

LAPORAN TUGAS BESAR
MATA KULIAH IF 1210 DASAR PEMROGRAMAN
PROGRAM GAME “OWCA”

Kelas 02 / Kelompok I



Dosen Pengampu

Dr. Fazat Nur Azizah ST, M.Sc

Disusun oleh

Nama	NIM
Adhimas Aryo Bimo	19623142
Hans Joseph B. W. Silitonga	19623132
Muhammad Rizain Firdaus	19623262
Mudzaki Kaarzaqiel Hakim	19623272
Muhamad Dzakwan Musfajra	16523252

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG

MEI 2024

PERNYATAAN KELOMPOK

Kami, selaku anggota kelompok I, mata kuliah Dasar Pemrograman kelas 02 yang beranggotakan

Nama	NIM
Adhimas Aryo Bimo	19623142
Hans Joseph B. W. Silitonga	19623132
Muhammad Rizain Firdaus	19623262
Mudzaki Kaarzaqiel Hakim	19623272
Muhamad Dzakwan Musfajra	16523252

Dengan ini, masing-masing dari kami menyatakan bahwa *“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.”*

DAFTAR ISI

DESAIN COMMAND.....	5
1. F00 - LCG.....	5
2. F01 - Register.....	5
3. F02 - Login.....	6
4. F03 - Logout.....	7
5. F04 - Menu & Help.....	7
6. F05 - Monster.....	7
7. F06 - Potion.....	7
8. F07 - Inventory.....	8
9. F08- Battle.....	8
10. F09 - Arena.....	9
11. F10 - Shop & Currency.....	9
12. F11 - Laboratory.....	10
13. F12 - Shop Management.....	10
14. F13 - Monster Management.....	14
DESAIN KAMUS DATA.....	18
1. F-00 LCG.....	18
2. F-01 Register.....	18
3. F-02 Login.....	18
4. F-03 Logout.....	18
5. F-04 Menu & Help.....	19
6. F-07 Inventory.....	19
7. F-08 Battle.....	19
8. F-09 Arena.....	20
9. F-10 Shop & Currency.....	21
10. F-11 Laboratory.....	22
11. F-12 Shop Management.....	22
12. F-13 Monster Management.....	23
13. F-14 Load.....	24
14. F-15 Save.....	24
15. F-16 Exit.....	24
DESAIN DEKOMPOSISI ALGORITMIK DAN DESAIN PROGRAM.....	26
1. F-00 LCG.....	26
2. F-01 Register.....	26
3. F-02 Login.....	27
4. F-03 Logout.....	27
5. F-04 Menu & Help.....	27

6. F-07 Inventory.....	28
7. F-08 Battle.....	29
8. F-09 Arena.....	29
9. F-10 Shop &Currency.....	30
10. F-11 Laboratory.....	31
11. F-12 Shop Management.....	32
12. F-13 Monster Management.....	34
13. F-14 Load.....	36
14. F-15 Save.....	36
15. F-16 Exit.....	37
SPESIFIKASI PROGRAM.....	38
1. F-00 LCG.....	38
2. F-01 Register.....	38
3. F-02 Login.....	39
4. F-03 Logout.....	40
5. F-04 Menu & Help.....	41
6. F-07 Inventory.....	42
7. F-08 Battle.....	43
8. F-09 Arena.....	51
9. F-10 Shop &Currency.....	52
10. F-11 Laboratory.....	52
11. F-12 Shop Management.....	54
12. F-13 Monster Management.....	58
13. F-14 Load.....	60
14. F-15 Save.....	60
15. F-16 Exit.....	62
TANGKAPAN GAMBAR UJI PROGRAM.....	64
1. F-01 Register.....	64
2. F-02 Login.....	67
3. F-03 Logout.....	70
4. F-04 Menu & Help.....	71
5. F-07 Inventory.....	72
6. F-08 Battle.....	74
7. F-09 Arena.....	80
8. F-10 Shop &Currency.....	83
9. F-11 Laboratory.....	89
10. F-12 Shop Management.....	91
11. F-13 Monster Management.....	95

12. F-14 Load.....	97
13. F-15 Save.....	98
14. F-16 Exit.....	100
LAMPIRAN.....	101

DAFTAR TABEL

Tabel 1.1 Daftar Pembagian Kerja.....	3
Tabel 1.2 Bagan Rangkaian Kerja.....	4

DAFTAR GAMBAR

Gambar 1.1 Input Username Valid.....	63
Gambar 1.2 Jika Input Choose Monster tidak Valid.....	64
Gambar 1.3 Jika Input Choose Monster Valid.....	64
Gambar 1.4 Jika Input Username Tidak Valid.....	65
Gambar 1.5 Jika Register setelah Register.....	66
Gambar 2.1: Pengujian saat login yang dilakukan berhasil.....	66
Gambar 2.2 Uji Login Username dan Password Tidak Valid.....	67
Gambar 2.3 Uji Password Salah.....	67
Gambar 2.2 Uji Username tidak Valid tapi Password Valid.....	68
Gambar 2.3 Uji Login Setelah Login.....	68
Gambar 3.1 Jika Logout Tidak Valid.....	69
Gambar 3.2 Logout Valid.....	69
Gambar 4.1 Help untuk User yang Belum Login.....	70
Gambar 4.2 Help untuk User yang Telah Login.....	70
Gambar 4.3 Help untuk Admin.....	71
Gambar 5.1 Tampilan Awal Inventory.....	71
Gambar 5.2 Input Tidak Valid di Inventory.....	72
Gambar 5.3 Input Valid di Inventory.....	72
Gambar 6.1: Muncul monster musuh secara acak.....	73
Gambar 6.2: Pengujian input pilihan monster (user) yang tidak tersedia di monster list..	73
Gambar 6.3: Pengujian input pilihan monster (user) yang tersedia di monster list dan user berhasil memilih monster yang diinginkan.....	74
Gambar 6.4: Pengujian saat pilihan perintah user kepada monster (user) tidak tersedia atau tidak valid.....	74
Gambar 6.5: Pengujian saat user memilih perintah 1 (attack).....	76
Gambar 6.6: Pengujian saat user memilih perintah 2 (use potion).....	76
Gambar 6.7: Pengujian saat user memilih potion 1 (strength potion).....	76
Gambar 6.8: Pengujian saat user memilih potion 2 (resilience potion).....	77
Gambar 6.9: Pengujian saat user memilih potion dengan stok yang kosong.....	77
Gambar 6.10: Pengujian saat user memilih perintah cancel/keluar dari pemilihan potion...	77
Gambar 6.11: Pengujian saat user memilih perintah yang tidak tersedia saat pemilihan potion.....	77
Gambar 6.12: Pengujian ketika user berhasil mengalahkan musuh.....	78
Gambar 6.13: Pengujian saat user gagal mengalahkan musuh.....	79
Gambar 6.14: Pengujian saat user memilih untuk quit sebelum battle selesai.....	79

Gambar 7.1: Tampilan awal arena.....	79
Gambar 7.2: Pengujian user berhasil memilih monster dan monster musuh muncul secara acak menyesuaikan level yang sama dengan stage arena.....	80
Gambar 7.3: Pengujian ketika user berhasil mengalahkan musuh dan lanjut ke stage berikutnya.....	81
Gambar 7.4: Pengujian ketika user telah dikalahkan oleh musuh dan sesi arena telah selesai.....	82
Gambar 8.1 Tampilan Awal Shop.....	82
Gambar 8.2 Input Tidak Valid di Shop.....	82
Gambar 8.3 Lihat Monster dan Potion di Shop.....	83
Gambar 8.4 Input tidak valid ketika ingin lihat.....	83
Gambar 8.5 Input Tidak Valid ketika Beli.....	84
Gambar 8.6 Input Monster id Tidak Valid di Beli Monster.....	85
Gambar 8.7 Input Valid di Beli Monster.....	85
Gambar 8.8 Input Tidak Valid di Beli Potion.....	87
Gambar 8.9 Input Valid Beli Potion.....	87
Gambar 9.1 Ketika Ingin Upgrade Lvl Max Monster.....	88
Gambar 9.2 Input Valid di Laboratory.....	89
Gambar 9.3 Input Tidak Valid di Laboratory.....	89
Gambar 10.1: Pengujian fungsi saat input pilihan aksi tidak valid.....	90
Gambar 10.2: Pengujian fungsi saat input pilihan aksi valid tetapi input lihat (monster/potion) tidak valid.....	90
Gambar 10.5: Pengujian fungsi saat input pilih aksi adalah tambah.....	91
Gambar 10.6: Pengujian fungsi saat input id, stok_baru, dan harga_baru tidak valid atau valid dan masih ada monster/potion yang belum termasuk dalam shop.....	91
Gambar 10.7: Pengujian fungsi saat tidak ada monster/potion yang belum termasuk dalam shop.....	92
Gambar 10.8: Pengujian fungsi saat pilihan aksi adalah ubah monster/potion dan input id tidak valid.....	92
Gambar 10.9: Pengujian fungsi saat stok_baru dan harga_baru tidak kosong.....	92
Gambar 10.10: Pengujian fungsi saat harga_baru kosong dan stok_baru tidak kosong....	92
Gambar 10.11: Pengujian fungsi saat harga_baru tidak kosong dan stok_baru kosong....	93
Gambar 10.12: Pengujian fungsi saat harga_baru dan stok_baru yang kosong.....	93
Gambar 10.13: Pengujian fungsi saat pilihan aksi hapus monster/potion dengan input id valid dan yakin ingin menghapus (y).....	93
Gambar 10.14: Pengujian fungsi saat pilihan aksi hapus monster/potion dengan input id valid dan tidak yakin ingin menghapus (n).....	93
Gambar 10.15: Pengujian fungsi saat pilihan aksi hapus monster/potion dengan input id valid dan input yakin tidak valid.....	93
Gambar 10.16: Pengujian fungsi saat input id tidak valid.....	94

Gambar 10.17: Pengujian fungsi saat pilihan aksi adalah keluar.....	94
Gambar 11.1 Pengujian fungsi saat input awal aksi tidak valid.....	94
Gambarr 11. 2 Pengujian fungsi saat input aksi tampilkan monster (1).....	94
Gambar 11.3 Pengujian fungsi saat pilihan aksi adalah tambah monster baru (2) dengan variasi input yang valid dan tidak valid.....	95
Gambar 11.4 Pengujian fungsi saat semua input valid dan input ingin menambahkan monster ke database tidak valid.....	95
Gambar 11.5 Pengujian fungsi saat input nya adalah y.....	95
Gambar 11.6 Pengujian fungsi saat input nya adalah n.....	95
Gambar 11.7 Pengujian fungsi saat input pilihan aksi adalah kembali ke menu admin....	95
Gambar 12.1 Uji Load.....	96
Gambar 12.2 Uji Load jika tidak ada Input.....	97
Gambar 12.3 Uji untuk mengecek apakah terdapat folder atau tidak.....	97
Gambar 13.1 Uji Input Valid pada Save.....	98
Gambar 13.2 Uji Save ketika Input Tidak Valid.....	98
Gambar 14.1 Uji Untuk Exit.....	99
Gambar 14.2 Uji Input Tidak Valid untuk Exit.....	99

DESKRIPSI PERSOALAN

Program "OWCA" adalah program game berbasis bahasa pemrograman *python*. Program ini ditujukan untuk membantu agen purry untuk melawan monster yang diciptakan oleh Asep Spakbor. Untuk mengakses fitur game pengguna harus mendaftarkan akun terlebih dahulu. Akun tersebut akan digunakan untuk login dan menyimpan data-data permainan yang dilakukan oleh pengguna.

Informasi yang dimasukkan pengguna pada saat mendaftarkan akun akan tersimpan dalam file "user.csv". Pada file tersebut tersimpan id, password, username, role, dan jumlah OC yang dimiliki oleh user.

DAFTAR PEMBAGIAN KERJA

Fitur	Implementasi *)	NIM Desainer **)	NIM Coder **)	NIM Tester **)
F00 - Random Number Generator	<u>function</u> LCG	19623272	19623272	19623272
F01 - Register	<u>function</u> validate_username <u>function</u> choose_monster <u>function</u> check_register <u>function</u> user_input <u>function</u> register_page	19623142	19623142	19623142
F02 - Login	<u>function</u> check_input <u>function</u> user_login <u>function</u> login_page	19623142	19623142	19623142
F03 - Logout	<u>function</u> logout	19623142	19623142	19623142
F04 - Menu & Help	<u>function</u> help_menu	19623142	19623142	19623142
F05 - Monster	file csv	16523252	16523252	16523252
F06 - Potion	file csv	19623262	19623262	19623262
F07 - Inventory	<u>procedure</u> print_monster <u>procedure</u> print_potion <u>procedure</u> print_monster_ball <u>procedure</u> display_inventory <u>procedure</u> print_monster_details <u>procedure</u> print_potion_details <u>procedure</u> print_details_by_id <u>function</u> atribut_by_level	19623142	19623142	19623142
F08 - Battle	<u>procedure</u> battle <u>function</u> atribut_by_level <u>procedure</u> update_user_data <u>procedure</u> print_hp_bar	19623262 19623142	19623262 19623142	19623262 19623142
F09 - Arena	<u>procedure</u> arena <u>function</u> atribut_by_level <u>procedure</u> update_user_data <u>procedure</u> print_hp_bar	19623262 19623142	19623262 19623142	19623262 19623142

F10 - Shop & Currency	<u>procedure</u> delay <u>function</u> shop_currency_page <u>function</u> lihat <u>function</u> beli <u>function</u> beli_monster <u>function</u> beli_potion <u>function</u> count_char_max <u>procedure</u> lihat_monster <u>procedure</u> lihat_potion <u>function</u> item_shop_arr	19623142	19623142	19623142
F11 - Laboratory	<u>function</u> laboratory <u>function</u> level_price <u>function</u> display_monster_inventory <u>procedure</u> display_upgrade_price	19623272 19623142	19623272 19623142	19623272 19623142
F12 - Shop Management	<u>function</u> monster_not_in_shop <u>function</u> item_not_in_shop <u>function</u> item_shop_arr <u>function</u> tampilan_awal <u>function</u> memilih <u>function</u> lihat_monster <u>function</u> lihat_potion <u>function</u> tambah_monster <u>function</u> tambah_potion <u>function</u> ubah_monster <u>function</u> ubah_potion <u>function</u> hapus_monster <u>function</u> hapus_potion <u>function</u> keluar <u>function</u> count_char_max <u>procedure</u> sort_data <u>procedure</u> delay <u>function</u> get_numeric_input <u>function</u> get_numeric_input_boleh_kosong	19623132 19623142	19623132 19623142	19623132 19623142

