

LAPORAN TUGAS BESAR

IF2110 Algoritma dan Struktur Data

Burung Biru (BurBir)

Dipersiapkan oleh:

Kelompok F

13320028 Muhammad Sulthan Mazaya

13522018 Ibrahim Ihsan Rasyid

13522019 Wilson Yusda

13522027 Muhammad Althariq Fairuz

13522037 Farhan Nafis Rayhan

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-F-01</i>	92
		Revisi	00
		24-11-2023	

Daftar Isi

1 Ringkasan.....	3
2 Penjelasan Tambahan Spesifikasi Tugas.....	4
2.1 Kelompok Teman.....	5
2.2 Tagar.....	5
2.3 FYB.....	5
3 Struktur Data (ADT).....	6
3.1 ADT Sederhana.....	6
3.2 ADT List dengan Struktur Data Array Statik.....	12
3.3 ADT Matriks.....	13
3.4 ADT List dengan Struktur Data Array Dinamik.....	14
3.5 ADT Mesin Karakter dan Mesin Kata.....	15
3.6 ADT Priority Queue.....	16
3.7 ADT Stack.....	17
3.8 ADT List dengan Struktur Data Berkait.....	18
3.9 ADT Tree.....	19
3.10 ADT Graf dengan representasi Adjacency Matrix.....	20
4 Program Utama.....	20
5 Algoritma-Algoritma Menarik.....	21
Isi dengan penjelasan mengenai algoritma-algoritma menarik yang Anda temukan atau Anda buat dalam Tugas Besar 1, jika ada. Berikan penjelasan singkat, di mana algoritma tersebut digunakan dan mengapa Anda anggap algoritma tersebut menarik.....	21
5.1 <Algoritma 1>.....	21
5.2 <Algoritma 2>.....	21
6 Data Test.....	21
6.1 Test Baca Config.....	21
6.2 Test Daftar, Masuk, dan Keluar.....	22
6.2.1 Test Daftar.....	22
6.2.2 Test Masuk.....	23
6.2.3 Test Keluar.....	23
6.3 Test Profil Akun.....	24
6.3.1 Test GANTI_PROFIL.....	24
6.3.2 Test LIHAT_PROFIL.....	25
6.3.3 Test UBAH_FOTO_PROFIL.....	26
6.3.4 Test ATUR_JENIS_AKUN.....	28
6.4 Test Pertemanan.....	29
6.4.1 Test DAFTAR TEMAN.....	29
6.4.2 Test HAPUS TEMAN.....	32
6.5 Test Permintaan Pertemanan.....	34
6.5.1 Test TAMBAH TEMAN.....	34
6.5.2 Test DAFTAR PERMINTAAN PERTEMANAN.....	37
6.5.3 Test SETUJUI PERTEMANAN.....	39

6.6 Test Kicauan.....	40
6.6.1 Test Kicau.....	40
6.6.2 Test KICAUAN.....	41
6.6.3 Test SUKA_KICAUAN.....	42
6.6.4 Test UBAH_KICAUAN.....	43
6.7 Test Balasan.....	43
6.8 Test Draf Kicauan.....	44
6.8.1 Test BUAT_DRAF.....	44
6.8.2 Test LIHAT_DRAF.....	46
6.9 Test Utas.....	49
6.9.1 Test UTAS.....	49
6.9.2 Test SAMBUNG_UTAS.....	51
6.9.3 Test CETAK_UTAS.....	53
6.9.4 Test HAPUS_UTAS.....	54
6.10 Test Simpan dan Muat.....	55
7 Test Script.....	55
8 Pembagian Kerja dalam Kelompok.....	75
9 Lampiran.....	76
9.1 Deskripsi Tugas Besar 1.....	76
9.2 Notulen Rapat.....	77
9.3 Log Activity Anggota Kelompok.....	78
9.4 Milestone.....	80

1 Ringkasan

Tujuan dari tugas besar kali ini adalah merealisasikan sebuah program berbasis *Command Line Interface* (CLI) yang dikembangkan menggunakan bahasa pemrograman C. Program ini memanfaatkan berbagai struktur data yang telah dipelajari dalam mata kuliah Algoritma dan Struktur Data.

Program ini harus dikompilasi menggunakan *Makefile* dan harus dapat dijalankan pada sistem operasi berbasis UNIX. Untuk Windows, disarankan menggunakan WSL (Windows Subsystem for Linux) untuk mendapatkan lingkungan seperti UNIX. Adapun ADT yang dapat digunakan adalah sebagai berikut:

- 1) ADT Sederhana
- 2) ADT List dengan Struktur Data Array Statik
- 3) ADT Matriks
- 4) ADT List dengan Struktur Data Array Dinamik
- 5) Mesin Karakter dan Mesin Kata
- 6) ADT Priority Queue/ADT Queue (pilih salah satu)
- 7) ADT Stack
- 8) ADT List dengan Struktur Data Berkait
- 9) ADT Tree
- 10) ADT Graf dengan menggunakan Adjacency Matrix

Laporan ini berisi tentang penjelasan dari ADT yang digunakan serta alasan pemilih beserta implementasi dalam pembuatan fitur. Selain itu, program ini menjelaskan cara jalan masing-masing fitur serta bukti screenshot implementasi fitur yang ada.

Kesimpulan dari tugas besar kali ini adalah untuk dapat menggunakan ADT yang telah dibuat dalam merealisasikan fitur sesuai spesifikasi. Kami berhasil merancang program utama ini agar sesuai dan nyaman digunakan oleh pengguna.

2 Penjelasan Tambahan Spesifikasi Tugas

Selain ADT wajib, ada beberapa ADT tambahan yang diimplementasikan pada tugas besar kali ini. Di antaranya adalah ADT Disjoint Set Union (DSU), Hash Table, serta ADT Max Heap. Disjoint Set Union (DSU) digunakan untuk mengimplementasikan fitur “Kelompok Teman” pada bagian bonus. Selain itu, Hash Table digunakan untuk fitur “Tagar”

yang digunakan untuk mencari kicauan tertentu. Terakhir, ADT Max Heap digunakan untuk mengimplementasikan fitur “FYB (For Your Burbir)” yang digunakan untuk menampilkan 8 kicauan dengan *like* tertinggi.

2.1 Kelompok Teman

Disjoint Set Union (DSU) adalah struktur data yang menyimpan sekumpulan elemen yang *disjoint* (tidak beririsan). Struktur data ini mendukung berbagai operasi, seperti menggabungkan *disjoint set* menjadi satu *set disjoint* menggunakan operasi *Union* dan memeriksa apakah dua set adalah *disjoint* atau tidak. Pada tugas besar kali ini, implementasi DSU digunakan untuk fitur “Kelompok Teman”, yaitu fitur untuk menentukan kelompok teman berdasarkan hubungan pertemanan antara pengguna. Dua pengguna dikatakan berada dalam kelompok teman yang sama jika mereka adalah teman secara langsung atau tidak langsung. Pengguna A dan B adalah teman langsung jika B ada dalam daftar teman A dan sebaliknya. Pengguna A dan B adalah teman tidak langsung jika ada Pengguna C sehingga A adalah teman C dan C adalah teman B. Kelompok teman dapat berisi satu pengguna saja. Fitur ini mengidentifikasi kelompok-kelompok sosial dalam jaringan pertemanan.

2.2 Tagar

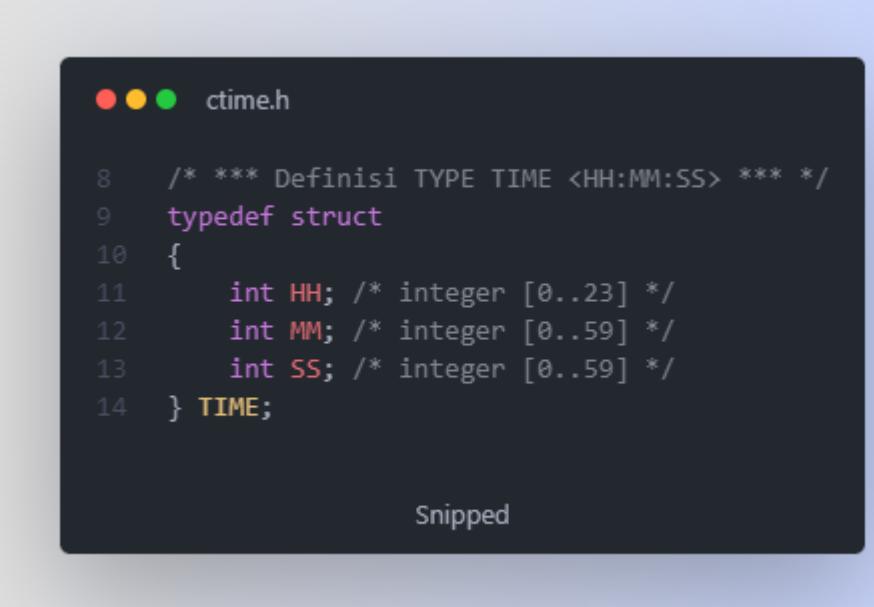
Tagar adalah salah satu fitur yang digunakan untuk menyaring/mencari kicauan dengan lebih cepat dengan bantuan Hash Table sehingga kompleksitas waktunya hanya O(1). Hash Table yang digunakan memiliki kapasitas maksimal 20 dan fungsi Hash yang digunakan adalah mod dari 20. Dua puluh dipilih sebagai kapasitas maksimal hash table mengikuti jumlah maksimum pengguna. Selain itu, fungsi mod 20 dipilih karena mengikuti jumlah maksimum kapasitas dari Hash Table itu sendiri.

2.3 FYB

For Your Burbir (FYB) adalah fitur yang digunakan untuk menampilkan delapan kicauan dengan *like* tertinggi. Kicauan yang ditampilkan juga merupakan kicauan umum dan jika ada kicauan dengan jumlah *like* yang sama, kicauan yang diterbitkan lebih dulu akan ditampilkan paling atas. Dalam implementasinya, digunakan bantuan ADT Max Heap..

3 Struktur Data (ADT)

3.1 ADT Sederhana

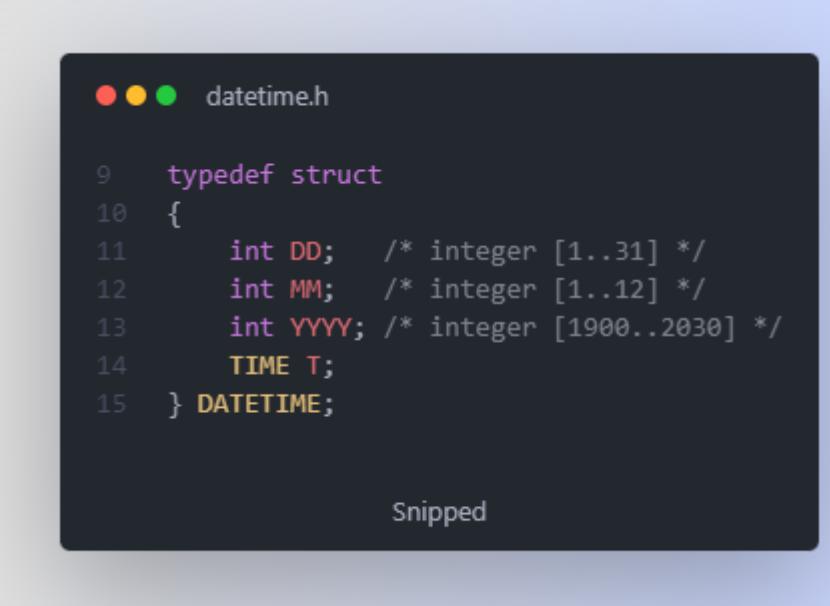


```
ctime.h

8  /* *** Definisi TYPE TIME <HH:MM:SS> *** */
9  typedef struct
10 {
11     int HH; /* integer [0..23] */
12     int MM; /* integer [0..59] */
13     int SS; /* integer [0..59] */
14 } TIME;
```

Snipped

Gambar 3.1 Struktur ADT Time



```
datetime.h

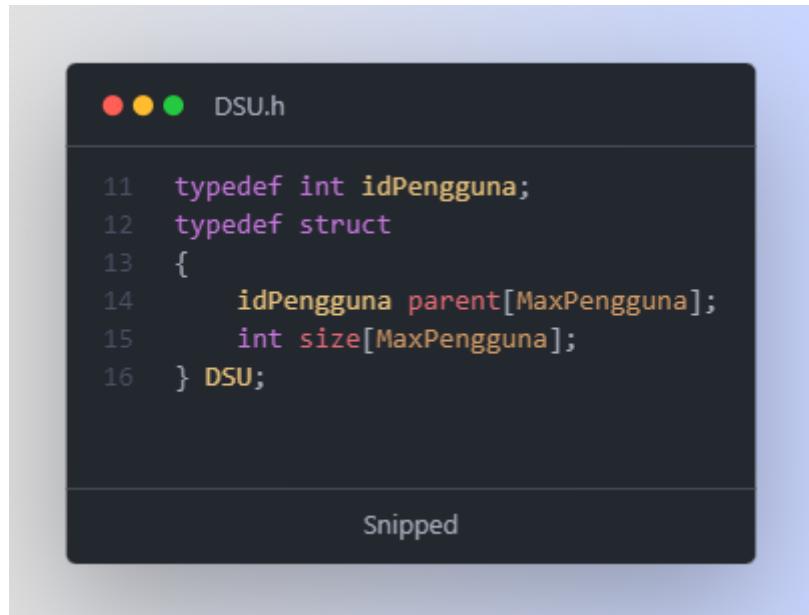
9  typedef struct
10 {
11     int DD; /* integer [1..31] */
12     int MM; /* integer [1..12] */
13     int YYYY; /* integer [1900..2030] */
14     TIME T;
15 } DATETIME;
```

Snipped

Gambar 3.2 Struktur ADT Datetime

ADT Sederhana terdiri dari berbagai jenis ADT diluar dari ADT yang diwajibkan. Pertama ada ADT Datetime mendefinisikan tipe data DATETIME yang merupakan struktur yang terdiri dari empat bagian: DD (hari), MM (bulan), YYYY (tahun), dan T (waktu). Tipe

data TIME sendiri adalah struktur yang berisi informasi tentang HH (jam), MM (menit), dan SS (detik). ADT ini diimplementasikan pada file *ctime.h* dan *datetime.h*. ADT ini digunakan pada semua fitur yang memerlukan waktu pengiriman informasi dalam format DATETIME seperti Kicauan, Utasan, Draf, dan sebagainya. Alasan pemilihan ADT ini yakni karena perlunya pengambilan informasi waktu dalam format yang lengkap (DD/MM/YYYY HH:MM:SS) sehingga ADT ini memudahkan implementasi dalam fitur-fitur lainnya.



The screenshot shows a code editor window titled "DSU.h". The code defines a structure named DSU with the following members:

```
11  typedef int idPengguna;
12  typedef struct
13  {
14      idPengguna parent[MaxPengguna];
15      int size[MaxPengguna];
16  } DSU;
```

A "Snipped" indicator is visible at the bottom right of the code block.

Gambar 3.3 Struktur ADT DSU

ADT DSU (Disjoint Set Union) digunakan untuk menyimpan dan memanipulasi data pengguna dalam bentuk *disjoint set*. ADT ini mendefinisikan tipe data DSU yang merupakan struktur yang terdiri dari parent dan size yang masing-masing merupakan array dengan ukuran maksimum MaxPengguna. ADT ini dapat digunakan untuk memecahkan berbagai persoalan yang melibatkan pengelolaan data pengguna, seperti mengecek apakah dua pengguna berada dalam suatu “Kelompok Pertemanan” atau set yang sama. Alasan pemilihan ADT ini karena kemudahan dan fleksibilitas yang ditawarkannya dalam memanipulasi data dalam bentuk *disjoint set*. ADT ini diimplementasikan pada file *dsu.h*.



