jeremi Bungaran S
18222020	Muhammad Daffa Ikram
18222034	Christoper Daniel
18222094	Yovanka Sandrina Maharaja
18222100	Ervina Limka

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung  
Jl. Ganesha 10, Bandung 40132

	<b>Sekolah Teknik Elektro dan Informatika ITB</b>	<b>Nomor Dokumen</b>		<b>Halaman</b>
		<i>IF2111-TB-02-01</i>		<i>47</i>
		<i>Revisi</i>	<i>0</i>	<i>25 Oktober 2023</i>

# Daftar Isi

<b>1 Ringkasan</b>	<b>3</b>
2 Penjelasan Tambahan Spesifikasi Tugas	3
3 Struktur Data (ADT)	4
3.1 ADT Boolean	4
3.2 ADT Array Statik	4
3.3 ADT Array Dinamis	4
3.4 ADT Mesin Karakter	5
3.5 ADT Mesin Kata	5
3.6 ADT Queue	6
3.7 ADT Stack	6
3.8 ADT Set	7
3.9 ADT Map	7
3.10 ADT Linked List	8
4 Program Utama	8
5 Algoritma Menarik	9
6 Data Test	9
6.1 START	9
6.2 LOAD	10
6.3 LIST	10
6.4 PLAY	12
6.5 QUEUE	14
6.6 SONG	18
6.7 PLAYLIST	20
6.8 STATUS	23
6.9 SAVE	24
6.10 QUIT	24
6.11 HELP	25
6.12 INVALID COMMAND	26
7 Test Script	26
8 Pembagian Kerja dalam Kelompok	32
9 Lampiran	33
9.1 Deskripsi Tugas Besar	33
9.2 Notulen Rapat	45
9.3 Log Activity Anggota Kelompok	46

# 1 Ringkasan

Sebagai makhluk kasat mata yang setia kepada Bondowoso, kami memahami perjuangan yang dialaminya dalam mencari teman dan mendekati hati Roro. Meskipun Bondowoso memiliki fisik yang menakutkan, hatinya penuh dengan kebaikan dan keahlian musik yang luar biasa.

Kami, makhluk kasat mata yang selalu setia menemaninya, melihat bagaimana Bondowoso berusaha mencari cara untuk mendekati Roro yang tergila-gila pada musik hip-hop. Suatu hari, ketika kami sedang berdiskusi dengan Bondowoso, muncullah ide brilian dari salah satu teman kasat mata yang dulunya pendiri startup produksi roda batu.

Pemikiran untuk membuat aplikasi WayangWave adalah langkah inovatif yang kami yakini dapat meluluhkan hati Roro. Kami, sebagai makhluk kasat mata yang selalu mendukung Bondowoso, sepakat untuk membantu membangun aplikasi tersebut. Kami pun bekerja sama dengan Bondowoso dan lima teman kasat mata lainnya untuk mengembangkan WayangWave.

Dengan semangat dan tekad yang kuat, kami memanfaatkan kekuatan Anak Intern AI yang kami miliki untuk menciptakan aplikasi simulasi berbasis CLI. Kami memilih bahasa C sebagai alat utama untuk mewujudkan ide WayangWave ini, memanfaatkan struktur data yang telah kami pelajari selama mata kuliah IF2111 Algoritma dan Struktur Data STI. Struktur data yang kami gunakan meliputi ADT Array, ADT Mesin Karakter dan Mesin Kata, ADT Queue, ADT Stack, ADT Set & Map, ADT Linked List, serta *library stdio.h*, *stdlib.h*, *time.h*, dan *math.h*.

WayangWave bukan hanya sekadar aplikasi pemutar musik biasa, tetapi juga dilengkapi dengan fitur-fitur canggih seperti membuat dan menghapus playlist, mengatur urutan lagu, dan menampilkan status aplikasi. Semua ini kami rancang dengan detail, sesuai dengan keinginan Bondowoso untuk menunjukkan keindahan musik kepada Roro.

Dengan dedikasi dan kerja keras, akhirnya WayangWave berhasil dibuat dan siap digunakan. Kami yakin bahwa aplikasi ini akan menjadi senjata ampuh Bondowoso untuk mendekati hati Roro. Sebagai makhluk kasat mata, kami bersatu padu untuk memberikan yang terbaik demi kebahagiaan teman baik kami, Bondowoso.

Laporan Tugas Besar ini secara keseluruhan menyajikan gambaran singkat mengenai penggunaan aplikasi, penjelasan tambahan mengenai spesifikasi yang diterapkan, rincian struktur data yang digunakan, keterangan terkait program utama aplikasi WayangWave, serta melampirkan berbagai dokumen pendukung yang memperkuat kelengkapan tugas ini.

## 2 Penjelasan Tambahan Spesifikasi Tugas

Tidak ada penjelasan tambahan, spesifikasi tugas sudah lengkap.

## 3 Struktur Data (ADT)

### 3.1 ADT Boolean

- Sketsa struktur data :

```
#ifndef __BOOLEAN_H__
#define __BOOLEAN_H__

#define boolean unsigned char
#define true 1
#define false 0

#endif
```

- Persoalan yang diselesaikan : ADT Boolean digunakan dalam kondisi persoalan yang memerlukan nilai logika benar atau salah (true or false).
- Alasan pemilihan : ADT Boolean diperlukan dalam banyak sekali kondisi, terutama dalam implementasi ADT lainnya. Kita ambil contoh dalam ADT Array Statik, ada function boolean IsFull(Array T) yang akan melakukan return true saat Neff(T) == 0.
- Implementasi : ADT Boolean diimplementasikan sebagai boolean.h.

### 3.2 ADT Array Statik

- Sketsa struktur data :

```
#define IdxMax 100
#define IdxMin 1
#define IdxUndef -999 /* indeks tak terdefinisi*/

/* Definisi elemen dan koleksi objek */
typedef int IdxType;
typedef char *ElType;

typedef struct
{
    ElType TI[IdxMax - IdxMin + 1];
    int Neff;
} Array;
```

- Persoalan yang diselesaikan : ADT Array Statik digunakan untuk menyimpan daftar penyanyi yang terdapat dalam aplikasi WayangWave. ADT ini dapat digunakan untuk menampilkan nama penyanyi secara berurutan.
- Alasan pemilihan : ADT Array Statik diperlukan untuk penyimpanan daftar penyanyi yang ada dalam aplikasi.
- Implementasi : ADT Array Statik diimplementasikan sebagai ADT array.c, array.h, driverarray.c

### 3.3 ADT Array Dinamis

- Sketsa struktur data :

```

#define InitialSizeListPL 100

typedef char* namaPlaylist;
typedef int countsongPlaylist;
typedef int arraysize;

typedef struct {
    namaPlaylist namePL;
    countsongPlaylist countsongPL;
} infotypePlaylist;

typedef struct {
    infotypePlaylist* playlist;
    arraysize count;
    arraysize capacity;
} ListPlaylist;

```

- Persoalan yang diselesaikan : ADT Array Dinamis digunakan untuk menyimpan daftar playlist yang terdapat dalam aplikasi WayangWave. ADT ini dapat digunakan untuk menampilkan playlist yang ada secara berurutan.
- Alasan pemilihan : ADT Array Dinamis diperlukan untuk penyimpanan daftar playlist yang ada dalam aplikasi.
- Implementasi : ADT Array Dinamis diimplementasikan sebagai ADT listplaylist.c, listplaylist.h, driverlistplaylist.c

### 3.4 ADT Mesin Karakter

- Sketsa struktur data :

```

#define MARK '.'
/* State Mesin */
extern char currentChar;
extern boolean EOP;

```

- Persoalan yang diselesaikan : ADT Mesin Karakter dapat digunakan untuk mengakses tipe data char dari sebuah input user ataupun saat mengakses file di ADT Mesin Kata nanti.
- Alasan pemilihan : ADT ini kami gunakan karena dibutuhkan fungsi START dan ADV untuk mengakses char dari input atau file.
- Implementasi : ADT Mesin Karakter diimplementasikan sebagai ADT mesinkarakter.c, mesinkarakter.h, drivermesin.c

### 3.5 ADT Mesin Kata

- Sketsa struktur data :

```

#define NMax 100
#define BLANK '\n'

typedef struct {
    char TabWord[NMax]; /* indeks yang dipakai [0..NMax-1] */
    int length;
} Word;

/* State Mesin Kata */
extern boolean EndWord;
extern Word currentWord;

```

- Persoalan yang diselesaikan : ADT Mesin Kata digunakan untuk command program dan juga pembacaan file ke dalam program. ADT ini juga digunakan untuk mengkonversi beberapa tipe data. Misal dari word to integer ataupun word to string.

- Alasan pemilihan : Di dalam program, banyak yang membutuhkan fungsi untuk membaca file konfigurasi dan membaca input/masukan dari user tanpa menggunakan scanf dan fgets. Hasil inputnya berupa word sehingga perlu dikonversi menjadi tipe data tertentu agar bisa digunakan dalam fungsi.
- Implementasi : ADT Mesin Kata diimplementasikan sebagai ADT mesinkata.h, mesinkata.c, word.h, word.c, dan drivermesin.c

### 3.6 ADT Queue

- Sketsa struktur data :

```
typedef struct
{
    int idPlaylist;
    char *artist;
    char *album;
    char *song;
} infotypeQueue;

typedef infotypeQueue ElTypeQueue;
typedef int IdxType;

typedef struct
{
    ElTypeQueue Tab[IDX_MAX + 1];
    IdxType idxHead;
    IdxType idxTail;
} Queue;
```

- Persoalan yang diselesaikan : ADT Queue digunakan untuk menyimpan urutan lagu yang akan diputar dalam aplikasi. ADT ini digunakan untuk menampilkan status lagu yang sedang diputar, lagu selanjutnya yang akan diputar, dan semua lagu yang berada di dalam antrian.
- Alasan pemilihan : ADT Queue diperlukan untuk mengurutkan lagu berdasarkan urutan suatu lagu ditambahkan ke dalam antrian. ADT ini dipilih karena untuk lagu yang akan ditambahkan akan langsung ditambah setelah urutan akhir (Tail) dan lagu yang akan diputar diambil dari urutan awal (Head). Hal ini sesuai dengan susunan elemen Queue yaitu FIFO (First In First Out). Penyisipan lagu selalu dilakukan setelah elemen terakhir, pemutaran lagu selalu dilakukan pada elemen pertama.
- Implementasi : ADT Queue diimplementasikan sebagai ADT circular\_queue.c, circular\_queue.h, dan driverqueue.c

### 3.7 ADT Stack

- Sketsa struktur data :

```
#define NilStackchar -1
#define MaxElStackchar 100

typedef struct
{
    char* artist;
    char* album;
    char* song;
} infotypeStackchar;

typedef infotypeStackchar ElTypeStackchar;
typedef int address;

typedef struct
{
    ElTypeStackchar T[MaxElStackchar];
    address TOPSTACKCHAR;
} Stackchar;
```

- Persoalan yang diselesaikan : ADT Stack digunakan untuk menyimpan urutan lagu yang telah diputar dalam aplikasi. ADT ini digunakan untuk menampilkan lagu sebelumnya yang telah diputar dan mengetahui isi dari riwayat lagu (kosong atau tidak).
- Alasan pemilihan : ADT Stack diperlukan untuk menampung riwayat lagu yang telah diputar. ADT ini dipilih karena setelah sebuah lagu diputar akan otomatis disimpan. Cara penyimpanan ini menggunakan stack karena diperlukan pemutaran lagu sebelumnya yang dapat diakses dengan mengambil lagu yang terakhir kali ditambahkan. Susunan pengambilan lagu adalah LIFO (Last In First Out). Penyisipan lagu dilakukan di atas TOP (lagu sebelumnya), pemutaran lagu sebelumnya dilakukan pada TOP.
- Implementasi : ADT Array Statik diimplementasikan sebagai ADT stackchar.c, stackchar.h, dan driverstackchar.c

### 3.8 ADT Set

- Sketsa struktur data :

```
#define Nil 0
#define MaxEl 100

typedef char *infotype;
typedef int address;

typedef struct
{
    infotype Elements[MaxEl];
    address Count;
} Set;
```

- Persoalan yang diselesaikan : ADT Set digunakan untuk menyimpan lagu dari sebuah album dimana Set merupakan valuetype dari Map Album.
- Alasan pemilihan : Dengan menggunakan ADT Set, kita bisa memastikan bahwa dalam sebuah album tidak ada lagu yang terduplikat/*double*.
- Implementasi : ADT Set diimplementasikan sebagai ADT set.h, set.c, driverset.c, serta digunakan pula dalam struct ADT Map.