F13 - Monster Management	<u>function</u> tampilan_awal <u>function</u> pilihan_monster_management <u>function</u> cek_kesamaan_nama <u>function</u> tambah_monster_baru <u>function</u> tambah_monster_ke_database <u>function</u> count_char_max <u>procedure</u> lihat_monster	19623132 19623142	19623132 19623142	19623132 19623142
F14 - Load	<u>procedure</u> load <u>procedure</u> delay	19623272	19623272 19623142	19623272 19623142
F15 - Save	<u>procedure</u> save <u>procedure</u> data_save <u>procedure</u> sort_data <u>function</u> is_valid_name <u>procedure</u> get_valid_input	19623272 19623142	19623272 19623142	19623272 19623142
F16 - Exit	<u>procedure</u> game_exit	19623142	19623142	19623142

Tabel 1.1 Daftar Pembagian Kerja

Fitur	Implementasi	Desainer	Coder	Tester
F00 - Random Number Generator	✓	✓	✓	✓
F01 - Register	✓	✓	✓	✓
F02 - Login	✓	✓	✓	✓
F03 - Logout	✓	✓	✓	✓
F04 - Menu & Help	✓	✓	✓	✓
F05 - Monster	✓	✓	✓	✓
F06 - Potion	✓	✓	✓	✓
F07 - Inventory	✓	✓	✓	✓
F08 - Battle	✓	✓	✓	✓
F09 - Arena	✓	✓	✓	✓
F10 - Shop & Currency	✓	✓	✓	✓
F11 - Laboratory	✓	✓	✓	✓
F12 - Shop Management	✓	✓	✓	✓
F13 - Monster Management	✓	✓	✓	✓
F14 - Load	✓	✓	✓	✓
F15 - Save	✓	✓	✓	✓
F16 - Exit	✓	✓	✓	✓

Tabel 1.2 Bagan Rangkaian Kerja

DESAIN COMMAND

1. F00 - LCG

- Command: LCG(lower_limit : integer, upper_limit : integer)
{membuat angka random pada rentang tertentu}

Masukan:

- >lower limit
- >upper limit

Keluaran:

- >angka sembarang pada rentang tersebut

2. F01 - Register

- Command:
 - > validate_username(username: str) {memvalidasi username user}
 - > choose_monster(monster_data: list[str],username:str) {memilih monster}
 - > check_register(username: str, password:str, user_data:list[dict]) {Meregister username dan memvalidasinya}
 - > user_input(user_data:list[dict]) {Input username dan password user}
 - > register_page(game_state: integer, username:str, monster_data:list[dict], monster_inventory:list[dict], user_data:list[dict]) {Membuat laman register untuk user}

Masukan:

- > Masukkan username
- > Masukkan password
- > Memilih monster

Keluaran:

Bila username mengandung karakter selain alfabet, angka, underscore, dan strip, maka akan dicetak:

- > **'Username hanya boleh berisi alfabet, angka, underscore dan strip!'**

Bila username sudah terdaftar sebelumnya, maka akan dicetak:

- > **'Username {username} sudah terpakai, silahkan gunakan username lain!'**

Bila user sudah login dan mencoba untuk register maka akan dicetak:

- > **'Register gagal!\nAnda telah login dengan username {username}, silahkan lakukan "LOGOUT" sebelum melakukan register!'**

Setelah user memilih monster akan dicetak:

> **'Selamat datang agent {username}. Mari kita mengalahkan Dr. Asep Spakbor dengan {monster_data[monster_id-1]['type']}!'**

Jika monster yang dimasukkan oleh user tidak ada maka akan dicetak:

> **'Monster pilihan anda tidak ada, silahkan pilih kembali.'**

3. F02 - Login

- Command: `check_input(username: str, password:str,user_data:list[dict])`
{mengecek input username dan password dari user}

>Keluaran:

Bila username dan password input sudah sesuai dengan data, maka akan dicetak:

> **'Selamat datang, Agent {username}!\nMasukkan command "help" untuk daftar command yang dapat kamu panggil.'**

Bila password salah, maka akan dicetak:

> **'Password salah!'**

Bila username belum terdaftar, maka akan dicetak:

> **'Username tidak terdaftar !'**

Command: `user_login(data:list[dict])`

>Masukan:

> Masukkan username

> Masukkan password

Command: `login_page(game_state: integer,username:str,user_data:list[dict])`
{membuat laman login untuk user}

>Keluaran:

Bila user sudah login dan ingin melakukan login maka akan tercetak:

> **'Login gagal!\nAnda telah login dengan username {username}, silahkan lakukan "LOGOUT" sebelum melakukan login kembali!'**

4. F03 - Logout

- Command: logout(game_state: integer, is_admin: bool)

>Masukan:

> Menuliskan “LOGOUT” di command

>Keluaran:

Bila user sudah login dan setelah menuliskan command logout maka user akan Keluar dari game dan akan tercetak di layar:

> **‘Anda telah logout!’**

Bila user belum login dan menuliskan command logout akan tercetak:

> **‘Logout gagal!’**

> **‘Anda belum login, silahkan login terlebih dahulu sebelum melakukan logout!’**

5. F04 - Menu & Help

- Command: help_menu(username: str, is_admin: bool)

>Masukan:

> mengetik “HELP” pada command

> Login

> Register

> Exit

>Keluaran:

> Jika user belum login dan memanggil command HELP maka akan ditampilkan di layar 1.Login, 2. Register, 3. Exit.

> Jika login sebagai admin maka akan ditampilkan menu 1. Shop Management dan 2. Monster Management.

> Jika user sudah login dan sebagai player maka akan ditampilkan menu utama berupa 1. Inventory, 2. Battle, 3. Arena, 4. Laboratory, 5. Shop.

6. F05 - Monster

Berupa file .csv

7. F06 - Potion

Berupa file .csv

8. F07 - Inventory

- command: `print_monster(cnt:integer,monster:dict)`, `print_potion(cnt:integer,potion:dict)`,
`print_monster_ball(cnt: integer,monster_ball:dict)`,
`display_inventory(player_inventory:list[dict],coin: integer) →list[dict]`,
`print_monster_details(monster:dict)`, `print_potion_details(potion:dict)`,
`print_details_by_id(data:dict)`, `atribute_by_level(atribute: integer, level: integer)`

Masukan:

- > `monster_inventory_data (list[dict])`: Data inventaris monster yang dimiliki oleh pengguna.
- > `item_inventory (list[dict])`: Data inventaris item yang dimiliki oleh Pengguna.

Keluaran:

- > Jumlah OWCA coin dari user
- > detail item yang terdapat dalam inventory player
- > detail mengenai monster, level, nama
- > detail mengenai potion atau monster ball

9. F08- Battle

- Command: `battle(username:str, monster_data:list[dict], monster_inventory_data:list[dict], potion_data:list[dict], item_inventory:list[dict], user_data:list[dict])`

Masukan:

- > `username (string)`: Nama pengguna yang akan bertarung.
- > `monster_data (list[dict])`: Data monster yang tersedia.
- > `monster_inventory_data (list[dict])`: Data inventaris monster yang dimiliki oleh pengguna.
- > `potion_data (list[dict])`: Data potion yang tersedia.
- > `item_inventory (list[dict])`: Data inventaris item yang dimiliki oleh pengguna
- > `user_data (list[dict])`: Data pengguna yang terdaftar.

Keluaran:

Print output sebagai berikut:

- > Informasi tentang monster musuh yang muncul, termasuk nama, level, ATK Power, DEF Power, dan HP.

- > Informasi tentang monster pengguna yang dipilih untuk bertarung, termasuk nama, level, ATK Power, DEF Power, dan HP.
- > Pesan-pesan interaktif yang menunjukkan perkembangan pertarungan, termasuk serangan, penggunaan potion, serta hasil akhir pertarungan.
- > Pesan output print tambahan dalam skenario-skenario tertentu, seperti jika pilihan tidak tersedia.

10. F09 - Arena

Nama Command: arena(username:str, monster_data:list[dict], monster_inventory_data:list[dict], potion_data:list[dict], item_inventory:list[dict], user_data:list[dict])

Masukan:

- > Monster yang dipilih

Keluaran:

- > Jika menang, maka akan mendapatkan OC
- > Item akan diupdate

11. F10 - Shop & Currency

- Command: shop_currency_page(username:str, monster_shop_data:list[dict], item_shop_data:list[dict], potion_data:list[dict], monster_inventory_data:list[dict], item_inventory:list[dict], monster_data:list[dict], user_data:list[dict])

Masukan:

- > username (string): Nama pengguna yang akan bertarung.
- > monster_shop_data (list[dict]): Data monster yang tersedia di shop.
- > item_shop_data (list[dict]): Data item yang tersedia di shop.
- > potion_data (list[dict]): Data potion yang tersedia.
- > monster_inventory_data (list[dict]): Data inventaris monster yang dimiliki oleh pengguna.
- > item_inventory (list[dict]): Data inventaris item yang dimiliki oleh pengguna.
- > monster_data (list[dict]): Data monster yang tersedia.
- > user_data (list[dict]): Data pengguna yang terdaftar.

Keluaran:

Print output sebagai berikut:

- > Informasi Shop: Selamat datang di SHOP dan perintah untuk memilih aksi.
- > Aksi lihat: Menampilkan data monster atau potion yang tersedia di shop.
- > Aksi beli: Membeli monster atau potion dari shop dan menambahkan ke inventory pengguna jika koin mencukupi.

> Pesan Kesalahan: Menampilkan pesan jika perintah tidak tersedia atau input tidak valid.

12. F11 - Laboratory

- Command: laboratory(username:str, monster_inventory_data:list[dict] , user_data:list[dict] , monster_data:list[dict])

Masukan:

- > ID monster yang ingin di upgrade levelnya.
- > memilih ingin di upgrade ke level berapa.
- > mengkonfirmasi kenaikan level.

Keluaran:

- > Tampilan tabel untuk menaikkan level monster.
- > Tampilan untuk menampilkan harga upgrade monster.
- > Pada fungsi ini user dapat mengupgrade monster dan jika berhasil maka OWCA coin mereka akan dikurangkan dengan harga untuk meng-upgradenya.

13. F12 - Shop Management

- Command : tampilan_awal(username:str, item_shop:list[dict], monster_shop_data:list[dict], monster_data:list[dict], potion_data:list[dict])
 - > {Mengeluarkan tampilan awal pada shop management}
 - > Keluaran :
"Halo {username}, Selamat datang kembali!"
"Terdapat beberapa pilihan aksi :"
- Command : memilih(username:str, item_shop_data:list[dict], monster_shop_data:list[dict], monster_data:list[dict], potion_data:list[dict])
 - > {Memilih opsi (lihat/tambah/ubah/hapus/keluar)}
 - > Masukan : pilihan (lihat/tambah/ubah/hapus/keluar)
- Command : lihat_monster(monster_shop_data:list[dict], monster_data:list[dict])
 - > {Menampilkan tabel dari monster yang ada di shop}
 - > Masukan :
 - >>> pilihan (lihat/tambah/ubah/hapus/keluar) : lihat
 - >>> lihat (monster/potion) : monster
 - > Keluaran :

(output berupa tabel dari monster yang berisi ID, Name/Type, ATK Power, DEF Power, HP, Stok, dan Harga)

- Command : lihat_potion(item_shop_list:list[dict])
 - > {Menampilkan tabel dari item (potion) yang ada di shop}
 - > Masukan :
 - >>> pilihan (lihat/tambah/ubah/hapus/keluar) : lihat
 - >>> lihat (monster/potion) : monster
 - > Keluaran :
(output berupa tabel dari potion yang berisi ID, Name/Type, Stok, dan Harga)
- Command : tambah_monster(monster_not_in_shop_list:list[dict], monster_shop_data:list[dict], monster_data:list[dict])
 - > {Menambah monster yang belum ada di shop menjadi terdaftar pada shop}
 - > Masukan :
 - >>> pilihan (lihat/tambah/ubah/hapus/keluar) : tambah
 - >>> lihat (monster/potion) : monster
 - >>> monster_id
 - >>> stok_baru
 - >>> harga_baru
 - > Keluaran :
 - >>> Jika tidak ada monster yang berada di database, diluar dari shop
“Semua monster telah ada di shop”
 - >>> Jika tidak : (Menampilkan tabel dari data monster yang belum ada di shop)
 - >>> Jika monster_id atau stok_baru atau harga_baru tidak berupa integer :
“Input harus berupa angka. Silakan coba lagi”
 - >>> Jika monster_id tidak terdapat dalam monster_not_in_shop_list :
“Tidak tersedia id tersebut, pilih id yang lain”
 - >>> Jika monster_id terdapat dalam monster_not_in_shop_list :
“Proses menambahkan {nama} ke dalam shop telah berhasil”
- Command : tambah_potion(item_not_in_shop_list:list[dict], item_shop_list:list[dict], item_shop_data, potion_data:list[dict])
 - > {Menambah item potion yang belum ada di shop menjadi terdaftar pada shop}
 - > Masukan :
 - >>> pilihan (lihat/tambah/ubah/hapus/keluar) : tambah
 - >>> lihat (monster/potion) : potion
 - >>> Id
 - >>> stok_baru
 - >>> harga_baru