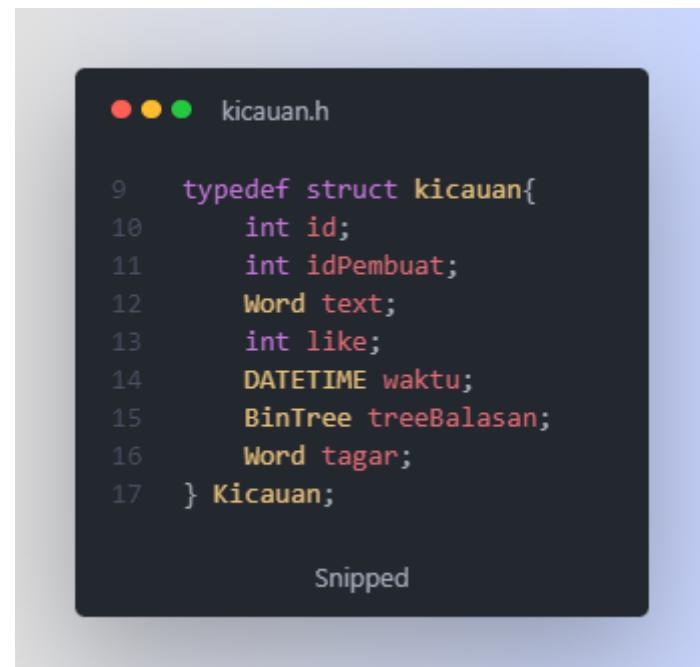
```
● ○ ● balasan.h

8     typedef struct balasan{
9         int id;
10        int idPembuat;
11        Word author;
12        Word text;
13        DATETIME waktu;
14    } Balasan;
```

Snipped

Gambar 3.4 Struktur ADT Balasan

ADT Balasan merupakan struktur data yang menyimpan balasan untuk suatu kicauan. ADT ini terdiri dari kicauanID yang menyimpan Id dari kicauan serta AddressKicauanNode yang berisi address dari balasan selanjutnya. Alasan penggunaan ADT ini adalah untuk memudahkan manajemen balasan dari suatu kicauan. ADT ini diimplementasikan dalam file *balasan.h*



```
● ○ ● kicauan.h

9     typedef struct kicauan{
10        int id;
11        int idPembuat;
12        Word text;
13        int like;
14        DATETIME waktu;
15        BinTree treeBalasan;
16        Word tagar;
17    } Kicauan;
```

Snipped

Gambar 3.5 Struktur ADT Kicauan

ADT Kicauan merupakan struktur data yang digunakan untuk menyimpan kicauan dari suatu pengguna. ADT Kicauan terdiri dari id, idPembuat, text (isi kicauan), *like*, waktu diterbitkan, treeBalasan (berisi balasan dari kicauan), dan tagar. Alasan diimplementasikannya ADT ini adalah untuk memudahkan manajemen kicauan tiap pengguna. ADT ini diimplementasikan dalam file *kicauan.h*



The screenshot shows a terminal window with a dark background. At the top, there are three colored window control buttons (red, yellow, green) followed by the file name "maxheap.h". Below this, the code for the MaxHeap structure is displayed:

```
9  typedef struct {
10    Kicauan *kicauan;
11    int capacity;
12    int size;
13 } MaxHeap;
```

At the bottom right of the terminal window, the word "Snipped" is visible.

Gambar 3.6 Struktur ADT MaxHeap

ADT MaxHeap merupakan ADT yang digunakan untuk mengimplementasikan fitur tambahan FYB. ADT MaxHeap terdiri dari pointer kicauan, capacity, serta size. ADT ini diimplementasikan karena kemudahan yang ditawarkannya dalam mengorganisir kicauan dengan *like* tertinggi. ADT ini diimplementasikan pada file *maxheap.h*.



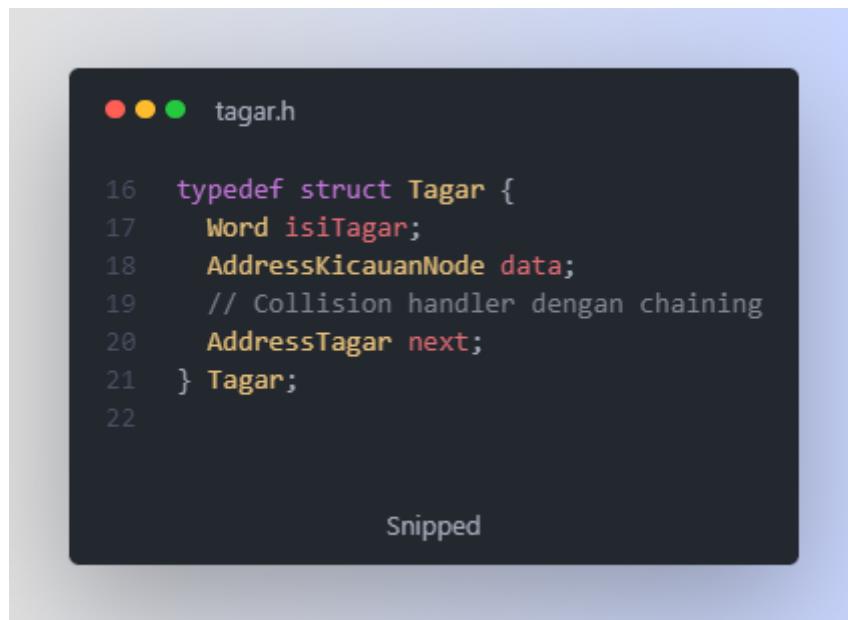
The screenshot shows a code editor window with a dark theme. At the top, there are three colored circular icons (red, yellow, green) followed by the file name "pengguna.h". The code itself is a C-style struct definition:

```
9  typedef struct pengguna {
10     boolean isValid;
11     int id;
12     Word Nama;
13     Word KataSandi;
14     Word NoHP;
15     Word BioAkun;
16     Word Weton;
17     Word JenisAkun;
18     Matriks FotoProfil;
19     PriorityQueue PermintaanBerteman;
20     Stack Draf;
21 } Pengguna;
```

In the bottom right corner of the code editor window, the word "Snipped" is visible.

Gambar 3.7 Struktur ADT Pengguna

ADT Pengguna merupakan ADT yang berisi data dari suatu pengguna. ADT ini sendiri terdiri dari informasi dari suatu pengguna, seperti id, Nama, KataSandi, NoHP, BioAkun, Weton, JenisAkun, FotoProfil, PermintaanBerteman, Draf, serta isValid yang digunakan untuk menentukan apakah user valid atau tidak. ADT ini digunakan karena memberikan kemudahan dalam mendefinisikan suatu pengguna. ADT ini diimplementasikan dalam file *pengguna.h*.



The screenshot shows a code editor window with a dark theme. At the top left, there are three colored dots (red, yellow, green) followed by the text "tagar.h". The code itself is as follows:

```
16  typedef struct Tagar {
17      Word isiTagar;
18      AddressKicauanNode data;
19      // Collision handler dengan chaining
20      AddressTagar next;
21  } Tagar;
22
```

In the bottom right corner of the code area, the word "Snipped" is visible.

Gambar 3.8 Struktur ADT Tagar

ADT Tagar merupakan ADT yang digunakan untuk mengimplementasikan fitur tambahan, yaitu “Tagar”. ADT ini terdiri dari isiTagar, data, serta next yang merupakan address dari tagar selanjutnya. Alasan digunakannya ADT ini karena kemudahan yang diberikannya dalam mengatur tagar-tagar yang ada. ADT ini diimplementasikan pada file *tagar.h*.



The screenshot shows a code editor window with a dark theme. At the top left, there are three colored dots (red, yellow, green) followed by the text "tagar.h". The code itself is as follows:

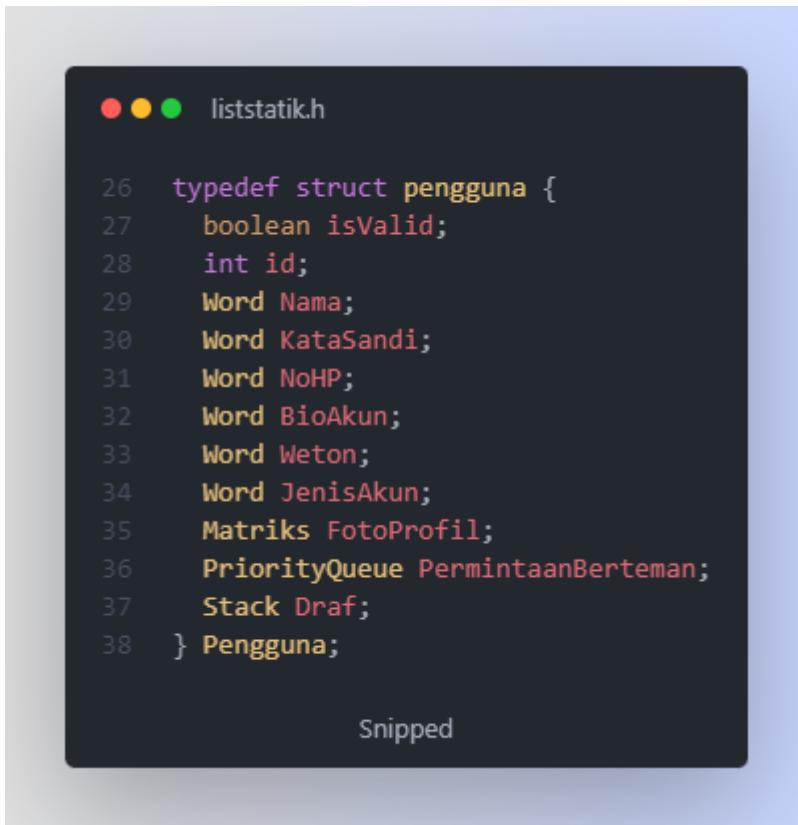
```
9   typedef struct KicauanNode {
10     int kicauanId;
11     AddressKicauanNode next;
12 } KicauanNode;
```

In the bottom right corner of the code area, the word "Snipped" is visible.

Gambar 3.9 Struktur ADT KicauanNode

ADT KicauanNode merupakan ADT yang berisi Node dari suatu kicauan, ADT ini terdiri dari kicauanId dan next yang merupakan address selanjutnya dari node suatu kicauan. Alasan digunakannya ADT ini adalah karena kemudahannya dalam manajemen tagar. ADT ini diimplementasikan pada file yang sama dengan ADT Tagar, yaitu *tagar.h*.

3.2 ADT List dengan Struktur Data Array Statik



```
liststatik.h

26  typedef struct pengguna {
27      boolean isValid;
28      int id;
29      Word Nama;
30      Word KataSandi;
31      Word NoHP;
32      Word BioAkun;
33      Word Weton;
34      Word JenisAkun;
35      Matriks FotoProfil;
36      PriorityQueue PermintaanBerteman;
37      Stack Draf;
38 } Pengguna;

Snipped
```

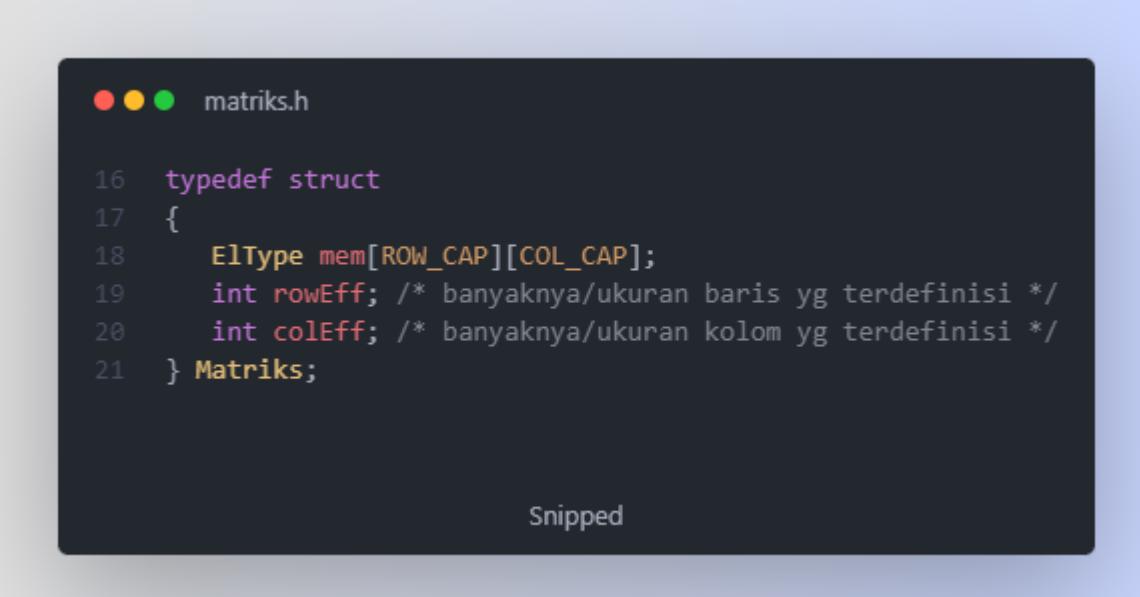
Gambar 3.10 Struktur ADT List Statik

ADT List dengan struktur Data Array Statik digunakan untuk menyimpan dan memanipulasi data pengguna dalam bentuk list. ADT ini mendefinisikan tipe data Pengguna yang merupakan struktur yang terdiri dari berbagai atribut pengguna seperti Nama, KataSandi, NoHP, BioAkun, Weton, JenisAkun, FotoProfil, dan PermintaanBerteman,Draf. Tipe data ListPengguna adalah array dari Pengguna. ADT ini diimplementasikan pada file *liststatik.h*. ADT ini menyelesaikan semua persoalan yang memerlukan kehadiran informasi pribadi pengguna seperti Nama (Dalam validasi apakah milik pribadi atau bukan), Draf

(Dalam fitur Draf), PermintaanBerteman (Dalam fitur Teman dan Permintaan Pertemanan), dan JenisAkun (Validasi apakah suatu akun privat atau tidak).

Alasan pemilihan ADT ini karena adanya kejelasan bahwa jumlah pengguna maksimum adalah 20. Oleh karena itu, tidak perlu memperhatikan penambahan ruang dalam array sehingga aplikasi fitur akan lebih nyaman jika dengan menggunakan List Statik.

3.3 ADT Matriks



```
matriks.h

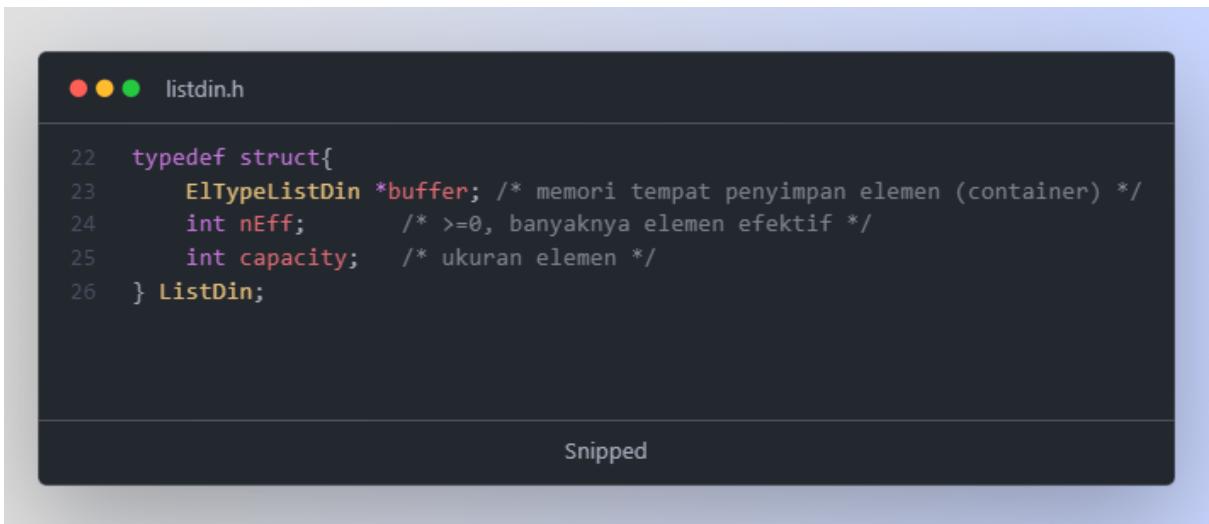
16  typedef struct
17  {
18      ElType mem[ROW_CAP][COL_CAP];
19      int rowEff; /* banyaknya/ukuran baris yg terdefinisi */
20      int colEff; /* banyaknya/ukuran kolom yg terdefinisi */
21  } Matriks;

Snipped
```

Gambar 3.11 Struktur ADT Matriks

ADT Matriks digunakan untuk menyimpan dan memanipulasi data dalam bentuk matriks. ADT Matriks digunakan untuk merepresentasikan foto profil dari suatu pengguna. Definisi Tipe Data: ADT ini mendefinisikan tipe data Matriks yang merupakan struktur yang terdiri dari array dua dimensi mem yang menyimpan elemen, dan rowEff dan colEff yang menyimpan banyaknya baris dan kolom yang terdefinisi. ADT Matriks diimplementasikan pada file *matriks.h*.

