

LAPORAN TUGAS BESAR

IF2110 Algoritma dan Struktur Data

BURBIR



Dipersiapkan oleh:

Kelompok I / K02

Nyoman Ganadipa Narayana 13522066

Albert 13522081


Fedrianz Dharma 13522090

William Glory Henderson 13522113

Indraswara Galih Jayanegara 13522119

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		<i>IF2110-TB-I-K02</i>		<i>89</i>
		<i>Revisi</i>	<i>0</i>	<i>24 November 2023</i>

Daftar Isi

1	Ringkasan	4
2	Penjelasan Tambahan Spesifikasi Tugas	6
2.1	Tagar	6
2.2	Kelompok Teman	6
3	Struktur Data (ADT)	7
3.1	ADT List Dengan Array Statik	7
3.2	ADT Matriks	8
3.3	ADT Datetime	9
3.4	ADT List Dinamik	10
3.5	ADT Mesin Karakter	11
3.6	ADT Mesin Kata	12
3.7	ADT Stack	13
3.8	ADT Priority Queue	14
3.9	ADT List Linier	15
3.10	ADT Tree	17
3.11	ADT Graf dengan Adjacency Matrix	18
3.12	ADT Tagar dengan hashmap	19
3.13	ADT DisjointSetUnion (Kelompok Teman)	20
4	Program Utama	22
5	Algoritma-Algoritma Menarik	27
5.1	Algoritma DFS (Depth First Search) pada Fitur Balasan	27
5.2	Hashing	28
6	Data Test	29
6.1	Inisialisasi	29
6.2	Perintah	29
6.3	Pengguna	29
6.4	Profil	33
6.5	Teman	37
6.6	Permintaan Pertemanan	38
6.7	Kicauan	40
6.8	Balasan	44
6.9	Draf Kicauan	50
6.10	Utas	55

6.11 Simpan dan Muat	58
6.12 TAGAR	73
6.13 Kelompok Teman	75
7 Test Script	76
8 Pembagian Kerja dalam Kelompok	85
9 Lampiran	86
9.1 Deskripsi Tugas Besar 1	86
9.2 Notulen Rapat	86
9.3 Log Activity Anggota Kelompok	88

STEI- ITB	IF2110-TB-I-K02	Halaman 4 dari 89 halaman	
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.			

1 Ringkasan

Dalam tugas besar IF2110 Algoritma dan Struktur Data, kami ditugaskan untuk mengembangkan program bernama "burbir", sebuah aplikasi berbasis Command Line Interface (CLI) dengan fitur mirip media sosial X atau Twitter. Program ini dirancang untuk memanfaatkan konsep algoritma dan struktur data yang telah diajarkan sebelumnya dalam mata kuliah Algoritma dan Struktur Data. Implementasi program ini melibatkan pembuatan fungsi-fungsi sesuai spesifikasi dari tugas besar, dengan fokus pada fitur-fitur seperti tweet, like, reply, dan sebagainya. Bahasa pemrograman yang digunakan adalah C, dengan pemanfaatan Abstract Data Type (ADT) yang telah dipelajari dan dimodifikasi sesuai kebutuhan dari fitur-fitur sesuai spesifikasi.

Laporan ini mencakup ringkasan, penjelasan spesifikasi tugas besar, penggunaan struktur data (ADT), program utama, serta data dan skrip yang digunakan. Dengan menjalankan program, pengguna dapat memulai dari awal atau melanjutkan dari progress sebelumnya menggunakan perintah MUAT. Aplikasi ini memberikan fleksibilitas kepada pengguna untuk melakukan berbagai aktivitas seperti di media sosial X, namun dalam format CLI. Pengguna dapat mengakhiri program dengan perintah TUTUP_PROGRAM.

Pada akhirnya, laporan ini juga mencakup pembagian kerja dalam kelompok, serta lampiran yang berisi deskripsi tugas besar, notulen rapat, log aktivitas kelompok, dan lampiran tambahan. Dengan kompilasi dan eksekusi kode, pengguna dapat mengakses dan mengelola aktivitasnya dalam "burbir" sesuai dengan kebutuhan dan preferensinya.

STEI- ITB	IF2110-TB-I-K02	Halaman 5 dari 89 halaman	
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.			

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 Tagar

Tagar merupakan fitur tambahan dari Kicauan. Pengguna dapat memberikan kicauan suatu tagar, tagar ini nantinya akan dicari menggunakan fitur cari tagar. Dengan adanya tagar ini maka pencarian kicauan akan lebih efisien. Tagar dan kicauan disimpan sebagai key dan value di hashmap. Kami memberikan kapasitas dari hashmap tagar ini sebesar 1000 indeks. Tagar bersifat case sensitive. Tagar hanya bisa memiliki maksimal 280 karakter. Tagar tidak akan valid jika kosong ataupun hanya berisi spasi.

2.2 Kelompok Teman

Kelompok teman merupakan fitur tambahan dari fitur teman. Dengan fitur ini, setiap pengguna yang dapat melihat memiliki *circle* dengan siapa saja. Tanpa digunakannya BFS ataupun DFS, setiap kali query 'KELOMPOK_TEMAN' akan lebih efisien dengan menggunakan DSU.

STEI- ITB	IF2110-TB-I-K02	Halaman 6 dari 89 halaman	
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.			

3 Struktur Data (ADT)

3.1 ADT List Dengan Array Statik

- Sketsa struktur data

```
/* Kamus Umum */
#define CAPACITY_ListUser 20
/* Kapasitas penyimpanan */
#define IDX_UNDEF -1

/* Definisi elemen dan koleksi objek */
typedef int IdxType;
typedef struct {
    User contents[CAPACITY_ListUser]; /* memori tempat penyimpan elemen (container) */
    int length;
} ListUser;
```

Selain itu User atau content dari List dengan struktur data array statik ini sendiri adalah ADT user yang sketsa datanya sebagai berikut.

```
#define USERCAP 20;
#define ID_UNDEF -1;

typedef struct user {
    int ID;
    int friendCount;
    String name;
    String password;
    Profile profile;
    RequestQueue friendRequest;
    DrafKicauDin drafKicau;
} User;
```

- Alasan Pemilihan :
Pada List dengan struktur data array berkait kami menggunakannya untuk menyelesaikan persoalan dari listuser, mengapa demikian, karena listuser memiliki kapasitas yang terbatas dan sudah diketahui yakni, sebanyak 20 user sehingga list statik adalah yang paling cocok untuk menyelesaikan permasalahan tersebut.
- Persoalan yang dapat diselesaikan :
ADT User ini digunakan untuk menyelesaikan fitur fitur pengguna dan profil seperti lihat_profil, ubah_profil dll. Dengan menggunakan ADT ini kita dapat menampilkan dan mengubah informasi user dengan lebih gampang dan cepat.
- Implementasi sebagai ADT
Nama file : ListUser.h, ListUser.c pada lib/Types.

STEI- ITB	IF2110-TB-I-K02	Halaman 7 dari 89 halaman	
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.			

3.2 ADT Matriks

- Struktur data

```
typedef struct photo {
    CharMatrix symbols;
    CharMatrix color;
} Photo;

#define SYMMATRIX(u) (u).symbols
#define COLORMATRIX(u) (u).color
```

- Alasan Pemilihan :
Kami memilih matriks sebagai penampung data dari simbol dan warna pada foto sebab foto berbentuk dua dimensi, sehingga struktur yang cocok adalah array 2 dimensi, selain itu, hal ini juga mendukung *readability* program.
- Persoalan yang dapat diselesaikan:
Persoalan yang diselesaikan dengan memilih array 2 dimensi sebagai penampung informasi adalah kenyamanan dan kemudahan dalam *men-develop* serta *readability* program.
- Implementasi sebagai ADT
Nama file : photo.c pada lib/Types.

3.3 ADT Datetime

- Struktur Datetime

STEI- ITB	IF2110-TB-I-K02	Halaman 8 dari 89 halaman	
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.			


```
typedef struct{
    int Day;
    int Month;
    int Year;
    int Hour;
    int Minute;
    int Second;
} DATETIME;
```

- Alasan Pemilihan :
ADT ini dipilih karena untuk memenuhi fitur untuk menyimpan hari, bulan, tahun, jam, menit dan detik. Kami mengambil hari, bulan, tahun, jam, menit, detik. Sesuai dengan ketentuan yang ada di spesifikasi kami menggunakan integer agar lebih efisien sehingga saat dibutuhkan kami bisa mengubahnya ke dalam string.
- Persoalan yang dapat diselesaikan :
ADT ini digunakan untuk menyelesaikan persoalan di kicauan, utas, draf utas, balasan yang membutuhkan waktu. Sehingga dengan menggunakan ADT ini kita dapat menyimpan format waktu yang diperlukan. ADT Datetime digunakan untuk menyelesaikan masalah yang berhubungan dengan waktu seperti kicauan dan draf yang mana mereka akan membutuhkan waktu dimana mereka dieksekusi dengan menggunakan waktu lokal dari user yang menggunakan.
- Implementasi sebagai ADT
Nama file : datetime.h, datetime.c

3.4 ADT List Dinamik

- Sketsa Struktur data:

STEI- ITB	IF2110-TB-I-K02	Halaman 9 dari 89 halaman	
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.			

```

/* Definisi elemen dan koleksi objek */
typedef Kicauan KicauanType; /* type elemen list */
typedef struct
{
    KicauanType *buffer; /* memori tempat penyimpan elemen (container) */
    int nEff;             /* >=0, banyaknya elemen efektif */
    int capacity;         /* ukuran list */
} ListKicau ;
/* Indeks yang digunakan [0..capacity-1] */

```

- Alasan Pemilihan :

Dalam implementasi ADT Kicauan, kami memilih menggunakan List Dinamik untuk memberikan fleksibilitas tanpa batas bagi setiap pengguna. Keputusan ini diambil karena setiap pengguna tidak dibatasi oleh jumlah maksimal kicauan yang dapat mereka buat. List Dinamik memberikan solusi yang optimal dengan memberikan kemampuan untuk membuat ruang penyimpanan kicauan sesuai kebutuhan, tanpa batasan yang kaku.

Dengan pendekatan ini, kita dapat dengan mudah menyesuaikan kapasitas penyimpanan kicauan ketika diperlukan, memastikan bahwa pengguna tidak terbatas dalam berbagi pemikiran atau informasi melalui kicauan. Selain itu, pengguna secara bersamaan menggunakan List yang sama, memfasilitasi pengelolaan dan akses kicauan antar pengguna secara terpadu.

Digunakan ADT list dinamik sebagai data struktur untuk Kicau dibandingkan list dengan implementasi lainnya seperti list linier, stack, queue adalah karena pada fitur utama dari program ini adalah kicau, sehingga hampir setiap user hampir dapat dipastikan selalu membuat kicauan, dengan mengimplementasi list dinamik, kita dapat mengoptimalkan kompleksitas *time* serta *space*, yaitu $O(1)$ (rata-rata) saat pembuatan setiap kicau.

- Persoalan yang dapat diselesaikan :

Dengan menggunakan List Dinamik, kita dapat dengan mudah menangani situasi di mana kapasitas list kicauan perlu ditingkatkan seiring dengan penambahan kicauan pengguna. Fleksibilitas ini memungkinkan penyimpanan yang efisien dan dinamis, memastikan bahwa user dapat membuat sebanyak mungkin kicauan tanpa khawatir tentang batasan kapasitas.

- Implementasi sebagai ADT

Nama file : listKicauan.h, listKicauan.c

STEI- ITB	IF2110-TB-I-K02	Halaman 10 dari 89 halaman	
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.			

3.5 ADT Mesin Karakter

Disini ada dua jenis ADT Mesin Karakter yang kami gunakan ADT-nya. Ada jenis yang digunakan untuk membaca input dari pengguna pada program dan ada jenis yang digunakan untuk membaca file .config untuk memuat data ke program. Struktur datanya sendiri tidak seperti ADT lain di mesin karakter menggunakan extern atau melakukan globalisasi variabel

- Struktur data

```
/* State Mesin */
extern char currentChar_WordMachine;
extern boolean EOP_WordMachine;
```

- Struktur data mesin karakter file

```
/* State Mesin */
extern char currentCharFile;
extern boolean EOP_FILE;
extern boolean EndFile;
extern int retvalfile;
extern FILE *pitafile;
```

- Alasan Pemilihan :
Pada spesifikasi Tugas Besar IF2110 ini mewajibkan untuk membaca file dan input dari pengguna menggunakan ADT Mesin Kata. ADT Mesin Kata akan memanfaatkan ADT Mesin Karakter sehingga ADT Mesin Karakter dipilih untuk digunakan.
- Persoalan yang dapat diselesaikan :
Dengan menggunakan ADT Mesin Karakter, kita dapat mengakuisisi tiap character yang dimasukkan oleh pengguna program sebagai perintah. Selain itu, ADT Mesin Karakter ini juga digunakan oleh ADT Mesin Kata.
Dengan menggunakan ADT Mesin Karakter File, kita dapat mengakuisisi tiap character yang ada pada saat pembacaan file. Selain itu, ADT Mesin Karakter File juga digunakan oleh ADT Mesin Kata File.
- Implementasi sebagai ADT
Nama file : charmachine.c dan charmachinefile.c

3.6 ADT Mesin Kata

Mesin kata digunakan untuk menyelesaikan persoalan membaca input atau perintah dari pengguna pada program utama

STEI- ITB	IF2110-TB-I-K02	Halaman 11 dari 89 halaman	
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.			

- Struktur data

```
#define NMax 50
#define BLANK ' '

typedef struct
{
    char TabWord[NMax]; /* container penyimpan kata, indeks yang dipakai [0..NMax-1] */
    int Length;
} Word;
```

Struktur data dari mesin kata ini memanfaatkan ADT mesin karakter file

- Struktur data mesin kata file

```
/* State Mesin Word */
extern String currentWordFile;
```

```
typedef struct string
{
    char buffer[STRCAP];
    int maxLength;
} String;
```

- Alasan Pemilihan :

Pada spesifikasi Tugas Besar IF2110 ini mewajibkan untuk membaca file dan input dari pengguna menggunakan ADT Mesin Kata. Dengan menggunakan ADT Mesin Kata, kita dapat dengan mudah membaca suatu perintah dengan sebuah MARK sebagai tanda akhir perintah. Selain itu dengan menggunakan ADT Mesin Kata File, kita dapat dengan mudah membaca tiap masukkan yang berasal dari *file config*. *File config* dapat dibaca per *line*, per kata, atau membaca *file photo* dengan menggunakan ADT Mesin Kata File.

- Persoalan yang dapat diselesaikan :

Dengan menggunakan ADT Mesin Kata, kita dapat mengakuisisi perintah yang diberikan oleh pengguna dengan titik koma (;) sebagai MARK. ADT Mesin Kata ini memanfaatkan ADT Mesin Karakter untuk mengakuisisi tiap karakter dan dijadikan satu dengan dimasukkan ke dalam tipe bentukan Word.

Dengan menggunakan ADT Mesin Kata File, kita dapat mengakuisisi *input* yang berasal dari file *config*. ADT Mesin Kata File mempunyai 3 metode akuisisi, yaitu akuisisi satu line, akuisisi kata dengan *whitespace* sebagai MARK, dan akuisisi untuk *profile photo*. ADT Mesin Kata File memanfaatkan ADT Mesin Karakter File untuk mengakuisisi tiap karakter yang ada pada *file* dan dijadikan satu dengan dimasukkan ke dalam tipe bentukan String.

STEI- ITB	IF2110-TB-I-K02	Halaman 12 dari 89 halaman	
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.			