- > Keluaran :
 - >>> Jika tidak ada item yang berada di database tetapi tidak ada di shop :
“Semua item telah berada di shop”
 - >>> Jika tidak : (Menampilkan tabel dari data potion yang belum ada di shop)
 - >>> Jika Id atau stok_baru atau harga_baru tidak berupa integer :
“Input harus berupa angka. Silakan coba lagi”
 - >>> Jika monster_id tidak terdapat dalam monster_not_in_shop_list :
“Tidak tersedia id tersebut, silakan pilih id yang lain”
 - >>> Jika monster_id terdapat dalam monster_not_in_shop_list :
“Proses menambahkan {nama_item} ke dalam shop telah berhasil”
- Command : ubah_monster(monster_shop_data:list[dict], monster_data:list[dict])
 - > {Mengubah Stok dan/atau Harga pada data monster yang ada di shop}
 - > Masukan :
 - >>> pilihan (lihat/tambah/ubah/hapus/keluar) : ubah
 - >>> lihat (monster/potion) : monster
 - >>> Id
 - >>> stok_baru
 - >>> harga_baru
 - > Keluaran :
 - >>> (Menampilkan tabel dari data monster yang ada di shop)
 - >>> Jika input Id atau stok_baru atau harga_baru tidak valid :
“Tidak tersedia id tersebut, pilih id yang lain”
 - >>> Jika stok_baru dan harga_baru menerima input valid:
“Pembaharuan data berhasil dilakukan pada monster {nama} dengan stok baru berjumlah {stok_baru} dan harga baru {harga_baru}”
 - >>> Jika stok_baru menerima input valid tetapi harga_baru kosong:
“Pembaharuan data berhasil dilakukan pada monster {nama} dengan stok baru {stok_baru}”
 - >>> Jika stok_baru kosong tetapi harga_baru menerima input valid :
“Pembaharuan data berhasil dilakukan pada monster {nama} dengan harga baru {harga_baru}”
 - >>> Jika stok_baru dan harga_baru kosong:
“Tidak ada perubahan”
- Command : ubah_potion(item_shop_list:list[dict],item_shop_data:list[dict])
 - > {Mengubah Stok dan/atau Harga pada potion yang ada di shop}
 - > Masukan :
 - >>> pilihan (lihat/tambah/ubah/hapus/keluar) : ubah
 - >>> lihat (monster/potion) : potion

- ```

>>> Id
>>> stok_baru
>>> harga_baru

```
- > Keluaran :
- ```

>>> (Menampilkan tabel dari data potion yang ada di shop)
>>> Jika input Id atau stok_baru atau harga_baru tidak valid :
“Tidak tersedia id tersebut, pilih id yang lain”
>>> Jika stok_baru dan harga_baru menerima input valid:
“Pembaharuan data berhasil dilakukan pada potion jenis {nama_potion} dengan stok baru berjumlah {stok_baru} dan harga baru {harga_baru}”
>>> Jika stok_baru menerima input valid tetapi harga_baru kosong:
“Pembaharuan data berhasil dilakukan pada potion jenis {nama_potion} dengan stok baru {stok_baru}”
>>> Jika stok_baru kosong tetapi harga_baru menerima input valid :
“Pembaharuan data berhasil dilakukan pada potion jenis {nama_potion} dengan harga baru {harga_baru}”
>>> Jika stok_baru dan harga_baru kosong:
“Tidak ada perubahan”

```
- Command : `hapus_monster(monster_shop_data:list[dict], monster_data:list[dict])`
 - > {Menghapus monster yang dipilih dari list data monster}
 - > Masukan :

```

>>> pilihan (lihat/tambah/ubah/hapus/keluar) : ubah
>>> lihat (monster/potion) : monster
>>> Id
>>> Yakin ingin menghapus? (y/n)

```
 - > Keluaran :

```

>>> (Menampilkan tabel dari data monster yang ada di shop)
>>> Jika input Id tidak valid :
“Tidak tersedia id tersebut, pilih id yang lain”
>>> Jika input (y/n) adalah (y) :
“Done, {type_monster} telah berhasil dihapus dari shop.”
>>> Jika input (y/n) adalah (n) :
“{type_monster} dibatalkan untuk dihapus dari shop.”
>>> Jika input (y/n) tidak valid :
“Input anda salah, ulangi!”

```
 - Command : `hapus_potion(item_shop_list:list[dict], item_shop_data:list[dict])`
 - > {Menghapus monster yang dipilih dari list data monster}
 - > Masukan :

- ```

>>> pilihan (lihat/tambah/ubah/hapus/keluar) : ubah
>>> lihat (monster/potion) : potion
>>> Id
>>> Yakin ingin menghapus? (y/n)
> Keluaran :
>>> (Menampilkan tabel dari data potion yang ada di shop)
>>> Jika input Id tidak valid :
“Tidak tersedia id tersebut, pilih id yang lain”
>>> Jika input (y/n) adalah (y) :
“Done, {type_potion} telah berhasil dihapus dari shop.”
>>> Jika input (y/n) adalah (n) :
“{type_potion} dibatalkan untuk dihapus dari shop.”
>>> Jika input (y/n) tidak valid :
“Input anda salah, ulangi!”

```
- Command : keluar(username:str)
    - > {Keluar dari shop management}
    - > Keluaran : **“Sampai jumpa lagi {username}”**

#### 14. F13 - Monster Management

- Command : tampilan\_awal(monster\_list:list[dict])
  - > {Mengeluarkan tampilan awal pada monster management}
  - > Keluaran :
    - “Selamat Datang Para Agen”**
    - “Di sini adalah tempat database para monster”**
- Command : pilihan\_monster\_management(monster\_list:list[dict])
  - > {Memilih fungsi yang ada dalam monster management}
  - > Masukan :
    - >>> pilihan (1/2/3)
      1. Tampilkan semua monster yang ada
      2. Tambahkan monster baru
      3. Untuk kembali ke menu admin
  - > Keluaran :
    - >>> Jika pilihan (1/2/3) = 3 :
      - “Keluar dari Monster Management”**
      - “...”**
    - >>> Jika pilihan tidak valid (diluar dari 1/2/3) :



**“Input anda salah, ulangi!”**

- Command : tambah\_monster\_baru(monster\_list:list[dict])
  - > {Menambah atribut monster baru}
  - > Masukan :
    - >>> pilihan (1/2/3) : 2
    - >>> nama
    - >>> atk
    - >>> defense
    - >>> hp
  - > Keluaran :
    - “... Proses pembuatan monster baru dimulai ...”**
    - >>> Jika nama sama dengan yang sudah ada dalam data :  
**“Ups.. Nama Monster sudah terdaftar...”**
    - “Silakan masukkan nama lain”**
    - >>> Jika atk tidak valid :  
**“ATK Power harus dalam angka yang bernilai positif...”**
    - “Silakan coba lagi”**
    - >>> Jika defense berupa bilangan bulat diluar rentang 0-50 :  
**“Def Power Monster harus bernilai 0-50...”**
    - “Silakan coba lagi”**
    - >>> Jika defense atau HP tidak berupa bilangan bulat :  
**“Input harus berupa bilangan bulat”**
    - >>> Jika hp tidak valid :  
**“HP Monster harus bernilai bilangan positif”**
    - “Silakan coba lagi”**
- Command : tambah\_monster\_ke\_database(monster\_list:list[dict],monster\_baru:dict)
  - > {Menambah monster yang telah dibuat ke database}
  - > Masukan :
    - >>> pilihan Ingin tambah monster baru ke database (y/n)
  - > Keluaran :
    - >>> Jika pilihan = y :  
**“Monster baru berhasil ditambahkan ke database!”**
    - >>> Jika pilihan = n :  
**“Monster baru tidak berhasil ditambahkan ke database”**
- Command : lihat\_monster(monster\_data:list[dict])
  - > {Mengeluarkan output berupa tabel yang berisi data monster yang ada di database}
  - > Keluaran :

### **“Memuat data Monster yang belum ada di shop”**

>>> (Tabel berisi id, type, atk\_power, def\_power, dan hp monster)

#### **15. F14 - Load**

{memvalidasi dan mengakses folder yang berisi file csv}

>masukan:

>> nama folder

>keluaran:

>> jika masukan kosong:

**"Tidak ada nama folder yang diberikan!\nUsage : python main.py"**

**<data/nama\_folder>**

>> jika nama folder tidak ditemukan:

**"Folder {folder} tidak ditemukan!"**

>> jika folder ditemukan:

**"Selamat Datang di program OWCA!"**

#### **16. F15 - Save**

- Command: save(user:list[dict], item\_inventories:list[dict], item\_shop:list[dict], monster, monster\_shop:list[dict], monster\_inventory:list[dict])  
>{memeriksa apakah folder yang dimaksud sudah ada dan akan membuat folder baru jika belum serta menyimpan file csv ke folder tersebut}  
> masukan:  
>> nama folder untuk menyimpan csv  
>keluaran:  
>> jika folder sudah ada:  
**“Berhasil menyimpan data di folder {folder}!”**  
>> jika folder belum ada:  
**"Membuat folder {folder}."**
- Command:data\_save(path : str, file\_name : str, data )  
>{menyimpan data dari bentuk list of dictionaries menjadi csv di suatu folder}  
>keluaran:  
>> bentuk csv dari list of dictionaries
- Command:sort\_data(data:list[dict],sortby:str)  
>{mengurutkan data sesuai urutan }  
> masukan:  
>> data  
>> index pengurut  
>keluaran  
>> data yang telah diurutkan berdasarkan index pengurut
- Command:is\_valid\_name(name:str)  
>{Memastikan bahwa input nama folder merupakan nama yang valid}  
>masukan:  
>> nama folder

- >keluaran:
  - >>jika nama folder nama yang valid maka akan True
  - >> jika nama folder tidak valid maka akan False
- Command `get_valid_input(prompt:str)`
  - >{Meminta user untuk memasukkan nama folder, jika salah maka akan meminta user untuk mengulangi}
  - >masukan:
    - >> nama folder
  - >keluaran:
    - >> jika nama folder tidak valid:  
**"Input invalid, gunakan nama lain!"**

## 17. F16 - Exit

- Command: `game_exit(username:str, user:list[dict], item_inventories:list[dict], item_shop:list[dict], monster:list[dict], monster_shop:list[dict], monster_inventory:list[dict])`
- Masukan:
  - > Input user berupa y/n untuk mengesave data
- Keluaran:
  - > Jika pilihan "y" maka data user akan di save dalam folder dan memasuki fungsi Save
  - > Jika pilihan "n" maka data user tidak akan di save

## DESAIN KAMUS DATA

### 1. F-00 LCG

Lower\_limit, Upper\_limit, seed, a, c, m : integer

function LCG(lower\_limit: integer, upper\_limit: integer) → integer

### 2. F-01 Register

game\_state : integer

username : string

monster\_data , monster\_inventory, user\_data : list of dict

function validate\_username(username: string) → bool

function choose\_monster(monster\_data: list[string], username: string)

function check\_register(username: string, password: string, user\_data: list[dict]) → tuple[string, string]

function user\_input(user\_data: list[dict])

function register\_page(game\_state: integer, username: string, monster\_data: list[dict], monster\_inventory: list[dict], user\_data: list[dict])

### 3. F-02 Login

is\_admin: bool

game\_state: integer

username, password : string

function check\_input(username: string, password: string, user\_data: list[dict]) -> tuple[integer, bool, string]

function user\_login(data: list[dict]) -> tuple[integer, bool, string]:

function login\_page(game\_state: integer, username: string, user\_data: list[dict]) -> tuple[integer, bool, string]

### 4. F-03 Logout

game\_state: integer

is\_admin: boolean

function logout (game\_state:integer, is\_admin:integer)

## 5. F-04 Menu & Help

username: str

is\_admin: bool

main\_text, pretext: string

function help\_menu (username: str, is\_admin: bool)

## 6. F-07 Inventory

cnt, level, hp, quantity, coin, index, atk\_power, def\_power, i, attribute : integer

type, item\_id, id : string

monster, potion, monster\_ball, data : list[dict]

player\_inventory, data : list[dict]

ids : list[string]

procedure print\_monster(cnt: integer, monster: list[dict])

procedure print\_potion(cnt: integer, potion: list[dict])

procedure print\_monster\_ball(cnt: integer, monster\_ball: list[dict])

function display\_inventory(player\_inventory: list of list[dict], coin: integer)

procedure print\_monster\_details(monster: list[dict])

procedure print\_potion\_details(potion: list[dict])

procedure print\_details\_by\_id(data: list[dict])

function attribute\_by\_level(attribute: integer, level: integer)

## 7. F-08 Battle

cnt, level, hp, quantity, coin, index, atk\_power, def\_power, i, attribute : integer

type, item\_id, id : string

monster, potion, data, desired\_monster\_enemy, user\_monster\_choice\_list :list[dict]

player\_inventory, monster\_data, monster\_inventory\_data, potion\_data, item\_inventory,  
user\_data : list[dict]

monster\_ids, ids : list[str]

procedure battle(username:string , monster\_data:list[dict],  
monster\_inventory\_data:list[dict], potion\_data:list[dict], item\_inventory:list[dict],  
user\_data:list[dict])

function attribute\_by\_level(attribute:integer, level:integer):

procedure update\_user\_data(username:string, item\_inventory:list[dict],  
user\_data:list[dict], qty\_str:integer, qty\_res:integer, qty\_heal:integer, coin:integer)

procedure print\_hp\_bar(name:string, hp:integer, max\_hp:integer, bar\_length=const)

## 8. F-09 Arena

cnt, level, hp, quantity, coin, index, atk\_power, def\_power, i, attribute, stage :integer

type, item\_id, id : string

monster, potion, monster\_ball, data, desired\_monster\_enemy, user\_monster\_choice\_list,  
strength\_, resilience\_, healing\_ :list[dict]

player\_inventory, monster\_data, monster\_inventory\_data, potion\_data, item\_inventory,  
user\_data : list[dict]

monster\_ids, ids : list[str]

procedure arena(username:string , monster\_data:list[dict],  
monster\_inventory\_data:list[dict], potion\_data:list[dict], item\_inventory:list[dict],  
user\_data:list[dict])

function attribute\_by\_level(attribute:integer, level:integer):

procedure update\_user\_data(username:string, item\_inventory:list[dict],  
user\_data:list[dict], qty\_str:integer, qty\_res:integer, qty\_heal:integer, coin:integer)

procedure print\_hp\_bar(name:string, hp:integer, max\_hp:integer, bar\_length=const)

