

LAPORAN TUGAS BESAR

IF1210 DASAR PEMROGRAMAN K08-B



Kelas : 08

Kelompok : B

Dosen Pengampu : Muhammad Romadhon Al-Ghazali, S.T., M.A.B.

Disusun oleh :

- Abu Dzar Al-Ghfari (16523058)
- Muhammad Refino Ramadhan (19623028)
- Darren Mansyl (19623168)
- Favian Rafi Laftiyanto (19623238)
- Mahesa Satria Prayata (19623278)

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2024**

Pernyataan Kelompok

“Saya menyatakan bahwa saya mengerjakan tugas besar ini dengan sejujur-jujurnya, tanpa menggunakan cara yang tidak dibenarkan. Apabila di kemudian hari diketahui saya mengerjakan tugas besar ini dengan cara yang tidak jujur, saya bersedia mendapatkan konsekuensinya, yaitu mendapatkan nilai E pada mata kuliah IF1210 Dasar Pemrograman Semester 2 2023/2024.”

Abu Dzar Al-Ghifari (16523058)

Muhammad Refino Ramadhan (19623028)

Darren Mansyl (19623168)

Favian Rafi Laftiyanto (19623238)

Mahesa Satria Prayata (19623278)

Daftar Isi

Pernyataan Kelompok.....	1
Daftar Isi.....	2
Daftar Tabel.....	3
Daftar Gambar.....	4
Deskripsi Persoalan.....	8
Daftar Pembagian Kerja Anggota Kelompok.....	11
Progress Hasil Rancangan.....	16
Desain Command.....	18
Spesifikasi, Dekomposisi Algoritmik dan Fungsional Program, dan Kamus Data.....	21
F00 - Random Number Generator.....	21
F01 - Register.....	22
F02 - Login.....	24
F03 - Logout.....	25
F04 - Menu & Help.....	26
F06 - Potion.....	27
F07 - Inventory.....	30
F08 - Battle.....	33
F09 - Arena.....	35
F10 - Shop.....	38
F11 - Laboratory.....	42
F12 - Shop Management.....	45
F13 - Monster Management.....	52
F14 - Load.....	60
F15 - Save.....	61
F16 - Exit.....	62
B03 - Monster Ball.....	62
B04 - Jackpot.....	64
Pengujian Program.....	67
F01 Register.....	67
F02 Login.....	68
F03 Logout.....	69
F04 Menu and Help.....	70
F07 Inventory.....	71
F08 Battle.....	73
F09 Arena.....	79
F10 Shop and Currency.....	83
F11 Laboratory.....	85
F12 Shop Management.....	87
F13 Monster Management.....	94
F14 Load.....	97
F15 Save.....	97
F16 Exit.....	98
B03 Monster Ball.....	98
B04 Jackpot.....	99
Lampiran.....	102

Daftar Tabel

Tabel 1 - Pembagian Kerja Anggota Kelompok dalam Spesifikasi Program.....	15
Tabel 2 - Pembagian Kerja Anggota Kelompok dalam Spesifikasi Bonus.....	15
Tabel 3 - Pembagian Kerja Laporan Anggota Kelompok.....	16
Tabel 4 - Progress Hasil Rancangan.....	18

Daftar Gambar

Gambar 1.1 - Register: Validasi Username (1).....	68
Gambar 1.2 - Register: Validasi Username (2).....	68
Gambar 1.3 - Register: Validasi Username Berhasil.....	68
Gambar 2.1 - Login: Username tidak ditemukan.....	68
Gambar 2.2 - Login: Password Salah.....	69
Gambar 2.3 - Login: Sukses.....	69
Gambar 3.1 - Logout: Tidak jadi logout.....	69
Gambar 3.2 - Logout: Validasi, input buka y/n.....	69
Gambar 3.3 - Logout: Berhasil logout.....	69
Gambar 4.1 - Menu dan Help: Tampilan sebelum login.....	70
Gambar 4.2 - Menu dan Help: Saat login sebagai agent.....	70
Gambar 4.3 - Menu dan Help: Saat login sebagai admin.....	71
Gambar 7.1 - Inventory: Melihat monster yang dimiliki.....	71
Gambar 7.2 - Inventory: Melihat item yang dimiliki.....	72
Gambar 7.3 - Inventory: Menampilkan monster.....	72
Gambar 7.4 - Inventory: Menampilkan detail item.....	72
Gambar 7.5 - Inventory: Keluar dari inventory.....	72
Gambar 7.6 - Inventory: Validasi input (1), input selain monster, item, dan back.....	73
Gambar 7.7 - Inventory: Validasi input(2), id tidak ada dalam list monster.....	73
Gambar 7.8 - Inventory: Validasi input (3), id tidak ada dalam list item.....	73
Gambar 8.1 - Battle: Tampilan awal battle.....	74
Gambar 8.2 - Battle: Tampilan memilih monster untuk battle.....	74
Gambar 8.3 - Battle: Monster yang dipilih tidak ada.....	75
Gambar 8.4 - Battle: Tampilan saat turn agen.....	75
Gambar 8.5 - Battle: Tampilan saat agent attack dan musuh menyerang balik serta hitungannya.	76
Gambar 8.6 - Battle: Tampilan saat agent memilih potion untuk diminum.....	76
Gambar 8.7 - Battle: Tampilan saat meminum potion strength.....	77
Gambar 8.8 - Battle: Tampilan saat agent meminum potion resilience.....	77
Gambar 8.9 - Battle: Tampilan saat agent meminum healing potion.....	77
Gambar 8.10 - Battle: Tampilan saat agent tidak memiliki potion.....	78
Gambar 8.11 - Battle: Tampilan saat potion sudah pernah diminum.....	78
Gambar 8.12 - Battle: Validasi Input (1), saat potion tidak ada dalam opsi.....	78
Gambar 8.13 - Battle: Tampilan saat agent memilih opsi “QUIT”.....	79
Gambar 8.14 - Battle: Tampilan saat agent menang.....	79
Gambar 8.15 - Battle: Tampilan saat agent kalah.....	79
Gambar 8.16 - Battle: Validasi input (2), saat opsi tidak ada pada opsi action.....	79
Gambar 9.1 - Arena: Tampilan awal arena.....	80
Gambar 9.2 - Arena: Tampilan saat agent memilih monster dari inventory.....	80
Gambar 9.3 - Arena: Validasi input (1), saat monster yang dipilih tidak tersedia.....	80

Gambar 9.4 - Arena: Tampilan setiap stage.....	81
Gambar 9.5 - Arena: Tampilan kondisi kemanangan setiap stage.....	81
Gambar 9.6 - Arena: Tampilan berhasil mencapai stage akhir.....	82
Gambar 9.7 - Arena: Tampilan kondisi kekalahan tiap stage.....	82
Gambat 9.8 - Arena: Tampilan saat memilih opsi quit di setiap stage.....	82
Gambar 9.9 - Arena: Tampilan statistik akhir sesi arena.....	83
Gambar 10.1 - Shop and C. : Tampilan awal.....	83
Gambar 10.2 - Shop and C. : Tampilan saat pilih lihat monster.....	83
Gambar 10.3 - Shop and C. : Tampilan saat pilih lihat item.....	84
Gambar 10.4 - Shop and C. : Tampilan saat berhasil membeli mosnter.....	84
Gambar 10.5 - Shop and C. : Tampilan saat mosnter yang dibeli sudah dimiliki.....	84
Gambar 10.6 - Shop and C. : Tampilan saat id monster tidak ada di shop.....	84
Gambar 10.7 - Shop and C. : Tampilan saat agent kekurangan coin.....	84
Gambar 10.8 - Shop and C. : Tampilan saat berhasil membeli item.....	85
Gambar 10.9 - Shop and C. : Tampilan saat stok tidak cukup.....	85
Gambar 10.10 - Shop and C. : Tampilan saat stok habis.....	85
Gambar 10.11 - Shop and C. : Tampilan saat id item tidak ada.....	85
Gambar 10.12 - Shop and C. : Tampilan saat player memilih untuk keluar shop.....	85
Gambar 11.1 - Lab: Tampilan awal.....	86
Gambar 11.2 - Lab: Tampilan saat id monster yang ingin diupgrade tidak tersedia.....	86
Gambar 11.3 - Lab: Tampilan saat sukses upgrade monster.....	86
Gambar 11.4 - Lab: Validasi (1), Tampilan saat input bukan y/n.....	87
Gambar 11.5 - Lab: Validasi (2), Tampilan saat level upgrade sudah tercapai.....	87
Gambar 11.6 - Lab: Validasi (3), level yang dituju lebih dari 5.....	87
Gambar 11.7 - Lab: Validasi (4), input level bukan integer.....	87
Gambar 11.8 - Lab: Coin tidak cukup.....	87
Gambar 11.9 - Lab: Tampilan keluar dari laboratory.....	87
Gambar 12.1 - Shop M. : Tampilan awal.....	88
Gambar 12.2 - Shop M. : Tampilan lihat monster.....	88
Gambar 12.3 - Shop M. : Tampilan lihat item.....	88
Gambar 12.4 - Shop M. : Tampilan tambah monster.....	88
Gambar 12.5 - Shop M. : Tampilan saat monster berhasil ditambah.....	89
Gambar 12.6 - Shop M. : Tampilan saat berhasil menambah monster.....	89
Gambar 12.7 - Shop M. : Validasi (1), Tampilan saat id monster yang ingin ditambah tidak ada pada file monster.....	89
Gambar 12.8 - Shop M. : Validasi (2), saat stok awal dan harga bukan integer.....	89
Gambar 12.9 - Shop M. : Tampilan saat ingin menambah item.....	89
Gambar 12.10 - Shop M. : Tampilan saat berhasil menambahkan item.....	89
Gambar 12.11 - Shop M. : Tampilan saat item berhasil ditambah ke shop.....	90
Gambar 12.12 - Shop M. : Validasi (3), saat id item diluar jangkauan.....	90
Gambar 12.13 - Shop M. : Validasi (4), saat id item bukan integer.....	90

Gambar 12.14 - Shop M. : Berhasil mengubah stok dan harga monster.....	90
Gambar 12.15 - Shop M. : Hanya mengubah stock.....	90
Gambar 12.16 - Shop M. : Hanya mengubah harga.....	90
Gambar 12.17 - Shop M. : Id monster yang ingin diubah tidak ada.....	91
Gambar 12.18 - Shop M. : Validasi (5), stok dan harga baru bukan integer.....	91
Gambar 12.19 - Shop M. : Berhasil mengubah item di shop.....	91
Gambar 12.20 - Shop M. : mengubah stok saja.....	91
Gambar 12.21 - Shop M. : mengubah harga saja.....	92
Gambar 12.22 - Validasi (6), id item yang ingin diubah tidak tersedia.....	92
Gambar 12.23 - Shop M. : Validasi (7), stok atau harga bukan integer.....	92
Gambar 12.24 - Shop M. : Berhasil menghapus monster dari shop.....	92
Gambar 12.25 - Shop M. : Penghapusan dibatalkan.....	92
Gambar 12.26 - Shop M. : Validasi (7) input bukan y/n.....	93
Gambar 12.27 - Shop M. : Validasi (8), id monster yang ingin dihapus tidak ada.....	93
Gambar 12.28 - Shop M. : Berhasil hapus item.....	93
Gambar 12.29 - Shop M. : Penghapusan item dibatalkan.....	93
Gambar 12.30 - Shop M. : Validasi (9), id item yang ingin dihapus tidak ada.....	93
Gambar 12.31 - Shop M. : Keluar dari shop management.....	94
Gamber 12.32 - Shop M. : Validasi (10), aksi tidak ada.....	94
Gambar 13.1 - Monster M. : Tampilan awal.....	94
Gamber 13.2 - Monster. M : Memilih aksi melihat tabel.....	94
Gamber 13.3 - Monster M. : Balik tampilan awal.....	95
Gamber 13.4 - Monster M. : Memilih aksi menambah monster dan berhasil.....	95
Gamber 13.5 - Monster M. : Nama monster sudah ada pada list.....	95
Gamber 13.6 - Monster M. : Validasi (1), saat input attack negative.....	95
Gambar 13.7 - Monster M. : Validasi (2), input hp negative.....	95
Gambar 13.8 - Monster M. : Validasi (3), defense diluar jangkauan.....	96
Gambar 13.9 - Monster M. : Validasi (4), input bukan integer.....	96
Gambar 13.10 - Monster M. : Monster tidak jadi ditambahkan.....	96
Gamber 13.11 - Monster M. : Validasi (5), input bukan y/n.....	96
Gambar 13.12 - Monster M. Validasi (6), input aksi tidak sesuai.....	96
Gambar 13.13 - Monster M. : Keluar dari monster management.....	97
Gambar 14.1 - Load: tidak ada data.....	97
Gambar 14.2 - Load: folder tidak ditemukan.....	97
Gambar 14.3 - Load: Berhasil.....	97
Gambar 15.1 - Save: Membuat folder baru.....	97
Gambar 15.2 - Save: Folder sudah ada data.....	98
Gambar 16.1 - Exit: exit program tanpa save.....	98
Gambar 16.2 - Exit: Validasi (1), input bukan y/n.....	98
Gambar 16.3 - Exit: exit program dengan save.....	98
Gamber B03.1 - Monster ball: Berhasil menangkap monster.....	99

Gambar B03.2 - Monster ball: gagal menangkap monster.....	99
Gambar B03.3 - Monster ball: tidak memiliki monster ball dalam inventory.....	99
Gambar B04.1 - Jackpot: Tampilan awal.....	100
Gambar B04.2 - Jackpot: Kurang beruntung.....	100
Gambar B04.3 - Jackpot : Beruntung!!.....	100
Gamber B04.4 - Jackpot: Player tidak yakin.....	100
Gambar B04.5 - Jackpot: Player tidak ingin melanjutkan jackpot.....	101
Gambar B04.6 - Jackpot: Validasi, input bukan y/n.....	101
Gambar Lampiran 1 - Asistensi 1.....	102
Gambar Lampiran 2 - Asistensi 2.....	102
Gambar Lampiran 4 - MoM Asistensi 1.....	103
Gambar Lampiran 4 - MoM Asistensi 2.....	104

Deskripsi Persoalan

Tugas besar ini meminta kita untuk membuat program sebuah permainan bertipe *turn-based* berdasarkan cerita Purry the Platypus, atau dikenal juga dengan Agent P, yang ingin menyelamatkan kota Danville dari Dr. Asep Spakbor. Karena monster-monster yang dimiliki Dr. Asep Spakbor terlalu kuat, maka Agent P membutuhkan bantuan Agent O.W.C.A. (para player) untuk mencari dan melatih para monster yang kuat di hutan terpencil yang diyakini menjadi tempat tinggal banyak monster-monster kuat.

Sesuai dengan cerita ini, program dapat diurai menjadi fungsional-fungsional sebagai berikut,

1. Fungsi-fungsi penyimpanan akun dan data dan fungsi RNG,
 - a. F00 - Random Number Generator, untuk menghasilkan bilangan acak menggunakan algoritma Linear Congruential Generator (LCG).
 - b. F01 - Register, untuk memasukkan *username* dan *password* dari akun yang akan digunakan untuk memulai permainan. *Username* hanya dapat mengandung alfabet A-Z, a-z, underscore “_”, strip “-”, dan angka 0-9. Bila *username* belum terdaftar, maka akun diharuskan untuk memilih 1 monster starter, 0 O.W.C.A. coin, dan terdaftar sebagai *agent*.
 - c. F02 - Login, untuk *login* ke akun yang sudah didaftar sebelumnya.
 - d. F03 - Logout, untuk keluar dari akun yang sedang digunakan.
 - e. F04 - Menu dan Help, untuk mengingatkan dan menuntun pengguna dan mengingatkan untuk melakukan validasi input dalam bentuk *footnote*.
 - f. F05 - Monster, file yang menyimpan data monster dengan format “monster.csv”. File tersebut berisi statistik monster berupa,
 - i. Type, nama dari monster yang bersifat unik.
 - ii. ATK Power, Kekuatan serangan setiap monster. Ketika monster melakukan serangan, nilai kekuatan serangannya diambil secara acak dengan rentang $\pm 30\%$ dari ATK Power. Contoh untuk ATK Power = 500, rentang nilai serangannya adalah 350 - 850.

- iii. DEF Power, Kekuatan pertahanan setiap monster dari serangan. DEF digunakan sebagai faktor pengali untuk mengurangi serangan musuh. Rentang nilainya adalah 0 - 50. Contoh untuk DEF Power = 50, maka monster dapat menurunkan serangan musuh sebesar 50%.
 - iv. HP, jumlah darah yang dimiliki para monster
- g. F06 - Potion, untuk membantu player dalam *battle*. Berikut adalah jenis-jenis potion serta efeknya,
 - i. Healing Potion, Mengisi darah sebanyak 25% dari Base HP. Pastikan HP tidak melebihi dari maksimal HP.
 - ii. Resilience Potion, Meningkatkan DEF Power sebanyak 5% dari DEF Power.
 - iii. Strength Potion, Meningkatkan ATK Power sebanyak 5% dari ATK Power.

ATK Power, DEF Power, dan HP mengikuti level Monster dari player, Masing-masing potion hanya dapat digunakan sekali dalam 1 battle dan efeknya hanya berlaku hingga battle itu selesai.

- h. F14 - Load, untuk masuk kembali ke akun yang sudah terdaftar.
 - i. F15 - Save, untuk menyimpan progress akun ke dalam folder yang ditentukan.
 - j. F16 - Exit, untuk keluar dari program. Ada juga fitur untuk save terlebih dahulu sebelum melakukan exit.
2. Fungsi-fungsi yang dapat diakses oleh *agent*,
- a. F07 - Inventory, berupa file untuk menyimpan *item* yang dimiliki oleh player seperti potion, monster, dan monster ball. Terdapat format file yang berbeda untuk *item* (item_inventory.csv) dan monster (monster_inventory.csv).
- b. F08 - Battle, fitur utama dalam game ini untuk melawan monster-monster liar yang dipilih secara random. berikut adalah mekanismenya,
 - i. Muncul monster (musuh) secara random (RNG) database Monster; Agent memilih monster (agent) yang ingin dipertarungkan.
 - ii. Setiap putaran Agent memiliki pilihan untuk “Attack”, “Use Potion”, atau “Quit”; Monster (musuh) hanya bisa “Attack”.

- iii. Kondisi kemenangan adalah saat nyawa monster (musuh) habis; Agent mendapatkan OC (OC yang diterima random (RNG), misal 5-30).
 - iv. Kondisi kekalahan adalah saat nyawa monster (agent) habis; Agent keluar dari pertempuran, nyawa monster yang dipertarungkan kembali penuh (tidak hilang dari inventory).
 - c. F09 - Arena, fitur latihan agar player dapat terbiasa dengan mekanisme *battle*. Mekanismenya sama seperti fitur *battle*. Player akan mendapatkan hadiah setara dengan jumlah stage yang berhasil dilewati. (OC = O.W.C.A. Coin)
 - i. Stage 1: 50 OC
 - ii. Stage 2: 100 OC
 - iii. Stage 3: 150 OC
 - iv. Stage 4: 200 OC
 - v. stage 5: 250 OC
 - d. F10 - Shop & Currency, tempat player membeli monster dan potion tambahan. Terdapat format file khusus item (item_shop.csv) dan monster (monster_shop.csv).
 - e. F11 - Laboratory, fitur untuk meng-*upgrade* level monster yang dimiliki oleh player. Untuk melakukan *upgrade* diperlukan OC. Berikut adalah harganya,
 - i. Level 1 - 2: 100,
 - ii. Level 2 - 3: 200,
 - iii. Level 3 - 4: 400,
 - iv. Level 4 - 5: 700
3. Fungsi-fungsi yang dapat diakses oleh admin,
- a. F12 - Shop Management, untuk mengubah dan menghapus item dalam shop
 - b. F13 - Monster Management, untuk menampilkan seluruh monster dalam program dan menambah monster ke program.