- Implementasi sebagai ADT
Nama file : wordmachine.c dan wordmachinefile.c

3.7 ADT Stack

- Sketsa struktur data

```
typedef struct nodeDraf *AddressDraf;
typedef struct nodeDraf {
    infodraf kicau;
    AddressDraf next;
} NodeDraf;

typedef struct {
    AddressDraf drafTop;
} DrafKicauDin;
```

- Struktur datanya kurang lebih seperti ini, infodrafnya menggunakan ADT Kicauan, dan AddressDraf ini sebuah pointer untuk mengakses elemen selanjutnya. drafTop sendiri adalah untuk selektor untuk mengambil elemen paling atas dari Draf.
- Alasan Pemilihan :
ADT Stack digunakan pada Draf Kicauan. Digunakan pada Draf karena saat user mengakses atau melihat draf maka yang terlihat adalah draf yang paling terbaru bukan yang terlama ini menandakan draf tersebut menggunakan sistem FIFO yang sama dengan stack. Namun, stack yang kami gunakan disini adalah stack dinamik karena jika kami menggunakan Stack yang bersifat statik maka kami harus mengatur kapasitas maksimal dari stack tersebut padahal seorang user belum tentu memiliki stack hanya segitu. Sehingga pemilihan Stack dinamik menurut kami adalah yang paling tepat. EType dari draf sendiri menggunakan ADT Kicauan karena EType dari draf memiliki kesamaan dengan Kicauan sehingga draf kami langsung menggunakan ADT Kicauan yang sudah dibuat.
- Persoalan yang dapat diselesaikan :
ADT Stack membantu mengatasi berbagai permasalahan terkait manajemen draf kicauan dengan mengimplementasikan sistem FIFO (First In, First Out). Dengan penerapan konsep FIFO, pengguna dapat dengan mudah dan cepat mengakses draf terbaru mereka. Keunggulan utama dari sistem ini adalah memungkinkan pengguna untuk melihat dan

STEI- ITB	IF2110-TB-I-K02	Halaman 13 dari 89 halaman	
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.			

mengelola kicauan yang paling baru dalam antrian. Dalam konteks ini, pengguna dapat dengan lancar mengeluarkan kicauan terbaru mereka tanpa harus mencari-cari di antara banyak draf. ADT Stack juga memanfaatkan struktur dinamis, memungkinkan pengguna untuk membuat sebanyak mungkin draf kicauan sesuai kebutuhan mereka. Fleksibilitas dari struktur Stack dinamis memastikan bahwa kapasitas draf dapat diperluas secara otomatis ketika dibutuhkan. Dengan implementasi Stack dinamis, pengguna memiliki kebebasan untuk membuat dan mengelola draf sebanyak mungkin tanpa khawatir tentang batasan kapasitas

- Implementasi sebagai ADT
Nama file : DraFDinKicauan.h DraFDinKicauan.c

3.8 ADT Priority Queue

- Sketsa struktur data :

```
#define IDX_UNDEF -1
#define CAPACITY_REQQUEUE 20
/* Konstanta untuk mendefinisikan address tak terdefinisi */

/* Definisi elemen dan address */
typedef struct {
    int friendCount;
    int userID;
} Friend;
typedef struct {
    Friend F[CAPACITY_REQQUEUE];
    int Head;
    int Tail;    /* Max elemen queue */
} RequestQueue;
```

Struktur datanya memanfaatkan ADT PrioQueue. Di sini ada struktur friend yang berisi jumlah teman dan id. Untuk struktur priority queue sendiri bersifat statik dan circular. Terdapat Head untuk mengakses elemen pertama dan Tail untuk mengakses elemen terakhir.

- Alasan Pemilihan :
Kami memilih menggunakan priority queue untuk mengatasi masalah pada permintaan pertemanan karena diperlukan adanya pengurutan permintaan pertemanan sesuai dengan banyaknya pengikut atau jumlah teman. Kami membuatnya statis dengan kapasitas 20 karena jumlah maksimal dari user 20 sehingga tidak mungkin melebihi kapasitas. Kami

STEI- ITB	IF2110-TB-I-K02	Halaman 14 dari 89 halaman	
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.			

juga memilih circular agar pergeseran ketika enqueue dan dequeue lebih efisien. Sehingga priority queue merupakan paling tepat karena adanya pengurutan permintaan pertemanan berdasarkan jumlah teman yang dimiliki oleh peminta pertemanan.

- Persoalan yang dapat diselesaikan :
Masalah antrian berdasarkan banyaknya teman yang dimiliki oleh peminta pertemanan. Dengan begitu, kita memaksimalkan potensi *priority queue* dalam kompleksitas waktu karena dengan menggunakan ADT ini, kita dapat mengambil permintaan pertemanan dari user dengan teman terbanyak dengan instan $O(1)$ tanpa skema searching apapun.
- Implementasi sebagai ADT
Nama file : Requestqueue.c, Requestqueue.h

3.9 ADT List Linier

- Sketsa struktur data

```
/* Definisi Node : */
typedef struct nodeUtas* AddressUtas;
typedef struct nodeUtas{
    String namaUser;    // Nama dari user yang melakukan utas
    String dateTime;    // dateTime Utas
    String teks;        // teks dari utas
    AddressUtas next;   // link untuk ke next dari linked list
} NodeUtas;
typedef AddressUtas listutas;
```

Struktur datanya seperti ini berisi namaUser, dateTime, teks yang semuanya merupakan String (ADT buatan kami sendiri). dateTime disini memanfaatkan ADT Datetime dan kami mengubahnya menjadi String

- Alasan Pemilihan :
Kami memilih menggunakan List Linier untuk mengatasi permasalahan pada Utas karena ketidakpastian akan batas maksimalnya. List Linier cocok karena mampu mengalokasikan tempat penyimpanan secara efisien, hanya saat diperlukan. Sifat dinamis dari List Linier memastikan penggunaan memori yang optimal. Lebih lanjut, kemampuan

STEI- ITB	IF2110-TB-I-K02	Halaman 15 dari 89 halaman	
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.			

akses hanya dimulai dari elemen pertama sangat sesuai dengan karakteristik Utas yang biasanya diakses secara sekuensial. Selain itu, kegunaan List Linier membantu memudahkan pengguna jika ingin menyambung atau menghapus Utas di antara elemen-elemen dalam list. Fleksibilitas ini memberikan kontrol yang lebih besar kepada pengguna untuk mengelola dan menyusun Utas sesuai kebutuhan mereka.

- Persoalan yang dapat diselesaikan :
Dengan menerapkan ADT List Linier, penanganan pencetakan Utas menjadi lebih efisien. Proses pencetakan dapat dilakukan dengan mudah dengan melanjutkan link dari List Linier, dimulai dari awal hingga mencapai akhirnya. Keberlanjutan link ini menciptakan aliran yang lancar dan terorganisir saat mencetak Utas, mengoptimalkan proses secara keseluruhan. Penggunaan List Linier juga mengatasi kendala ketika perlu menghapus atau menyambungkan Utas di berbagai posisi, baik di tengah-tengah atau di akhir. Selain itu, fitur dinamis dari List Linier memberikan kebebasan kepada pengguna untuk menambahkan Utas sebanyak yang mereka butuhkan. Kemampuan ini menciptakan lingkungan yang bebas batas, di mana pengguna dapat dengan mudah menyesuaikan jumlah dan konten Utas sesuai dengan keinginan mereka.
- Implementasi sebagai ADT
Nama file : listUtas.h , listUtas.c

3.10 ADT Tree

- Struktur data

STEI- ITB	IF2110-TB-I-K02	Halaman 16 dari 89 halaman	
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.			


```

typedef struct reply {
    int id;
    String body;
    int authorID;
    DATETIME time;
    boolean mainReply; // artinya dia attached to root atau ngga
} Reply;

typedef struct reply *ReplyAddress;

#define REPLYID(r) (r).id
#define BODY(r) (r).body
#define AUTHORID(r) (r).authorID
#define DTIME(r) (r).time
#define ISMAIN(r) (r).mainReply

typedef struct listreply{
    ReplyAddress *buffer;
    int neff;
    int capacity;
} ListReply;

#define CAPLR(lr) (lr).capacity
#define NEFFLR(lr) (lr).neff
#define ADDR(lr, i) (lr).buffer[i]

typedef struct replytree {
    ListDin *adjlists; // pointer to first list dinamis
    ListReply listreply;
    ListDin used;
    ListDin parent;
    int numReplyEff;
    int availableID;
    int maxReply;
    int availableIDX;
} ReplyTree;

```

- Alasan Pemilihan :
Dipilihnya ADT tree sebagai fondasi untuk fitur balasan ialah karena kemampuan dari tree dalam mengakses anak-anaknya (anak balasan) serta kedalaman dari anak balasan tersebut – seberapa jauh dari parent – dengan lebih mudah, dan tidak berantakan, karena pasti pemilihan ADT seperti stack dan queue bisa saja tetapi akan *strange*.
- Persoalan yang dapat diselesaikan :
Dengan dipilihnya tree sebagai fondasi, kita dapat menambahkan balasan dengan kompleksitas waktu, secara rata-rata, $O(1)$ sebab hanya menambahkan elemen pada *adjacency list* parent.

STEI- ITB	IF2110-TB-I-K02	Halaman 17 dari 89 halaman	
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.			

- Implementasi sebagai ADT
Nama file : ReplyTree.c ReplyTree.

3.11 ADT Graf dengan Adjacency Matrix

- Sketsa Struktur data

```
/**
 * Definisi Graf :
 * 1. numEdge menyatakan jumlah elemen
 * 2. numVertex menyatakan jumlah koneksi.
 * 3. adjMatrix menyatakan adjacency matrix.
 */
typedef int ElType;
typedef struct graf {
    int numEdges;
    int numVertices;
    BoolMatrix adjacencyMatrix;
} Graf;
```

```
typedef struct application {
    boolean isLoggedIn;
    int loginID;
    ListUser users; // User list dengan array statik
    Graf friendships;
    ListKicau listKicauan;
    int JumlahUtas;
    HashTable hashMap;
} Application;
```

Struktur datanya kurang lebih seperti ini. Seperti pada penjelasan di atas numEdge adalah jumlah elemen, numVertex adalah jumlah koneksi, adjMatrix adalah adjacency matrix.

- Alasan Pemilihan :
ADT ini kami gunakan untuk menyelesaikan masalah matriks pertemanan dimana setiap user yang berteman akan memiliki nilai pada matriks pertemanan. Melihat bahwa jumlah pengguna maksimal adalah sebesar 20, tentu *adjacency matrix* menjadi pilihan utama

STEI- ITB	IF2110-TB-I-K02	Halaman 18 dari 89 halaman	
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.			

pada program sebab memberi kelebihan seperti *lookup table* pertemanan antara prang kesatu dengan orang kedua.

- Persoalan yang dapat diselesaikan :
Pada program ini, hampir di setiap titik digunakan pengecekan apakah orang pertama dengan orang kedua sudah berteman atau belum. Hal ini tentu memaksimalkan ADT matrix dengan struktur adjacency matrix karena memberikan kelebihan *look up table* pertemanan sehingga setiap pengecekan pertemanan antara dua orang hanya diperlukan waktu $O(1)$.
- Implementasi sebagai ADT
Nama file : Salah satu properti dalam struct Application pada lib/Types/Application/Application.h

3.12 ADT Tagar dengan hashmap

- Sketsa Struktur data :

```
#define TABLE_SIZE 1000 // Ukuran Hashmap

typedef struct {
    String tagar;
    Kicauan *kicauan;
    int isOccupied;
} HashNode;

#define GET_TAGAR(node) ((node).tagar)
#define GET_KICAUAN(node) ((node).kicauan)
#define IS_OCCUPIED(node) ((node).isOccupied)
```

- Alasan Pemilihan :
ADT ini kami gunakan untuk menyelesaikan masalah tagar dimana pengguna dapat memasukkan tagar ke kicauan. Pengguna dapat mencari kicauan yang mereka ingin cari dengan menggunakan fungsi cari kicauan. Alasan pemilihan hashmap untuk tagar ini adalah agar pencarian kicauan dapat menjadi lebih efisien karena tagar akan diubah menjadi key, dan akan langsung mengakses hashmap tanpa perlu pencarian lagi.
- Persoalan yang dapat diselesaikan :
Persoalan yang diselesaikan dari ADT pencarian kicauan dapat menjadi lebih efisien karena tagar akan diubah menjadi key, dan akan langsung mengakses hashmap tanpa

STEI- ITB	IF2110-TB-I-K02	Halaman 19 dari 89 halaman	
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.			

perlu pencarian lagi. Pencarian kicauan dapat langsung dilakukan tanpa perlu looping seluruh kicauan yang ada.

- Implementasi sebagai ADT
Nama file : Kicauan.c, Kicauan.c, kicauan.c, kicauan.h
Salah satu properti dalam struct Application pada lib/Types/Application/Application.h

3.13 ADT DisjointSetUnion (Kelompok Teman)

Sketsa Struktur data :

```
typedef struct {  
    int arr [20];  
    int count;  
} DisjointSet;  
  
#define ELMT_SET(ds, i) (ds).arr[(i)]
```

- Alasan Pemilihan :
ADT ini kami gunakan untuk menyelesaikan masalah kelompok teman dimana pengguna dapat melihat kelompok teman. Pengguna dapat menampilkan kelompok teman yang isinya adalah teman langsung dan teman tidak langsung. Alasan pemilihan dsu karena sesuai spesifikasi bonus dan ADT dsu dapat mewakili bagaimana algoritma dfs atau bfs bekerja. Selain itu, dalam fitur ini dsu bekerja lebih efisien daripada algoritma dfs.
- Persoalan yang dapat diselesaikan :
Persoalan yang diselesaikan dari ADT ini adalah menentukan sebuah kelompok pertemanan yang isinya merupakan teman langsung dan teman tidak langsung. Contoh: jika a teman b maka disebut a disebut teman langsung dengan b. Kemudian b teman langsung dengan c. Maka a termasuk teman tidak langsung dengan c. Untuk hasil kelompok teman a adalah b dan c.
- Implementasi sebagai ADT
Nama file : dsu.c, dsu.h, teman.c, teman.h

STEI- ITB	IF2110-TB-I-K02	Halaman 20 dari 89 halaman	
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.			

4 Program Utama

Algoritma program utama terletak di `app/main.c`, sedangkan fitur-fiturnya sebagai komponen pendukung utama dari program ini terletak pada `lib/Types/Application/*`. Dengan masing-masing folder pada `lib/Types/Application` mewakili setiap spesifikasi program.

STEI- ITB	IF2110-TB-I-K02	Halaman 21 dari 89 halaman	
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.			

1. Program akan dimulai dari main.c. Fungsi yang akan dipanggil ketika pertama kali adalah AppInitialization. Fungsi ini akan membuat inisialisasi awal dari program memuat file-config yang ingin di muat.

```
int main()
{
    String command;
    createEmptyString(&command, 1000);

    Application app;
    START();

    // 1. Inisialisasi
    AppInitialization(&app);

    boolean finish = false;
    while (!finish) {
        // 2. Perintah
        printf("\n>> ");
        readString(&command, 1000);

        handleCommand(&app, command, &finish);
    }
    CLOSE();
    return 0;
}
```

2. Ketika muat file config sudah berhasil, program akan masuk ke dalam loop dimana program akan meminta perintah dari pengguna. Pada saat menerima perintah program akan mencari fungsi dari perintah tersebut dan menjalankannya. Program akan berhenti ketika pengguna memasukkan perintah TUTUP_PROGRAM. Pengguna dapat memilih untuk melakukan MASUK jika program telah menyimpan pengguna-pengguna yang valid, namun jika tidak, pengguna dapat melakukan DAFTAR untuk menambahkan jumlah pengguna yang terdapat pada list pengguna. Ketika pengguna sudah melakukan MASUK, maka fitur-fitur yang dapat diakses menjadi lebih banyak yaitu sebagai berikut.