3.4 ADT List dengan Struktur Data Array Dinamik



```
listdin.h

22  typedef struct{
23      ElTypeListDin *buffer; /* memori tempat penyimpan elemen (container) */
24      int nEff;           /* >=0, banyaknya elemen efektif */
25      int capacity;      /* ukuran elemen */
26  } ListDin;
```

Snipped

Gambar 3.12 Struktur ADT List Dinamik

ADT List dengan Struktur Data Array Dinamik merupakan list dinamis untuk menyimpan dan memanipulasi data secara dinamis. ADT ini dapat digunakan untuk memecahkan berbagai persoalan yang melibatkan pengelolaan kicauan, seperti menambahkan dan menghapus kicauan. Alasan pemilihan ADT ini adalah karena kemudahan dan fleksibilitas yang ditawarkannya dalam memanipulasi data dalam bentuk list dinamis. ADT ini mendefinisikan tipe data ListDin yang merupakan struktur yang terdiri dari buffer yang merupakan array dinamis untuk menyimpan elemen, nEff yang menyimpan banyaknya elemen efektif, dan capacity yang menyimpan kapasitas maksimum list. ADT ini diimplementasikan pada file *listdin.h*.

3.5 ADT Mesin Karakter dan Mesin Kata

```
● ● ● wordmachine.h

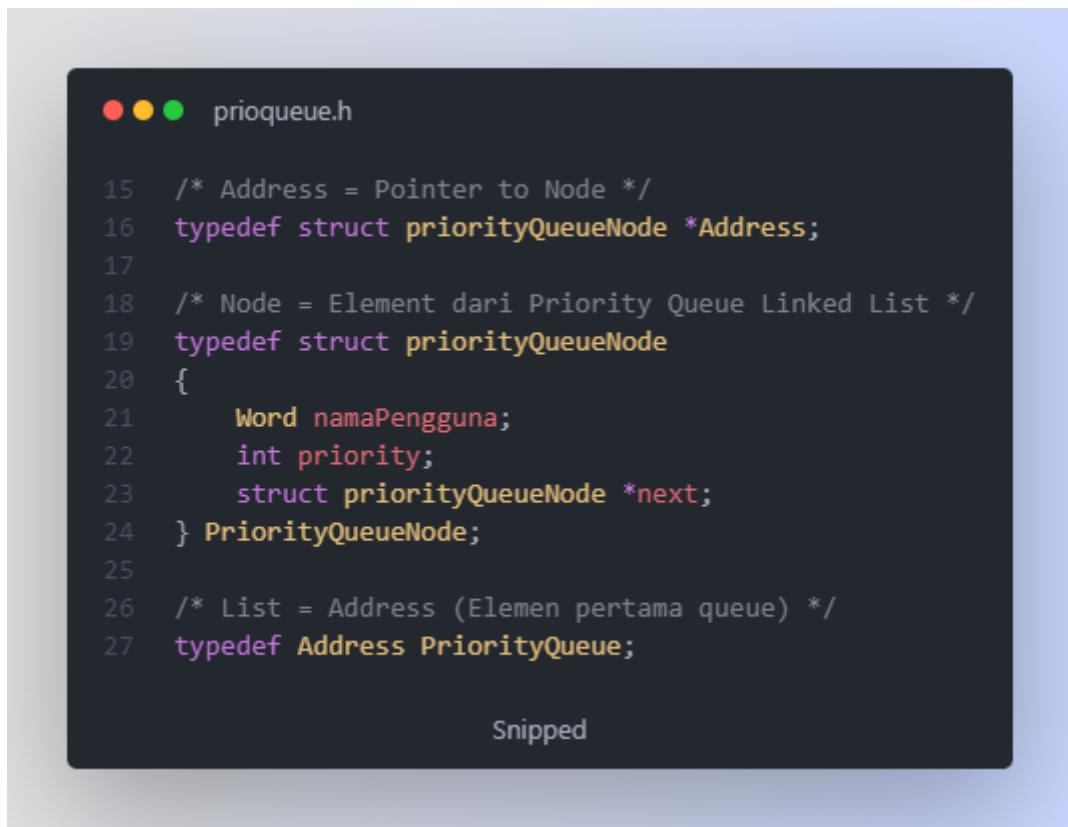
13  typedef struct {
14      char TabWord[NMax]; /* container penyimpan kata, indeks yang dipakai
15                               [0..NMax-1] */
16      int Length;
17  } Word;
18
19  /* State Mesin Word */
20  extern boolean EndWord;
21  extern Word currentWord;
```

Snipped

Gambar 3.13 Struktur ADT Mesin Kata

ADT Mesin Kata merupakan lanjutan dari ADT Mesin Karakter. ADT Mesin Karakter digunakan untuk memecahkan berbagai persoalan yang melibatkan pembacaan dan pemrosesan input karakter demi karakter, seperti membaca dan memproses perintah yang dimasukkan oleh pengguna atau membaca dan memproses file teks, sedangkan ADT Mesin Kata digunakan untuk menyimpan dan memanipulasi kata atau string. ADT Mesin Kata ini mendefinisikan tipe data Word yang merupakan struktur yang terdiri dari array karakter TabWord dan panjang kata Length. ADT ini diimplementasikan dalam file *charmachine.h* dan *wordmachine.h*.

3.6 ADT Priority Queue



```
● ● ● prioqueue.h

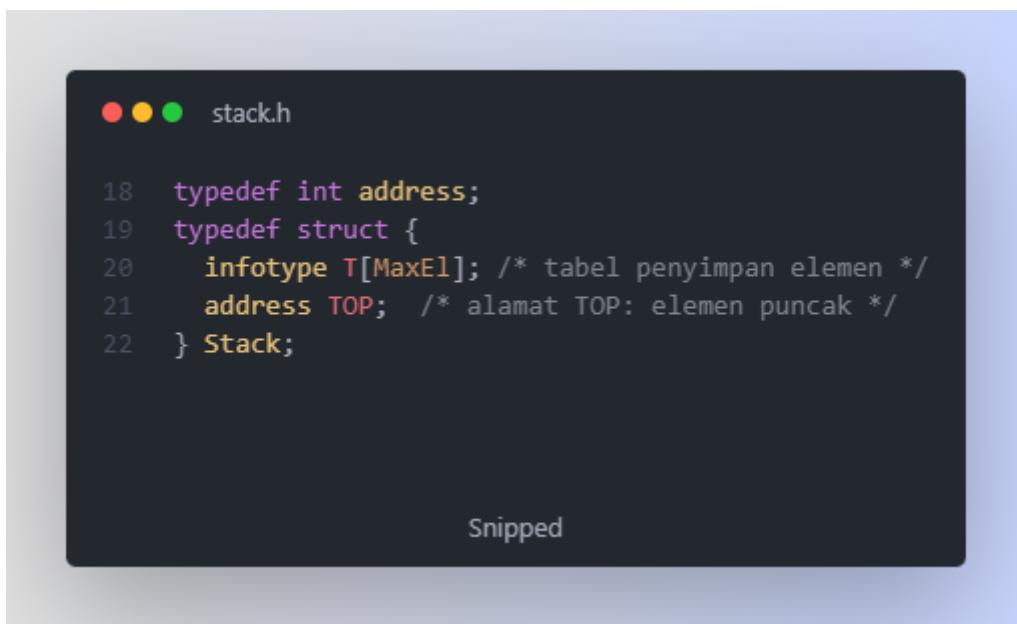
15  /* Address = Pointer to Node */
16  typedef struct priorityQueueNode *Address;
17
18  /* Node = Element dari Priority Queue Linked List */
19  typedef struct priorityQueueNode
20  {
21      Word namaPengguna;
22      int priority;
23      struct priorityQueueNode *next;
24  } PriorityQueueNode;
25
26  /* List = Address (Elemen pertama queue) */
27  typedef Address PriorityQueue;

Snipped
```

Gambar 3.14 Struktur ADT Priority Queue

ADT Priority Queue digunakan untuk menyimpan dan memanipulasi data pengguna dalam bentuk antrian prioritas berdasarkan jumlah pengikut/teman. ADT ini mendefinisikan tipe data PriorityQueueNode yang merupakan elemen dari Priority Queue Linked List. Setiap PriorityQueueNode berisi namaPengguna, priority, dan pointer next ke PriorityQueueNode berikutnya. Tipe data PriorityQueue adalah pointer ke elemen pertama queue. Alasan pemilihan ADT Priority Queue dibandingkan ADT Queue biasa karena ADT Priority Queue memberikan kemudahan dan fleksibilitas yang jauh lebih baik dalam memanipulasi data dalam bentuk antrian prioritas dibandingkan ADT Queue biasa. ADT Priority Queue diimplementasikan pada file *prioqueue.h*.

3.7 ADT Stack



```
stack.h

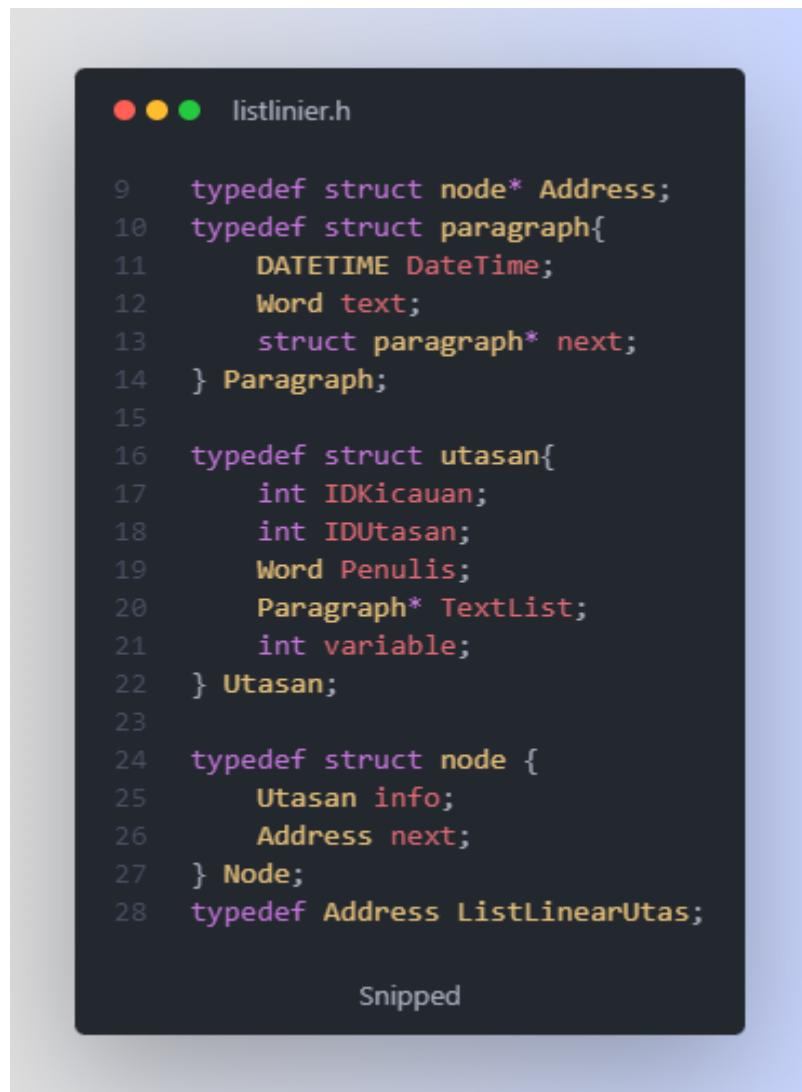
18  typedef int address;
19  typedef struct {
20      infotype T[MaxEl]; /* tabel penyimpan elemen */
21      address TOP; /* alamat TOP: elemen puncak */
22  } Stack;

Snipped
```

Gambar 3.15 Struktur Data ADT Stack

ADT ini dapat digunakan untuk memecahkan berbagai persoalan yang melibatkan pemrosesan data dalam bentuk LIFO (Last In First Out), hal ini diperlukan untuk mengimplementasikan fitur ‘Draf Kicauan’. Setiap draf yang baru akan di *push* ke *top stack* sehingga hanya draf yang terakhir kali dibuat yang bisa dilihat, diedit, ataupun dihapus. ADT Stack mendefinisikan tipe data Stack yang merupakan struktur yang terdiri dari array T yang menyimpan elemen dan TOP yang menunjuk ke elemen puncak. Alasan pemilihan ADT ini adalah karena kemudahan dan fleksibilitas yang ditawarkannya dalam memanipulasi data dalam bentuk tumpukan atau *stack*. ADT ini diimplementasikan dalam file *stack.h*.

3.8 ADT List dengan Struktur Data Berkait



```
listlinier.h

9  typedef struct node* Address;
10 typedef struct paragraph{
11     DATETIME DateTime;
12     Word text;
13     struct paragraph* next;
14 } Paragraph;
15
16 typedef struct utasan{
17     int IDKicauan;
18     int IDUtasan;
19     Word Penulis;
20     Paragraph* TextList;
21     int variable;
22 } Utasan;
23
24 typedef struct node {
25     Utasan info;
26     Address next;
27 } Node;
28 typedef Address ListLinearUtas;

Snipped
```

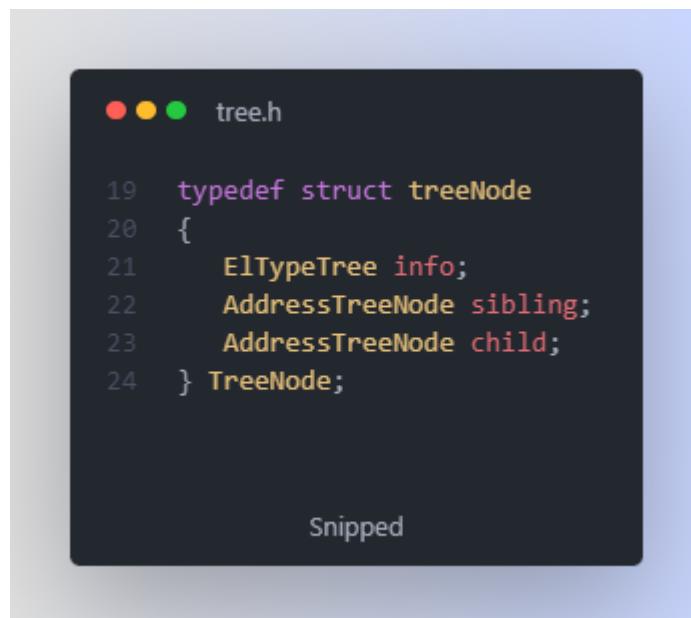
Gambar 3.16 Struktur Data ADT List Dengan Data Berkait

ADT List dengan Struktur Data Berkait digunakan untuk menyimpan dan memanipulasi data utasan. ADT ini mendefinisikan tipe data Utasan yang merupakan struktur yang terdiri dari berbagai atribut utasan seperti IDKicauan, IDUtasan, Penulis, TextList. Tipe data ListLinearUtas adalah pointer ke elemen pertama list. ADT ini dapat digunakan untuk memecahkan berbagai persoalan yang melibatkan pengelolaan data utasan, seperti menambahkan dan menghapus utasan. ADT ini diimplementasikan pada file *listlinier.h*.