### 3.9 ADT Map

- Sketsa struktur data :

```
#define NilAlbum 0
#define MaxElAlbum 100

// typedef int bool;
typedef char *keytype;
typedef Set valuetype;
typedef int addressAlbum;

typedef struct
{
    keytype Key;
    valuetype Value;
} infotypeAlbum;

typedef struct
{
    infotypeAlbum Elements[MaxElAlbum];
    addressAlbum Count;
} Album;
```

- Persoalan yang diselesaikan : ADT Map digunakan dalam pembuatan ADT Album dimana keytype nya adalah nama dari album itu sendiri sedangkan valuetype nya adalah set yang terdiri dari lagu dalam album.

- Alasan pemilihan : Dengan menggunakan ADT Map, kita bisa mengakses lagu berdasarkan nama dari album tempat lagu itu berada dengan menggunakan fungsi, misalnya ValueInAlbum.
- Implementasi : ADT Map diimplementasikan sebagai ADT album.h, album.c, driveralbum.c, penyanyi.h, penyanyi.c, dan driverpenyanyi.c.

### 3.10 ADT Linked List

- Sketsa struktur data :

```
#define MAX_STRING_LENGTH 100
#define MAX_SONGS 100

typedef struct {
    char* lagu;
    char* artist;
    char* album;
} Song;

typedef struct SongNode {
    Song data;
    Struct SongNode* next;
} SongNode;

typedef struct {
    SongNode* head;
    int numSongs;
} Playlist;

#define MAX_PLAYLISTS 100

typedef struct {
    Playlist playlists[MAX_PLAYLISTS];
    int numPlaylists;
} PlaylistManager;
```

- Persoalan yang diselesaikan : ADT Linked List digunakan untuk menyimpan playlist yang berisi kumpulan lagu yang kontigu. ADT ini dapat menyelesaikan proses pembuatan playlist baru (Create Playlist), penambahan lagu dan album ke dalam playlist (Playlist add song/album), penggantian urutan lagu di dalam playlist (Playlist swap), penghapusan lagu di dalam playlist (Playlist remove), dan penghapusan playlist (Playlist delete).
- Alasan pemilihan : ADT Linked List dipilih untuk menyimpan playlist yang berisi kumpulan lagu untuk memudahkan penambahan, penghapusan, dan pengubahan pada playlist.
- Implementasi : ADT Linked List diimplementasikan sebagai ADT playlist.c, playlist.h, dan driverplaylist.c.

## 4 Program Utama

Program utama dimulai dari file main.c yang berfungsi sebagai driver program lain dengan memasukkan header library yang boleh digunakan yaitu stdio.h, stdlib.h, serta header dari fungsi/prosedur yang telah dibuat sebelumnya, yaitu console.h dan load.h. Program utama akan menampilkan *string* selamat datang dan Main Menu WayangWave yang berisi pilihan *command*, yaitu START, LOAD, dan HELP. Pengguna dapat memilih untuk memulai sesi baru dengan *command* START, atau memilih untuk melanjutkan sesi sebelumnya dengan *command* LOAD.



Jika pengguna memasukkan *command* LOAD maka harus diikuti nama file yang kemudian dapat dibaca oleh program. Jika pengguna memasukkan *command* START, pengguna akan bisa mengakses *command* yang terdapat di WayangWave yaitu LIST DEFAULT, LIST PLAYLIST, PLAY SONG, PLAY PLAYLIST, QUEUE SONG, QUEUE PLAYLIST, QUEUE SWAP, QUEUE REMOVE, QUEUE CLEAR, SONG NEXT, SONG PREVIOUS, PLAYLIST CREATE, PLAYLIST ADD SONG, PLAYLIST ADD ALBUM, PLAYLIST SWAP, PLAYLIST REMOVE, PLAYLIST DELETE, STATUS, SAVE, QUIT, dan HELP. Semua *command* yang valid akan otomatis menjalankan fungsi masing-masing.

## 5 Algoritma Menarik

Algoritma yang menarik pada program WayangWave adalah function insertHead pada circular queue. Function insertHead menarik karena penyisipan elemen dilakukan pada HEAD queue, berbeda dengan enqueue yang penyisipan elemen dilakukan setelah TAIL queue. Function ini digunakan untuk memasukkan kembali current song ke HEAD queue ketika dijalankan *command* SONG PREVIOUS.

## 6 Data Test

Dalam program WayangWave, ada beberapa fitur yang akan dites, yaitu perintah atau *command* yang akan digunakan saat pengguna aplikasi, antara lain :

### 6.1 START

Program akan membaca file konfigurasi default yang berisi daftar penyanyi dan album yang dimiliki. Jika berhasil, program akan menuliskan pesan “File konfigurasi aplikasi berhasil dibaca. WayangWave berhasil dijalankan.”.

→ *Output* apabila *input* valid :



Gambar 6.1.1

→ *Output* apabila *input* tidak valid :

```

                                w E L c o m e   t o

W A Y A N G   N I N G

===== [ Main Menu WayangWave ] =====
1. START
2. LOAD <namafile>
3. HELP
ENTER COMMAND: LALALA
Command tidak bisa dieksekusi. Mohon masukkan command yang sesuai.
ENTER COMMAND: █

```

Gambar 6.1.2

## 6.2 LOAD

Program akan membaca file yang sudah disimpan <filename> yang berisi list penyanyi, album, dan lagu yang bisa diputar.

→ *Output* apabila *input* valid dan file terdapat di folder data :

```

                                w E L c o m e   t o

W A Y A N G   N I N G

===== [ Main Menu WayangWave ] =====
1. START
2. LOAD <namafile>
3. HELP
ENTER COMMAND: LOAD hivi.txt
Save file berhasil dibaca. WayangWave berhasil dijalankan.

```

Gambar 6.2.1

→ *Output* apabila *input* file tidak dapat ditemukan:

```

                                w E L c o m e   t o

W A Y A N G   N I N G

===== [ Main Menu WayangWave ] =====
1. START
2. LOAD <namafile>
3. HELP
ENTER COMMAND: LOAD hahahoho.txt
Save file tidak ditemukan. WayangWave tidak berhasil dijalankan.
ENTER COMMAND: █

```

Gambar 6.2.2

## 6.3 LIST

### 6.3.1 LIST DEFAULT

Program akan menampilkan daftar list penyanyi yang ada, melihat album dari penyanyi yang dipilih, dan lagu dari album yang dipilih. Untuk melihat album ataupun lagu diperlukan adanya persetujuan.

→ *Output apabila input valid :*

```
ENTER COMMAND: LIST DEFAULT
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Ingin melihat album yang ada?(Y/N) : █
```

Gambar 6.3.1.1

◆ *Output lanjutan apabila input selanjutnya Y :*

```
ENTER COMMAND: LIST DEFAULT
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Ingin melihat album yang ada?(Y/N) : Y
Pilih penyanyi untuk melihat album mereka: █
```

Gambar 6.3.1.2

→ *Output apabila input valid :*

```
Pilih penyanyi untuk melihat album mereka: BLACKPINK
Daftar Album oleh BLACKPINK :
1. SQUARE ONE
2. SQUARE TWO
3. SQUARE UP
4. BORN PINK
5. THE ALBUM
Ingin melihat lagu yang ada?(Y/N) : █
```

Gambar 6.3.1.3

◆ *Output apabila input valid :*

```
Ingin melihat lagu yang ada?(Y/N) : Y
Pilih album untuk melihat lagu yang ada di album : THE ALBUM
Daftar Lagu di THE ALBUM:
1. How You Like That
2. Ice Cream (with Selena Gomez)
3. Pretty Savage
4. Bet You Wanna (Feat. Cardi B)
5. Lovesick Girls
6. Crazy Over You
7. Love To Hate Me
8. You Never Know
```

Gambar 6.3.1.4

◆ *Output apabila input tidak valid :*

```
Daftar Album oleh BLACKPINK :
1. SQUARE ONE
2. SQUARE TWO
3. SQUARE UP
4. BORN PINK
5. THE ALBUM
Ingin melihat lagu yang ada?(Y/N) : Y
Pilih album untuk melihat lagu yang ada di album : SQUARE THREE
Album SQUARE THREE tidak ada dalam daftar.
```

Gambar 6.3.1.5

→ *Output apabila input tidak valid :*

```
ENTER COMMAND: LIST DEFAULT
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Ingin melihat album yang ada?(Y/N) : Y
Pilih penyanyi untuk melihat album mereka: Dewa 19
Penyanyi Dewa 19 tidak ada dalam daftar.
```

Gambar 6.3.1.6

◆ *Output* lanjutan apabila *input* selanjutnya N :

```
ENTER COMMAND: LIST DEFAULT
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Ingin melihat album yang ada?(Y/N) : N
```

```
===== [ Main Menu WayangWave ] =====
1. LIST DEFAULT
2. LIST PLAYLIST
3. PLAY SONG
4. PLAY PLAYLIST
5. QUEUE SONG
6. QUEUE PLAYLIST
7. QUEUE SWAP <x> <y>
8. QUEUE REMOVE <id>
9. QUEUE CLEAR
10. SONG NEXT
11. SONG PREVIOUS
12. PLAYLIST CREATE
13. PLAYLIST ADD SONG
14. PLAYLIST ADD ALBUM
15. PLAYLIST SWAP <id> <x> <y>
16. PLAYLIST REMOVE <id> <n>
17. STATUS
18. SAVE <filename>
19. QUIT
20. HELP
```

Gambar 6.3.1.7

→ *Output* apabila *input* tidak valid :

*Input* selain Y dianggap tidak valid dan dieksekusi dengan cara yang sama seperti mengeksekusi N.

```
ENTER COMMAND: LIST DEFAULT
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Ingin melihat album yang ada?(Y/N) : HAH
```

Gambar 6.3.1.8

## 6.3.2 LIST PLAYLIST

Program akan menampilkan daftar list playlist pengguna.

→ *Output* apabila pengguna tidak memiliki playlist :

```
ENTER COMMAND: LIST PLAYLIST
Kamu tidak memiliki playlist.
```

Gambar 6.3.2.1

→ *Output* apabila pengguna memiliki playlist :

```
ENTER COMMAND: LIST PLAYLIST
Daftar Playlist yang kamu miliki:
1. Playlist #1
2. Ubuntu (Usus Buntu)
```

Gambar 6.3.2.2

## 6.4 PLAY

### 6.4.1 PLAY SONG

Program Play Song akan memutar lagu sesuai dengan penyanyi, album, dan id lagu yang dipilih. Setelah lagu diputar, antrian dan riwayat lagu menjadi kosong.