## 9. F-10 Shop & Currency

monster\_data, monster\_inventory\_data, potion\_data, item\_inventory, user\_data,  
monster\_shop : list[dict]  
username, user\_id, coin, monster\_type : string  
monster\_cost, monster\_stock, item\_cost, item\_stock, qty, char\_max : integer  
item\_shop\_list, list\_of\_len, temp= list

procedure delay()

function shop\_currency\_page(username:string, monster\_shop\_data:list[dict] ,  
item\_shop\_data:list[dict] , potion\_data:list[dict], monster\_inventory\_data :list[dict] ,  
item\_inventory:list[dict] , monster\_data:list[dict] , user\_data:list[dict] )

function lihat(username:string, monster\_shop\_data:list[dict] , item\_shop\_data:list[dict] ,  
potion\_data:list[dict], monster\_inventory\_data :list[dict], item\_inventory:list[dict] ,  
monster\_data:list[dict] , user\_data:list[dict] )

function beli (username:string, monster\_shop\_data:list[dict] , item\_shop\_data:list[dict] ,  
potion\_data:list[dict], monster\_inventory\_data :list[dict], item\_inventory:list[dict] ,  
monster\_data:list[dict] , user\_data:list[dict] )

function beli\_monster (username:string, monster\_shop\_data:list[dict] ,  
item\_shop\_data:list[dict] , potion\_data:list[dict], monster\_inventory\_data :list[dict],  
item\_inventory:list[dict] , monster\_data:list[dict] , user\_data:list[dict], user\_id:string)  
coin:string)

function beli\_potion(username:string, monster\_shop\_data:list[dict] ,  
item\_shop\_data:list[dict] , potion\_data:list[dict], monster\_inventory\_data :list[dict],  
item\_inventory:list[dict] , monster\_data:list[dict] , user\_data:list[dict] ser\_id:string,  
coin:string)

function count\_char\_max(data\_list:list[dict[str, str]], kolom:string, header:string)

procedure lihat\_monster(monster\_shop\_data:list[dict], monster\_data:list[dict])

procedure lihat\_potion(item\_shop\_list:list[dict])

function item\_shop\_arr(item\_shop\_data:list[dict],potion\_data:list[dict])

## 10. F-11 Laboratory

monster\_data, monster\_inventory\_data, user\_data, temp, monster\_list : list[dict]  
username, user\_id, monster\_name: string  
coin, level, level\_up, price: integer

function laboratory(username:string, monster\_inventory\_data:list[dict] ,  
user\_data:list[dict] , monster\_data:list[dict])

function level\_price (level : integer)-> integer

function display\_monster\_inventory(user\_id : string, monster\_inventory\_data:list[dict],  
monster\_data:list[dict])->string

procedure display\_upgrade\_price()

## 11. F-12 Shop Management

monster\_not\_in\_shop\_list, id\_monster\_in\_shop, item\_list, item\_type\_in\_shop,  
item\_not\_in\_shop\_list, item\_shop\_list, list\_of\_len, ids\_list, name\_potion : list[dict]

char\_max, n, value : integer

pilihan, jenis, monster\_id, stok\_baru, harga\_baru, yes\_no, user\_input : string

validasi : boolean

function monster\_not\_in\_shop(monster\_shop\_data:list[dict], monster\_data:list[dict])

function item\_not\_in\_shop(item\_shop:list[dict])

function item\_shop\_arr(item\_shop\_data:list[dict],potion\_data:list[dict])

function tampilan\_awal(username:str, item\_shop:list[dict], monster\_shop\_data:list[dict],  
monster\_data:list[dict], potion\_data:list[dict])

function memilih(username:str, item\_shop\_data:list[dict], monster\_shop\_data:list[dict],  
monster\_data:list[dict], potion\_data:list[dict])



function lihat\_monster(monster\_shop\_data:list[dict], monster\_data:list[dict])

function lihat\_potion(item\_shop\_list:list[dict])

function tambah\_monster(monster\_not\_in\_shop\_list:list[dict],  
monster\_shop\_data:list[dict], monster\_data:list[dict])

function tambah\_potion(item\_not\_in\_shop\_list:list[dict], item\_shop\_list:list[dict],  
item\_shop\_data, potion\_data:list[dict])

function ubah\_monster(monster\_shop\_data:list[dict], monster\_data:list[dict])

function ubah\_potion(item\_shop\_list:list[dict], item\_shop\_data:list[dict])

function hapus\_monster(monster\_shop\_data:list[dict], monster\_data:list[dict])

function hapus\_potion(item\_shop\_list:list[dict], item\_shop\_data:list[dict])

function keluar(username:str)

function count\_char\_max(data\_list:list[dict[str, str]], kolom:str, header:str)

function sort\_data(data:list[dict], sortby:str)

function delay()

function get\_numeric\_input(prompt:str)

function get\_numeric\_input\_boleh\_kosong(prompt: str, allow\_empty: bool = False)-> str

## 12. F-13 Monster Management

List\_of\_len : list[dict]

pil, nama, atk, : string

defense, Atk, hp, char\_max, : integer

function tampilan\_awal(monster\_list:list[dict])

function pilihan\_monster\_management(monster\_list:list[dict])

function cek\_kesamaan\_nama(monster\_list:list[dict], nama:str)

function tambah\_monster\_baru(monster\_list:list[dict])

function tambah\_monster\_ke\_database(monster\_list:list[dict], monster\_baru:dict)

function count\_char\_max(data\_list:list[dict[str, str]], kolom:str, header:str)

procedure lihat\_monster(monster\_data:list[dict])

### 13. F-14 Load

parser : any

procedure load()

procedure delay()

### 14. F-15 Save

folder, parent\_folder, file\_name, sort\_by, name, prompt: string

procedure save(user:list[dict], item\_inventories:list[dict], item\_shop:list[dict], monster,  
monster\_shop:list[dict], monster\_inventory:list[dict]) -> None

procedure data\_save(path : string, file\_name : string, data:list[dict]) -> None

procedure sort\_data(data:list[dict], sortby:string)

function is\_valid\_name(name:string)

procedure get\_valid\_input(prompt:string)

### 15. F-16 Exit

exit\_input, username: string

monster\_data, monster\_inventory\_data, potion\_data, item\_inventory, user\_data,  
monster\_shop : list[dict]

procedure game\_exit(username:string, user:list[dict], item\_inventories:list[dict],  
item\_shop:list[dict], monster:list[dict], monster\_shop:list[dict],  
monster\_inventory:list[dict])

## DESAIN DEKOMPOSISI ALGORITMIK DAN DESAIN PROGRAM

### 1. F-00 LCG

function LCG(lower\_limit, upper\_limit )  
{Fungsi untuk membuat random generator, dapat digunakan pada *battle* dan *arena*}  
{I.S parameter LCG terpenuhi , Lower\_Limit, Upper\_Limit}  
{F.S menghasilkan *random number*}

### 2. F-01 Register

function register\_page(game\_state, username monster\_data, monster\_inventory, user\_data)  
{Fungsi untuk mendaftarkan akun baru ke dalam game, akun tersebut akan di simpan dalam array}  
{I.S parameter register\_page terpenuhi}  
{F.S game\_state akan berubah, akan menambah username, password, dan monster sesuai dengan array masing-masing}

function validate\_username(username)  
{Fungsi untuk mengecek apakah username valid atau tidak}  
{I.S parameter terpenuhi}  
{F.S akan memberikan keluaran bool apakah username valid atau tidak}

function choose\_monster(monster\_data, username)  
{Fungsi untuk memilih monster}  
{I.S validate\_username benar}  
{F.S akan memberikan output monster yang dipilih oleh user}

function check\_register(username, password, user\_data)  
{Fungsi untuk mengecek apakah username benar dan unik}  
{I.S user\_data memiliki minimal satu element, parameter fungsi terpenuhi}  
{F.S jika benar akan memberikan keluaran username dan password}  
{F.S jika salah akan memberikan keluaran username dan password kosong dan register gagal}

function user\_input(user\_data)  
{Fungsi untuk user input username dan password}  
{I.S user\_data memiliki minimal satu element}  
{F.S akan memberikan keluaran username dan password}

### 3. F-02 Login

function login\_page(game\_state, username user\_data)  
{Fungsi untuk menampilkan halaman login dan tempat fungsi-fungsi lain di panggil}  
{I.S username dan password telah terdaftar dalam user.csv, parameter login\_page terpenuhi}  
{F.S jika username dan password benar dan ada, maka login berhasil, akan mengubah game\_state}  
{F.S jika login sebagai admin, maka akan mengubah kondisi is\_admin}  
{F.S jika username tidak ada atau password salah, maka login akan gagal, game\_state tidak berubah dan kembali ke start menu}

function user\_login(data)  
{Fungsi untuk input login}  
{I.S list user\_data terdapat dalam parameter dan user\_data minimal memiliki satu element}  
{F.S username dan password akan terisi lalu akan di cek validasinya di fungsi check\_input)

function check\_input(username, password, user\_data)  
{Fungsi untuk mengecek apakah sebuah input login benar atau tidak}  
{I.S parameter check\_input terpenuhi dan user\_data minimal memiliki satu element}  
{F.S jika check\_input benar, maka akan login akan berhasil, game\_state akan berubah dan username akan berubah}  
{F.S jika check\_input salah, maka login akan gagal, game\_state akan tetap dan tidak bisa memulai permainan}

### 4. F-03 Logout

function logout (game\_state, is\_admin)  
{Fungsi untuk keluar dari akun yang telah login}  
{I.S game\_state = 1 atau is\_admin = True}  
{F.S akan mengubah game\_state dan is\_admin}

### 5. F-04 Menu & Help

function help\_menu (username,is\_admin)  
{Fungsi untuk menampilkan command apa saja yang dapat dimasukkan oleh user}

{F.S akan memberikan petunjuk tergantung pada kondisi user apakah sudah login atau apakah admin}

## 6. F-07 Inventory

procedure print\_monster(cnt, monster)

{Mengeprint monster yang ada di inventory user}

{I.S minimal terdapat 1 element monster dalam player\_inventory}

{F.S akan mengeprint monster\_inventory user}

procedure print\_potion(cnt, potion)

{Mengeprint potion yang ada di inventory user}

{I.S minimal terdapat 1 element potion dalam player\_inventory}

{F.S akan mengeprint potion yang dimiliki user}

procedure print\_monster\_ball(cnt, monster\_ball)

{Mengeprint monster\_ball yang ada di inventory}

{I.S minimal terdapat 1 element monster\_ball dalam player inventory}

{F.S akan mengeprint monster ball yang dimiliki user}

function display\_inventory(player\_inventory, coin)

{Menampilkan player\_inventory dan coin yang dimiliki user}

{I.S parameter terpenuhi}

{F.S akan mengeprint monster dan item yang dimiliki user secara general}

procedure print\_monster\_details(monster)

{Menampilkan detail monster yang dimiliki oleh user}

{I.S terdapat minimal 1 monster dalam monster inventory}

{F.S akan mengeprint detail monster yang dimiliki user}

procedure print\_potion\_details(potion)

{Menampilkan detail potion yang dimiliki oleh user}

{I.S terdapat minimal 1 item dalam inventory}

{F.S akan mengeprint detail item yang dimiliki user}

procedure print\_details\_by\_id(data)

{Menampilkan detail item user}

{I.S terdapat minimal 1 item dalam inventory}

{F.S akan mengeprint detail item yang dipilih oleh user dengan memanggil prosedur lain}

function attribute\_by\_level(attribute, level)  
{Menyesuaikan stats dengan level monster}  
{I.S terdapat minimal 1 monster dalam inventory}  
{F.S akan menampilkan stats monster yang telah disesuaikan dengan level monster}

## 7. F-08 Battle

procedure battle(username , monster\_data, monster\_inventory\_data, potion\_data, item\_inventory, user\_data)  
{I.S parameter terpenuhi dan dipastikan user telah login}  
{F.S jika menang dalam battle akan mendapatkan OC dan item user akan diupdate sesuai dengan pemakaian item}  
{F.S jika kalah maka user tidak akan mendapatkan apa apa dan item user akan diupdate sesuai dengan pemakaian item}

function attribute\_by\_level(attribute, level)  
{Menyesuaikan stats dengan level monster}  
{F.S akan menampilkan stats monster yang telah disesuaikan dengan level monster}

procedure update\_user\_data(username, item\_inventory, user\_data, qty\_str, qty\_res, qty\_heal, coin)  
{Mengupdate item yang dimiliki user}  
{F.S Update item user berdasarkan pemakaian ketika battle}

procedure print\_hp\_bar(name, hp, max\_hp, bar\_length)  
{Mengeprint bar hp monster}  
{F.S Mengeprint hp monster yang tersisa}

## 8. F-09 Arena

procedure arena(username , monster\_data, monster\_inventory\_data, potion\_data, item\_inventory, user\_data)  
{I.S parameter terpenuhi dan dipastikan user telah login}  
{F.S jika menang dalam arena akan mendapatkan ke stage berikutnya dan item user akan diupdate sesuai dengan pemakaian item}  
{F.S jika kalah maka user akan mendapatkan OC sesuai dengan stage terakhir yang ditaklukan, item user akan diupdate sesuai dengan pemakaian item}

function attribute\_by\_level(attribute, level)  
{Menyesuaikan stats dengan level monster}

{F.S akan menampilkan stats monster yang telah disesuaikan dengan level monster}

procedure update\_user\_data(username, item\_inventory, user\_data, qty\_str, qty\_res, qty\_heal, coin)

{Mengupdate item yang dimiliki user}

{F.S Update item user berdasarkan pemakaian ketika battle}

procedure print\_hp\_bar(name, hp, max\_hp, bar\_length)

{Mengeprint bar hp monster}

{F.S Mengeprint hp monster yang tersisa}

## 9. F-10 Shop & Currency

procedure delay()

{Melakukan delay dan clear terminal}

{F.S terminal akan clear}

function shop\_currency\_page(username, monster\_shop\_data, item\_shop\_data, potion\_data, monster\_inventory\_data, item\_inventory, monster\_data, user\_data)

{Akan masuk ke laman shop and currency untuk membeli item atau monster}

{I.S parameter terpenuhi}

{F.S akan masuk ke laman shop currency dan user dapat memilih untuk melihat dan membeli potion atau monster}

function lihat(username, monster\_shop\_data, item\_shop\_data, potion\_data, monster\_inventory\_data, item\_inventor, monster\_data, user\_data)