Alasan Pemilihan ADT List dengan Struktur Data Berkait dalam penerapan fitur utasan salah satunya yaitu karena perlunya fungsi *deleteParagraphAtPosition* yang pengaplikasiannya lebih mudah jika digunakan List dengan Struktur Data Berkait. Selain itu,

karena ukuran utasan yang tidak ada batasnya, maka dari itu penggunaan List Dinamis dan List dengan Struktur Data Berkait lebih cocok. Untuk menghindari adanya bentrokan dengan rekan lain yang mengerjakan List Dinamis. Kami merasa agar lebih baik untuk mengaplikasikan konsep dengan Struktur Data Berkait (sebagai penyimpan index utasan) dalam List dengan Struktur Data Berkait (sebagai kerangka atas utas yang dibuat) dalam pengaplikasian fitur Utasan.

3.9 ADT Tree



```
tree.h

19  typedef struct treeNode
20  {
21      ElTypeTree info;
22      AddressTreeNode sibling;
23      AddressTreeNode child;
24  } TreeNode;
```

Snipped

Gambar 3.17 Struktur ADT Tree

ADT Tree digunakan untuk mengimplementasikan fitur balasan, yaitu fitur untuk mengomentari sebuah kicauan. ADT Tree sendiri terdiri dari ElTypeTree yang berisi Balasan yang disimpan di info, address dari saudara current node yang disimpan di sibling, serta address dari child node dari current node yang disimpan di child. Alasan pemilihan ADT Tree adalah karena hierarkinya. Setiap kicauan dapat dipandang sebagai parent dan setiap balasan dapat dipandang sebagai child dari sebuah kicauan. Setiap kicauan dapat terdiri lebih dari satu balasan/child. Dengan demikian, struktur data jadi lebih teratur.

3.10 ADT Graf dengan representasi Adjacency Matrix



```
graph.h
21 /* Define struktur untuk graph */
22 typedef struct graph
23 {
24     int simpul; // simpul = jumlah user
25     int adjMatrix[MAX_SIMPUL][MAX_SIMPUL]; // adjacency matrix yg berisi pointer to integer untuk menyimpan hubungan pertemanan
26 } Graph;
27
```

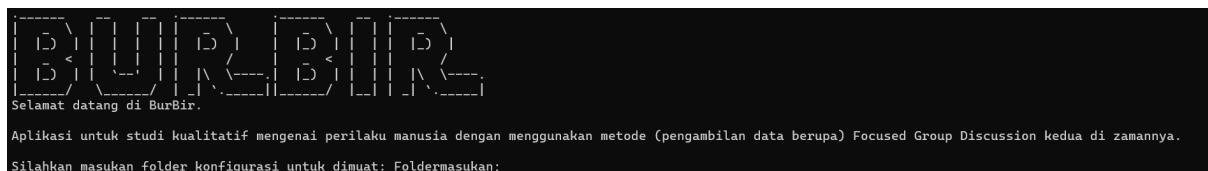
Snipped

Gambar 3.18 Struktur ADT Graph dengan Adjacency Matrix

ADT Graph digunakan untuk menyimpan hubungan/koneksi pertemanan dari suatu pengguna dalam bentuk *adjacency matrix* (matriks kedekatan). ADT Graf mendefinisikan tipe data Graph yang merupakan struktur yang terdiri dari simpul yang menyimpan jumlah pengguna dan adjMatrix yang merupakan *adjacency matrix* untuk menyimpan hubungan pertemanan antar pengguna. Alasan pemilihan ADT Graf sendiri karena ADT Graf memberikan kemudahan dalam memanipulasi hubungan antar pengguna. ADT ini diimplementasikan pada file *graph.h*.

4 Program Utama

Pada program utama, saat pertama kali dijalankan, program akan menampilkan tulisan BurBir. Kemudian, program akan meminta input file konfigurasi yang akan dimuat. Jika file berhasil dimuat, pengguna dapat mendaftarkan akun baru ataupun masuk dengan akun yang sudah ada. Setelah pengguna berhasil masuk, pengguna dapat menggunakan berbagai fitur yang ada, seperti tambah teman, hapus teman, membuat kicauan, membalas sebuah kicauan, dan lain-lain. Jika pengguna ingin mengganti akun, pengguna harus keluar dari akun yang digunakan sekarang dan masuk lagi dengan akun yang diinginkan. Pengguna juga dapat menyimpan kondisi BurBir saat ini dalam bentuk file konfigurasi dengan syarat pengguna belum masuk ke dalam akun. Pengguna juga dapat memuat konfigurasi BurBir yang telah disimpan sebelumnya.



5 Algoritma-Algoritma Menarik

5.1 Disjoint Set Union Find

Digunakan Disjoint Set Union (DSU) untuk merepresentasikan kelompok teman sebagai fitur bonus. Algoritma ini menarik karena sebuah member dari suatu set direpresentasikan dengan *root* yang sama. Agar dapat memahami tentang root, perlu ditinjau tersebut dahulu bagaimana cara membuat dan *merge* sebuah set pada DSU. Dapat dilihat sebagai berikut.

```
● ● ●

void createSet(DSU *d, idPengguna id) {
    (*d).parent[id] = id;
    (*d).size[id] = 1;
}

void mergeSet(DSU *d, idPengguna s1, idPengguna s2) {
    if (!isSameRoot(*d, s1, s2)) {
        if (d->size[s1] > d->size[s2]) {
            d->parent[s2] = s1;
            d->size[s1] += d->size[s2];
        } else {
            d->parent[s1] = s2;
            d->size[s2] += d->size[s1];
        }
    }
}
```

Dapat diperhatikan bahwa membuat sebuah set baru berarti menyimpan sebuah nilai pada index yang merupakan nilai itu sendiri. Saat dilakukan merge, maka size akan diperbanyak sesuai banyaknya set dan parent dari yang memiliki member yang lebih banyak. Set index yang merupakan nilai itu sendiri merupakan cara agar dapat dilakukan traversal untuk mengecek root dari sebuah member set. Dapat dilakukan dengan cara sebagai berikut.

```
idPengguna findRoot(DSU d, idPengguna id) {
    while (id != d.parent[id]) {
        id = d.parent[id];
    }
    return id;
}
```

Pada kasus kelompok teman, agar dapat mendapatkan semua pengguna yang berteman, maka dapat melakukan traversal dari semua member set dan melihat apakah masing-masing elemen memiliki root yang sama atau tidak. Bila ya, maka elemen-elemen tersebut merupakan kelompok teman. Berikut merupakan detail implementasi dari cara untuk melihat seluruh pengguna yang merupakan kelompok teman.

```
void printKelompokTeman(ListStatik listUser, DSU d, Pengguna currentUser) {
    int length = getLength(d, currentUser.id);
    printf("Terdapat %d orang dalam Kelompok Teman ", length);
    PrintWord(currentUser.Nama);
    printf(":\\n");
    int id = currentUser.id;
    idPengguna root = findRoot(d, id);
    for (int i = 0; i < MaxPengguna; i++) {
        if (d.parent[i] == root) {
            Pengguna p;
            GetUserById(listUser, &p, i);
            PrintWord(p.Nama);
            printf("\\n");
        }
    }
    printf("\\n");
}
```

5.2 Max Heap Update

Max Heap merupakan sebuah data struktur yang dapat diakses nilai maksimalnya secara sangatlah efisien. Hal ini dikarenakan pada *insert*-pun heap sudah diorganisir agar merupakan tree yang yang masing-masing root dari subtree-nya (maupun tree-nya itu sendiri) merupakan nilai tersebesar dibanding seluruh anak-anaknya. Proses memasukkan sebuah elemen kepada heap disebut sebagai Heapify, berikut merupakan implementasi heapify pada kasus max heap yang digunakan untuk fitur FYB (For Your Burbur).

```
● ● ●

void heapifyDown(MaxHeap *heap, int index) {
    int left = 2 * index + 1;
    int right = 2 * index + 2;
    int largest = index;

    if (left < heap->size && heap->kicauan[left].like > heap-
>kicauan[largest].like) {
        largest = left;
    }

    if (right < heap->size && heap->kicauan[right].like > heap-
>kicauan[largest].like) {
        largest = right;
    }

    if (largest != index) {
        swap(&heap->kicauan[index], &heap->kicauan[largest]);
        heapifyDown(heap, largest);
    }
}
```

Secara singkat, heapify ini melakukan panggilan rekursi yang melakukan pergeseran apabila terdapat elemen yang lebih besar dibanding elemen-elemen dibawahnya. Heapify ini juga digunakan untuk update heap apabila terdapat update like sebagai berikut.

```
void updateLikeInHeap(MaxHeap *heap, int kicauanId) {
    for (int i = 0; i < heap->size; i++) {
        if (heap->kicauan[i].id == kicauanId) {
            heap->kicauan[i].like++;
            heapifyDown(heap, 0);
            return;
        }
    }
}
```

Lalu, untuk mendapatkan nilai FYB dengan 8 like terbesar dapat dilakukan dengan algoritma sebagai berikut.

```
void tampilkanKicauanTerbaik(MaxHeap heap, ListStatik listUser) {
    MaxHeap *copiedHeap = copyMaxHeap(&heap);
    int totalKicauan = heap.size < 8 ? heap.size : 8;
    printf("Berikut %d kicauan dengan like tertinggi di FYB\n\n", totalKicauan);

    for (int i = 0; i < totalKicauan; i++) {
        Kicauan kicauan = extractMax(copiedHeap);
        showKicauan(kicauan);
    }
}
```

ExtractMax dilakukan dengan cara mengambil head yang ada pada heap. Sehingga, dipastikan sebagai elemen dengan nilai terbesar.

6 Data Test

Berikut adalah data test yang digunakan beserta fitur yang diujikan dan beberapa penjelasan mengenai fitur tersebut. Semua pengujian dilakukan di WSL.

6.1 Test Baca Config

Fitur yang dites adalah fitur untuk membaca file *config*. Harapannya adalah program mampu membaca file *config* yang sudah ada. Hasil yang diharapkan adalah sebagai berikut:

```
----- -- -- -----  
| _ \ | | | | | | _ \ | _ \ | | | | _ \ | | | | | | | |
| |_) | | | | | | | |_) | | | | | | | |_) | |  
| _ < | | | | | | | / | | _ < | | | | | | | / |  
| |_) | | | `--' | | | | \ | \----.| |_) | | | | | | \ |  
\----.  
|_____| / \_____| / | _| ` ._____| |_____| / |__| | _|  
` ._____|
```

Selamat datang di BurBir.

Aplikasi untuk studi kualitatif mengenai perilaku manusia dengan menggunakan metode (pengambilan data berupa) Focused Group Discussion kedua di zamannya.

Silahkan masukan folder konfigurasi untuk dimuat: tes.txt;

File konfigurasi berhasil dimuat! Selamat berkicau!

>>

6.2 Test Daftar, Masuk, dan Keluar

6.2.1 Test Daftar

Hasil yang diharapkan beserta input:

```
>> DAFTAR;
```

Masukkan nama:
WILSON;

Wah, sayang sekali nama tersebut telah diambil.

Masukkan nama:
YUSDA;

Masukkan kata sandi:
WILSON;

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

6.2.2 Test Masuk

Hasil yang diharapkan beserta input:

```
>> MASUK;
```

Masukkan nama:
USER;

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

Masukkan nama:
WILSON;

Masukkan kata sandi:
YUSUF;

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

Masukkan kata sandi:
YUSDA;

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

```
>> MASUK;
```

Wah Anda sudah masuk. Keluar dulu yuk!

6.2.3 Test Keluar

Hasil yang diharapkan beserta input:

```
>> KELUAR;
```

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

```
>> KELUAR;
```

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

6.3 Test Profil Akun

6.3.1 Test GANTI_PROFIL

Hasil yang diharapkan beserta input:

```
>> DAFTAR;
```

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

```
>> MASUK;
```

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

```
>> GANTI_PROFIL;
```

| Nama: WILSON

| Bio Akun:

| No HP:

| Weton:

Masukkan Bio Akun:RAPSODI JKT48;

Masukkan No HP:033923483434312211231212;

Masukkan Weton:Pahig;

Weton anda tidak valid.

Masukkan Weton:Pahing;

Profil Anda sudah berhasil diperbarui!

6.3.2 Test LIHAT_PROFIL

Hasil yang diharapkan beserta input:

>> LIHAT_PROFIL WILSON;

| Nama: WILSON

| Bio Akun: RAPSODI JKT48

| No HP: 033923483434312211231212

| Weton: Pahing

Foto Profil:

>> ATUR_JENIS_AKUN;

Saat ini, akun Anda adalah akun Publik.

Ingin mengubah ke akun Privat? (YA/TIDAK) YA;

Akun anda sudah diubah menjadi akun Privat.

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> DAFTAR;

Masukkan nama:

SHANI;

Masukkan kata sandi:

INDIRA;

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

>> MASUK;

Masukkan nama:

SHANI;

Masukkan kata sandi:

INDIRA;

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

>> LIHAT_PROFIL WILSON;

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

6.3.3 Test UBAH_FOTO_PROFIL

Hasil yang diharapkan beserta input:

>> DAFTAR;

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

>> MASUK;

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

>> GANTI_PROFIL;

| Nama: WILSON
| Bio Akun:
| No HP:
| Weton:

Masukkan Bio Akun:RAPSODI JKT48;

Masukkan No HP:033923483434312211231212;

Masukkan Weton:Pahig;

Weton anda tidak valid.

Masukkan Weton:Pahing;

Profil Anda sudah berhasil diperbarui!

>> LIHAT_PROFIL WILSON;

| Nama: WILSON
| Bio Akun: RAPSODI JKT48
| No HP: 033923483434312211231212
| Weton: Pahing

Foto Profil:

>> UBAH_FOTO_PROFIL;

Foto profil Anda saat ini adalah

Masukkan foto profil yang baru

R * R * R * R * R *

```
R * G @ B * G @ R *
R * G @ G @ G @ R *
R * G @ B * G @ R *
R * R * R * R * R *;
```

Foto profil anda sudah berhasil diganti!

```
>> LIHAT_PROFIL WILSON;
```

```
| Nama: WILSON
| Bio Akun: RAPSODI JKT48
| No HP: 033923483434312211231212
| Weton: Pahing
```

Foto Profil:

```
*****  
*@*@*  
*@@@*  
*@*@*  
*****
```

6.3.4 Test ATUR_JENIS_AKUN

Hasil yang diharapkan beserta input:

```
>> ATUR_JENIS_AKUN;
```

```
Saat ini, akun Anda adalah akun Publik.  
Ingin mengubah ke akun Privat? (YA/TIDAK) YA;
```

Akun anda sudah diubah menjadi akun Privat.

```
>> KELUAR;
```

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

```
>> DAFTAR;
```

Masukkan nama:
SHANI;

Masukkan kata sandi:
INDIRA;

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

```
>> MASUK;

Masukkan nama:
SHANI;

Masukkan kata sandi:
INDIRA;

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

>> LIHAT_PROFIL WILSON;

Wah, akun WILSON diprivat nih. Ikuti dulu yuk untuk bisa melihat
profil WILSON
```

6.4 Test Pertemanan

6.4.1 Test DAFTAR TEMAN

Hasil yang diharapkan beserta input:

```
>> DAFTAR;

Masukkan nama:
WILSON;

Masukkan kata sandi:
YUSDA;

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

>> DAFTAR_TEMAN;

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

>> MASUK;

Masukkan nama:
WILSON;

Masukkan kata sandi:
YUSDA;

Anda telah berhasil masuk dengan nama pengguna WILSON. Mari
```

menjelajahi BurBir bersama Ande-Ande Lumut!

>> DAFTAR_TEMAN;

WILSON tidak memiliki teman

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> DAFTAR;

Masukkan nama:

SHANIA;

Masukkan kata sandi:

GRACIA;

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

>> MASUK;

Masukkan nama:

SHANIA;

Masukkan kata sandi:

GRACIA;

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

>> TAMBAH_TEMAN;

Masukkan nama pengguna:

ASEP;

Pengguna bernama ASEP tidak ditemukan

>> TAMBAH_TEMAN;

Masukkan nama pengguna:

WILSON;

Permintaan pertemanan kepada WILSON telah dikirim. Tunggu beberapa

saat hingga permintaan Anda disetujui.

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

>> DAFTAR_PERMINTAAN_PERTEMANAN;

Terdapat 1 permintaan pertemanan untuk Anda.

| SHANIA

| Jumlah teman: 0

>> SETUJUI_PERTEMANAN;

| SHANIA

| Jumlah teman: 0

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

Permintaan pertemanan dari SHANIA telah disetujui. Selamat! Anda telah berteman dengan SHANIA.

