**TUGAS BESAR**

**SIMULASI ENGI’S KITCHEN**

**IF1210 DASAR PEMROGRAMAN**

**KELAS 05**

Oleh

**Kelompok 4**

|  |  |
| --- | --- |
| **Yudhistira Qasthari Putra** | **16516148** |
| **Muhammad Isfan Rahadi** | **16517082** |
| **R. Rafi Sageri** | **16517383** |
| **Ilham Muliawan Hamzah** | **16517040** |
| **Christian Justin** | **16517348** |
| **Hafizh Mulya Harjono** | **16517376** |
|  |  |



## SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

**INSTITUT TEKNOLOGI BANDUNG  
BANDUNG  
2018**

**DAFTAR ISI**

DAFTAR ISI2

DAFTAR TABEL3

1. PEMBAGIAN KERJA ANGGOTA KELOMPOK4
2. CHECKLIST HASIL RANCANGAN, IMPLEMENTASI, DAN TESTING4
3. DESKRIPSI PERSOALAN5
4. DESAIN COMMAND 7
5. DESAIN KAMUS DATA9
6. DESAIN DEKOMPOSISI ALGORITMIK DAN FUNGSIONAL

PROGRAM UTAMA15

1. SPESIFIKASI SETIAP FITUR DALAM UNIT20
2. PENGUJIAN24

LAMPIRAN26

**DAFTAR TABEL**

Tabel 1. Tabel Pembagian Kerja Anggota Kelompok4

Tabel 2. Tabel Checklist Hasil Rancangan, Implementasi, dan Testing4

Tabel 3. Tabel Spesifikasi Tiap Fitur Dalam Unit20

Tabel 4. Tabel Pengujian Program24

**1. PEMBAGIAN KERJA ANGGOTA KELOMPOK**

Tabel 1. Tabel Pembagian Kerja Anggota Kelompok

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fitur** | **Implementasi** | **NIM Desainer** | **NIM Koder** | **NIM Tester** |
| F1 | Type Bentukan  Procedure load | 16517040 | 16517040 | 16517383 |
| F2 | Procedure Exit | 16517376 | 16517376 | 16517376 |
| F3 | Procedure startsimulasi | 16517376 | 16517376 | 16517376 |
| F4 | Procedure stopsimulasi | 16517376 | 16517376 | 16517376 |
| F5 | Procedure belibahan | 16517082 | 16517082 | 16517082 |
| F6 | Procedure olahbahan | 16517082 | 16517082 | 16517082 |
| F7 | Procedure jualolahan | 16517148 | 16517148 | 16517148 |
| F8 | Procedure jualresep | 16517148 | 16517148 | 16517383 |
| F9 | Procedure Makan | 16517348 | 16517348 | 16517348 |
| F10 | Procedure Istirahat | 16517348 | 16517348 | 16517348 |
| F11 | Procedure Tidur | 16517348 | 16517383 | 16517383 |
| F12 | Procedure LihatStatistik | 16517376 | 16517376 | 16517376 |
| F13 | Procedure LihatInventori | 16517383 | 16517383 | 16517383 |
| F14 | Procedure LihatResep | 16517383 | 16517383 | 16517383 |
| F15 | Procedure CariResep | 16517082 | 16517082 | 16517082 |
| F16 | Procedure TambahResep | 16517148 | 16517148 | 16517148 |
| F17 | Procedure UpgradeInventori | 16517082 | 16517082 | 16517082 |
| B4 | Procedure Restock | 16517082 | 16517082 | 16517082 |

**2. CHECKLIST HASIL RANCANGAN, IMPLEMENTASI, DAN TESTING**

Tabel 2. Tabel Checklist Hasil Rancangan, Implementasi, dan Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Fitur** | **Desain** | **Implementasi** | **Testing** |
| F1 | V | V | V |
| F2 | V | V | V |
| F2 | V | V | V |
| F3 | V | V | V |
| F4 | V | V | V |
| F5 | V | V | V |
| F6 | V | V | V |
| F7 | V | V | V |
| F8 | V | V | V |
| F9 | V | V | V |
| F10 | V | V | V |
| F11 | V | V | V |
| F12 | V | V | V |
| F13 | V | V | V |
| F14 | V | V | V |
| F15 | V | V | V |
| F16 | V | V | V |
| F17 | V | V | V |
| B4 | V | V | V |

**3. DESKRIPSI PERSOALAN**

Program ini ditujukan kepada seorang chef yang ingin membuka sebuah tempat makan bernama Engi’s kitchen. Program dibuat untuk membuat simulasi bagaimana menjalankan restoran selama maksimum 10 hari. Chef hari melakukan load terlebih dahulu yaitu memasukan data data sesuai dengan spesifikasi yang diinginkan. Setelah load, chef dapat memilih start simulasi yang akan di lakukan. Ada Pilihan simulasi berada diantara 0 sampai 10, yang berarti apabila memilih 0 artinya belum mulai dan jika memilih 10 artinya sudah selesai.

Setelah memilih simulasi yang akan dilakukan, baru chef dapat melakukan aktivitas ( aksi aksi) yang terdefinisi. Saat pertama kali Engi’s kitchen dibekali modal tertentu sebagai pendapatan awal seperti restoran pada umum nya. Dengan modal awal itu, Chef dapat melakukan hal hal yang dapat mengurangi energi ataupun menambah energi. pertama, chef melakukan beli bahan mentah di supermarket. Untuk setiap bahan mentah dicatat harga satuan untuk tiap *piece-*nya dan durasi kadaluarsanya (dalam hari). Bahan mentah yang dibeli dari supermarket disimpan dalam inventori. Didata jumlah unit yang dibeli dan tanggal pembelian untuk setiap bahan mentah. Tanggal kadaluarsa suatu bahan mentah yang disimpan dalam inventori dihitung berdasarkan tanggal pembelian.