{Akan masuk ke pilihan lihat monster atau potion}

{I.S user memilih lihat}

{F.S user akan diminta untuk memilih melihat monster atau potion}

function beli (username, monster\_shop\_data, item\_shop\_data, potion\_data, monster\_inventory\_data, item\_inventory, monster\_data, user\_data)

{Akan masuk ke pilihan beli monster atau potion}

{I.S user memilih beli}

{F.S user akan diminta untuk memilih membeli monster atau potion}

function beli\_monster (username, monster\_shop\_data, item\_shop\_data, potion\_data, monster\_inventory\_data, item\_inventory: , monster\_data: , user\_data: , user\_id)

{Fungsi untuk membeli monster yang ada di shop}

{I.S user memilih untuk membeli monster dan terdapat minimal 1 monster di monster\_shop}



{F.S user akan diminta untuk memilih id monster, jika OC tercukupi maka monster tersebut akan masuk ke inventory user}  
{F.S jika OC tidak tercukupi atau input id monster salah, maka pembayaran akan gagal}

function beli\_potion(username, monster\_shop\_data: , item\_shop\_data: , potion\_data:, monster\_inventory\_data :, item\_inventory: , monster\_data: , user\_data, user\_id, coin)  
{Fungsi untuk membeli potion yang ada di shop}  
{I.S user memilih untuk membeli potion dan terdapat minimal 1 potion di potion\_shop}  
{F.S user akan diminta untuk memilih id potion, jika OC tercukupi maka potion tersebut akan masuk ke inventory user}  
{F.S jika OC tidak tercukupi atau input id potion salah, maka pembayaran akan gagal}

function count\_char\_max(data\_list, kolom, header)  
{Fungsi untuk mengetahui karakter maksimal dalam kolom di list}  
{F.S Akan memberikan karakter maksimal yang di pakai dalam procedure lihat\_monster dan lihat\_potion}

procedure lihat\_monster(monster\_shop\_data, monster\_data)  
{Prosedur untuk melihat monster yang ada di shop}  
{I.S minimal terdapat 1 element monster dalam monster shop}  
{F.S menampilkan monster yang ada di monster shop beserta statsnya dan harganya}

procedure lihat\_potion(item\_shop\_list)  
{Prosedur untuk melihat potion yang ada di shop}  
{I.S minimal terdapat 1 element potion dalam shop}  
{F.S menampilkan potion yang ada di potion shop beserta statsnya dan harganya}

function item\_shop\_arr(item\_shop\_data,potion\_data)  
{Fungsi untuk membuat list item yang ada di item shop}  
{I.S parameter terpenuhi}  
{F.S memiliki variabel item\_shop\_list sebagai array yang berisi type, quantity, dan harga potion}

## 10. F-11 Laboratory

function laboratory(username, monster\_inventory\_data , user\_data , monster\_data)  
{Fungsi untuk memasuki laman laboratory}  
{I.S parameter terpenuhi}  
{F.S user akan diminta untuk memilih monster yang ingin diupgrade, jika input benar, maka monster akan terupgrade}  
{F.S jika OC-user kurang atau input user salah maka upgrade monster gagal}

function level\_price (level)

{Fungsi untuk menentukan harga upgrade level monster}

{I.S jika input valid}

{F.S price yang dibayar user akan sesuai dengan kenaikan level yang dimiliki oleh user, jika level monster kurang dari 5 maka price akan valid}

{F.S input tidak valid maka price fungsi ini tidak akan berjalan}

function display\_monster\_inventory(user\_id, monster\_inventory\_data, monster\_data)

{Prosedur untuk menampilkan monster yang dimiliki oleh user}

{I.S minimal memiliki 1 element monster dalam monster\_inventory\_data}

{F.S akan menampilkan monster apa saja yang dimiliki beserta levelnya}

procedure display\_upgrade\_price()

{Prosedur untuk menampilkan harga tiap upgrade level}

{I.S Menjalankan function laboratory}

{F.S Akan menampilkan harga untuk upgrade monster tiap levelnya}

## 11. F-12 Shop Management

function monster\_not\_in\_shop(monster\_shop\_data, monster\_data)

{Membuat list untuk monster yang tidak ada dalam shop}

function item\_not\_in\_shop(item\_shop)

{Membuat list untuk item yang tidak ada dalam shop}

function item\_shop\_arr(item\_shop\_data, potion\_data)

{Membuat list item yang terdapat dalam shop dan menambah id nya}

procedure tampilan\_awal(username, item\_shop, monster\_shop\_data, monster\_data, potion\_data)

{Memberikan tampilan awal pada shop management}

{F.S. Menampilkan sambutan bagi agen yang masuk ke shop management}

function memilih(username, item\_shop, monster\_shop\_data, monster\_data, potion\_data)

{Menampilkan menu opsi dalam shop management}

{I.S. input opsi valid}

{F.S. Menjalankan opsi yang dipilih (lihat/tambah/ubah/hapus/keluar)}

procedure lihat\_monster(monster\_shop\_data, monster\_data)

{Menghasilkan output data monster yang terdapat pada shop}  
{F.S. Menampilkan tabel berisi data monster yang terdapat pada shop}

procedure lihat\_potion(item\_shop\_list)

{Menghasilkan output data item yang terdapat pada shop}  
{F.S. Menampilkan tabel berisi data item yang terdapat pada shop}

function tambah\_monster(monster\_not\_in\_shop\_list, monster\_shop\_data, monster\_data)

{Menambah data monster yang belum terdaftar menjadi terdaftar pada shop}  
{I.S. Sembarang input, monster\_not\_in\_shop\_list tidak kosong}  
{F.S. Jika input tidak valid, maka akan mengulang untuk memasukkan input. Jika input valid, maka akan menambahkan data monster ke dalam shop}

function tambah\_potion(item\_not\_in\_shop\_list, item\_shop\_list, item\_shop\_data, potion\_data)

{Menambah data potion yang belum terdaftar menjadi terdaftar pada shop}  
{I.S. Sembarang input, item\_not\_in\_shop\_list tidak kosong}  
{F.S. Jika input tidak valid, maka akan diulang untuk memasukkan input. Jika valid, maka akan menambahkan data potion baru ke dalam shop}

function ubah\_monster(monster\_shop\_data, monster\_data)

{Mengubah data harga dan/atau stok pada monster}  
{I.S. Sembarang input}  
{F.S. Jika input tidak valid, maka dilakukan perulangan. Jika input ada dan valid, maka mengubah stok dan/atau harga pada data monster yang ada di shop}

function ubah\_potion (item\_shop\_list, item\_shop\_data)

{Mengubah data harga dan/atau stok pada potion}  
{I.S. Sembarang input}  
{F.S. Jika input tidak valid, maka dilakukan perulangan. Jika input ada dan valid, maka mengubah stok dan/atau harga pada potion yang ada di shop}

function hapus\_monster(monster\_shop\_data, monster\_data)

{Menghapus data monster beserta atribut nya dari shop}  
{I.S. Sembarang Id monster yang ingin dihapus}  
{F.S. Jika Id monster tidak valid, maka akan dilakukan perulangan. Jika valid, maka monster dengan Id tersebut beserta atributnya akan terhapus dari shop}

function hapus\_potion(item\_shop\_list, item\_shop\_data)

{Menghapus data potion beserta atributnya dari shop}

{I.S. Sembarang Id potion yang ingin dihapus}  
{F.S. Jika Id potion tidak valid, maka akan dilakukan perulangan. Jika valid, maka potion dengan Id tersebut beserta atributnya akan terhapus dari shop}

procedure keluar(username)  
{Keluar dari shop management}  
{F.S. Menampilkan selamat tinggal (username), lalu keluar dari shop management}

procedure delay()  
{Melakukan delay dan clear terminal}  
{F.S terminal akan clear}

function count\_char\_max(data\_list, kolom, header)  
{Mengetahui karakter maksimal dalam kolom}  
{F.S. Akan mengeluarkan karakter maksimal yang ada pada kolom}

procedure sort\_data(data, sortby)  
{Mengurutkan data yang ada sesuai urutan}

function get\_numeric\_input(prompt)  
{Mengecek input merupakan integer valid atau tidak}  
{I.S. Sembarang input}  
{F.S. Jika input valid, maka akan mengembalikan nilai input. Jika input tidak valid, maka akan melakukan perulangan}

function get\_numeric\_input(prompt, allow\_empty)-> str  
{Mengecek input merupakan integer valid atau tidak dan input kosong juga dianggap valid}  
{I.S. Sembarang input}  
{F.S. Jika input valid, maka akan mengembalikan nilai input. Jika input tidak valid, maka akan melakukan perulangan}

## 12. F-13 Monster Management

procedure delay()  
{Melakukan delay dan clear terminal}  
{F.S terminal akan clear}

procedure tampilan awal(monster\_list)

{Memberikan tampilan awal pada monster management}  
{F.S. Menampilkan sambutan bagi agen yang masuk ke monster management}

function pilihan\_monster\_management(monster\_list)  
{Memilih antara 3 aksi dalam menu utama monster management}  
{I.S. Input sembarang}  
{F.S. Jika input valid, maka akan dilanjut dengan fitur masing masing aksi yang dipilih.  
Jika input tidak valid, maka akan dilakukan perulangan}

function cek\_kesamaan\_nama(monster\_list, nama)  
{Mengecek nama yang sama dalam data}  
{I.S. Input nama sembarang}  
{F.S. Jika nama yang diinput sama dengan nama/type yang ada pada data, maka output bernilai true. Jika tidak, output bernilai false.

function tambah\_monster\_baru(monster\_list)  
{Menambah monster baru beserta atributnya}  
{I.S. Input nama, atk, defense, dan hp sembarang}  
{F.S. Jika input valid, maka program akan menambahkan menyimpan terlebih dahulu data monster baru yang diberikan. Jika input tidak valid, maka akan dilakukan perulangan}

function tambah\_monster\_ke\_database(monster\_list,monster\_baru)  
{Menambah monster yang telah dibuat ke database}  
{I.S. Input berupa pilihan y/n sembarang}  
{F.S. Jika input tidak valid, maka akan dilakukan perulangan. Jika input adalah y, maka monster baru berhasil ditambahkan ke database. Jika input adalah n, maka monster baru gagal ditambahkan ke database}

procedure count\_char\_max(data\_list, kolom, header)  
{Mengetahui karakter maksimal dalam kolom}  
{F.S. Akan mengeluarkan karakter maksimal yang ada pada kolom}

procedure lihat\_monster(monster\_data)  
{Menghasilkan output data monster yang terdapat pada database}  
{F.S. Menampilkan tabel berisi data monster yang berisi id, type, atk\_power, def\_power, dan hp}

### 13. F-14 Load

procedure load()  
{Melakukan load file pada data}  
{I.S dalam terminal perlu memberikan argumen file path data}  
{F.S akan mengeload file target}

procedure delay()  
{Prosedur untuk delay dan clear terminal}  
{F.S terminal akan di clear}

### 14. F-15 Save

procedure save(user, item\_inventories, item\_shop, monster, monster\_shop, monster\_inventory)  
{Melakukan prosedur save untuk menyimpan file game}  
{I.S semua parameter terpenuhi}  
{F.S File akan di save berdasarkan input nama dari user}

procedure data\_save(path, file\_name, data)  
{Melakukan sort data berdasarkan kriteria data dan menggabungkan list of dict menjadi string dengan format csv}  
{I.S procedure save dilakukan}  
{F.S File telah diurutkan dan di write di csv baru dengan folder sesuai dari input user}

procedure sort\_data(data, sortby)  
{Melakukan sort data berdasarkan kriteria tertentu}  
{I.S melakukan procedure data\_save}  
{F.S data akan terurut}

function is\_valid\_name(name)  
{Mengecek apakah input folder dari user valid}  
{I.S menjalankan procedure get\_valid\_input}  
{F.S Jika nama valid maka nama dapat digunakan sebagai nama folder}  
{F.S Jika nama tidak valid, maka nama tidak dapat digunakan dalam folder dan user diminta untuk input nama baru}

procedure get\_valid\_input(prompt)  
{Mengecek apakah input nama folder valid}  
{I.S Menjalankan procedure save}  
{F.S Jika nama valid maka nama dapat digunakan sebagai nama folder}

{F.S Jika nama tidak valid, maka nama tidak dapat digunakan dalam folder dan user diminta untuk input nama baru}

## **15. F-16 Exit**

procedure game\_exit(username, user, item\_inventories, item\_shop, monster, monster\_shop, monster\_inventory)

{Prosedur untuk exit dari game}

{F.S user akan memilih untuk save atau tidak, jika ya maka file akan di save dalam folder lalu akan keluar dari game}

{F.S jika tidak save, maka data game tidak akan tersimpan lalu akan keluar dari game}

## SPESIFIKASI PROGRAM

### 1. F-00 LCG

```
function LCG(lower_limit: integer, upper_limit: integer) -> integer
 const a <- 1664525
 const c <- 1013904223
 const m <- 2^32
 seed <- current time in seconds (as integer)
 seed <- (a * seed + c) % m

 while seed % (upper_limit + 1) < lower_limit do
 seed <- (a * seed + c) % m

 -> seed % (upper_limit + 1)
```

### 2. F-01 Register

```
function validate_username(username: string) -> boolean
 valid_characters <- set of
 "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz_-0123456789"
 for each char in username do
 if char not in valid_characters then
 -> False
 -> True
```

```
function choose_monster(monster_data: list, username: string) ->
integer
 output "Silahkan pilih salah satu monster sebagai monster awalmu."
 for each monster in monster_data do
 output monster['id'], ".", monster['type']

 ids <- empty list
 for each monster in monster_data do
 append int(monster['id']) to ids

 monster_id <- input("Monster pilihanmu: ") as integer

 if monster_id in ids then
 output ("Selamat datang agent ", username, ". Mari kita
mengalahkan Dr. Asep Spakbor dengan "),
monster_data[monster_id-1]['type'], "!"
 -> monster_id
 else
 output ("Monster pilihan anda tidak ada, silahkan pilih
kembali.")
 -> choose_monster(monster_data, username)
```



```

function check_register(username: string, password: string, user_data:
list) -> tuple
 if not validate_username(username) then
 output ("Username hanya boleh berisi alfabet, angka,
underscore, dan strip!")
 -> (False, False)

 for each data in user_data do
 if username = data['username'] then
 output ("Username ", username, " sudah terpakai, silahkan
gunakan username lain!")
 -> (False, False)

 -> (username, password)