>> DAFTAR_TEMAN;

WILSON memiliki 1 teman

Daftar teman WILSON:

| SHANIA

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:
SHANIA;

Masukkan kata sandi:
GRACIA;

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

>> DAFTAR TEMAN;

SHANIA memiliki 1 teman
Daftar teman SHANIA:
| WILSON

>> HAPUS TEMAN;

Masukkan nama pengguna:
WILSON;

Apakah Anda yakin ingin menghapus WILSON dari daftar teman Anda?
(YA/TIDAK) YA;

WILSON berhasil dihapus dari daftar teman Anda.

>> DAFTAR TEMAN;

SHANIA tidak memiliki teman

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:
WILSON;

Masukkan kata sandi:
YUSDA;

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

>> DAFTAR TEMAN;

WILSON tidak memiliki teman

6.4.2 Test HAPUS TEMAN

Hasil yang diharapkan beserta input:

>> MASUK;

Masukkan nama:

SHANIA;

Masukkan kata sandi:

GRACIA;

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

>> DAFTAR TEMAN;

SHANIA memiliki 1 teman

Daftar teman SHANIA:

| WILSON

>> HAPUS TEMAN;

Masukkan nama pengguna:

WILSON;

Apakah Anda yakin ingin menghapus WILSON dari daftar teman Anda?
(YA/TIDAK) YA;

WILSON berhasil dihapus dari daftar teman Anda.

>> DAFTAR TEMAN;

SHANIA tidak memiliki teman

```
>> KELUAR;
```

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

```
>> MASUK;
```

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

```
>> DAFTAR_TEMAN;
```

WILSON tidak memiliki teman

6.5 Test Permintaan Pertemanan

6.5.1 Test TAMBAH_TEMAN

Hasil yang diharapkan beserta input:

```
>> DAFTAR;
```

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

```
>> DAFTAR_TEMAN;
```

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

```
>> MASUK;
```

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

>> DAFTAR_TEMAN;

WILSON tidak memiliki teman

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> DAFTAR;

Masukkan nama:

SHANIA;

Masukkan kata sandi:

GRACIA;

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

>> MASUK;

Masukkan nama:

SHANIA;

Masukkan kata sandi:

GRACIA;

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

>> TAMBAH_TEMAN;

Masukkan nama pengguna:

ASEP;

Pengguna bernama ASEP tidak ditemukan

>> TAMBAH_TEMAN;

Masukkan nama pengguna:
WILSON;

Permintaan pertemanan kepada WILSON telah dikirim. Tunggu beberapa saat hingga permintaan Anda disetujui.

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:
WILSON;

Masukkan kata sandi:
YUSDA;

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

>> DAFTAR_PERMINTAAN_PERTEMANAN;

Terdapat 1 permintaan pertemanan untuk Anda.

| SHANIA
| Jumlah teman: 0

>> SETUJUI_PERTEMANAN;

| SHANIA
| Jumlah teman: 0

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

Permintaan pertemanan dari SHANIA telah disetujui. Selamat! Anda telah berteman dengan SHANIA.

>> DAFTAR TEMAN;

WILSON memiliki 1 teman
Daftar teman WILSON:

| SHANIA

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:

SHANIA;

Masukkan kata sandi:

GRACIA;

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

>> DAFTAR TEMAN;

SHANIA memiliki 1 teman

Daftar teman SHANIA:

| WILSON

>> HAPUS TEMAN;

Masukkan nama pengguna:

WILSON;

Apakah Anda yakin ingin menghapus WILSON dari daftar teman Anda?
(YA/TIDAK) YA;

WILSON berhasil dihapus dari daftar teman Anda.

>> DAFTAR TEMAN;

SHANIA tidak memiliki teman

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

>> DAFTAR_TEMAN;

WILSON tidak memiliki teman

6.5.2 Test DAFTAR_PERMINTAAN_PERTEMANAN

Hasil yang diharapkan beserta input:

>> MASUK;

Masukkan nama:

SHANIA;

Masukkan kata sandi:

GRACIA;

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

>> TAMBAH_TEMAN;

Masukkan nama pengguna:

ASEP;

Pengguna bernama ASEP tidak ditemukan

>> TAMBAH_TEMAN;

Masukkan nama pengguna:

WILSON;

Permintaan pertemanan kepada WILSON telah dikirim. Tunggu beberapa saat hingga permintaan Anda disetujui.

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:

WILSON;

Masukkan kata sandi:

YUSDA;

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

>> DAFTAR_PERMINTAAN_PERTEMANAN;

Terdapat 1 permintaan pertemanan untuk Anda.

| SHANIA

| Jumlah teman: 0

>> SETUJUI_PERTEMANAN;

| SHANIA

| Jumlah teman: 0

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

Permintaan pertemanan dari SHANIA telah disetujui. Selamat! Anda telah berteman dengan SHANIA.

>> DAFTAR TEMAN;

WILSON memiliki 1 teman

Daftar teman WILSON:

| SHANIA

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:

SHANIA;

Masukkan kata sandi:
GRACIA;

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

>> DAFTAR_TEMAN;

SHANIA memiliki 1 teman
Daftar teman SHANIA:
| WILSON

6.5.3 Test SETUJUI_PERTEMANAN

Hasil yang diharapkan beserta input:

>> DAFTAR_PERMINTAAN_PERTEMANAN;

Terdapat 1 permintaan pertemanan untuk Anda.

| SHANIA
| Jumlah teman: 0

>> SETUJUI_PERTEMANAN;

| SHANIA
| Jumlah teman: 0

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

Permintaan pertemanan dari SHANIA telah disetujui. Selamat! Anda telah berteman dengan SHANIA.

6.6 Test Kicauan

6.6.1 Test Kicau

Hasil yang diharapkan beserta input:

>> KICAU;

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

>> DAFTAR;

Masukkan nama:
USER;

Masukkan kata sandi:
PASS;

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

>> KICAU;

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

>> MASUK;

Masukkan nama:
USER;

Masukkan kata sandi:
PASS;

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

>> KICAU;

Masukkan kicauan:
HALO;

Masukkan tagar:
BRO;

Selamat! kicauan telah diterbitkan!

Detil kicauan:

ID = 1
USER
24/11/2023 18:24:7
HALO
#BRO
Disukai: 0

>> KICAU;

Masukkan kicauan:
MAKAN;

```
Masukkan tagar:  
MINUM;  
  
Selamat! kicauan telah diterbitkan!  
Detil kicauan:  
| ID = 2  
| USER  
| 24/11/2023 18:24:16  
| MAKAN  
| #MINUM  
| Disukai: 0
```

6.6.2 Test KICAUAN

Hasil yang diharapkan beserta input:

```
>> KICAUAN;  
  
| ID = 1  
| USER  
| 24/11/2023 18:24:7  
| HALO  
| #BRO  
| Disukai: 0  
  
| ID = 2  
| USER  
| 24/11/2023 18:24:16  
| MAKAN  
| #MINUM  
| Disukai: 0
```

6.6.3 Test SUKA_KICAUAN

Hasil yang diharapkan beserta input:

```
>> SUKA_KICAUAN 1;  
  
Selamat! kicauan telah disukai!  
Detil kicauan:  
| ID = 1  
| USER  
| 24/11/2023 18:24:7  
| HALO  
| #BRO  
| Disukai: 1
```

```
>> SUKA_KICAUAN 2;

Selamat! kicauan telah disukai!
Detil kicauan:
| ID = 2
| USER
| 24/11/2023 18:24:16
| MAKAN
| #MINUM
| Disukai: 1

>> SUKA_KICAUAN 3;

Tidak ditemukan kicauan dengan ID = 3;

>> UBAH_KICAUAN 1;

Masukkan kicauan baru:
MASAK;

Selamat! kicauan telah diterbitkan!
Detil kicauan:
| ID = 1
| USER
| 24/11/2023 18:24:7
| MASAK
| #BRO
| Disukai: 1
```

6.6.4 Test UBAH_KICAUAN

Hasil yang diharapkan beserta input:

```
>> UBAH_KICAUAN 1;
```

Masukkan kicauan baru:

MASAK;

Selamat! kicauan telah diterbitkan!

Detil kicauan:

```
| ID = 1  
| USER  
| 24/11/2023 18:24:7  
| MASAK  
| #BRO  
| Disukai: 1
```

```
>> UBAH_KICAUAN 3;
```

Tidak ditemukan kicauan dengan ID = 3;

6.7 Test Balasan

6.7.1 Test BALAS

Hasil yang diharapkan beserta input:

6.7.2 Test BALASAN

Hasil yang diharapkan beserta input:

6.7.3 Test HAPUS_BALASAN

Hasil yang diharapkan beserta input:

6.8 Test Draf Kicauan

6.8.1 Test BUAT_DRAF

Hasil yang diharapkan beserta input:

```
>> DAFTAR;  
  
Masukkan nama:  
USER;  
  
Masukkan kata sandi:  
PASS;  
  
Pengguna telah berhasil terdaftar. Masuk untuk menikmati  
fitur-fitur BurBir.  
  
>> MASUK;  
  
Masukkan nama:  
USER;  
  
Masukkan kata sandi:  
PASS;  
  
Anda telah berhasil masuk dengan nama pengguna USER. Mari  
menjelajahi BurBir bersama Ande-Ande Lumut!  
  
>> BUAT_DRAF;  
  
Masukkan draf:  
SEVENTEEN;  
  
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?  
HAPUS;  
  
Draf telah berhasil dihapus!  
  
>> BUAT_DRAF;  
  
Masukkan draf:  
RAPSODI;  
  
Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?  
SIMPAN;  
  
>> LIHAT_DRAF;  
  
Ini draf terakhir Anda:
```

| 23/11/2023 18:27:42

| RAPSODI

Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?

(KEMBALI jika ingin kembali)

HAPUS;

Draf telah berhasil dihapus!

>> BUAT_DRAF;

Masukkan draf:

RAPSODI V2;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?

SIMPAN;

>> LIHAT_DRAF;

Ini draf terakhir Anda:

| 23/11/2023 18:28:14

| RAPSODI V2

Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?

(KEMBALI jika ingin kembali)

UBAH;

Masukkan draf:

RAPSODI V3;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?

SIMPAN;

>> LIHAT_DRAF;

Ini draf terakhir Anda:

| 23/11/2023 18:28:29

| RAPSODI V3

Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?

(KEMBALI jika ingin kembali)

TERBIT;

Selamat! Draf kicauan telah diterbitkan!

Detil kicauan:

| ID = 1

| USER

| 23/11/2023 18:28:38

| RAPSODI V3

| Disukai: 0

```
>> BUAT_DRAF;

Masukkan draf:
RAPSODI V4;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
TERBIT;

Selamat! Draf kicauan telah diterbitkan!
Detil kicauan:
| ID = 2
| USER
| 23/11/2023 18:28:55
| RAPSODI V4
| Disukai: 0

>> KICAUAN;

| ID = 1
| USER
| 23/11/2023 18:28:38
| RAPSODI V3
| Disukai: 0

| ID = 2
| USER
| 23/11/2023 18:28:55
| RAPSODI V4
| Disukai: 0
```

6.8.2 Test LIHAT_DRAF

Hasil yang diharapkan beserta input:

```
>> BUAT_DRAF;

Masukkan draf:
RAPSODI;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
SIMPAN;
```

```
>> LIHAT_DRAF;

Ini draf terakhir Anda:
| 23/11/2023 18:27:42
| RAPSODI
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?
(KEMBALI jika ingin kembali)
HAPUS;

Draf telah berhasil dihapus!

>> BUAT_DRAF;

Masukkan draf:
RAPSODI V2;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
SIMPAN;

>> LIHAT_DRAF;

Ini draf terakhir Anda:
| 23/11/2023 18:28:14
| RAPSODI V2
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?
(KEMBALI jika ingin kembali)
UBAH;

Masukkan draf:
RAPSODI V3;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
SIMPAN;

>> LIHAT_DRAF;

Ini draf terakhir Anda:
| 23/11/2023 18:28:29
| RAPSODI V3
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini?
(KEMBALI jika ingin kembali)
TERBIT;

Selamat! Draf kicauan telah diterbitkan!
Detil kicauan:
| ID = 1
```

```
| USER
| 23/11/2023 18:28:38
| RAPSODI V3
| Disukai: 0

>> BUAT_DRAF;

Masukkan draf:
RAPSODI V4;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
TERBIT;

Selamat! Draf kicauan telah diterbitkan!
Detil kicauan:
| ID = 2
| USER
| 23/11/2023 18:28:55
| RAPSODI V4
| Disukai: 0

>> KICAUAN;

| ID = 1
| USER
| 23/11/2023 18:28:38
| RAPSODI V3
| Disukai: 0

| ID = 2
| USER
| 23/11/2023 18:28:55
| RAPSODI V4
| Disukai: 0
```

6.9 Test Utas

6.9.1 Test UTAS

Hasil yang diharapkan beserta input:

```
>> DAFTAR;

Masukkan nama:
```

USER;

Masukkan kata sandi:

PASS;

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

>> MASUK;

Masukkan nama:

USER;

Masukkan kata sandi:

PASS;

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

>> KICAU;

Masukkan kicauan:

INI KICAU UTAMA 1;

Selamat! kicauan telah diterbitkan!

Detil kicauan:

| ID = 1
| USER
| 23/11/2023 17:53:46
| INI KICAU UTAMA 1
| Disukai: 0

>> KICAU;

Masukkan kicauan:

INI KICAU UTAMA 2;

Selamat! kicauan telah diterbitkan!

Detil kicauan:

| ID = 2
| USER
| 23/11/2023 17:53:52
| INI KICAU UTAMA 2
| Disukai: 0

>> UTAS 3;

```
Kicauan tidak ditemukan
```

```
>> UTAS 1;
```

Masukkan Kicauan:

```
INI UTAS INDEX 1 DARI KICAU 1;
```

Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):

```
YA;
```

Masukkan Kicauan:

```
INI UTAS INDEX 2 DARI KICAU 1;
```

Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):

```
TIDAK;
```

```
Utas Selesai
```

```
>> CETAK_UTAS 1;
```

```
| ID = 1
```

```
| USER
```

```
| 23/11/2023 17:53:46
```

```
| INI KICAU UTAMA 1
```

```
| INDEX:1
```

```
| USER
```

```
| 23/11/2023 17:54:13
```

```
| INI UTAS INDEX 1 DARI KICAU 1
```

```
| INDEX:2
```

```
| USER
```

```
| 23/11/2023 17:54:23
```

```
| INI UTAS INDEX 2 DARI KICAU 1
```

```
>> CETAK_UTAS 2;
```

Utas tidak ditemukan!

```
>> UTAS 2;
```

Masukkan Kicauan:

```
INI UTAS INDEX 1 DARI KICAU 2;
```

Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):

```
YA;
```

Masukkan Kicauan:

```
INI UTAS INDEX 2 DARI KICAU 2;
```

Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):

```
YA;
```

Masukkan Kicauan:

```
INI UTAS INDEX 3 DARI KICAU 2;
```

```
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):  
TIDAK;  
Utas Selesai  
  
>> CETAK_UTAS 2;  
  
| ID = 2  
| USER  
| 23/11/2023 17:53:52  
|INI KICAU UTAMA 2  
  
| INDEX:1  
| USER  
| 23/11/2023 17:54:46  
|INI UTAS INDEX 1 DARI KICAU 2  
  
| INDEX:2  
| USER  
| 23/11/2023 17:54:56  
|INI UTAS INDEX 2 DARI KICAU 2  
  
| INDEX:3  
| USER  
| 23/11/2023 17:55:7  
|INI UTAS INDEX 3 DARI KICAU 2
```