Selanjutnya chef dapat melakukan olah bahan yaitu bahan mentah yang telah dibeli, diolah menjadi bahan olahan yang memiliki harga lebih tinggi. Bahan olahan terbuat dari 1 jenis bahan mentah atau lebih (maksimum dibuat dari 10 jenis bahan mentah). Diasumsikan dibutuhkan hanya 1 *piece* untuk tiap bahan mentah. Harga jual bahan olahan lebih tinggi atau minimum sama dengan total penjumlahan harga tiap bahan mentah yang menyusunnya. Kadaluarsa semua bahan olahan adalah sama, yaitu 3 hari. Tanggal kadaluarsa bahan olahan dihitung berdasarkan tanggal dibuatnya bahan olahan tersebut. Pembuatan bahan olahan mengurangi jumlah bahan mentah yang tersimpan di inventori. Bahan olahan ini juga dijual di restoran ini untuk menambah keuntungan.

Lalu bahan mentah dan bahan olahan itu disimpan dalam inventori, namun total bahan mentah dan inventori memiliki batasan yang dapat di upgrade dengan harga tertentu. Selanjutnya chef dapat membuat resep tanpa mengurangi energi. Untuk setiap resep dicatat daftar bahan yang dibutuhkan untuk membuat 1 porsi hidangan dari resep tersebut (maksimum 20 bahan untuk menyusun resep). Bahan penyusun resep dapat berupa bahan mentah maupun bahan olahan. Diasumsikan dibutuhkan hanya 1 *piece* bahan mentah atau bahan olahan untuk membuat 1 porsi hidangan dari suatu resep. Untuk tiap resep, disimpan harga hidangan dari resep tersebut. Chef memutuskan setiap hidangan memiliki harga jual minimum 12,5% lebih tinggi dari total seluruh bahan penyusunnya. Setiap pembuatan masakan dari suatu resep, mengurangi jumlah bahan mentah dan bahan olahan yang tersimpan di inventori. Chef dapat menambahkan resep baru yang dibuat dari bahan yang ada

Untuk melakukan aktivitias yang dapat mengurangi energi seperti hal diatas, maka chef memiliki sejumlah hal yang dapat menambah energi diantara nya yaitu : Tidur. Istirahat, dan makan. Namun masing masing hal yang dilakukan memiliki limit atau batasan tertentu untuk setiap hari nya. yaitu: 1) Makan: akan menambah energi sebanyak 3 buah, namun maksimum dalam 1 hari hanya boleh 3x makan; 2) Istirahat: akan menambah energi sebanyak 1 buah, maksimum dalam 1 hari hanya boleh istirahat sebanyak 6x; 3) Tidur: untuk memulihkan energi menjadi 10 buah. Jika energi habis, aksi yang yang dapat dilakukan oleh Chef hanya tidur. Jika Chef tidur, terjadi pergantian hari. Jika terjadi pergantian hari, secara otomatis bahan-bahan mentah dan olahan yang kadaluarsa pada hari yang baru, dihapus dari inventori.

Selain hal hal yang dapat dilakukan diatas, chef juga dapat melakukan hal seperti : melihat inventori, lihat resep, cari resep, tambah resep, dan upgrade inventori yang dapat dilakukan kapanpun tanpa mengurangi energi, simulasi dihentikan jika jumlah hari sudah mencapai 10 hari. Simulasi dapat dihentikan pada hari tertentu atau dimulai pada hari tertentu.

**4. DESAIN COMMAND**

1. > load //F1

2. >>

>exit //F2

>startSimulasi //F3

>>Simulasi berapa yang ingin dijalankan?

>Start

Mulai simulasi

>lihatInventori //F13

>>Inventori simulasi berapa yang ingin dilihat?

>Membaca Inventori simulasi

>lihatResep //F14

>cariResep //F15

>>Cari resep:

>tambahResep //F16

>> Nama resep:

>> Bahan mentah:

>> Bahan olahan:

>> Harga:

>upgradeInventori //F17

>>Inventori simulasi berapa yang ingin di-upgrade?

Meng-upgrade inventori simulasi

>Command Salah //command ga ada di menu

3. >>

>stopSimulasi //F4

Simulasi dihentikan

>beliBahan //F5

>>Beli bahan:

>> Jumlah:

>olahBahan //F6

>>Buat olahan:

>>Jumlah:

>jualOlahan //F7

>>Nama bahan olahan:

>> Jumlah:

>jualResep //F8

>>Masukkan nama resep:

>makan //F9

Makan sukses

>istirahat //F10

Istirahat sukses

>tidur //F11

Anda telah tidur dan tubuh Anda telah kembali fit.

>lihatStatistik //F12

Nomor simulasi:

Tanggal awal simulasi:

Jumlah hari hidup:

Jumlah energi:

Kapasitas maksimum inventori:

Total bahan mentah dibeli:

Total bahan olahan dibuat:

Total bahan olagan dijual:

Total resep dijual:

Total pemasukan:

Total pengeluaran:

Total uang:

>lihatInventori //F13

>lihatResep //F14

>cariResep //F15

>>Cari resep:

>tambahResep //F16

>> Nama resep:

>> Bahan mentah:

>> Bahan olahan:

>> Harga:

>upgradeInventori //F17

Inventori berhasil diupgrade

**5. DESAIN KAMUS DATA**

constant arrMin : integer = 1

constant arrMax : integer = 100

type tanggal :

< hari : integer,

bulan : integer,

tahun : integer >

type mentah :

< nama : array [arrMin..arrMax] of char,

harga : integer,

kadaluarsa : integer >

type tabMentah :

< tab : array [arrMin..arrMax] of mentah,

neff : integer >

type olahan :

< nama : array [arrMin..arrMax] of char,

harga : integer,

n : integer,

komposisi : array [arrMin..arrMax] of mentah >

type tabOlahan :

< tab : array [arrMin..arrMax] of olahan,

neff : integer >

type inventoriM :

< nama : array [arrMin..arrMax] of char,

tglbeli : tanggal,

jumlah : integer >

type tabInventoriM :

< tab : array [arrMin..arrMax] of inventoriM,

neff : integer,

ntot : integer >

type listInventoriM :

< list : array [1..10] of tabInventoriM,

neff : integer >

type inventoriO :