→ *Output* apabila *input* valid dan terdapat antrian sebelum *command* Play Song :

*Sebelum :*

```

ENTER COMMAND: STATUS
Now Playing:
NewJeans - Hurt - New Jeans

Queue:
1. NewJeans - Cookie - New Jeans
2. NewJeans - Hype Boy - New Jeans
3. NewJeans - Attention - New Jeans

```

Gambar 6.4.1.1

### ***Play Song :***

→ *Output* apabila input valid :

```

ENTER COMMAND: PLAY SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : Hivi!
Daftar Album oleh Hivi! :
1. CERITERA
2. Kereta Kencan
3. Say Hi To Hivi!
Masukkan Nama Album yang dipilih : CERITERA
Daftar Lagu Album CERITERA oleh Hivi! :
1. Jatuh, Bangkit Kembali!
2. Satu-Satunya
3. Bahagia
4. Patung Batu
5. Musim Hujan
6. Tersenyum, Untuk Siapa?
7. Bumi Dan Bulan
8. Pemuda
9. Teman Sejati
Masukkan ID Lagu yang dipilih : 2
Memutar lagu Satu-Satunya oleh Hivi!

```

Gambar 6.4.1.2

### ***Sesudah :***

```

ENTER COMMAND: STATUS
Now Playing:
Hivi! - Satu-Satunya - CERITERA

Queue:
Your queue is empty.

```

Gambar 6.4.1.3

→ *Output* apabila *input* tidak valid :

```

ENTER COMMAND: PLAY SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : NewJeans
Daftar Album oleh NewJeans :
1. New Jeans
2. OMG
3. Get Up
Masukkan Nama Album yang dipilih : Get Up
Daftar Lagu Album Get Up oleh NewJeans :
1. New Jeans
2. Super Shy
3. ETA
4. Cool With You
5. Get Up
6. ASAP
Masukkan ID Lagu yang dipilih : 7
Tidak ada lagu dengan ID 7

```

Gambar 6.4.1.4

## **6.4.2 PLAY PLAYLIST**

***Sebelum :***

Gambar 6.4.2.1

Gambar 6.4.2.2

→ *Output* apabila *input* ID Playlist valid :

Gambar 6.4.2.3

Gambar 6.4.2.4

Gambar 6.4.2.5

***Sebelum :***

Gambar 6.5.1.1

STEI- ITB	IF2111-TB-02-01	Halaman 14 dari 48 halaman
<p>Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB</p>		

→ *Output* apabila seluruh *input* valid :

```
ENTER COMMAND: QUEUE SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi: Hivi!
Daftar Album oleh Hivi! :
1. CERITERA
2. Kereta Kencan
3. Say Hi To Hivi!
Masukkan Nama Album yang dipilih : CERITERA
Daftar Lagu Album CERITERA oleh Hivi! :
1. Jatuh, Bangkit Kembali!
2. Satu-Satunya
3. Bahagia
4. Patung Batu
5. Musim Hujan
6. Tersenyum, Untuk Siapa?
7. Bumi Dan Bulan
8. Pemuda
9. Teman Sejati
Masukkan ID Lagu yang dipilih: 2
Berhasil menambahkan lagu Satu-Satunya oleh Hivi! ke queue.
```

Gambar 6.5.1.2

→ *Output* apabila *input* tidak valid :

```
ENTER COMMAND: QUEUE SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi: Wedhus
Penyanyi Wedhus tidak ditemukan!
```

Gambar 6.5.1.3

```
ENTER COMMAND: QUEUE SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi: Arctic Monkeys
Daftar Album oleh Arctic Monkeys :
1. Favourite Worst Nightmare
2. Humbug
3. AM
Masukkan Nama Album yang dipilih : TIKTOK HOTS
Album penyanyi Arctic Monkeys dengan judul TIKTOK HOTS tidak ditemukan!
```

Gambar 6.5.1.4

```
ENTER COMMAND: QUEUE SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi: Arctic Monkeys
Daftar Album oleh Arctic Monkeys :
1. Favourite Worst Nightmare
2. Humbug
3. AM
Masukkan Nama Album yang dipilih : AM
Daftar Lagu Album AM oleh Arctic Monkeys :
1. Do I Wanna Know?
2. R U Mine?
3. One For The Road
4. Arabella
5. I Want It All
6. No. 1 Party Anthem
7. Mad Sounds
8. Fireside
9. Why'd You Only Call Me When You're High?
10. Snap Out Of It
11. Knee Socks
12. I Wanna Be Yours
Masukkan ID Lagu yang dipilih: 13
Tidak ada lagu dengan ID 13
```

Gambar 6.5.1.5

**Sesudah :**

```
ENTER COMMAND: STATUS
Now Playing:
No songs have been played yet. Please search for a song to begin playback.
Queue:
1. Hivi! - Satu-Satunya - CERITERA
```

Gambar 6.5.1.6

## 6.5.2 QUEUE PLAYLIST

Program dilakukan untuk menambahkan sebuah playlist ke dalam queue. Playlist dipilih berdasarkan input id playlist yang ingin dimasukkan ke queue.

**Sebelum :**

```
ENTER COMMAND: STATUS
Now Playing:
No songs have been played yet. Please search for a song to begin playback.
Queue:
Your queue is empty.
```

Gambar 6.5.2.1

**Queue Playlist :**

→ **Output** apabila **input** ID Playlist valid :

```
ENTER COMMAND: PLAY PLAYLIST
Masukkan ID Playlist: 1
Memutar playlist K-Pop ON!
```

Gambar 6.5.2.2

→ **Output** apabila **input** ID Playlist tidak valid :

```
ENTER COMMAND: PLAY PLAYLIST
Masukkan ID Playlist: 3
Tidak terdapat playlist dengan ID 3
```

Gambar 6.5.2.3

**Sesudah :**

```
ENTER COMMAND: STATUS
Current Playlist: K-Pop ON!

Now Playing:
BLACKPINK - PLAYING WITH FIRE - SQUARE TWO

Queue:
1. BLACKPINK - STAY - SQUARE TWO
2. NewJeans - OMG - OMG
3. NewJeans - Ditto - OMG
```

Gambar 6.5.2.4

## 6.5.3 QUEUE SWAP <x> <y>

Program dilakukan untuk menukar lagu pada urutan ke x dengan lagu yang berada di urutan ke y.

**Sebelum :**

```
ENTER COMMAND: STATUS
Current Playlist: K-Pop ON!

Now Playing:
BLACKPINK - PLAYING WITH FIRE - SQUARE TWO

Queue:
1. BLACKPINK - STAY - SQUARE TWO
2. NewJeans - OMG - OMG
3. NewJeans - Ditto - OMG
```

Gambar 6.5.3.1

**Queue Swap :**

→ **Output** apabila **input** valid :

```
ENTER COMMAND: QUEUE SWAP 1 2
Lagu STAY berhasil ditukar dengan OMG.
```

Gambar 6.5.3.2



→ *Output* apabila *input* tidak valid :

```
ENTER COMMAND: QUEUE SWAP 1 5
Lagu dengan urutan ke 5 tidak terdapat dalam queue!
```

Gambar 6.5.3.3

**Sesudah :**

```
ENTER COMMAND: STATUS
Current Playlist: K-Pop ON!

Now Playing:
BLACKPINK - PLAYING WITH FIRE - SQUARE TWO

Queue:
1. NewJeans - OMG - OMG
2. BLACKPINK - STAY - SQUARE TWO
3. NewJeans - Ditto - OMG
```

Gambar 6.5.3.4

#### 6.5.4 QUEUE REMOVE <id>

Program dilakukan untuk menghapus sebuah lagu dari queue berdasarkan input id yang merupakan urutan lagu dalam queue.

**Sebelum :**

```
ENTER COMMAND: STATUS
Current Playlist: K-Pop ON!

Now Playing:
BLACKPINK - PLAYING WITH FIRE - SQUARE TWO

Queue:
1. BLACKPINK - STAY - SQUARE TWO
2. NewJeans - OMG - OMG
3. NewJeans - Ditto - OMG
```

Gambar 6.5.4.1

**Queue Remove :**

→ *Output* apabila *input* valid :

```
ENTER COMMAND: QUEUE REMOVE 3
Lagu Ditto oleh NewJeans telah dihapus dari queue!
```

Gambar 6.5.4.2

→ *Output* apabila *input* tidak valid :

```
ENTER COMMAND: QUEUE REMOVE 15
Lagu dengan urutan ke 15 tidak ada.
```

Gambar 6.5.4.3

**Sesudah :**

```
ENTER COMMAND: STATUS
Current Playlist: K-Pop ON!

Now Playing:
BLACKPINK - PLAYING WITH FIRE - SQUARE TWO

Queue:
1. BLACKPINK - STAY - SQUARE TWO
2. NewJeans - OMG - OMG
```

Gambar 6.5.4.4

#### 6.5.5 QUEUE CLEAR

Program dilakukan untuk menghapus/mengosongkan semua lagu yang ada di dalam queue.

**Sebelum :**

```
ENTER COMMAND: STATUS
Now Playing:
No songs have been played yet. Please search for a song to begin playback.
Queue:
1. Hivi! - Satu-Satunya - CERITERA
```

Gambar 6.5.5.1

**Queue Clear :**

```
ENTER COMMAND: QUEUE CLEAR
Queue berhasil dikosongkan!
```

Gambar 6.5.5.2

**Sesudah :**

```
ENTER COMMAND: STATUS
Now Playing:
No songs have been played yet. Please search for a song to begin playback.
Queue:
Your queue is empty.
```

Gambar 6.5.5.3

## 6.6 SONG

### 6.6.1 SONG NEXT

Program akan memutar lagu selanjutnya yang berada di dalam queue. Current song akan dimasukkan ke dalam history. Jika queue kosong, akan diputar kembali current song.

**Sebelum :**

→ Apabila queue terisi :

```
ENTER COMMAND: STATUS
Now Playing:
Arctic Monkeys - 505 - Favourite Worst Nightmare

Queue:
1. Hivi! - Kereta Kencan - Kereta Kencan
```

Gambar 6.6.1.1

→ Apabila queue kosong :

```
ENTER COMMAND: STATUS
Now Playing:
Arctic Monkeys - 505 - Favourite Worst Nightmare

Queue:
Your queue is empty.
```

Gambar 6.6.1.2

**Song Next :**

→ *Output* apabila queue terisi :

```
ENTER COMMAND: SONG NEXT
Memutar lagu selanjutnya Kereta Kencan oleh Hivi!
```

Gambar 6.6.1.3

→ *Output* apabila queue kosong:

```
ENTER COMMAND: SONG NEXT
Queue kosong, memutar kembali lagu 505 oleh Arctic Monkeys
```

Gambar 6.6.1.4

**Sesudah :**

→ Apabila queue terisi :

```
ENTER COMMAND: STATUS
Now Playing:
Hivi! - Kereta Kencan - Kereta Kencan

Queue:
Your queue is empty.
```

Gambar 6.6.1.5

→ Apabila queue kosong :

```
ENTER COMMAND: STATUS
Now Playing:
Arctic Monkeys - 505 - Favourite Worst Nightmare

Queue:
Your queue is empty.
```

Gambar 6.6.1.6

## 6.6.2 SONG PREVIOUS

Program akan memutar lagu sebelumnya yang berada pada history. Current song akan dimasukkan ke dalam queue pada urutan pertama. Jika history kosong, akan diputar kembali current song.