6.9.2 Test SAMBUNG_UTAS

Hasil yang diharapkan beserta input:

```
>> SAMBUNG_UTAS 3 1;  
  
Utas tidak ditemukan!  
>> SAMBUNG_UTAS 2 4;  
  
Masukkan kicauan:  
INI UTAS INDEX 4 DARI KICAU 2;  
  
>> CETAK_UTAS 2;  
  
| ID = 2  
| USER  
| 23/11/2023 17:53:52  
|INI KICAU UTAMA 2  
  
| INDEX:1  
| USER  
| 23/11/2023 17:54:46
```

```
|INI UTAS INDEX 1 DARI KICAU 2
```

```
|INDEX:2
```

```
|USER
```

```
|23/11/2023 17:54:56
```

```
|INI UTAS INDEX 2 DARI KICAU 2
```

```
|INDEX:3
```

```
|USER
```

```
|23/11/2023 17:55:7
```

```
|INI UTAS INDEX 3 DARI KICAU 2
```

```
|INDEX:4
```

```
|USER
```

```
|23/11/2023 17:55:47
```

```
|INI UTAS INDEX 4 DARI KICAU 2
```

```
>> SAMBUNG_UTAS 2 3;
```

Masukkan kicauan:

```
INI AKAN JADI UTAS INDEX 3 DARI KICAU 2 (BARU);
```

```
>> CETAK_UTAS 2;
```

```
|ID = 2
```

```
|USER
```

```
|23/11/2023 17:53:52
```

```
|INI KICAU UTAMA 2
```

```
|INDEX:1
```

```
|USER
```

```
|23/11/2023 17:54:46
```

```
|INI UTAS INDEX 1 DARI KICAU 2
```

```
|INDEX:2
```

```
|USER
```

```
|23/11/2023 17:54:56
```

```
|INI UTAS INDEX 2 DARI KICAU 2
```

```
|INDEX:3
```

```
|USER
```

```
|23/11/2023 17:56:26
```

```
|INI AKAN JADI UTAS INDEX 3 DARI KICAU 2 (BARU)
```

```
|INDEX:4
```

```
|USER
```

```
|23/11/2023 17:55:7
```

```
|INI UTAS INDEX 3 DARI KICAU 2
```

```
| INDEX:5  
| USER  
| 23/11/2023 17:55:47  
|INI UTAS INDEX 4 DARI KICAU 2
```

6.9.3 Test CETAK_UTAS

Hasil yang diharapkan beserta input:

```
>> UTAS 1;  
  
Masukkan Kicauan:  
INI UTAS INDEX 1 DARI KICAU 1;  
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):  
YA;  
  
Masukkan Kicauan:  
INI UTAS INDEX 2 DARI KICAU 1;  
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):  
TIDAK;  
Utas Selesai  
  
>> CETAK_UTAS 1;  
  
| ID = 1  
| USER  
| 23/11/2023 17:53:46  
|INI KICAU UTAMA 1  
  
| INDEX:1  
| USER  
| 23/11/2023 17:54:13  
|INI UTAS INDEX 1 DARI KICAU 1  
  
| INDEX:2  
| USER  
| 23/11/2023 17:54:23  
|INI UTAS INDEX 2 DARI KICAU 1  
  
>> CETAK_UTAS 2;  
Utas tidak ditemukan!  
>> UTAS 2;
```

6.9.4 Test HAPUS_UTAS

Hasil yang diharapkan beserta input:

```
>> HAPUS_UTAS 2 3;
```

Kicauan sambungan berhasil dihapus!

```
>> CETAK_UTAS 2;
```

```
| ID = 2  
| USER  
| 23/11/2023 17:53:52  
|INI KICAU UTAMA 2
```

```
| INDEX:1  
| USER  
| 23/11/2023 17:54:46  
|INI UTAS INDEX 1 DARI KICAU 2
```

```
| INDEX:2  
| USER  
| 23/11/2023 17:54:56  
|INI UTAS INDEX 2 DARI KICAU 2
```

```
| INDEX:3  
| USER  
| 23/11/2023 17:55:7  
|INI UTAS INDEX 3 DARI KICAU 2
```

```
| INDEX:4  
| USER  
| 23/11/2023 17:55:47  
|INI UTAS INDEX 4 DARI KICAU 2
```

```
>> HAPUS_UTAS 4 1;
```

Utas tidak ditemukan!

```
>> HAPUS_UTAS 2 5;
```

Kicauan sambungan dengan index 5 tidak ditemukan pada utas!

6.10 Test Simpan dan Muat

Fitur yang akan diujikan adalah Simpan dan Muat, yaitu fitur yang digunakan untuk menyimpan serta membuka kembali kondisi BurBir saat ini. Simpan dapat dilakukan baik sebelum maupun sesudah masuk ke suatu akun. Akan tetapi, Muat hanya dapat dilakukan ketika pengguna belum masuk ke suatu akun. Hasil yang diharapkan adalah program mampu menyimpan serta memuat berkas konfigurasi dengan baik. Berikut adalah hasil implementasinya:

6.11 Test Kelompok Teman

```
>> MASUK;
```

Masukkan nama:

```
a1;
```

Masukkan kata sandi:

```
p1;
```

Anda telah berhasil masuk dengan nama pengguna a1. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

```
>> TAMBAH TEMAN;
```

Masukkan nama pengguna:

```
a2;
```

Permintaan pertemanan kepada a2 telah dikirim. Tunggu beberapa saat hingga permintaan Anda disetujui.

>> KELAUR;

Perintah tidak valid!

>> KELUAR;

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;

Masukkan nama:

a2;

Masukkan kata sandi:

p2;

Anda telah berhasil masuk dengan nama pengguna a2. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> SETUJUI_PERTEMANAN;

| a1

| Jumlah teman: 0

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

Permintaan pertemanan dari a1 telah disetujui. Selamat! Anda telah berteman dengan a1.

```
>> TAMBAH_TEMAN;
```

Masukkan nama pengguna:

```
a3;
```

Permintaan pertemanan kepada a3 telah dikirim. Tunggu beberapa saat hingga permintaan Anda disetujui.

```
>> KELUAR;
```

Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

```
>> MASUK;
```

Masukkan nama:

```
a3;
```

Masukkan kata sandi:

```
p3;
```

Anda telah berhasil masuk dengan nama pengguna a3. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

```
>> SETUJUI_PERTEMANAN;
```

```
| a2
```

```
| Jumlah teman: 1
```

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

Permintaan pertemanan dari a2 telah disetujui. Selamat! Anda telah berteman dengan a2.

```
>> KELOMPOK TEMAN;
```

Terdapat 2 orang dalam Kelompok Teman a3:

a1

a2

6.12 Test Tagar

```
>> KICAU;
```

Masukkan kicauan:
testing;

Masukkan tagar:
testing123;

Selamat! kicauan telah diterbitkan!
Detil kicauan:

```
| ID = 1
| a1
| 24/11/2023 21:1:39
| testing
| #testing123
| Disukai: 0
```

```
>> CARI_KICAUAN;
```

```
Tidak ditemukan kicauan dengan tagar #!
```

```
>> CARI_KICAUAN testing123;
```

```
| ID = 1  
| a1  
| 24/11/2023 21:1:39  
| testing  
| #testing123  
| Disukai: 0
```

```
>>
```

6.10 Test FYB

```
>> FYB;
```

```
Berikut 2 kicauan dengan like tertinggi di FYB
```

```
Kicauan 1
```

```
| A  
| 24/11/2023 20:49:25  
| SAS  
| #SA  
| Disukai: 7
```

```
Kicauan 2
```

```
| A  
| 24/11/2023 21:0:26  
| ASEP  
| #ASEP  
| Disukai: 0
```

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	Baca Config	Menyiapkan file config yang akan dibaca oleh program.	Pengujian dilakukan dengan memasukkan nama file config beserta extensionnya.	Tertulis pada bagian	Dijelaskan di dalam data tes.	
2	DAFTAR	Memastikan dapat melakukan daftar untuk menginisialisasi adanya user.	1. Masukkan perintah DAFTAR 2. Masukkan Nama (Nama harus unik, jika tidak akan diminta input ulang) 3. Masukkan password 4. Anda sudah terdaftar	Tertulis pada bagian 6.2.1	Tertulis pada bagian 6.2.1	Pada list gambar 7.1
3.	MASUK	Memastikan pengguna dapat masuk menggunakan akun yang telah didaftarkan sebelumnya.	1. Pastikan sudah terdaftar 2. Masukkan perintah MASUK 3. Akan diminta input username dan password sampai sesuai	Tertulis pada bagian 6.2.2	Tertulis pada bagian 6.2.2	Pada list gambar 7.1

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
			4. Anda sudah masuk ke akun anda.			
4.	KELUAR	Memastikan pengguna dapat keluar dengan memastikan status user.	1. Pastikan anda sedang terlogin, jika tidak tidak akan bisa keluar 2. Masukkan perintah KELUAR 3. Anda sudah keluar dari akun anda	Tertulis pada bagian 6.2.3	Tertulis pada bagian 6.2.3	Pada list gambar 7.1
5	GANTI_PROFIL	Memastikan bahwa ketika pengguna melakukan save pada profil juga terganti pada dataset, dengan memastikan weton tepat.	1. Pastikan anda terlogin 2. Masukkan perintah GANTI_PROFIL 3. Masukkan identitas, dan cocokkan validasi Weton 4. Profil akan berubah	Tertulis pada bagian 6.3.1	Tertulis pada bagian 6.3.1	Pada list gambar 7.2
6.	LIHAT_PROFIL	Memastikan user dapat melihat	1. Pastikan sudah login	Tertulis pada bagian 6.3.2	Tertulis pada bagian 6.3.2	Pada list gambar 7.2

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		profil siapapun dengan mempertimbangkan keberadaan user serta jenis akun.	<p>2. Masukkan perintah LIHAT_PROFIL beserta nama user yang ingin dilihat</p> <p>3. Tes apakah user tersebut private atau tidak, jika private tidak bisa melihat</p> <p>4. Jika publik, maka profil akan muncul.</p>			
7.	UBAH_FOTO_PROFIL	Memastikan user dapat mengubah foto profil dan terupdate di dataset.	<p>1. Pastikan sudah login.</p> <p>2. Masukkan perintah UBAH_FOTO_PROFIL</p> <p>3. Masukkan input gambar baru</p> <p>4. Foto profil akan berubah</p>	Tertulis pada bagian 6.3.3	Tertulis pada bagian 6.3.3	Pada list gambar 7.2
8.	ATUR_JENIS_AKUN	Memastikan user dapat merubah	1. Pastikan sudah login terlebih dahulu	Tertulis pada bagian 6.3.4	Tertulis pada bagian 6.3.4	Pada list gambar 7.2

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		jenis akun dengan kebalikan jenis akunnya sekarang.	2. Masukkan perintah ATUR_JENIS_AKUN 3. Input persetujuan 4. Akun akan berubah ke lawan jenis akun.			
9.	DAFTAR_TEMAN	Memastikan user dapat melihat siapa saja temannya.	1. Pastikan sudah login 2. Masukkan perintah DAFTAR_TE MAN 3. Status pertemanan akan keluar	Tertulis pada bagian 6.4.1	Tertulis pada bagian 6.4.1	Pada list gambar 7.3
10.	HAPUS_TE MAN	Memastikan bahwa ketika menghapus teman teman tetap terhapus secara 2 arah, namun juga memastikan	1. Pastikan sudah login 2. Masukkan perintah HAPUS_TEMAN 3. Masukkan nama yang ingin dihapus 4. Pengecekan nama teman	Tertulis pada bagian 6.4.2	Tertulis pada bagian 6.4.2	Pada list gambar 7.3

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		validasi kebenaran nama inputan yang ingin dihapus.	dan jika valid akan terhapus.			
11.	TAMBAH_TEMAN	Memastikan bahwa ketika menambah teman teman tetap bertambah secara 2 arah, namun juga memastikan validasi kebenaran nama inputan yang ingin ditambah.	1. Pastikan sudah login 2. Masukkan perintah TAMBAH_TEMAN 3. Masukkan nama teman 4. Akan ada validasi dan jika sesuai akan terkirim	Tertulis pada bagian 6.5.1	Tertulis pada bagian 6.5.1	Pada list gambar 7.3
12.	DAFTAR_PERTEMUAN_ANAN	Memastikan bahwa setiap kiriman pertemanan benar terkirim.	1. Pastikan sudah login 2. Masukkan perintah DAFTAR_PERTEMUAN_ANAN	Tertulis pada bagian 6.5.2	Tertulis pada bagian 6.5.2	Pada list gambar 7.3

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
			PERTEMANAN 3. Akan muncul daftar permintaan pertemanan			
13.	SETUJUI_PERTEMANAN	Memastikan bahwa setiap pemintaan pertemanan bisa diterima ataupun ditolak secara 2 arah	1. Pastikan sudah login 2. Masukkan perintah SETUJUI_PERTEMANAN 3. Akan ada input apakah ingin setuju. 4. Jika setuju teman bertambah dan jika tidak permintaan pertemanan akan dihapus	Tertulis pada bagian 6.5.3	Tertulis pada bagian 6.5.3	Pada list gambar 7.3
14.	KICAU	Memastikan user dapat membuat kicauan dengan tagar.	1. Pastikan sudah login 2. Masukkan perintah kicau 3. Masukkan Kicauan 4. Masukkan tagar jika ada	Tertulis pada bagian 6.6.1	Tertulis pada bagian 6.6.1	Pada list gambar 7.4

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
			5. Kicau akan tersimpan			
15.	KICAUAN	Memastikan user dapat melihat semua kicauan yang dibuat	1. Pastikan sudah login 2. Masukkan perintah kicauan 3. Semua kicauan akan terlihat	Tertulis pada bagian 6.6.2	Tertulis pada bagian 6.6.2	Pada list gambar 7.4
16.	SUKA_KICAUAN	Memastikan user dapat melakukan <i>like</i> pada kicauan sehingga jumlah like terupdate pada dataset	1. Pastikan sudah login 2. Masukkan perintah suka_kicauan 3. Masukkan idKicau yang disukai 4. Akan ada validasi kebenaran IDKicau, jika benar maka akan terdaftar sebagai suka.	Tertulis pada bagian 6.6.3	Tertulis pada bagian 6.6.3	Pada list gambar 7.4
17.	UBAH_KICAUAN	Memastikan user dapat mengubah kicauan sesuai	1. Pastikan sudah login 2. Masukkan perintah	Tertulis pada bagian 6.6.4	Tertulis pada bagian 6.6.4	Pada list gambar 7.4

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		dengan input ID yang divalidasi , dan jika benar akan mengubah kicau pada database	UBAH_KICAUAN 3. Masukkan input IDKicau 4. Akan ada validasi input dan jika benar akan ada input kicauan dan akan diubah.			
18.	BALAS			Tertulis pada bagian 6.7.1	Tertulis pada bagian 6.7.1	
19.	BALASAN			Tertulis pada bagian 6.7.2	Tertulis pada bagian 6.7.1	
20.	HAPUS_BALASAN			Tertulis pada bagian 6.7.3	Tertulis pada bagian 6.7.3	
21.	BUAT_DRAFT	Memastikan user dapat membuat Draf dengan berbagai pilihan seperti hapus, simpan ataupun terbit,	1. Pastikan sudah login 2. Masukkan perintah BUAT_DRAF 3. Akan diminta masukkan kicauan	Tertulis pada bagian 6.8.1	Tertulis pada bagian 6.8.1	Pada list gambar 7.5

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		dengan jika disimpan akan dimasukkan ke database, hapus dibatalkan, dan terbit berarti dijadikan kicauan	4. Input pilihan terbit,simpan, atau hapus 5. Program berjalan sesuai input			
22.	LIHAT_DR AF	Memastikan user dapat melihat draft terbaru yang dibuatnya secara tepat.	1. Pastikan sudah login 2. Masukkan perintah LIHAT_DRAF 3. Akan keluar output draft	Tertulis pada bagian 6.8.2	Tertulis pada bagian 6.8.2	Pada list gambar 7.5
23	UTAS	Memastikan user dapat membuat Utas secara benar dengan validasi input ID Kicau dan milik dirinya.	1. Pastikan sudah login 2. Masukan perintah UTAS beserta IDKicau 3. Akan ada validasi IDKicau dan jika benar utas dibuat	Tertulis pada bagian 6.9.1	Tertulis pada bagian 6.9.1	Pada list gambar 7.6