STEI- ITB	IF2110-TB-I-K02	Halaman 22 dari 89 halaman	
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.			

```

void handleCommand(Application* app, String command, boolean *finish)
/**
 *
 */
{
    int numword = countWord(command);
    if (numword == 0) return;

    String firstWord;
    String rest;
    SplitIntoTwo(command, &firstWord, &rest);

    String secondWord;
    String thirdWord;

    if (numword >= 1) secondWord = getWordAt(command, 1);
    if (numword >= 2) thirdWord = getWordAt(command, 2);

    if (compareString(firstWord, "DAFTAR")){
        Daftar(app);
    } else if (compareString(firstWord, "MASUK")){
        Masuk(app);
    } else if (compareString(firstWord, "KELUAR")){
        Keluar(app);
    } else if (compareString(firstWord, "DEVTOOLS")) {
        /**
         * Command Developer Tools akan dihilangkan saat pengumpulan, ini hanya alat untuk debugging.
         */
        DevTools(*app);
    } else if (compareString(firstWord, "TUTUP_PROGRAM")) {
        TutupProgram(app, finish);
    } else if (compareString(firstWord, "GANTI_PROFIL")) {
        GantiProfil(app);
    } else if (compareString(firstWord, "LIHAT_PROFIL")) {
        if (numword < 2) {
            printf("\nPerintah yang valid: \"LIHAT_PROFIL <nama> ;\".\n");
            return;
        }
        LihatProfil(app, rest);
    } else if (compareString(firstWord, "ATUR_JENIS_AKUN")) {
        AturJenisAkun(app);
    } else if (compareString(firstWord, "UBAH_FOTO_PROFIL")) {
        UbahFotoProfil(app);
    } else if (compareString(firstWord, "DAFTAR_TEMAN")) {
        DaftarTeman(*app);
    } else if (compareString(firstWord, "HAPUS_TEMAN")) {
        HapusTeman(app);
    }
}

```

- **UBAH_FOTO_PROFIL**
Untuk mengubah foto profil dari suatu pengguna.
- **GANTI_PROFIL**
Untuk memodifikasi informasi yang disimpan pada profil pengguna. Contohnya bio, akun, nomor HP, Weton.

STEI- ITB	IF2110-TB-I-K02	Halaman 23 dari 89 halaman	
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.			

- **LIHAT_PROFIL [NAMAPENGGUNA]**
Untuk menampilkan informasi profil yang dimiliki oleh suatu pengguna
- **ATUR_JENIS_AKUN**
Untuk melakukan modifikasi jenis akun dari publik ke privat.
- **DAFTAR_TEMAN**
Untuk menampilkan daftar teman dari pengguna.
- **HAPUS_TEMAN**
Untuk melakukan penghapusan teman sesuai dengan masukan nama dari pengguna.
- **KICAU**
Untuk mengunggah sebuah kicauan. Kicauan ini mempunyai panjang maksimal 280 karakter, yang akan dimasukkan ke dalam sebuah list kicau, ID nya akan sesuai dari elemen efektif dari list kicau, *datetime* akan sesuai dengan waktu saat mengunggah, dan jumlah disukai dimulai dari nol. Pengguna juga dapat memasukkan tagar ke kicauan agar dapat dicari dengan mudah.
- **KICAUAN**
Untuk menampilkan list kicauan. Kicauan yang ditampilkan hanya kicauan yang merupakan milik pengguna dan teman dari pengguna.
- **SUKA_KICAUAN [IDKICAU]**
Untuk memberikan like kepada kicauan dengan id kicau yang dimasukkan.
- **UBAH_KICAUAN [IDKICAU]**
Untuk mengubah kicauan yang sudah pernah dimasukkan. Pengguna hanya bisa mengubah kicauan milik sendiri. Saat mengubah kicauan yang berubah hanyalah kicauan tersebut. Like, *datetime*, dan nama pengguna tidak akan pernah berubah.
- **BALAS [IDKICAU] [IDBALASAN]**
Untuk melakukan balasan pada id kicau tertentu dengan id balasan tertentu pada id kicau tersebut. Balasan hanya dapat dibalas jika balasan yang ingin dibalas dibuat oleh akun yang bersifat publik atau berteman dengan pengguna yang membalas.
- **BALASAN [IDKICAU]**
Untuk menampilkan seluruh balasan yang terdapat pada ID kicau tertentu.

STEI- ITB	IF2110-TB-I-K02	Halaman 24 dari 89 halaman	
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.			

- **HAPUS_BALASAN [IDKICAU] [IDBALASAN]**
Untuk melakukan penghapusan sebuah balasan yang spesifik pada ID kicau dan id balasan tertentu.
- **BUAT_DRAF**
Untuk melakukan pembuatan draf kicauan yang dapat diubah oleh pengguna lagi. Pengguna dapat melakukan SIMPAN untuk menyimpan draf tersebut.
- **LIHAT_DRAF**
- Untuk menampilkan draf terakhir yang dibuat oleh pengguna. HAPUS untuk jika ingin menghapus draf terakhir, TERBIT untuk melakukan pengubahan dari draf terakhir menjadi kicauan, dan KEMBALI untuk kembali ke menu sebelumnya tanpa melakukan aksi pada draf yang ditampilkan
- **DAFTAR_TEMAN**
Untuk menampilkan daftar teman yang dimiliki oleh pengguna.
- **TAMBAH_TEMAN**
Untuk menambah dan mengajukan pertemanan kepada pengguna lain yang belum berteman dengan pengguna saat ini. Daftar permintaan pertemanan dari pengguna yang diajukan akan selalu diurutkan sesuai kepopularitas pengguna saat ini.
- **HAPUS_TEMAN**
Untuk menghapus hubungan pertemanan antara pengguna dengan teman dari pengguna.
- **DAFTAR_PERMINTAAN_PERTEMANAN**
Untuk menampilkan daftar permintaan pertemanan dari pengguna yang melakukan perintah tersebut.
- **SETUJUI_PERTEMANAN**
Untuk menyetujui pertemanan yang berada pada daftar permintaan pertemanan. Setujui pertemanan akan hanya bisa menyetujui pertemanan di dalam daftar pertemanan yang paling atas.
- **UTAS**
Untuk membuat utas dari daftar kicauan yang sudah ada. Saat pesan ini dipanggil, kicauan utama akan dipanggil dan dijadikan kicauan sambungan dengan indeks 0. Pengguna akan diminta untuk memasukkan kicauan sambungan yang akan disambung dengan kicauan utama.
- **SAMBUNG_UTAS**

STEI- ITB	IF2110-TB-I-K02	Halaman 25 dari 89 halaman	
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.			

Untuk menyambung kicauan sambungan sesuai dengan ID Utas dan Indeks yang diinginkan.

- **HAPUS_UTAS**
Untuk menghapus kicauan sambungan sesuai dengan ID Utas dan Indeks yang diinginkan.
- **CETAK_UTAS**
Untuk mencetak kicauan utama dan kicauan sambungan dari ID utas yang dimasukkan oleh pengguna.
- **SIMPAN**
Untuk melakukan penyimpanan progress yang sudah ada saat ini ke dalam folder yang berisi file file config. Program akan meminta input dari user berupa nama dari folder yang akan digunakan untuk menyimpan progress saat ini. Jika folder belum ada maka program akan otomatis membuat folder terlebih dahulu lalu melakukan penyimpanan file-file config
- **MUAT**
Untuk memuat file-file config yang sudah ada di dalam folder yang dimasukkan oleh pengguna.
- **CARI_KICAUAN [TAGAR]**
Untuk mencari kicauan dengan tagar yang dimasukkan oleh pengguna.
- **KELOMPOK_TEMAN**
Untuk mencari *circle*-nya.
- **KELUAR**
Untuk keluar dari akun pengguna yang sedang dimasuki

STEI- ITB	IF2110-TB-I-K02	Halaman 26 dari 89 halaman	
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.			

5 Algoritma-Algoritma Menarik

5.1 Algoritma DFS (Depth First Search) pada Fitur Balasan

Algoritma DFS pada fitur balasan cukup menarik sebab dapat menyederhanakan masalah yang sebelumnya terlihat sulit dengan algoritma yang sederhana. Berikut adalah *pseudocode* dari bagaimana cara balasan itu ditampilkan menggunakan *DFS* secara sederhana.

```
procedure displayAllReply(input rt: ReplyTree)
procedure displayReply(input r: Reply)
procedure displayAllReply_Helper (input rt: ReplyTree, input currDepth)

{Realisasi prosedur}
procedure displayAllReply(input rt: ReplyTree)
{I.S. rt terdefinisi}
{F.S. menampilkan balasan dengan indentasi lebih menjorok untuk kedalaman yang lebih dalam.}
```

KAMUS LOKAL

length, i : integer
r : Reply

ALGORITMA

```
length <- NEFF(rt)
i traversal [0...length-1]
  r <- ELMT(rt, i)
  if (isAttachedToKicauan(r)) then
    displayAllReply_helper(rt, 0)
```

```
procedure displayAllReply_helper (input rt: ReplyTree, input currDepth, input r: Reply)
```

```
{Fungsi helper displayAllReply}
```

KAMUS LOKAL

length, i : integer

ALGORITMA

```
displayAllReply(r)
length <- NUMCHILD(r)
i traversal [0..length-1]
  displayAllReply_helper(rt, currDepth + 1, CHILD(r, 0))
```

STEI- ITB	IF2110-TB-I-K02	Halaman 27 dari 89 halaman	
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.			

5.2 Hashing

Algoritma hashing menarik karena kemampuannya mengubah data kompleks seperti string menjadi angka yang lebih sederhana dan pasti. Proses ini memungkinkan penanganan data secara efisien, khususnya dalam pencarian dan penyimpanan. Dalam kasus tagar kicauan, algoritma hashing mengonversi tagar yang berupa teks menjadi indeks numerik yang unik, memudahkan akses cepat ke data terkait. Secara keseluruhan, algoritma hashing adalah solusi yang menarik dan efektif untuk manajemen data dalam aplikasi, terutama ketika membutuhkan akses cepat dan efisien ke data yang berjumlah besar. Dengan kemampuannya mengubah data teks menjadi indeks numerik dan penanganan bentrokan yang efektif, hashing menjadi pilihan populer dalam pengembangan perangkat lunak.

function hashFunction (String tagar)

KAMUS LOKAL

hash : unsigned long long int

TABLE_SIZE : int

ALGORITMA

hash <- 0

i traversal [0...length-1]

hash <- (hash + tagar.buffer[i])*97 mod TABLE_SIZE

-> hash

STEI- ITB	IF2110-TB-I-K02	Halaman 28 dari 89 halaman	
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.			

6 Data Test

6.1 Inisialisasi

Inisialisasi dilakukan dengan memuat semua file config. Jika ada salah satu file config yang gagal dimuat, maka pada inisialisasi akan dilakukan tanpa adanya file config. Inisialisasi dilakukan pada saat program pertama kali dijalankan.

```
Silahkan masukan folder konfigurasi untuk dimuat: ;  
Folder tidak berhasil dimuat, jadi bikin baru
```

```
Silahkan masukan folder konfigurasi untuk dimuat: TESTING1;  
  
File konfigurasi berhasil dimuat! Selamat berkicau!
```

6.2 Perintah

Pada bagian ini, akan dilakukan uji coba untuk fitur-fitur yang terdapat di dalam perintah. Jika terdapat *typo* pada perintah, maka perintah tidak berhasil dan program mengeluarkan pesan kesalahan.

```
>> asasa;  
  
Tidak ada perintah asasa. Baca dokumentasi (Spesifikasi) untuk melihat seluruh perintah.
```

6.3 Pengguna

Pada *test* pertama, akan dilakukan uji coba untuk fitur-fitur yang terdapat di pengguna.

- Fitur yang dites:
 - DAFTAR
 - MASUK
 - KELUAR
 - TUTUP_PROGRAM

1. DAFTAR

Fitur DAFTAR bertujuan untuk mendaftarkan akun baru agar dapat menggunakan fitur-fitur BurBir. Perintah DAFTAR akan meminta *user* untuk memasukkan nama dan password.

- Hasil yang diberikan jika input nama dan password sesuai dengan ketentuan, yaitu maksimal 20 karakter.

STEI- ITB	IF2110-TB-I-K02	Halaman 29 dari 89 halaman	
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.			

- Jika *user* memasukkan perintah DAFTAR dalam keadaan sudah login, maka program akan meminta *user* untuk keluar/logout dari akun terlebih dahulu sebelum bisa melakukan DAFTAR.

```
>> DAFTAR;
```

```
Anda sudah punya akun!!! Keluar terlebih dahulu untuk membuat akun yang lain.
```

2. MASUK

- Jika *user* memasukkan nama yang telah terdaftar dan password yang benar untuk password tersebut, maka *user* akan berhasil masuk dengan akun tersebut.

```
>> MASUK;
```

```
Masukkan nama:  
gana friend;
```

```
Masukkan kata sandi:  
ganakeren;
```

```
Anda telah berhasil masuk dengan nama pengguna gana friend. Mari menjelajahi BurBir bersama Ande-Ande Lumut!
```

- Jika *user* mencoba masuk dengan nama yang tidak tersedia atau password yang tidak tepat, maka program akan mengeluarkan pesan kesalahan dan meminta input ulang.

```
>> MASUK;
```

```
Masukkan nama:  
adada;
```

```
Wah, nama yang Anda cari tidak ada. Masukkan nama lain!
```

```
Masukkan nama:  
gana;
```

```
Masukkan kata sandi:  
adadada;
```

```
Wah, kata sandi yang Anda masukkan belum tepat. Periksa kembali kata sandi Anda!
```

```
Masukkan kata sandi:  
gana;
```

```
Anda telah berhasil masuk dengan nama pengguna gana. Mari menjelajahi BurBir bersama Ande-Ande Lumut!
```

STEI- ITB	IF2110-TB-I-K02	Halaman 31 dari 89 halaman	
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.			

- Jika *user* dalam keadaan sudah login dan memasukkan perintah MASUK, program akan mengeluarkan pesan kesalahan dan meminta *user* untuk logout terlebih dahulu.

```
>> MASUK;
```

```
Wah Anda sudah masuk. Keluar dulu yuk!
```

3. KELUAR

- Jika *user* sudah melakukan login dan memasukkan perintah KELUAR, maka *user* akan keluar dari akun tersebut.

```
>> KELUAR;
```

```
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!
```

- Jika *user* belum melakukan login dan memasukkan perintah keluar, program akan mengeluarkan pesan kesalahan.

```
>> KELUAR;
```

```
Anda belum login! Masuk terlebih dahulu untuk menikmati layanan BurBir.
```

4. TUTUP_PROGRAM

- Jika *user* memasukkan perintah TUTUP_PROGRAM, maka *user* akan keluar dari program.

```
>> TUTUP_PROGRAM;
```

```
Anda telah keluar dari program BurBir. Sampai jumpa di penjelajahan berikutnya.
```

STEI- ITB	IF2110-TB-I-K02	Halaman 32 dari 89 halaman	
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.			

6.4 Profil

Pada *test* kedua, akan dilakukan uji coba untuk fitur-fitur yang terdapat di profil.