< nama : array [arrMin..arrMax] of char,

tglbuat : tanggal,

jumlah : integer,

harga : integer, >

type tabInventoriO :

< tab : array [arrMin..arrMax] of inventoriO,

neff : integer,

ntot : integer >

type listInventoriO :

< list : array [1..10] of tabInventoriO,

neff : integer >

type resep :

< nama : array [arrMin..arrMax] of char,

harga : integer,

nmentah : integer,

partmentah : array [arrMin..arrMax] of mentah,

nolahan : integer,

partolahan : array [arrMin..arrMax] of olahan >

type tabResep :

< tab : array [arrMin..arrMax] of resep,

neff : integer >

type Mresep :

< bahan : array [arrMin..arrMax] of string,

neff : integer >

type Oresep :

< bahan : array [arrMin..arrMax] of string,

neff : integer >

type simulasi :

< nomor : integer,

awalsim : tanggal,

totalhari : integer,

energi : integer,

kapasitas : integer,

totalmentahbeli : integer,

totalbahanbuat : integer,

totalbahanjual : integer,

totalresepjual : integer,

pemasukan : integer,

pengeluaran : integer,

uang : integer >

type listSimulasi :

list : array [1..10] of simulasi,

neff : integer >

function cekBahanMentah (bahan : string ; daftarBahan : tabmentah) -> integer

function cekBahanOlahan (bahan : string ; daftarBahan : tabmentah) -> integer

procedure LoadBahanMentah (input/output bahanMentah : tabmentah)

procedure LoadBahanOlahan (input/output bahanOlahan : tabolahan)

procedure LoadInventoriMentah (input tabmentah : tabInventoriM)

procedure LoadInventoriOlahan (input tabolahan : tabInventoriO)

procedure LoadResep (input/output bahanResep : tabResep; daftarmentah : tabmentah; daftar olahan : tabolahan)

procedure LoadInventoriFileSimulasi (input nfile : integer; input nama : string; input/output arr : listSimulasi)

procedure SaveFileSimulasi (input nfile : integer; input nama : string; input/output arr : listSimulasi)

procedure SavedInventoriMentah (input nfile : integer; input nama : string; input/output arr : listInventoriM)

procedure SaveInventoriOlahan (input nfile : integer; input nama : string; input/output arr : listInventoriO)

procedure SaveResep (input nfile : integer; input nama : string; input/output resep: tabResep)

procedure SaveBahanOlahan (input nfile : integer; input nama : string; input/output olahan : tabOlahan)

procedure Call0 (input/output simN : integer; input/output eoSim : boolean; input/output eoProg : boolean; input/output DftrSim : listSimulasi; input/output DftrResep : tabResep; input/output DftrMentah : tabMentah; input/output DftrOlahan : tabOlahan; input/output invMentah : listInventoriM; input/output invOlahan : listInventoriO)

procedure Call1 (input/output simN : integer; input/output nMakan : integer; input/output nIstirahat : integer; input/output eoSim : boolean; input/output eoProg : boolean; input/output bTidur : boolean; input/output DftrSim : listSimulasi; input/output DftrResep : tabResep; input/output DftrMentah : tabMentah; input/output DftrOlahan : tabOlahan; input/output invMentah : listInventoriM; input/output invOlahan : listInventoriO; input/output TglHariIni : tanggal)

procedure beliBahan ( input a : tabMentah, input/output b : tabInventoriM, input/output c : tabInventoriO, input/output u : integer, input/output e : integer, input/output imax: integer, input/output tglnow : tanggal)

procedure olahBahan (input o : tabOlahan, input/output a : tabInventoriM, input/output b : tabInventoriO, input/output e {energi} : integer, input/output tglnow : tanggal)

procedure makan (input/output daftarsimulasi: listSimulasi, input n : integer, input/output nmakan: integer)

procedure istirahat (input/output daftarsimulasi: listSimulasi; input n : integer, input/output nistirahat: integer)

procedure lihatStatistik (input/output nfile: integer, input/output arr: listSimulasi)

procedure lihatInventoriM (input/output x : tabInventoriM)

procedure lihatInventoriO (input/output x : tabInventoriO)

procedure lihatInventori (input/output x1 : tabInventoriM, input/output x2 : tabInventoriO)

procedure lihatResep (input/output x : tabResep)

procedure urutString (input/output s1, s2 : string)

function isUrut (s1, s2 : string) -> boolean

procedure cariResep (input/output a : tabResep)

procedure upgradeInventori (input/output imax : integer)

function cariBahanMentah (bahan : string; daftarBahan : TabInventoriM) -> integer

function cariBahanOlahan (bahan : string; daftarBahan : TabInventoriO) -> integer

function cariResep (resep : string; daftarResep : tabResep) -> integer

function isBahanMentahValid (bahan : string; n : integer; daftarBahan : TabInventoriM) -> boolean

function isBahanOlahanValid (bahan : string; n : integer; daftarBahan : TabInventoriO) -> boolean

procedure jualResep (input/output BahanMentah : ListInventoriM, input/output BahanOlahan : ListInventoriO, input ResepJadi : tabresep, input/output Energi : integer, input/output Pendapatan : integer, input Nsim : integer)

procedure CariOlahan (input/output JualO : String, input BahanOlahan : ListInventoriO, input/output FoundOlahan : Boolean, input Nsim : integer, input/output i : integer)

procedure KurangiOlahan (input JumlahJualO : integer, input BahanOlahan : ListInventoriO, input/output CountOlahan : Boolean, input Nsim : integer, input i : integer)

procedure KurangiEnergi (input/output Energi : integer)

procedure TambahPendapatanOlahan (input/output Pendapatan : integer, input JumlahJualO : integer, input BahanOlahan : ListInventoriO, input Nsim : integer, input i : integer)

procedure JualOlahan (input/output BahanOlahan : ListInventoriO, input/output Energi : integer, input/output Pendapatan : integer, input Nsim : integer)

procedure InputDataResep (input/output NamaResep : string, input/output MentahResep : Mresep, input/output OlahanResep : Oresep, input/output HargaResep : integer)