**Sebelum :**

```
ENTER COMMAND: STATUS
Now Playing:
Hivi! - Kereta Kencan - Kereta Kencan

Queue:
Your queue is empty.
```

Gambar 6.6.2.1

**Song Previous :**

→ *Output* apabila history terisi :

```
ENTER COMMAND: SONG PREVIOUS
Memutar lagu sebelumnya 505 oleh Arctic Monkeys
```

Gambar 6.6.2.2

→ *Output* apabila history kosong :

```
ENTER COMMAND: SONG PREVIOUS
Riwayat lagu kosong, memutar kembali lagu Kereta Kencan oleh Hivi!
```

Gambar 6.6.2.3

**Sesudah :**

→ Apabila history terisi :

```
ENTER COMMAND: STATUS
Now Playing:
Arctic Monkeys - 505 - Favourite Worst Nightmare

Queue:
1. Hivi! - Kereta Kencan - Kereta Kencan
```

Gambar 6.6.2.4

→ Apabila history kosong :

```
ENTER COMMAND: STATUS
Now Playing:
Hivi! - Kereta Kencan - Kereta Kencan

Queue:
Your queue is empty.
```

Gambar 6.6.2.5

## 6.7 PLAYLIST

### 6.7.1 PLAYLIST CREATE

Program dilakukan untuk membuat sebuah playlist baru dan ditambahkan ke daftar playlist yang dimiliki oleh seorang user.

**Sebelum :**

```
ENTER COMMAND: LIST PLAYLIST
Kamu tidak memiliki playlist.
```

Gambar 6.7.1.1

**Playlist Create :**

→ *Output* apabila nama playlist valid :

```
ENTER COMMAND: PLAYLIST CREATE
Masukkan nama playlist yang ingin dibuat : K-Pop ON!
Playlist K-Pop ON! berhasil dibuat! Silakan masukkan lagu - lagu artis terkini kesayangan Anda!
```

Gambar 6.7.1.2

→ *Output* apabila nama playlist tidak valid :

```
ENTER COMMAND: PLAYLIST CREATE
Masukkan nama playlist yang ingin dibuat : a
Minimal terdapat 3 karakter selain whitespace dalam nama playlist. Silakan coba lagi.
```

Gambar 6.7.1.3

**Sesudah :**

```
ENTER COMMAND: LIST PLAYLIST
Daftar Playlist yang kamu miliki:
1. K-Pop ON!
```

Gambar 6.7.1.4

### 6.7.2 PLAYLIST ADD SONG

Program dilakukan untuk menambahkan sebuah lagu ke dalam playlist user. Lagu dipilih berdasarkan penyanyi, album, dan id lagu yang ditentukan dari input user.

→ *Output* apabila seluruh *input* valid :

```

ENTER COMMAND: PLAYLIST ADD SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : BLACKPINK
Daftar Album oleh BLACKPINK :
1. SQUARE ONE
2. SQUARE TWO
3. SQUARE UP
4. BORN PINK
5. THE ALBUM
Masukkan Judul Album yang dipilih: SQUARE UP
Daftar Lagu Album SQUARE UP oleh BLACKPINK :
1. DDU-DU DDU-DU
2. Forever Young
3. Really
4. See U Later
Masukkan ID Lagu yang dipilih : 1
1. K-Pop ON!
Masukkan ID Playlist yang dipilih : 1
Lagu dengan judul DDU-DU DDU-DU pada album SQUARE UP oleh penyanyi BLACKPINK berhasil ditambahkan ke dalam playlist K-Pop ON!.

```

Gambar 6.7.2.1

→ *Output* apabila ID Playlist tidak valid :

```

ENTER COMMAND: PLAYLIST ADD SONG
Daftar Penyanyi :
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : BLACKPINK
Daftar Album oleh BLACKPINK :
1. SQUARE ONE
2. SQUARE TWO
3. SQUARE UP
4. BORN PINK
5. THE ALBUM
Masukkan Judul Album yang dipilih: SQUARE TWO
Daftar Lagu Album SQUARE TWO oleh BLACKPINK :
1. PLAYING WITH FIRE
2. STAY
Masukkan ID Lagu yang dipilih : 2
1. K-Pop ON!
Masukkan ID Playlist yang dipilih : 3
Tidak ada playlist dengan playlist ID 3

```

Gambar 6.7.2.2

### 6.7.3 PLAYLIST ADD ALBUM

Program dilakukan untuk menambahkan lagu-lagu yang ada di dalam sebuah album ke dalam playlist user. Album dipilih berdasarkan penyanyi dan album dari input user.

→ *Output* apabila seluruh *input* valid :

```

ENTER COMMAND: PLAYLIST ADD ALBUM
Daftar Penyanyi:
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : NewJeans
Daftar Album oleh NewJeans :
1. New Jeans
2. OMG
3. Get Up
Masukkan Judul Album yang dipilih: Get Up
1. K-Pop ON!
Masukkan ID Playlist yang dipilih : 1
Album dengan judul Get Up berhasil ditambahkan ke dalam playlist pengguna K-Pop ON!.

```

Gambar 6.7.3.1

→ *Output* apabila *input* tidak valid :

◆ *Output* apabila *input* nama penyanyi tidak valid :

```

ENTER COMMAND: PLAYLIST ADD ALBUM
Daftar Penyanyi:
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : nwh
Penyanyi nwh tidak ada dalam daftar. Silakan coba lagi.

```

Gambar 6.7.3.2

◆ *Output apabila input nama album tidak valid :*

```

ENTER COMMAND: PLAYLIST ADD ALBUM
Daftar Penyanyi:
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : NewJeans
Daftar Album oleh NewJeans :
1. New Jeans
2. OMG
3. Get Up
Masukkan Judul Album yang dipilih: newjeans
Album newjeans tidak ada dalam daftar. Silakan coba lagi.

```

Gambar 6.7.3.3

◆ *Output apabila input ID playlist tidak valid :*

```

ENTER COMMAND: PLAYLIST ADD ALBUM
Daftar Penyanyi:
1. Arctic Monkeys
2. NewJeans
3. Hivi!
4. BLACKPINK
Masukkan Nama Penyanyi yang dipilih : NewJeans
Daftar Album oleh NewJeans :
1. New Jeans
2. OMG
3. Get Up
Masukkan Judul Album yang dipilih: OMG
1. K-Pop ON!
Masukkan ID Playlist yang dipilih : 2
Tidak ada playlist dengan playlist ID 2

```

Gambar 6.7.3.4

## 6.7.4 PLAYLIST SWAP <id> <x> <y>

Program dilakukan untuk menukar lagu pada urutan ke x dengan lagu pada urutan ke y yang ada di playlist dengan urutan id dalam daftar playlist user.

**Sebelum :**

```

1. BLACKPINK - PLAYING WITH FIRE - SQUARE TWO
2. BLACKPINK - STAY - SQUARE TWO
3. NewJeans - OMG - OMG
4. NewJeans - Ditto - OMG

```

Gambar 6.7.4.1

**Playlist Swap :**

→ *Output apabila input urutan lagu dan ID playlist valid :*

```

ENTER COMMAND: PLAYLIST SWAP 1 1 2
Berhasil menukar lagu dengan nama STAY dengan PLAYING WITH FIRE di playlist K-Pop ON!.

```

Gambar 6.7.4.2

→ *Output apabila input urutan lagu dan ID playlist tidak valid :*

```

ENTER COMMAND: PLAYLIST SWAP 3 2 1
Tidak ada playlist dengan playlist ID 3

```

Gambar 6.7.4.3

**Sesudah :**

```

1. BLACKPINK - STAY - SQUARE TWO
2. BLACKPINK - PLAYING WITH FIRE - SQUARE TWO
3. NewJeans - OMG - OMG
4. NewJeans - Ditto - OMG

```

Gambar 6.7.4.4

### 6.7.5 PLAYLIST REMOVE <id> <n>

Program dilakukan untuk menghapus lagu pada urutan ke n yang ada di dalam playlist dengan urutan id dalam daftar playlist user.

**Sebelum :**

```
1. BLACKPINK - STAY - SQUARE TWO
2. BLACKPINK - PLAYING WITH FIRE - SQUARE TWO
3. NewJeans - OMG - OMG
4. NewJeans - Ditto - OMG
```

Gambar 6.7.5.1

**Playlist Remove :**

→ Output apabila *input* urutan lagu dan ID playlist valid :

```
ENTER COMMAND: PLAYLIST REMOVE 1 3
Lagu OMG oleh NewJeans telah dihapus dari playlist K-Pop ON!!
```

Gambar 6.7.5.2

→ Output apabila *input* urutan lagu dan ID playlist tidak valid :

```
ENTER COMMAND: PLAYLIST REMOVE 3 12
Tidak ada playlist dengan playlist ID 3
```

Gambar 6.7.6.3

**Sesudah :**

```
ENTER COMMAND: STATUS
Now Playing:
No songs have been played yet. Please search for a song to begin playback.
Queue:
1. BLACKPINK - STAY - SQUARE TWO
2. BLACKPINK - PLAYING WITH FIRE - SQUARE TWO
3. NewJeans - Ditto - OMG
```

Gambar 6.7.5.4

### 6.7.6 PLAYLIST DELETE

Program dilakukan untuk menghapus playlist yang ada di dalam daftar playlist user.

→ Output apabila *input* id playlist valid

```
ENTER COMMAND: PLAYLIST DELETE
1. PLAYLIST 1
2. PLAYLIST 2
Masukkan ID Playlist yang dipilih : 1
Playlist ID 1 dengan judul PLAYLIST 1 berhasil dihapus.
```

Gambar 6.7.6.1

→ Output apabila *input* id playlist tidak valid

```
ENTER COMMAND: PLAYLIST DELETE
1. PLAYLIST 2
2. PLAYLIST 1
Masukkan ID Playlist yang dipilih : 3
Tidak ada playlist dengan ID 3 dalam daftar playlist pengguna. Silakan coba lagi.
```

Gambar 6.7.6.2

## 6.8 STATUS

Program akan menampilkan lagu yang sedang diputar beserta queue song dan asal playlist dari lagu tersebut.

→ *Output* apabila tidak ada lagu yang diputar

```
ENTER COMMAND: STATUS
Now Playing:
No songs have been played yet. Please search for a song to begin playback.
Queue:
Your queue is empty.
```

Gambar 6.8.1

→ *Output* apabila ada lagu tetapi queue kosong

```
ENTER COMMAND: STATUS
Now Playing:
Hivi! - Kereta Kencan - Kereta Kencan

Queue:
Your queue is empty.
```

Gambar 6.8.2

→ *Output* apabila ada lagu dan queue

```
ENTER COMMAND: STATUS
Now Playing:
Arctic Monkeys - 505 - Favourite Worst Nightmare

Queue:
1. Hivi! - Kereta Kencan - Kereta Kencan
```

Gambar 6.8.3

→ *Output* apabila ada lagu, queue, dan playlist

```
ENTER COMMAND: STATUS
Current Playlist: K-Pop ON!

Now Playing:
BLACKPINK - BOOMBAYAH - SQUARE ONE

Queue:
1. BLACKPINK - WHISTLE - SQUARE ONE
2. NewJeans - Ditto - OMG
3. NewJeans - OMG - OMG
```

Gambar 6.8.4

## 6.9 SAVE

Program akan menyimpan state aplikasi terbaru ke dalam suatu file. Penyimpanan file dilakukan pada folder tertentu dengan nama file yang akan disimpan.

→ *Output* apabila nama file tidak valid :

```
ENTER COMMAND: SAVE
File tidak dapat ditemukan!
```

Gambar 6.9.1

→ *Output* apabila nama file valid :

```
ENTER COMMAND: SAVE hivivihi.txt
File berhasil disimpan!
```

Gambar 6.9.2

## 6.10 QUIT

Program untuk keluar dari aplikasi WayangWave. Terdapat 2 pilihan yang diberikan bagi pengguna, yaitu ingin menyimpan data atau tidak. Jika iya, akan memanggil program SAVE, dan jika tidak akan langsung keluar dari aplikasi WayangWave.