- Fitur yang dites:
 - LIHAT_PROFIL
 - UBAH_FOTO_PROFIL
 - GANTI_PROFIL
 - ATUR_JENIS_AKUN

1. LIHAT_PROFIL

- Jika *user* memasukkan perintah LIHAT_PROFIL tanpa memasukkan nama akun yang ingin dilihat, program akan mengeluarkan pesan cara menggunakan perintah LIHAT_PROFIL yang benar.

```
>> LIHAT_PROFIL;
```

```
Perintah yang valid: "LIHAT_PROFIL <nama> ;".
```

- Jika *user* memasukkan perintah LIHAT_PROFIL dalam keadaan belum login, program akan meminta *user* untuk login terlebih dahulu.

```
>> LIHAT_PROFIL a;
```

```
Anda belum login! Masuk terlebih dahulu untuk menikmati layanan BurBir.
```

- Jika *user* memasukkan perintah LIHAT_PROFIL dengan nama akun yang tidak tersedia, program akan memberikan pesan kesalahan.

```
>> LIHAT_PROFIL a;
```

```
Tidak ditemukan akun tersebut.
```

STEI- ITB	IF2110-TB-I-K02	Halaman 33 dari 89 halaman	
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.			

- Jika *user* memasukkan perintah LIHAT_PROFIL untuk akun yang diprivat, program akan memberikan pesan untuk mengikuti akun tersebut jika ingin melihat profilnya.

```
>> LIHAT_PROFIL wg;
```

Wah, akun wg diprivat nih. Ikuti dulu yuk untuk bisa melihat profil wg !

- Jika *user* sudah login dan memasukkan perintah LIHAT_PROFIL dengan nama akun yang benar, maka program akan menampilkan profil dari akun tersebut.

```
>> LIHAT_PROFIL wg;
```

```
| nama: wg
| Bio Akun:
| No HP:
| Weton:
```

```
Foto Profil:
```

```
*****
*****
*****
*****
*****
```

STEI- ITB	IF2110-TB-I-K02	Halaman 34 dari 89 halaman	
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.			

2. UBAH_FOTO_PROFIL

- VALID

```
Masukkan foto profil yang baru:
G 8 G 8 G 8 G 8 G 8
G 8 G 8 G 8 G 8 G 8
G 1 G 1 G 1 G 1 G 1
R 2 R 2 R 2 R 2 R 2
R 3 R 3 R 3 R 3 R 3;
Foto profil Anda sudah berhasil diganti!.
```

```
>> UBAH_FOTO_PROFIL;
Foto profil Anda saat ini adalah:
*****
*****
*****
*****
*****
```

```
Foto Profil:
88888
88888
11111
22222
33333
```

- Jika *user* belum melakukan login dan memanggil perintah UBAH_FOTO_PROFIL, maka program akan mengeluarkan pesan kesalahan.

```
>> UBAH_FOTO_PROFIL;

Anda belum login! Masuk terlebih dahulu untuk menikmati layanan BurBir.
```

3. GANTI_PROFIL

- Jika *user* belum melakukan login dan memanggil perintah UBAH_FOTO_PROFIL, maka program akan mengeluarkan pesan kesalahan.

```
>> GANTI_PROFIL;

Anda belum login! Masuk terlebih dahulu untuk menikmati layanan BurBir.
```

- Setelah *user* login, *user* dapat mengubah profilnya dengan memanggil perintah GANTI_PROFIL.

STEI- ITB	IF2110-TB-I-K02	Halaman 35 dari 89 halaman	
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.			

```

>> GANTI_PROFIL;

| nama: gana
| Bio Akun:
| No HP:
| Weton:

Masukkan Bio Akun:
test;

Masukkan No HP:
08187901;

Masukkan Weton:
PaHING;

Profil Anda telah berhasil diperbarui!

>> LIHAT_PROFIL gana;

| nama: gana
| Bio Akun: test
| No HP: 08187901
| Weton: Pahing

Foto Profil:
*****
*****
*****
*****
*****

```

4. ATUR_JENIS_AKUN

- Jika *user* belum melakukan login dan memanggil perintah ATUR_JENIS_PROFIL, maka program akan mengeluarkan pesan kesalahan.

```

>> ATUR_JENIS_AKUN;

Anda belum login! Masuk terlebih dahulu untuk menikmati layanan BurBir.

```

STEI- ITB	IF2110-TB-I-K02	Halaman 36 dari 89 halaman	
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.			

- Jika akun *user* adalah akun publik, memanggil perintah ATUR_JENIS_AKUN akan mengubah akun *user* menjadi privat dan sebaliknya

```
>> ATUR_JENIS_AKUN;

Saat ini, akun Anda adalah akun publik. Ingin mengubah ke akun privat? (YA/TIDAK)YA;
Akun berhasil diganti.

>> ATUR_JENIS_AKUN;

Saat ini, akun Anda adalah akun privat. Ingin mengubah ke akun publik? (YA/TIDAK)YA;
Akun berhasil diganti.
```

- Jika *user* memasukkan input tidak, maka akun tidak jadi diubah status publik/privatnya. ATUR_JENIS_AKUN akan menerima input berupa “YA” atau “TIDAK”. Jika *user* memasukkan input selain itu, maka perintah akan dibatalkan.

```
>> ATUR_JENIS_AKUN;

Saat ini, akun Anda adalah akun publik. Ingin mengubah ke akun privat? (YA/TIDAK)TIDAK;
Akun tidak berhasil diganti.

>> ATUR_JENIS_AKUN;

Saat ini, akun Anda adalah akun publik. Ingin mengubah ke akun privat? (YA/TIDAK)ok;
Masukkan Anda bukan berupa "YA" atau "TIDAK". Perintah dibatalkan.
```

STEI- ITB	IF2110-TB-I-K02	Halaman 37 dari 89 halaman	
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.			

6.5 Teman

Pada *test* ketiga, akan dilakukan uji coba untuk fitur-fitur yang terdapat di Teman.

- Fitur yang dites:
 - DAFTAR_TEMAN
 - HAPUS_TEMAN

1. DAFTAR_TEMAN

- Perintah DAFTAR_TEMAN akan menunjukkan teman yang dimiliki oleh *user* setelah login.

```
>> DAFTAR_TEMAN;  
  
gana belum memiliki teman.
```

```
>> DAFTAR_TEMAN;  
  
gana memiliki 1 teman:  
| wg
```

HAPUS_TEMAN

- Perintah HAPUS_TEMAN akan menghapus teman yang dimiliki oleh *user*.

```
>> HAPUS_TEMAN;  
Masukkan nama pengguna:  
wg;  
  
Apakah anda yakin ingin menghapus wg dari daftar teman Anda? (YA/TIDAK)YA;  
  
>> DAFTAR_TEMAN;  
  
gana belum memiliki teman.
```

STEI- ITB	IF2110-TB-I-K02	Halaman 38 dari 89 halaman	
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.			

6.6 Permintaan Pertemanan

Pada *test* kelima ini, akan dilakukan uji coba untuk fitur-fitur yang terdapat di Permintaan Pertemanan.

- Fitur yang dites:
 - TAMBAH_TEMAN
 - DAFTAR_PERMINTAAN_PERTEMANAN
 - SETUJUI_PERTEMANAN

1. TAMBAH_TEMAN

Fitur TAMBAH_TEMAN yang bertujuan untuk menambahkan pertemanan dengan mengirimkan permintaan teman kepada orang lain.

- Perintah TAMBAH_TEMAN digunakan untuk menambah teman. Setelah memasukkan perintah TAMBAH_TEMAN, *user* akan diminta untuk memasukkan username dari akun yang ingin ditambahkan sebagai teman. Jika username akun yang dimasukkan tidak terdaftar maka TAMBAH_TEMAN akan gagal. Jika username akun yang dimasukkan terdaftar maka TAMBAH_TEMAN akan berhasil.

```
>> TAMBAH_TEMAN;

Masukkan nama pengguna:
tes;

Pengguna bernama tes tidak ditemukan.

>> TAMBAH_TEMAN;

Masukkan nama pengguna:
bert;
```

- Jika ada permintaan pertemanan yang masuk, maka *user* tidak dapat melakukan TAMBAH_TEMAN.

```
>> TAMBAH_TEMAN;

Terdapat permintaan pertemanan yang belum Anda setujui. Silakan kosongkan daftar permintaan pertemanan untuk Anda terlebih dahulu.
```

2. DAFTAR_PERMINTAAN_PERTEMANAN

Fitur DAFTAR_PERMINTAAN_PERTEMANAN yang bertujuan untuk melihat daftar permintaan pertemanan.

- DAFTAR_PERMINTAAN_PERTEMANAN akan menampilkan daftar akun yang meminta pertemanan kepada *user* diurutkan berdasarkan jumlah teman yang dimiliki oleh peminta

STEI- ITB	IF2110-TB-I-K02	Halaman 39 dari 89 halaman	
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.			

```
>> DAFTAR_PERMINTAAN_PERTEMANAN;
```

```
Terdapat 1 permintaan pertemanan untuk Anda.
```

```
| Nama: wg  
| Jumlah teman: 0
```

```
>> DAFTAR_PERMINTAAN_PERTEMANAN;
```

```
Terdapat 0 permintaan pertemanan untuk Anda.
```

3. SETUJUI_PERTEMANAN

Fitur SETUJUI_PERTEMANAN yang bertujuan untuk menerima atau menolak permintaan pertemanan.

- Perintah SETUJUI_PERTEMANAN dapat digunakan untuk menerima atau menolak permintaan pertemanan yang masuk kepada *user*.

```
>> SETUJUI_PERTEMANAN;
```

```
Permintaan pertemanan teratas dari wg
```

```
| Nama: wg  
| Jumlah teman: 0
```

```
Apakah Anda ingin menyetujui permintaan pertemanan ini? (YA/TIDAK) YA;
```

```
Permintaan pertemanan dari wg telah disetujui. Selamat! Anda telah berteman dengan wg.
```

- Jika tidak ada permintaan pertemanan yang masuk, program akan mengeluarkan pesan yang memberitahu bahwa *user* tidak mempunyai permintaan pertemanan.

```
>> SETUJUI_PERTEMANAN;
```

```
Belum ada yang mengajak Anda berteman.
```

STEI- ITB	IF2110-TB-I-K02	Halaman 40 dari 89 halaman	
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.			

6.7 Kicauan

Pada *test* keenam ini, akan dilakukan uji coba untuk fitur-fitur yang terdapat di kicauan.

- Fitur yang dites:
 - KICAU
 - KICAUAN
 - SUKA_KICAUAN
 - UBAH_KICAUAN

1. KICAU

Fitur KICAU yang bertujuan untuk membuat sebuah kicauan baru

- Hasil yang seharusnya diberikan:
Kicauan hanya bisa menampung kata kata sebanyak 280 huruf, jika kicauan yang diinput melebihi 280 huruf maka kata kata akan otomatis dipotong.

```
Masukkan Kicauan : KICAU 1;

Detail Kicauan :
| ID = 1
| S
| 23/11/2023 02:21:01
| KICAU 1
| Disukai : 0

>> KICAU2;

Tidak ada perintah KICAU2. Baca dokumentasi (Spesifikasi) untuk melihat seluruh perintah.

>> KICAU;

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : KICAU2;

Detail Kicauan :
| ID = 2
| S
| 23/11/2023 02:21:07
| KICAU2
| Disukai : 0
```

2. KICAUAN

Fitur KICAUAN yang bertujuan untuk menampilkan seluruh kicauan milik teman dan diri sendiri

- Hasil apa yang seharusnya diberikan:

STEI- ITB	IF2110-TB-I-K02	Halaman 41 dari 89 halaman	
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.			

Pada saat memasukkan perintah kicauan, program akan mencetak kicauan yang dimiliki oleh akun publik atau teman dari pengguna saat ini.

```
>> KICAUAN;

Detail Kicauan :
| ID = 2
| S
| 23/11/2023 02:21:07
| KICAU2
| Disukai : 0

Detail Kicauan :
| ID = 1
| S
| 23/11/2023 02:21:01
| KICAU 1
| Disukai : 0
```

```
>> KICAU;

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : ;
Kicauan tidak boleh hanya spasi
Ulang masukkan kicauan : ;
Kicauan tidak boleh hanya spasi
Ulang masukkan kicauan : INI KICAUAN FED2 YANG KEDUA;

Detail Kicauan :
| ID = 2
| FED2
| 24/11/2023 08:57:53
| INI KICAUAN FED2 YANG KEDUA
| Disukai : 0
```

3. SUKA KICAUAN

Fitur suka kicauan yang bertujuan untuk memberikan like kepada kicauan milik siapapun

- Hasil apa yang seharusnya diberikan

Pada saat memasukkan perintah suka kicauan, pengguna hanya bisa menyukai kicauan yang dimiliki akun publik atau pengguna merupakan teman dari memiliki kicauan tersebut.

STEI- ITB	IF2110-TB-I-K02	Halaman 42 dari 89 halaman	
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.			

```
>> SUKA_KICAUAN 2;

Detail Kicauan :
| ID = 2
| S
| 23/11/2023 02:21:07
| KICAU2
| Disukai : 1

>> SUKA_KICAUAN 2;

Detail Kicauan :
| ID = 2
| S
| 23/11/2023 02:21:07
| KICAU2
| Disukai : 2
```

```
>> SUKA_KICAUAN 1;
Tidak ditemukan kicauan dengan ID = 1
```

4. UBAH_KICAUAN

Fitur ubah kicauan untuk mengubah isi dari kicauan milik diri sendiri.

- Hasil apa yang seharusnya diberikan
 Pada saat memasukkan perintah ubah kicauan, pengguna hanya bisa mengedit kicauan yang dimiliki. Jika pengguna memasukkan kicauan milik orang lain atau melebihi jumlah kicauan yang ada akan muncul pesan error. Cara penggunaan ubah_kicauan adalah dengan UBAH_KICAUAN <IDKicau>;

STEI- ITB	IF2110-TB-I-K02	Halaman 43 dari 89 halaman	
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.			

```
>> UBAH_KICAUAN 2;
Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.
Masukkan Kicauan : KICAU2 DIUBAH;

>> KICAUAN;

Detail Kicauan :
| ID = 2
| S
| 23/11/2023 02:21:07
| KICAU2 DIUBAH
| Disukai : 2

Detail Kicauan :
| ID = 1
| S
| 23/11/2023 02:21:01
| KICAU 1
| Disukai : 0
```

```
>> UBAH_KICAUAN 1;
Tidak ditemukan kicauan dengan ID = 1
```

```
>> UBAH_KICAUAN 1;
Kamu hanya bisa mengubah kicauan milikmu sendiri
```

6.8 Balasan

Pada test ini akan dilakukan uji coba untuk fitur yang ada di dalam Balasan.

- Fitur yang dites:
 - Balas
 - Balasan
 - HAPUS_BALASAN

STEI- ITB	IF2110-TB-I-K02	Halaman 44 dari 89 halaman	
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.			

- Input tidak tepat pada saat balas

```
>> BALAS;

Perintah yang valid: BALAS "<IDKicau> <IDBalasan>".

>> BALAS 1 1;

Wah, tidak terdapat balasan yang ingin Anda balas! Mungkin typo IDBalasan ?
```

- Balasan ke kicau utama

```
>> BALAS 1 -1;

Masukkan balasan:
INI BALASAN KE KICAU UTAMA;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 0
| INDRA
| 24/11/2023 18:39:01
| INI BALASAN KE KICAU UTAMA
```

- Balasan ke Kicauan yang sama

```
>> BALAS 1 -1;

Masukkan balasan:
INI BALASAN KEDUA KE KICAU UTAMA;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 1
| INDRA
| 24/11/2023 18:40:08
| INI BALASAN KEDUA KE KICAU UTAMA
```

STEI- ITB	IF2110-TB-I-K02	Halaman 45 dari 89 halaman	
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.			