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
			4. Terdapat pilihan menyambung utas			
24.	SAMBUNG_UTAS	Memastikan user dapat menyambung Utas secara benar dengan validasi input ID Utas dan index inputan dirinya.	1. Pastikan sudah login 2. Masukan perintah SAMBUNG_UTAS beserta IDUtas dan index 3. Akan ada validasi IDUtas dan Index dan jika benar utas disambung	Tertulis pada bagian 6.9.2	Tertulis pada bagian 6.9.2	Pada list gambar 7.6
25.	CETAK_UTAS	Memastikan user dapat mencetak sebuah utas sesuai dengan validasi, yaitu tidak boleh privat.	1. Pastikan sudah login 2. Masukan perintah SAMBUNG_UTAS beserta IDUtas 3. Akan ada validasi IDUtas dan jika benar utas ditampilkan	Tertulis pada bagian 6.9.3	Tertulis pada bagian 6.9.3	Pada list gambar 7.6

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
26.	HAPUS_UTAS	Memastikan user dapat mencetak sebuah utas dengan validasi milik dirinya, dan terupdate pada database.	1. Pastikan sudah login 2. Masukan perintah HAPUS_UTAS beserta IDUtas dan index 3. Akan ada validasi IDUtas dan index dan jika benar utas tersebut dihapus	Tertulis pada bagian 6.9.4	Tertulis pada bagian 6.9.4	Pada list gambar 7.6
27	TAGAR	Memastikan user dapat menambah tagar pada kicauan yg dilakukan	1.Pastikan Sudah login 2. Buat Kicauan 3. Masukkan Tagar 4. Lakukan fungsi KICAUAN untuk melihat tagar	Tertulis pada bagian 6.12	Tertulis pada bagian 6.12	Pada list gambar 7.9
28.	KELOMPOK TEMAN	Memastikan apakah fitur kelompok teman dapat	1.Buat 3 user dummy 2.Temangkan user 1 dengan	Tertulis pada bagian 6.11	Tertulis pada bagian 6.11	Pada list gamabr 7.7

No.	Fitur yang Dtes	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
		bekerja dengan baik	user 2 dan user 2 dengan user 3 3.Tampilkan output KELOMPOK_PERTEMANA N			
29.	FYB	Memastikan user dapat melihat kicauan dengan like tertinggi.	1. Pastikan sudah login 2. Masukkan Perintah FYB 3. Cocokkan dengan hasil yang diharapkan	Pada bagian 6.10	Pada bagian 6.10	Pada list gambar 7.8

Screenshot pengujian program:

Test Daftar,Masuk dan Keluar

List Gambar 7.1 (Daftar , Masuk, dan Keluar)

```
>> DAFTAR;
Masukkan nama:
WILSON;
Masukkan kata sandi:
YUSDA;
Pengguna telah berhasil terdaftar. Masuk untuk menikmati fitur-fitur BurBir.
>> DAFTAR;
Masukkan nama:
WILSON;
Wah, sayang sekali nama tersebut telah diambil.
Masukkan nama:
YUSDA;
Masukkan kata sandi:
WILSON;
Pengguna telah berhasil terdaftar. Masuk untuk menikmati fitur-fitur BurBir.
>> MASUK;
Masukkan nama:
USER;
Wah, nama yang Anda cari tidak ada. Masukkan nama lain!
Masukkan nama:
WILSON;
Masukkan kata sandi:
YUSUF;
Wah, kata sandi yang Anda masukkan belum tepat. Periksa kembali kata sandi Anda!
```

```
Wah, kata sandi yang Anda masukkan belum tepat. Periksa kembali kata sandi Anda!
Masukkan kata sandi:
YUSDA;

Anda telah berhasil masuk dengan nama pengguna WILSON. Mari menjelajahi BurBir bersama Ande-Ande Lumut!
>> MASUK;
Wah Anda sudah masuk. Keluar dulu yuk!

>> KELUAR;
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!
>> KELUAR;
Anda belum login! Masuk terlebih dahulu untuk menikmati layanan BurBir.
```

Test Profil Akun

List Gambar 7.2 Profil Akun

```
>> DAFTAR;
Masukkan nama:
WILSON;
Masukkan kata sandi:
YUSDA;
Pengguna telah berhasil terdaftar. Masuk untuk menikmati fitur-fitur BurBir.
>> MASUK;
Masukkan nama:
WILSON;
Masukkan kata sandi:
YUSDA;
Anda telah berhasil masuk dengan nama pengguna WILSON. Mari menjelajahi BurBir bersama Ande-Ande Lumut!
>> GANTI_PROFIL;
| Nama: WILSON
| Bio Akun:
| No HP:
| Weton:
Masukkan Bio Akun:RAPSODI JKT48;
Masukkan No HP:033923483434312211231212;
Masukkan Weton:Pahig;
Weton anda tidak valid.
Masukkan Weton:Pahing;
Profil Anda sudah berhasil diperbarui!
```

```
>> LIHAT_PROFIL WILSON;
| Nama: WILSON
| Bio Akun: RAPSODI JKT48
| No HP: 033923483434312211231212
| Weton: Pahing

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



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

Masukkan foto profil yang baru
R * R * R * R * R *
R * G @ B * G @ R *
R * G @ G @ G @ R *
R * G @ B * G @ R *
R * R * R * R * R *;

Foto profil anda sudah berhasil diganti!

```
>> LIHAT_PROFIL WILSON;
| Nama: WILSON
| Bio Akun: RAPSODI JKT48
| No HP: 033923483434312211231212
| Weton: Pahing

Foto Profil:
*****
*@@@*
*@@@*
*@@@*
*****
```



```
>> ATUR_JENIS_AKUN;
Saat ini, akun Anda adalah akun Publik.
Ingin mengubah ke akun Privat? (YA/TIDAK) YA;
```

Akun anda sudah diubah menjadi akun Privat.


```
>> KELUAR;
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!
```

```
>> MASUK;
Masukkan nama:
SHANI;

Masukkan kata sandi:
INDIRA;

Anda telah berhasil masuk dengan nama pengguna SHANI. Mari menjelajahi BurBir bersama Andre-Andre Lumut!
>> LIHAT_PROFIL WILSON;
Wah, akun WILSON diprivat nih. Ikuti dulu yuk untuk bisa melihat profil WILSON
```

Test Pertemanan + Tes Permintaan Pertemanan

List Gambar 7.3 Permintaan Pertemanan

```
>> DAFTAR;
Masukkan nama:
WILSON;
Masukkan kata sandi:
YUSDA;
Pengguna telah berhasil terdaftar. Masuk untuk menikmati fitur-fitur BurBir.

>> DAFTAR TEMAN;
Anda belum masuk! Masuk terlebih dahulu untuk menikmati layanan BurBir.

>> MASUK;
Masukkan nama:
WILSON;
Masukkan kata sandi:
YUSDA;
Anda telah berhasil masuk dengan nama pengguna WILSON. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> DAFTAR TEMAN;
WILSON tidak memiliki teman

>> KELUAR;
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!
```

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

>> MASUK;
Masukkan nama:
SHANIA;
Masukkan kata sandi:
GRACIA;
Anda telah berhasil masuk dengan nama pengguna SHANIA. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> TAMBAH TEMAN;
Masukkan nama pengguna:
ASEP;

Pengguna bernama ASEP tidak ditemukan

>> TAMBAH TEMAN;
Masukkan nama pengguna:
WILSON;

Permintaan pertemanan kepada WILSON telah dikirim. Tunggu beberapa saat hingga permintaan Anda disetujui.

>> KELUAR;
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!
```

```
Masukkan kata sandi:  
YUSDA;  
  
Anda telah berhasil masuk dengan nama pengguna WILSON. Mari menjelajahi BurBur bersama Ande-Ande Lumut!  
>> DAFTAR_PERMINTAAN_PERTEMANAN;  
Terdapat 1 permintaan pertemanan untuk Anda.  
| SHANIA  
| Jumlah teman: 0  
  
>> SETUJUI_PERTEMANAN;  
| SHANIA  
| Jumlah teman: 0  
  
Apakah Anda ingin menyetujui permintaan pertemanan ini? (YA/TIDAK) YA;  
Permintaan pertemanan dari SHANIA telah disetujui. Selamat! Anda telah berteman dengan SHANIA.  
  
>> DAFTAR TEMAN;  
WILSON memiliki 1 teman  
Daftar teman WILSON:  
| SHANIA  
  
>> KELUAR;  
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!
```

```
>> MASUK;  
Masukkan nama:  
SHANIA;  
Masukkan kata sandi:  
GRACIA;  
  
Anda telah berhasil masuk dengan nama pengguna SHANIA. Mari menjelajahi BurBur bersama Ande-Ande Lumut!  
>> DAFTAR TEMAN;  
SHANIA memiliki 1 teman  
Daftar teman SHANIA:  
| WILSON  
  
>> HAPUS TEMAN;  
Masukkan nama pengguna:  
WILSON;  
  
Apakah Anda yakin ingin menghapus WILSON dari daftar teman Anda? (YA/TIDAK)YA;  
WILSON berhasil dihapus dari daftar teman Anda.  
  
>> DAFTAR TEMAN;  
SHANIA tidak memiliki teman  
  
>> KELUAR;  
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!
```

```
>> MASUK;  
Masukkan nama:  
WILSON;  
Masukkan kata sandi:  
YUSDA;  
  
Anda telah berhasil masuk dengan nama pengguna WILSON. Mari menjelajahi BurBur bersama Ande-Ande Lumut!  
>> DAFTAR TEMAN;  
WILSON tidak memiliki teman
```

Test Kicau

List Gambar 7.4 Kicau

```
>> KICAU;
Anda belum masuk! Masuk terlebih dahulu untuk menikmati layanan BurBir.

>> DAFTAR;
Masukkan nama:
USER;

Masukkan kata sandi:
PASS;

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

>> KICAU;
Anda belum masuk! Masuk terlebih dahulu untuk menikmati layanan BurBir.

>> MASUK;
Masukkan nama:
USER;

Masukkan kata sandi:
PASS;

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

>> KICAU;
Masukkan kicauan:
HALO;

Masukkan tagar:
BRO;
```

Selamat! kicauan telah diterbitkan!

Detil kicauan:
| ID = 1
| USER
| 24/11/2023 18:24:7
| HALO
| #BRO
| Disukai: 0

>> KICAU;

Masukkan kicauan:
MAKAN;

Masukkan tagar:
MINUM;

Selamat! kicauan telah diterbitkan!

Detil kicauan:
| ID = 2
| USER
| 24/11/2023 18:24:16
| MAKAN
| #MINUM
| Disukai: 0

>> KICAUAN;

| ID = 1
| USER
| 24/11/2023 18:24:7
| HALO
| #BRO
| Disukai: 0

```
| ID = 2
| USER
| 24/11/2023 18:24:16
| MAKAN
| #MINUM
| Disukai: 0

>> SUKA_KICAUAN 1;

Selamat! kicauan telah disukai!
Detil kicauan:
| ID = 1
| USER
| 24/11/2023 18:24:7
| HALO
| #BRO
| Disukai: 1

>> SUKA_KICAUAN 2;

Selamat! kicauan telah disukai!
Detil kicauan:
| ID = 2
| USER
| 24/11/2023 18:24:16
| MAKAN
| #MINUM
| Disukai: 1

>> SUKA_KICAUAN 3;

Tidak ditemukan kicauan dengan ID = 3;
```

```
>> UBAH_KICAUAN 1;

Masukkan kicauan baru:
MASAK;

Selamat! kicauan telah diterbitkan!
Detil kicauan:
| ID = 1
| USER
| 24/11/2023 18:24:7
| MASAK
| #BRO
| Disukai: 1

>> UBAH_KICAUAN 3;

Tidak ditemukan kicauan dengan ID = 3;
```

Test Draft kicauan

List Gambar 7.5 Draft Kicau

```
>> LIHAT_DRAF;
Ini draf terakhir Anda:
| 23/11/2023 18:27:42
| RAPSODI
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? (KEMBALI jika ingin kembali)
HAPUS;
Draf telah berhasil dihapus!

>> BUAT_DRAF;
Masukkan draf:
RAPSODI V2;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
SIMPAN;

>> LIHAT_DRAF;
Ini draf terakhir Anda:
| 23/11/2023 18:28:14
| RAPSODI V2
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? (KEMBALI jika ingin kembali)
UBAH;
Masukkan draf:
RAPSODI V3;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
SIMPAN;
```

```
>> LIHAT_DRAF;
Ini draf terakhir Anda:
| 23/11/2023 18:28:29
| RAPSODI V3
Apakah anda ingin mengubah, menghapus, atau menerbitkan draf ini? (KEMBALI jika ingin kembali)
TERBIT;
Selamat! Draf kicauan telah diterbitkan!
Detail kicauan:
| ID = 1
| USER
| 23/11/2023 18:28:38
| RAPSODI V3
| Disukai: 0

>> BUAT_DRAF;
Masukkan draf:
RAPSODI V4;

Apakah anda ingin menghapus, menyimpan, atau menerbitkan draf ini?
TERBIT;
Selamat! Draf kicauan telah diterbitkan!
Detail kicauan:
| ID = 2
| USER
| 23/11/2023 18:28:55
| RAPSODI V4
| Disukai: 0
```

```
>> KICAUAN;
| ID = 1
| USER
| 23/11/2023 18:28:38
| RAPSODI V3
| Disukai: 0

| ID = 2
| USER
| 23/11/2023 18:28:55
| RAPSODI V4
| Disukai: 0
```

Test Utas

List Gambar 7.6 Utas

```
>> DAFTAR;
Masukkan nama:
USER;

Masukkan kata sandi:
PASS;

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

>> MASUK;
Masukkan nama:
USER;

Masukkan kata sandi:
PASS;

Anda telah berhasil masuk dengan nama pengguna USER. Mari menjelajahi BurBur bersama Ande-Ande Lumut!

>> KICAU;

Masukkan kicauan:
INI KICAU UTAMA 1;

Selamat! kicauan telah diterbitkan!
Detil kicauan:
| ID = 1
| USER
| 23/11/2023 17:53:46
| INI KICAU UTAMA 1
| Disukai: 0

>> KICAU;
```

```
Masukkan kicauan:
INI KICAU UTAMA 2;

Selamat! kicauan telah diterbitkan!
Detil Kicauan:
| ID = 2
| USER
| 23/11/2023 17:53:52
| INI KICAU UTAMA 2
| Disukai: 0

>> UTAS 3;

Kicauan tidak ditemukan

>> UTAS 1;

Masukkan Kicauan:
INI UTAS INDEX 1 DARI KICAU 1;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):
YA;

Masukkan Kicauan:
INI UTAS INDEX 2 DARI KICAU 1;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):
TIDAK;
Utas Selesai
```

```
>> CETAK_UTAS 1;

| ID = 1
| USER
| 23/11/2023 17:53:46
| INI KICAU UTAMA 1
    | INDEX:1
    | USER
    | 23/11/2023 17:54:13
    | INI UTAS INDEX 1 DARI KICAU 1

    | INDEX:2
    | USER
    | 23/11/2023 17:54:23
    | INI UTAS INDEX 2 DARI KICAU 1

>> CETAK_UTAS 2;

Utas tidak ditemukan!
>> UTAS 2;

Masukkan Kicauan:
INI UTAS INDEX 1 DARI KICAU 2;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):
YA;

Masukkan Kicauan:
INI UTAS INDEX 2 DARI KICAU 2;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):
YA;
Masukkan Kicauan:
INI UTAS INDEX 3 DARI KICAU 2;
Apakah Anda ingin melanjutkan utas ini? (YA/TIDAK):
TIDAK;
Utas Selesai
```

```
>> CETAK_UTAS 2;  
|ID = 2  
|USER  
|23/11/2023 17:53:52  
|INI KICAU UTAMA 2  
  
|INDEX:1  
|USER  
|23/11/2023 17:54:46  
|INI UTAS INDEX 1 DARI KICAU 2  
  
|INDEX:2  
|USER  
|23/11/2023 17:54:56  
|INI UTAS INDEX 2 DARI KICAU 2  
  
|INDEX:3  
|USER  
|23/11/2023 17:55:7  
|INI UTAS INDEX 3 DARI KICAU 2  
  
>> SAMBUNG_UTAS 3 1;  
Utas tidak ditemukan!  
>> SAMBUNG_UTAS 2 4;  
  
Masukkan kicauan:  
INI UTAS INDEX 4 DARI KICAU 2;
```

```
>> CETAK_UTAS 2;  
|ID = 2  
|USER  
|23/11/2023 17:53:52  
|INI KICAU UTAMA 2  
  
|INDEX:1  
|USER  
|23/11/2023 17:54:46  
|INI UTAS INDEX 1 DARI KICAU 2  
  
|INDEX:2  
|USER  
|23/11/2023 17:54:56  
|INI UTAS INDEX 2 DARI KICAU 2  
  
|INDEX:3  
|USER  
|23/11/2023 17:55:7  
|INI UTAS INDEX 3 DARI KICAU 2  
  
|INDEX:4  
|USER  
|23/11/2023 17:55:47  
|INI UTAS INDEX 4 DARI KICAU 2  
  
>> SAMBUNG_UTAS 2 3;  
  
Masukkan kicauan:  
INI AKAN JADI UTAS INDEX 3 DARI KICAU 2 (BARU);
```

```
>> CETAK_UTAS 2;  
|ID = 2  
|USER  
|23/11/2023 17:53:52  
|INI KICAU UTAMA 2  
  
|INDEX:1  
|USER  
|23/11/2023 17:54:46  
|INI UTAS INDEX 1 DARI KICAU 2  
  
|INDEX:2  
|USER  
|23/11/2023 17:54:56  
|INI UTAS INDEX 2 DARI KICAU 2  
  
|INDEX:3  
|USER  
|23/11/2023 17:56:26  
|INI AKAN JADI UTAS INDEX 3 DARI KICAU 2 (BARU)  
  
|INDEX:4  
|USER  
|23/11/2023 17:55:7  
|INI UTAS INDEX 3 DARI KICAU 2  
  
|INDEX:5  
|USER  
|23/11/2023 17:55:47  
|INI UTAS INDEX 4 DARI KICAU 2  
  
>> HAPUS_UTAS 2 3;  
Kicauan sambungan berhasil dihapus!
```

```

>> CETAK_UTAS 2;
|ID = 2
|USER
|23/11/2023 17:53:52
|INI KICAU UTAMA 2
|INDEX:1
|USER
|23/11/2023 17:54:46
|INI UTAS INDEX 1 DARI KICAU 2
|INDEX:2
|USER
|23/11/2023 17:54:56
|INI UTAS INDEX 2 DARI KICAU 2
|INDEX:3
|USER
|23/11/2023 17:55:7
|INI UTAS INDEX 3 DARI KICAU 2
|INDEX:4
|USER
|23/11/2023 17:55:47
|INI UTAS INDEX 4 DARI KICAU 2

>> HAPUS_UTAS 4 1;
Utas tidak ditemukan!
>> HAPUS_UTAS 2 5;
Kicauan sambungan dengan index 5 tidak ditemukan pada utas!