```

```

function user_input(user_data: list) -> tuple
 username <- input ("Masukkan username: ")
 password <- input ("Masukkan password: ")
 -> check_register(username, password, user_data)

```

```

function register_page(game_state: integer, username: string,
monster_data: list, monster_inventory: list, user_data: list) -> tuple
 if game_state = 0 then
 (username, password) <- user_input(user_data)
 if username then
 monster_id <- choose_monster(monster_data, username)
 append {'id': str(len(user_data) + 1), 'username':
username, 'password': password, 'role': 'agent', 'oc': '0'} to
user_data
 append {'user_id': str(len(user_data)), 'monster_id':
str(monster_id), 'level': '1'} to monster_inventory
 game_state <- 1
 -> (username, game_state)
 else
 -> (username, game_state)
 else
 output ("Register gagal!\nAnda telah login dengan username ",
username, ", silahkan lakukan 'LOGOUT' sebelum melakukan register!")
 -> (username, game_state)

```

### 3. F-02 Login

```

function check_input(username: string, password: string, user_data:
list of dictionary) -> (integer, boolean, string)

```

```

for each data in user_data do
 if (username = data['username']) then
 if (password = data['password']) then
 output("Selamat datang, Agent ", username,
"!\\nMasukkan command 'help' untuk daftar command yang dapat kamu
panggil.")
 game_state <- 1
 if (data['role'] = 'admin') then
 is_admin <- True
 game_state <- 0
 -> (game_state, is_admin, username)
 else
 output("Password salah!")
 game_state <- 0
 username <- ''
 -> (game_state, is_admin, username)r

 output("Username tidak terdaftar!")
 game_state <- 0
 username <- ''
 -> (game_state, is_admin, username)

```

```

function user_login(data: list of dictionary) -> (integer, boolean,
string)

```

```

 input(username, "Username: ")
 input(password, "Password: ")

 -> check_input(username, password, data)

```

```

function login_page(game_state: integer, username: string, user_data:
list of dictionary) -> (integer, boolean, string)

```

```

 if (game_state = 0) then
 result <- user_login(user_data)
 -> result
 else
 output("Login gagal!\\nAnda telah login dengan username ",
username, ", silahkan lakukan 'LOGOUT' sebelum melakukan login
kembali!")
 -> (game_state, is_admin, username)

```

#### 4. F-03 Logout

```

function logout(game_state: integer, is_admin: boolean) -> (integer,

```

```

boolean)

 if (game_state = 1) or (is_admin) then
 output("Anda telah logout!")
 game_state <- 0
 is_admin <- False
 -> (game_state, is_admin)
 else
 output("Logout gagal!")
 output("Anda belum login, silahkan login terlebih dahulu
sebelum melakukan logout!")
 -> (game_state, is_admin)

```

## 5. F-04 Menu & Help

```

function help_menu(username: string, is_admin: boolean) -> None

 if (username = '') and (not is_admin) then
 pretext <- "Kamu belum login sebagai role apapun. Silahkan
login terlebih dahulu."
 main_text <- ""
 1. Login: Masuk ke dalam akun yang sudah terdaftar
 2. Register: Membuat akun baru
 3. Exit: keluar dari game
 ""
 elif (is_admin) then
 pretext <- "Halo admin! Selamat datang di menu bantuan. Kala
tak tau arah, menu inilah tujuan-mu"
 main_text <- ""
 Untuk dapat mengakses menu admin, anda perlu memasukkan
command spesial pada start menu
asepspakbortheboss ~~~ merupakan kata kunci tersebut!

 Di dalam menu admin akan terdapat:
 1. Shop Management: Untuk mengubah data dalam shop
 2. Monster Management: Untuk mengubah data monster
 ""
 else
 pretext <- "Halo Agent ", username, ". Kamu memanggil command
HELP. Kamu memilih jalan yang benar, semoga kamu tidak sesat kemudian.
Berikut adalah hal-hal yang dapat kamu lakukan sekarang"
 main_text <- ""
 1. Logout: Keluar dari akun yang sedang digunakan
 2. Menu: Memasuki main menu dan mulai petualanganmu!

 Dalam menu terdapat beberapa hal yang dapat kamu akses:

 1. Inventory : Untuk mengakses item/monster apa yang kamu

```

```

punya
 2. Battle : Suka tantangan? Lawan monster galak di sini!
 3. Arena : Tempat latihan menantang dengan stage tertentu!
 4. Laboratory : Upgrade monster-mu hingga ke level maksimal
 5. Shop : Beli monster/item untuk membantu petualangan-mu!
 ""

 output(pretext)
 output(main_text)
 input(cmd, "Tekan apapun untuk kembali: ")
 -> None

```

## 6. F-07 Inventory

```

procedure print_monster(cnt: integer, monster: dictionary) -> None

 name <- monster['type']
 level <- monster['level']
 hp <- atribute_by_level(int(monster['hp']), int(monster['level']))
 output(cnt, ".Monster (Name: ", name, ", Lvl: ", level, ",
HP: ", hp, ")")

```

```

procedure print_potion(cnt: integer, potion: dictionary) -> None

 potion_type <- potion['type']
 quantity <- potion['quantity']
 output(cnt, ".Potion (Type: ", potion_type, ", Qty: ",
quantity, ")")

```

```

procedure print_monster_ball(cnt: integer, monster_ball: dictionary)
-> None

 quantity <- monster_ball['quantity']
 output(cnt, ".Monster Ball (Qty: ", quantity, ")")

```

```

function display_inventory(player_inventory: list of dict, coin: int)
-> list of dict
 output("Jumlah O.W.C.A Coin-mu sekarang ", coin, "\n")
 for index from 1 to length(player_inventory) do
 item <- player_inventory[index - 1]
 if 'id' in item then
 call print_monster(index, item)

```

```

 else if 'quantity' in item then
 if item['type'] != 'monster ball' then
 print_potion(index, item)
 else
 print_monster_ball(index, item)
 -> print_details_by_id(player_inventory)

```

**procedure** print\_monster\_details(monster: dictionary) -> None

```

 output("Monster")
 output("Name : ", monster['type'])
 output("ATK Power : ",
attribute_by_level(int(monster['atk_power']), int(monster['level'])))
 output("DEF Power : ",
attribute_by_level(int(monster['def_power']), int(monster['level'])))
 output("HP : ", attribute_by_level(int(monster['hp']),
int(monster['level'])))
 output("Level : ", monster['level'])

```

**procedure** print\_potion\_details(potion: dictionary) -> None

```

if potion['type'] != 'Monster Ball' then
 output("Potion")
else
 output("Monster Balls")
 output("Type : ", potion['type'])
 output("Quantity : ", potion['quantity'])

```

**procedure** print\_details\_by\_id(data: list of dictionary) -> None

```

while True do
 output("!Input 'back' untuk kembali!")
 input(item_id, "Ketikkan id untuk menampilkan detail
item:\n>>> ")
 ids <- []
 for i from 1 to len(data) do
 append(ids, str(i))
 if item_id = 'back' then
 -> None
 else if item_id in ids then
 for each (index, item) in enumerate(data, start=1) do
 if index = int(item_id) then
 if 'id' in item then
 print_monster_details(item)
 else if 'quantity' in item then
 print_potion_details(item)
 else
 output("Id item tidak ada di inventory, gunakan Id lain.")

```

```
function attribute_by_level(attribute: integer, level: integer) ->
integer
 if level > 1 then
 attribute <- attribute + round(level * attribute * 0.1)
 -> attribute
```

## 7. F-08 Battle

```
procedure battle(username: string, monster_data: list of dict,
monster_inventory_data: list of dict, potion_data: list of dict,
item_inventory: list of dict, user_data: list of dict):
 player_inventory, coin <- pi.player_inventory(username, user_data,
monster_inventory_data, item_inventory, monster_data)
 rng_monster <- lcg.LCG(0, length(monster_data))
 monster_enemy <- monster_data
 desired_monster_enemy <- monster_enemy[rng_monster - 1]
 monster_enemy_level <- [1,2,3,4,5]
 rng_monster <- lcg.LCG(0, length(monster_enemy_level))
 level_monster_enemy <- monster_enemy_level[rng_monster - 1]
 monster_enemy_name <- desired_monster_enemy['type']
 attack_power_monster_enemy <-
attribute_by_level(int(desired_monster_enemy['atk_power']),
level_monster_enemy)
 defense_power_monster_enemy <-
attribute_by_level(int(desired_monster_enemy['def_power']),
level_monster_enemy)
 health_monster_enemy <-
attribute_by_level(int(desired_monster_enemy['hp']),
level_monster_enemy)
 health_monster_enemy_max <-
attribute_by_level(int(desired_monster_enemy['hp']),
level_monster_enemy)
```

```
 output("RAWRRR, Monster ", monster_enemy_name, " telah muncul
!!!")
```

```
 output(""
 /\----/_
 / \ /|
 | | O O | /|
 | | .vvvvv. | / /
 / | |_____| /
 / | ^^^^^^ /
 | /| | /
 / | |_____|
 \ | |_____|
 | | |_____|
 \.-\ \.-\

 """)
```

```

 print_hp_bar(monster_enemy_name, health_monster_enemy,
health_monster_enemy_max)
 output("Name : ", monster_enemy_name, "\nATK Power : ",
attack_power_monster_enemy, "\nDEF Power : ",
defense_power_monster_enemy, "\nHP : ", health_monster_enemy, "\nLevel
: ", level_monster_enemy, "\n")

 output("===== MONSTER LIST =====")
 monster_ids <- []
 for index, data in enumerate(player_inventory, start=1):
 if 'id' in data:
 output(index, ". ", data["type"], ", Level = ",
data["level"])
 monster_ids.append(str(index))

 user_monster_choice <- input("Pilih monster untuk bertarung: ")

 while user_monster_choice not in monster_ids:
 output("Pilihan monster tidak tersedia! Coba lagi!\n")
 user_monster_choice <- input("Pilih monster untuk bertarung:
")

 user_monster_choice <- int(user_monster_choice)
 user_monster_choice_list <-
player_inventory[int(user_monster_choice) - 1]
 user_monster_level <-
int(player_inventory[int(user_monster_choice) - 1]['level'])

 user_monster_choice_name <- user_monster_choice_list['type']
 user_monster_choice_attack_power <-
atribute_by_level(int(user_monster_choice_list['atk_power']),
user_monster_level)
 user_monster_choice_defense_power <-
atribute_by_level(int(user_monster_choice_list['def_power']),
user_monster_level)
 user_monster_choice_health_power <-
atribute_by_level(int(user_monster_choice_list['hp']),
user_monster_level)
 user_monster_choice_health_power_max <-
atribute_by_level(int(user_monster_choice_list['hp']),
user_monster_level)

 output("RAWRRR, Agent ", username, " mengeluarkan monster ",
user_monster_choice_name, " !!!\n")

 output(""
 /\-----/\
 /
 | O O | /\
 | .vvvvv. | |
 / | | | |
 / | ^^^^^^ | |

```





```

100))
 defense_enemy_rng <- math.floor(lcg.LCG(0,
user_monster_choice_defense_power))
 reduced_attack_enemy <- math.floor((100 - defense_enemy_rng) *
attack_enemy_rng / 100)

 output("===== TURN ", turn_round, " (",
user_monster_choice_name, ") =====")
 output("===== USER TURN =====")
 output("1. Attack\n2. Use Potion\n3. Quit")

 while True:
 try:
 user_choice <- input("Pilih perintah: ")
 break
 except ValueError:
 output("Input harus berupa bilangan! Ulangi")

 subprocess.run('cls', shell=True)
 if user_choice == 1:
 health_monster_enemy <- math.floor(health_monster_enemy -
reduced_attack_user)
 if health_monster_enemy <= 0:
 health_monster_enemy <- 0
 win <- True

 output("SKADIDODOO, ", user_monster_choice_name, "
menyerang ", monster_enemy_name, " !!!\n")
 output(""
 /\-----/_
 /
 | | o o | / |
 | | .vvvvv. | / |
 / | | ^^^^^^ | /
 / | | | /
 | / | | | /
 / | | |
 \ | | |
 | | | |
 \._\ \._\
 """)
 print_hp_bar(monster_enemy_name, health_monster_enemy,
health_monster_enemy_max)
 output("Name : ", monster_enemy_name, "\nATK Power : ",
attack_power_monster_enemy, "\nDEF Power : ",
defense_power_monster_enemy, "\nHP : ", health_monster_enemy, "\nLevel
: ", level_monster_enemy, "\n")
 output("# Penjelasan: ATT: ", attack_user_rng, ", Reduced
by: ", defense_user_rng, ", ATT Results: ", reduced_attack_user, "\n")

 if win:
 output("Selamat anda mengalahkan monster ",
monster_enemy_name, " !!!")