- Test balasan lagi

```
>> BALAS 1 0;

Masukkan balasan:
INI BALAS KE BALASAN YANG BALES KE KICAU UTAMA;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 2
| INDRA
| 24/11/2023 18:43:31
| INI BALAS KE BALASAN YANG BALES KE KICAU UTAMA

>> BALAS 1 1;

Masukkan balasan:
INI BALAS KE BALASAN YANG BALES KE BALESAN YANG BALES KICAU UTAAMA;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 3
| INDRA
| 24/11/2023 18:44:13
| INI BALAS KE BALASAN YANG BALES KE BALESAN YANG BALES KICAU UTAAMA
```

STEI- ITB	IF2110-TB-I-K02	Halaman 46 dari 89 halaman	
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.			

```
>> BALAS 1 2;

Masukkan balasan:
INI BALASAN LAGI;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 4
| INDRA
| 24/11/2023 18:46:51
| INI BALASAN LAGI
```

- **Tampilan**

```
>> BALASAN 1;
| ID = 0
| INDRA
| 24/11/2023 18:39:01
| INI BALASAN KE KICAU UTAMA

| ID = 1
| INDRA
| 24/11/2023 18:40:08
| INI BALASAN KEDUA KE KICAU UTAMA
```

STEI- ITB	IF2110-TB-I-K02	Halaman 47 dari 89 halaman	
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.			

```
>> BALASAN 1;
| ID = 0
| INDRA
| 24/11/2023 18:39:01
| INI BALASAN KE KICAU UTAMA

| ID = 2
| INDRA
| 24/11/2023 18:43:31
| INI BALAS KE BALASAN YANG BALES KE KICAU UTAMA

| ID = 1
| INDRA
| 24/11/2023 18:40:08
| INI BALASAN KEDUA KE KICAU UTAMA

| ID = 3
| INDRA
| 24/11/2023 18:44:13
| INI BALAS KE BALASAN YANG BALES KE BALEAN YANG BALES KICAU UTAMA
```

- **Hapus Balasan**

```
>> HAPUS_BALASAN 1;

Perintah yang valid: "HAPUS_BALASAN <IDKicau> <IDBalasan>".
```

```
>> HAPUS_BALASAN;

Perintah yang valid: "HAPUS_BALASAN <IDKicau> <IDBalasan>".

>> HAPUS_BALASAN 1 4;

Balasan berhasil dihapus.

>> BALASAN 1;
| ID = 0
| INDRA
| 24/11/2023 18:39:01
| INI BALASAN KE KICAU UTAMA

| ID = 2
| INDRA
| 24/11/2023 18:43:31
| INI BALAS KE BALASAN YANG BALES KE KICAU UTAMA
```

STEI- ITB	IF2110-TB-I-K02	Halaman 48 dari 89 halaman	
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.			

- **Hapus Balasan Parent**

```
>> BALAS 1 2;

Masukkan balasan:
INI BALASAN KE BALASAN DENGAN ID 2 YANG BALAS KE BALASAN ID 0;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 5
| INDRA
| 24/11/2023 18:51:40
| INI BALASAN KE BALASAN DENGAN ID 2 YANG BALAS KE BALASAN ID 0

>> BALASAN 1;
| ID = 0
| INDRA
| 24/11/2023 18:39:01
| INI BALASAN KE KICAU UTAMA

| ID = 2
| INDRA
| 24/11/2023 18:43:31
| INI BALAS KE BALASAN YANG BALES KE KICAU UTAMA

| ID = 5
| INDRA
| 24/11/2023 18:51:40
| INI BALASAN KE BALASAN DENGAN ID 2 YANG BALAS KE BALASAN ID 0

| ID = 1
| INDRA
| 24/11/2023 18:40:08
| INI BALASAN KEDUA KE KICAU UTAMA

| ID = 3
| INDRA
| 24/11/2023 18:44:13
| INI BALAS KE BALASAN YANG BALES KE BALESAN YANG BALES KICAU UTAAMA
```

STEI- ITB	IF2110-TB-I-K02	Halaman 49 dari 89 halaman	
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.			

```
>> HAPUS_BALASAN 1 2;

Balasan berhasil dihapus.

>> BALASAN 1;
| ID = 0
| INDRA
| 24/11/2023 18:39:01
| INI BALASAN KE KICAU UTAMA

| ID = 1
| INDRA
| 24/11/2023 18:40:08
| INI BALASAN KEDUA KE KICAU UTAMA

| ID = 3
| INDRA
| 24/11/2023 18:44:13
| INI BALAS KE BALASAN YANG BALES KE BALESAN YANG BALES KICAU UTAAMA
```

6.9 *Draf Kicauan*

Pada test ini akan dilakukan uji coba untuk fitur yang ada di dalam Draf.

- Fitur yang dites:
 - BUAT_DRAF
 - LIHAT_DRAF
 - TERBIT
 - UBAH
 - SIMPAN
 - HAPUS

1. BUAT_DRAF

- **Hasil yang Diharapkan**

Ekspektasi saat menjalankan perintah BUAT_DRAF adalah tidak bisa membuat draf saat belum login sebagai user yang terdaftar dan hasilnya sesuai

```
>> BUAT_DRAF;

Anda belum login! Masuk terlebih dahulu untuk menikmati layanan BurBir.
```

STEI- ITB	IF2110-TB-I-K02	Halaman 50 dari 89 halaman	
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.			

- **Hasil saat sudah login**

Ekspektasi saat sudah login kita bisa membuat drafnya untuk setiap user. Hasilnya sendiri sesuai dengan ekspektasi kita bisa membuat draf saat sudah login sebagai user yang sudah terdaftar

```
Anda telah berhasil masuk dengan nama pengguna INDRA. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> BUAT_DRAF;
Masukkan Draf:
```

- **Buat Draf saat berhasil disimpan**

Saat draf berhasil disimpan seharusnya akan tersimpan di dalam stack dan dapat terlihat jika kita gunakan perintah LIHAT_DRAF. Hasilnya sesuai dengan ekspektasi dan dapat melihat elemen draf paling atas dari stack draft tersebut

```
>> BUAT_DRAF;
Masukkan Draf: INI DRAF TESTING;
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini? SIMPAN;
Draft berhasil disimpan

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:01:35
|INI DRAF TESTING
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? _
```

- **Buat Draf lalu berhasil dihapus**

Test kali ini digunakan untuk mengecek buat draf lalu langsung menghapus draf tersebut, ekspektasi yang diinginkan adalah di stack tidak akan tersimpan apapun. Hasilnya adalah di stack tidak menghasilkan apapun

```
>> MASUK;

Masukkan nama:
INDRA;

Masukkan kata sandi:
INDRA;

Anda telah berhasil masuk dengan nama pengguna INDRA. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> BUAT_DRAF;
Masukkan Draf: INI DRAF TESTING INDRA;
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini? HAPUS;
Draft berhasil dihapus

>> LIHAT_DRAF;
Yah, anda belum memiliki draf apapun! Buat dulu dong
```

2. LIHAT_DRAF

- **Lihat Draf sekaligus Hapus**

Saat *user* memasukkan perintah LIHAT_DRAF, *user* harus memilih salah satu dari ketiga pilihan. Jika user memilih HAPUS, maka draf akan dihapus.

STEI- ITB	IF2110-TB-I-K02	Halaman 51 dari 89 halaman	
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.			

```
>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:01:35
|INI DRAF TESTING
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? HAPUS;
Draf berhasil dihapus
```

- **Lihat Draf saat tidak ada Draf**

Ketika *user* memasukkan perintah LIHAT_DRAF saat tidak ada draf, program akan mengeluarkan pesan bahwa tidak ada draf.

```
>> LIHAT_DRAF;
Yah, anda belum memiliki draf apapun! Buat dulu dong
```

- **Lihat Draf lalu menerbitkan Draf**

Ketika *user* memilih TERBIT saat memasukkan perintah LIHAT_DRAF, maka draf paling atas akan diterbitkan menjadi kicauan.

```
Masukkan nama:
INDRA;

Masukkan kata sandi:
INDRA;

Anda telah berhasil masuk dengan nama pengguna INDRA. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:06:45
|INI DRAF MILIK INDRA
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? TERBIT;

Detail Kicauan :
| ID = 5
| INDRA
| 23/11/2023 12:06:45
| INI DRAF MILIK INDRA
| Disukai : 0
```

STEI- ITB	IF2110-TB-I-K02	Halaman 52 dari 89 halaman	
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.			

- **Lihat Draf saat berganti user**

```
>> MASUK;

Masukkan nama:
BRI;

Masukkan kata sandi:
BRA;

Anda telah berhasil masuk dengan nama pengguna BRI. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> LIHAT_DRAF;
Yah, anda belum memiliki draf apapun! Buat dulu dong
>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:
INDRA;

Masukkan kata sandi:
INDRA;

Anda telah berhasil masuk dengan nama pengguna INDRA. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:06:45
|INI DRAF MILIK INDRA
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? _
```

3. TERBIT

- **LIHAT_DRAF lalu TERBIT**

Ketika *user* memilih TERBIT saat memasukkan perintah LIHAT_DRAF, maka draf paling atas akan diterbitkan menjadi kicauan.

```
Masukkan nama:
INDRA;

Masukkan kata sandi:
INDRA;

Anda telah berhasil masuk dengan nama pengguna INDRA. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:06:45
|INI DRAF MILIK INDRA
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? TERBIT;

Detail Kicauan :
| ID = 5
| INDRA
| 23/11/2023 12:06:45
| INI DRAF MILIK INDRA
| Disukai : 0
```

STEI- ITB	IF2110-TB-I-K02	Halaman 53 dari 89 halaman	
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.			

- **BUAT_DRAF lalu TERBIT**

Saat membuat draf dengan perintah **BUAT_DRAFT**, *user* dapat langsung menerbitkan draf tersebut menjadi sebuah kicauan dengan memilih **TERBIT**

```
Selamat! Draf telah berhasil diterbitkan
>> BUAT_DRAF;
Masukkan Draf: INI DRAF MILIK INDRA2;
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini? TERBIT;

Detail Kicauan :
| ID = 6
| INDRA
| 23/11/2023 12:11:37
| INI DRAF MILIK INDRA2
| Disukai : 0
```

4. UBAH

- **LIHAT_DRAF lalu UBAH**

Ketika *user* memilih **UBAH** saat memasukkan perintah **LIHAT_DRAF**, maka *user* dapat melakukan perubahan pada draf dan *datetime* pada draf akan berubah.

```
Selamat! Draf telah berhasil diterbitkan
>> BUAT_DRAF;
Masukkan Draf: INI DRAF INDRA1;
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini? SIMPAN;
Draft berhasil disimpan

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:43:03
|INI DRAF INDRA1
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? UBAH;
Masukkan Perubahan: INI PERUBAHAN DRAF INDRA1;
Draf berhasil diubah

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:43:14
|INI PERUBAHAN DRAF INDRA1
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?
```

5. SIMPAN

- **SIMPAN setelah BUAT_DRAF**

Ketika *user* memilih **SIMPAN** setelah selesai membuat draft dengan perintah **BUAT_DRAF**, maka draf akan tersimpan.

```
>> BUAT_DRAF;
Masukkan Draf: INI DRAF BUAT DISIMPAN;
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini? SIMPAN;
Draft berhasil disimpan

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:44:17
|INI DRAF BUAT DISIMPAN
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?
```

STEI- ITB	IF2110-TB-I-K02	Halaman 54 dari 89 halaman	
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.			

6. HAPUS

- **LIHAT_DRAF lalu HAPUS**

Ketika *user* memilih HAPUS saat memasukkan perintah LIHAT_DRAF, maka draf paling atas akan dihapus dari stack.

```
>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:44:17
|INI DRAF BUAT DISIMPAN
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? HAPUS;
Draf berhasil dihapus

>> LIHAT_DRAF;
Ini draf terakhir anda
|23/11/2023 12:43:14
|INI PERUBAHAN DRAF INDRA1
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? _
```

- **BUAT_DRAF lalu HAPUS**

Ketika *user* memilih HAPUS setelah selesai membuat draf dengan perintah BUAT_DRAF, maka draf tersebut akan dihapus.

```
>> BUAT_DRAF;
Masukkan Draf: DRAF LAGI;
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini? HAPUS;
Draf berhasil dihapus
```

6.10 Utas

- Fitur yang dites:
 - UTAS
 - SAMBUNG_UTAS
 - HAPUS_UTAS
 - CETAK_UTAS

1. UTAS

Fitur utas ini untuk membuat utas dari id kicau yang diberikan

- Hasil apa yang seharusnya diberikan

Pada saat memasukkan perintah Utas maka program akan meminta kicauan yang akan dimasukkan oleh user. Huruf maksimal dari Kicauan yang dimasukkan adalah sejumlah 280 huruf, jika pengguna memasukkan huruf melebihi 280 huruf maka program akan otomatis memotong huruf sampai 280. Cara penggunaan utas adalah dengan UTAS <IDKicau>;

STEI- ITB	IF2110-TB-I-K02	Halaman 55 dari 89 halaman	
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.			

```

>> UTAS 1;
Utas Berhasil Dibuat!
Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.
Masukkan Kicauan : UTAS 1;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK)YA;
Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.
Masukkan Kicauan : UTAS 2;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK)YA;
Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.
Masukkan Kicauan : UTAS 3;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK)TIDAK;
Utas selesai!

```

2. SAMBUNG_UTAS

Fitur sambung utas ini untuk menyambung utas dengan urutan indeks di id utas yang sudah pernah dibuat sebelumnya.

- Hasil apa yang seharusnya diberikan
 Utas dapat disambung di bagian tengah ataupun akhir dari utasnya. Jika pengguna memasukkan id utas dan indeks utas yang tidak ada maka akan keluar pesan error. Cara penggunaan ubah_kicauan adalah dengan SAMBUNG_UTAS <IDUtas> <Indeks Utas>;

```

>> SAMBUNG_UTAS 1 1;
Masukkan Kicauan: UTAS 1 DIPERPANJANG ;
Utas berhasil ditambahkan!
>> SAMBUNG_UTAS 1 3;
Masukkan Kicauan: UTAS 1 DIPERPANJANG LAGI;
Utas berhasil ditambahkan!
>> UTAS 1 9;

Perintah yang valid: "UTAS <IDKICAU>;".

>> SAMBUNG_UTAS 1 -1;
Kamu tidak bisa menyambung Utas dengan indeks -1
>> 

```

STEI- ITB	IF2110-TB-I-K02	Halaman 56 dari 89 halaman	
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.			

3. HAPUS_UTAS

Fitur hapus utas ini untuk menghapus utas dengan urutan indeks di id utas yang sudah pernah dibuat sebelumnya.

- Hasil apa yang seharusnya diberikan
Utas dapat dihapus di bagian tengah ataupun akhir dari utasnya. Jika pengguna memasukkan id utas dan indeks utas yang tidak ada maka akan keluar pesan error. Cara penggunaan ubah_kicauan adalah dengan HAPUS_UTAS <IDUtas> <Indeks Utas>;

```
Kamu tidak bisa menghapus Utas dengan indeks -1
>> HAPUS_UTAS 1 1;
Utas berhasil dihapus!
>> HAPUS_UTAS 1 -1;
Kamu tidak bisa menghapus Utas dengan indeks -1
>> CETAK_UTAS 1;
```

4. CETAK_UTAS

Fitur cetak utas ini untuk mencetak kicauan utama dan semua utas yang ada dengan id utas yang diminta.

- Hasil apa yang seharusnya diberikan
Cetak utas akan mencetak kicauan utama dan semua utas yang berada di kicauan dengan id utas yang diminta. Cara penggunaan ubah_kicauan adalah dengan CETAK_UTAS <IDUtas>;

```
Kamu tidak bisa menghapus Utas dengan indeks -1
>> CETAK_UTAS 1;

Detail Kicauan :
| ID = 1
| S
| 23/11/2023 04:01:04
| KICAUQ
| Disukai : 0

| INDEX = 1
| S
| 23/11/2023 04:01:15
| UTAS 1

| INDEX = 2
| S
| 23/11/2023 04:08:26
| UTAS 1 DIPERPANJANG LAGI

| INDEX = 3
| S
| 23/11/2023 04:01:22
| UTAS 2

| INDEX = 4
| S
| 23/11/2023 04:01:27
| UTAS 3
```

STEI- ITB	IF2110-TB-I-K02	Halaman 57 dari 89 halaman	
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.			

