

LAPORAN TUGAS BESAR

IF2111 Algoritma dan Struktur Data STI

PURRMART


Dipersiapkan oleh:

Kelompok 04 K-03

Muhamad Hasbullah Faris	18223014
Mudzaki Kaarzaqiel Hakim	18223024
Princessfa Azzahra Alvin	18223044
Fathimah Nurhumaida Ramadhani	18223052
Muhammad Rafi Ramadhan	18223094

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		<i>IF2111-TB-02-04</i>		<i>34</i>
		<i>Revisi</i>	<i>0</i>	<i>25/11/2024</i>

Daftar Isi

1	Ringkasan	2
2	Penjelasan Tambahan Spesifikasi Tugas	4
2.1	Spesifikasi Tampilan Menu Utama	4
3	Struktur Data (ADT)	5
3.1	Mesin Karakter	5
3.2	Mesin Kata	6
3.3	User	7
3.4	Boolean	7
3.5	List Dinamis	8
3.6	Queue	9
4	Program Utama	10
5	Data Test	11
5.1	Data Test Start	11
5.2	Data Test Load	12
5.3	Data Test Register	12
5.4	Data Test Login	13
5.5	Data Test Logout	13
5.6	Data Test Help	14
5.7	Data Test Save	15
5.8	Data Test Quit	15
5.9	Data Test Work	16
6	Test Script	19
7	Pembagian Kerja dalam Kelompok	23
8	Lampiran	25
8.1	Deskripsi Tugas Besar	25
8.2	Notulen Rapat	28
8.3	Log Activity Anggota Kelompok	29

1 Ringkasan

Agen Purry mendapatkan laporan bahwa Dr.Asep Spakbor sedang membuat sebuah mesin bernama ‘Oppenheimer-inator’ yang bisa menghancurkan wilayah 3 negara bagian sehingga Agen Purry memiliki misi untuk menghentikan Dr. Asep Spakbor. Agen Purry memiliki senjata rahasia dengan bantuan Toko Borma, karena Toko Borma menyediakan barang-barang untuk keperluan perang. Namun, satu-satunya hal yang menjadi hambatan bagi Agen Purry adalah Toko Borma masih beroperasi secara tatap muka dan tidak memiliki transport untuk ke Bojongsoang, sehingga tim programmer harus merancang suatu sistem jual beli ke Toko Borma untuk menyelamatkan misi Agen Purry.

Tim Programmer diminta untuk membuat sebuah aplikasi simulasi berbasis CLI (command-line interface) dengan menggunakan bahasa C dan memanfaatkan struktur data yang sudah dipelajari di kuliah, serta tim programmer juga diperbolehkan untuk memodifikasi struktur data tersebut. Tim Programmer diberikan panduan terkait daftar ADT yang bisa digunakan dan library yang boleh digunakan yakni `stdio.h`, `stdlib.h`, `time.h`, dan `math.h`.

Aplikasi PURRMART memiliki beberapa fitur utama seperti: menampilkan barang Toko Borma, meminta dan menyuplai barang baru ke Toko Borma, menyimpan dan membeli barang dalam keranjang, menampilkan barang yang sudah dibeli oleh user, membuat dan menghapus wishlist serta user dapat bekerja untuk menghasilkan uang. Langkah awal untuk memulai program adalah setelah menjalankan program, aplikasi akan memperlihatkan main menu untuk user dapat START, LOAD, dan HELP. Selain itu, user juga dapat melakukan REGISTER dan LOGIN untuk memulai eksplorasi pada aplikasi dan melakukan LOGOUT, SAVE, dan QUIT apabila sudah selesai.

Pada laporan ini akan dibahas semua terkait proses pembuatan PURRMART secara detail, dimulai dari penjelasan spesifikasi, struktur data yang digunakan, program utama, data test, test script, pembagian tugas antar anggota, dan lampiran untuk mendukung penjelasan.

3 Struktur Data (ADT)

3.1 Mesin Karakter

```
1 #ifndef MESIN_KARAKTER_H
2 #define MESIN_KARAKTER_H
3
4 #include "boolean.h"
5 #include <stdio.h>
6 #define MARK '\n'
7
8 /* State Mesin */
9 extern char currentChar;
10 extern boolean EOP;
11 extern FILE *pita;
12
13
14 void START();
15
16 void StartFile(char *filename);
17 /* Mesin siap dioperasikan. Pita disiapkan untuk dibaca.
18 Karakter pertama yang ada pada pita posisinya adalah pada jendela.
19 Pita baca diambil dari stdin.
20 I.S. : sembarang
21 F.S. : currentChar adalah karakter pertama pada pita
22        Jika currentChar != MARK maka EOP akan padan (false)
23        Jika currentChar == MARK maka EOP akan menyala (true) */
24
25 void ADV();
26
27 void AdvFile();
28 /* Pita dinajukan satu karakter.
29 I.S. : Karakter pada jendela = currentChar, currentChar != MARK
30 F.S. : currentChar adalah karakter berikutnya dari currentChar yang lama,
31        currentChar == MARK
32        Jika currentChar == MARK maka EOP akan menyala (true) */
33
34 char GetCCC();
35 /* Mengirinkan currentChar */
36
37 boolean EOP();
38 /* Mengirinkan true jika currentChar == MARK */
39
40 #endif
```

Gambar 3.1.1

Pada struktur data ini, akan dimanfaatkan nilai EOP (End of Process) yang bertipe boolean untuk menyatakan bila pita sudah mencapai akhir sehingga akan bernilai true dan jika belum akan bernilai false. ADT mesin karakter digunakan untuk mengambil input berupa string dari pengguna dan membaca file konfigurasi ke dalam aplikasi PURRMART. Alasan pemilihan struktur data mesin karakter adalah karena struktur data ini dapat digunakan untuk pembacaan dan pemrosesan data teks baik itu dari pengguna maupun dari standard input, dengan cara mendeteksi MARK atau EOF usebagai penanda akhir pembacaan.