function IsResepAda (ResepJadi : tabresep, NamaResep : string) -> boolean

function IsBahanPembentukCukup (MentahResep : Mresep; OlahanResep : Oresep) -> boolean

function IsBahanMentahAda (MentahResep : Mresep, BahanMentah : TabMentah) -> boolean

function IsBahanOlahanAda (OlahanResep : Oresep, BahanOlahan : TabOlahan) -> boolean

function IsHargaValid (HargaResep : integer, MentahResep : Mresep, OlahanResep : Oresep, BahanMentah : TabMentah, BahanOlahan : TabOlahan) -> boolean

procedure KurangiEnergi (input/output Energi : integer)

procedure InputResepBaru (input NamaResep : string, input MentahResep : Mresep, input OlahanResep : Oresep, input HargaResep : integer, input/output ResepJadi : tabresep)

procedure TambahResepBaru (input BahanMentah : TabMentah, input BahanOlahan : TabOlahan, input/output ResepJadi : tabresep, input/output Energi : integer)

procedure Restock (input/output a : tabInventoriM; input/output b : tabInventoriO; input/output imax : integer; input s : listSimulasi; input n : integer)

function cariKadaluarsa (bahan : string; daftarMentah : tabmentah) -> integer

function isKadaluarsaM (i : integer; daftarmentah : tabMentah; inventoriM : listInventoriM; sim : listSimulasi; nfile : integer) -> boolean

function isKadaluarsaO (i : integer; daftarolahan : tabOlahan; inventoriO : listInventoriO; sim : listSimulasi; nfile : integer)-> boolean;

procedure buangKadaluarsa (input sim : listSimulasi; input InventoriM : listInventoriM; input mentah : tabmentah; input InventoriO : listInventoriO; input olahan : tabolahan; input nfile : integer);

procedure tidur (output simulasi : listSimulasi; output BTidur : boolean; input nfile : integer);

**6. DESAIN DEKOMPOSISI ALGORITMIK DAN FUNGSIONAL PROGRAM UTAMA**

**NOTASI ALGORITMIK**

**program <main>**

uses

F1,

F2,

F12,

Typeuniverse,

Penanggalan,

CallSim

**KAMUS**

i, nMakan, nIstirahat, simN : integer

bTidur, eoDay, eoSim, eoProg : Boolean

TglHariIni : tanggal

DftrResep : tabResep

DftrMentah : tabMentah

DftrOlahan : tabOlahan

invMentah : listInventoriM

invOlahan : listInventoriO

DftrSim : listSimulasi

**ALGORITMA**

Output(“> load”)

LoadFileSimulasi(1, 'simul1.txt', DftrSim)

LoadFileSimulasi(2, 'simul2.txt', DftrSim)

LoadFileSimulasi(3, 'simul3.txt', DftrSim)

LoadFileSimulasi(4, 'simul4.txt', DftrSim)

LoadFileSimulasi(5, 'simul5.txt', DftrSim)

LoadFileSimulasi(6, 'simul6.txt', DftrSim)

LoadFileSimulasi(7, 'simul7.txt', DftrSim)

LoadFileSimulasi(8, 'simul8.txt', DftrSim)

LoadFileSimulasi(9, 'simul9.txt', DftrSim)

LoadFileSimulasi(10, 'simul10.txt', DftrSim)

LoadInventoriMentah(1, 'inventoribahanmentah1.txt', invMentah)

LoadInventoriMentah(2, 'inventoribahanmentah2.txt', invMentah)

LoadInventoriMentah(3, 'inventoribahanmentah3.txt', invMentah)

LoadInventoriMentah(4, 'inventoribahanmentah4.txt', invMentah)

LoadInventoriMentah(5, 'inventoribahanmentah5.txt', invMentah)

LoadInventoriMentah(6, 'inventoribahanmentah6.txt', invMentah)

LoadInventoriMentah(7, 'inventoribahanmentah7.txt', invMentah)

LoadInventoriMentah(8, 'inventoribahanmentah8.txt', invMentah)

LoadInventoriMentah(9, 'inventoribahanmentah9.txt', invMentah)

LoadInventoriMentah(10, 'inventoribahanmentah10.txt', invMentah)

LoadInventoriOlahan(1, 'inventoribahanolahan1.txt', invOlahan)

LoadInventoriOlahan(2, 'inventoribahanolahan2.txt', invOlahan)

LoadInventoriOlahan(3, 'inventoribahanolahan3.txt', invOlahan)

LoadInventoriOlahan(4, 'inventoribahanolahan4.txt', invOlahan)

LoadInventoriOlahan(5, 'inventoribahanolahan5.txt', invOlahan)

LoadInventoriOlahan(6, 'inventoribahanolahan6.txt', invOlahan)

LoadInventoriOlahan(7, 'inventoribahanolahan7.txt', invOlahan)

LoadInventoriOlahan(8, 'inventoribahanolahan8.txt', invOlahan)

LoadInventoriOlahan(9, 'inventoribahanolahan9.txt', invOlahan)

LoadInventoriOlahan(10, 'inventoribahanolahan10.txt', invOlahan)

LoadBahanMentah(DftrMentah)

LoadBahanOlahan(DftrOlahan)

LoadResep(DftrResep, DftrMentah, DftrOlahan)

repeat

eoSim<-True

Call0(simN, eoSim, eoProg, DftrSim, DftrResep, DftrMentah, DftrOlahan, invMentah, invOlahan)

if not(eoSim) then

TglHariIni <- DftrSim.list[simN].awalsim

depend on DftrSim.list[simN].totalhari

DftrSim.list[simN].totalhari >10:

output(“Simulasi sudah melewati 10 hari”)

output(“Simulasi tidak dapat dilanjutkan”)

lihatStatistik(simN, DftrSim)

eoSim<-True

(DftrSim.list[simN].totalhari>=1) and not(eoSim):

for i:=1 to DftrSim.list[simN].totalhari do

GantiHari(TglHariIni)

while not(eoSim) and not(eoProg) do

eoDay<- false

nIstirahat <-0

nMakan <-0

bTidur <-false

repeat

if DftrSim.list[simN].energi=0 then

Output(“Energi habis, anda hanya dapat tidur”)