```

Test Kelompok Teman:

List Gambar 7.7 Kelompok Teman

```

>> TAMBAH TEMAN;
Masukkan nama pengguna:
a3;

Permintaan pertemanan kepada a3 telah dikirim. Tunggu beberapa saat hingga permintaan Anda disetujui.

>> KELUAR;
Anda berhasil logout. Sampai jumpa di pertemuan berikutnya!

>> MASUK;
Masukkan nama:
a3;
Masukkan kata sandi:
p3;
Anda telah berhasil masuk dengan nama pengguna a3. Mari menjelajahi BurBir bersama Ande-Ande Lumut!

>> SETUJUI_PERTEMANAN;
| a2
| Jumlah teman: 1
Apakah Anda ingin menyetujui permintaan pertemanan ini? (YA/TIDAK) YA;
Permintaan pertemanan dari a2 telah disetujui. Selamat! Anda telah berteman dengan a2.

>> KELOMPOK TEMAN;
Terdapat 2 orang dalam Kelompok Teman a3:
a1
a2

```

List Gambar 7.8 FYB

```

>> FYB;

Berikut 2 kicauan dengan like tertinggi di FYB

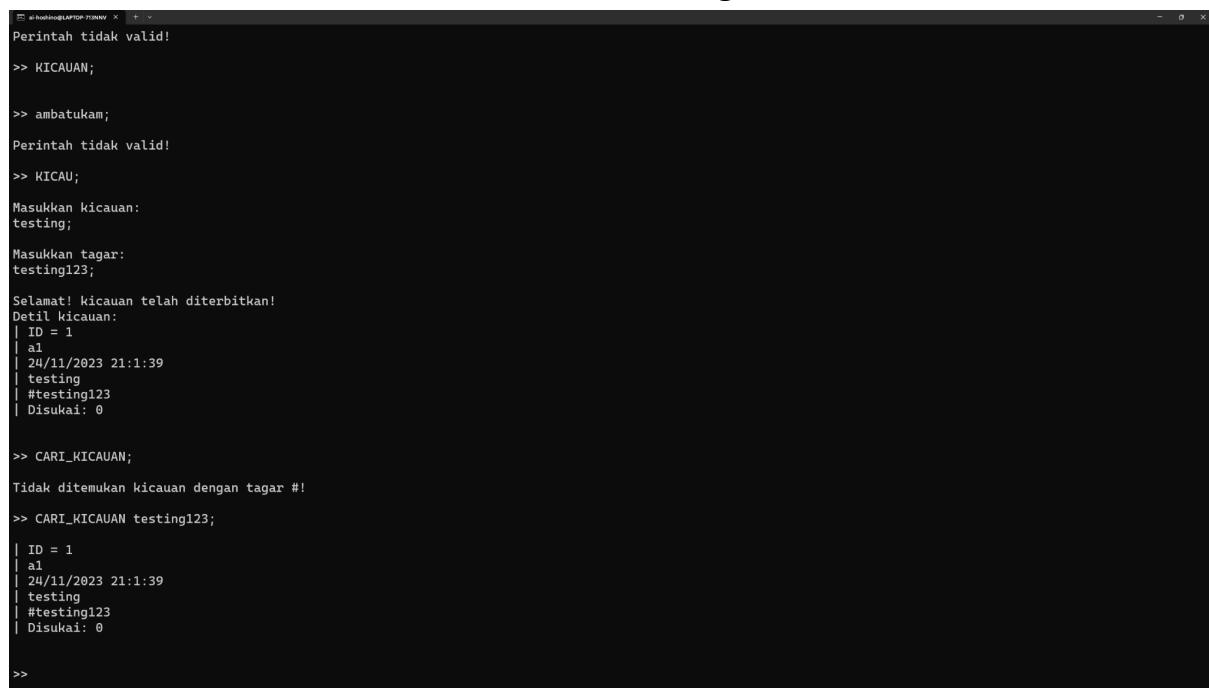
Kicauan 1
| A
| 24/11/2023 20:49:25
| SAS
| #SA
| Disukai: 7

Kicauan 2
| A
| 24/11/2023 21:0:26
| ASEP
| #ASEP
| Disukai: 0

```

Test Tagar

List Gambar 7.9 Tagar



```

$ aksion@APTOP:~ %
Perintah tidak valid!

>> KICAUAN;

>> ambatukam;
Perintah tidak valid!

>> KICAU;
Masukkan kicauan:
testing;

Masukkan tagar:
testing123;

Selamat! kicauan telah diterbitkan!
Detil kicauan:
| ID = 1
| a1
| 24/11/2023 21:1:39
| testing
| #testing123
| Disukai: 0

>> CARI_KICAUAN;
Tidak ditemukan kicauan dengan tagar #!

>> CARI_KICAUAN testing123;
| ID = 1
| a1
| 24/11/2023 21:1:39
| testing
| #testing123
| Disukai: 0

>>

```

8 Pembagian Kerja dalam Kelompok

Nama	NIM	Tugas
Wilson Yusda	13522019	Pembuatan fitur Utas, config

		Utas, Simpan Utas, Debug Main dan Laporan, daftar masuk keluar.
Muhammad Althariq Fairuz	13522027	Pembuatan fitur Pertemanan, Daftar Permintaan Pertemanan, laporan
Ibrahim Ihsan Rasyid	13522018	Pembuatan Draft, ADT Stack, ADT Matriks, ADT Sederhana (datetime)
Farhan Nafis Rayhan	13522037	Pembuatan ADT List dengan representasi array Dinamis, Tree, DSU, Kicauan, Balasan, Config Kicauan, Config Balasan
Muhammad Sulthan Mazaya	13320028	Makefile, Mesin Kata, Mesin Karakter, Tagar, FYB, Kelompok Teman, Pengguna, Config.

9 Lampiran

9.1 Deskripsi Tugas Besar 1

Buatlah sebuah program simulasi berbasis **CLI** (*command-line interface*). Program ini dibuat dalam bahasa C dengan menggunakan struktur data yang sudah kalian pelajari di mata kuliah ini. Kalian boleh menggunakan struktur data yang sudah kalian buat untuk praktikum pada tugas besar ini. Daftar ADT dapat dilihat pada bagian [Daftar ADT](#). Kompilasi program **wajib** menggunakan **Makefile** dan wajib dapat dikompilasi dan dijalankan pada sistem operasi berbasis UNIX. Jika sistem operasi Anda adalah Windows, gunakan [WSL](#) untuk memiliki *environment* seperti UNIX. Contoh Makefile dapat dilihat pada Olympia. Perhatikan bahwa Makefile yang diberikan merupakan contoh, Anda dapat membuat Makefile yang lebih sesuai dengan kebutuhan Anda.

9.2 Notulen Rapat

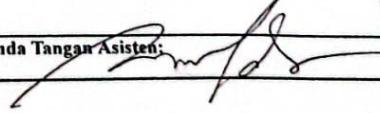
Asistensi I

<p>Tanggal : 2 Nov 2023 Tempat : Jangro Kehadiran Anggota Kelompok:</p> <table border="1"> <thead> <tr> <th>No</th><th>NIM</th><th>Tanda tangan</th></tr> </thead> <tbody> <tr><td>1</td><td>13522019</td><td>Wkt</td></tr> <tr><td>2</td><td>13320023</td><td>Jdttr</td></tr> <tr><td>3</td><td>13522018</td><td></td></tr> <tr><td>4</td><td></td><td></td></tr> <tr><td>5</td><td>13522027</td><td></td></tr> <tr><td>6</td><td>13522037</td><td></td></tr> </tbody> </table>	No	NIM	Tanda tangan	1	13522019	Wkt	2	13320023	Jdttr	3	13522018		4			5	13522027		6	13522037		<p>Catatan Asistensi: Boleh Pakai Union untuk Generic Struktur data Di laporan, tulisnya ADT list dengan Array statis/dihabis Driver dibuat seperti contoh di olympia. Reasoning pemilihan ADT untuk fitur dituliskan dengan jelas. Untuk Bonus 1, list dinamis Kicauan juga dibuat. Value dari hashmap berisi id kicauan Tagar hanga untuk Kicauan</p>
No	NIM	Tanda tangan																				
1	13522019	Wkt																				
2	13320023	Jdttr																				
3	13522018																					
4																						
5	13522027																					
6	13522037																					
	Tanda Tangan Asisten:																					

Asistensi II

<p>Tanggal : 16/11/2023 Tempat : Jangro Kehadiran Anggota Kelompok:</p> <table border="1"> <thead> <tr> <th>No</th><th>NIM</th><th>Tanda tangan</th></tr> </thead> <tbody> <tr><td>1</td><td>13522019</td><td>Wkt</td></tr> <tr><td>2</td><td>13522027</td><td></td></tr> <tr><td>3</td><td>13522037</td><td></td></tr> <tr><td>4</td><td>13522018</td><td></td></tr> <tr><td>5</td><td>13720028</td><td></td></tr> <tr><td>6</td><td>13720028</td><td></td></tr> </tbody> </table>	No	NIM	Tanda tangan	1	13522019	Wkt	2	13522027		3	13522037		4	13522018		5	13720028		6	13720028		<p>Catatan Asistensi: 1) Profil, Name apakah sudah log in apa belum 2) Wkt Profil belum keluar 3) struktur tree boleh simpan address sibling & child 4) Contig int tan jadi word dulu 5) Setuju - perintah ^{luring} freeze 6) Hapus Teman, kala buk tan tahu Nama 7) Hapus Teman, Nama masih Bub 8) Pada wkt log in mail diambil</p>
No	NIM	Tanda tangan																				
1	13522019	Wkt																				
2	13522027																					
3	13522037																					
4	13522018																					
5	13720028																					
6	13720028																					
	Tanda Tangan Asisten:																					

Asistensi III

Tanggal : 23 November 2023	Catatan Asistensi:
Tempat : Online	
Kehadiran Anggota Kelompok:	
No NIM Tanda tangan	<ol style="list-style-type: none"> 1) Untuk laporan , nama bisa di buat tabel 2) Untuk test , Nanti di bagian 7 ketik <code>sementara</code> 3) Nama ADT harus Sesuai 4) Kalau untuk driver, sebaiknya Npm harus ada <code>0001</code> 5) Untuk Data Test bisa dalam Tabel 6) Kalau mau coba laporan , boleh dokumentasi saja.
1 Wilson 13522019	
2 13522027	
3 13522018 13522027	
4 13522027	
5 13522027	
6 eusther	
	Tanda Tangan Asisten: 

9.3 Log Activity Anggota Kelompok

Nama	Week-1	Week-2	Week-3	Week-4	Week-5
Wilson Yusda	Penambahan ADT List dengan struktur data berkait	Penambahan struct utasan dan ADT sederhana Utasan	Pelengkapan fungsi kecil pada Utasan.c , daftar masuk, keluar	Penambahan fitur utasan, simpan dan konfigurasi	Penambahan Simpan dan Konfigurasi Utasan dengan merging fitur Utasan ke Main dan laporan,

Muhammad Sulthan Mazaya (13520028)	Inisialisasi make file	Penambahan fitur fitur mesin kata	Perbaikan fitur mesin kata, penambahan main dan pelengkapan fitur	Menambah beberapa fitur kicau , memperbaiki beberapa fitur	Penambahan config dan simpan pada profil
Ibrahim Ihsan Rasyid (13522018)	Inisialisasi ADT ListStatik	Penambahan ADT Datetime	Pengerjaan fitur profil	Pengerjaan fitur stack	Pengerjaan config stack
Muhammad Althariq Fairuz (13522027)	Inisialisasi ADT Priority Queue	Penambahan ADT Graph	Inisialisasi laporan	Pembersihan beberapa code dan penambahan fitur teman dan permintaan teman	Penambahan driver
Farhan Nafis Rayhan (1352237)	Inisialisasi Github	Penambahan file Kicau dan ADT ListDin	Penambahan ADT Tree	Pembuatan fitur Kicauan	Pembuatan fitur balasan dan config kicau dan balasan

9.4 Milestone

Tanggal			16-11-2023
No	Fitur	Progress (0-100%)	Keterangan (Komponen fitur yang telah selesai ataupun kendala)
1.	Inisialisasi	100%	Setting main sudah ada, hanya perlu mengintegrasikan fungsi lain
2.	Perintah	100%	Akuisisi sudah menggunakan mesin kata dan dibandingkan dengan input pengguna
3.	Pengguna	100%	Sudah ada pengecekan dan program sudah berfungsi dengan baik
4.	Profil	80%	Profil belum mengecek seseorang sudah login atau belum. Serta error pada LIHAT_PROFIL. Inisialisasi private publik masih kurang tepat, sekarang sedang dalam perbaikan
6.	Teman	90%	Pada fitur HAPUS TEMAN, output tidak sesuai test case dimana harus ada nama jika nama yang ingin dihapus tidak ada
7.	Permintaan Pertemanan	95%	Terdapat kondisi error ketika menyetujui pertemanaan
8.	Kicauan	80%	Semua fitur sudah diintegrasikan, hanya perlu pemindahan ke list dinamis.

9.	Balasan	5%	Sedang dalam proses (ADT sudah ada)
10.	Draf Kicauan	5%	Sedang dalam proses (ADT sudah ada)
11.	Utas	90%	UTAS,SAMBUNG_UTAS,CETAK_UTAS,HAPUS_UTAS sudah selesai namun masih ada bug ketika mengakses index pertama.
12.	Simpan dan Muat	10%	Config Utas sudah ada namun belum dicoba.
13.	Laporan	0%	Sedang berfokus pada fitur