3.2 Mesin Kata

```
1 #ifndef MESIN_KATA_H
2 #define MESIN_KATA_H
3
4 #include "mesinkarakter.h"
5
6 #define MAX_LEN 50
7 #define BLANK ' '
8 #define NEWLINE '\n'
9 #define STRIP '-'
10 #define STOP '\0'
11
12 typedef struct
13 {
14     char TabKata[MAX_LEN]; /* container menyimpan kata, indeks yang dipakai [0..Max-1] */
15     int length;
16 } Word;
17
18 /* State Mesin Kata */
19 extern boolean EndWord;
20 extern Word currentWord;
21 extern Word currentCommand;
22 extern Word hasil;
23
24 void IgnoreBlank();
25
26 void IgnoreNewline();
27
28 void StartWordInput();
29
30 // void StartWordMark();
31
32 // void StartWordBlank();
33
34 Word ConcatWord(Word word_1, Word word_2);
35
36 void StartWordFile(char *filename, int type);
37
38 void StartCommand();
39
40 void CopyCommand();
41
42 void ResetCommand();
43
44 void AddWord();
```

Gambar 3.2.1

```
46 void AddInput();
47
48 boolean IsCommandEqual(Word Input, char * kata);
49
50 void copyString(char dest[], const char src[], int max_length);
51
52 int strlen(char *string);
53
54 Word stringToWord(char *str);
55
56 void AddLine();
57
58 void CopyWord();
59
60 Word CopyWordFile();
61
62 void CopyWordInput();
63
64 void AddSemicolon();
65
66 void DisplayWord (Word Word);
67
68 boolean Command(Word hasil);
69
70 Word ConcatWord(Word word_1, Word word_2);
71
72 Word SplitWordBlank(Word word);
73
74 Word SplitWordMark(Word word);
75
76 Word SplitWordLeftBlank(Word word);
77
78 Word SplitWordLeftMark(Word word);
79
80 boolean isEqualWords(Word word1, char *str);
81
82 #endif
```

Gambar 3.2.2

Struktur data ini digunakan untuk dapat membaca input yang dimasukkan oleh pengguna seperti pada fungsi register dimana pengguna memasukkan username beserta password untuk di validasi dan apabila masukan sudah sesuai maka akan disimpan dalam variabel usernameWord dan passwordWord yang bertipe Word untuk kemudian di convert menjadi bentuk string dan disimpan dalam file. Kami memilih untuk menggunakan struktur data mesin kata karena adanya batasan dalam menggunakan scanf atau fgets pada

command dan file konfigurasi, sehingga ADT mesin kata berperan untuk mengambil masukan dari pengguna.

3.3 User

```
typedef struct {  
    char name[MAX_LEN];  
    char password[MAX_LEN];  
    int money;  
    boolean logged;  
} User;
```

Gambar 3.3.1

Struktur data untuk user merupakan tipe data bentukan yang bisa menyimpan nilai nama (name) yang bertipe string, password yang bertipe string, uang (money) yang bertipe integer dan status login pengguna (logged) yang bertipe boolean dan akan bernilai true jika user sudah login ke dalam aplikasi. ADT user dimanfaatkan untuk mengelola data pengguna dalam aplikasi PURRMART sehingga informasi pengguna dapat disimpan dan dikelola serta mudah untuk diakses. Struktur data ini juga digunakan untuk pengguna melakukan login dan logout.

3.4 Boolean

```
#ifndef _BOOLEAN_h  
#define _BOOLEAN_h  
  
#define boolean unsigned char  
#define true 1  
#define false 0  
  
#endif
```

Gambar 3.4.1

Struktur data boolean berfungsi untuk mengeluarkan nilai true (1) apabila nilai benar dan false (0) bila salah, dan untuk default dari penggunaan boolean sendiri adalah true. ADT

boolean berperan untuk membandingkan suatu kondisi yang juga dimanfaatkan dalam ADT mesin kata dan mesin karakter, serta juga membantu dalam menentukan EOP suatu proses. Kami memilih untuk memanfaatkan struktur data boolean karena memudahkan pemrosesan fungsi register login, logout, store request, store run, store supply, help, work, dan store remove.

3.5 List Dinamis

```
src > ADT > C list_dinamis > isEmptyDin(ListDin)
1  #ifndef LIST_DINAMIS_H
2  #define LIST_DINAMIS_H
3
4  #include "list.h"
5  #include <stdlib.h>
6
7  typedef struct {
8      char **items;
9      int size;
10     int capacity;
11 } ListDin;
12
13 void createListDin(ListDin *list, int capacity);
14 void dealocateListDin(ListDin *list);
15 int listLengthDin(ListDin list);
16 int isEmptyDin(ListDin list);
17 void insertLastDin(ListDin *list, char *item);
18 void deleteAtDin(ListDin *list, int index);
19 void deleteFirstDin(ListDin *list);
20 char* ELMTDin(ListDin list, int index);
21 void copyFirstDin(ListDin list, char *variable);
22
23 #endif
```

Gambar 3.5.1

ADT List Dinamis berfungsi untuk mengelola array dinamis dengan elemen berupa string. Modul ini mencakup fungsi untuk membuat list baru dengan kapasitas tertentu (CreateListDin), menghapus list dan membebaskan memori (dealocateListDin), memeriksa panjang list (listLengthDin) dan apakah list kosong (isEmptyDin), serta menambah string di akhir (insertLastDin) atau menghapus elemen baik di indeks tertentu (deleteAtDin) maupun elemen pertama (deleteFirstDin). Elemen dapat diakses menggunakan fungsi akses (ELMTDin), dan elemen pertama dapat disalin ke variabel eksternal (copyFirstDin). Memori dikelola secara manual dengan fungsi malloc dan free untuk setiap string, memastikan efisiensi dan fleksibilitas dalam menangani data berbasis string. Kami menggunakan ADT List Dinamis untuk memanipulasi daftar barang-barang yang ada di STORE melalui fungsi-fungsi terkait (storeList, storeRequest, storeSupply, storeRemove).

3.6 Queue

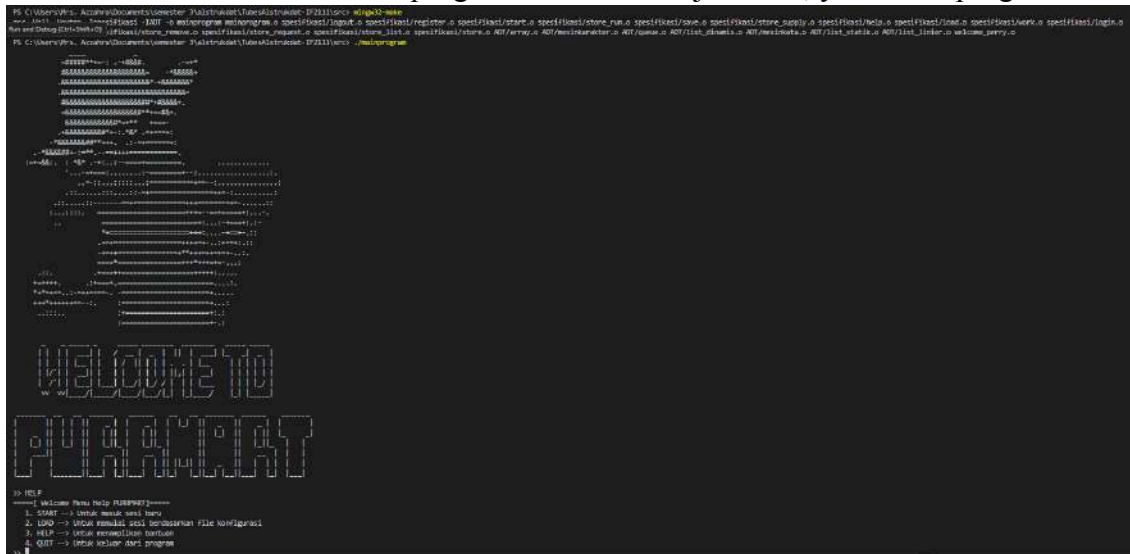
```
ADT > C queue.h > ...
1  #ifndef QUEUE_H
2  #define QUEUE_H
3
4  #include "boolean.h"
5  #include "mesinkata.h"
6  #include "mesinkarakter.h"
7
8  #define CAPACITY_QUEUE 100
9  #define MAX_LEN_QUEUE 100
10 #define IDX_UNDEF -1
11
12 typedef char QueueElementType[MAX_LEN_QUEUE];
13 typedef struct {
14     QueueElementType buffer[CAPACITY_QUEUE];
15     int idxHead;
16     int idxTail;
17 } Queue;
18
19 #define IDX_HEAD(q) (q).idxHead
20 #define IDX_TAIL(q) (q).idxTail
21 #define HEAD(q) (q).buffer[(q).idxHead]
22 #define TAIL(q) (q).buffer[(q).idxTail]
23
24 void CreateQueue(Queue *q);
25
26 boolean isEmptyQueue(Queue q);
27
28 boolean isFullQueue(Queue q);
29
30 int lengthQueue(Queue q);
31
32 void enqueue(Queue *q, QueueElementType val);
33
34 void dequeue(Queue *q, QueueElementType *val);
35
36 void displayQueue(Queue q);
37
38 void copyFirstQueue(Queue q, char *destChar);
39
40 #endif
```