→ *Output* apabila ingin menyimpan data :



```

ENTER COMMAND: QUIT
Apakah kamu ingin menyimpan data sesi sekarang?(Y/N) Y
File akan disimpan dengan nama : kpop.txt
File berhasil disimpan!
Kamu keluar dari WayangWave.
Dadah ^_^/

```

Gambar 6.10.1

→ *Output* apabila tidak ingin menyimpan data :

```

ENTER COMMAND: QUIT
Apakah kamu ingin menyimpan data sesi sekarang?(Y/N) N
Kamu keluar dari WayangWave.
Dadah ^_^/

```

Gambar 6.10.2

## 6.11 HELP

Program akan menampilkan daftar command yang dapat dieksekusi beserta deskripsinya.

→ *Output* ketika help dipanggil sebelum memasuki program

```

                                w E L c o m e   t o

  W A Y A N G W A V E

===== [ Main Menu WayangWave ] =====
1. START
2. LOAD <namafile>
3. HELP
ENTER COMMAND: HELP
===== [ Menu Help WayangWave ] =====
1. START -> Untuk masuk sesi baru
2. LOAD -> Untuk memulai sesi berdasarkan file konfigurasi
ENTER COMMAND: █

```

Gambar 6.11.1

→ *Output* ketika help dipanggil setelah memasuki program

```

                                w E L c o m e   t o

  W A Y A N G W A V E

===== [ Main Menu WayangWave ] =====
1. START
2. LOAD <namafile>
3. HELP
ENTER COMMAND: START
File konfigurasi aplikasi berhasil dibaca. WayangWave berhasil dijalankan.

```

Gambar 6.11.2

```

ENTER COMMAND: HELP
=====
[ Menu Help WayangWave ]=====
1. LIST DEFAULT -> Untuk menampilkan list penyanyi, list album dari seorang penyanyi, serta list lagu dari suatu album
2. LIST PLAYLIST -> Untuk menampilkan playlist yang ada pada pengguna
3. PLAY SONG -> Untuk memainkan lagu berdasarkan nama penyanyi, nama album, dan ID lagu
4. PLAY PLAYLIST -> Untuk memainkan lagu berdasarkan ID playlist
5. QUEUE SONG -> Untuk menambahkan lagu ke dalam queue
6. QUEUE PLAYLIST -> Untuk menambahkan lagu yang ada dalam playlist ke dalam queue
6. QUEUE SWAP <x> <y> -> Untuk menukar lagu pada urutan ke (x) dan juga urutan ke (y)
7. QUEUE REMOVE <id> -> Untuk menghapus lagu dari queue dengan inputan urutan (id) lagu yang ingin dihapus dari queue
8. QUEUE CLEAR -> Untuk mengosongkan queue
9. SONG NEXT -> Untuk memutar lagu selanjutnya dalam queue jika ada
10. SONG PREVIOUS -> Untuk memutar lagu yang sebelumnya diputarkan jika ada
11. PLAYLIST CREATE -> Untuk membuat playlist baru dan ditambahkan pada daftar playlist pengguna
12. PLAYLIST ADD -> Untuk menambahkan lagu atau album ke dalam playlist yang sudah dimiliki pengguna dalam daftar playlist
13. PLAYLIST SWAP <id> <x> <y> -> Untuk menukar lagu pada urutan ke (x) dan juga urutan ke (y) di playlist dengan index (id)
14. PLAYLIST REMOVE <id> <n> -> Untuk menghapus lagu dengan urutan (n) pada playlist dengan index (id)
15. PLAYLIST DELETE -> Untuk melakukan penghapusan suatu existing playlist dalam daftar playlist
16. STATUS -> Untuk menampilkan lagu yang sedang diputarkan, dari playlist mana lagu tersebut, serta queue song (jika ada)
17. SAVE <filename> -> Untuk menyimpan state WayangWave terbaru ke dalam suatu file bernama (filename) dengan tipe txt
18. QUIT -> Untuk keluar dari aplikasi WayangWave

```

Gambar 6.11.3

## 6.12 INVALID COMMAND

Program akan mengeluarkan teks error apabila command yang dimasukkan tidak valid.

→ *Output* ketika masukan tidak valid

```

ENTER COMMAND: TUBES
Command tidak bisa dieksekusi!

```

Gambar 6.12.1

→ *Output* ketika mengeksekusi command load sesudah memasuki program

```

ENTER COMMAND: load savefile.txt
Command tidak bisa dieksekusi!

```

Gambar 6.12.2

→ *Output* ketika mengeksekusi command sebelum memasuki program

```

=====
[ Main Menu WayangWave ]=====
1. START
2. LOAD <namafile>
3. HELP
ENTER COMMAND: QUEUE SONG
Command tidak bisa dieksekusi. Mohon masukkan command yang sesuai.

```

Gambar 6.12.3

## 7 Test Script

No .	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	START	Memastikan program bisa membaca file konfigurasi yang berisi daftar penyanyi dan album yang dimiliki.	Mengetikkan START	<b>1. Command valid</b> Gambar 6.1.1 <b>2. Command tidak valid</b> Gambar 6.1.2	File konfigurasi default terbaca dan program berisi data dari file default.	(Sesuai yang diharapkan)
2	LOAD	Memastikan program dapat membaca file konfigurasi hasil <i>input</i>	Mengetikkan LOAD filename	<b>1. Command valid dan file input valid</b> Gambar 6.2.1	File konfigurasi hasil <i>input</i> yang berisi daftar	(Sesuai yang diharapkan)

		yang berisi daftar penyanyi, album, lagu, <i>queue</i> , <i>history</i> , dan <i>playlist</i> yang dimiliki.		<b>2. Command valid dan file input tidak valid</b> Gambar 6.2.2	penyanyi, album, lagu, <i>queue</i> , <i>history</i> , dan <i>playlist</i> dapat terbaca.	
3	LIST DEFAULT	Memastikan list penyanyi, album penyanyi, dan lagu yang ada dari album yang dipilih berdasarkan konfirmasi pengguna dapat ditampilkan.	1. Mengetikkan LIST DEFAULT 2. Memilih Y/N 3. Mengetikkan nama album/penyanyi	<b>1. Command valid, input Y, nama valid</b> Gambar 6.3.1.1 Gambar 6.3.1.2 Gambar 6.3.1.3 Gambar 6.3.1.4 <b>2. Command valid, input Y, nama tidak valid</b> Gambar 6.3.1.1 Gambar 6.3.1.2 Gambar 6.3.1.5 Gambar 6.3.1.6 <b>3. Command valid, input N</b> Gambar 6.3.1.7 <b>4. Command tidak valid</b> Gambar 6.3.1.8	List penyanyi, album penyanyi, dan lagu yang ada dari album yang dipilih berdasarkan konfirmasi pengguna dapat ditampilkan.	(Sesuai yang diharapkan)
4	LIST PLAYLIST	Memastikan daftar playlist yang ada dapat ditampilkan.	Mengetikkan LIST PLAYLIST	<b>1. Pengguna tidak memiliki playlist</b> Gambar 6.3.2.1 <b>2. Pengguna memiliki playlist</b> Gambar 6.3.2.2	Daftar playlist yang ada dapat ditampilkan.	(Sesuai yang diharapkan)
5	PLAY SONG	Memastikan lagu diputar berdasarkan masukan, queue dan history lagu kosong.	1. Mengetikkan PLAY SONG 2. Memasukkan nama penyanyi 3. Memasukkan nama album 4. Memasukkan id lagu	<b>1. Command valid, seluruh input valid</b> Gambar 6.4.1.2 <b>2. Command valid, ID playlist tidak valid</b> Gambar 6.4.1.4	Lagu diputar berdasarkan masukan, queue dan history lagu kosong.	(Sesuai yang diharapkan)
6	PLAY PLAYLIST	Memastikan lagu diputar berdasarkan id playlist.	1. Mengetikkan PLAY PLAYLIST	<b>1. Command valid, ID playlist valid</b> Gambar 6.4.2.3	Lagu diputar berdasarkan ID playlist.	(Sesuai yang diharapkan)

			2. Memasukkan id playlist	<b>2. Command valid, ID playlist tidak valid</b> Gambar 6.4.2.4		
7	QUEUE SONG	Memastikan lagu ditambah ke dalam queue.	1. Mengetikkan QUEUE SONG 2. Memasukkan nama penyanyi 3. Memasukkan nama album 4. Memasukkan id lagu	<b>1. Command valid, seluruh input valid</b> Gambar 6.5.1.2 <b>2. Command valid, penyanyi tidak valid</b> Gambar 6.5.1.3 <b>3. Command valid, album tidak valid</b> Gambar 6.5.1.4 <b>4. Command valid, ID lagu tidak valid</b> Gambar 6.5.1.5	Lagu ditambah ke dalam queue.	(Sesuai yang diharapkan)
8	QUEUE PLAYLIST	Memastikan lagu yang ada di dalam playlist ditambah ke dalam queue.	1. Mengetikkan QUEUE PLAYLIST 2. Memasukkan id playlist	<b>1. Command valid, ID playlist valid</b> Gambar 6.5.2.2 <b>2. Command valid, ID playlist tidak valid</b> Gambar 6.5.2.3	Lagu yang ada di dalam playlist ditambah ke dalam queue.	(Sesuai yang diharapkan)
9	QUEUE SWAP	Memastikan urutan lagu dapat ditukar dari x ke y pada queue.	Mengetikkan QUEUE SWAP x y	<b>1. Command valid, ID queue valid</b> Gambar 6.5.3.2 <b>2. Command valid, ID queue tidak valid</b> Gambar 6.5.3.3	Lagu dengan urutan x ditukar dengan lagu dengan urutan y dalam queue.	(Sesuai yang diharapkan)
10	QUEUE REMOVE	Memastikan program dapat menghapus lagu sesuai urutan dari queue.	Mengetikkan QUEUE REMOVE id lagu	<b>1. Command valid, ID queue valid</b> Gambar 6.5.4.2 <b>2. Command valid, ID queue tidak valid</b> Gambar 6.5.4.3	Lagu dapat dihapus sesuai urutan dari queue.	(Sesuai yang diharapkan)
11	QUEUE CLEAR	Memastikan queue kosong.	Mengetikkan QUEUE CLEAR	<b>Command valid</b> Gambar 6.5.5.2	Queue kosong.	(Sesuai yang diharapkan)
12	SONG NEXT	Memastikan lagu yang diputar adalah lagu	Mengetikkan SONG NEXT	<b>1. Command valid, queue terisi</b> Gambar 6.6.1.3	Memutar lagu selanjutnya yang berada di dalam queue.	(Sesuai yang diharapkan)

		selanjutnya yang berada di dalam queue. Jika queue kosong, lagu yang diputar adalah current song.		<b>2. Command valid, queue kosong</b> Gambar 6.6.1.4	Jika queue kosong, memutar current song.	
13	SONG PREVIOUS	Memastikan lagu yang diputar adalah lagu yang terakhir kali diputar. Jika tidak ada lagu yang diputar sebelumnya, lagu yang diputar adalah current song.	Mengetikkan SONG PREVIOUS	<b>1. Command valid, history terisi</b> Gambar 6.6.2.2 <b>2. Command valid, history kosong</b> Gambar 6.6.2.3	Memutar lagu yang terakhir kali diputar. Jika riwayat lagu kosong, memutar current song.	(Sesuai yang diharapkan)
14	PLAYLIST CREATE	Memastikan program bisa membuat sebuah playlist kosong yang baru dan dimasukkan ke daftar playlist pengguna.	1. Mengetikkan PLAYLIST CREATE 2. Memasukkan nama playlist yang ingin dibuat	<b>1. Command valid, nama playlist valid</b> Gambar 6.7.1.2 <b>2. Command valid, nama playlist tidak valid</b> Gambar 6.7.1.3	Membuat sebuah playlist kosong.	(Sesuai yang diharapkan)
15	PLAYLIST ADD SONG	Memastikan program bisa menambahkan lagu ke dalam playlist pengguna.	1. Mengetikkan PLAYLIST ADD SONG 2. Memasukkan nama penyanyi 3. Memasukkan nama album 4. Memasukkan id lagu 5. Memasukkan id playlist	<b>1. Command valid, seluruh input valid</b> Gambar 6.7.2.1 <b>2. Command valid, ID playlist tidak valid</b> Gambar 6.7.2.2	Menambahkan lagu ke dalam playlist sesuai dengan masukan nama penyanyi, album dan lagu.	(Sesuai yang diharapkan)
16	PLAYLIST ADD ALBUM	Memastikan program bisa menambahkan lagu yang ada di dalam sebuah album ke dalam playlist pengguna.	1. Mengetikkan PLAYLIST ADD ALBUM 2. Memasukkan nama penyanyi	<b>1. Command valid, seluruh input valid</b> Gambar 6.7.3.1 <b>2. Command valid, penyanyi tidak valid</b> Gambar 6.7.3.2	Menambahkan lagu yang berada di dalam sebuah album ke dalam playlist.	(Sesuai yang diharapkan)