Output(“> Tidur”)

else

Call1(simN, nMakan, nIstirahat, eoSim, eoDay, bTidur, DftrSim, DftrResep, DftrMentah, DftrOlahan, invMentah, invOlahan, TglHariIni)

until eoDay or eoSim

if not(eoSim) then

DftrSim.list[simN].totalhari<-DftrSim.list[simN].totalhari+1

GantiHari(TglHariIni)

SaveFileSimulasi(1, 'simul1.txt', DftrSim)

SaveFileSimulasi(2, 'simul2.txt', DftrSim)

SaveFileSimulasi(3, 'simul3.txt', DftrSim)

SaveFileSimulasi(4, 'simul4.txt', DftrSim)

SaveFileSimulasi(5, 'simul5.txt', DftrSim)

SaveFileSimulasi(6, 'simul6.txt', DftrSim)

SaveFileSimulasi(7, 'simul7.txt', DftrSim)

SaveFileSimulasi(8, 'simul8.txt', DftrSim)

SaveFileSimulasi(9, 'simul9.txt', DftrSim)

SaveFileSimulasi(10, 'simul10.txt', DftrSim)

SaveInventoriMentah(1, 'inventoribahanmentah1.txt', invMentah)

SaveInventoriMentah(2, 'inventoribahanmentah2.txt', invMentah)

SaveInventoriMentah(3, 'inventoribahanmentah3.txt', invMentah)

SaveInventoriMentah(4, 'inventoribahanmentah4.txt', invMentah)

SaveInventoriMentah(5, 'inventoribahanmentah5.txt', invMentah)

SaveInventoriMentah(6, 'inventoribahanmentah6.txt', invMentah)

SaveInventoriMentah(7, 'inventoribahanmentah7.txt', invMentah)

SaveInventoriMentah(8, 'inventoribahanmentah8.txt', invMentah)

SaveInventoriMentah(9, 'inventoribahanmentah9.txt', invMentah)

SaveInventoriMentah(10, 'inventoribahanmentah10.txt', invMentah)

SaveInventoriOlahan(1, 'inventoribahanolahan1.txt', invOlahan)

SaveInventoriOlahan(2, 'inventoribahanolahan2.txt', invOlahan)

SaveInventoriOlahan(3, 'inventoribahanolahan3.txt', invOlahan)

SaveInventoriOlahan(4, 'inventoribahanolahan4.txt', invOlahan)

SaveInventoriOlahan(5, 'inventoribahanolahan5.txt', invOlahan)

SaveInventoriOlahan(6, 'inventoribahanolahan6.txt', invOlahan)

SaveInventoriOlahan(7, 'inventoribahanolahan7.txt', invOlahan)

SaveInventoriOlahan(8, 'inventoribahanolahan8.txt', invOlahan)

SaveInventoriOlahan(9, 'inventoribahanolahan9.txt', invOlahan)

SaveInventoriOlahan(10, 'inventoribahanolahan10.txt', invOlahan)

SaveBahanOlahan(DftrOlahan)

SaveResep(DftrResep)

until eoProg = True

**DEKOMPOSISI FUNGSIONAL**

1. F1 berfungsi untuk melakukan load seluruh file eksternal. Prosedur yang dibutuhkan yaitu procedure loadBahanMentah, LoadBahanOlahan, LoadInventoriMentah, LoadInventoriOlahan, LoadResep, LoadInventoriFileSimulasi.
2. F2 berfungsi untuk melakukan exit dan save game. Prosedur yang dibutuhkan yaitu : procedure SaveFileSimulasi, SavedInventoriMentah, SaveInventoriOlahan, SaveResep.
3. F3 dan F4 terdapat pada unit callSim.
4. Callsim, berfungsi untuk melakukan memulai simulasi. Prosedur yang dibutuhkan yaitu procedure call0, call1.
5. F5 berfungsi untuk pembelian bahan makanan dari supermarket. Prosedur yang dibutuhkan yaitu procedure beliBahan.
6. F6 berfungsi untuk mengolah bahan mentah menjadi bahan olahan. Prosedur yang dibutuhkan yaitu procedure olahBahan.
7. F7 berfungsi untuk menjual bahan olahan. Prosedur yang dibutuhkan yaitu procedure JualOlahan.
8. F8 berfungsi untuk menjual hidangan berdasarkan resep. Prosedur yang dibutuhkan yaitu procedure JualResep.
9. F9 berfungsi untuk melakukan aksi makan. Prosedur yang dibutuhkan yaitu procedure makan.
10. F10 berfungsi untuk melakukan istirahat. Prosedur yang dibutuhkan yaitu procedure istirahat.
11. F11, berfungsi untuk tidur. Prosedur yang dibutuhkan yaitu procedure tidur.
12. F12, berfungsi untuk menampilkan data statistik. Prosedur yang dibutuhkan yaitu procedure lihatStatistik.
13. F13, berfungsi untuk menampilan daftar bahan mentah dan bahan olahan yang tersedia di inventori. Prosedur yang dibutuhkan yaitu procedure lihatInventori, urutString.
14. F14, berfungsi untuk menampilkan daftar resep yang tersedia. Prosedur yang dibutuhkan yaitu procedure lihatResep, urutString.
15. F15, berfungsi untuk melakukan pencarian resep. Prosedur yang dibutuhkan yaitu procedure cariResep.
16. F16, berfungsi untuk melakukan penambahan resep. Prosedur yang dibutuhkan yaitu procedure TambahResepBaru.
17. F17, berfungsi untuk melakukan upgrade inventori. Prosedur yang dibutuhkan yaitu procedure upgradeInventori.
18. B4, berfungsi untuk melakukan restock secara berkala. Prosedur yang dibutuhkan yaitu procedure Restock