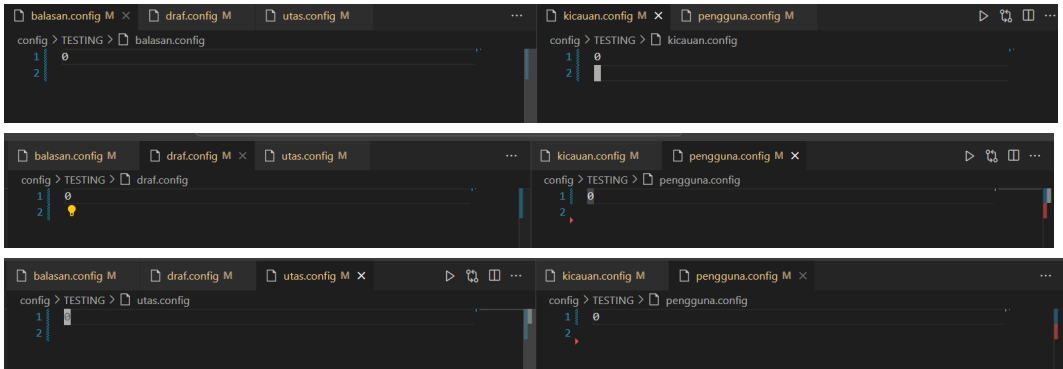
6.11 Simpan dan Muat

Pada test kali ini akan diuji fungsi simpan untuk menyimpan progress yang sudah dilakukan dari program utama sehingga sewaktu-waktu bisa melanjutkan progress tersebut.

1. Simpan pertama kali

Ekspektasi yang dihasilkan dari testing ini adalah seluruh file yang ada hanya akan muncul angka 0 yang menandakan tidak ada apa-apa di dalamnya. Hasil yang didapatkan sesuai dengan apa yang diharapkan.

```
>> SIMPAN;
Masukkan nama folder: TESTING;
Belum terdapat TESTING. Akan dilakukan pemuatan TESTING terlebih dahulu
Mohon tunggu...
1...
2...
3...
TESTING berhasil dibuat
TESTING sudah dibuat
```



The image shows three sequential screenshots of a code editor interface. The first screenshot shows the creation of the 'TESTING' folder and the initial state of three config files: 'balasan.config', 'kicauan.config', and 'pengguna.config'. Each file contains a single line '0' at index 1. The second screenshot shows the same files, but the 'kicauan.config' file now has a second line '2' at index 2. The third screenshot shows the same files, but the 'pengguna.config' file now has a second line '2' at index 2.

2. Saving Pengguna

- **Daftar pertama kali**

Pada saat pertama kali daftar kemudian disimpan, akan terlihat jumlah pengguna pada baris pertama, kemudian nama pengguna dan passwordnya, kemudian jenis akun, foto profil *default*, dan matrix pertemanan yang berjumlah satu.

STEI- ITB	IF2110-TB-I-K02	Halaman 58 dari 89 halaman	
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.			

```
>> DAFTAR;

Masukkan nama:
WG;

Masukkan kata sandi:
WG;

Pengguna telah berhasil terdaftar. Masuk untuk menikmati fitur-fitur BurBir.

>> SIMPAN;
Masukkan nama folder: TESTING;
Folder sudah ada
Folder akan diperbarui
ROI
>>
```

kicauan.config M pengguna.config M x

config > TESTING > pengguna.config

```
1      1
2      WG
3      WG
4
5
6
7      Publik
8      R * R * R * R * R *
9      R * R * R * R * R *
10     R * R * R * R * R *
11     R * R * R * R * R *
12     R * R * R * R * R *
13     1
14     0
15
```

- **Mengubah profil**
Pada bagian ini akan mengetes simpan setelah mengubah profil pada pengguna. Setelah memanggil perintah GANTI_PROFIL dan mengubah profil dengan benar kemudian disimpan, maka file *config* akan berubah sesuai dengan hasil GANTI_PROFIL.

Anda telah berhasil masuk dengan nama pengguna WG. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> GANTI_PROFIL;

| nama: WG
| Bio Akun:
| No HP:
| Weton:

Masukkan Bio Akun:
ALSTRUKDAT MANTAP;

Masukkan No HP:
081234567;

Masukkan Weton:
PAHING;

Profil Anda telah berhasil diperbarui!

>> SIMPAN;
Masukkan nama folder: TESTING;
Folder sudah ada
Folder akan diperbarui

2

WG

WG

ALSTRUKDAT MANTAP

081234567

Pahing

Publik

R * R * R * R * R *

R * R * R * R * R *

R * R * R * R * R *

R * R * R * R * R *

R * R * R * R * R *

STEI- ITB	IF2110-TB-I-K02	Halaman 60 dari 89 halaman	
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.			

- **Mengubah Foto Profil**

Pada bagian ini akan menunjukkan hasil simpan setelah pengguna mengubah foto profilnya. Setelah sukses mengubah foto profil dan memanggil simpan, maka akan terlihat perubahan foto profil pada file config hasil simpan.

```
>> UBAH_FOTO_PROFIL;
Foto profil Anda saat ini adalah:
*****
*****
*****
*****
*****
```

```
Masukkan foto profil yang baru:
G 8 G 8 G 8 G 8 G 8
G 8 G 8 G 8 G 8 G 8
G 1 G 1 G 1 G 1 G 1
R 2 R 2 R 2 R 2 R 2
R 3 R 3 R 3 R 3 R 3;
Foto profil Anda sudah berhasil diganti!.
```

```
Foto Profil:
88888
88888
11111
22222
33333
```

```
2
WG
WG
ALSTRUKDAT MANTAP
081234567
Pahing
Publik
G 8 G 8 G 8 G 8 G 8
G 8 G 8 G 8 G 8 G 8
G 1 G 1 G 1 G 1 G 1
R 2 R 2 R 2 R 2 R 2
R 3 R 3 R 3 R 3 R 3
FED
FED
```

- **User saat lebih dari satu disimpan**

STEI- ITB	IF2110-TB-I-K02	Halaman 61 dari 89 halaman	
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.			

```

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> DAFTAR;

Masukkan nama:
WG;

Masukkan kata sandi:
WG;

Pengguna telah berhasil terdaftar. Masuk untuk menikmati fitur-fitur BurBir.

>> MASUK;

Masukkan nama:
WG;

Masukkan kata sandi:
WG;

Anda telah berhasil masuk dengan nama pengguna WG. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> KICAU;

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : SIMPAN;

Detail Kicauan :
| ID = 2
| WG
| 23/11/2023 23:57:24
| SIMPAN
| Disukai : 0

>> SIMPAN;

Masukkan nama folder: NEW;
Folder sudah ada
Folder akan diperbarui
total dalam IF: 1

```

- **Keluaran matrix pertemanan hasil saving**

```

>> SIMPAN;
Masukkan nama folder: FOLDER;
Folder sudah ada
Folder akan diperbarui

```

```

28
29 Publik
30 R * R * R * R * R *
31 R * R * R * R * R *
32 R * R * R * R * R *
33 R * R * R * R * R *
34 R * R * R * R * R *
35 1 1 0
36 1 1 0
37 0 0 1
38 1
39 0 2 1
40

```

STEI- ITB	IF2110-TB-I-K02	Halaman 62 dari 89 halaman	
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.			

2
g
g

Publik

R * R * R * R * R *
R * R * R * R * R *
R * R * R * R * R *
R * R * R * R * R *
R * R * R * R * R *

WG
WG

Publik

R * R * R * R * R *
R * R * R * R * R *
R * R * R * R * R *
R * R * R * R * R *
R * R * R * R * R *

1 0
0 1
0

STEI- ITB	IF2110-TB-I-K02	Halaman 63 dari 89 halaman	
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.			

3. Test saving kicauan

```
>> KICAU;

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : KICAUAN PERTAMA;

Detail Kicauan :
| ID = 1
| WG
| 23/11/2023 19:14:08
| KICAUAN PERTAMA
| Disukai : 0

>> SIMPAN;
Masukkan nama folder: TESTING;
Folder sudah ada
Folder akan diperbarui
```

```
config > TESTING > kicauan.config
1 1
2 1
3 KICAUAN PERTAMA
4 0
5 WG
6 23/11/2023 19:14:08
7
```

- Ubah Kicauan

Pada testing kali ini kami mengubah kicauan lalu menyimpannya ke dalam file kicauan.config, hasil yang diharapkan adalah perubahan akan terjadi ke dalam file. Hasil yang didapatkan adalah sesuai dengan apa yang diharapkan

```
>> UBAH_KICAUAN 1;
Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.
Masukkan Kicauan : PERUBAHAN KICAUAN PERTAMA;

config > TESTING > kicauan.config
1 1
2 1
3 PERUBAHAN KICAUAN PERTAMA
4 0
5 WG
6 23/11/2023 19:14:08
7
```

STEI- ITB	IF2110-TB-I-K02	Halaman 64 dari 89 halaman	
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.			

- **Kicauan lebih dari satu**

Pada testing ini kami mengecek kicauan lebih dari satu, ekspektasinya adalah ID-nya bisa lebih dari satu dan hasil yang didapatkan adalah ID bisa lebih dari satu dan di file kicauan.config akan menampilkan 2 kicauan dengan ID 1 dan 2.

```
>> KICAU;

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : INI KICAU TEST KEDUA;

Detail Kicauan :
| ID = 2
| WG
| 23/11/2023 20:05:17
| INI KICAU TEST KEDUA
| Disukai : 0

>> SIMPAN;
Masukkan nama folder: TESTING;
Folder sudah ada
Folder akan diperbarui
```

```
2
1
PERUBAHAN KICAUAN PERTAMA
0
WG
23/11/2023 19:14:08
2
INI KICAU TEST KEDUA
0
WG
23/11/2023 20:05:17
```

4. Simpan Draf

Pada testing simpan draf kami mengecek jika berhasil membuat draf dan menyimpan ke dalam file ekspektasinya di dalam file akan ditampilkan jumlah user yang memiliki draf dan list draf yang ada. Hasil yang didapatkan sesuai dengan apa yang diharapkan.

STEI- ITB	IF2110-TB-I-K02	Halaman 65 dari 89 halaman	
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.			

5. Simpan Utas

- **Simpan utas pertama kali**

Pada simpan utas kami, kami membuat utas di dalam kicauan dan di dalam kicauan tersebut dibuat utas 3 kali, sehingga di dalam file ekspektasinya akan menghasilkan kicauan dengan ID 1 dan mempunya 3 utas. Hasil yang didapatkan sesuai dengan yang diharapkan.

- **Simpan utas saat dihapus**

Hasil yang diharapkan adalah saat menyimpan file, utas kedua hilang dari file dan hanya ada utas 1 dan utas 3 yang disimpan di dalam file. Hasil yang didapatkan sesuai dengan ekspektasi

STEI- ITB	IF2110-TB-I-K02	Halaman 66 dari 89 halaman	
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.			

```

simpan.c 1  pengguna.config M
config > NEW > utas.config
1 1
2 1
3 2
4 INI UTAS KICAU1
5 g
6 23/11/2023 23:40:12
7 INI UTAS 3 KICAU1
8 g
9 23/11/2023 23:40:36
10

>> SIMPAN;
Masukkan nama folder: NEW;
Folder sudah ada
Folder akan diperbarui

>> HAPUS_UTAS 1 2;
Utas berhasil dihapus!
>> SIMPAN;
Masukkan nama folder: NEW;
Folder sudah ada
Folder akan diperbarui

>>

```

6. Simpan Balasan

- **Simpan awal**

Hasil yang diharapkan adalah ditampilkannya 3 balasan untuk ID kicau 1. Hasil yang didapatkan sesuai dengan ekspektasi.

```

config > NEW > balasan.config
1 1
2 1
3 3
4 -1 0
5 INI BALASAN BUAT KICAU ID 1
6 g
7 23/11/2023 23:47:53
8 0 1
9 INI BALASAN UNTUK BALASAN DENGAN ID 0
10 g
11 23/11/2023 23:48:11
12 1 2
13 INI BALASAN UNTUK BALASAN ID 1
14 g
15 23/11/2023 23:49:25
16

>> BALAS 1 -1;
here
Masukkan balasan:
INI BALASAN BUAT KICAU ID 1;
Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 0
| g
| 23/11/2023 23:47:53
| INI BALASAN BUAT KICAU ID 1

>> BALAS 1 0;
here
Masukkan balasan:
INI BALASAN UNTUK BALASAN DENGAN ID 0;
Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 1
| g
| 23/11/2023 23:48:11
| INI BALASAN UNTUK BALASAN DENGAN ID 0

```

- **Simpan balasan lebih dari satu**

Hasil yang diinginkan di dalam file terdapat seluruh balasan dan hasil yang didapatkan sesuai dengan ekspektasi. Jumlah kicau yang memiliki balasan adalah

STEI- ITB	IF2110-TB-I-K02	Halaman 67 dari 89 halaman	
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.			

2, Kicau yang memiliki balasan adalah kicauan dengan ID 1 dan kicauan dengan ID 2 dan masing-masing kicauan punya 4 Balasan

```
2
1
4
-1 0
INI BALASAN UNTUK KICAUAN 1 FED2
INDRA2
24/11/2023 09:15:50
-1 1
INI BALASAN LAGI UNTUK KICAUAN 1 FED2
INDRA2
24/11/2023 09:16:24
1 3
INI BALASAN FED2 UNTUK BALASAN INDRA2
FED2
24/11/2023 09:17:44
-1 2
INI BALASAN FED2 UNTUK KICAU FED2
FED2
24/11/2023 09:17:14
```

```
2
4
-1 0
INI BALASAN UNTUK KICAU ID2
FED2
24/11/2023 09:19:05
0 1
INI BALASAN LAGI UNTUK KICAU ID 2
FED2
24/11/2023 09:19:40
0 2
INI BALASAN UNTUK BALASAN DENGAN ID 0
FED2
24/11/2023 09:20:10
-1 3
INI BALASAN LAGI UNTUK ID KICAU 2
FED2
24/11/2023 09:20:37
```

```
Masukkan nama:
FED2;

Masukkan kata sandi:
FED2;

Anda telah berhasil masuk dengan nama pengguna FED2. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> BALAS 1 -1;

Masukkan balasan:
INI BALASAN FED2 UNTUK KICAU FED2;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 2
| FED2
| 24/11/2023 09:17:14
| INI BALASAN FED2 UNTUK KICAU FED2
```

```
>> BALAS 1 -1;

Masukkan balasan:
INI BALASAN UNTUK KICAUAN 1 FED2;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 0
| INDRA2
| 24/11/2023 09:15:50
| INI BALASAN UNTUK KICAUAN 1 FED2
```

```
>> BALAS 1 -1;

Masukkan balasan:
INI BALASAN LAGI UNTUK KICAUAN 1 FED2;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 1
| INDRA2
| 24/11/2023 09:16:24
| INI BALASAN LAGI UNTUK KICAUAN 1 FED2
```

```
>> BALAS 1 1;

Masukkan balasan:
INI BALASAN FED2 UNTUK BALASAN INDRA2;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 3
| FED2
| 24/11/2023 09:17:44
| INI BALASAN FED2 UNTUK BALASAN INDRA2

>> BALAS 2 -1;

Masukkan balasan:
INI BALASAN UNTUK KICAU ID2;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 0
| FED2
| 24/11/2023 09:19:05
| INI BALASAN UNTUK KICAU ID2

>> BALAS 2 -1
;

Masukkan balasan:
INI BALASAN LAGI UNTUK KICAU ID 2;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 1
| FED2
| 24/11/2023 09:19:40
| INI BALASAN LAGI UNTUK KICAU ID 2

>> BALAS 2 0;

Masukkan balasan:
INI BALASAN UNTUK BALASAN DENGAN ID 0;

Selamat! balasan telah diterbitkan!
Detil balasan:
| ID = 2
| FED2
| 24/11/2023 09:20:10
| INI BALASAN UNTUK BALASAN DENGAN ID 0
```

Di atas ini adalah langkah-langkahnya.