			3. Memasukkan nama album 4. Memasukkan id playlist	<b>3. Command valid, album tidak valid</b> Gambar 6.7.3.3 <b>4. Command valid, ID playlist tidak valid</b> Gambar 6.7.3.4		
17	PLAYLIST SWAP	Memastikan program bisa menukar 2 lagu yang ada dalam suatu playlist.	Mengetikkan PLAYLIST SWAP idplaylist x y	<b>1. Command valid, seluruh input valid</b> Gambar 6.7.4.1 <b>2. Command valid, ID playlist tidak valid</b> Gambar 6.7.4.3	Lagu dengan urutan x ditukar dengan lagu urutan y pada sebuah playlist.	(Sesuai yang diharapkan)
18	PLAYLIST REMOVE	Memastikan program bisa menghapus lagu dengan urutan tertentu pada suatu playlist.	Mengetikkan PLAYLIST REMOVE idplaylist idlagu	<b>1. Command valid, seluruh input valid</b> Gambar 6.7.5.2 <b>2. Command valid, ID playlist tidak valid</b> Gambar 6.7.5.3	Lagu dapat dihapus sesuai urutan pada playlist.	(Sesuai yang diharapkan)
19	PLAYLIST DELETE	Memastikan program bisa menghapus playlist yang ada di daftar playlist pengguna.	1. Mengetikkan PLAYLIST DELETE 2. Memasukkan id playlist	<b>1. Command valid, ID playlist valid</b> Gambar 6.7.6.1 <b>2. Command valid, ID playlist tidak valid</b> Gambar 6.7.6.2	Menghapus playlist yang ada di daftar pengguna.	(Sesuai yang diharapkan)
20	STATUS	Memastikan program bisa menampilkan lagu yang sedang dimainkan dan daftar queue song dan asal playlist.	Mengetikkan STATUS	<b>1. Tidak ada lagu yang diputar</b> Gambar 6.8.1 <b>2. Ada lagu yang diputar, namun queue kosong</b> Gambar 6.8.2 <b>3. Ada lagu dan queue terisi</b> Gambar 6.8.3 <b>4. Ada lagu dari playlist yang diputar, dan queue terisi</b> Gambar 6.8.4	Menampilkan nama lagu, album, dan playlist yang sedang diputar serta queue.	(Sesuai yang diharapkan)

21	SAVE	Memastikan bisa menyimpan state aplikasi ke suatu file txt.	1. Mengetikkan SAVE namafile	<b>1. Command valid, nama file valid</b> Gambar 6.9.1 <b>2. Command valid, nama file tidak valid</b> Gambar 6.9.2	State aplikasi tersimpan ke dalam file txt.	(Sesuai yang diharapkan)
22	QUIT	Memastikan command bisa digunakan untuk keluar dari WayangWave.	1. Mengetikkan QUIT 2. Memilih Y/N	<b>1. Command valid, ingin menyimpan data Y</b> Gambar 6.10.1 <b>2. Command valid, tidak ingin menyimpan N</b> Gambar 6.10.2	Keluar dari aplikasi WayangWave.	(Sesuai yang diharapkan)
23	HELP	Memastikan program bisa menampilkan daftar command yang bisa dieksekusi.	Mengetikkan HELP.	<b>1. Command help sebelum memasuki program</b> Gambar 6.11.1 <b>2. Command help valid setelah memasuki program</b> Gambar 6.11.2 Gambar 6.11.3	Program dapat menampilkan daftar command yang bisa dieksekusi.	(Sesuai yang diharapkan)
24	INVALID COMMAND	Memastikan adanya keluaran teks error jika command yang dimasukkan tidak sesuai.	Mengecek adanya command yang tidak valid sesuai input command.	<b>1. Command tidak valid</b> Gambar 6.12.1 <b>2. Command load sesudah memasuki program</b> Gambar 6.12.2 <b>3. Command queue sebelum memasuki program</b> Gambar 6.12.3	Program mengeluarkan teks error pada command yang tidak valid.	(Sesuai yang diharapkan)

## 8 Pembagian Kerja dalam Kelompok

NAMA	NIM	DESKRIPSI TUGAS
Muhammad Daffa Ikram	18222020	<ul style="list-style-type: none"> <li>- Membuat command START, QUEUE SWAP, QUEUE REMOVE, QUEUE CLEAR, SAVE, dan INVALID COMMAND.</li> <li>- Mengisi laporan bagian TEST SCRIPT untuk beberapa command</li> <li>- Menambahkan perubahan pada deskripsi alasan pemilihan dan implementasi ADT</li> </ul>
Christoper Daniel	18222034	<ul style="list-style-type: none"> <li>- Membuat command PLAYLIST CREATE, PLAYLIST ADD, PLAYLIST SWAP, PLAYLIST REMOVE, dan PLAYLIST DELETE.</li> <li>- Mengisi laporan bagian ADT untuk sebagian besar struktur data.</li> <li>- Mengisi laporan bagian DATA TEST untuk beberapa command.</li> <li>- Mengisi laporan bagian TEST SCRIPT untuk beberapa command.</li> </ul>
Yovanka Sandrina Maharaja	18222094	<ul style="list-style-type: none"> <li>- Membuat command LOAD, LIST DEFAULT, LIST PLAYLIST, PLAY SONG, QUEUE SONG, QUIT, dan HELP.</li> <li>- Mengisi laporan bagian DATA TEST untuk hampir semua command.</li> <li>- Mengisi laporan bagian TEST SCRIPT untuk beberapa command.</li> </ul>
Ervina Limka	18222100	<ul style="list-style-type: none"> <li>- Membuat command PLAY PLAYLIST, QUEUE PLAYLIST, SONG NEXT, SONG PREVIOUS, dan STATUS.</li> <li>- Mengisi laporan bagian ADT untuk sebagian besar struktur data.</li> <li>- Mengisi laporan bagian Algoritma Menarik</li> <li>- Mengisi laporan bagian DATA TEST untuk beberapa command.</li> <li>- Mengisi laporan bagian TEST SCRIPT untuk beberapa command.</li> </ul>



## 9 Lampiran

### 9.1 Deskripsi Tugas Besar

#### Spesifikasi Umum

Buatlah sebuah aplikasi simulasi berbasis CLI (command-line interface). Sistem ini dibuat dalam **bahasa C** dengan menggunakan **struktur data yang sudah kalian pelajari** di mata kuliah ini. Kalian boleh menggunakan (atau memodifikasi) struktur data yang sudah kalian buat untuk praktikum pada tugas besar ini. Daftar ADT yang wajib digunakan dapat dilihat pada bagian [Daftar ADT](#). Library yang boleh digunakan hanya **stdio.h**, **stdlib.h**, **time.h**, dan **math.h**

#### Command

##### a. START

START merupakan salah satu command yang dimasukkan pertama kali dalam WayangWave. Setelah menekan Enter, dibaca file konfigurasi default yang berisi daftar penyanyi serta album yang dimiliki.

```
ENTER COMMAND : START;  
File konfigurasi aplikasi berhasil dibaca. WayangWave berhasil dijalankan.
```

##### b. LOAD <filename>

LOAD merupakan salah satu command yang dimasukkan pertama kali dalam WayangWave. Command ini memiliki satu argumen yaitu filename yang merepresentasikan suatu save file yang ingin dibuka. File didapatkan dari folder tertentu, contohnya save. Setelah menekan Enter, akan dibaca save file <filename> yang berisi list penyanyi, album, dan lagu yang bisa diputar. Lebih detailnya bisa dilihat pada Konfigurasi Aplikasi.

```
ENTER COMMAND : LOAD filenihil.txt;  
// Meload /save/filenihil.txt  
  
Save file tidak ditemukan. WayangWave gagal dijalankan.  
  
ENTER COMMAND : LOAD savefile.txt;  
// Meload /save/savefile.txt  
  
Save file berhasil dibaca. WayangWave berhasil dijalankan.
```

##### c. LIST

LIST merupakan command yang digunakan untuk menampilkan list playlist yang ada, list penyanyi, list album dari penyanyi, dan list lagu yang ada di album. Terdapat dua jenis list, DEFAULT dan PLAYLIST.

##### 1. LIST DEFAULT

Command LIST DEFAULT digunakan untuk melihat list penyanyi yang ada. Selanjutnya dapat memilih untuk melihat album dari penyanyi yang dipilih. Kemudian melihat lagu yang ada dari album yang dipilih. Terdapat konfirmasi apakah ingin melihat album/lagu.

```

ENTER COMMAND : LIST DEFAULT;

Daftar Penyanyi :
1. New Jeans
2. BLACKPINK
3. Central Cee
4. Rich Brian
5. Denny Caknan
6. Via Vallen

Ingin melihat album yang ada?(Y/N) : Y;
Pilih penyanyi untuk melihat album mereka: BLACKPINK;

Daftar Album oleh BLACKPINK :
1. BORN PINK
2. THE ALBUM
3. SQUARE ONE
4. SQUARE TWO

Ingin melihat lagu yang ada?(Y/N) : Y;
Pilih album untuk melihat lagu yang ada di album : SQUARE ONE;

Daftar Lagu di SQUARE ONE:
1. WHISTLE
2. BOOMBAYAH

```

---

```

ENTER COMMAND : LIST DEFAULT;

Daftar Penyanyi :
1. New Jeans
2. BLACKPINK
3. Central Cee
4. Rich Brian
5. Denny Caknan
6. Via Vallen

Ingin melihat album yang ada?(Y/N) : N;

```

## 2. LIST PLAYLIST

Command LIST PLAYLIST digunakan untuk menampilkan playlist yang ada pada pengguna.

```

ENTER COMMAND : LIST PLAYLIST;

Daftar playlist yang kamu miliki:
1. K-pop Enjoyer
2. Z E R O 4.0
3. Be the main character
4. Hey i liked you.

```

---

```

ENTER COMMAND : LIST PLAYLIST;

```

Daftar playlist yang kamu miliki:  
Kamu tidak memiliki playlist.

#### d. **PLAY**

PLAY merupakan command yang digunakan untuk memutar lagu atau playlist yang dipilih. Ketika command PLAY dieksekusi, queue yang ada dihapus ketika memainkan lagu atau digantikan oleh lagu dalam playlist ketika memainkan playlist. Terdapat dua jenis play, SONG dan PLAYLIST.