**7. SPESIFIKASI TIAP FITUR DALAM UNIT**

Tabel 3. Tabel Spesifikasi Tiap Fitur Dalam Unit

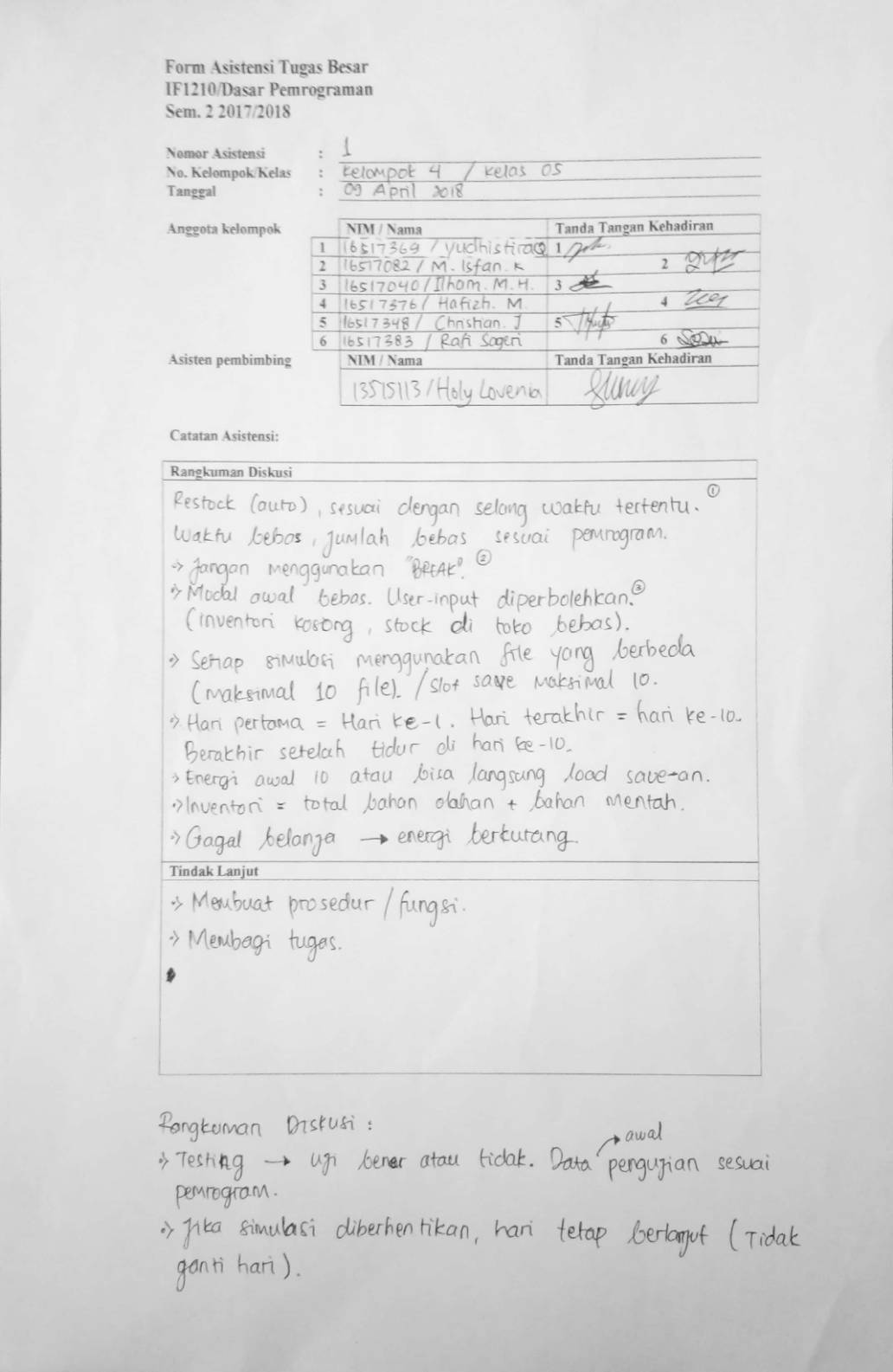
|  |  |
| --- | --- |
| **Unit** | **Fitur dan Spesifikasi** |
| typeuniverse | - |
| F1 | procedure LoadBahanMentah (input/output bahanMentah : tabmentah)  procedure LoadBahanOlahan (input/output bahanOlahan : tabolahan)  procedure LoadInventoriMentah (input tabmentah : tabInventoriM)  procedure LoadInventoriOlahan (input tabolahan : tabInventoriO)  procedure LoadResep (input/output bahanResep : tabResep; daftarmentah : tabmentah; daftar olahan : tabolahan)  procedure LoadInventoriFileSimulasi (input nfile : integer; input nama : string; input/output arr : listSimulasi) |
| F2 | procedure SaveFileSimulasi (input nfile : integer; input nama : string; input/output arr : listSimulasi)  procedure SavedInventoriMentah (input nfile : integer; input nama : string; input/output arr : listInventoriM)  procedure SaveInventoriOlahan (input nfile : integer; input nama : string; input/output arr : listInventoriO)  procedure SaveResep (input nfile : integer; input nama : string; input/output resep: tabResep)  procedure SaveBahanOlahan (input nfile : integer; input nama : string; input/output olahan : tabOlahan) |
| CallSim (F3-F4 di dalam) | procedure Call0 (input/output simN : integer; input/output eoSim : boolean; input/output eoProg : boolean; input/output DftrSim : listSimulasi; input/output DftrResep : tabResep; input/output DftrMentah : tabMentah; input/output DftrOlahan : tabOlahan; input/output invMentah : listInventoriM; input/output invOlahan : listInventoriO)  procedure Call1 (input/output simN : integer; input/output nMakan : integer; input/output nIstirahat : integer; input/output eoSim : boolean; input/output eoProg : boolean; input/output bTidur : boolean; input/output DftrSim : listSimulasi; input/output DftrResep : tabResep; input/output DftrMentah : tabMentah; input/output DftrOlahan : tabOlahan; input/output invMentah : listInventoriM; input/output invOlahan : listInventoriO; input/output TglHariIni : tanggal) |
| F5 | procedure beliBahan ( input a : tabMentah, input/output b : tabInventoriM, input/output c : tabInventoriO, input/output u : integer, input/output e : integer, input/output imax: integer, input/output tglnow : tanggal) |
| F6 | procedure olahBahan (input o : tabOlahan, input/output a : tabInventoriM, input/output b : tabInventoriO, input/output e {energi} : integer, input/output tglnow : tanggal) |
| F7 | procedure CariOlahan (input/output JualO : String, input BahanOlahan : ListInventoriO, input/output FoundOlahan : Boolean, input Nsim : integer, input/output i : integer)  procedure KurangiOlahan (input JumlahJualO : integer, input BahanOlahan : ListInventoriO, input/output CountOlahan : Boolean, input Nsim : integer, input i : integer)  procedure KurangiEnergi (input/output Energi : integer)  procedure TambahPendapatanOlahan (input/output Pendapatan : integer, input JumlahJualO : integer, input BahanOlahan : ListInventoriO, input Nsim : integer, input i : integer)  procedure JualOlahan (input/output BahanOlahan : ListInventoriO, input/output Energi : integer, input/output Pendapatan : integer, input Nsim : integer) |