7. Muat

Muat dilakukan untuk semua file config. Jika ada salah satu file config yang gagal dimuat, program akan memberitahu bahwa tidak ada folder yang dimaksud. Muat dapat dilakukan ditengah program saat pengguna tidak login.

```
>> MUAT;
Anda harus keluar terlebih dahulu untuk melakukan pemuatan.

Silahkan masukan folder konfigurasi untuk dimuat: fedz2;
File konfigurasi berhasil dimuat! Selamat berkicau!

>> DEVTOOLS;

isLoggedIn : FALSE

Graph pertemanan:
[1 0 1 0]
[0 1 0 0]
[1 0 1 0]
[0 0 0 1]

isi ListUser:

ID : 0
IS PRIVATE: 1

| nama: Tuan Man
| Bio Akun:
| No HP: 081
| Weton: Pahing
*****
* @* @*
* @* @*
* @* @*
*****

>> MUAT;
Masukkan nama folder yang hendak dimuat.
TESTING;

Anda akan melakukan pemuatan dari TESTING.
Mohon tunggu...
1...
2...
3...
Pemuatan selesai!
```

STEI- ITB	IF2110-TB-I-K02	Halaman 69 dari 89 halaman	
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.			

```
isi ListUser:
```

```
ID : 0
```

```
IS PRIVATE: 0
```

```
| nama: WG
```

```
| Bio Akun: ALSTRUKDAT MANTAP
```

```
| No HP: 081234567
```

```
| Weton: Pahing
```

```
88888
```

```
88888
```

```
11111
```

```
22222
```

```
33333
```

```
ID : 1
```

```
IS PRIVATE: 0
```

```
| nama: FED
```

```
| Bio Akun:
```

```
| No HP:
```

```
| Weton:
```

```
*****
```

```
*****
```

```
*****
```

```
*****
```

```
*****
```

```
>> MUAT;
```

```
Masukkan nama folder yang hendak dimuat.
```

```
aasa;
```

```
Tidak ada folder yang dimaksud!
```

STEI- ITB	IF2110-TB-I-K02	Halaman 70 dari 89 halaman	
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.			

```

>> MUAT;
Masukkan nama folder yang hendak dimuat.
TESTING1;

Anda akan melakukan pemuatan dari TESTING1.
Mohon tunggu...
1...
2...
3...
Pemuatan selesai!

>> MUAT;
Masukkan nama folder yang hendak dimuat.
TESTING2;

Anda akan melakukan pemuatan dari TESTING2.
Mohon tunggu...
1...
2...
3...
Pemuatan selesai!

>> BALASAN 1;
| ID = 0
| INDRA2
| 24/11/2023 09:15:50
| INI BALASAN UNTUK KICAUAN 1 FED2

| ID = 1
| INDRA2
| 24/11/2023 09:16:24
| INI BALASAN LAGI UNTUK KICAUAN 1 FED2

| ID = 3
| FED2
| 24/11/2023 09:17:44
| INI BALASAN FED2 UNTUK BALASAN INDRA2

| ID = 2
| FED2
| 24/11/2023 09:17:14
| INI BALASAN FED2 UNTUK KICAU FED2

>> BALASAN 2;
| ID = 0
| FED2
| 24/11/2023 09:19:05
| INI BALASAN UNTUK KICAU ID2

| ID = 1
| FED2
| 24/11/2023 09:19:40
| INI BALASAN LAGI UNTUK KICAU ID 2

| ID = 2
| FED2
| 24/11/2023 09:20:10
| INI BALASAN UNTUK BALASAN DENGAN ID 0

```

STEI- ITB	IF2110-TB-I-K02	Halaman 71 dari 89 halaman	
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.			

```

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MUAT;
Masukkan nama folder yang hendak dimuat.
TESTING2;

Anda akan melakukan pemuatan dari TESTING2.
Mohon tunggu...
1...
2...
3...
Pemuatan selesai!

>> MASUK;

Masukkan nama:
FED2;

Masukkan kata sandi:
FED2;

Anda telah berhasil masuk dengan nama pengguna FED2. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> MUAT;
Masukkan nama folder yang hendak dimuat.
TESTING2;

Anda akan melakukan pemuatan dari TESTING2.
Mohon tunggu...
1...
2...
3...
Pemuatan selesai!

>> MASUK;

Masukkan nama:
FED2;

Masukkan kata sandi:
FED2;

Anda telah berhasil masuk dengan nama pengguna FED2. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> KICAUAN;

Detail Kicauan :
| ID = 1
| FED2
| 24/11/2023 08:55:06
| INI KICAUAN FED2
| Disukai : 0

```

6.12 TAGAR

Pada *test* ini akan dilakukan uji coba untuk fitur-fitur yang terdapat di bonus Tagar.

- Fitur yang dites:
 - KICAU
 - CARI_KICAUAN

1. KICAU

Hasil yang Diharapkan:

Ekspektasi saat menjalankan perintah KICAU adalah pengguna dapat memasukkan tagar yang memiliki isi ke dalam kicauan.

STEI- ITB	IF2110-TB-I-K02	Halaman 72 dari 89 halaman	
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.			


```
>> KICAU;
```

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : KICAU2;

Masukkan tagar (maks 280 karakter) : #AKUKEREN1;

Detail Kicauan :

| ID = 2

| A

| 24/11/2023 19:43:16

| KICAU2

| Disukai : 0

- **Tagar kosong**

```
>> KICAU;
```

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : DFBSDFFB;

Masukkan tagar (maks 280 karakter) : ;

Detail Kicauan :

| ID = 1

| A

| 24/11/2023 19:59:48

| DFBSDFFB

| Disukai : 0

Tagar tidak masuk ke dalam kicauan

- **Tidak ada tagar**

```
>> KICAU;
```

Kicauan yang dimasukkan terpotong secara otomatis apabila jumlah karakter lebih dari 280.

Masukkan Kicauan : KICAU;

Masukkan tagar (maks 280 karakter) : ;

Detail Kicauan :

| ID = 2

| A

| 24/11/2023 20:03:11

| KICAU

| Disukai : 0

Tagar tidak masuk ke dalam kicauan

STEI- ITB	IF2110-TB-I-K02	Halaman 73 dari 89 halaman	
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.			

6.13 Kelompok Teman

Pada test ini akan dilakukan uji coba untuk fitur bonus Kelompok Teman.

- Fitur yang dites:
 - KELOMPOK_TEMAN

Hasil yang Diharapkan:

Jika a teman dengan b dan b teman dengan c maka kelompok teman dari a adalah b dan c. Karena b merupakan teman langsung dan c merupakan teman tidak langsung.

```
>> KELOMPOK_TEMAN;  
Terdapat 1 orang dalam kelompok teman a :  
| b
```

```
>> KELOMPOK_TEMAN;  
Terdapat 2 orang dalam kelompok teman a :  
| b  
| c
```

STEI- ITB	IF2110-TB-I-K02	Halaman 74 dari 89 halaman	
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.			

7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	Inisialisasi	Mengecek Program saat dimulai	1. Jalankan program 2. Muat folder yang akan dipakai	6.1	1. Saat folder tidak ada Program memulai dari awal 2. Saat folder ada program akan memuat folder tersebut	1. Sesuai 2. Sesuai
2	MASUK	Mengecek Login akun	1. perintah masuk 2. Masukkan nama akun 3. Masukkan password akun	6.3-2	1. Saat akun tidak ada Program akan keluar dari perintah masuk 2. Saat akun ada, akan meminta password 3. Saat salah password, akan memintan password 4. Saat password benar akan masuk ke program	1. Sesuai 2. Sesuai 3. Sesuai 4. Sesuai
3	DAFTAR	Mengecek daftar akun	1. jalankan perintah daftar 2. masukkan nama 3. masukkan password	6.3-1	1. Saat nama belum ada akan meminta password 2. Saat nama lebih dari 20 akan mengulangi sampai benar 3. Password benar, akun akan terbuat 4. Password lebih dari 20 karakter akan meminta sandi ulang	1. sesuai 2. sesuai 3. sesuai 4. sesuai
4	KELUAR	Mengecek keluar dari akun yang login	1. Jalankan perintah keluar	6.3-3	1. Saat perintah dijalankan, tetapi belum login akan ditampilkan pesan anda belum login 2. Jika akan sudah login dan menjalankan perintah KELUAR, maka pengguna akan keluar dari akun tersebut	1. sesuai 2. sesuai
5	TUTUP_PROGRAM	Digunakan untuk keluar dari program	1. Jalankan perintah TUTUP_PROGR AM	6.3-4	1. Saat pengguna menjalankan perintah, maka program akan tertutup	1. sesuai
6	GANTI_PROFIL	Mengecek ganti profil	1. Masuk ke dalam akun	6.4-3	1. Jika belum masuk ke dalam akun, program	1. sesuai 2. sesuai

STEI- ITB	IF2110-TB-I-K02	Halaman 75 dari 89 halaman	
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.			