##### 1. **PLAY SONG**

Command PLAY SONG digunakan untuk memainkan lagu berdasarkan masukan nama penyanyi, nama album, dan id lagu. Ketika command ini berhasil dieksekusi, queue dan riwayat lagu akan menjadi kosong.

```
ENTER COMMAND : PLAY SONG;

Daftar Penyanyi :
1. New Jeans
2. BLACKPINK
3. Central Cee
4. Rich Brian
5. Denny Caknan
6. Via Vallen

Masukkan Nama Penyanyi yang dipilih : BLACKPINK;

Daftar Album oleh BLACKPINK :
1. BORN PINK
2. THE ALBUM
3. SQUARE ONE
4. SQUARE TWO

Masukkan Nama Album yang dipilih : BORN PINK;

Daftar Lagu Album BORN PINK oleh BLACKPINK :
1. Pink Venom
2. Shut Down
3. Typa Girl
4. Yeah Yeah Yeah
5. Hard to Love
6. The Happiest Girl
7. Tally
8. Ready for Love

Masukkan ID Lagu yang dipilih : 3;

Memutar lagu "Type Girl" oleh "BLACKPINK".
```

##### 2. **PLAY PLAYLIST**

Command PLAY PLAYLIST digunakan untuk memainkan lagu berdasarkan id playlist. Ketika command ini berhasil dieksekusi, current song akan menjadi lagu pada urutan pertama playlist dan queue akan berisi semua lagu yang ada dalam

playlist yang akan dimainkan dan isi riwayat lagu sama dengan queue, tetapi dengan urutan yang di-reverse.

```
ENTER COMMAND : PLAY PLAYLIST;  
  
Masukkan ID Playlist: 2;  
  
Memutar playlist "K-pop Enjoyer".
```

#### e. **QUEUE**

QUEUE merupakan command yang digunakan untuk memanipulasi queue lagu. Command ini memiliki 5 tipe, yaitu SONG, PLAYLIST, SWAP, REMOVE, dan CLEAR.

##### 1. **QUEUE SONG**

Command QUEUE SONG digunakan untuk menambahkan lagu ke dalam queue. Command ini menerima input lagu berdasarkan nama penyanyi, nama album, dan id dari lagu yang ingin dimasukkan ke dalam queue.

```
ENTER COMMAND : QUEUE SONG;  
  
Daftar Penyanyi :  
  1. New Jeans  
  2. BLACKPINK  
  3. Central Cee  
  4. Rich Brian  
  5. Denny Caknan  
  6. Via Vallen  
  
Masukkan Nama Penyanyi: BLACKPINK;  
  
Daftar Album oleh BLACKPINK :  
  1. BORN PINK  
  2. THE ALBUM  
  3. SQUARE ONE  
  4. SQUARE TWO  
  
Masukkan Nama Album yang dipilih : BORN PINK;  
  
Daftar Lagu Album BORN PINK oleh BLACKPINK :  
  1. Pink Venom  
  2. Shut Down  
  3. Typa Girl  
  4. Yeah Yeah Yeah  
  5. Hard to Love  
  6. The Happiest Girl  
  7. Tally  
  8. Ready for Love  
  
Masukkan ID Lagu yang dipilih: 3;  
  
Berhasil menambahkan lagu "Type Girl" oleh "BLACKPINK" ke queue.
```

##### 2. **QUEUE PLAYLIST**

Command **QUEUE PLAYLIST** digunakan untuk menambahkan lagu yang ada dalam playlist ke dalam queue. Command ini menerima input dari id playlist yang ingin dimasukkan ke dalam queue.

```
ENTER COMMAND : QUEUE PLAYLIST;

Masukkan ID Playlist: 2;

Berhasil menambahkan playlist "K-pop Enjoyer" ke queue.
```

### 3. **QUEUE SWAP <x> <y>**

Command **QUEUE SWAP** digunakan untuk menukar lagu pada urutan ke x dan juga urutan ke y.

```
ENTER COMMAND : QUEUE SWAP 1 2;

Lagu "Kokoro wo Komete" berhasil ditukar dengan "Mirror"

ENTER COMMAND : QUEUE SWAP 80 2;

Lagu dengan urutan ke 80 tidak terdapat dalam queue!
```

### 4. **QUEUE REMOVE <id>**

Command **QUEUE REMOVE** digunakan untuk menghapus lagu dari queue. Command ini menerima input berupa urutan lagu (id) yang ingin dihapus dari queue.

```
ENTER COMMAND : QUEUE REMOVE 2;

Lagu "Kokoro wo Komete" oleh "Aoi Teshima" telah dihapus dari queue!

ENTER COMMAND : QUEUE REMOVE 40;

Lagu dengan urutan ke 40 tidak ada.
```

### 5. **QUEUE CLEAR**

Command **QUEUE CLEAR** digunakan untuk mengosongkan queue.

```
ENTER COMMAND : QUEUE CLEAR;

Queue berhasil dikosongkan.
```

## f. **SONG**

**SONG** merupakan command yang digunakan untuk navigasi lagu yang ada pada queue lagu saat ini. Terdapat 2 tipe navigasi yaitu **NEXT** dan **PREVIOUS**.

### 1. **SONG NEXT**

Command **SONG NEXT** digunakan untuk memutar lagu yang berada di dalam queue. Lagu yang sedang diputar kemudian ditambah ke dalam daftar riwayat putar lagu. Jika queue kosong, yang diputar adalah lagu yang sedang diputar.

ENTER COMMAND : **SONG NEXT;**

Memutar lagu selanjutnya  
"Hype Boy" oleh "New Jeans"

ENTER COMMAND : **SONG NEXT;**

Queue kosong, memutar kembali lagu  
"Mirror" oleh "Yasuda Rei"

## 2. SONG PREVIOUS

Command SONG PREVIOUS digunakan untuk memutar lagu yang terakhir kali diputar. Lagu yang sedang diputar kemudian ditambah ke dalam queue dengan urutan pertama. Jika daftar riwayat lagu kosong, yang diputar adalah lagu yang sedang diputar.

ENTER COMMAND : **SONG PREVIOUS;**

Memutar lagu sebelumnya  
"Hype Boy" oleh "New Jeans"

ENTER COMMAND : **SONG PREVIOUS;**

Riwayat lagu kosong, memutar kembali lagu  
"Mirror" oleh "Yasuda Rei"

## g. PLAYLIST

Command ini digunakan untuk melakukan basic command untuk playlist yaitu CREATE, ADD, SWAP, REMOVE dan DELETE.

### 1. PLAYLIST CREATE

Command PLAYLIST CREATE digunakan untuk membuat playlist baru dan ditambahkan pada daftar playlist pengguna. Keadaan awal playlist adalah kosong. Nama playlist dapat sama dengan playlist yang sudah ada.

ENTER COMMAND : **PLAYLIST CREATE;**

Masukkan nama playlist yang ingin dibuat : **K-pop Enjoyer;**

Playlist **K-pop Enjoyer** berhasil dibuat!  
Silakan masukkan lagu - lagu artis terkini kesayangan Anda!

ENTER COMMAND : **PLAYLIST CREATE;**

Masukkan nama playlist yang ingin dibuat : ;

Minimal terdapat 3 karakter selain *whitespace* dalam nama playlist.  
Silakan coba lagi.

### 2. PLAYLIST ADD

Command PLAYLIST ADD digunakan untuk menambahkan lagu pada suatu playlist yang telah ada sebelumnya pada daftar playlist pengguna. Pada

defaultnya, command ini hanya dapat menambahkan satu spesifik lagu atau semua lagu yang ada pada album kepada suatu existing playlist.

Apabila lagu atau lagu - lagu di dalam album yang ingin ditambahkan sudah ada di dalam suatu playlist pengguna maka lagu - lagu yang ditambahkan adalah yang belum ada di playlist pengguna.

Tampilkan pesan error apabila masukkan pengguna tidak valid pada setiap permintaan masukkan.

ENTER COMMAND : **PLAYLIST ADD SONG;**

Daftar Penyanyi :

1. New Jeans
2. BLACKPINK
3. Central Cee
4. Rich Brian
5. Denny Caknan
6. Via Vallen

Masukkan Nama Penyanyi yang dipilih : **BLACKPINK;**

Daftar Album oleh BLACKPINK :

1. BORN PINK
2. THE ALBUM
3. SQUARE ONE
4. SQUARE TWO

Masukkan Judul Album yang dipilih : **BORN PINK;**

Daftar Lagu Album BORN PINK oleh BLACKPINK :

1. Pink Venom
2. Shut Down
3. Typa Girl
4. Yeah Yeah Yeah
5. Hard to Love
6. The Happiest Girl
7. Tally
8. Ready for Love

Masukkan ID Lagu yang dipilih : **1;**

Daftar Playlist Pengguna :

1. RnB
2. Indo Songs
3. Dangdut Enjoyer
4. WIBU Songs
5. RANDOM SONGS
6. On Repeat Daily
7. For Night Ride
8. Romantic
9. Crying Session
10. K-Pop Enjoyer

Masukkan ID Playlist yang dipilih : **10;**

Lagu dengan judul "Pink Venom" pada album BORN PINK oleh penyanyi BLACKPINK berhasil ditambahkan ke dalam playlist K-Pop Enjoyer.

ENTER COMMAND : **PLAYLIST ADD ALBUM;**

Daftar Penyanyi :

1. New Jeans
2. BLACKPINK
3. Central Cee
4. Rich Brian
5. Denny Caknan
6. Via Vallen

Masukkan Nama Penyanyi yang dipilih : **BLACKPINK;**

Daftar Album oleh BLACKPINK :

1. BORN PINK
2. THE ALBUM
3. SQUARE ONE
4. SQUARE TWO

Masukkan Judul Album yang dipilih : **BORN PINK;**

Daftar Playlist Pengguna :

1. RnB
2. Indo Songs
3. Dangdut Enjoyer
4. WIBU Songs
5. RANDOM SONGS
6. On Repeat Daily
7. For Night Ride
8. Romantic
9. Crying Session
10. K-Pop Enjoyer

Masukkan ID Playlist yang dipilih : **10;**

Album dengan judul "BORN PINK" berhasil ditambahkan ke dalam pada playlist pengguna "K-Pop Enjoyer".

ENTER COMMAND : **PLAYLIST ADD SONG;**

Daftar Penyanyi :

1. New Jeans
2. BLACKPINK
3. Central Cee
4. Rich Brian
5. Denny Caknan
6. Via Vallen

Masukkan Nama Penyanyi yang dipilih : **BLACKPIN;**

Penyanyi BLACKPIN tidak ada dalam daftar. Silakan coba lagi.

ENTER COMMAND : **PLAYLIST ADD ALBUM;**

Daftar Penyanyi :

1. New Jeans
2. BLACKPINK
3. Central Cee
4. Rich Brian
5. Denny Caknan



6. Via Vallen

Masukkan Nama Penyanyi yang dipilih : **BLACKPINK;**

Daftar Album oleh BLACKPINK :

1. BORN PINK
2. THE ALBUM
3. SQUARE ONE
4. SQUARE TWO

Masukkan Judul Album yang dipilih : **BORN PI;**

Album BORN PI tidak ada dalam daftar. Silakan coba lagi.

### 3. PLAYLIST SWAP <id> <x> <y>

Command PLAYLIST SWAP digunakan untuk menukar lagu pada urutan ke x dan juga urutan ke y di playlist dengan urutan ke id.

ENTER COMMAND : **PLAYLIST SWAP 1 1 2;**

Berhasil menukar lagu dengan nama "Blue Bird" dengan "Silhouette" di playlist "Naruto Vibes 🍌"

ENTER COMMAND : **PLAYLIST SWAP 500 1 2;**

Tidak ada playlist dengan playlist ID 500

ENTER COMMAND : **PLAYLIST SWAP 1 900 2;**

Tidak ada lagu dengan urutan 900 di playlist "Galau"

### 4. PLAYLIST REMOVE <id> <n>

Command PLAYLIST REMOVE digunakan untuk menghapus lagu dengan urutan n pada playlist dengan index id.

ENTER COMMAND : **PLAYLIST REMOVE 2 4;**

Lagu "Ruang Rindu" oleh "Letto" telah dihapus dari playlist "Indo Songs"!