Gambar 3.5.2

ADT Queue berfungsi untuk mengelola queue sirkular statis dengan elemen berupa string, menggunakan array sebagai buffer berkapasitas tetap. Queue dikelola dengan indeks kepala (idxHead) dan ekor (idxTail), yang diatur untuk menambah (enqueue) atau menghapus elemen (dequeue) secara efisien menggunakan logika indeks sirkular. Fungsi pendukung mencakup pembuatan queue baru (CreateQueue), pengecekan status kosong atau penuh (isEmptyQueue, isFullQueue), menghitung panjang queue (lengthQueue), menampilkan isi queue (displayQueue), dan menyalin elemen pertama (copyFirstQueue). Kami menggunakan ADT Queue untuk memanipulasi daftar antrian yang ada di STORE melalui fungsi-fungsi terkait (storeList, storeRequest, storeSupply, storeRemove).

4 Program Utama

Berikut adalah contoh keluaran program utama saat dijalankan, yaitu mainprogram.c:



Gambar 4.1

Program utama dimulai dengan pendefinisian header yang digunakan yaitu, <stdio.h>, <stdlib.h>, dan mainprogram.h. Total file header yang telah dibuat ada 15 yaitu: help.h, load.h, login.h, logout.h, quit.h, register.h, save.h, start.h, store_list.h, store_remove.h, store_request.h, store_run.h, store_supply.h, store.h, dan work.h. Setelah semua header telah didefinisikan, tahap berikutnya adalah pembuatan fungsi pada welcome_perry.h yang berfungsi sebagai header untuk tampilan awal menyambut user saat baru mulai menjalankan program, seperti pada gambar 5.1 diatas.

Pada bagian utama dari program int main() yang diawali dengan pemanggilan prosedur welcome_perry. Sebelum memanggil fungsi/ prosedur tertentu, program akan menampilkan tampilan menu HELP dimana para pengguna dapat memilih aksi yang ingin mereka lakukan selanjutnya meliputi START, LOAD, HELP, dan QUIT. Untuk memulai program, para pengguna bisa memilih START dan dapat melakukan REGISTER bila pengguna merupakan user baru, LOGIN bila pengguna sudah terdaftar sebagai user, dan QUIT apabila pengguna ingin keluar dari program, namun sebelum keluar dari program pengguna akan diminta untuk memasukkan file penyimpanan.. Jika pengguna memilih LOAD maka harus diikuti nama file yang kemudian akan dibaca oleh sistem. Dari masing-masing command yang berhasil dibaca oleh program, akan langsung memanggil masing-masing fungsi.

```
>> START
default.txt../save/default.txtSave file berhasil dibaca. PURRMART berhasil dijalankan.
File berhasil dimuat.
>>>> HELP
====[ Welcome Menu Help PURRMART]====
1. REGISTER --> Untuk melakukan pendaftaran akun baru
2. LOGIN --> Untuk masuk ke dalam akun dan memulai sesi
3. QUIT --> Untuk keluar dari program
```

Gambar 4.2

5 Data Test

5.1 Data Test Start

Tes ini dilakukan untuk memeriksa berjalannya program utama dan tampilan kondisi awal program. Saat program dimulai akan ditampilkan juga welcome_perry serta pesan welcome menu.



Gambar 5.1.1

```

>> HELP
====[ Welcome Menu Help PURRMART]====
  1. START --> Untuk masuk sesi baru
  2. LOAD --> Untuk memulai sesi berdasarkan file konfigurasi
  3. HELP --> Untuk menampilkan bantuan
  4. QUIT --> Untuk keluar dari program
>> START
default.txt../save/default.txtSave file berhasil dibaca. PURRMART berhasil dijalankan.
File berhasil dimuat.
3
2

```

Gambar 5.1.2

5.2 Data Test Load

Program akan membaca file yang sudah disimpan dan berisi informasi pengguna dan barang dari aplikasi PURRMART.

```

>> HELP
====[ Welcome Menu Help PURRMART]====
  1. START --> Untuk masuk sesi baru
  2. LOAD --> Untuk memulai sesi berdasarkan file konfigurasi
  3. HELP --> Untuk menampilkan bantuan
  4. QUIT --> Untuk keluar dari program
>> LOAD default.txt
default.txt../save/default.txtSave file berhasil dibaca. PURRMART berhasil dijalankan.
File berhasil dimuat.