| F8 | function cariBahanMentah (bahan : string; daftarBahan : TabInventoriM) -> integer  function cariBahanOlahan (bahan : string; daftarBahan : TabInventoriO) -> integer  function cariResep (resep : string; daftarResep : tabResep) -> integer  function isBahanMentahValid (bahan : string; n : integer; daftarBahan : TabInventoriM) -> boolean  function isBahanOlahanValid (bahan : string; n : integer; daftarBahan : TabInventoriO) -> boolean  procedure jualResep (input/output BahanMentah : ListInventoriM, input/output BahanOlahan : ListInventoriO, input ResepJadi : tabresep, input/output Energi : integer, input/output Pendapatan : integer, input Nsim : integer) |
| F9 | procedure makan (input/output daftarsimulasi: listSimulasi, input n : integer, input/output nmakan: integer) |
| F10 | procedure istirahat (input/output daftarsimulasi: listSimulasi; input n : integer, input/output nistirahat: integer) |
| F11 | function cariKadaluarsa (bahan : string; daftarMentah : tabmentah) -> integer  function isKadaluarsaM (i : integer; daftarmentah : tabMentah; inventoriM : listInventoriM; sim : listSimulasi; nfile : integer) -> boolean  function isKadaluarsaO (i : integer; daftarolahan : tabOlahan; inventoriO : listInventoriO; sim : listSimulasi; nfile : integer)-> boolean;  procedure buangKadaluarsa (input sim : listSimulasi; input InventoriM : listInventoriM; input mentah : tabmentah; input InventoriO : listInventoriO; input olahan : tabolahan; input nfile : integer);  procedure tidur (output simulasi : listSimulasi; output BTidur : boolean; input nfile : integer); |
| F12 | procedure lihatStatistik (input/output nfile: integer, input/output arr: listSimulasi) |
| F13F14 | procedure lihatInventoriM (input/output x : tabInventoriM)  procedure lihatInventoriO (input/output x : tabInventoriO)  procedure lihatInventori (input/output x1 : tabInventoriM, input/output x2 : tabInventoriO)  procedure lihatResep (input/output x : tabResep)  procedure urutString (input/output s1, s2 : string)  function isUrut (s1, s2 : string) -> boolean |
| F15 | procedure cariResep (input/output a : tabResep) |
| F16 | procedure InputDataResep (input/output NamaResep : string, input/output MentahResep : Mresep, input/output OlahanResep : Oresep, input/output HargaResep : integer)  function IsResepAda (ResepJadi : tabresep, NamaResep : string) -> boolean  function IsBahanPembentukCukup (MentahResep : Mresep; OlahanResep : Oresep) -> boolean  function IsBahanMentahAda (MentahResep : Mresep, BahanMentah : TabMentah) -> boolean  function IsBahanOlahanAda (OlahanResep : Oresep, BahanOlahan : TabOlahan) -> boolean  function IsHargaValid (HargaResep : integer, MentahResep : Mresep, OlahanResep : Oresep, BahanMentah : TabMentah, BahanOlahan : TabOlahan) -> boolean  procedure KurangiEnergi (input/output Energi : integer)  procedure InputResepBaru (input NamaResep : string, input MentahResep : Mresep, input OlahanResep : Oresep, input HargaResep : integer, input/output ResepJadi : tabresep)  procedure TambahResepBaru (input BahanMentah : TabMentah, input BahanOlahan : TabOlahan, input/output ResepJadi : tabresep, input/output Energi : integer) |
| F17 | procedure upgradeInventori (input/output imax : integer) |
| B4 | procedure Restock (input/output a : tabInventoriM; input/output b : tabInventoriO; input/output imax : integer; input s : listSimulasi; input n : integer) |