Daftar Pembagian Kerja Anggota Kelompok

1. Pembagian Kerja Program

Fitur	Implementasi	NIM Designer	NIM Coder	NIM Tester
F00 - Random Number Generator	function LCG function GenerateNumber function RNG	19623278	19623278	16523058 19623028 19623168 19623238 19623278
F01 - Register	function register	19623028	19623028	16523058 19623028 19623168 19623238 19623278
F02 - Login	function login	19623028	19623028	16523058 19623028 19623168 19623238 19623278
F03 - Logout	procedure logout	19623168	19623168	16523058 19623028 19623168 19623238 19623278
F04 - Menu dan Help	function help	19623028	19623028	16523058 19623028 19623168 19623238 19623278
F05 - Monster	file monster.csv	19623028 19623278	19623278	16523058 19623028

				19623168 19623238 19623278
F06 - Potion	function jumlahPot function indexPot function potion	19623168	19623168	16523058 19623028 19623168 19623238 19623278
F07 - Inventory	function inventory	19623168	19623168	16523058 19623028 19623168 19623238 19623278
F08 - Battle	procedure battleart function battlerule function choose procedure opening function battle	16523058	16523058	16523058 19623028 19623168 19623238 19623278
F09 - Arena	procedure arenaart function arenarule function arena	16523058	16523058	16523058 19623028 19623168 19623238 19623278
F10 - Shop &	procedure	19623238	19623238	16523058 19623028

Currency	shopInterface function shopOpen function			19623168 19623238 19623278
F11 - Laboratory	function laboratory	19623168	19623168	16523058 19623028 19623168 19623238 19623278
F12 - Shop Management	function shopmanageme nt	19623238	19623238	16523058 19623028 19623168 19623238 19623278
F13 - Monster Management	function cekint function checkType function checkAttack function checkDefense function checkHp function newType function newAttack function newDefense function newHp	19623278	19623278	16523058 19623028 19623168 19623238 19623278

	procedure checkUserInput procedure validUserInput procedure newMonster function monsterManagement			
F14 - Load	function load	16523058	16523058	16523058 19623028 19623168 19623238 19623278
F15 - Save	function save function write	16523058	16523058	16523058 19623028 19623168 19623238 19623278
F16 - Exit	procedure exit	19623238	19623238	16523058 19623028 19623168 19623238 19623278

Tabel 1 - Pembagian Kerja Anggota Kelompok dalam Spesifikasi Program

Spesifikasi Bonus

Fitur	Implementasi	NIM Designer	NIM Coder	NIM Tester
B01- Git Best Practice	Telah diimplementasi	16523058 19623028	16523058 19623028	16523058 19623028

	kan pada GitHub	19623168 19623238 19623278	19623168 19623238 19623278	19623168 19623238 19623278
B02 - Typing	-	-	-	-
B03 - Monster Ball	function monsterball	16523058	16523058	16523058 19623028 19623168 19623238 19623278
B04 - JACKPOT!!!	function prob function gacha function jackpot	16523058 19623168	16523058 19623168	16523058 19623028 19623168 19623238 19623278
B05 - Peta Kota Danville	-	-	-	-

Tabel 2 - Pembagian Kerja Anggota Kelompok dalam Spesifikasi Bonus

2. Pembagian Kerja Proposal

Bagian	NIM
Halaman Cover	19623168
Daftar Isi	19623278
Daftar Tabel	19623238
Daftar Gambar	19623238 19623278
Deskripsi Persoalan	19623278
Daftar Pembagian Tugas	19623238

Proses Hasil Rancangan	19623238
Desain Command	19623238
Desain Dekomposisi Algoritmik dan Fungsional Program, Kamus data, Spesifikasi Fungsi/Prosedur	<ul style="list-style-type: none"> - 16523058 - 19623028 - 19623168 (F06, F07, F11, B04) - 19623238 (F04,F10,F11,F12,F14,F15,F16) - 19623278 (F00, F09, F13, B03)
Hasil Pengujian Program	16523058
Lampiran	19623028

Tabel 3 - Pembagian Kerja Laporan Anggota Kelompok

Progress Hasil Rancangan

Fitur	Desain	Implementasi	Testing
F00 - Random Number Generator	V	V	V
F01 - Register	V	V	V
F02 - Login	V	V	V
F03 - Logout	V	V	V
F04 - Menu dan Help	V	V	V
F05 - Monster	V	V	V
F06 - Potion	V	V	V
F07 - Inventory	V	V	V
F08 - Battle	V	V	V

F09 - Arena	V	V	V
F10 - Shop & Currency	V	V	V
F11 - Laboratory	V	V	V
F12 - Shop Management	V	V	V
F13 - Monster Management	V	V	V
F14 - Load	V	V	V
F15 - Save	V	V	V
F16 - Exit	V	V	V
B01 - Git Best Practice	V	V	V
B02 - Typing	-	-	-
B03 - Monster Ball	V	V	V
B04 - JACKPOT!!!	V	V	V
B05 - Peta Kota Danville	-	-	-

Tabel 4 - Progress Hasil Rancangan

Desain *Command*

- —> Login
function login()
{ I.S. Melakukan login dengan cara menginput username dan password }

- —> Logout

procedure logout()

{ I.S. Menerima input berupa konfirmasi apakah ingin logout, dengan catatan status user harus dalam keadaan telah login }

{ F.S. Apabila input berupa ‘y’, maka user akan logout dan dapat melakukan login kembali atau register. Apabila input ‘n’, maka user akan membatalkan logout dan dapat melakukan command lain sebagai user agent }
- —> Register

function register()

{ I.S. Menerima input berupa username dan password, kemudian meminta input user untuk memilih monster pertama dari 3 pilihan yang telah tersedia. }

{ F.S. Apabila username telah terpakai/ada di file csv user, maka akan meminta input username lagi. Jika username, password, dan pilihan monster telah valid, maka monster yang dipilih akan disimpan dan disimpan pula data inventory dari user baru, serta menampilkan pesan selamat datang }
- —> Help

procedure help()

{ I.S. Mengecek kondisi status login user dan role user sebelum menampilkan apa yang dapat dilakukan pada program }

{ F.S. Apabila user belum login, maka akan menampilkan pilihan yaitu Login, Register, Save, dan Exit. Apabila telah login dan user merupakan Agent, maka ditampilkan pilihan Battle, Arena, Shop, Inventory, Laboratory, Jackpot, dan Logout. Apabila telah login dan user merupakan Admin, maka ditampilkan pilihan Shop (management), Monster (management), dan Logout }
- —> Battle

function battle()

{ I.S. Menerima input berupa monster yang akan digunakan pada battle. }

{ F.S. Melakukan proses Battle, mulai dari menampilkan Turn, monster yang dilawan, dan action yang dapat dilakukan (yaitu attack, potion, monster ball, dan quit). Apabila melakukan attack, akan digunakan fungsi RNG. Apabila menang, user akan mendapat sejumlah O.W.C.A Coins. Apabila kalah, akan ditampilkan sebuah pesan jangan menyerah. }

Fungsi tambahan Battle

—> procedure battleart()

{ I.S. User memasuki Battle }

{ F.S. Menampilkan ASCII Art dari Battle }

- > procedure battlerule()
 - { I.S. User memasuki Battle }
 - { F.S. Menampilkan list peraturan untuk Battle }
- —> Arena
 - { I.S. Menerima input user berupa konfirmasi memasuki arena, lalu apabila iya akan meminta input monster yang akan digunakan pada arena }
 - { F.S. Melakukan proses Battle tipe arena, dengan tahapan bertotal 5 stage yang sudah di-preset level monster musuhnya dari stage ke stage (Contoh: pada stage 1, level monster musuh juga level 1), apabila menyelesaikan 5 stage arena, user akan mendapatkan sejumlah O.W.C.A Coins. Apabila kalah pada salah satu Stage, maka arena akan di-reset dan ditampilkan pesan silahkan coba lagi. }

Fungsi tambahan Arena

 - > procedure arenaart()
 - { I.S. User mengakses arena }
 - { F.S. Menampilkan ASCII Art dari Arena }
 - > procedure arenarule()
 - { I.S. User mengakses arena }
 - { F.S. Menampilkan list peraturan dari Arena }
- —> Shop
 - function shopOpen()
 - { I.S. Menerima input user antara lihat, beli, atau keluar. }
 - { F.S. Apabila lihat, maka akan ditampilkan sebuah tabel monster atau potion (sesuai pilihan user). Apabila beli, maka akan ditampilkan Coins yang dimiliki dan menerima masukan id monster/potion agar kemudian item/potion dapat terbeli dan masuk ke dalam inventory. Apabila Coins kurang dari harga yang akan dibeli, maka akan ditampilkan pesan Coins tidak cukup. Apabila keluar program Shop akan berhenti }

Fungsi tambahan Shop

 - > procedure shopInterface()
 - { I.S. Mengecek role user }
 - { F.S. Menampilkan ASCII Art tergantung role user pada Shop }
- —> Inventory
 - procedure inventory()
 - { I.S. Menerima input berupa hal yang ingin dicek di inventory (monster atau item) }
 - { F.S. Akan ditampilkan jumlah Coins yang dimiliki user, lalu apabila user memilih monster, akan ditampilkan monster beserta status lengkapnya. Apabila user memilih item, akan ditampilkan potion dan monster ball yang dimiliki beserta jumlahnya. }
- —> Laboratory
 - function laboratory()
 - { I.S. Menerima input berupa monster yang ingin di-upgrade levelnya dan pilihan ingin upgrade hingga level berapa }

- { F.S. Apabila input valid, akan ditampilkan harga upgrade yang akan dilakukan. Apabila Coins user mencukupi, maka upgrade akan dilakukan dan monster Level Up hingga level yang telah dipilih. }
- —> Jackpot


```
function jackpot()
{ I.S. Menerima input berupa konfirmasi ingin main jackpot atau tidak }
{ F.S. Apabila user konfirmasi dan Coins mencukupi, maka akan dilakukan jackpot. User tetap akan mendapatkan Coins meskipun tidak mendapatkan Jackpot, sejumlah benda yang tampil pada layar jackpot. Apabila jackpot pertama kali, user akan mendapat sebuah monster eksklusif. Jika jackpot kedua kali, akan mendapat Coins. Apabila coins tidak cukup, akan ditampilkan pesan Coins tidak cukup. }
```
 - —> Exit


```
procedure exit()
{ I.S. Menerima input user apakah ingin save sebelum keluar dari game }
{ F.S. Apabila user menjawab 'y', maka akan dijalankan fungsi save() lalu keluar dari game. Apabila 'n', maka akan langsung keluar dari game. }
```
 - —> Save


```
procedure save()
{ I.S. Menerima input user berupa nama folder untuk save file }
{ F.S. Apabila folder sudah ada, maka akan dilakukan overwrite save pada file tersebut. Apabila folder belum ada, maka akan dibuat sebuah folder baru, lalu file akan disave pada folder tersebut. File yang disave yaitu inventory item, monster user, data user, item shop, monster shop, dan inventory monster. }
```
 - —> Load


```
function load()
{ I.S. Menerima input user berupa nama folder yang akan di-Load }
{ F.S. Apabila folder kosong atau tidak ditemukan, akan ditampilkan sebuah pesan. Apabila folder ditemukan, semua file csv akan dibaca dan di-load ke dalam array. }
```
 - —> Shop (Management)


```
procedure shopmanagement()
{ I.S. Menerima input user (Admin) antara lihat, tambah, ubah, hapus, atau keluar }
{ F.S. Apabila lihat, akan ditampilkan tabel monster/item. Apabila tambah, akan ditambahkan sebuah monster/item yang belum ada di shop sesuai input user. Apabila ubah, akan diubah harga dan stok dari sebuah monster/item sesuai input user. Apabila hapus, sebuah monster/item yang dipilih akan dihapus dari shop. Apabila keluar, maka program akan berhenti }
```
 - —> Monster (Management)


```
procedure monsterManagement()
{ I.S. Menerima input user (Admin) antara lihat, tambah monster baru, atau keluar }
```

- { F.S. Apabila lihat, akan ditampilkan list monster yang telah ada pada database. Apabila tambah monster, akan dibuat monster dengan nama, attack, defense, HP, sesuai input user. Apabila keluar, maka program akan keluar. }
- Random Number Generator


```
function RNG
{ I.S. Belum diperoleh Random Number }
{ F.S. memunculkan angka random dengan menggunakan konsep Linear Congruential Generator dengan range tertentu. }
```

Spesifikasi, Dekomposisi Algoritmik dan Fungsional Program, dan Kamus Data

F00 - Random Number Generator

```
SPESIFIKASI DAN DESKRIPSI
function RNG(x,y : float) -> integer
{ I.S. x dan y terdefinisi, F.S fungsi untuk mendapatkan angka random dari interval tertentu }

KAMUS
    a, c, m : integer
    function GenerateNumber(x2, num_range : integer) ->
integer
    { I.S x2 dan num_range terdefinisi, F.S fungsi untuk mendapatkan angka random }
    function LCG(a, c, m : integer) -> integer
    { I.S a, c, dan m terdefinisi, F.S fungsi berisi rumus dasar Linear Congruential Generator }

ALGORITMA UTAMA
    import os
    import time
    a <- 48271
    c <- 0
    m <- 2**31 - 1
    -> GenerateNumber(LCG(a,c,m),[x,y])

{REALISASI FUNGSI}
function GenerateNumber(x2, num_range : integer) -> integer
{ I.S. x2 dan num_range terdefinisi, F.S fungsi untuk mendapatkan angka random }

KAMUS LOKAL
    a, c, m, x2 : integer
ALGORITMA
```

```

a <- 48271
c <- 0
m <- 2**31 - 1
x2 <- (a * x2 + c) mod m
-> int((x2/(m-1)) * (num_range[1]-num_range[0]) +
num_range[0])

function LCG(a, c, m : integer) -> integer
{ I.S a, c, dan m terdefinisi, F.S fungsi berisi rumus dasar
Linear Congruential Generator }

KAMUS LOKAL
    seed : real
    x2 : integer

ALGORITMA
    seed <- int(os.getpid())123 + time.time()45
    x2 <- (a * seed + c) mod m
    -> x2

```

F01 - Register

SPESIFIKASI DAN DESKRIPSI

Program Register

{Program menerima input berupa konfirmasi untuk melakukan logout
serta mengganti status login menjadi False bila logout telah
dikonfirmasi}

KAMUS

```

usernamebaru : string { meminta username baru }
status : boolean { status login, True jika sudah login }
valid : boolean { apakah username valid, True jika sudah
sesuai format (huruf,angka,strip dan underscore) }
append(l: list, x: eltype) { menambahkan elemen x ke list l }
ascii_value : string

```

ALGORITMA UTAMA

```

while true do
    input "Masukkan Username :"
    status <- false
    i traversal usernamebaru do
        ascii_value <- ord(char)
        valid <- (

```

```

        (65 <= ascii_value <= 90) or # A-Z
        (97 <= ascii_value <= 122) or # a-z
        (48 <= ascii_value <= 57) or # 0-9
        ascii_value = 45 or           # strip (-)
        ascii_value = 95             # underscore (_)
    )
    if valid = false then
        output "Username hanya boleh mengandung
huruf, angka, strip dan underscore"
        break
    else then
        status <- true
    if status = false then
        continue
    traversal i in user do
        if i[1] = usernamebaru then
            output "Username sudah terpakai"
            status <- false
            break
        else then
            status <- true
        if status = false then
            continue
        break
    input "Masukkan password :"
    input "PILIH MONSTER PERTAMA ANDA :"
    for i in range(3)
        output "(i+1) + ". " + monster[i+1][1]"
    input "--->"
    idbaru <- length(user)
    append (monsterinventory, [str(idbaru),
monster[int(idpilihan)][1], '1'])
    append (user, [str(idbaru), usernamebaru, passwordbaru,
'agent', '0'])
    append (inventory, [str(idbaru), 'healingpotion', '0'])
    append (inventory, [str(idbaru), 'respotion', '0'])
    append (inventory, [str(idbaru), 'strenghpotion', '0'])

    player_inventory <- []
    monster_inventory <- []
    input "SELAMAT DATANG AGENT BARU"
    for j in inventory
        if j[0] = str(idbaru) then
            append (player_inventory, j)
    for j in monsterinventory

```

```

    if j[0] = str(idbaru) then
        append (monster_inventory, j)
    -> player_inventory, monster_inventory, 'agent', 0
if(loginStatus = false) then
    end

```

F02 - Login

SPESIFIKASI DAN DESKRIPSI

Program Login

{ Program menerima input berupa username dan password dari pengguna serta memiliki fitur seperti melakukan pemeriksaan apakah username dan password tersebut sesuai dan menambahkan barang serta monster ke dalam inventory}

KAMUS

```

username : string
password : string
player_inventory : array
monster_inventory : array
append(l: list, x: eltype) { menambahkan elemen x ke list l }

```

ALGORITMA UTAMA

```



```

```

break

elif i[3] = 'admin' then
    -> 1, 1, i[3], 1
else then
    output "password salah"
    -> 0, 0, 0, 0
    break
else then
    output "username tidak ditemukan"
    -> 0, 0, 0, 0

```

F03 - Logout

SPESIFIKASI DAN DESKRIPSI

Program Logout

{ Program mengganti status login menjadi False bila logout telah dikonfirmasi }

KAMUS

```

confirm : string      { konfirmasi logout }
loginStatus : boolean { status login, True jika sudah
login }

```

ALGORITMA UTAMA

```

if(loginStatus = True) then
    iterate
        input(confirm)
        if(confirm = 'y') then
            output("Logout berhasil")
            loginStatus = false
            break
        else if(confirm = 'n') then
            break
        else then
            output("Masukkan input yang benar")
        if(loginStatus = false) then
            break

```

F04 - Menu & Help

SPESIFIKASI DAN DESKRIPSI

Program Help

{ Program menampilkan bantuan dari Almighty God berupa fitur yang dapat diakses/dilakukan oleh pengguna pada saat pengguna menggunakan Help }

KAMUS

{ Deklarasi Fungsi dan Prosedur }
procedure help(status, role : string)