```

Gambar 5.2.1

5.3 Data Test Register

Tes ini dilakukan untuk menguji kemampuan program membaca command REGISTER dari pengguna dan pengguna akan memasukkan username serta password sesuai ketentuan yang ada.

```

>> HELP
====[ Welcome Menu Help PURRMART]====
  1. REGISTER --> Untuk melakukan pendaftaran akun baru
  2. LOGIN --> Untuk masuk ke dalam akun dan memulai sesi
  3. QUIT --> Untuk keluar dari program
>> REGISTER
Enter username: arra
Enter password: bukan
Akun dengan username telah berhasil dibuat dengan saldo awal 0. Silakan LOGIN untuk melanjutkan.
>> HELP
====[ Welcome Menu Help PURRMART]====
  1. REGISTER --> Untuk melakukan pendaftaran akun baru
  2. LOGIN --> Untuk masuk ke dalam akun dan memulai sesi
  3. QUIT --> Untuk keluar dari program
>> LOGIN
Username: arra
Password: bukan
>> HELP
====[ Welcome Menu Help PURRMART]====
  1. WORK --> Untuk bekerja dan mendapatkan uang
  2. WORK CHALLENGE --> Untuk mengerjakan challenge
  3. STORE --> Untuk mengakses menu store
  4. LOGOUT --> Untuk keluar dari sesi
  5. SAVE --> Untuk menyimpan state ke dalam file
  6. QUIT --> Untuk keluar dari program

```

Gambar 5.3.1

5.4 Data Test Login

Tes ini dilakukan untuk menguji jika program berhasil menjalankan fungsi login dimana pengguna dengan user dan password yang sudah tersimpan di file bisa melakukan login ke aplikasi PURRMART kembali.

```
>> HELP
====[ Welcome Menu Help PURRMART]====
1. REGISTER --> Untuk melakukan pendaftaran akun baru
2. LOGIN --> Untuk masuk ke dalam akun dan memulai sesi
3. QUIT --> Untuk keluar dari program
>> LOGIN
Username: admin
Password: alstrukdatkeren
Anda telah login ke PURRMART sebagai admin.
>> HELP
====[ Welcome Menu Help PURRMART]====
1. WORK --> Untuk bekerja dan mendapatkan uang
2. WORK CHALLENGE --> Untuk mengerjakan challenge
3. STORE --> Untuk mengakses menu store
4. LOGOUT --> Untuk keluar dari sesi
5. SAVE --> Untuk menyimpan state ke dalam file
6. QUIT --> Untuk keluar dari program
>>
```

Gambar 5.4.1

```
>> HELP
====[ Welcome Menu Help PURRMART]====
1. REGISTER --> Untuk melakukan pendaftaran akun baru
2. LOGIN --> Untuk masuk ke dalam akun dan memulai sesi
3. QUIT --> Untuk keluar dari program
>> LOGIN
Username: admin
Password: alstrukdat
Username atau password salah.
```

Gambar 5.4.2

5.5 Data Test Logout

Tes ini dilakukan untuk menguji jika program berhasil mengeluarkan pengguna dari akunnya.

```
>> LOGIN
Username: admin
Password: alstrukdatkeren
Anda telah login ke PURRMART sebagai admin.
>> HELP
====[ Welcome Menu Help PURRMART]====
1. WORK --> Untuk bekerja dan mendapatkan uang
2. WORK CHALLENGE --> Untuk mengerjakan challenge
3. STORE --> Untuk mengakses menu store
4. LOGOUT --> Untuk keluar dari sesi
5. SAVE --> Untuk menyimpan state ke dalam file
6. QUIT --> Untuk keluar dari program
>> LOGOUT
>>>> LOGOUT
admin telah logout dari sistem PURRMART. Silakan REGISTER/LOGIN kembali untuk melanjutkan.
>> HELP
====[ Welcome Menu Help PURRMART]====
1. REGISTER --> Untuk melakukan pendaftaran akun baru
2. LOGIN --> Untuk masuk ke dalam akun dan memulai sesi
3. QUIT --> Untuk keluar dari program
```

Gambar 5.5.1

5.6 Data Test Help

Tes ini dilakukan untuk menguji jika program berhasil menampilkan opsi pilihan help untuk user.

```
..*0000##+:-:*~---+---+-----,
:-+00.. : *0* .-+!:-+-----, .....
'...+==!.....!-----+!.....!.....
..-!.....!.....!-----+!.....!.....
.:.....!.....!-----+!-----+!.....!
.:.....!-----+-----+!-----+!.....!
:.....!-----+-----+!-----+!.....
..      =====+!.....+!.....!:-
      *+=====+!.....+!.....!:-
      .+-----+!-----+!-----+!.....
      -+-----+!*+-----+!.....
      =====+!+*+-----+!.....
.:..      .+=====+!+*+-----+!.....
+=====+!+*+-----+!+*+-----+!.....
+*+=====+!+*+-----+!+*+-----+!.....
++*+=====+!+*+-----+!+*+-----+!.....
.:.....      :=====+!+*+-----+!.....
      :=====+!+*+-----+!.....
      :=====+!+*+-----+!.....

WELCOME TO
PURRMART

>> HELP
====[ Welcome Menu Help PURRMART]====
1. START --> Untuk masuk sesi baru
2. LOAD --> Untuk memulai sesi berdasarkan file konfigurasi
3. HELP --> Untuk menampilkan bantuan
4. QUIT --> Untuk keluar dari program
>> |
```

Gambar 5.6.1

```
>> HELP
====[ Welcome Menu Help PURRMART]====
1. REGISTER --> Untuk melakukan pendaftaran akun baru
2. LOGIN --> Untuk masuk ke dalam akun dan memulai sesi
3. QUIT --> Untuk keluar dari program
>> |
```

Gambar 5.6.2

```
>> HELP
====[ Welcome Menu Help PURRMART]====
1. WORK --> Untuk bekerja dan mendapatkan uang
2. WORK CHALLENGE --> Untuk mengerjakan challenge
3. STORE --> Untuk mengakses menu store
4. LOGOUT --> Untuk keluar dari sesi
5. SAVE --> Untuk menyimpan state ke dalam file
6. QUIT --> Untuk keluar dari program
>> |
```

Gambar 5.6.3

5.7 Data Test Save

Tes ini dilakukan untuk menguji jika program berhasil menyimpan data pengguna kedalam file teks.

```
>> SAVE p.txt
Data berhasil disimpan ke p.txt
File berhasil dimuat.
```

Gambar 5.7.1

```
save > ≡ p.txt
1 5
2 110 AK47
3 20 Lalabu
4 20 Ayam Goreng Crisbar
5 54 pokebol
6 67 fiesta nugget
7 5
8 100 admin alstrukdatkeren
9 25 praktikan kerenbangetkak
10 24 hakim 1234
11 34 pepe tobat
12 1000 rafi keren
```

Gambar 5.7.2

5.8 Data Test Quit

Tes ini dilakukan untuk menguji jika fungsi quit berhasil mengeluarkan user dari program dan mengkonfirmasi apakah user ingin menyimpan datanya atau tidak

```
>> QUIT
>>>> QUIT
Apakah kamu ingin menyimpan data sesi sekarang (Y/N)?
>> N
Data tidak disimpan.
Dadah ^_ ^/
```

Gambar 5.8.1

```
>> QUIT
>>>> QUIT
Apakah kamu ingin menyimpan data sesi sekarang (Y/N)?
>> Y
Masukkan nama file penyimpanan: pe.txt
Data disimpan ke pe.txt
Data berhasil disimpan ke pe.txt
File berhasil dimuat.
Dadah ^_ ^/
```