```



```

 if user_choice == 2 then
 output("===== POTION LIST =====")
 output(potion_list(strength_qty, resilience_qty,
healing_qty))

 Repeat
 Try
 user_potion_choice <- input("Pilih perintah: ")
 Break
 Catch ValueError
 output("Input anda salah! Ulangi.")

 if user_potion_choice is 1 then
 if strength_qty == 0 then
 output ("Wah, kamu sedang tidak memiliki ramuan ini,
silahkan pilih ramuan lain!")
 else
 user_monster_choice_attack_power <-
(strength_percentage + 100) / 100
 user_monster_choice_attack_power to
<-floor(user_monster_choice_attack_power)
 strength_qty -= 1
 output(user_monster_details(user_monster_choice_name,
user_monster_choice_attack_power, user_monster_choice_defense_power,
user_monster_choice_health_power, user_monster_level))
 output(
attack_power_increase_explanation(strength_percentage,
user_monster_choice_attack_power))

 else if user_potion_choice == 2 then
 if resilience_qty == 0 then
 output ("Wah, kamu sedang tidak memiliki ramuan ini,
silahkan pilih ramuan lain!")
 else
 user_monster_choice_defense_power <-
(resilience_percentage + 100) / 100
 user_monster_choice_defense_power <-
floor(user_monster_choice_defense_power)
 resilience_qty -= 1
 output(user_monster_details(user_monster_choice_name,
user_monster_choice_attack_power, user_monster_choice_defense_power,
user_monster_choice_health_power, user_monster_level))
 output(
defense_power_increase_explanation(resilience_percentage,
user_monster_choice_defense_power))

 else if user_potion_choice == 3 then
 if healing_qty == 0 then
 output ("Wah, kamu sedang tidak memiliki ramuan ini,
silahkan pilih ramuan lain!")
 else
 user_monster_choice_health_power<-(healing_percentage
+ 100) / 100

```

```

 user_monster_choice_health_power <-
floor(user_monster_choice_health_power)
 healing_qty -= 1
 output(user_monster_details(user_monster_choice_name,
user_monster_choice_attack_power, user_monster_choice_defense_power,
user_monster_choice_health_power, user_monster_level))
 output
(health_power_increase_explanation(healing_percentage,
user_monster_choice_health_power))

 if user_monster_choice_health_power >
user_monster_choice_health_power_max then
 user_monster_choice_health_power <
-user_monster_choice_health_power_max

 else if user_potion_choice == 4 then
 over <- False

 else
 output("Perintah yang diberikan tidak tersedia, coba
ulangi!")

 else if user_choice == 3 then
 output("Anda berhasil kabur dari BATTLE !")
 Break

 else
 output("Perintah yang diberikan tidak tersedia, coba
ulangi!")

 update_user_data(username, item_inventory, user_data,
strength_qty, resilience_qty, healing_qty, coin)

```

```

function attribute_by_level(attribute: integer, level: integer) ->
integer
{ Menyesuaikan atribut monster berdasarkan level }
if level > 1 then
 attribute <- attribute + round(level * attribute * 0.1)
-> attribute

```

```

procedure update_user_data(username: string, item_inventory: list of
dictionary, user_data: list of dictionary, qty_str: integer, qty_res:
integer, qty_heal: integer, coin: integer)
{ Update data dari user setelah game }
for each data in user_data do
 if username = data['username'] then
 user_id <- data['id']
 data['oc'] <- str(coin)

```

```

for each data in item_inventory do
 if user_id = data['user_id'] then
 if data['type'] = 'strength' then
 data['quantity'] <- str(qty_str)
 if data['type'] = 'resilience' then
 data['quantity'] <- str(qty_res)
 if data['type'] = 'healing' then
 data['quantity'] <- str(qty_heal)

```

```

procedure print_hp_bar(name: string, hp: integer, max_hp: integer,
bar_length: integer = 50)
{ Mengeprint bar hp monster }
 filled_length <- int(bar_length * hp / max_hp)
 bar <- repeat('█', filled_length) + repeat('-', bar_length -
filled_length)

```

```

output(name, " HP: [" , bar, "]" , hp, "/" , max_hp)

```

## 8. F-09 Arena

```

procedure battle(username: string, monster_data: list of dict,
monster_inventory_data: list of dict, potion_data: list of dict,
item_inventory: list of dict, user_data: list of dict):
 player_inventory, coin <- pi.player_inventory(username, user_data,
monster_inventory_data, item_inventory, monster_data)

```

```

output("===== MONSTER LIST =====")
monster_ids <- []
for index, data in enumerate(player_inventory, start=1):
 if 'id' in data:
 output(index, ". ", data["type"], ", Level = ",
data["level"])
 monster_ids.append(str(index))

user_monster_choice <- input("Pilih monster untuk bertarung: ")

while user_monster_choice not in monster_ids:
 output("Pilihan monster tidak tersedia! Coba lagi!\n")
 user_monster_choice <- input("Pilih monster untuk bertarung: ")

user_monster_choice <- int(user_monster_choice)
user_monster_choice_list <-
player_inventory[int(user_monster_choice) - 1]
user_monster_level <-
int(player_inventory[int(user_monster_choice) - 1]['level'])

```



```

for strength_ in potion_data:
 if strength_['potion_name'] == 'strength':
 strength_percentage <- int(strength_['percentage'])
for resilience_ in potion_data:
 if resilience_['potion_name'] == 'resilience':
 resilience_percentage <- int(resilience_['percentage'])
for healing_ in potion_data:
 if healing_['potion_name'] == 'resilience':
 healing_percentage <- int(healing_['percentage'])
clear <- False
stage_round <- 1
turn_round <- 1
oc_prize <- 0
total_attack_user <- 0
total_attack_enemy <- 0

while not clear do
 output("===== STAGE ", stage_round, " =====")

 rng_monster <- lcg.LCG(0, len(monster_data))
 desired_monster_enemy <- monster_data[rng_monster - 1]
 level_monster_enemy <- stage_round

 monster_enemy_name <- desired_monster_enemy['type']
 attack_power_monster_enemy <-
atribute_by_level(desired_monster_enemy['atk_power'],
level_monster_enemy)
 defense_power_monster_enemy <-
atribute_by_level(desired_monster_enemy['def_power'],
level_monster_enemy)
 health_monster_enemy <-
atribute_by_level(desired_monster_enemy['hp'], level_monster_enemy)
 health_monster_enemy_max <-
atribute_by_level(desired_monster_enemy['hp'], level_monster_enemy)

 output("RAWRRR, Agent ", username, " mengeluarkan monster ",
user_monster_choice['name'], " !!!")
 output(user_monster_choice['ascii_art'])

 print_hp_bar(user_monster_choice['name'],
user_monster_choice['health_power'],
user_monster_choice['health_power_max'])
 output("Name: ", user_monster_choice['name'])
 output("ATK Power: ", user_monster_choice['attack_power'])
 output("DEF Power: ", user_monster_choice['defense_power'])
 output("HP: ", user_monster_choice['health_power'])
 output("Level: ", user_monster_choice['level'])

 output("RAWRRR, Monster ", monster_enemy_name, " telah muncul
!!!")
 output(monster_enemy['ascii_art'])

```

```

 print_hp_bar(monster_enemy_name, health_monster_enemy,
health_monster_enemy_max)
 output("Name: ", monster_enemy_name)
 output("ATK Power: ", attack_power_monster_enemy)
 output("DEF Power: ", defense_power_monster_enemy)
 output("HP: ", health_monster_enemy)
 output("Level: ", level_monster_enemy)

 over <- False
 while not over do
 attack_user_rng <- math.floor(lcg.LCG(30 *
user_monster_choice['attack_power'] / 100, 70 *
user_monster_choice['attack_power'] / 100))
 defense_user_rng <- math.floor(lcg.LCG(0,
defense_power_monster_enemy))
 reduced_attack_user <- math.floor((100 - defense_user_rng)
* attack_user_rng / 100)

 attack_enemy_rng <- math.floor(lcg.LCG(30 *
attack_power_monster_enemy / 100, 70 * attack_power_monster_enemy /
100))
 defense_enemy_rng <- math.floor(lcg.LCG(0,
user_monster_choice['defense_power']))
 reduced_attack_enemy <- math.floor((100 -
defense_enemy_rng) * attack_enemy_rng / 100)

 output("===== TURN ", turn_round, " (",
user_monster_choice['name'], ") =====")
 output("===== USER TURN =====")
 output("1. Attack\n2. Use Potion\n3. Quit")

 while True do
 try
 user_choice <- int(input("Pilih perintah: "))
 break
 except ValueError
 output("Input anda salah! Ulangi.")

 if user_choice == 1 then
 health_monster_enemy <-
math.floor(health_monster_enemy - reduced_attack_user)
 total_attack_user <- total_attack_user +
reduced_attack_user

 if health_monster_enemy <= 0 then
 health_monster_enemy <- 0
 win <- True
 output(monster_enemy['ascii_art'])
 output("SKADIDODOO, ", user_monster_choice['name'], "
menyerang ", monster_enemy_name, " !!!")
 print_hp_bar(monster_enemy_name, health_monster_enemy,
health_monster_enemy_max)
 output("Name: ", monster_enemy_name)

```



```

 output("ATK Power: ", attack_power_monster_enemy)
 output("DEF Power: ", defense_power_monster_enemy)
 output("HP: ", health_monster_enemy)
 output("Level: ", level_monster_enemy)
 output("# Penjelasan: ATT: ", attack_user_rng, ",
Reduced by: ", defense_user_rng, ", ATT Results: ",
reduced_attack_user)

 if win then
 output("Selamat anda mengalahkan monster ",
monster_enemy_name, " !!!")
 over <- True
 clear <- True
 break

 output("===== TURN ", turn_round, " (",
monster_enemy_name, ") =====")
 output("===== ENEMY TURN =====")
 user_monster_choice['health_power'] <-
math.floor(user_monster_choice['health_power'] - reduced_attack_enemy)
 total_attack_enemy <- total_attack_enemy +
reduced_attack_enemy

 if user_monster_choice['health_power'] <= 0 then
 user_monster_choice['health_power'] <- 0
 lose <- True

 output(user_monster_choice['ascii_art'])
 output("SKADLIDODOR, ", monster_enemy_name, "
menyerang ", user_monster_choice['name'], " !!!")
 print_hp_bar(user_monster_choice['name'],
user_monster_choice['health_power'],
user_monster_choice['health_power_max'])
 output("Name: ", user_monster_choice['name'])
 output("ATK Power: ",
user_monster_choice['attack_power'])
 output("DEF Power: ",
user_monster_choice['defense_power'])
 output("HP: ", user_monster_choice['health_power'])
 output("Level: ", user_monster_choice['level'])
 output("# Penjelasan: ATT: ", attack_enemy_rng, ",
Reduced by: ", defense_enemy_rng, ", ATT Results: ",
reduced_attack_enemy)

 if lose then
 output("Tidakkk, kamu telah dikalahkan oleh ",
monster_enemy_name, " !!!")
 over <- True
 clear <- True
 turn_round <- turn_round + 1

 elif user_choice == 2 then
 output("===== POTION LIST =====")

```

```

 output("1. Strength Potion (Qty: ", strength_qty, ") -
Increases ATK Power")
 output("2. Resilience Potion (Qty: ", resilience_qty,
") - Increases DEF Power")
 output("3. Healing Potion (Qty: ", healing_qty, ") -
Restores Health")
 output("4. Cancel")

 while True do
 try
 user_potion_choice <- int(input("Pilih
perintah: "))
 break
 except ValueError
 output("Input anda salah! Ulangi.")

 if user_potion_choice == 1 then
 if strength_qty == 0 then
 output("Wah, kamu sedang tidak memiliki ramuan
ini, silahkan pilih ramuan lain!")
 else
 user_monster_choice['attack_power'] <-
user_monster_choice['attack_power'] * (strength_percentage + 100) /
100

 strength_qty <- strength_qty - 1
 output("Name: ", user_monster_choice['name'])
 output("ATK Power: ",
user_monster_choice['attack_power'])
 output("DEF Power: ",
user_monster_choice['defense_power'])
 output("HP: ",
user_monster_choice['health_power'])
 output("Level: ",
user_monster_choice['level'])
 output("# Penjelasan: Attack power monster
bertambah sebesar ", strength_percentage, "% menjadi: ",
user_monster_choice['attack_power'])

 elif user_potion_choice == 2 then
 if resilience_qty == 0 then
 output("Wah, kamu sedang tidak memiliki ramuan
ini, silahkan pilih ramuan lain!")
 else
 user_monster_choice['defense_power'] <-
user_monster_choice['defense_power'] * (resilience_percentage + 100) /
100

 resilience_qty <- resilience_qty - 1
 output("Name: ", user_monster_choice['name'])
 output("ATK Power: ",
user_monster_choice['attack_power'])
 output("DEF Power: ",
user_monster_choice['defense_power'])
 output("HP: ",

```

```

user_monster_choice['health_power'])
 output("Level: ",
user_monster_choice['level'])
 output("# Penjelasan: Defense power monster
bertambah sebesar ", resilience_percentage, "% menjadi: ",
user_monster_choice['defense_power'])

 elif user_potion_choice == 3 then
 if healing_qty == 0 then
 output("Wah, kamu sedang tidak memiliki ramuan
ini, silahkan pilih ramuan lain!")
 else
 user_monster_choice['health_power'] <-
user_monster_choice['health_power'] * (healing_percentage + 100) / 100
 healing_qty <- healing_qty - 1
 output("Name: ", user_monster_choice['name'])
 output("ATK Power: ",
user_monster_choice['attack_power'])
 output("DEF Power: ",
user_monster_choice['defense_power'])
 output("HP: ",
user_monster_choice['health_power'])
 output("Level: ",
user_monster_choice['level'])
 output("# Penjelasan: Health power monster
bertambah sebesar ", healing_percentage, "% menjadi: ",
user_monster_choice['health_power'])

 elif user_potion_choice == 4 then
 output("Membatalkan penggunaan potion")

 elif user_choice == 3 then
 over <- True
 clear <- True
 output("Keluar dari pertarungan")
 if lose then # Jika kalah dalam battle di arena
 output(f'GAME OVER! Sesi latihan berakhir pada stage
{stage_round}')
 output('===== STATS =====')
 output(f'OC Prize : {oc_prize}\nJumlah stage :
{stage_round}\nDamage diberikan :
{math.floor(total_attack_user)}\nDamage diterima :
{math.floor(total_attack_enemy)}\n')
 coin <- int(coin)
 coin <- coin + oc_prize
 cmd <- input('Tekan apapun untuk keluar: ')

 elif stage_round >= 5 then # Jika stage telah selesai
 output('Selamat! Anda telah menyelesaikan semua stage')
 output('===== STATS =====')
 output(f'OC Prize : {oc_prize}\nJumlah stage :
{stage_round}\nDamage diberikan :
{math.floor(total_attack_user)}\nDamage diterima :