KAMUS LOKAL

status, role : string

ALGORITMA UTAMA

```
if status = false then
    output("List command yang dapat anda lakukan : ")
    output("1. Login")
    output("    Login ke akun admin/agent")
    output("2. Register")
    output("    Membuat akun agent baru")
    output("3. Save")
    output("    Menyimpan semua perubahan dalam sebuah
    folder")
    output("4. Exit")
    output("    Keluar dari program")
else
    if role = "agent" then
        output("List command yang dapat anda lakukan : ")
        output("1. Battle")
        output("    1v1 dengan monster random")
        output("2. Arena")
        output("    tempat pelatihan monster anda")
        output("3. Shop")
        output("    Tempat jual beli potion dan monster")
        output("4. Inventory")
        output("    Cek semua item dan monster anda")
        output("5. Laboratory")
        output("    Tempat untuk meng-upgrade monster anda")
        output("6. Jackpot")
        output("    Tempat untuk menguji keberuntungan anda")
        output("7. Logout")
        output("    Keluar dari akun anda")
    if role = "admin" then
        output("List command yang dapat anda lakukan : ")
        output("1. Shop")
```

```

output(" Manage shop")
output("2. Monster")
output(" Manage monster")
output("3. Logout")
output(" Keluar dari akun anda")

```

F06 - Potion

SPESIFIKASI DAN DESKRIPSI

Program Potion

{ Program memberi efek penambahan stats pada monster saat battle tergantung jenis potion yang diminum }

KAMUS

```

{ Deklarasi Variabel }

{ Deklarasi fungsi/prosedur }
function jumlahPot(userInventory: array of array of string) ->
[jumlahStr, jumlahRes, jumlahHeal] : array of integer
{ I.S. User menjalankan command potion, F.S. Didapatkan array
jumlah masing-masing jenis potion di inventory }

```

KAMUS LOKAL

jumlahStr, jumlahRes, jumlahHeal: string

ALGORITMA

```

for i in userInventory
    if i[1] = 'Strength Potion' then
        jumlahStr <- i[2]
        break
    else then
        jumlahStr <- 0
for i in userInventory
    if i[1] = 'Resilience Potion' then
        jumlahRes <- i[2]
        break
    else then
        jumlahRes <- 0
for i in userInventory
    if i[1] = 'Healing Potion' then
        jumlahHeal <- i[2]
        break

```

```

        else then
            jumlahHeal <- 0
        -> [int(jumlahStr), int(jumlahRes), int(jumlahHeal)]
```

function indexPot(userInventory:array of array of string) -> [indexStr, indexRes, indexHeal] : array of integer
{ I.S. User menjalankan command potion, F.S. index masing-masing jenis potion di array userInventory }

KAMUS LOKAL

```

    indexStr, indexRes, indexHeal: integer
```

ALGORITMA

```

    indexStr <- 0
    indexRes <- 0
    indexRes <- 0
    for i in userInventory
        if i[1] /= 'Strength Potion' then
            indexStr <- indexStr + 1
        else if i[1] = 'Strength Potion' then
            break
    for i in userInventory
        if i[1] /= 'Resilience Potion' then
            indexStr <- indexStr + 1
        else if i[1] = 'Resilience Potion' then
            break
    for i in userInventory
        if i[1] /= 'Healing Potion' then
            indexStr <- indexStr + 1
        else if i[1] = 'Healing Potion' then
            break
    -> [indexStr, indexRes, indexHeal]
```

function potion(jumlahpot, indexpot, used_Pot_Array: array of integer, userInventory:array of array of string, attack, defense, HP, max_HP, pilihan: integer, mons_Name: string) -> attack, defense, HP : integer
{ I.S. User menjalankan command potion, F.S. stats monster user dapat bertambah/tidak berubah sama sekali }

KAMUS LOKAL

ALGORITMA

```

    if pilihan = 1 then
        if int(jumlahpot[0]) > 0 then
            if used_Pot_Array[0] = 0 then
                used_Pot_Array[0] <- used_Pot_Array[0] + 1
```

```

        attack <- attack * 105/100
        output("Potion STRENGTH telah diminum,
"+mons_Name+" menjadi lebih KUAT!")
        jumlahPot[0] <- jumlahPot[0] - 1
        userInventory[indexPot[0]][2] <-
str(int(userInventory[indexPot[0]][2])-1)
        -> attack
    else then
        output("Potion STRENGTH hanya bisa digunakan
sekali")
        -> 0
    else then
        output("Yahh.. Sayangnya Anda tidak memiliki potion
STRENGTH")
        -> 0

    if pilihan = 2 then
        if int(jumlahpot[1]) > 0 then
            if used_Pot_Array[1] = 0 then
                used_Pot_Array[1] <- used_Pot_Array[1] + 1
                defense <- defense * 110/100
                output("Potion RESILIENCE telah diminum,
"+mons_Name+" menjadi lebih SULIT DILUKAI!")
                jumlahPot[1] <- jumlahPot[1] - 1
                userInventory[indexPot[1]][2] <-
str(int(userInventory[indexPot[1]][2])-1)
                -> defense
            else then
                output("Potion RESILIENCE hanya bisa digunakan
sekali")
                -> 0
        else then
            output("Yahh.. Sayangnya Anda tidak memiliki potion
RESILIENCE")
            -> 0

    if pilihan = 3 then
        if int(jumlahpot[2]) > 0 then
            if used_Pot_Array[2] = 0 then
                used_Pot_Array[2] <- used_Pot_Array[2] + 1
                if HP * 125/100 <= max_HP then
                    HP <- HP * 125/100
                else HP * 125/100 <= max_HP then
                    HP <- max_HP
                output("Potion HEALING telah diminum, HP

```

```

"+mons_Name+" telah REGENERASI")
        jumlahPot[2] <- jumlahPot[2] - 1
        userInventory[indexPot[2]][2] <-
str(int(userInventory[indexPot[2]][2])-1)
        -> HP
    else then
        output("Potion HEALING hanya bisa digunakan
sekali")
        -> 0
    else then
        output("Yahh.. Sayangnya Anda tidak memiliki potion
HEALING")
        -> 0

```

F07 - Inventory

SPESIFIKASI DAN DESKRIPSI

Program Inventory

{ Program mengganti status login menjadi False bila logout telah dikonfirmasi }

KAMUS

```

{ Deklarasi Variabel }

{ Deklarasi fungsi/prosedur }
procedure inventory(userInventory, yourMonsData, monsterData:
array of array of string, role: string, coin: integer)
{ I.S. User menjalankan command inventory, F.S. menunjukkan
inventory user serta jumlah coin yang dimiliki}

```

KAMUS LOKAL

```

check, itemNumber, monsNumber: string
number, monsLevel: integer
inputNum: boolean
itemArray: array of string

```

ALGORITMA

```

output("jumlah OWCA coin anda saat ini : "+coin+"")
while true do
    input(check)
    if check = 'item' then

```

```

number <- 1
itemArray <- []
for i in userInventory
    append i[1] to itemArray
    output("+"+number+". "+i[1]+")
    number <- number + 1
while true do
    input(itemNumber)
    inputNum <- true
    for char in itemNumber
        if ord(char) < ord("0") or ord(char) >
            ord("9") then
                inputNum <- false
            if inputNum /= false then
                if itemNumber /= "back" then
                    if int(itemNumber) > len(itemArray) then
                        output("Nomor item tersebut tidak
tersedia")
                    else then
                        itemNumber <- int(itemNumber)
                        output("Type:
"+itemArray[itemNumber-1]+")
                        output("Quantity:
"+userInventory[itemNumber-1][2]+")
                    else if itemNumber = "back" then
                        break
                    else then
                        output("Masukkan input yang benar")
                else if check = "monster" then
                    number <- 1
                    for i in yourMonsData
                        output("+"+number+". "+i[1]+")
                        number <- number + 1
                    while true do
                        input(monsNumber)
                        inputNum <- true
                        for char in monsNumber
                            if ord(char) < ord("0") or ord(char) >
                                ord("9") then
                                inputNum <- false
                            if inputNum /= false then
                                if monsNumber /= "back" then
                                    if int(monsNumber) > len(yourMonsData)
                                        or int(monsNumber) < 1 then
                                            output("Nomor monster tersebut tidak
")

```

```

tersedia")
    else then
        monsNumber <- int(monsNumber)
        monsLevel <-
int(yourMonsData[monsNumber-1][2])
        for i in monsterData
            if i[1] =
yourMonsData[monsNumber-1][2] then
            output("Nama : "+i[1]+")
            output("ATK Power :
"+int(i[2])+(monsLevel-1)*(5/100)*(int(i[2]))+")
            output("DEF Power :
"+int(i[3])+(monsLevel-1)*(5/100)*(int(i[3]))+")
            output("HP :
"+int(i[4])+(monsLevel-1)*(5/100)*(int(i[4]))+")
            output("Level :
"+monsLevel+"))
        else if monsNumber = "back" then
            output("Sampai jumpa lagi !!")
            break
        else then
            output("Mohon berikan input yang benar")

```

F08 - Battle

SPESIFIKASI DAN DESKRIPSI

```

function
battle(monsterdata,monsterinventory,yourmonsterdata,userinventory,
chosen,random_number,random_level,type)
KAMUS
    file1 : array of array of string
    used_Pot : array of integer
    level,name,action,potionchoice : string
    indeks,originalhealth,yourattack,yourdefense,yourhealth,enemyhealth,
    enemyattack,enemydefese,damagetaken,damagedealt,turun,damagerng : integer
    procedure battlerule()
    {procedure untuk menampilkan mekanisme battle}
    procedure battleart()
    {procedure untuk menampilkan ascii art battle}
    function choose()
    {function untuk memilih monster yang akan anda gunakan}

```

```
procedure opening()
{procedure untuk menampilkan monster musuh memasuki battle}
```

ALGORITMA UTAMA


```

        break
    else if potionchoice = 3 then
        if newstat = 0 then
            continue
        else
            yourhealth <- int(newstat)
            break
    else if potionchoice = 4 then
        break
    if potionchoice = 4 then
        continue
    else if action = 3 then
        if type = 'battle' then
            success <-
                monsterball(random_number,random_level,userinventory,monster
                data,monsterinventory,yourmonsterinventory,)
            if success = 1 then
                output(file1[random_number][1].upper + "IS ADDED
                TO YOUR INVENTORY")
                -> 0,damagetaken,damagedealt
            else if success = 2 then
                continue
            else if type = 'arena' then
                output("=====")
                output("you left the arena")
                output("=====")
                coin <- 0
                -> coin,damagetaken,damagedealt
            else if action = 4 then
                if type = 'arena; then
                    output("OPTION 4 IS NOT AVAILABLE")
                else if type = 'battle' then
                    output("=====")
                    output('YOU LEFT THE BATTLEFIELD')
                    output("=====")
                    coin <- 0
                    -> coin,damagetaken,damagedealt
                else
                    output("Option" +action+"is not available")
                    continue
            if enemyhealth <= 0 then
                output("=====")
                output("YOU DEFEATED YOUR ENEMY")
                output('WELL DONE')
                coin <- rng(1,100)
                if type.lower = 'battle' then
                    output("you got" +coin+ "coins")
                    output("=====")
                    -> coin,damagetaken,damagedealt
                damagemusuhrng <- rng(70,130)
                damagemusuh <- enemyattack*(damagemusuhrng/100)

```

```

    output(">>>>>>"> +enemy+ "attack back<<<<<<")
    if yourhealth >= (damagemusuh -
    (yourdefense/100*damagemusuh)) then
        damagetaken <= damagetaken + (damagemusuh -
    (yourdefense/100*damagemusuh))
    else
        damagetaken <- damagetaken + yourhealth
        yourhealth <- yourhealth - (damagemusuh -
    (yourdefense/100*damagemusuh))
        yourhealth <- int(yourhealth)
    if damagemusuh < 100 then
        output("attack damage ;" damagemusuh (damagemusuhrng-100)
    "%")
    else
        output("attack damage ;" damagemusuh
    (+damagemusuhrng-100) "%")
    output("reduced damage :" yourdefense/100*damagemusuh
    (yourdefense"))
    output("Total damage :" (damagemusuh -
    (yourdefense/100*damagemusuh)))
    if yourhealth <= 0 then
        output(" YOU ARE DEAD DONT GIVE UP")
        -> return 0,damagetaken,damagedealt
    turn <- turn +1
{REALISASI FUNGSI DAN PROSEDUR}
function choose
while True do
    output("please select amonster"
x<-0
for i in yourmonsterdata :
    output(x+1. i[1].upper)
    x<-x+1
chosen <-input(1/2/3/4)
k<-0
i traversal (1.. len(yourmonsterdata+1))
if str(i)=chosen then
    k<-1
    break
if k=1 then break
else : output(pilihan ke chosen tidak tersedia)
-> int(chosen)
procedure opening
output("monster mu

```

F09 - Arena

```
SPESIFIKASI DAN DESKRIPSI
function arena(monsterdata, monsterinventory, yourmonsterdata,
userinventory : array of arrays) -> integer
{ fungsi untuk menjalankan fitur arena pada permainan }

KAMUS
    reward : array of integer
    coins, totaldamageTaken, totaldamageDealt, chosen,
random_number, stage : integer
    procedure arenarule()
    { Procedure untuk menampilkan aturan dari arena }
    procedure arenaart()
    { Procedure untuk mencetak art ascii untuk memperindah
tampilan arena }

ALGORITMA UTAMA
    from src.battle import *
    from src.RandomNumberGenerator import *
    reward <- [30,45,60,75,90]
    coins <- 0
    totaldamageTaken <- 0
    totaldamageDealt <- 0
    confirm <- arenarule
    if confirm.lower = "n" then
        output("SEE YOU ANOTHER DAY")
    else
        chosen <- choose(yourmonsterdata)
        stage <- 1
        i traversal [0..4]
            random_number <- RNG(1,len(monsterdata))
            enemylevel <- i + 1
            output("----->>>STAGE " + (i+1) +
"-----")
            opening(monsterdata,random_number)
            coin,damageTaken,damageDealt <-
battle(monsterdata, monsterinventory, yourmonsterdata,
userinventory, chosen, random_number, enemylevel, 'arena')
            totaldamageTaken <- totaldamageTaken +
damageTaken
```

```

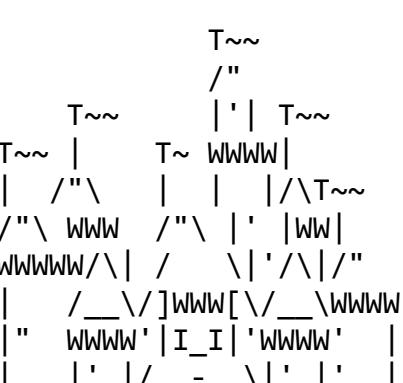
        totaldamageDealt <- totaldamageDealt +
damageDealt
        if coin = 0 then
            output("ARENA IS OVER, TRY AGAIN")
            end
        else
            stage <- stage + 1
        i traversal [1..stage-1]
            coins <- coins + reward[i-1]
        if stage = 6 then
            output("CONGRATULATIONS YOU MADE IT ALL THE
WAY!!!!")
            output("HERE A BONUS 50 COINS FOR YOU!!!")
            coins <- coins + 50
        output("=====")
        output("STATISTICS FROM THE ARENA :")
        output("NUMBER OF STAGE : " + (stage-1))
        output("TOTAL COINS GAINED FROM THE ARENA : " + coins)
        output("TOTAL DAMAGE TAKEN : " + totaldamageTaken)
        output("TOTAL DAMAGE DEALT : " + totaldamageDealt)
        output("=====")

{ REALISASI FUNGSI DAN PROCEDURE }
procedure arenarule()
{ Procedure untuk menampilkan aturan dari arena }
KAMUS LOKAL
    confirm : char
    procedure arenaart()
        { Procedure untuk mencetak art ascii untuk memperindah
tampilan arena }
ALGORITMA
    arenaart()
    output("")

=====WELCOME TO THE
ARENA=====
ARENA MECHANISM :
    1. THERE WILL BE A TOTAL OF 5 STAGES IN THE ARENA, EACH
STAGE YOU WILL BE BATTLING
        A RANDOMLY SELECTED MONSTER
    2. YOUR ENEMY LEVEL WILL BE EQUAL TO THE STAGE NUMBER YOUR
CURRENTLY ON
    3. DEFEATING EACH STAGE WILL GIVE YOU A REWARD, THE HIGHER
THE STAGE, THE
        THE GREATER THE REWARD
    4. IF YOU LOSE ONCE, THE ARENA WILL BE OVER

```

5. IF YOU WIN ALL THE WAY, YOU WILL GET A BONUS REWARD
 6. POTIONS EFFECT ONLY LAST TILL THE END OF EACH STAGE
 7. MONSTER BALL IS PROHIBITED IN THE ARENA """")
iterate
 input(confirm)
 until confirm.lower() = 'y' or confirm.lower() = 'n'
 output("BE CLEAR")
 -> confirm

procedure arenaart()
{ Procedure untuk mencetak art ascii untuk memperindah tampilan arena }
KAMUS LOKAL
ALGORITMA
 output(“””

 T~~
 /"
 T~~ | ' | T~~
 T~~ | T~ WWW |
 | /"\ | | | | / \ T~~
 /"\ WWW /"\ | ' | WWW |
 WWWWW /\ | / \ | ' / \ | / "
 | / \] WWW [\ / \ WWW
 " WWW ' | I_I | ' WWW '
'	/ - \	'	'	
'	LI=H=LI	'	'	
'	[_]		'	
_	###	_		'
 ----- / \ -----)
 ”””)

F10 - Shop

SPESIFIKASI DAN DESKRIPSI
Program ShopOpen
{ Program menampilkan tampilan Shop, dimana user Agent bisa melakukan lihat, beli, atau keluar }

```
KAMUS
{ Deklarasi Variabel }
itemID : array of string
{ Deklarasi fungsi/prosedur }
procedure shopInterface()
```

```

{ I.S. User memasuki Shop, F.S. Ditampilkan ASCII Art
tampilan dari Shop }

KAMUS LOKAL
ALGORITMA
output(" ████ ██████████ SHOP ██████████ ████")
output(" |_Δ||Δ|||Δ| |$_| ")
output(" ||Δ|||Δ|| | || ")
output(" |Δ_||Δ_||Δ||| | \\" )
output(" ||_Δ|||Δ|_| KASIR ███")
output(" ======")

function
shopOpen(role,potionShop,userInventory,monsterdata,monster
shop,yourmonsterdata,monsterinventory : array of array of
string, coin : integer) -> yourcoin : integer
KAMUS LOKAL
yourcoin,idMonsterLihat,idPotion,transactionStatus,idM
onsterBeli,idPotionBeli,jumlahPotionBeli : integer
pilihMenu, lihat_pilih : string
lihatSelesai,beliSelesai : boolean
ALGORITMA
yourcoin <- coin
if role = "agent" then
    output("SELAMAT DATANG DI SHOP")
    output()
    shopInterface()
    output()
    while true
        do pilihMenu <- input(">>> Pilih aksi
(lihat/beli/keluar): ")
        if pilihMenu = "lihat" then
            lihat_pilih <- input("Pilih antara
monster/item: ")
            lihatSelesai <- false
            if lihat_pilih = "monster" then
                output("ID | Type | ATK Power | DEF Power
| HP | Stok | Harga")
                i traversal [1..length(monstershop)]
                idMonsterLihat <- monstershop[i][0]
                output("monstershop[idMonsterLihat][0] |
monstershop[idMonsterLihat][1] |
monstershop[idMonsterLihat][2] |
monstershop[idMonsterLihat][3] |
monstershop[idMonsterLihat][4] |")