Gambar 5.8.2

```
save > pe.txt

1 5
2 110 AK47
3 20 Lalabu
4 20 Ayam Goreng Crisbar
5 54 pokebol
6 67 fiesta nugget
7 5
8 100 admin alstrupdatkeren
9 25 praktikan kerenbangetkak
10 24 hakim 1234
11 34 pepe tobat
12 1000 rafi keren
```

Gambar 5.8.3

5.9 Data Test Work

Tes ini dilakukan untuk menguji jika program berhasil menambahkan uang pengguna dengan menggunakan fitur kerja.

```
Daftar pekerjaan:
1. Evil Lab Assistant (pendapatan=100, durasi=14s)
2. OWCA Hiring Manager (pendapatan=4200, durasi=21s)
3. Cikapundunginator Caretaker (pendapatan=7000, durasi=30s)
4. Mewing Specialist (pendapatan=10000, durasi=22s)
5. Inator Connoisseur (pendapatan=997, durasi=15s)
Masukkan pekerjaan yang dipilih: Evil Lab Assistant
Pendapatan : 100
Durasi Kerja : 14
Anda sedang bekerja sebagai Evil Lab Assistant... harap tunggu.

Pekerjaan selesai, +100 rupiah telah ditambahkan ke akun Anda.
Isi Rekening : 200
```

Gambar 5.9.1

5.10 Data Test WorkChallenge

Tes ini dilakukan untuk menguji jika program berhasil menambahkan uang pengguna dengan menggunakan fitur permainan. Jika uang tidak cukup untuk membayar biaya main, pengguna akan ditolak. Sebaliknya jika cukup pengguna bisa bermain.


```
>> WORKCHALLENGE
Daftar challenge yang tersedia:
1. Tebak Angka (biaya main=200)
2. W0RDL399 (biaya main=500)

Masukan challenge yang hendak dimainkan: 1

Uang Anda tidak cukup! Silahkan bermain saat Anda memiliki 200 rupiah atau lebih.
```

Gambar 5.10.1

```
Tebak angka: 62

Tebakanmu benar! +500 rupiah telah ditambahkan ke akun anda.
Isi Rekening: 1300
```

Gambar 5.10.2

```
Masukan challenge yang hendak dimainkan: 2

WELCOME TO W0RDL3, YOU HAVE 5 CHANCES TO ANSWER BEFORE YOU LOSE!

-----

Masukan kata tebakan Anda: APPLE
A* P* P* L% E*

-----

Masukan kata tebakan Anda: RAVEN
A* P* P* L% E*
R* A* V% E* N%

-----

Masukan kata tebakan Anda: OPERA
A* P* P* L% E*
R* A* V% E* N%
O P E R A

-----

Selamat, Anda menang!

+1500 rupiah telah ditambahkan ke akun Anda.
```

Gambar 5.10.3

5.11 Data Test Store List

Tes ini dilakukan untuk menguji jika program berhasil membaca command “STORE LIST” dan menampilkan list barang dalam toko atau menampilkan “TOKO KOSONG” dan menampilkan list barang dalam antrian atau menampilkan “ANTRIAN KOSONG”.

```

=== STORE MENU ===
1. STORE LIST
2. STORE REQUEST
3. STORE SUPPLY
4. STORE REMOVE
5. EXIT
>> STORE LIST
TOKO KOSONG

ANTRIAN KOSONG

>> |

```

Gambar 5.11.1

5.12 Data Test Store Request

Tes ini dilakukan untuk menguji jika program berhasil membaca command “STORE REQUEST” dan menerima input nama barang yang akan ditambahkan ke antrian serta menambahkan barang tersebut ke antrian.

```

=== STORE MENU ===
1. STORE LIST
2. STORE REQUEST
3. STORE SUPPLY
4. STORE REMOVE
5. EXIT
>> STORE REQUEST
Masukkan nama barang yang diminta: buku
buku berhasil ditambahkan ke antrian.
>> STORE REQUEST
Masukkan nama barang yang diminta: laptop
laptop berhasil ditambahkan ke antrian.
>> |

```

Gambar 5.12.1

5.13 Data Test Store Supply

Tes ini dilakukan untuk menguji jika program berhasil membaca command “STORE SUPPLY” dan membaca barang yang berada di urutan pertama antrian dan menerima command “Terima/Tunda/Tolak” dan menambahkan barang ke toko, tidak menambahkan barang ke toko, atau menghapus barang dari antrian berdasarkan command. Jika barang ingin ditambahkan ke STORE LIST, maka program meminta user untuk input harga barang.

```

>> STORE SUPPLY
Apakah kamu ingin menambahkan barang buku? (Terima/Tunda/Tolak): Terima
Harga barang: 1000
buku dengan harga 1000 telah ditambahkan ke toko.
>>

```

Gambar 5.13.1

5.14 Data Test Store Remove

Tes ini dilakukan untuk menguji jika program berhasil membaca command “STORE REMOVE” dan menerima input berisi nama barang yang ingin dihapus dari toko, serta menghapus barang dari toko.

```
>> STORE REMOVE
Input barang yang akan dihapus: buku
buku telah berhasil dihapus dari toko.
>>
```

Gambar 5.14.1

```
>> STORE LIST
TOKO KOSONG

List barang yang ada di antrian:
- laptop

>> |
```

Gambar 5.14.2

6 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
-----	------------------	----------------	-------------------------	-----------------	-----------------------	-------------------

STEI- ITB	IF2111-TB-02-04	Halaman 19 dari 34 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	START	Memeriksa status file konfigurasi berhasil atau tidaknya dibaca dan menampilkan tampilan welcome_perry bila berhasil dibuka	Memberikan command 'START'	START	Gambar 5.1.1	Sesuai hasil yang diharapkan
2	LOAD	Memeriksa jika fungsi load bisa dibaca dan menampilkan tampilan ketika file berhasil dibuka	Memberikan command 'LOAD' diikuti nama file yang valid	LOAD testfile.txt	Gambar 5.2.1	Sesuai hasil yang diharapkan
3	REGISTER	Memastikan bahwa pengguna yang belum terdaftar bisa mendaftarkan username dan password baru serta default money yang dimiliki oleh pengguna baru adalah 0.	Memberikan command 'REGISTER'	REGISTER	Gambar 5.3.1	Sesuai hasil yang diharapkan
4	LOGIN	Memastikan bahwa pengguna yang sudah pernah melakukan registrasi bisa melakukan login dengan username dan password yang sesuai	Memberikan command 'LOGIN'	LOGIN	Gambar 5.4.1	Sesuai hasil yang diharapkan