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
			2. Jalankan perintah GANTI_PROFIL		akan meminta pengguna untuk login terlebih dahulu 2. Saat perintah dijalankan, pengguna dapat mengganti profil akunya, tidak termasuk foto profil	
7	ATUR_JENIS_AKUN	Mengecek perubahan jenis akun	1. Masuk ke dalam akun 2. Jalankan perintah ATUR_JENIS_AKUN	6.4-4	1. Jika belum masuk ke dalam akun, program akan meminta pengguna untuk login terlebih dahulu 2. Setelah masuk ke dalam akun dan menjalankan perintah ATUR_JENIS_AKUN, pengguna dapat mengubah jenis akunya dari privat menjadi publik maupun sebaliknya.	1. sesuai 2. sesuai
8	UBAH_FOTO_PROFIL	Mengecek perubahan foto profil	1. Masuk ke dalam akun 2. Jalankan perintah UBAH_FOTO_PROFIL	6.4-2	1. Jika belum masuk ke dalam akun, program akan meminta pengguna untuk login terlebih dahulu 2. Setelah masuk ke dalam akun dan menjalankan perintah UBAH_FOTO_PROFIL, maka pengguna dapat mengubah foto profilnya	1. sesuai 2. sesuai
9	LIHAT_PROFIL [NAMA]	Mengecek profil	1. Jalankan perintah LIHAT_PROFIL saat belum login 2. Jalankan perintah LIHAT_PROFIL saat sudah login	6.4-1	1. Jika perintah tersebut dijalankan sebelum pengguna login, maka program akan mengeluarkan pesan kesalahan 2. Jika nama akun yang ingin dilihat profilnya tidak ada, program akan mengeluarkan pesan bahwa akun tersebut tidak ditemukan	1. sesuai 2. sesuai 3. sesuai 4. sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
					3. Jika akun yang ingin dilihat profilnya merupakan akun privat, maka program akan mengeluarkan pesan kesalahan 4. Jika pengguna sudah login dan memasukkan nama akun yang ingin dilihat profilnya dengan benar, maka program akan menampilkan profil dari akun tersebut	
10	BALAS [IDKicau] [IDBalasan]	Mengecek Bisa membuat balas	1. Masuk sebagai akun yang terdaftar 2. Jalankan perintah balas <IDKicau> <IDBalas>	6.8-1	1. Jika balas dengan <IDKicau> dan <IDBalas> yang tidak tepat akan menampilkan pesan balasan tidak ada 2. Jika balas dengan <IDKicau> dan <IDBalas> yang sesuai akan meminta masukkan balasan 3. Menghapus balasan yang tidak ada akan menampilkan pesan kesalahan 4. Menghapus balasan dengan IDKicau dan IDBalas yang tepat akan menghapus balasan dengan IDBalas yang berada di IDKicau 5. Menghapus Balasan yang memiliki balasan akan menghapus balasan yang ada di bawahnya	1. sesuai 2. sesuai 3. sesuai 4. sesuai 5. sesuai 6.
11	BALASAN [IDKicauan]	Mengecek balasan	1. Jalankan perintah BALASAN <IDKicau>	6.8-2	1. saat tidak login akan menunjukkan pesan kesalahan 2. saat login dan <IDKicau> sesuai akan cek seluruh balasan	1. sesuai 2. sesuai 3. sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
					3. saat login dan <IDKicau> tidak ada	
12	HAPUS_BALASAN [IDKicau] [IDBalasan]	Mengecek Hapus Balasan	1. Jalankan perintah HAPUS_KICAUAN <IDKicau> <IDBalasan>	6.8-3	1. Saat tidak ada balasan akan menunjukkan pesan kesalahan 2. Saat balasan ada akan menghapus balasan tersebut 3. Saat balasan dihapus dan dia parent balasan maka childnya akan dihapus	1. sesuai 2. sesuai 3. sesuai
13	BUAT_DRAF	Mengecek Pembuatan Draf	1. Daftar akun 2. Masuk akun 3. Buat draf dengan perintah BUAT_DRAF; 4. Pilih SIMPAN, TERBIT, atau HAPUS	6.9 - 1	1. Saat belum login tidak bisa membuat draf 2. Saat sudah login bisa membuat draf	1. Sesuai 2. Sesuai
14	LIHAT_DRAF	Mengecek setiap user sudah memiliki Draf atau belum	1. Lakukan BUAT_DRAF terlebih dahulu lalu SIMPAN 2. LIHAT_DRAF lalu pilih KEMBALI, TERBIT, HAPUS, UBAH	6.9-2	1. Saat blm login tidak bisa lihat_draf 2. saat draf saat kosong, hanya menampilkan pesan draf tidak ada 3. Saat draf ada, akan menampilkan draf paling atas	1. Sesuai 2. Sesuai 3. Sesuai
15	HAPUS DRAF	Menghapus Draf (jika ada)	1. BUAT_DRAF terlebih dahulu 2. PILIH HAPUS 1. BUAT_DRAF 2. SIMPAN 3. LIHAT_DRAF 4. pilih HAPUS	6.9-6	1. Hapus draf setelah buat draf, draf tidak akan masuk ke dalam stack 2. Hapus draf setelah lihat draf akan menghapus draf paling atas di stack	1. Sesuai 2. Sesuai
16	UBAH DRAF	Mengedit draf yang sudah tersimpan	1. LIHAT_DRAF 2. UBAH_DRAF 3. masukkan perubahan	6.9-4	1. Saat tidak ada draf tidak akan bisa mengubah apapun 2. Saat draf ada dan mengubah draf maka draf akan terupdate sesuai dengan datetima diganti	1. sesuai 2. sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
17	SIMPAN DRAF	Menyimpan Draf yang sudah dibuat	<ol style="list-style-type: none"> 1. BUAT_DRAF 2. SIMPAN 3. LIHAT_DRAF 4. SIMPAN 	6.9-5	<ol style="list-style-type: none"> 1. saat menjalankan buat draf dan langsung simpan maka draf akan tersimpan di data stack 2. Saat lihat draf dan menjalankan perintah kembali maka akan tersimpan kembali di data draf 	<ol style="list-style-type: none"> 1. sesuai 2. sesuai
18	TERBIT DRAF		<ol style="list-style-type: none"> 1. BUAT_DRAF 2. TERBIT_DRAF 3. LIHAT_DRAF 4. TERBIT_DRAF 	6.9-6	<ol style="list-style-type: none"> 1. Saat buat draf dan langsung di terbitkan akan langsung menjadi kicauan 2. Saat lihat draf dan menerbitkan akan menjadi kicauan juga 	<ol style="list-style-type: none"> 1. sesuai 2. sesuai
19	KICAU	Mengecek Pembuatan kicau	<ol style="list-style-type: none"> 1. Masuk ke dalam akun 2. Jalankan perintah KICAU untuk berkicau 	6.7-1	<ol style="list-style-type: none"> 1. Jika belum masuk ke dalam akun, program akan meminta pengguna untuk login terlebih dahulu 2. Setelah masuk ke dalam akun , pengguna dapat mulai berkicau dengan menjalankan perintah KICAU. Setelah menjalankan perintah tersebut, pengguna dapat mulai menulis kicauan. Setelah itu, program akan menampilkan detail kicauan yang dibuat. 3. Kicau juga tidak boleh hanya spasi isinya. Program akan meminta ulang input hingga benar 	<ol style="list-style-type: none"> 1. sesuai 2. sesuai 3. sesuai
20	KICAUAN	Mengecek kicauan	<ol style="list-style-type: none"> 1. Masuk ke dalam akun 2. Jalankan perintah KICAU untuk membuat kicauan 3. Jalankan perintah KICAUAN untuk 	6.7-2	<ol style="list-style-type: none"> 1. Jika belum masuk ke dalam akun, program akan meminta pengguna untuk login terlebih dahulu 2. Jika tidak ada kicauan, maka perintah 	<ol style="list-style-type: none"> 1. sesuai 2. sesuai 3. sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
			melihat daftar kicauan yang pernah dibuat oleh teman, akun publik, dan pengguna		KICAUAN tidak akan menampilkan apa pun 3. Jika terdapat kicauan milik teman, akun publik, ataupun kicauan milik sendiri, maka perintah tersebut akan menampilkan semua detil kicauan tersebut	
21	SUKA_KICAUAN [IDKicau]	Mengecek fungsi suka kicau	1. Masuk ke dalam akun 2. Jalankan perintah SUKA_KICAUAN diikuti dengan ID kicau yang ingin disukai	6.7-3	1. Pengguna hanya dapat menyukai kicauan yang dimiliki oleh akun publik atau teman dari pengguna 2. Jika pengguna menjalankan perintah SUKA_KICAUAN diikuti dengan id kicau yang tidak ada, program akan mengeluarkan pesan kesalahan	1. sesuai 2. sesuai
22	UBAH_KICAUAN [IDKicau]	Mengecek fungsi untuk mengubah kicauan	1. Masuk ke dalam akun 2. Buat kicauan dengan menggunakan perintah KICAU 3. Jalankan perintah UBAH_KICAUAN diikuti dengan id kicau yang ingin diubah	6.7-4	1. Pengguna hanya dapat mengubah kicauan yang dimilikinya sendiri 2. Jika pengguna memasukkan id kicau yang tidak tersedia, program akan mengeluarkan pesan kesalahan 3. Jika pengguna memasukkan id kicau yang dimilikinya sendiri, maka pengguna dapat mengubah detil kicauan tersebut	1. sesuai 2. sesuai 3. sesuai
23	SIMPAN	Mengecek fungsi simpan	1. Jalankan perintah apa pun 2. Jalankan perintah SIMPAN	6.11	1. Hasil yang diharapkan adalah setiap perubahan yang terjadi pada program akan tersimpan pada file config ketika pengguna menjalankan perintah SIMPAN	1. sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
24	MUAT	Mengecek fungsi muat	<ol style="list-style-type: none"> 1. Jika dalam keadaan login, masukkan perintah KELUAR 2. Jalankan perintah MUAT 	6.11-7	<ol style="list-style-type: none"> 1. Jika pengguna dalam keadaan login dan memasukkan perintah login , program akan mengeluarkan pesan kesalahan dan meminta pengguna untuk keluar terlebih dahulu sebelum melakukan MUAT. 2. Jika folder yang ingin dimuat tidak ada, program juga akan mengeluarkan pesan kesalahan dan memberitahu pengguna bahwa tidak ada folder yang dimaksud 3. Jika MUAT berhasil dijalankan, maka program akan menggunakan file <i>config</i> baru dari folder yang dimasukkan 	<ol style="list-style-type: none"> 1. sesuai 2. sesuai 3. sesuai
25	TAGAR	Mengecek fungsi Tagar	<ol style="list-style-type: none"> 1. Jika tagar valid 2. Jika tagar hanya berisi spasi 3. Jika tagar kosong 	6.12	<ol style="list-style-type: none"> 1. Hasil yang diharapkan adalah jika pengguna memasukkan tagar yang valid maka tagar kicauan tersebut akan memiliki tagar 2. Hasil yang diharapkan adalah jika pengguna memasukkan tagar yang hanya berisi spasi maka kicauan tersebut tidak akan memiliki tagar 3. Hasil yang diharapkan jika pengguna memasukkan tagar yang kosong maka kicauan tersebut tidak akan memiliki tagar 	<ol style="list-style-type: none"> 1. sesuai 2. sesuai 3. sesuai
26	KELOMPOK_ PERTEMANA N	Mengecek fungsi kelompok pertemanan	<ol style="list-style-type: none"> 1. a berteman dengan b 2. b berteman dengan c 	6.13	<ol style="list-style-type: none"> 1. Hasil yang diharapkan adalah kelompok teman dari a adalah b dan c karena b merupakan 	<ol style="list-style-type: none"> 1. Sesuai 2. Sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
					teman langsung a dan c merupakan teman tidak langsung dari a	
27	TAMBAH_TEMAN	Mengecek fungsi tambah teman	<ol style="list-style-type: none"> 1. Nama pengguna yang dimasukkan ada 2. Nama pengguna yang dimasukkan tidak ada 	6.6.1	<ol style="list-style-type: none"> 1. Hasil yang diharapkan adalah pengajuan pertemanan masuk 2. Hasil yang diharapkan pengajuan pertemanan tidak masuk ke pengguna 	<ol style="list-style-type: none"> 1. Sesuai 2. Sesuai
28	DAFTAR_TEMAN	Mengecek fungsi daftar teman	<ol style="list-style-type: none"> 1. Pengguna tidak memiliki teman 2. Pengguna memiliki teman 	6.5.1	<ol style="list-style-type: none"> 1. Hasil yang diharapkan adalah pengguna tidak memiliki teman 2. Hasil yang diharapkan adalah pengguna memiliki teman dan menampilkan nama teman 	<ol style="list-style-type: none"> 1. Sesuai 2. Sesuai
29	HAPUS_TEMAN	Mengecek fungsi hapus teman	<ol style="list-style-type: none"> 1. Pengguna memiliki teman 2. Pengguna tidak memiliki teman 	6.5-2	<ol style="list-style-type: none"> 1. Hasil yang diharapkan adalah pengguna berhasil menghapus pertemanan. 2. Hasil yang diharapkan Pengguna belum memiliki teman sehingga tidak ada yang dihapus 	<ol style="list-style-type: none"> 1. Sesuai 2. Sesuai
30	SETUJUI_PERTEMANAN	Mengecek fungsi setuju pertemanan	<ol style="list-style-type: none"> 1. Pengguna tidak ada yang meminta pertemanan 2. Pengguna menerima pertemanan 3. Pengguna tidak menerima pertemanan 	6.6-3	<ol style="list-style-type: none"> 1. Hasil yang diharapkan adalah program memberi tahu pengguna bahwa pengguna blm mendapatkan permintaan pertemanan 2. Pertemanan berhasil disetujui dan jumlah teman bertambah 	<ol style="list-style-type: none"> 1. Sesuai 2. Sesuai

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
31	DAFTAR_PEMINTAAN_PERTEMANAAN	Mengecek fungsi daftar perminaatn pertemanan	<ol style="list-style-type: none"> Pengguna memiliki teman Pengguna tidak memiliki teman 	6.6-2	<ol style="list-style-type: none"> Hasil yang diharapkan adalah semua permintaan pertemanan pengguna terlihat Hasil yang diharapkan bahwa pengguna tidak memiliki permintaan pertemanan 	<ol style="list-style-type: none"> Sesuai Sesuai

8 Pembagian Kerja dalam Kelompok

NIM	Nama	Pembagian Kerja
13522066	Nyoman Ganadipa Narayana	ADT Graf, ADT Tree, ADT String, Teman, Pengguna, Profil, Balasan, Kelompok Teman, Laporan
13522081	Albert	Kicauan, Utas, Tagar, Matriks, ADT List dengan struktur data array statik, Laporan
13522090	Fedrianz Dharma	ADT Mesin Karakter, ADT Mesin Kata, Inisialisasi, Muat, Laporan
13522113	William Glory Henderson	ADT List dengan struktur data array dinamis, ADT Datetime, Permintaan Pertemanan, Utas, Kelompok Teman, README, Laporan
13522119	Indraswara Galih Jayanegara	Draf dengan struktur data berkait, Simpan, ADT Stack, ADT Queue, Laporan

STEI- ITB	IF2110-TB-I-K02	Halaman 84 dari 89 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>			

9 Lampiran

9.1 Deskripsi Tugas Besar 1

Ande-Ande Lumut yang iba mendengar kisah Klenting Kuning kemudian berencana untuk membuat pengganti sosial media yang beken di zaman itu dimana ibu tiri dan saudara jahatnya dicekal dari pendaftaran, yang kelak dinamai BurBir (Burung Biru). Akan tetapi, dia tidak pandai dalam pemrograman bahasa C, lebih-lebih dia acap kali menggunakan Github Copilot dan ChatGPT saat mengerjakan pra-praktikum IF2110 Algoritma & Struktur Data. Kalian, anak buah dari Yuyu Kangkang, diminta untuk membantu Ande-Ande Lumut. Yuk bantu Ande-Ande Lumut membuat aplikasi BurBir di CLI!

9.2 Notulen Rapat

Kertas Form Asistensi :

Form Asistensi Tugas Besar IF2110/Algoritma dan Struktur Data Sem. 1 2023/2024	
No. Kelompok/Kelas	: I / K-02
Nama Kelompok	:
Anggota Kelompok (Nama/NIM)	: 1. Nyoman Ganadipa Narayana (13522066) 2. Albert (13522081) 3. Fedrianz Dharma (13522090) 4. William Glory Henderson (13522113) 5. Indraswara Galih Jayanegara (13522119) 6.
Asisten Pembimbing	: Brianaldo Phuneliarta 13520113

Asistensi 1 :

STEI- ITB	IF2110-TB-I-K02	Halaman 85 dari 89 halaman	
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.			

Asistensi I	
Tanggal: 30 Oktober 2023	Catatan Asistensi:
Tempat: Lab Pro	1. ADT semua dipakai (wajib)
Kehadiran Anggota Kelompok:	2. Prioritas untuk add friend
No	3. Datetime dari time.h buat kecuran
NIM	4. Fokus ke fungsionalitas dan use casenya
Tanda tangan	5. List dinamis buat kecuran
1 13522066	6. ADT Giat boleh pake matrix
<i>[Signature]</i>	7. Ada 2 pertemuan (bonus)
2 13522081	8. Save program cuma di Exit
<i>[Signature]</i>	9. Di laporan tulis list dengan array statik
3 13522090	
<i>[Signature]</i>	
4 13522113	
<i>[Signature]</i>	
5 13522119	
<i>[Signature]</i>	
6	
	Tanda Tangan Asisten: <i>[Signature]</i>

Asistensi 2 :

Asistensi II	
Tanggal: 15 November 2023	Catatan Asistensi:
Tempat: Lab Pro	1. Pastikan bisa jalan di linux ga coba di wsl doang
Kehadiran Anggota Kelompok:	2. kalo bisa baca config di kernel secepatnya biar ga chaos
No	3. Protol masih error
NIM	4. Oke stringnya perlu dibenerin
Tanda tangan	5. kecur cuma bisa teman atau di sendiri yang lihat
1 13522066	6. kasus-kasus kayak private perlu dihandle
<i>[Signature]</i>	7. error handling -1 pas utas
2 13522081	8. req request queue error
<i>[Signature]</i>	
3 13522090	
<i>[Signature]</i>	
4 13522113	
<i>[Signature]</i>	
5 13522119	
<i>[Signature]</i>	
6	
	Tanda Tangan Asisten: <i>[Signature]</i>

9.3 Log Activity Anggota Kelompok

Nama	Minggu 1	Minggu 2	Minggu 3	Minggu 4
Fedrianz	Mengerjakan ADT Mesin Karakter dan ADT Mesin Kata	Mengerjakan ADT Mesin Karakter dan Kata untuk membaca file	Mengerjakan Load	Mengerjakan Load, Laporan, Testing
William	Mengerjakan ADT List dengan struktur data array dinamik, ADT Datetime	Mengerjakan Permintaan Pertemanan	Mengerjakan Utas	Mengerjakan Kelompok Teman, Laporan, Testing
Gana	Mengerjakan ADT Graf, ADT Tree ADT String	Mengerjakan Profil, Pengguna	Mengerjakan Balasan	Mengerjakan Kelompok Teman, Laporan, Testing
Albert	Mengerjakan ADT List dengan struktur data array statik, ADT Matriks	Mengerjakan Kicauan	Mengerjakan Utas	Mengerjakan Tagar, Testing
Indra	Mengerjakan ADT Queue, Stack	Mengerjakan Draf	Mengerjakan Draf, Simpan	Mengerjakan Simpan, Laporan, Testing

9.4 Progress Milestone'

1. Sebelum masa asistensi 2 berakhir, kelompok harus melaporkan *progress* dari pengerjaan tugas besar kepada asisten pembimbing.
2. *Progress* pengerjaan berupa tabel berikut wajib dilampirkan di laporan.
3. *Progress* minimal yang harus dilaporkan kepada asisten pembimbing adalah sebagai berikut:

Tanggal	16-11-2023
----------------	-------------------

STEI- ITB	IF2110-TB-I-K02	Halaman 87 dari 89 halaman	
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.			

No	Fitur	Progress (0-100%)	Keterangan (Komponen fitur yang telah selesai ataupun kendala)
1.	Inisialisasi	90%	Fitur yang belum diimplementasi: 1. Simpan 2. Muat
2.	Perintah	90%	Fitur yang belum diimplementasi: 3. Simpan 4. Muat
3.	Pengguna	100%	Fitur yang telah diimplementasi: 1. Daftar 2. Keluar 3. Masuk 4. Tutup Program
4.	Profil	100%	Fitur yang telah diimplementasi: 1. Ganti Profil 2. Lihat Profil 3. Atur Jenis Akun 4. Ubah Foto Profil
6.	Teman	100%	Fitur yang telah diimplementasi: 1. Daftar Teman 2. Hapus Teman
7.	Permintaan Pertemanan	100%	Fitur yang telah diimplementasi: 1. Tambah Teman 2. Daftar Permintaan Pertemanan 3. Setujui Pertemanan
8.	Kicauan	100%	Fitur yang telah diimplementasi: 1. Kicau 2. Kicauan 3. Suka Kicauan 4. Ubah Kicauan
9.	Balasan	100%	Fitur yang telah diimplementasi: 1. Balas 2. Balasan 3. Hapus Balasan

10.	Draf Kicauan	100%	Fitur yang telah diimplementasi: 1. Buat Draf 2. Lihat Draf
11.	Utas	90%	Fitur yang telah diimplementasi: 1. Utas 2. Sambung Utas 3. Hapus Utas (Masih perlu diupdate) 4. Cetak Utas
12.	Simpan dan Muat	0%	
13.	Laporan	0%	