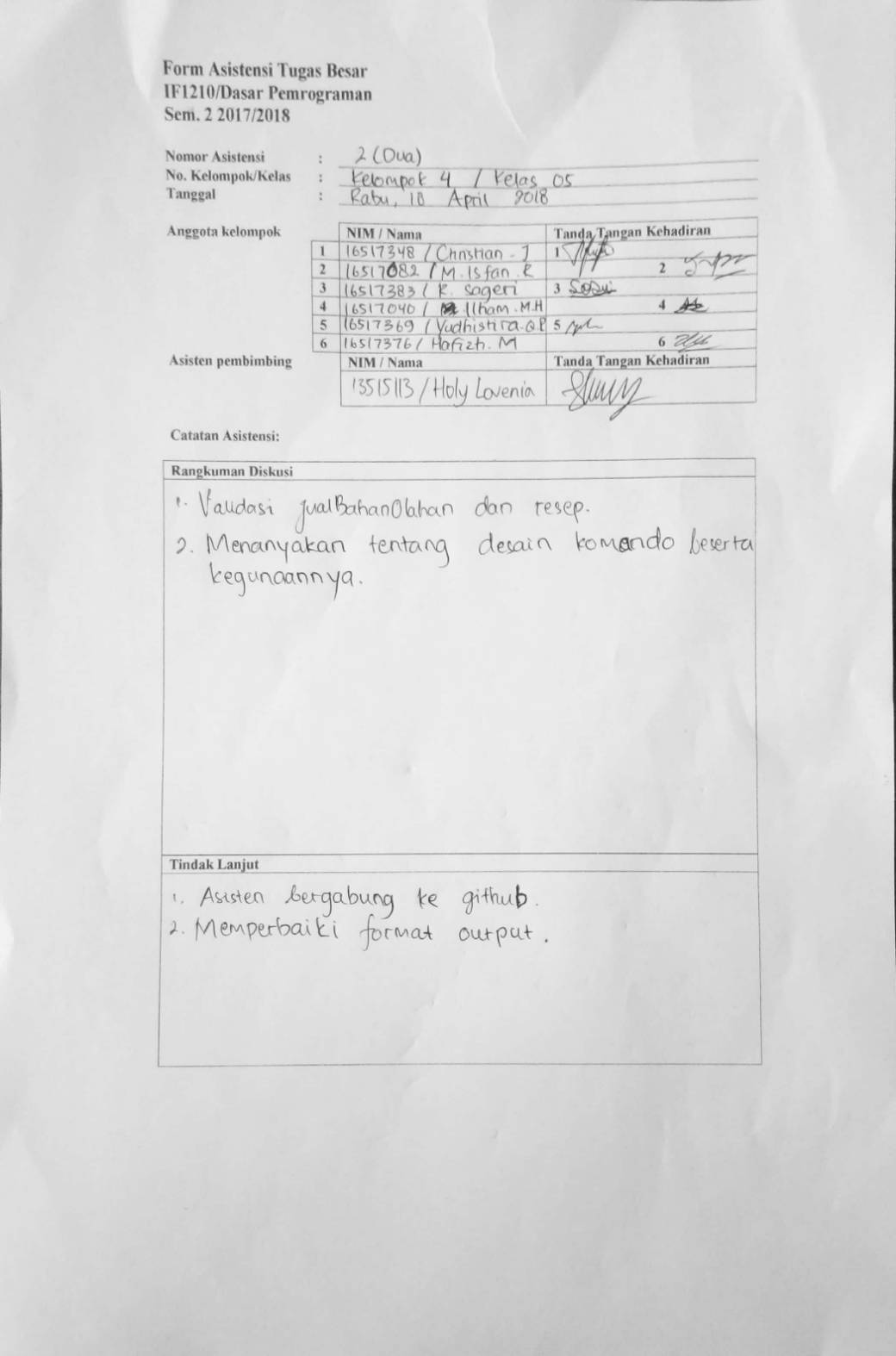
**8. PENGUJIAN**

Tabel 4. Tabel Pengujian Program

|  |  |  |  |
| --- | --- | --- | --- |
| **Fitur** | **Data awal** | **Hasil Pengujian** | **Evaluasi** |
| **F1** | Semua file eksternal | Membaca data dari file eksternal dan menuliskannya ke dalam array type bentukan | Semua file terbaca |
| **F2** | Semua file eksternal | Keluar dari program, menyimpan semua prubahan data ke file eksernal | Semua file tersimpan |
| **F3** | File Simulasi | Memulai suatu simulai nomor tertentu dari daftar simulasi yang ad | Simulasi dimulai |
| **F4** | File Simulasi | Menghentikan simulasi tertentu yang sedang berjalan dan menampilkan statistik simulasi. | Simulasi berhenti |
| **F5** | File Bahan Mentah  File Inventori Bahan Mentah  File Inventori Bahan Olahan | Membeli bahan mentah, melakukan validasi apakah barang dijual lalu memasukannya ke inventori | Membeli bahan, memasukkan ke inventori, lalu dimasukkan ke file eksternal |
| **F6** | File Bahan Olahan  File Inventori Bahan Mentah  File Inventori Bahan Olahan | Mengubah bahan mentah menjadi olahan dan memasukannya ke inventori olahan | Mengolah bahan, memasukkan ke inventori, lalu dimasukkan ke file eksternal |
| **F7** | File Inventori Bahan Olahan  File Bahan Olahan | Menjual bahan olahan. Penjualan bahan olahan mengurangi inventori, menambah pemasukan, dan mengurangi energi. | Menjual bahan olahan, mengurangi inventori |
| **F8** | File Resep  File Bahan Mentah  File Bahan Olahan  File Inventori Bahan Mentah  File Inventori Bahan Olahan | Mengolah dan menjual hidangan berdasarkan resep. Hanya bisa dibuat jika bahan mentah dan olahan tersedia di Inventori. | Mengolah hidangan, mengurangi inventori, menambah pemasukan |
| **F9** | File Simulasi | Melakukan aksi makan dan menambah 3 buah energi | Energi bertambah |
| **F10** | File Simulasi | Melakukan aksi istirahat dan menambah 1 buah energi | Energi bertambah |
| **F11** | File Simulasi | Melakukan aksi tidur dan menyebabkan energi bertambah sampai menjadi 10 dan hari berganti. | Energi bertambah, hari berganti, inventori kadaluarsa hilang |
| **F12** | File Simulasi | Menampilan statistik simulasi | Menampilkan statistik |
| **F13** | File Inventori Bahan Mentah  File Inventori Bahan Olahan  File Resep | Menampilkan data daftar bahan mentah dan bahan olahan yang tersedia di inventori | Menampilkan data daftar file eksternal |
| **F14** | File Inventori Bahan Mentah  File Inventori Bahan Olahan  File Resep | Melihat daftar resep yang tersedia, termasuk semua daftar bahan penyusunnya | Menampilkan daftar resep |
| **F15** | File Resep | Melakukan pencarian resep berdasarkan nama resep yang tersedia | Menampilkan resep yang dicari |
| **F16** | File Mentah  File Olahan  File Resep | Melakukan penambahan resep dengan minimum | Resep bertambah |
| **F17** | - | Mengupgrade inventori maksimal | Inventori bertambah |

**LAMPIRAN**

****

****