```

```

        monstershop[i][1] | monstershop[i][2]"")
else if lihat_pilih = "item" then
    output("ID | Stok | Harga")
    i traversal [1,length(potionShop)]
    if potionShop[i][0] = "Healing Potion"
    then
        idPotion <- 1
    else if potionShop[i][0] = "Resilience
    Potion" then
        idPotion <- 2
    else if potionShop[i][0] = "Strength
    Potion" then
        idPotion <- 3
    else if potionShop[i][0] = "Monster
    Ball" then
        idPotion <- 4
    output("idPotion | potionShop[i][1] |
    potionShop[i][2]")
    lihatSelesai <- true
else if pilihMenu = "beli" then
    output("Jumlah O.W.C.A Coin-mu sekarang
    yourcoin.")
    output()
    beliTipe <- input(">>> Pilih antara
    monster/item: ")
    if beliTipe = "monster" then
        while true
            transactionStatus <- 0
            idMonsterBeli <- input(">>> Masukkan id
            monster: ")
            i traversal [0..length(monstershop)]
            if idMonsterBeli = i[0] then break
            else output("Id monster tersebut
            tidak tersedia")
            continue
        i traversal [0..length(yourmonsterdata)]
        if monsterdata[idMonsterBeli][1] =
        i[1] then output("Monster sudah
        dimiliki")
        break
        else
            i traversal
            [0..length(monstershop)]
            if yourcoin >= i[2] then
            output("Monster berhasil
            dibeli")
    
```

```

ditambahkan")
    yourcoin = yourcoin - i[2]
    yourmonsterdata =
    yourmonsterdata +
    [userinventory[0][0],monsterdata[
    idMonsterBeli][1],'1']
    monsterinventory =
    monsterinventory +
    [userinventory[0][0],monsterdata[
    idMonsterBeli][1],'1']
    monsterShopLocation <- 0
    i traversal
    [0..length(monstershop)]
        if i[0] = idMonsterBeli then
            monstershop[monsterShopLocatio
            n][1] =
            monstershop[monsterShopLocation]
            [1] - 1
            break
            monsterShopLocation =
            monsterShopLocation + 1
            transactionStatus <- 1
            break
        else
            output("Maaf O.W.C.A coin-mu
            kurang.")
            break
        if transactionStatus = 1 then break
        output()
else_if beliType "item" then
    while true
        idPotion <- input(">>> Masukkan id item:
        ")
        i traversal [1..5]
        if idPotion = i then break
        else output("Id tersebut tidak
        tersedia")
        continue
    while true
        jumlahPotionBeli <- input(">>>
        Masukkan jumlah: ")
        if potionShop[idPotion][1] > 0 then
            jumlahPotionBeli =
            jumlahPotionBeli
            if jumlahPotionbeli >

```

```

potionShop[idPotion][1] then
    output("Stock tidak cukup")
else if yourcoin >=
jumlahPotionBeli*potionShop[idPotio
n][2] then output("Berhasil membeli
item. Item sudah masuk ke
inventory-mu.")
    userinventory[idPotion-1][2]=us
erinventory[idPotion-1][2]+jumlah
PotionBeli
    yourcoin = yourcoin -
jumlahPotionBeli*potionShop[idPot
ion][1]
    potionShop[idPotion][1] =
potionShop[idPotion][1] -
jumlahPotionBeli
    break
else output("Maaf O.W.C.A Coin-mu
kurang") break
else output("Stock Potion habis!")
    break
else output("Tolong masukkan integer
yang benar")
    break
else if pilihMenu = "keluar" then
    output("Sampai bertemu lagi,user!")
else output("Aksi tidak diketahui")
else output("Anda tidak memiliki akses.")

```

ALGORITMA

F11 - Laboratory

SPESIFIKASI DAN DESKRIPSI

Program Laboratory

{ Program menjalankan menu Laboratory dimana user dapat
meng-upgrade monster yang dimiliki dengan menggunakan O.W.C.A
Coins }

KAMUS

{ Deklarasi Variabel }
pilihMons,monsUpgrade,lanjut : string

```

number,harga : integer
inputMonsUp,inputMonsNum : boolean
arrayHarga : array of integer
{ Deklarasi Fungsi dan Prosedur }
function laboratory(yourMonsData:array of array of string,
coin:integer, role:string) -> coin : integer

ALGORITMA UTAMA
if role /= "admin" then
    arrayHarga <- [100,200,400,700]
    while true
        output("JUMLAH COIN ANDA :" + coin)
        output("===== MONSTER LIST =====")
        number <- 1
        i traversal in yourMonsData then
            output(number+". "+i[1]+" Level: "+i[2])
            number <- number + 1
        output("===== UPGRADE PRICE =====")
        i traversal [1..5]
            output(i+. Level"+i+" -> Level "+(i+1)+":
            "+arrayHarga[i-1])

pilihMons <- input("Pilih monster yang ingin di-upgrade
(1/2/dst/Back)")
if pilihMons = "back" then
    output("Sampai bertemu lagi !")
    break
if int(pilihMons) > len(yourMonsData) or int(pilihMons)
< 1 then
    inputMons = false
if inputMons /= false then
    inputMonsNum <- true
    char traversal in pilihMons
        if ord(char) < ord('0') or ord(char) > ord('9')
            inputMonsNum <- false
        if inputMonsNum /= false then
            while true
                if yourMonsData[pilihMons-1][2] /= '5' then
                    monsUpgrade <- input("Silahkan pilih ke level
berapa kah anda ingin meng-upgrade monster Anda
(1/2/dst/back): ")
                    inputMonsUp <- true
                    char traversal in monsUpgrade
                        if ord(char) < ord('0') or ord(char) >
ord('9') then
                            inputMonsUp = false

```

```

output(monsUpgrade)
if inputMonsUp /= false then
    if monsUpgrade <=
yourMonsData[pilihMons-1][2] then
    output("Level monster anda sudah lebih
tinggi dari level yang anda pilih!")
else if monsUpgrade > 5 then
    output("Level monster tidak bisa melebihi
5!")
else
    harga <- 0
    i traversal
[yourMonsData[pilihMons-1][2]..monsUpgrade]
    harga <- harga + arrayHarga[i-1]

    output(yourMonsData[pilihMons-1][1]+"akan
di-upgrade ke level"+monsUpgrade)
output("harga untuk meng-upgrade
"+yourMonsData[pilihMons-1][1]+" adalah
"+harga)

while true
    lanjut <- input("lanjutkan upgrade
(Y/N): ")
    if lanjut = "y" then
        if coin >= harga then
            yourMonsData[pilihMons-1][2] <-
monsUpgrade
            output("Selamat!
"+yourMonsData[pilihMons-1][1]+""
berhasil di-upgrade ke level
"+monsUpgrade+'!')
            output("Coin anda tersisa "+coin)
            break
        else
            output("Maaf! anda tidak memiliki
Coin yang cukup!")
            break
    else if lanjut = "n" then
        break
    else
        output("Mohon masukkan input yang
benar")
        output()
else if monsUpgrade = "back" then

```

```

        break
    else
        output("Masukkan input level yang benar")
        output()
    else
        output("Maaf! Monster yang Anda pilih sudah
memiliki level maksimum. Silahkan pilih monster
lain!")
        break
    else if inputMonsNum = "back" then
        break
    else
        output("Masukkan nomor monster yang benar!")
        output()
    else
        output("Maaf! Anda tidak memiliki akses sebagai
Admin")
-> coin

```

F12 - Shop Management

SPESIFIKASI DAN DESKRIPSI

Program shopmanagement

{ Program manajemen shop dengan fitur lihat, tambah, ubah, hapus, dan keluar yang hanya bisa diakses oleh user dengan role Admin }

KAMUS

{ Deklarasi Variabel }

itemID : array of string

{ Deklarasi Fungsi dan Prosedur }

procedure shopmanagement(potionShop,monstershop,monsterdata
: array of array of string)

KAMUS LOKAL

pilihMenu, lihat_pilih,

tambah_pilih,ubah_pilih,pesanUbah,hapusPilih, yakinHapus :
string

idMonsterLihat, idPotion, idMonsterTambah,
stokMonstertambah,hargaMonsterTambah,countFalse,idItemTambah,st
okPotionTambah,hargaPotionTambah,idMonsterUbah,stokMonsterBaru,
hargaMonsterBaru,iditemUbah,stokItemBaru,hargaItemBaru,idMonste
rHapus,iditemhapus : integer

lihatSelesai,potionFound,sudahadadiashop : boolean

```

idMonsterShop, idMonsterData, newpotion, addMonsterShop :
array of integer
ALGORITMA
    output("SELAMAT DATANG DI SHOP")
    output()
    shopInterface()
    output()
    while true
        pilihMenu <- input(">>> Pilih aksi
        (lihat/tambah/ubah/hapus/keluar): ")
        if pilihMenu = "lihat" then
            lihat_pilih <- input("Pilih antara monster/item:
            ")
            lihatSelesai <- false
            if lihat_pilih = "monster" then
                output("ID | Type | ATK Power | DEF Power | HP |
                Stok | Harga")
                i traversal [1..length(monstershop)]
                idMonsterLihat <- monstershop[i][0]
                output("monstershop[idMonsterLihat][0] |
                monstershop[idMonsterLihat][1] |
                monstershop[idMonsterLihat][2] |
                monstershop[idMonsterLihat][3] |
                monstershop[idMonsterLihat][4] | monstershop[i][1]
                | monstershop[i][2]")
                else if lihat_pilih = "item" then
                output("ID | Stok | Harga")
                i traversal [1,length(potionShop)]
                if potionShop[i][0] = "Healing Potion" then
                    idPotion <- 1
                else if potionShop[i][0] = "Resilience Potion"
                then
                    idPotion <- 2
                else if potionShop[i][0] = "Strength Potion" then
                    idPotion <- 3
                else if potionShop[i][0] = "Monster Ball" then
                    idPotion <- 4
                output("idPotion | potionShop[i][1] |
                potionShop[i][2]")
                lihatSelesai <- true
            else if pilihMenu = "tambah" then
                idMonsterShop <- []
                idMonsterData <- []
                i traversal [1..length(monstershop[i][0])]
                idMonsterShop = idMonsterShop +

```

```

monstershop[i][0]
i traversal [1..length(monsterdata[i][0])]
idMonsterData = idMonsterData +
monsterdata[i][0]
tambah_pilih <- output("Mau tambah apa?
(monster/item): ")
if tambah_pilih = "monster" then
    output("ID | Type | ATK Power | DEF Power | HP")
    i traversal [1..length(monsterdata)]
    if monsterdata[i][0] in idMonsterShop then
        continue
    else
        idMonsterLihat <- monsterdata[i][0]
        output("i | monsterdata[idMonsterLihat][1]
| monsterdata[idMonsterLihat][2] |
monsterdata[idMonsterLihat][3] |
monsterdata[idMonsterLihat][4]")
output()
idMonsterTambah <- input("">>>> Masukkan id
monster: ")
while idMonsterTambah in idMonsterShop or
idMonsterTambah not in idMonsterData do
    if idMonsterTambah not in idMonsterData then
        output("ID tidak valid! Pilih ID yang
terdapat pada tabel!")
        idMonsterTambah <- input("">>>> Masukkan id
monster: ")
    else if idMonsterTambah in idMonsterShop then
        output("monster sudah ada di Shop!")
        idMonsterTambah <- input("">>>> Masukkan id
monster: ")
    stokMonsterTambah <- input("">>>> Masukkan stok
awal: ")
    while cekint(stokMonsterTambah) = false do
        output("Masukkan integer yang benar")
        stokMonsterTambah <- input("">>>> Masukkan stok
awal: ")
    hargaMonsterTambah <- input("">>>> Masukkan harga:
")
    while cekint(hargaMonsterTambah) = false do
        output("Masukkan integer yang benar")
        hargaMonsterTambah <- input("">>>> Masukkan
harga: ")
    addMonsterShop =
[idMonsterTambah,stokMonsterTambah,hargaMonsterT

```

```

ambah]
monstershop = monstershop + addMonsterShop
output("monsterdata[idMonsterTambah][1] telah
berhasil ditambahkan ke dalam Shop!")
else if tambah_pilih = "item" then
    if length(potionShop) /= 5 then
        output("ID | Type")
        potionFound <- false
        countFalse <- 0
        i traversal [1..5]
        j traversal [1..length(potionShop)]
        if itemID[i] in potionShop[j][0] then
            potionFound <- true
            break
        else
            potionFound <- false
        if countFalse = 0 then
            output("Semua Item sudah tersedia pada
Shop!")
            output()
        else
            while true
                iditemTambah <- input(">>> Masukkan id
                item (1-4): ")
                if cekint(iditemTambah) = false then
                    output("Masukkan integer yang benar")
                    continue
                if iditemTambah < 1 or iditemTambah > 4
                then
                    output("id tidak valid!")
                    continue
                sudahadadishop <- true
                i traversal in potionShop
                if itemID[iditemTambah] = i[0] then
                    output("Item sudah terdapat di Shop")
                    break
                else
                    sudahadadishop <- false
                if sudahadadishop = true then
                    continue
                break
                stokPotionTambah <- input("Masukkan stok
awal: ")
                while cekint(stokPotionTambah) = false do
                    output("masukkan integer yang benar")

```

```

stokPotionTambah <- input("Masukkan stok awal: ")
hargaPotionTambah <- input("Masukkan harga: ")
while cekint(hargaPotionTambah) = false do
    output("masukkan integer yang benar")
    hargaPotionTambah <- input("Masukkan harga: ")
newPotion <-
    [itemID[iditemtambah],stokPotontambah,harga PotionTambah]
potionShop = potionShop + newPotion
output("itemID[iditemTambah telah berhasil ditambahkan ke dalam Shop!")
output()
else if pilihMenu = "ubah" then
    ubah_pilih <- input("Mau ubah apa? (monster/item): ")
    if ubah_pilih = "monster" then
        while true
            idMonsterUbah <- input(>>> Masukkan id monster: ")
            i traversal in monstershop
            if idMonsterUbah = i[0] then
                break
            else
                output("ID tersebut tidak tersedia")
                continue
            stokMonsterTambah <- input("Masukkan stok baru: ")
            while cekint(stokMonsterTambah) = false do
                output("masukkan integer yang benar")
                stokMonsterTambah <- input("Masukkan stok baru: ")
            hargaMonsterTambah <- input("Masukkan harga baru: ")
            while cekint(hargaMonsterTambah) = false do
                output("masukkan integer yang benar")
                hargaMonsterTambah <- input("Masukkan harga baru: ")
            pesanUbah = "monsterdata[idMonsterUbah][1]"
            telah berhasil diubah "
            if (stokMonsterBaru /= "") then
                pesanUbah = pesanUbah + "dengan stok baru sejumlah stokMonsterBaru"

```

```

    i traversal [0..length(monstershop)]
    if monstershop[i][0] = idMonsterUbah
    then
        monstershop[i][1] = stokMonsterBaru
    if stokMonsterBaru /= "" and
    hargaMonsterBaru /= "" then
        pesanUbah = pesanUbah + " dan "
    if hargaMonsterBaru /= "" then
        pesanUbah = pesanUbah + "dengan harga baru
        hargaMonsterBaru"
    i traversal [0..length(monstershop)]
    if monstershop[i][0] = idMonsterUbah
    then
        monstershop[i][2] = hargaMonsterBaru
    pesanUbah = pesanUbah + "!"
    output(pesanUbah)
    break
else if ubah_pilih = "item" then
    while true
        iditemUbah <- input("">>> Masukkan id item: ")
        if cekint(iditemUbah) = false do
            output("masukkan integer yang benar")
            continue
        if iditemUbah > 4 or iditemUbah < 1 then
            output("Id tersebut tidak tersedia")
            continue
        i traversal in potionShop then
            if itemID[iditemUbah] = i[0]
                break
        else
            output("Id tersebut tidak tersedia")
            continue
        stokitemBaru <- input("Masukkan stok baru: ")
        while cekint(stokitemBaru) = false do
            output("Masukkan integer yang benar")
            stokitemBaru <- input("Masukkan stok baru:
            ")
        hargaitemBaru <- input("Masukkan harga baru: ")
        while cekint(hargaitemBaru) = false do
            output("Masukkan integer yang benar")
            hargaitemBaru <- input("Masukkan harga baru:
            ")
        pesanUbah = "itemID[iditemUbah] telah berhasil
        diubah "
        if (stokitemBaru /= "") then

```

```

pesanUbah = pesanUbah + "dengan stok baru
sejumlah stokitemBaru"
i traversal [0..length(potionShop)]
if potionShop[i][0] = iditemUbah then
potionShop[i][1] = stokitemBaru
if stokitemBaru /= "" and hargaitemBaru /= ""
then
pesanUbah = pesanUbah + " dan "
if hargaitemBaru /= "" then
pesanUbah = pesanUbah + "dengan harga baru
hargaitemBaru"
i traversal [0..length(potionShop)]
if potionShop[i][0] = iditemUbah then
potionShop[i][2] = hargaitemBaru
pesanUbah = pesanUbah + "!"
output(pesanUbah)
break
else if pilihMenu = "hapus" then
hapusPilih <- input("">>> Mau hapus apa?
(monster/item): "
if hapusPilih = "monster" then
while true
idMonsterHapus <- input("">>> Masukkan id
monster: ")
i traversal in monstershop
if idMonsterHapus = i[0] then break
else
output("Id tersebut tidak tersedia")
continue
while true
yakinHapus <- input("Apakah anda yakin ingin
menghapus monsterdata[idMonsterHapus][1]
dari shop (y/n)? "
if yakinhapus = "y" then
newMonsterList =
[[{"monster_id","stock","price"]]]
i traversal [1..length(monstershop)]
if monstershop [i][0] = idMonsterHapus
then
continue
else
newMonsterList = newMonsterList +
monstershop[i]
monstershop <- newMonsterList
output("monsterdata[idMonsterHapus][1]

```

```

        telah berhasil dihapus dari Shop!")
        break
    else if yakinHapus = "n" then
        output("Penghapusan dibatalkan!")
        output()
        break
    else
        output("Masukkan Y/N")
        break
else if hapusPilih = "item" then
    while true
        iditemhapus <- input(">>> Masukkan id item: ")
        if cekint(iditemHapus) = false then
            output("masukkan integer yang benar")
            continue
        if iditemHapus > 4 or iditemHapus < 1 then
            output("Id tersebut tidak tersedia")
            continue
        i traversal in potionShop
        if itemID[iditemHapus] = i[0] then
            break
        else
            output("Id tersebut tidak tersedia")
            continue
    while true
        yakinHapus <- input("Apakah anda yakin
        ingin menghapus itemID[iditemhapus] dari
        Shop (y/n)?")
        if yakinHapus = "y" then
            newitemList <- []
            i traversal [0..length(potionShop)]
            if potionShop[i][0] =
                itemID[iditemHapus] then
                    continue
            else
                newitemList <- newitemList +
                potionShop[i]
            potionShop <- potionShop + newitemList
            output(itemID[iditemHapus]) telah
            berhasil dihapus dari Shop!
            output()
            break
        else if yakinHapus = "n" then
            output("Penghapusan dibatalkan!")
            output()

```

```

        else
            output("Masukkan (Y/N)")
        break
else if pilihMenu = "keluar" then
    output("Sampai bertemu lagi Admin!")
    -> monstershop, potionshop
else
    output("Aksi tidak diketahui")