5	LOGIN	Memastikan bahwa program bisa mendeteksi apabila pengguna memberi masukan yang tidak sesuai	Memberikan command 'LOGIN'	LOGIN	Gambar 5.4.2	Sesuai hasil yang diharapkan
6	LOGOUT	Memastikan bahwa pengguna sudah keluar dari akun yang digunakan	Memberikan command 'LOGOUT'	LOGOUT	Gambar 5.5.1	Sesuai hasil yang diharapkan
7	HELP	Memeriksa jika fungsi bisa menampilkan opsi apa saja yang bisa dilakukan oleh pengguna dalam 3 state berbeda	Memberikan command 'HELP'	HELP	Gambar 5.6.1 Gambar 5.6.2 Gambar 5.6.3	Sesuai hasil yang diharapkan.
8	SAVE	Memastikan bahwa fungsi load dapat menuliskan data kedalam file .txt	Memberikan command 'SAVE' diikuti nama file yang valid	SAVE p.txt	Gambar 5.7.1 Gambar 5.7.2	Sesuai hasil yang diharapkan
9	QUIT	Mengeluarkan user dari program namun tidak mensave progres kedalam txt	Memberikan command 'SAVE' dan mengkonfirmasi dengan command "N"	QUIT	Gambar 5.8.1	Sesuai hasil yang diharapkan
10	QUIT	Mengeluarkan user dari program dan mensave progres ke txt	Memberikan command 'SAVE' dan mengkonfirmasi dengan command "Y"	QUIT	Gambar 5.8.2 Gambar 5.8.3	Sesuai hasil yang diharapkan
11	WORK	Bekerja dan mendapatkan uang	Memberikan command 'WORK' dan mengkonfirmasi dengan menuliskan pekerjaan yang diinginkan.	WORK Evil Lab Assistant/ OWCA Hiring Manager/ Cikapundungin ator/ Mewing Specialist/	Gambar 5.9.1	Sesuai hasil yang diharapkan

				Inator Connossur		
12	WORK CHALLENGE	Memainkan challenge untuk mendapatkan uang	Memberikan command 'WORK CHALLENGE' dan mengkonfirmasi dengan menuliskan '1' atau '2' sesuai jenis challenge yang diinginkan.	WORK CHALLENGE 1 / 2	Gambar 5.10.1	Sesuai hasil yang diharapkan
13	TEBAK ANGKA	Memainkan challenge berupa menebak angka yang digenerate secara random untuk mendapatkan uang	Memberikan command input kata yang ingin ditebak.	62	Gambar 5.10.2	Sesuai hasil yang diharapkan
14	WORDL3	Memainkan challenge berupa menebak kata yang digenerate secara random dengan 5 kesempatan untuk mendapatkan uang	Memberikan input kata tebak yang katanya terdiri dari 5 character.	APPLE RAVEN OPERA	Gambar 5.10.3	Sesuai hasil yang diharapkan
15	STORE LIST	Memastikan bahwa program berhasil menampilkan list barang dalam toko dan menampilkan list barang dalam antrian.	Memberikan command 'STORE LIST'	STORE LIST	Gambar 5.11.1	Sesuai hasil yang diharapkan
16	STORE REQUEST	Memastikan bahwa program menerima input nama barang yang akan ditambahkan ke antrian serta menambahkan barang tersebut ke antrian.	Memberikan command 'STORE REQUEST' dan nama barang yang ingin ditambahkan ke antrian	STORE REQUEST	Gambar 5.12.1	Sesuai hasil yang diharapkan

17	STORE SUPPLY	Memastikan bahwa program membaca barang yang berada di urutan pertama antrian dan menambahkan barang ke toko, tidak menambahkan barang ke toko, atau menghapus barang dari antrian berdasarkan command. Program meminta user untuk input harga barang.	Memberikan command 'STORE SUPPLY' dan nama barang yang ingin ditambahkan ke toko dan harga barang	STORE SUPPLY	Gambar 5.13.1	Sesuai hasil yang diharapkan
17	STORE REMOVE	Memastikan bahwa program menerima input berisi nama barang yang ingin dihapus dari toko, serta menghapus barang dari toko.	Memberikan command 'STORE REMOVE' dan nama barang yang ingin dihapus dari toko	STORE REMOVE	Gambar 5.14.1 Gambar 5.14.2	Sesuai hasil yang diharapkan

7 Pembagian Kerja dalam Kelompok

NIM	Nama Lengkap	Deskripsi Tugas
18223052	Fathimah Nurhumaida Ramadhani	<ul style="list-style-type: none"> - Membuat implementasi spesifikasi store list, store request, store supply, store remove menggunakan ADT mesin kata, mesin karakter, list dinamis, dan queue - Membuat dan memodifikasi ADT serta driver untuk keperluan implementasi yang telah disebutkan di atas (ADT mesin kata, mesin karakter, list, list statis, list dinamis, dan queue). - Menyusun dan merapikan struktur file. - Membuat makefile, main program, dan membuat welcome page yang berisi desain menggunakan ASCII. - Debugging dan fixing store, store list, store request, store supply, store remove, login, register untuk mencapai kebutuhan sesuai spesifikasi. - Debugging maiprogram dan makefile serta integrasi seluruh fitur/fungsi ke mainprogram. - Mengerjakan laporan bagian deskripsi tugas besar, log activity anggota kelompok, struktur data (ADT), data test store list, data test store request, data test store supply, dan data test store remove. - Membuat driver semua ADT dan README.
18223044	Princessfa Azzahra Alvin	<ul style="list-style-type: none"> - Mengerjakan spesifikasi login, logout, register - Debugging login, logout, register - Mengisi form asistensi - Mengerjakan laporan bagian ringkasan, ADT, Notulen rapat, Log activity kelompok, Data test untuk start, load, register, login, dan logout - Mengisi laporan bagian test script untuk start, load, register, login, logout, help

		<ul style="list-style-type: none"> - Mengisi laporan bagian penjelasan tambahan spesifikasi tugas
18223014	Muhamad Hasbullah Faris	<ul style="list-style-type: none"> - Mengerjakan spesifikasi start, help, quit, dan mainprogram. - Debugging maiprogram serta integrasi seluruh fitur ke mainprogram. - Debugging load, save, register, dan login. - Mengerjakan laporan bagian test script bagian quit, work, work challenge, tebak angka, dan wordl3. - Merapikan struktur laporan. - Mengerjakan laporan bagian data test bagian help, work, work challenge.
18223024	Mudzaki Kaarzaqiel Hakim	<ul style="list-style-type: none"> - Mengerjakan spesifikasi load dan save. - Debugging maiprogram serta integrasi seluruh fitur ke mainprogram. - Debugging program load dan save - Mengerjakan laporan bagian data test - mengerjakana laporan bagian test script
18223094	Muhammad Rafi Ramadhan	<ul style="list-style-type: none"> - Mengerjakan spesifikasi work dan workchallenge

8 Lampiran

8.1 Deskripsi Tugas Besar

1. About the System

PURRMART adalah sebuah aplikasi yang dapat mensimulasikan aktivitas beli barang pada e-commerce. PURRMART memiliki beberapa fitur utama, yaitu:

- Menampilkan barang toko
- Meminta dan menyuplai barang baru ke toko
- Menyimpan dan membeli barang dalam keranjang
- Menampilkan barang yang sudah dibeli
- Membuat dan menghapus wishlist
- Bekerja untuk menghasilkan uang

2. Menu Program

Ketika program pertama kali dijalankan, PURRMART akan memperlihatkan main menu yang berisi welcome menu dan beberapa command yaitu START, LOAD, dan juga HELP. Setelah itu, program akan memasuki login menu yang memiliki command LOGIN, REGISTER, dan juga HELP. Jika pengguna berhasil memasuki kredensial suatu akun, maka mereka akan masuk ke menu selanjutnya. Main menu menerima masukan berupa command yang akan dijelaskan pada bagian berikutnya. Program akan terus menerima command sampai diberikan command QUIT yang berlaku pada seluruh menu.