ENTER COMMAND : **PLAYLIST REMOVE 100 2;**

Tidak ada playlist dengan ID 100.

ENTER COMMAND : **PLAYLIST REMOVE 4 40;**

Tidak ada lagu dengan urutan 40 di playlist "WIBU Songs"!

### 5. PLAYLIST DELETE

Command PLAYLIST DELETE digunakan untuk melakukan penghapusan suatu existing playlist dalam daftar playlist pengguna. Tampilkan pesan error apabila masukkan pengguna tidak valid pada setiap permintaan masukkan.

ENTER COMMAND : **PLAYLIST DELETE;**

Daftar Playlist Pengguna :

1. RnB
2. Indo Songs
3. Dangdut Enjoyer
4. WIBU Songs
5. RANDOM SONGS
6. On Repeat Daily
7. For Night Ride
8. Romantic
9. Crying Session
10. K-Pop Enjoyer

Masukkan ID Playlist yang dipilih : 10

Playlist ID 10 dengan judul "K-pop Enjoyer" berhasil dihapus.

ENTER COMMAND : **PLAYLIST DELETE;**

Daftar Playlist Pengguna :

1. RnB
2. Indo Songs
3. Dangdut Enjoyer
4. WIBU Songs
5. RANDOM SONGS
6. On Repeat Daily
7. For Night Ride
8. Romantic
9. Crying Session
10. K-Pop Enjoyer

Masukkan ID Playlist yang dipilih : 11

Tidak ada playlist dengan ID 11 dalam daftar playlist pengguna. Silakan coba lagi.

## h. STATUS

STATUS merupakan command yang digunakan untuk menampilkan lagu yang sedang dimainkan beserta Queue song yang ada dan dari playlist mana lagu itu diputar.

ENTER COMMAND : **STATUS;**

// Tidak ada lagu yang diputar

Now Playing:

No songs have been played yet. Please search for a song to begin playback.

Queue:

Your queue is empty.

ENTER COMMAND : **STATUS;**

// Ada lagu, tapi queue kosong

Now Playing:

Coldplay - Up&Up - A Head Full of Dreams

Queue:

<p>Your queue is empty.</p>
<pre> ENTER COMMAND : <b>STATUS</b>; // Ada lagu dan ada queue  Now Playing: Coldplay - Up&amp;Up - A Head Full of Dreams  Queue: 1. Arctic Monkeys - R U Mine? - AM 2. BLACKPINK - WHISTLE - SQUARE ONE </pre>
<pre> ENTER COMMAND : <b>STATUS</b>; // Ada lagu, queue, dan playlist  // Hanya ditampilkan ketika melakukan playlist. Tidak ditampilkan jika terdapat lagu selain dari playlist Current Playlist: Copium  Now Playing: Coldplay - Up&amp;Up - A Head Full of Dreams  Queue: 1. Arctic Monkeys - R U Mine? - AM 2. BLACKPINK - WHISTLE - SQUARE ONE </pre>

#### i. **SAVE <filename>**

SAVE merupakan command yang digunakan untuk menyimpan state aplikasi terbaru ke dalam suatu file. Command SAVE memiliki satu argumen yang merepresentasikan nama file yang akan disimpan. Penyimpanan dilakukan pada folder tertentu, misal folder save.

<pre> ENTER COMMAND : <b>SAVE savefile.txt</b>;  Save file berhasil disimpan. // File disimpan pada /save/savefile.txt </pre>
---

#### j. **QUIT**

QUIT merupakan command yang digunakan untuk keluar dari aplikasi WayangWave.

<pre> ENTER COMMAND : <b>QUIT</b>;  Apakah kamu ingin menyimpan data sesi sekarang? <b>N</b>;  Kamu keluar dari WayangWave. Dadah ^_^/ </pre>
<pre> ENTER COMMAND : <b>QUIT</b>;  Apakah kamu ingin menyimpan data sesi sekarang? <b>Y</b>; // memanggil save </pre>

#### k. **HELP**

<b>STEI- ITB</b>	<b>IF2111-TB-02-01</b>	<b>Halaman 43 dari 48 halaman</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

HELP merupakan command yang digunakan menampilkan daftar command yang mungkin untuk dieksekusi dengan deskripsinya. Penjelasan dari deskripsi dibebaskan selama masih mendeskripsikan command sesuai dengan spek.

```
ENTER COMMAND : HELP; // ketika perintah dipanggil sebelum memasuki sesi
```

```
====[ Menu Help WayangWave ]=====
```

1. START -> Untuk masuk sesi baru
2. LOAD -> Untuk memulai sesi berdasarkan file konfigurasi

```
ENTER COMMAND : HELP; // ketika perintah dipanggil setelah memasuki sesi
```

```
====[ Menu Help WayangWave ]=====
```

1. LIST -> Untuk menampilkan daftar lagu, playlist...
2. PLAY -> Untuk memulai suatu lagu
- ...
9. SAVE -> Untuk menyimpan state ke dalam file
10. QUIT -> Untuk keluar dari sesi

### I. <Invalid Command>

Command-command selain yang disebutkan di atas dinyatakan akan tidak valid dan hanya akan mengeluarkan teks error.

```
ENTER COMMAND : TUBES ALSTRUKDAT;
```

```
Command tidak diketahui!
```

```
// Mengeksekusi LOAD sesudah memasuki sesi
```

```
ENTER COMMAND : LOAD filekonfigurasi.txt;
```

```
Command tidak bisa dieksekusi!
```

```
// Mengeksekusi LIST sebelum memasuki sesi
```

```
ENTER COMMAND : LIST DEFAULT;
```


```
Command tidak bisa dieksekusi!
```


## 9.2 Notulen Rapat

**Form Asistensi Tugas Besar  
IF2111/Algoritma dan Struktur Data STI  
Sem. 1 2023/2024**

No. Kelompok/Kelas : 01/02  
Nama Kelompok : brot  
Anggota Kelompok (Nama/NIM) : 1. Michael Jeremi Bungaran S / 18221136  
2. Muhammad Daffa Ikram / 18222020  
3. Christoper Daniel / 18222034  
4. Yovanka Sandrina Maharaja / 18222094  
5. Ervina Limka / 18222100  
  
Asisten Pembimbing : Rifqi Naufal Abdjul

Asistensi 1	
Tanggal	Rabu, 1 November 2023
Tempat	Google Meet
Anggota yang hadir : 1. Muhammad Daffa Ikram 2. Christoper Daniel 3. Yovanka Sandrina Maharaja 4. Ervina Limka	Catatan Asistensi : <ul style="list-style-type: none"><li>- Load and Save sebaiknya pembagian tugas ke 1 orang saja (hanya susah di pendefinisian struktur datanya).</li><li>- Play playlist sama add playlist to queue sebenarnya serupa.</li><li>- Load dan save konfigurasi, diparsing katanya dengan mesin kata.</li><li>- Tentang Linked List :</li><li>- Di setiap playlist bias disimpan lagu, bisa menambah lagu dan menambah album. Kenapa pake linked list? Karena ada operasi swap dan remove di spesifik angka. Kalau di array/queue/stack akan kurang sesuai karena ada rearrangement. Misal ada 5 lagu di playlist, mau tidak mau yang sampingnya bakal digeser ke depannya. Makanya dibuat ke linked list, tanpa tahu panjangnya ada berapa, bisa tahu dan menentukan swap dan remove.</li><li>- Lalu untuk penyimpanan struktur datanya?</li><li>- Tentukan entitasnya apa aja, yaitu playlist, lagu, history, penyanyi, album, antrian lagu.</li><li>- Contoh implementasi : List statis of penyanyi struct Penyanyi { nama: string album: Map &lt;key : string (nama album), value : set&lt;string&gt;&gt; } - Untuk penyimpanan struktur data, lebih baik didiskusikan bersamaan agar disamakan dengan function masing-masing</li></ul>

	<b>Tanda Tangan Asisten :</b> 
--	--

Asistensi 2	
<b>Tanggal</b>	Rabu, 15 November 2023
<b>Tempat</b>	Google Meet
<b>Anggota yang hadir :</b> <ol style="list-style-type: none"> <li>1. Muhammad Daffa Ikram</li> <li>2. Christoper Daniel</li> <li>3. Ervina Limka</li> </ol>	<b>Catatan Asistensi</b> <ul style="list-style-type: none"> <li>- Setelah load selesai bisa selanjutnya bisa dilanjutkan secara paralel</li> <li>- Untuk struct tipe data, dibuat dengan logika bahwa penyanyi mempunyai sebuah album dan di dalam sebuah album terdapat daftar song/lagu</li> <li>- Disarankan untuk mengerjakan lebih cepat dan membuat milestone-milestone kecil sebagai target penyelesaian</li> </ul>
	<b>Tanda Tangan Asisten :</b> 

### 9.3 Log Activity Anggota Kelompok

Nama Anggota	Log Activity
Muhammad Daffa Ikram	17/11/2023 : Menambah invalid command untuk parameter 17/11/2023: Menambah main 17/11/2023: Menambah command start 17/11/2023: Menambah command queue swap, clear, dan remove 18/11/2023: Menambah perubahan minor pada load 17/11/2023: Memperbaiki error pada console.c dan console.h 19/11/2023: Menambah perubahan sedikit

	<p>pada circular_queue</p> <p>21/11/2023: Membuat command save</p> <p>22/11/2023: Mengajukan saran perubahan pada addsong di ADT playlist</p>
Christoper Daniel	<p>30/10/2023 : Pembagian tugas</p> <p>11/11/2023 : Memasukkan beberapa ADT</p> <p>17/11/2023 : Membuat ADT untuk playlist</p> <p>18/11/2023 : Mengubah ADT untuk playlist</p> <p>18/11/2023 : Melakukan perubahan minor untuk ADT playlist</p> <p>18/11/2023 : Membuat ADT untuk daftar playlist</p> <p>19/11/2023 : Membuat command untuk bagian playlist (create, swap, delete, remove)</p> <p>20/11/2023 : Membuat command untuk playlist (add song, add album)</p> <p>21/11/2023 : Membuat driver untuk beberapa ADT (set, penyanyi, map, list playlist)</p> <p>24/22/2023 : Menambah invalid command untuk parameter</p>
Yovanka Sandrina Maharaja	<p>30/10/2023 : Pembagian tugas</p> <p>03/11/2023 : Membuat <i>repository</i> Github dan menambahkan beberapa ADT (boolean, array, string, word, mesin kata, dan mesin karakter).</p> <p>14/11/2023 : Menambahkan file load, album, penyanyi, dan set untuk command load</p> <p>17/11/2023 : Menambahkan driver album dan memperbaiki load</p> <p>19/11/2023 : Memperbaiki ADT playlist, menambahkan file main, mengupdate load history, queue, stack, dan playlist, command listdefault, listplaylist, queueplaylist, console, dan debug main</p> <p>20/11/2023 : Memperbaiki command createplaylist dan debug main</p> <p>22/11/2023 : Memperbaiki ADT playlist menjadi linked list</p> <p>23/11/2023 : Debug main</p> <p>24/22/2023 : Menambah invalid command untuk parameter</p>
Ervina Limka	<p>30/10/2023 : Pembagian tugas</p>

	<p>18/11/2023 : Membuat ADT stackchar untuk history, membuat command song next, status dan queue playlist</p> <p>19/11/2023 : Membuat command song previous, list playlist dan memperbaiki beberapa ADT, memperbaiki queue playlist</p> <p>20/11/2023 : Membuat command play playlist</p> <p>21/11/2023 : Membuat driver ADT stackchar dan current</p> <p>21/11/2023 : Memperbaiki song previous</p> <p>24/22/2023 : Menambah invalid command untuk parameter</p>
--	---