```

ALGORITMA

```
import cekint
```

F13 - Monster Management

SPESIFIKASI DAN DESKRIPSI

```

procedure monsterManagement(listMonster : array of arrays)
{ Program menampilkan tabel berisi monster beserta stats
masing-masing monster dan ada juga fitur untuk menambah
monster}
KAMUS
    aksiUser : string
    procedure tabelMonster (listMonster : array of arrays)
        {I.S listMonster terdefinsi, F.S mencetak tabel monster
beserta dengan stats masing-masing}
    procedure newMonster (listMonster : array of arrays)
        {I.S listMonster terdefinisi, F.S memasukkan stats monster
baru ke dalam array of array listMonster}

```

ALGORITMA UTAMA

```

    iterate
        output("SELAMAT DATANG DI DATABASE PARA MONSTER
!!!!")
        output("1. Tampilkan Semua Monster")
        output("2. Tambah Monster Baru")
        output("(Ketik nomor untuk aksi yang diinginkan,
ketik 0 untuk keluar)")
        input(aksiUser)
    stop aksiUser = "0"
    if (aksiUser = "1") then
        tabelMonster(listMonster)
        input()
    else if (aksiUser = "2")
        newMonster(listMonster)

```

```

    else {aksiUser tidak bukan 0/1/2}
        output("Input tidak valid, input yang diterima
hanyalah 0/1/2")

{REALISASI FUNGSI/PROCEDURE PROGRAM UTAMA}
procedure tabelMonster (listMonster : array of arrays)
{ I.S listMonster terdefinisi, F.S mencetak tabel monster
lengkap dengan stats masing-masing}
KAMUS LOKAL
    largestElement : array of integer
    newHeader : array of string
    function changeInt (listMonster : array of arrays) ->
array of arrays
    {Fungsi untuk mengubah integer pada listMonster menjadi
string}
    function findLargestElement(largestElement : array of
integer, listMonster : array of arrays) -> array of integer
    {Fungsi untuk mencari jumlah elemen yang terbesar pada
setiap data di listMonster, untuk pencetakan tabel agar lebih
rapih}
    procedure changeHeader (newHeader : array of string,
listMonster : array of arrays)
    {I.S newHeader berupa header berupa judul masing-masing
kolom yang berbeda seperti format pada monster.csv dan
listMonster terdefinisi, F.S untuk mengganti header/judul
setiap kolom pada listMonster menjadi newHeader}
    procedure printTabel (listMonster : array of arrays,
largestElement : array of integer)
    {I.S listMosnter dan largestElement terdefinisi, F.S untuk
mencetak tabel monster lengkap dengan statsnya masing-masing}
ALGORITMA
    largestElement <- [-1,-1,-1,-1,-1]
    newHeader <- [“ID”, “Type”, “Atk Power”, “Def Power”, “HP”]
    changeInt(listMonster)
    findLargestElement(largestElement, listMonster)
    changeHeader(newHeader, listMonster)
    printTabel(listMonster, largestElement)

procedure newMonster (listMonster : array of arrays)
{I.S listMonster terdefinisi, F.S memasukkan stats monster baru
ke dalam array of array listMonster}
KAMUS LOKAL
    nAttack, nDefense, nHp : integer
    nType, userInput : string
    function newType(listMonster : array of arrays) -> string

```

```

{fungsi untuk menyimpan data type monster baru}
function newAttack() -> integer
{fungsi untuk menyimpan data atk_power monster baru}
function newDefense() -> integer
{fungsi untuk menyimpan data def_power monster baru}
function newHp() -> integer
{fungsi untuk menyimpan data hp monster baru}
procedure validUserInput(listMonster : array of arrays,
userInput : string, nType : string, nAttack : integer, nDefense
: integer, nHp : integer)
{I.S listMonster, userInput, nType, nAttack, nDefense, dan
nHp semua terdefinisi, F.S validasi userInput apakah sudah
sesuai dengan ketentuan program masing-masing data}

ALGORITMA
    nType <- newType(listMonster)
    nAttack <- newAttack()
    nDefense <- newDefense()
    nHp <- newHp()
    userInput <- ""
    validUserInput(listMonster, userInput, nType, nAttack,
nDefense, nHp)

{REALISASI FUNGSI DAN PROCEDURE PADA PROCEDURE tabelMonster}
function changeInt (listMonster : array of arrays) -> array of
arrays
{Fungsi untuk mengubah integer pada listMonster menjadi string}

KAMUS LOKAL
ALGORITMA
    i traversal [1..len(listMonster)-1]
        for j traversal <- j traversal [1..4]
            if j ≠ 1 then
                listMonster[i][j] <- str(listMonster[i][j])
    -> listMonster

function findLargestElement(largestElement : array of integer,
listMonster : array of arrays) -> array of integer
{Fungsi untuk mencari jumlah elemen yang terbesar pada setiap
data di listMonster, untuk pencetakan tabel agar lebih rapih}

KAMUS LOKAL
ALGORITMA
    i traversal [0..len(largestElement)-1]
        for j traversal <- j traversal
[0..len(listMonster)-1]
            if largestElement[i] < len(listmonster[j][i])
then

```

```

        largestElement[i] <-
len(listMonster[j][i])
-> largestElement

procedure changeHeader (newHeader : array of string,
listMonster : array of arrays)
{I.S newHeader berupa header berupa judul masing-masing kolom
yang berbeda seperti format pada monster.csv dan listMonster
terdefinisi, F.S untuk mengganti header/judul setiap kolom pada
listMonster menjadi newHeader}
KAMUS LOKAL
ALGORITMA
    i traversal [1..4]
        listMonster[0][i] <- newHeader[i]

procedure printTabel (listMonster : array of arrays,
largestElement : array of integer)
{I.S listMosnter dan largestElement terdefinisi, F.S untuk
mencetak tabel monster lengkap dengan statsnya masing-masing}
KAMUS LOKAL
ALGORITMA
    i traversal [0..len(listMonster)-1]
        output(listMonster[i][0], end="")
        for j traversal <- j traversal
[abs(largestElement[0]-(len(listMonster[i][0])))]
        output(" ",end="")
        print(" |", end="")
        output(listMonster[i][1], end="")
        for j traversal
[abs(largestElement[1]-(len(listMonster[i][1])))]
        output(" ",end="")
        print(" |", end="")
        output(listMonster[i][2], end="")
        for j traversal
[abs(largestElement[2]-(len(listMonster[i][2])))]
        output(" ",end="")
        print(" |", end="")
        output(listMonster[i][3], end="")
        for j traversal <- j traversal
[abs(largestElement[3]-(len(listMonster[i][3])))]
        output(" ",end="")
        print(" |", end="")
        output(listMonster[i][4], end="")
        for j traversal <- j j traversal
[abs(largestElement[4]-(len(listMonster[i][4])))]

```

```

        output(" ", end="")
print(" |", end="")
print("")

{REALISASI FUNGSI DAN PROCEDURE PADA PROCEDURE newMonster}
function newType(listMonster : array of arrays) -> string
{fungsi untuk menyimpan data type monster baru}

KAMUS LOKAL
    type : string
    function checkType(type : string, listMonster: array of
arrays) -> string
    {I.S type dan listMonster terdefinisi, F.S mengecek
apabila type/nama monster sudah ada dalam listMonster}

ALGORITMA
    input(Type)
    -> checkType(type,listMonster)

function newAttack() -> string
{fungsi untuk menyimpan data atk_power monster baru}

KAMUS LOKAL
    attack : string
    function checkAttack(attack : integer) -> integer
    {I.S attack terdefinisi, F.S mengecek apabila attack
merupakan integer positif}

ALGORITMA
    input(attack)
    -> checkAttack(attack)

function newDefense() -> string
{fungsi untuk menyimpan data def_power monster baru}

KAMUS LOKAL
    defense : string
    function checkDefense(defense : integer) -> integer
    {I.S defense terdefinisi, F.S mengecek apabila defense
adalah integer dalam rentang 0-50}

ALGORITMA
    input(defense)
    -> checkDefense(defense)

function newHp() -> string
{fungsi untuk menyimpan data hp monster baru}

KAMUS LOKAL
    hp : string
    function checkHp(hp : integer) -> integer

```

```

{I.S hp terdefinisi, F.S mengecek apabila hp adalah
integer positif}
ALGORITMA
input(hp)
-> checkHp(hp)

{REALISASI FUNGSI DAN PROCEDURE PADA FUNGSI newType, newAttack,
newDefense, dan newHp}
function checkType(type : string, listMonster: array of arrays)
-> string
{I.S type dan listMonster terdefinisi, F.S mengecek apabila
type/nama monster sudah ada dalam listMonster, bila tidak ada
maka baru bisa ditambahkan}

KAMUS LOKAL
    monsterExist : boolean           {True bila type monster
sudah ada dalam listMonster}
    type : string
    listMonster : array of arrays

ALGORITMA
repeat
    monsterExist <- False
    i traversal [1..len(listMonster)]
        if listMonster[i][1] = type then
            output("nama sudah terdaftar. Coba lagi!")
            input(type)
            monsterExist = True
until monsterExist = False
-> type

function checkAttack(attack : integer) -> integer
{I.S attack terdefinsi, F.S mengecek apabila attack merupakan
integer positif}
KAMUS LOKAL
    attack : string
    function cekint(input : character) -> boolean
    { fungsi untuk mengecek apakah input berupa angka positif
dan mengubahnya menjadi integer, True berarti input merupakan
integer positif }

ALGORITMA
repeat
    cekint(attack)
    if cekint(attack) = False then
        output("Masukkan integer berupa integer
positif, coba lagi!")
        input(attack)

```

```

until cekint(attack) = True
attack <- int(attack)
-> attack

function checkDefense(defense : integer) -> integer
{I.S defense terdefinisi, F.S mengecek apabila defense adalah
integer dalam rentang 0-50}
KAMUS LOKAL
    defense : string
    function cekint(input : character) -> boolean
        { fungsi untuk mengecek apakah input berupa angka positif
        dan mengubahnya menjadi integer, True berarti input merupakan
        integer positif }
ALGORITMA
    repeat
        cekint(defense)
        if cekint(defense) = False then
            output("Masukkan bukan bertipe integer, coba
lagi!")
            input(defense)
        else if cekint(defense) = True and (def < 0 or def >
50)
            output("Masukkan harus bernilai 0-50")
            input(defense)
        until cekint(defense) = True and (-1 < defense < 51)
        defense <- int(defense)
        -> defense

function checkHp(hp : integer) -> integer
{I.S hp terdefinisi, F.S mengecek apabila hp adalah integer
positif}
KAMUS LOKAL
    hp : string
    function cekint(input : character) -> boolean
        { fungsi untuk mengecek apakah input berupa angka positif
        dan mengubahnya menjadi integer, True berarti input merupakan
        integer positif }
ALGORITMA
    repeat
        cekint(hp)
        if cekint(hp) = False then
            output("Masukkan bukan bertipe integer, coba
lagi!")
            input(hp)
        until cekint(hp) = True

```

```

hp <- int(hp)
-> hp

{REALISASI FUNGSI cekInt}
function cekint(input : character) -> boolean
{ fungsi untuk mengecek apakah input berupa angka positif dan
mengubahnya menjadi integer, True berarti input merupakan
integer positif }
KAMUS LOKAL
    input : character
ALGORITMA
    if input = "" then
        -> False
        end
    input <- str(input)
    if input[0] = "-" then
        if len(input) = 1 then
            -> False
        end
        input <- input[1:]
    for char in input
        if ord(char) < ord('0') or ord(char) > ord('9') then
            -> False
        end
    -> True

```

F14 - Load

```

SPESIFIKASI DAN DESKRIPSI
Program Load
{ Program membuka sebuah folder yang sudah agar pengguna dapat
melanjutkan file yang pernah di-save sebelumnya }
KAMUS
    { Deklarasi variabel }
    parser,args,path : string
    { Deklarasi fungsi dan prosedur}
    function load() -> list
ALGORITMA
    import os
    import argparse
    parser <- argparse.ArgumentParser(add_help <- false, usage

```

```

<- “%(prog)s <nama_folder>”
add_arguments(“nama_folder”, nargs<-”?”, type<-string, default
<-””) to parser
path <- “data/”+args.nama_folder
if path = “data/” then
    output(“Tidak ada nama folder yang diberikan!”)
    output()
    parser.print_usage()
    -> []
else_if os.path.exists(path) then
    output(“Loading...”)
    output(“Selamat datang!”)
    user <- read_csv(path+”/user.csv”)
    inventory <- read_csv(path+”/item_inventory.csv”)
    monster <- read_csv(path+”/monster.csv”)
    monsterinventory <-
        read_csv(path+”/monster_inventory.csv”)
    itemshop <- read_csv(path+”/item_shop.csv”)
    monstershop <- read_csv(path+”/monster_shop.csv”)
    ->
    [user,inventory,monster,monsterinventory,itemshop,monster
    shop]
else
    output(“Folder “+path+”tidak ditemukan”)
    -> []

```

F15 - Save

SPESIFIKASI DAN DESKRIPSI

Program Save

{ Program menyimpan semua data player dan shop dari Array/List ke dalam sebuah Folder yang sudah ada atau Folder Baru }

KAMUS

{ Deklarasi Variabel }
path : string
{ Deklarasi Fungsi dan Prosedur }
procedure write(path,data: string)
KAMUS LOKAL

file : string

procedure
save(inventory,monster,user,monstershop,itemshop,monsterinv

```

    entory:array of array of string)
ALGORITMA
{ Realisasi Fungsi write(path,data) }
file <- open(path,'w')
row traversal in data
    file.write(','.join(map(str,row)) + '\n')
file.close()
output("CSV file "+path+" has been created successfully")
path <- "data/" + input("Masukkan nama folder: ")
output("Saving...")
if os.path.exists(path) then
    write_csv(path+"/item_inventory.csv",inventory)
    write_csv(path+"/user.csv",user)
    write_csv(path+"/monster.csv",monster)
    write_csv(path+"/monster_shop",monstershop)
    write_csv(path+"/item_shop",itemshop)
    write_csv(path+"/monster_inventory",monsterinventory)
else
    os.mkdir(path)
    write_csv(path+"/item_inventory.csv",inventory)
    write_csv(path+"/user.csv",user)
    write_csv(path+"/monster.csv",monster)
    write_csv(path+"/monster_shop",monstershop)
    write_csv(path+"/item_shop",itemshop)
    write_csv(path+"/monster_inventory",monsterinventory)

```

F16 - Exit

SPESIFIKASI DAN DESKRIPSI

Program Exit

{ Spesifikasi : Program untuk menutup keseluruhan program/game.
Menerima input berupa apakah ingin Save file sebelum mengakhiri
program atau tidak }

KAMUS

```

{ Deklarasi fungsi dan prosedur }
import Save
procedure exit (user, monsterinventory, itemshop, monster,
monstershop, iteminventory : array of array of string)

```

KAMUS LOKAL

```
    confirmExit : string
```

ALGORITMA

```

confirmExit <- input("Apakah Anda mau melakukan
penyimpanan file yang sudah diubah (y/n)")
while not confirmExit = "y" or confirmExit = "n"
do confirmExit <- input("Apakah Anda mau melakukan
penyimpanan file yang sudah diubah (y/n)")

if confirmExit = "y" then
Save
break
else { confirmExit = "n" }
break

```

B03 - Monster Ball

SPESIFIKASI DAN DESKRIPSI

function monsterball(monsterid, monsterlevel, userinventory,
monsterdata, monsterinventory, your monster data : array of
arrays) -> integer

KAMUS

```

jumlahball, chance, success, monsterlevel : integer
append(l : list, x : elmt type)
{ menambah elemen x ke list }

```

ALGORITMA

```

from src.RandomNumberGenerator import *
jumlahball <- int(userInventory[3][2])
chance <- RNG(1,100)
success <- 0
if jumlahball ≥ 1 then
    for element in yourmonsterdata
        if element = monsterdata[monsterid][1] then
            output("Anda sudah memiliki monster
tersebut")
            success <- 2
            -> success
            end
        else
            output("WHOOSH YOU THREW THE MONSTER
BALL")
            jumlahball <- jumlahball - 1
            userInventory[3][2] <- str(jumlahball)
            if monsterlevel = 1 then
                if chance ≥ 1 and chance ≤ 75 then
                    output("SUCCESS!!")

```

```

success <- 1
else
    output("THE MONSTER ESCAPED THE
BALL!!")
else if monsterlevel = 2
    if chance ≥ 1 and chance ≤ 50 then
        output("SUCCESS!!")
        success <- 1
    else
        output("THE MONSTER ESCAPED THE
BALL!!")
else if monsterlevel = 3
    if chance ≥ 1 and chance ≤ 25 then
        output("SUCCESS!!")
        success <- 1
    else
        output("THE MONSTER ESCAPED THE
BALL!!")
else if monsterlevel = 4
    if chance ≥ 1 and chance ≤ 10 then
        output("SUCCESS!!")
        success <- 1
    else
        output("THE MONSTER ESCAPED THE
BALL!!")
else if monsterlevel = 5
    if chance ≥ 1 and chance ≤ 5 then
        output("SUCCESS!!")
        success <- 1
    else
        output("THE MONSTER ESCAPED THE
BALL!!")
else
    output("YOU DON'T HAVE A MONSTER BALL")
    success <- 2
if success = 1 then
    append ([userinventory[0][0],
monsterdata[monsterid][1], str(monsterlevel)]) to
yourmonsterdata
    append ([userinventory[0][0],
monsterdata[monsterid][1], str(monsterlevel)]) to
monsterinventory
-> success

```

B04 - Jackpot

SPESIFIKASI DAN DESKRIPSI

Program Jackpot

{ Program memberikan 3 output random yang mana bila didapat 3 output yang sama, user akan mendapatkan hadiah }

KAMUS

{ Deklarasi Variabel }

{ Deklarasi fungsi/prosedur }

function prob(RNG: integer) -> integer

{ I.S. User menjalankan command jackpot, F.S. menghasilkan item apa yang muncul melalui program RNG }

KAMUS LOKAL

ALGORITMA

if 1 <= RNG <= 40 then

-> 0

if 41 <= RNG <= 65 then

-> 1

if 66 <= RNG <= 80 then

-> 2

if 81 <= RNG <= 95 then

-> 3

if 96 <= RNG <= 100 then

-> 4

function gacha(id, coin: integer, monsInv, yourMonsInv: array of array of string) -> coin: integer

{ I.S. User menjalankan command jackpot, F.S. menentukan apa yang didapat dari hasil random }

KAMUS LOKAL

RNG1, RNG2, RNG3, indexItem1, indexItem2, indexItem3, totalcoin: integer

icon: array of string

iconValue: array of integer

ALGORITMA

icon <- ['TOPI', 'PEDANG', 'BAJU', 'CELANA', 'JAM']

iconValue <- [60, 80, 120, 170, 300]

RNG1 <- RNG(1,100)

time.sleep(0.2)

```

RNG2 <- RNG(1,100)
time.sleep(0.1)
RNG3 <- RNG(1,100)
indexItem1 <- prob(RNG1)
indexItem2 <- prob(RNG2)
indexItem3 <- prob(RNG3)
totalcoin <- iconValue[indexItem1] +
iconValue[indexItem2] + iconValue[indexItem3]
output("=====")
output(""+icon[indexItem1]+" | "+icon[indexItem2]+" |
+icon[indexItem3]}+"")
output("=====")
if indexItem1 = indexItem2 and indexItem2 = indexItem3
then
    output("JACKPOT!! ANDA MENDAPATKAN GORLOCK THE
DESTROYER! DESTROY EVERYTHING!")
    if any(item[1] = "Gorlock" for item in monsInv) then
        output("GORLOCK SUDAH ANDA MILIKI, GORLOCK AKAN
DIUBAH KE O.W.C.A Coin sebanyak 2000 Coin")
        coin = coin + 2000
    else then
        append [str(id),"Gorlock",'1'] in monsInv
        append [str(id),"Gorlock",'1'] in yourMonsInv
    else then
        output("ANDA TIDAK DAPAT JACKPOT, TAPI ANDA DAPAT
"+totalcoin+" COIN !!!")
        coin = coin + totalcoin
-> coin

function jackpot(id, coin: integer, monsInv, yourMonsInv:
array of array of string) -> coin: integer
{ I.S. User menjalankan command jackpot, F.S. menentukan
bisa tidaknya jackpot dimainkan dan mengupdate coin}

```

KAMUS LOKAL

inp, confirm: string

ALGORITMA

```

coin = int(coin)
output("SELAMAT DATANG DI JACKPOT 888!!!")
output("ANDA DAPAT MENDAPATKAN GORLOCK THE DESTROYER
DENGAN HARGA 500 COIN")
output("COIN ANDA SAATINI SEJUMLAH "+coin+" ")
output("APAKAH ANDA INGIN MENCoba KEBERUNTUNGAN
ANDA??")