3. Command

Pengguna dapat memasukkan command-command berikut.

a. START

START merupakan salah satu command yang dimasukkan pertama kali dalam Toko Purrmart. Setelah menekan Enter, dibaca file konfigurasi default yang berisi daftar barang pada toko.

b. LOAD <filename>

LOAD merupakan salah satu command yang dimasukkan pertama kali dalam PURRMART. Command ini memiliki satu argumen yaitu filename yang merepresentasikan suatu save file yang ingin dibuka. File didapatkan dari folder tertentu, contohnya save. Setelah menekan Enter, akan dibaca save file <filename> yang berisi daftar barang pada toko.

c. LOGIN

Login merupakan command yang baru dapat dipanggil setelah pengguna memulai sesi. Login berguna untuk masuk ke akun di sistem PURRMART yang sudah didaftarkan sebelumnya.

d. LOGOUT

LOGOUT merupakan salah satu command yang baru dapat digunakan setelah pengguna telah memasuki sebuah sesi.

e. REGISTER

Register merupakan command yang baru dapat dipanggil setelah pengguna memulai sesi. Register berguna untuk mendaftarkan akun baru ke dalam sistem PURRMART. Sebuah akun setidaknya memiliki atribut username dan password. Username dan password hanya terdiri dari 1 kata.

f. WORK

WORK merupakan command yang digunakan pengguna untuk mendapatkan uang. Terdapat sejumlah pekerjaan yang bisa dipilih. Setiap pekerjaan memiliki waktu tunggu yang berbeda-beda dan dengan nominal pendapatan yang berbeda-beda pula. Selama pengguna sedang bekerja, maka sistem tidak bisa digunakan hingga pekerjaan selesai dilakukan.

g. WORK CHALLENGE

WORK CHALLENGE merupakan command alternatif sebagai cara mendapatkan uang dengan melakukan challenge-challenge di OWCA. Pemain membutuhkan uang dengan jumlah tertentu untuk bisa memainkan challenge. Uang yang dibayarkan untuk bermain challenge tidak akan dikembalikan, meskipun pemain kalah dalam permainan. Terdapat dua challenge yang dapat dipilih:

a) Tebak Angka

Challenge Tebak Angka merupakan permainan yang meminta pemain menebak sebuah angka yang ditentukan oleh program. Pemain memiliki 10 (sepuluh) kesempatan untuk menebak angka yang benar. Program akan memberikan feedback apakah angka tebakan lebih besar, lebih kecil, atau sama dengan angka target. Jumlah kesempatan yang dipakai oleh pengguna akan mempengaruhi uang yang didapatkan.

b) WORDL3

Challenge WORDL3 merupakan permainan tebak kata berjumlah lima karakter. Pemain memiliki 5 (lima) kesempatan untuk menebak kata yang benar. Pada setiap giliran, program akan mencetak ulang kata yang dimasukan, tetapi dengan penanda tertentu. Huruf yang benar dan berada pada tempat yang tepat dicetak biasa. Huruf yang benar, tetapi berada di tempat yang salah diberi tanda “*” setelah hurufnya. Huruf yang tidak ada sama sekali pada kata diberi tanda “%” setelah hurufnya.

h. STORE LIST

STORE LIST adalah command yang digunakan untuk melihat barang-barang apa saja yang ada di dalam toko. Setiap barang yang ditampilkan haruslah bersifat unique.

i. STORE REQUEST

STORE REQUEST adalah command yang digunakan untuk meminta penambahan barang baru ke dalam toko. Barang-barang yang diminta akan

disimpan di dalam sebuah antrian dan akan dimasukkan ke toko menggunakan command selanjutnya. Nama barang yang masuk tidak boleh sama dengan nama barang yang sudah ada di toko atau di antrian.

j. **STORE SUPPLY**

STORE SUPPLY adalah command yang digunakan untuk menambahkan barang baru ke dalam toko berdasarkan antrian permintaan. Barang yang berada pada antrian paling depan akan dimasukkan ke toko. Pengguna dapat menerima, menunda, atau menolak permintaan.

- Jika diterima, maka program akan meminta harga dari barang dan dimasukkan ke toko.
- Jika ditunda, maka barang akan kembali masuk ke antrian
- Jika ditolak, maka barang akan dihapus dari antrian

Harus terdapat validasi agar harga barang merupakan angka yang valid (berupa angka dan bernilai lebih dari nol).

k. **STORE REMOVE**

STORE REMOVE adalah command yang dapat menghapus barang yang ada di toko. Akan dilakukan input akan barang yang akan dihapus. Beri tahu apabila proses berhasil (barang terdapat pada toko dan berhasil dihapus) ataupun tidak (barang tidak terdapat di toko).

l. **HELP**

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

m. **SAVE <filename>**

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

n. **QUIT**


QUIT merupakan command yang digunakan untuk keluar dari sesi aplikasi PURRMART.

8.2 Notulen Rapat

Form Asistensi Tugas Besar
IF2111/Algoritma dan Struktur Data STI
Sem. 1 2024/2025

No. Kelompok/Kelas : 4 / 02
Nama Kelompok : KingKuadrat
Anggota Kelompok (Nama/NIM) : 1. Muhamad Hasbullah Faris / 18223014
2. Mudzaki Kaarzaqiel Hakim / 18223024
3. Princessfa Azzahra Alvin / 18223044
4. Fathimah Nurhumaida R. / 18223052
5. Muhammad Rafi Ramadhan / 18223094

Asisten Pembimbing : Syafiq Ziyadul Arifin

Asistensi I	
Tanggal : 22/11/2024	Catatan Asistensi: 1. Menampilkan progress dari fungsi-fungsi yang sudah dibuat 2. Melakukan revisi dan debugging untuk fungsi yang masih belum dapat dijalankan 3. Memperbarui ADT agar dapat menangani stremp 4. Buatlah validasi input, disarankan mengecek panjang input (misal untuk menolak masukan kosong cek apakah panjang string adalah 0, untuk fungsi WORDL3 lakukan hal serupa) 5. Buatlah validasi jumlah uang pengguna, jika jumlah uang pengguna kurang dari biaya permainan, tolak pengguna
Tempat : Selasar Labtek 6	
Kehadiran Anggota Kelompok:	
1 Muhamad Hasbullah Faris 18223014 	
2 Mudzaki Kaarzaqiel Hakim 18223024 	
3 Princessfa Azzahra Alvin 18223044 	
4 Fathimah Nurhumaida R. 18223052 	
5 Muhammad Rafi Ramadhan 18223094 	
Syafiq Ziyadul Arifin	Tanda Tangan Asisten: 

8.3 Log Activity Anggota Kelompok

Tanggal	Anggota	Aktivitas
13 November 2024	18223014 Muhamad Hasbullah Faris 18223024 Mudzaki Kaarzaqiel Hakim 18223044 Princessfa Azzahra Alvin 18223052 Fathimah Nurhumaida R. 18223094 Muhammad Rafi Ramadhan	Diskusi awal, membagi tugas, membuat repository GitHub, dan membuat ADT berdasarkan spesifikasi.
13 November 2024	18223014 Muhamad Hasbullah Faris	Initial commit
13 November 2024	18223052 Fathimah Nurhumaida R.	feat ADT files feat store_list fix refactoring all files feat data & docs
14 November 2024	18223014 Muhamad Hasbullah Faris	Merge pull request
17 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	feat load
18 November 2024	18223014 Muhamad Hasbullah Faris	Merge pull request
18 November 2024	18223044 Princessfa Azzahra Alvin	feat logout feat register
18 November 2024	18223014 Muhamad Hasbullah Faris	Merge pull request fix path load.h and load.c
19 November 2024	18223014 Muhamad Hasbullah Faris	revert feat start and merge pull request initial commit on a new repository
19 November 2024	18223052 Fathimah Nurhumaida R.	feat ADT array, liststat, listdin, mesinkar, mesinkata, queue add docs & test data re-feat load logout register
19 November 2024	18223014 Muhamad Hasbullah Faris	feat start and merge pull request
19 November 2024	18223044 Princessfa Azzahra Alvin	refactor register
19 November 2024	18223052 Fathimah Nurhumaida R.	feat store, store_list, custom

		ADT store
19 November 2024	18223044 Princessfa Azzahra Alvin	feat login
19 November 2024	18223052 Fathimah Nurhumaida R.	feat main
19 November 2024	18223044 Princessfa Azzahra Alvin	Merge pull request
19 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	feat save
19 November 2024	18223052 Fathimah Nurhumaida R.	fix store & store_list, feat store_request, store_supply feat testing main file feat store_remove
20 November 2024	18223014 Muhamad Hasbullah Faris	feat quit fix path feat register merge branch 'main' into be merge pull request
20 November 2024	18223052 Fathimah Nurhumaida R.	fix store_request feat makefile fix makefile, cobamain, store_request
20 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	feat load & save
21 November 2024	18223014 Muhamad Hasbullah Faris	fix feat quit and merge pull request
21 November 2024	18223052 Fathimah Nurhumaida R.	fix ADT mesinkarakter, store_request, cobamain
21 November 2024	18223014 Muhamad Hasbullah Faris	fix feat quit fix feat start merge pull request
21 November 2024	18223052 Fathimah Nurhumaida R.	fix ADT list_dinamis, store_request, store_supply fix all store, fix cobainmain fix ADT list_dinamis fix all store final
21 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	feat load & save
21 November 2024	18223044 Princessfa Azzahra Alvin	refactor login

		refactor register merge pull request
21 November 2024	18223014 Muhamad Hasbullah Faris	fix feat quit
22 November 2024	18223052 Fathimah Nurhumaida R.	fix store_supply fix boolean directory fix store feat isEqualWords in ADT mesinkata
22 November 2024	18223094 Muhammad Rafi Ramadhan	feat add work and workchallenge feature feat add sleep function fix: fixdriverWork by removing sleep function
22 November 2024	18223014 Muhamad Hasbullah Faris 18223024 Mudzaki Kaarzaqiel Hakim 18223044 Princessfa Azzahra Alvin 18223052 Fathimah Nurhumaida R. 18223094 Muhammad Rafi Ramadhan	Asistensi 1
22 November 2024	18223052 Fathimah Nurhumaida R.	debug main fix ADT list fix ADT list_statik
22 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	fix load and save
22 November 2024	18223044 Princessfa Azzahra Alvin	refactor login logout register
22 November 2024	18223052 Fathimah Nurhumaida R.	add list.h, fix conflict list_dinamis & list_statik
22 November 2024	18223044 Princessfa Azzahra Alvin	refactor logout merge branch main refactor login
22 November 2024	18223052 Fathimah Nurhumaida R.	fix integration of store and load
22 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	fix load
22 November 2024	18223052 Fathimah Nurhumaida R.	feat welcome page fix ADT mesin karakter & mesin kata, input ends with \n

23 November 2024	18223052 Fathimah Nurhumaida R.	fix all store: use queue in antrian
23 November 2024	18223052 Fathimah Nurhumaida R.	fix login register validation and data type, fix mainprogram and makefile, feat welcome page to mainprogram
23 November 2024	18223052 Fathimah Nurhumaida R.	fix welcome page
23 November 2024	18223044 Princessfa Azzahra Alvin	fix logout
23 November 2024	18223014 Muhamad Hasbullah Faris	fix feat help fix feat quit add main fix main
24 November 2024	18223052 Fathimah Nurhumaida R.	fix help state assignment, store_request, main program fix main program remove cobamain & all headers fix makefile error
24 November 2024	18223094 Muhammad Rafi Ramadhan	fix: change isiRekening parameter into a pointer
24 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	fix load and save merge branch main
25 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	fixing error fix load and save
25 November 2024	18223014 Muhamad Hasbullah Faris	fix feat load fix main fix feat start fix main
25 November 2024	18223052 Fathimah Nurhumaida R.	fix startload, revert startload fix work.c work.h from rafi
25 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	fix save merge branch
25 November 2024	18223014 Muhamad Hasbullah Faris	fix feat quit

25 November 2024	18223024 Mudzaki Kaarzaqiel Hakim	fix quit
25 November 2024	18223044 Princessfa Azzahra Alvin	fix register
25 November 2024	18223052 Fathimah Nurhumaida R.	feat driver ADT for boolean, list_dinamis, queue feat driver ADT for mesinkarakter, mesinkata, list_statik update README
25 November 2024	18223044 Princessfa Azzahra Alvin	fix load, login, mainprogram
25 November 2024	18223014 Muhamad Hasbullah Faris	fix main fix feat help update mainprogram.c