```

```

while true do
    input(inp)
    if inp = "y" then
        if confirm = "y" then
            if coin >= 500 then
                coin = gacha(id, coin, monsInv,
yourMonsInv)
                coin -= 500
                output("COIN ANDA TERSISA "+coin+"")
                output("APAKAH ANDA INGIN MELANJUTKAN??")
            else then
                output("YAH COIN ANDA TIDAK CUKUP")
                output("SILAKAN COBA JACKPOT DI WAKTU
LAIN")
                break
            else if confirm = "n" then
                -> coin
            else then
                output("Masukkan input yang benar")
            else if inp = "n" then
                -> coin
            else then
                output("Masukkan input yang benar")

```

ALGORITMA UTAMA

```

import time
from src.RandomNumberGenerator import RNG

```

Pengujian Program

F01 Register

Validasi jika username sudah ada

```

-->register
Masukkan Username :bagas
Username sudah terpakai
Masukkan Username :bagus
Username sudah terpakai
Masukkan Username :bagos
Username sudah terpakai
Masukkan Username :aburizalbakri
Masukkan password :■

```

Gambar 1.1 - Register: Validasi Username (1)

Validasi jika username mengandung unsur selain huruf, angka, strip dan underline

```
-->register
Masukkan Username :bagas^
Username hanya boleh mengandung huruf,angka,strip dan underscore
Masukkan Username :bagas=[_
Username hanya boleh mengandung huruf,angka,strip dan underscore
Masukkan Username :123!
Username hanya boleh mengandung huruf,angka,strip dan underscore
Masukkan Username :bagas123-__
Masukkan password :█
```

Gambar 1.2 - Register: Validasi Username (2)

Jika berhasil, pengguna akan memilih monster pertamanya

```
Masukkan Username :aburizalbakri
Masukkan password :lukis
PILIH MONSTER PERTAMA ANDA :
1. Godzilla
2. Mimic
3. Makima
--->1
SELAMAT DATANG AGENT BARU
```

Gambar 1.3 - Register: Validasi Username Berhasil

F02 Login

Jika username tidak ditemukan

```
-->login
username :aburizalbakri
password :12345
username tidak ditemukan
```

Gambar 2.1 - Login: Username tidak ditemukan

Jika password salah

```
-->login
username :bagas
password :password
password salah
```

Gambar 2.2 - Login: Password Salah

Jika login sukses

```
-->login
username :bagas
password :12345
Login berhasil
```

Gambar 2.3 - Login: Sukses

F03 Logout

Jika tidak jadi logout

```
Ketik 'Help' untuk mendapatkan petunjuk dari Almighty God
---> logout
Apakah anda yakin untuk logout?(Y/N)--> n
Ketik 'Help' untuk mendapatkan petunjuk dari Almighty God
```

Gambar 3.1 - Logout: Tidak jadi logout

Jika input bukan y/n

```
--> logout
Apakah anda yakin untuk logout?(Y/N)--> ok
Masukkan input yang benar
Apakah anda yakin untuk logout?(Y/N)--> w
```

Gambar 3.2 - Logout: Validasi, input buka y/n

Jika berhasil logout

```
Apakah anda yakin untuk logout?(Y/N)--> y
Logout berhasil
Ketik 'Help' untuk mendapatkan petunjuk dari Almighty God
```

Gambar 3.3 - Logout: Berhasil logout

F04 Menu and Help

Jika belum login

Gambar 4.1 - Menu dan Help: Tampilan sebelum login

Jika sudah login sebagai agent

```
| ! | -`-| : ""**" : [-`-] | | |
| [-`-] -`-| : H | -`-| -`-| |
| | H ::| |
| . H !|| |
| : H :|| |
| ! H !|| |
| | H |||
| | H |||
/_,'V.L|.
```

Gambar 4.2 - Menu dan Help: Saat login sebagai agent

Jika sudah login sebagai admin

Gambar 4.3 - Menu dan Help: Saat login sebagai admin

F07 Inventory

Jika ingin melihat list monster yg kita miliki

```
-->inventory
jumlah OWCA coin anda saat ini : 100000000000
Silakan pilih jenis item yang ingin diketahui Anda (Monster/Item/Back): Monster
1. Mimic
2. Makima
3. Jalangkung
```

Gambar 7.1 - Inventory: Melihat monster yang dimiliki

Jika ingin melihat list item yg kita miliki

```
--->inventory
jumlah OWCA coin anda saat ini : 100000000000
Silakan pilih jenis item yang ingin diketahui Anda (Monster/Item/Back): item
1. Healing Potion
2. Resilience Potion
3. Strength Potion
4. Monster Ball
Masukkan nomor item untuk menampilkan detail item (1/2/3/Back):
```

Gambar 7.2 - Inventory: Melihat item yang dimiliki

Jika ingin melihat detail monster

```
Masukkan nomor monster untuk menampilkan detail monster (1/2/dst/Back): 3
Nama      : Jalangkung
ATK Power : 4070.0
DEF Power : 13.2
HP        : 715.0
Level     : 3
Masukkan nomor monster untuk menampilkan detail monster (1/2/dst/Back): █
```

Gambar 7.3 - Inventory: Menampilkan monster

Jika ingin melihat detail item yang kita punya

```
Masukkan nomor item untuk menampilkan detail item (1/2/3/Back): 3
Type: Strength Potion
Quantity: 2
Masukkan nomor item untuk menampilkan detail item (1/2/3/Back): █
```

Gambar 7.4 - Inventory: Menampilkan detail item

Jika ingin keluar dari inventory

```
Silakan pilih jenis item yang ingin diketahui Anda (Monster/Item/Back): back
Sampai jumpa lagi !!
Type 'help' to seek wisdom from the ALMIGHT GOD
--->█
```

Gambar 7.5 - Inventory: Keluar dari inventory

Jika input selain monster,item dan back

```
--->inventory
jumlah OWCA coin anda saat ini : 100000000000
Silakan pilih jenis item yang ingin diketahui Anda (Monster/Item/Back): hehe
Mohon berikan input yang benar
```

Gambar 7.6 - Inventory: Validasi input (1), input selain monster, item, dan back

Jika id yg dimasukkan tidak terdapat pada list monster diinventory

```
Silakan pilih jenis item yang ingin diketahui Anda (Monster/Item/Back): monster
1. Mimic
2. Makima
3. Jalangkung
Masukkan nomor monster untuk menampilkan detail monster (1/2/dst/Back): 4
Nomor monster tersebut tidak tersedia
```

Gambar 7.7 - Inevntory: Validasi input(2), id tidak ada dalam list monster

Jika id item yang dimasukkan tidak terdapat pada list item diinventory

```
Silakan pilih jenis item yang ingin diketahui Anda (Monster/Item/Back): item
1. Healing Potion
2. Resilience Potion
3. Strength Potion
4. Monster Ball
Masukkan nomor item untuk menampilkan detail item (1/2/3/Back): 5
Nomor item tersebut tidak tersedia
```

Gambar 7.8 - Inventory: Validasi input (3), id tidak ada dalam list item

F08 Battle

Tampilan sebelum memulai battle

Gambar 8.1 - Battle: Tampilan awal battle

```
PROCEED? (Y/N)--->y
=====
>>>>>>>GODZILLA HAS ARRIVED !!!<<<<<<
=====
=====PLEASE CHOOSE A MONSTER=====
1. MIMIC
2. MAKIMA
3. JALANGKUNG
(1/2/3/dst)--->|
```

Gambar 8.2 - Battle: Tampilan memilih monster untuk battle

Jika monster yg dipilih tidak ada

```
=====
>>>>>>GIGIDI HAS ARRIVED !!!<<<<<<
=====
=====PLEASE CHOOSE A MONSTER=====
1. MIMIC
2. MAKIMA
3. JALANGKUNG
4. ROGA
(1/2/3/dst)--->5
pilihan ke-5 tidak tersedia
```

Gambar 8.3 - Battle: Monster yang dipilih tidak ada

tampilan untuk setiap turn agent

```
MAKIMA READY TO HELP
>>>>>>>>>>TURN 1<<<<<<<<<<
=====
ENEMY : GODZILLA
HEALTH : 702
ATTACK : 91
DEFENSE : 41
LEVEL : 4
=====
YOUR MONSTER : MAKIMA
HEALTH : 275
ATTACK : 148
DEFENSE : 5
LEVEL : 2
=====
CHOOSE AN ACTION :
1. ATTACK
2. DRINK POTION
3. MONSTER BALL
4. QUIT
=====
choose an action(1/2/3/4)--->|
```

Gambar 8.4 - Battle: Tampilan saat turn agen
t
tampilan jika memilih untuk attack dan musuh menyerang balik beserta perhitungannya

```
=====
>>>>>>>>>YOU ATTACKED !!!<<<<<<<<<
ATTACK DAMAGE : 156.88 (+6%)
REDUCED DAMAGE : 64.32079999999999 (41%)
TOTAL DAMAGE : 92.5592
ENEMY MONSTER HEALTH IS DOWN TO : 609
>>>>>>>>>Godzilla ATTACK BACK !!!<<<<<<<<
ATTACK DAMAGE : 65.52 (-28%)
REDUCED DAMAGE : 3.276 (5%)
TOTAL DAMAGE : 62.244
Makima HEALTH IS DOWN TO : 212
```

Gambar 8.5 - Battle: Tampilan saat agent attack dan musuh menyerang balik serta hitungannya

Jika memilih untuk meminum potion

```
choose an action(1/2/3/4)--->2
=====CHOOSE A POTION !=====
1. STRENGHT
    you have : 2
2. RESILIENSE
    you have : 3
3. HEAL
    you have : 2
4. CANCEL
(1/2/3/4)--->|
```

Gambar 8.6 - Battle: Tampilan saat agent memilih potion untuk diminum

jika meminum potion strength

```
POTION HAS BEEN CONSUMED, MAKIMA FEELS A BIT MORE STRONGER
>>>>>>>>>Godzilla ATTACK BACK !!!<<<<<<<<<
ATTACK DAMAGE : 80.99 (-11%)
REDUCED DAMAGE : 4.0495 (5%)
TOTAL DAMAGE : 76.9405
Makima HEALTH IS DOWN TO : 135
```

Gambar 8.7 - Battle: Tampilan saat meminum potion strength

jika meminum potion resilience

```
POTION HAS BEEN CONSUMED, MAKIMA FEELS A BIT MORE STURDIER
>>>>>>>>>Godzilla ATTACK BACK !!!<<<<<<<<
ATTACK DAMAGE : 70.07000000000001 (-23%)
REDUCED DAMAGE : 3.5035000000000007 (5%)
TOTAL DAMAGE : 66.5665
Makima HEALTH IS DOWN TO : 68
```

Gambar 8.8 - Battle: Tampilan saat agent meminum potion resilience

jika meminum healing potion

```
POTION HAS BEEN CONSUMED, MAKIMA REGENERATES SOME OF HIS HEALTH
>>>>>>>>>Godzilla ATTACK BACK !!!<<<<<<<<
ATTACK DAMAGE : 99.19000000000001 (+9%)
REDUCED DAMAGE : 4.9595000000000001 (5%)
TOTAL DAMAGE : 94.2305
Makima HEALTH IS DOWN TO : 0
```

Gambar 8.9 - Battle: Tampilan saat agent meminum healing potion

jika tidak memiliki potion

```
=====CHOOSE A POTION !=====
1. STRENGHT
    you have : 0
2. RESILIENSE
    you have : 2
3. HEAL
    you have : 1
4. CANCEL
(1/2/3/4)--->1
YOU DONT HAVE THE POTION
```

Gambar 8.10 - Battle: Tampilan saat agent tidak memiliki potion

jika potion sudah pernah diminum

```
=====CHOOSE A POTION !=====
1. STRENGHT
    you have : 0
2. RESILIENSE
    you have : 1
3. HEAL
    you have : 1
4. CANCEL
(1/2/3/4)--->2
POTION CAN ONLY BE USE ONCE
```

Gambar 8.11 - Battle: Tampilan saat potion sudah pernah diminum

jika memilih potion diluar opsi

```
=====CHOOSE A POTION !=====
1. STRENGHT
    you have : 0
2. RESILIENSE
    you have : 1
3. HEAL
    you have : 1
4. CANCEL
(1/2/3/4)--->5
Opsi 5 tidak ada
```

Gambar 8.12 - Battle: Validasi Input (1), saat potion tidak ada dalam opsi

jika memilih opsi quit

```
CHOOSE AN ACTION :
1. ATTACK
2. DRINK POTION
3. MONSTER BALL
4. QUIT
=====
choose an action(1/2/3/4)--->4
=====
YOU LEFT THE BATTLE FIELD
=====
```

Gambar 8.13 - Battle: Tampilan saat agent memilih opsi “QUIT”

```
kondisi kemenangan
>>>>>>>>>>>YOU ATTACKED !!!<<<<<<<<<
ATTACK DAMAGE : 5328.0 (+20%)
REDUCED DAMAGE : 319.68 (6%)
TOTAL DAMAGE : 5008.32
ENEMY MONSTER HEALTH IS DOWN TO : 0
=====
YOU DEFEATED YOUR ENEMY
WELL DONE
YOU GOT 9 COINS
=====
```

Gambar 8.14 - Battle: Tampilan saat agent menang

```
kondisi kekalahan
>>>>>>>>>Jalangkung ATTACK BACK !!!<<<<<<<
ATTACK DAMAGE : 3071.0 (-17%)
REDUCED DAMAGE : 153.55 (5%)
TOTAL DAMAGE : 2917.45
Makima HEALTH IS DOWN TO : 0
=====
YOU ARE DEAD
DONT GIVE UP
=====
```

Gambar 8.15 - Battle: Tampilan saat agent kalah

jika opsi yg dipilih tidak ada

```
CHOOSE AN ACTION :
1. ATTACK
2. DRINK POTION
3. MONSTER BALL
4. QUIT
=====
choose an action(1/2/3/4)--->5
OPTION 5 IS NOT AVAILABLE
```

Gambar 8.16 - Battle: Validasi input (2), saat opsi tidak ada pada opsi action

F09 Arena

Tampilan awal arena

```

--->arena

          T~~
          /"
        |' | T~~
T~~ |   Tr WWWW|
 | /"\ | | | |/\T~~
 /"\ WWW /"\ |' |WW|
WWWW\| / \|\|/\|/
| /_\V]WWW[\/_\WWWW
| " WWW'|I_I|'WWW'
| ' | / - \| ' | |
| . | [LI=H=LI] ' |
| ' | [L_] | ' |
| ' | [-[##]_-| ' |
|-----|/___\|-----|


=====WELCOME TO THE ARENA=====

ARENA MEKANISM :
1. THERE WILL BE A TOTAL OF 5 STAGES IN THE ARENA, EACH STAGE YOU WILL BE BATTLING
A RANDOMLY SELECTED MONSTER
2. YOUR ENEMY LEVEL WILL BE EQUAL TO THE STAGE NUMBER YOUR CURRENTLY ON
3. DEFEATING EACH STAGE WILL GIVE YOU A REWARD, THE HIGHER THE STAGE, THE
THE GREATER THE REWARD
4. IF YOU LOSE ONCE, THE ARENA WILL BE OVER
5. IF YOU WIN ALL THE WAY, YOU WILL GET A BONUS REWARD
6. POTIONS EFFECT ONLY LAST TILL THE END OF EACH STAGE
7. MONSTER BALL IS PROHIBITED IN THE ARENA

PROCEED? (Y/N)--->

```

Gambar 9.1 - Arena: Tampilan awal arena

tampilan memilih monster

```

PROCEED? (Y/N)--->y
=====PLEASE CHOOSE A MONSTER=====
1. MIMIC
2. MAKIMA
3. JALANGKUNG
4. ROGA
(1/2/3/dst)--->

```

Gambar 9.2 - Arena: Tampilan saat agent memilih monster dari inventory

jika monster yg dipilih tidak ada

```

=====PLEASE CHOOSE A MONSTER=====
1. MIMIC
2. MAKIMA
3. JALANGKUNG
4. ROGA
(1/2/3/dst)--->5
pilihan ke-5 tidak tersedia

```

Gambar 9.3 - Arena: Validasi input (1), saat monster yang dipilih tidak tersedia

tampilan untuk setiap stage

```
----->>>STAGE 1<<<-----  
=====  
>>>>>>MIMIC HAS ARRIVED !!!<<<<<<  
=====  
JALANGKUNG READY TO HELP  
>>>>>>>>>>>>TURN 1<<<<<<<<<<<<<<  
=====  
ENEMY : MIMIC  
HEALTH : 600  
ATTACK : 45  
DEFENSE : 28  
LEVEL : 1  
=====  
YOUR MONSTER : JALANGKUNG  
HEALTH : 780  
ATTACK : 4440  
DEFENSE : 14  
LEVEL : 3  
=====  
CHOOSE AN ACTION :  
1. ATTACK  
2. DRINK POTION  
3. QUIT  
=====  
choose an action(1/2/3/4)-->|
```

Gambar 9.4 - Arena: Tampilan setiap stage

kondisi kemenangan pada tiap stage

```
>>>>>>>>>>YOU ATTACKED !!!<<<<<<<<<<  
ATTACK DAMAGE : 3152.3999999999996 (-29%)  
REDUCED DAMAGE : 630.48 (20%)  
TOTAL DAMAGE : 2521.9199999999996  
ENEMY MONSTER HEALTH IS DOWN TO : 0  
=====  
YOU DEFEATED YOUR ENEMY  
WELL DONE  
=====  
STAGE CLEARED  
  
PREPARING NEXT STAGE
```

Gambar 9.5 - Arena: Tampilan kondisi kemanangan setiap stage

kondisi kemenangan saat berhasil mencapai stage terakhir

```
>>>>>>>>>YOU ATTACKED !!!<<<<<<<<<  
ATTACK DAMAGE : 3463.200000000003 (-22%)  
REDUCED DAMAGE : 969.696000000001 (28%)  
TOTAL DAMAGE : 2493.504  
ENEMY MONSTER HEALTH IS DOWN TO : 0  
=====  
YOU DEFEATED YOUR ENEMY  
WELL DONE  
=====  
STAGE CLEARED  
=====  
CONGRATULATION YOU MADE IT ALL THE WAY!!!!  
HERE A BONUS 50 COINS FOR YOU!!!  
=====
```

Gambar 9.6 - Arena: Tampilan berhasil mencapai stage akhir

kondisi kekalahan di setiap stage

```
>>>>>>>>>Malenia ATTACK BACK !!!<<<<<<<<  
ATTACK DAMAGE : 50.82 (-23%)  
REDUCED DAMAGE : 2.541000000000004 (5%)  
TOTAL DAMAGE : 48.27899999999996  
Makima HEALTH IS DOWN TO : 0  
=====  
YOU ARE DEAD  
DONT GIVE UP  
=====  
ARENA IS OVER, TRY AGAIN
```

Gambar 9.7 - Arena: Tampilan kondisi kekalahan tiap stage

kondisi jika memilih opsi quit disetiap stage

```
choose an action(1/2/3/4)--->3  
=====  
YOU LEFT THE BATTLE FIELD  
=====  
ARENA IS OVER, TRY AGAIN  
=====
```

Gambat 9.8 - Arena: Tampilan saat memilih opsi quit di setiap stage

menampilkan statistik di akhir sesi arena

```
=====
STATISTICS FROM THE ARENA :
NUMBER OF STAGE : 1
TOTAL COIN GAINED FROM THE ARENA : 20
TOTAL DAMAGE TAKEN : 440
TOTAL DAMAGE DEALT : 1156
=====
```

Gambar 9.9 - Arena: Tampilan statistik akhir sesi arena

F10 Shop and Currency



Gambar 10.1 - Shop and C. : Tampilan awal

jika pilih lihat monster

```
>>> Pilih aksi (lihat/beli/keluar): lihat
>>> Mau lihat apa? (monster/item): monster
ID | Type      | ATK Power | DEF Power | HP   | Stok | Harga
1  | Godzilla  | 70        | 32        | 540  | 6    | 800
2  | Mimic     | 45        | 28        | 600  | 13   | 300
3  | Makima    | 135       | 5         | 250  | 2    | 750
>>> Pilih aksi (lihat/beli/keluar): |
```

Gambar 10.2 - Shop and C. : Tampilan saat pilih lihat monster

jika pilih lihat item

```

>>> Pilih aksi (lihat/beli/keluar): lihat
>>> Mau lihat apa? (monster/item): item
ID | Type           | Stok | Harga
1  | Healing Potion | 5    | 25
2  | Resilience Potion | 5    | 30
3  | Strength Potion | 5    | 40
4  | Monster Ball   | 5    | 100
>>> Pilih aksi (lihat/beli/keluar): 

```

Gambar 10.3 - Shop and C. : Tampilan saat pilih lihat item

tampilan berhasil membeli monster

```

>>> Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A Coin-mu sekarang 100000000000.

>>> Mau beli apa? (monster/item): monster
>>> Masukkan id monster: 1
monster berhasil ditambahkan

```

Gambar 10.4 - Shop and C. : Tampilan saat berhasil membeli mosnter

jika monster sudah dimiliki

```

>>> Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A Coin-mu sekarang 99999999200

>>> Mau beli apa? (monster/item): monster
>>> Masukkan id monster: 1
Monster sudah dimiliki

```

Gambar 10.5 - Shop and C. : Tampilan saat mosnter yang dibeli sudah dimiliki

jika id monster tidak ada di shop

```

>>> Masukkan id monster: 5
Id monster tersebut tidak tersedia

```

Gambar 10.6 - Shop and C. : Tampilan saat id monster tidak ada di shop

jika coin kurang

```

>>> Mau beli apa? (monster/item): monster
>>> Masukkan id monster: 1
Maaf O.W.C.A. Coin-mu kurang.

```

Gambar 10.7 - Shop and C. : Tampilan saat agent kekurangan coin

```
kondisi jika berhasil membeli item  
>>> Pilih aksi (lihat/beli/keluar): beli  
Jumlah O.W.C.A Coin-mu sekarang 100.  
  
>>> Mau beli apa? (monster/item): item  
>>> Masukkan id item: 1  
>>> Masukkan jumlah: 2  
Berhasil membeli item: 2 Healing Potion. Item sudah masuk ke inventory-mu!
```

Gambar 10.8 - Shop and C. : Tampilan saat berhasil membeli item

```
jika stok tidak cukup  
>>> Mau beli apa? (monster/item): item  
>>> Masukkan id item: 1  
>>> Masukkan jumlah: 6  
Stock tidak cukup
```

Gambar 10.9 - Shop and C. : Tampilan saat stok tidak cukup

```
jika stok habis  
>>> Masukkan id item: 1  
>>> Masukkan jumlah: 1  
Stock Potion habis!
```

Gambar 10.10 - Shop and C. : Tampilan saat stok habis

```
jika id item tidak ada  
>>> Mau beli apa? (monster/item): item  
>>> Masukkan id item: 6  
Id tersebut tidak tersedia
```

Gambar 10.11 - Shop and C. : Tampilan saat id item tidak ada

```
jika pilih keluar shop  
>>> Pilih aksi (lihat/beli/keluar): keluar  
Sampai bertemu lagi, User!
```

Gambar 10.12 - Shop and C. : Tampilan saat player memilih untuk keluar shop

F11 Laboratory

tampilan awal laboratory

```
--> laboratory
JUMLAH COIN ANDA : 99999999875
===== MONSTER LIST =====
1. Mimic LEVEL: 1
2. Makima LEVEL: 2
3. Jalangkung LEVEL: 3
===== UPGRADE PRICE =====
1. Level 1 --> Level 2: 100
2. Level 2 --> Level 3: 200
3. Level 3 --> Level 4: 400
4. Level 4 --> Level 5: 700
Pilih monster yang ingin di-upgrade (1/2/dst/Back): █
```

Gambar 11.1 - Lab: Tampilan awal

jika id monster yg ingin di upgrade tidak ada

```
JUMLAH COIN ANDA : 100000000000
===== MONSTER LIST =====
1. Mimic LEVEL: 1
2. Makima LEVEL: 2
3. Jalangkung LEVEL: 3
===== UPGRADE PRICE =====
1. Level 1 --> Level 2: 100
2. Level 2 --> Level 3: 200
3. Level 3 --> Level 4: 400
4. Level 4 --> Level 5: 700
Pilih monster yang ingin di-upgrade (1/2/dst/Back): 0
Masukkan nomor monster yang benar
```

Gambar 11.2 - Lab: Tampilan saat id monster yang ingin diupgrade tidak tersedia

jika sukses upgrade monster

```
Pilih monster yang ingin di-upgrade (1/2/dst/Back): 2
Silakan pilih ke level berapa kah Anda ingin meng-upgrade monster Anda (1/2/dst/Back): 4
Makima akan di-upgrade ke level 4
Harga untuk meng-upgrade Makima adalah 600
Lanjutkan upgrade (Y/N): y
Selamat! Makima berhasil di-upgrade ke level 4!
Coin Anda tersisa 99999999400
Silakan pilih ke level berapa kah Anda ingin meng-upgrade monster Anda (1/2/dst/Back): █
```

Gambar 11.3 - Lab: Tampilan saat sukses upgrade monster

saat input bukan y/n

```
Lanjutkan upgrade (Y/N): k
Mohon masukkan input yang benar

Lanjutkan upgrade (Y/N): █
```

Gambar 11.4 - Lab: Validasi (1), Tampilan saat input bukan y/n

jika level sudah dicapai

```
Pilih monster yang ingin di-upgrade (1/2/dst/Back): 2
Silakan pilih ke level berapa kah Anda ingin meng-upgrade monster Anda (1/2/dst/Back): 4
Masa mau upgrade ke level yg sama/lebih rendah sih.. Itu mah downgrade kak
```

Gambar 11.5 - Lab: Validasi (2), Tampilan saat level upgrade sudah tercapai

jika level yg dituju diatas 5

```
Silakan pilih ke level berapa kah Anda ingin meng-upgrade monster Anda (1/2/dst/Back): 6
Upgrade cuman sampai level 5 kakak, level di atas 5 itu level kesulitan tubes daspro
```

Gambar 11.6 - Lab: Validasi (3), level yang dituju lebih dari 5

jika input level bukan integer

```
Silakan pilih ke level berapa kah Anda ingin meng-upgrade monster Anda (1/2/dst/Back): iiwjie
Masukkan input level yang benar
```

Gambar 11.7 - Lab: Validasi (4), input level bukan integer

jika coin tidak cukup

```
Pilih monster yang ingin di-upgrade (1/2/dst/Back): 2
Silakan pilih ke level berapa kah Anda ingin meng-upgrade monster Anda (1/2/dst/Back): 5
Mimic akan di-upgrade ke level 5
Harga untuk meng-upgrade Mimic adalah 1400
Lanjutkan upgrade (Y/N): y
Maaf! Anda tidak memiliki Coin yang cukup
```

Gambar 11.8 - Lab: Coin tidak cukup

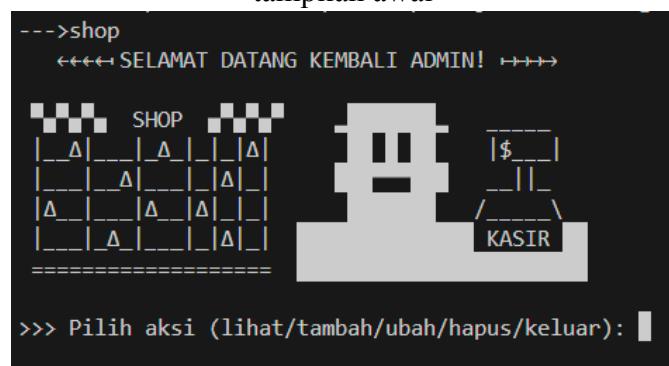
jika keluar dari laboratory

```
Pilih monster yang ingin di-upgrade (1/2/dst/Back): back
Sampai bertemu lagi !
```

Gambar 11.9 - Lab: Tampilan keluar dari laboratory

F12 Shop Management

tampilan awal



Gambar 12.1 - Shop M. : Tampilan awal

tampilan lihat monster

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): lihat
>>> Mau lihat apa? (monster/item): monster
ID | Type          | ATK Power | DEF Power | HP      | Stok   | Harga
1  | Godzilla     | 70        | 32        | 540     | 6       | 800
2  | Mimic         | 45        | 28        | 600     | 13      | 300
3  | Makima        | 135       | 5         | 250     | 2       | 750
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar):
```

Gambar 12.2 - Shop M. : Tampilan lihat monster

tampilan lihat item

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): lihat
>>> Mau lihat apa? (monster/item): item
ID | Type          | Stok   | Harga
1  | Healing Potion | 5      | 25
2  | Resilience Potion | 5    | 30
3  | Strength Potion | 5    | 40
4  | Monster Ball | 5    | 100
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar):
```

Gambar 12.3 - Shop M. : Tampilan lihat item

tampilan tambah monster

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): tambah
Mau tambah apa? (monster/item): monster
ID | Type          | ATK Power | DEF Power | HP      |
4  | Jalangkung    | 3700     | 12        | 650
5  | gigidi        | 54        | 19        | 420
6  | Gorlock       | 22        | 20        | 700
7  | Bunny          | 65        | 24        | 550
8  | Roga           | 88        | 8         | 222
9  | Molded         | 66        | 24        | 440
10 | Mohg           | 50        | 20        | 600
11 | Malenia        | 55        | 30        | 500
12 | Kraken          | 20        | 35        | 570
13 | Mothra          | 57        | 21        | 520
14 | Parkinson       | 44        | 31        | 470
15 | Ifrit           | 90        | 5         | 330
>>> Masukkan id monster: 15
```

Gambar 12.4 - Shop M. : Tampilan tambah monster

jika monster berhasil ditambah

```
>>> Masukkan id monster: 15
>>> Masukkan stok awal: 111
>>> Masukkan harga: 111
Ifrit telah berhasil ditambahkan ke dalam Shop!
```

Gambar 12.5 - Shop M. : Tampilan saat monster berhasil ditambah

jika sudah ada dishop
>>> Masukkan id monster: 15
Monster sudah ada di Shop!

Gambar 12.6 - Shop M. : Tampilan saat berhasil menambah monster

jika id monster yang ingin ditambah tidak ada dipilihan
>>> Masukkan id monster: 17
ID tidak valid! Pilih ID yang terdapat pada tabel!

Gambar 12.7 - Shop M. : Validasi (1), Tampilan saat id monster yang ingin ditambah tidak ada pada file monster

jika stok awal dan harga bukan integer
>>> Masukkan stok awal: oo
masukkan integer yg benar
Masukkan stok awal: 13
>>> Masukkan harga: pp
masukkan integer yg benar

Gambar 12.8 - Shop M. : Validasi (2), saat stok awal dan harga bukan integer

tampilan tambah item
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): tambah
Mau tambah apa? (monster/item): item
ID | Type
2 | Resilience Potion
>>> Masukkan id item (1-4):

Gambar 12.9 - Shop M. : Tampilan saat ingin menambah item

jika berhasil menambahkan item
>>> Masukkan id item (1-4): 2
Masukkan stok awal: 111
Masukkan harga: 111
Resilience Potion telah berhasil ditambahkan ke dalam Shop!

Gambar 12.10 - Shop M. : Tampilan saat berhasil menambahkan item

jika item yg ingin ditambah sudah ada dishop
>>> Masukkan id item (1-4): 3
Item sudah terdapat di shop

Gambar 12.11 - Shop M. : Tampilan saat item berhasil ditambah ke shop

```
jika id item diluar jangkauan  
>>> Masukkan id item (1-4): 5  
id tidak valid!
```

Gambar 12.12 - Shop M. : Validasi (3), saat id item diluar jangkauan

```
jika bukan integer  
>>> Masukkan id item (1-4): fiufe  
Masukkan integer yang benar
```

Gambar 12.13 - Shop M. : Validasi (4), saat id item bukan integer

```
kondisi berhasil merubah stok dan harga monster  
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah  
Mau ubah apa? (monster/item): monster  
>>> Masukkan id monster: 2  
>>> Masukkan stok baru: 111  
>>> Masukkan harga baru: 111  
Mimic telah berhasil diubah dengan stok baru sejumlah 111 dan dengan harga baru 111!  
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar):
```

Gambar 12.14 - Shop M. : Berhasil mengubah stok dan harga monster

```
jika yg berubah hanya stok  
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah  
Mau ubah apa? (monster/item): monster  
>>> Masukkan id monster: 2  
>>> Masukkan stok baru: 111  
>>> Masukkan harga baru:  
Mimic telah berhasil diubah dengan stok baru sejumlah 111!
```

Gambar 12.15 - Shop M. : Hanya mengubah stock

```
jika yang berubah hanya harga  
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah  
Mau ubah apa? (monster/item): monster  
>>> Masukkan id monster: 2  
>>> Masukkan stok baru:  
>>> Masukkan harga baru: 111  
Mimic telah berhasil diubah dengan harga baru 111!  
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar):
```

Gambar 12.16 - Shop M. : Hanya mengubah harga

jika id monster yg ingin diubah tidak ada

```
>>> Masukkan id monster: 5  
ID tersebut tidak tersedia  
>>> Masukkan id monster: aijs  
ID tersebut tidak tersedia  
>>> Masukkan id monster: qis  
ID tersebut tidak tersedia
```

Gambar 12.17 - Shop M. : Id monster yang ingin diubah tidak ada

jika stok baru atau harga baru bukan integer

```
>>> Masukkan stok baru: errt  
Masukkan integer yg benar  
Masukkan stok baru:12  
>>> Masukkan harga baru: rt  
Masukkan integer yg benar
```

Gambar 12.18 - Shop M. : Validasi (5), stok dan harga baru bukan integer

jika berhasil merubah item di shop

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah  
Mau ubah apa? (monster/item): item  
>>> Masukkan id item: 1  
>>> Masukkan stok baru: 111  
>>> Masukkan harga baru:  
Healing Potion telah berhasil diubah dengan stok baru sejumlah 111 dan dengan harga baru 111!
```

Gambar 12.19 - Shop M. : Berhasil mengubah item di shop

jika yg diubah stok saja

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah  
Mau ubah apa? (monster/item): item  
>>> Masukkan id item: 1  
>>> Masukkan stok baru: 111  
>>> Masukkan harga baru:  
Healing Potion telah berhasil diubah dengan stok baru sejumlah 111!
```

Gambar 12.20 - Shop M. : mengubah stok saja

jika yang diubah harga saja

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah  
Mau ubah apa? (monster/item): item  
>>> Masukkan id item: 1  
>>> Masukkan stok baru:  
>>> Masukkan harga baru: 111  
Healing Potion telah berhasil diubah dengan harga baru 111!
```

Gambar 12.21 - Shop M. : mengubah harga saja

jika id item yg ingin diubah tidak ada

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): ubah  
Mau ubah apa? (monster/item): item  
>>> Masukkan id item: qwe  
Masukkan integer yang benar  
>>> Masukkan id item: 5  
Id tersebut tidak tersedia
```

Gambar 12.22 - Validasi (6), id item yang ingin diubah tidak tersedia

jika stok atau harga baru bukan integer

```
>>> Masukkan stok baru: dijie  
Masukkan integer yg benar  
Masukkan stok baru:100  
>>> Masukkan harga baru: isis  
Masukkan integer yg benar
```

Gambar 12.23 - Shop M. : Validasi (7), stok atau harga bukan integer

kondisi jika berhasil menghapus monster dari shop

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus  
>>> Mau hapus apa? (monster/item): monster  
>>> Masukkan id monster: 2  
Apakah anda yakin ingin menghapus Mimic dari Shop (y/n)? y  
Mimic telah berhasil dihapus dari Shop!
```

Gambar 12.24 - Shop M. : Berhasil menghapus monster dari shop

jika penghapusan dibatalkan

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus  
>>> Mau hapus apa? (monster/item): monster  
>>> Masukkan id monster: 1  
Apakah anda yakin ingin menghapus Godzilla dari Shop (y/n)? n  
Penghapusan dibatalkan!
```

Gambar 12.25 - Shop M. : Penghapusan dibatalkan

jika bukan y/n

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus
>>> Mau hapus apa? (monster/item): monster
>>> Masukkan id monster: 1
Apakah anda yakin ingin menghapus Godzilla dari Shop (y/n)? ok
Masukkan Y/N
```

Gambar 12.26 - Shop M. : Validasi (7) input bukan y/n

jika id monster yg ingin dihapus tidak ada

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus
>>> Mau hapus apa? (monster/item): monster
>>> Masukkan id monster: 2
Id tersebut tidak tersedia
>>> Masukkan id monster: ok
Id tersebut tidak tersedia
```

Gambar 12.27 - Shop M. : Validasi (8), id monster yang ingin dihapus tidak ada

kondisi jika berhasil hapus item dari shop

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus
>>> Mau hapus apa? (monster/item): item
>>> Masukkan id item: 1
Apakah anda yakin ingin menghapus Healing Potion dari Shop (y/n)? y
Healing Potion telah berhasil dihapus dari Shop!
```

Gambar 12.28 - Shop M. : Berhasil hapus item

jika penghapusan dibatalkan

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus
>>> Mau hapus apa? (monster/item): item
>>> Masukkan id item: 2
Apakah anda yakin ingin menghapus Resilience Potion dari Shop (y/n)? n
Penghapusan dibatalkan!
```

Gambar 12.29 - Shop M. : Penghapusan item dibatalkan

jika id item yang ingin dihapus tidak ada

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): hapus
>>> Mau hapus apa? (monster/item): item
>>> Masukkan id item: 1
Id tersebut tidak tersedia
>>> Masukkan id item: ok
Masukkan integer yang benar
```

Gambar 12.30 - Shop M. : Validasi (9), id item yang ingin dihapus tidak ada

jika keluar dari shop management

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): keluar  
Sampai bertemu lagi Admin!
```

Gambar 12.31 - Shop M. : Keluar dari shop management

jika aksi tidak ada

```
>>> Pilih aksi (lihat/tambah/ubah/hapus/keluar): capekgwaslidah  
aksi tidak diketahui
```

Gamber 12.32 - Shop M. : Validasi (10), aksi tidak ada

F13 Monster Management

tampilan awal

```
--->monster  
  
SELAMAT DATANG DI DATABASE PARA MONSTER !!!  
1. Tampilkan Semua Monster  
2. Tambah Monster Baru  
( Ketik nomor untuk aksi yang diinginkan, ketik 0 untuk keluar )  
  
Pilih Aksi: |
```

Gambar 13.1 - Monster M. : Tampilan awal

tampilan jika aksi 1

Pilih Aksi: 1						
ID	Type	ATK Power	DEF Power	HP		
1	Godzilla	70	32	540		
2	Mimic	45	28	600		
3	Makima	135	5	250		
4	Jalangkung	3700	12	650		
5	gigidi	54	19	420		
6	Gorlock	22	20	700		
7	Bunny	65	24	550		
8	Roga	88	8	222		
9	Molded	66	24	440		
10	Mohg	50	20	600		
11	Malenia	55	30	500		
12	Kraken	20	35	570		
13	Mothra	57	21	520		
14	Parkinson	44	31	470		
15	Ifrit	90	5	330		

Ketik apapun untuk balik ke menu: |

Gamber 13.2 - Monster. M : Memilih aksi melihat tabel

```
balik ketampilan awal
Ketik apapun untuk balik ke menu: bisagilagw

SELAMAT DATANG DI DATABASE PARA MONSTER !!!
1. Tampilkan Semua Monster
2. Tambah Monster Baru
( Ketik nomor untuk aksi yang diinginkan, ketik 0 untuk keluar )
```

Gamber 13.3 - Monster M. : Balik tampilan awal

jika aksi 2 dan berhasil menambahkan monster baru

```
Pilih Aksi: 2
Masukkan Type Monster : Tubesasik
Masukkan attack power : 99999999
Masukkan DEF power (0-50) : 50
Masukkan hp : 1000000000
Apakah mau tambahkan monster? (Y/N): y
Monster berhasil ditambahkan
```

Gamber 13.4 - Monster M. : Memilih aksi menambah monster dan berhasil

jika nama sudah ada

```
Pilih Aksi: 2
Masukkan Type Monster : Jalangkung
Nama sudah terdaftar. Coba lagi!
```

Gamber 13.5 - Monster M. : Nama monster sudah ada pada list

jika attack minus

```
Pilih Aksi: 2
Masukkan Type Monster : tubesgacor
Masukkan attack power : -99
Masa attack mines yg bener aje
```

Gamber 13.6 - Monster M. : Validasi (1), saat input attack negative

jika hp mines

```
Masukkan hp : -99
masa hp mines yang bener aje
```

Gamber 13.7 - Monster M. : Validasi (2), input hp negative

jika defense diluar jangkauan

```
Masukkan DEF power (0-50) : 60
Masukkan harus bernilai 0-50
Masukkan DEF power (0-50) : -3
Masukkan harus bernilai 0-50
```

Gambar 13.8 - Monster M. : Validasi (3), defense diluar jangkauan

jika stat bukan integer

```
Masukkan attack power : sigma
Masukkan bukan bertipe integer, coba lagi!
Masukkan attack power : 100
Masukkan DEF power (0-50) : ohio
Masukkan bukan bertipe integer, coba lagi!
Masukkan DEF power (0-50) : 50
Masukkan hp : gyatt
Masukkan bukan bertipe integer, coba lagi!
```

Gambar 13.9 - Monster M. : Validasi (4), input bukan integer

jika mosnter tidak jadi ditambah

```
Apakah mau tambahkan monster? (Y/N): n
Monster gagal ditambahkan
```

Gambar 13.10 - Monster M. : Monster tidak jadi ditambahkan

jika bukan y/n

```
Apakah mau tambahkan monster? (Y/N): ok
input tidak valid, hanya menerima input Y/N
(Y/N):
```

Gamber 13.11 - Monster M. : Validasi (5), input bukan y/n

jika aksi tidak diketahui

```
SELAMAT DATANG DI DATABASE PARA MONSTER !!!
1. Tampilkan Semua Monster
2. Tambah Monster Baru
( Ketik nomor untuk aksi yang diinginkan, ketik 0 untuk keluar )

Pilih Aksi: ok
Input tidak valid, input yang diterima hanya 0/1/2
```

Gambar 13.12 - Monster M. Validasi (6), input aksi tidak sesuai

jika keluar dari monster management

```
Pilih Aksi: 0  
Terima kasih telah mengunjungi database monster
```

Gambar 13.13 - Monster M. : Keluar dari monster management

F14 Load

jika tidak ada data yang diload

```
Tidak ada nama folder yang diberikan!  
  
usage: main.py <nama_folder>  
PS C:\Users\USER\Documents\debuging\if1210-2024-tubes-k08-b>
```

Gambar 14.1 - Load: tidak ada data

jika folder tidak ditemukan

```
PS C:\Users\USER\Documents\debuging\if1210-2024-tubes-k08-b> python main.py data5  
Folder "data/data5" tidak ditemukan.
```

Gambar 14.2 - Load: folder tidak ditemukan

jika load berhasil

```
PS C:\Users\USER\Documents\debuging\if1210-2024-tubes-k08-b> python main.py data1  
Loading...  
Selamat datang  
Ketik 'Help' untuk mendapatkan petunjuk dari Almighty God  
---> |
```

Gambar 14.3 - Load: Berhasil

F15 Save

jika membuat folder baru

```
---> save  
Masukkan nama folder: data4  
Saving...  
Membuat folder baru data/data4  
Berhasil melakukan Save pada folder data/data4  
---> |
```

Gambar 15.1 - Save: Membuat folder baru

jika folder sudah ada

```
---> save  
Masukkan nama folder: data4  
Saving...  
Berhasil melakukan Save pada folder data/data4
```

Gambar 15.2 - Save: Folder sudah ada data

F16 Exit

jika exit program tanpa save

```
---> exit  
Apakah Anda mau melakukan penyimpanan file yang sudah diubah (y/n)  
Program berhasil ditutup!
```

Gambar 16.1 - Exit: exit program tanpa save

Jika bukan y/n

```
---> exit  
Apakah Anda mau melakukan penyimpanan file yang sudah diubah (y/n) :ok  
Masukkan y/n saja :ok  
Masukkan y/n saja :■
```

Gambar 16.2 - Exit: Validasi (1), input bukan y/n

jika exit program dengan save

```
---> exit  
Apakah Anda mau melakukan penyimpanan file yang sudah diubah (y/n) :y  
Masukkan nama folder: data1  
Saving...  
Berhasil melakukan Save pada folder data/data1  
Program berhasil disave dan ditutup!
```

Gambar 16.3 - Exit: exit program dengan save

B03 Monster Ball

jika berhasil menangkap monster musuh saat battle

```
CHOOSE AN ACTION :  
1. ATTACK  
2. DRINK POTION  
3. MONSTER BALL  
4. QUIT  
=====choose an action(1/2/3/4)--->3  
WHOOSSHHH YOU THREW THE MONSTER BALL  
SUCCESS!!  
BUNNY IS ADDED TO YOUR INVENTORY
```

Gamber B03.1 - Monster ball: Berhasil menangkap monster

jika gagal menangkap monster musuh

```
CHOOSE AN ACTION :  
1. ATTACK  
2. DRINK POTION  
3. MONSTER BALL  
4. QUIT  
=====choose an action(1/2/3/4)--->3  
WHOOSSHHH YOU THREW THE MONSTER BALL  
THE MONSTER ESCAPED THE BALL!!
```

Gamber B03.2 - Monster ball: gagal menangkap monster

jika tidak memiliki monster ball

```
CHOOSE AN ACTION :  
1. ATTACK  
2. DRINK POTION  
3. MONSTER BALL  
4. QUIT  
=====choose an action(1/2/3/4)--->3  
YOU DONT HAVE A MONSTER BALL
```

Gamber B03.3 - Monster ball: tidak memiliki monster ball dalam inventory

B04 Jackpot

tampilan awal

```
---> jackpot
SELAMAT DATANG DI JACKPOT 888!!!
ANDA DAPAT MENDAPATKAN GORLOCK THE DESTROYER DENGAN HARGA 500 COIN
COIN ANDA SAAT INI SEJUMLAH 100000000000
APAKAH ANDA INGIN MENCoba KEBERUNTUNGAN ANDA???
(Y/N)----->
```

Gambar B04.1 - Jackpot: Tampilan awal

jika kurang beruntung

```
APAKAH ANDA INGIN MENCoba KEBERUNTUNGAN ANDA???
(Y/N)----->y
APAKAH ANDA YAKIN? (Y/N) :y
=====
TOPI | CELANA | TOPI
=====
ANDA TIDAK DAPAT JACKPOT, TAPI ANDA DAPAT 290 COIN !!!
COIN ANDA TERSISA 9999999790
APAKAH ANDA INGIN MELANJUTKAN?? (Y/N)----->
```

Gambar B04.2 - Jackpot: Kurang beruntung

jika hoki

```
(Y/N)----->y
APAKAH ANDA YAKIN? (Y/N) :y
TOPI | TOPI | TOPI
SELAMAT!! ANDA MENDAPATKAN GORLOCK THE DESTROYER! DESTROYER
COIN ANDA TERSISA 9999999500
APAKAH ANDA INGIN MELANJUTKAN?? (Y/N)----->
```

Gambar B04.3 - Jackpot : Beruntung!!

jika tidak yakin

```
APAKAH ANDA INGIN MENCoba KEBERUNTUNGAN ANDA???
(Y/N)----->y
APAKAH ANDA YAKIN? (Y/N) :n
Sampai bertemu lagi :)
```

Gamber B04.4 - Jackpot: Player tidak yakin

jika tidak ingin melanjutkan

```
COIN ANDA TERSISA 99999999790
APAKAH ANDA INGIN MELANJUTKAN?? (Y/N)---->n
Sampai bertemu lagi :)
```

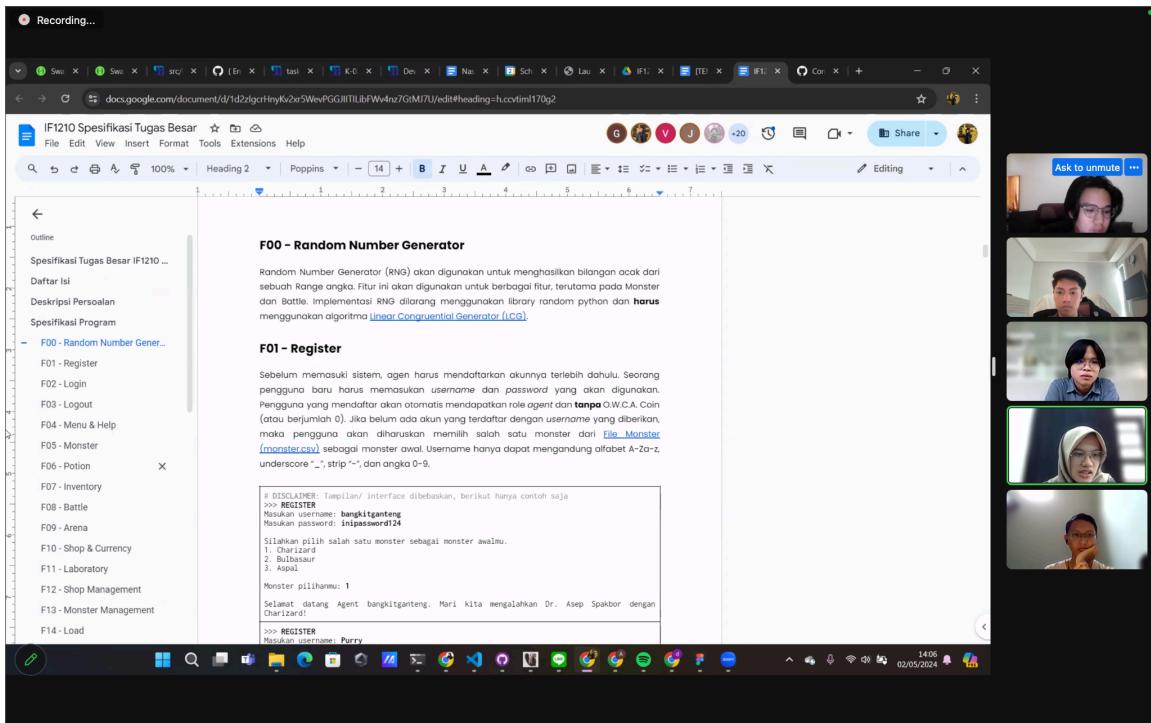
Gambar B04.5 - Jackpot: Player tidak ingin melanjutkan jackpot

jika selain y/n

```
APAKAH ANDA INGIN MENCoba KEBERUNTUNGAN ANDA???
(Y/N)---->ok
Masukkan input yang benar
(Y/N)---->y
APAKAH ANDA YAKIN? (Y/N) :ok
Masukkan input yang benar
(Y/N)---->
```

Gambar B04.6 - Jackpot: Validasi, input bukan y/n

Lampiran



Gambar Lampiran 1 - Asistensi 1

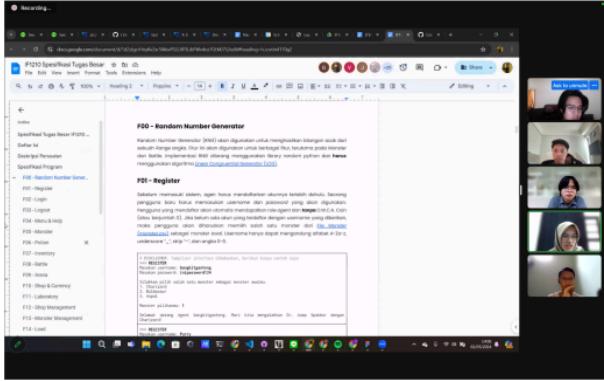


Gambar Lampiran 2 - Asistensi 2

Form MoM Asistensi Tugas Besar
IF1210/Dasar Pemrograman
Sem. 2 2023/2024

Nomor Asistensi	:	1														
No. Kelompok/Kelas	:	B/K-08														
Tanggal asistensi	:	2 Mei 2024														
Anggota kelompok	<table border="1"> <thead> <tr> <th colspan="2">NIM / Nama (Hanya yang Hadir)</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>19623028 / Muhammad Refino Ramadhan</td> </tr> <tr> <td>2</td> <td>19623168 / Darren Mansyl</td> </tr> <tr> <td>3</td> <td>19623238 / Favian Rafi Laftiyanto</td> </tr> <tr> <td>4</td> <td>19623278 / Mahesa Satria Prayata</td> </tr> <tr> <td>5</td> <td></td> </tr> <tr> <td>6</td> <td></td> </tr> </tbody> </table>		NIM / Nama (Hanya yang Hadir)		1	19623028 / Muhammad Refino Ramadhan	2	19623168 / Darren Mansyl	3	19623238 / Favian Rafi Laftiyanto	4	19623278 / Mahesa Satria Prayata	5		6	
NIM / Nama (Hanya yang Hadir)																
1	19623028 / Muhammad Refino Ramadhan															
2	19623168 / Darren Mansyl															
3	19623238 / Favian Rafi Laftiyanto															
4	19623278 / Mahesa Satria Prayata															
5																
6																
Asisten pembimbing	<table border="1"> <thead> <tr> <th colspan="2">NIM / Nama</th> </tr> </thead> <tbody> <tr> <td colspan="2">13521129/Chiquita Ahsanunnisa</td> </tr> </tbody> </table>		NIM / Nama		13521129/Chiquita Ahsanunnisa											
NIM / Nama																
13521129/Chiquita Ahsanunnisa																

Catatan Asistensi:

Rangkuman Diskusi
Masing-masing mahasiswa memaparkan progressnya. Terdapat diskusi spesifik mengenai cara membaca file .csv atau dalam pembuatan csv Parser/csv reader.
Tindak Lanjut
Asisten Praktikum menjelaskan alternatif yang dapat digunakan untuk membuat fungsi yang dapat membaca file .csv, selain itu asisten praktikum juga membahas kembali spesifikasi wajib dari tugas besar
Dokumentasi


Keterangan : Abu Dzar Al-Ghfari dengan NIM 16523058 tidak hadir karena sedang melaksanakan Praktikum Kimia pada saat yang bersamaan

Gambar Lampiran 4 - MoM Asistensi 1

Form MoM Asistensi Tugas Besar
IF1210/Dasar Pemrograman
Sem. 2 2023/2024

Nomor Asistensi : 2
No. Kelompok/Kelas : B/K-08
Tanggal asistensi : 10 Mei 2024

Anggota kelompok

NIM / Nama (Hanya yang Hadir)	
1	<u>16523058 / Abu Dzar Al-Ghfari</u>
2	<u>19623028 / Muhammad Refino Ramadhan</u>
3	<u>19623168 / Darren Mansyl</u>
4	<u>19623238 / Favian Rafi Laftiyanto</u>
5	<u>19623278 / Mahesa Satria Prayata</u>
6	

Asisten pembimbing

NIM / Nama
<u>13521129/Chiquita Ahsanunnisa</u>

Catatan Asistensi:

Rangkuman Diskusi

Terdapat kesalahan looping pada fungsi register yang membuat fungsi tersebut tidak bisa dijalankan secara optimal. Selain itu, ada juga miskonsepsi terkait penyimpanan data baru pada fitur Shop (data baru seharusnya disimpan pada array, bukan secara langsung di-write ke file csv, penyimpanan dari array ke file csv dilakukan pada saat user/player melakukan save)

Tindak Lanjut

Pemeriksaan lebih lanjut mengenai skema pengulangan yang bisa menggunakan beberapa metode. Lalu mencoba segala metode agar code berjalan secara optimal. Untuk masalah miskonsepsi, asisten praktikum sedikit menjelaskan kembali terkait cara penyimpanan data pada F10 dan F12 (Shop dan Shop management)

Dokumentasi



Gambar Lampiran 4 - MoM Asistensi 2