```

```

{math.floor(total_attack_enemy)}\n')
 coin <- int(coin)
 coin <- coin + oc_prize
 clear <- True
 cmd <- input('Tekan apapun untuk keluar: ')

 elif win then # Jika menang dalam battle di arena
 output('Menuju ke stage berikutnya....')
 over <- False
 clear <- False
 lose <- False
 win <- False
 user_monster_choice['health_power'] <-
user_monster_choice['health_power_max']
 health_monster_enemy <- health_monster_enemy_max
 oc_prize <- oc_prize + 30
 stage_round <- stage_round + 1
 turn_round <- 1

 update_user_data(username, item_inventory, user_data,
strength_qty, resilience_qty, healing_qty, coin)

```

```

function attribute_by_level(attribute: integer, level: integer) ->
integer
{ Menyesuaikan atribut monster berdasarkan level }
if level > 1 then
 attribute <- attribute + round(level * attribute * 0.1)
-> attribute

```

```

procedure update_user_data(username: string, item_inventory: list of
dictionary, user_data: list of dictionary, qty_str: integer, qty_res:
integer, qty_heal: integer, coin: integer)
{ Update data dari user setelah game }
for each data in user_data do
 if username = data['username'] then
 user_id <- data['id']
 data['oc'] <- str(coin)

 for each data in item_inventory do
 if user_id = data['user_id'] then
 if data['type'] = 'strength' then
 data['quantity'] <- str(qty_str)
 if data['type'] = 'resilience' then
 data['quantity'] <- str(qty_res)
 if data['type'] = 'healing' then
 data['quantity'] <- str(qty_heal)

```

```

procedure print_hp_bar(name: string, hp: integer, max_hp: integer,
bar_length: integer = 50)
{ Mengeprint bar hp monster }
filled_length <- int(bar_length * hp / max_hp)

```

```
bar <- repeat('█', filled_length) + repeat('-', bar_length -
filled_length)
```

```
output(name, " HP: [", bar, "]" ", hp, "/", max_hp)
```

```
procedure delay() -> None:
 time.sleep(1)
 if os.name = 'nt' then
 os.system('cls')
 else
 os.system('clear')
```

## 9. F-10 Shop & Currency

```
procedure delay() -> None:
 time.sleep(1)
 if os.name = 'nt' then
 os.system('cls')
 else
 os.system('clear')
```

```
function shop_currency_page(username: string, monster_shop_data: list
of dictionaries, item_shop_data: list of dictionaries, potion_data:
list of dictionaries, monster_inventory_data: list of dictionaries,
item_inventory: list of dictionaries, monster_data: list of
dictionaries, user_data: list of dictionaries) -> None:
```

```
 output 'Irrasshaimase! Selamat datang di SHOP!!'
 cmd <- input('Pilih aksi (lihat/beli/keluar): ')
 for each data in user_data do
 if username = data['username'] then
 user_id <- data['id']
 if cmd = 'lihat' then
 -> lihat(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data)
 elif cmd = 'beli' then
 -> beli(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data, user_id)
 elif cmd = 'keluar' then
 output ('Selamat tinggal ', username, '! Sampai bertemu
lagi.')
 else
 output ('Perintah anda salah! Ulangi perintah anda.')
 -> shop_currency_page(username, monster_shop_data,
item_shop_data, potion_data, monster_inventory_data, item_inventory,
monster_data, user_data)
```

```
function lihat(username: string, monster_shop_data: list of
```

```

dictionaries, item_shop_data: list of dictionaries, potion_data: list
of dictionaries, monster_inventory_data: list of dictionaries,
item_inventory: list of dictionaries, monster_data: list of
dictionaries, user_data: list of dictionaries) -> None:
 cmd <- input('Mau lihat apa? (monster/potion)? : ')
 if cmd == 'monster' then
 -> lihat_monster(monster_shop_data, monster_data)
 -> shop_currency_page(username, monster_shop_data,
item_shop_data, potion_data, monster_inventory_data, item_inventory,
monster_data, user_data)
 elif cmd == 'potion' then
 item_shop_list <- item_shop_arr(item_shop_data, potion_data)
 -> lihat_potion(item_shop_list)
 -> shop_currency_page(username, monster_shop_data,
item_shop_data, potion_data, monster_inventory_data, item_inventory,
monster_data, user_data)
 else
 output ('Perintah anda salah! Ulangi perintah anda.')
 -> delay()
 -> lihat(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data)

```

```

function beli(username: string, monster_shop_data: list of
dictionaries, item_shop_data: list of dictionaries, potion_data: list
of dictionaries, monster_inventory_data: list of dictionaries,
item_inventory: list of dictionaries, monster_data: list of
dictionaries, user_data: list of dictionaries, user_id: string) ->
None:

```

```

 for each data in user_data do
 if username == data['username'] then
 coin <- data['oc']
 output 'Jumlah O.W.C.A coin mu sekarang ', coin
 cmd <- input('Mau beli apa? (monster/potion): ')
 if cmd == 'monster' then
 -> beli_monster(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data, user_id, coin)
 elif cmd == 'potion' then
 -> beli_potion(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data, user_id, coin)
 else
 output ('Perintah anda salah! Ulangi perintah anda!')
 -> beli(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data, user_id)

```

```

function beli_monster(username: string, monster_shop_data: list of
dictionaries, item_shop_data: list of dictionaries, potion_data: list
of dictionaries, monster_inventory_data: list of dictionaries,
item_inventory: list of dictionaries, monster_data: list of
dictionaries, user_data: list of dictionaries, user_id: string, coin:

```

```

string) -> None:
 coin <- convert coin to integer
 ids_list <- []
 for each data in monster_shop_data do
 ids_list.append(data['monster_id'])
 monster_id <- input('Masukkan monster id: ')
 if monster_id not in ids_list then
 output 'Id tidak tersedia, silahkan coba id yang lain.'
 -> delay()
 -> shop_currency_page(username, monster_shop_data,
item_shop_data, potion_data, monster_inventory_data, item_inventory,
monster_data, user_data)
 for each data in monster_data do
 for each subdata in monster_shop_data do
 if monster_id = data['id'] then
 monster_type <- data['type']
 monster_cost <- convert subdata['price'] to integer
 monster_stock <- convert subdata['stock'] to integer
 for each data in monster_inventory_data do
 if monster_id = data['monster_id'] and user_id =
data['user_id'] then
 output ('Monster', monster_type, ', sudah ada dalam
inventory-mu! Pembelian dibatalkan.')
 -> shop_currency_page(username, monster_shop_data,
item_shop_data, potion_data, monster_inventory_data, item_inventory,
monster_data, user_data)
 if monster_stock = 0 then
 output ('Stock monster sudah habis, silahkan pilih yang lain')
 elif monster_cost > coin then
 output ('OC-mu tidak cukup')
 else
 append {'user_id': user_id, 'monster_id': monster_id, 'level':
'1'} to monster_inventory_data
 output 'Berhasil membeli item', monster_type, '. Item sudah
masuk ke inventory-mu!'
 for each subdata in monster_shop_data do
 if monster_id = subdata['monster_id'] then
 subdata['stock'] <- convert monster_stock-1 to string
 for each data in user_data do
 if user_id = data['id'] then
 data['oc'] <- convert coin-monster_cost to string
 -> shop_currency_page(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data)

function beli_potion(username: string, monster_shop_data: list of
dictionaries, item_shop_data: list of dictionaries, potion_data: list
of dictionaries, monster_inventory_data: list of dictionaries,
item_inventory: list of dictionaries, monster_data: list of
dictionaries, user_data: list of dictionaries, user_id: string, coin:
string) -> None:
 coin <- convert coin to integer

```

```

ids_list <- []
for each data in item_shop_data do
 for each subdata in potion_data do
 if data['type'] = subdata['potion_name'] then
 ids_list.append(subdata['id'])
item_id <- input('Masukkan id potion: ')
if item_id not in ids_list then
 output 'Id tidak tersedia, silahkan coba id yang lain.'
 -> delay()
 -> shop_currency_page(username, monster_shop_data,
item_shop_data, potion_data, monster_inventory_data, item_inventory,
monster_data, user_data)
 while True do
 try
 qty <- convert input to integer
 if qty > 0 then
 break
 else
 output ('Input harus bilangan positif dan lebih dari
0!')
 except ValueError do
 output ('Input harus berupa bilangan bulat! Ulangi')
 for each data in potion_data do
 for each subdata in item_shop_data do
 if item_id = data['id'] and data['potion_name'] =
subdata['type'] then
 item_type <- subdata['type']
 item_cost <- convert subdata['price'] to integer
 item_stock <- convert subdata['stock'] to integer
 if item_stock = 0 then
 output ('Stock item sudah habis, silahkan pilih yang lain')
 elif item_cost*qty > coin then
 output ('OC-mu tidak cukup')
 elif qty > item_stock then
 output ('Kuantitas barang tidak mencukupi, silahkan ulangi')
 else
 output ('Berhasil membeli item:', qty, item_type, '. Item
sudah masuk ke inventory-mu!')
 for each subdata in item_shop_data do
 if item_type = subdata['type'] then
 subdata['stock'] <- convert item_stock-qty to string
 for each data in user_data do
 if user_id = data['id'] then
 data['oc'] <- convert coin-(qty*item_cost) to string
 temp <- []
 for each data in item_inventory do
 if user_id = data['user_id'] and item_type = data['type']
then
 data['quantity'] <- convert int(data['quantity'])+qty
to string
 temp.append(data['type'])
 if not temp then

```



```

 append {'user_id': user_id, 'type': item_type, 'quantity':
qty} to item_inventory
 -> delay()
 -> shop_currency_page(username, monster_shop_data, item_shop_data,
potion_data, monster_inventory_data, item_inventory, monster_data,
user_data)

function count_char_max(data_list: list of dictionaries, kolom:
string, header: string) -> integer:
 char_max <- length of header
 for each i in range(length of data_list) do
 if length of data_list[i][kolom] > char_max then
 char_max <- length of data_list[i][kolom]
 -> char_max

procedure lihat_monster(monster_shop_data: list of dictionaries,
monster_data: list of dictionaries) -> None:
 list_of_len <- [count_char_max(monster_shop_data, 'monster_id',
'ID'), count_char_max(monster_data, 'type', 'Name/Type'),
count_char_max(monster_data, 'atk_power', 'ATK Power'),
count_char_max(monster_data, 'def_power', 'DEF Power'),
count_char_max(monster_data, 'hp', 'HP'),
count_char_max(monster_shop_data, 'stock', 'Stok'),
count_char_max(monster_shop_data, 'price', 'Harga')]
 output ('ID | Name/Type | ATK Power | DEF Power | HP | Stok |
Harga')
 output
 '-----'
 for each data in monster_shop_data do
 for each subdata in monster_data do
 if subdata['id'] = data['monster_id'] then
 output (subdata['id'], ' | ', subdata['type'], ' | ',
subdata['atk_power'], ' | ', subdata['def_power'], ' | ',
subdata['hp'], ' | ', data['stock'], ' | ', data['price'])

procedure lihat_potion(item_shop_list: list of dictionaries) -> None:
 list_of_len <- [count_char_max(item_shop_list, 'id', 'ID'),
count_char_max(item_shop_list, 'type', 'Name/Type'),
count_char_max(item_shop_list, 'stock', 'Stok'),
count_char_max(item_shop_list, 'price', 'Harga')]
 output ('ID | Name/Type | Stok | Harga')
 output ('-----')
 for each data in item_shop_list do
 output (data['id'], ' | ', data['type'], ' | ', data['stock'], ' |
', data['price'])

function item_shop_arr(item_shop_data: list of dictionaries,
potion_data: list of dictionaries) -> list of dictionaries:
 item_shop_list <- []
 for each data in potion_data do
 for each subdata in item_shop_data do
 if subdata['type'] = data['potion_name'] then

```

```

 item_shop_list.append({'id': data['id'], 'type':
subdata['type'], 'stock': subdata['stock'], 'price':
subdata['price']})
 -> item_shop_list

```

## 10. F-11 Laboratory

```

procedure laboratory(username: string, monster_inventory_data: list of
dictionary, user_data: list of dictionary, monster_data: list of
dictionary)
 for each name in user_data do
 if username = name['username'] then
 user_id <- name['id']
 coin <- int(name['oc'])

 output("Selamat datang di laboratory ", username, "!!")
 output("Tekan 0 untuk kembali ke menu utama.")

 monster_list <- display_monster_inventory(user_id,
monster_inventory_data, monster_data)
 display_upgrade_price()

 try
 input(monster_id, "Pilih monster: ")
 monster_id <- int(monster_id)

 if 1 <= monster_id <= length(monster_list) then
 selected_monster <- monster_list[monster_id - 1]
 monster_id, monster_name, level <- selected_monster
 level <- int(level)
 price <- level_price(level)

 if level >= 5 then
 output("Maaf, monster yang Anda pilih sudah memiliki
level maksimum")
 call delay()
 -> laboratory(username, monster_inventory_data,
user_data, monster_data)
 else
 level_up <- level + 1
 output(monster_name, " akan di upgrade ke level ",
level_up, ".")
 input(confirm, "Lanjutkan upgrade (Y/N): ")
 confirm <- lower(confirm)

 if confirm = 'y' then
 if coin >= price then
 coin <- coin - price
 level <- level_up
 output("Selamat, ", monster_name, " berhasil

```

```

di-upgrade ke level ", level_up, "!")
 output("Jumlah O.W.C.A. Coin kamu sekarang ",
coin, ".")

 // Update jumlah coin
 for each user in user_data do
 if user['id'] = user_id then
 user['oc'] <- str(coin)

 // Update data level
 for each monster in monster_inventory_data do
 if monster['user_id'] = user_id and
monster['monster_id'] = monster_id then
 monster['level'] <-str(level)

 call delay()
 -> laboratory(username,
monster_inventory_data, user_data, monster_data)
 else
 output("Maaf, Anda tidak memiliki cukup
O.W.C.A. Coin untuk melakukan upgrade.")
 else if confirm = 'n' then
 output("Upgrade monster dibatalkan.")
 else
 output("Masukkan tidak valid")
 call delay()
 -> laboratory(username, monster_inventory_data,
user_data, monster_data)

 else if monster_id = 0 then
 output("Terimakasih telah berkunjung! Sampai bertemu
kembali!")
 call delay()
 else
 output("Tidak ada monster tersebut")
 call delay()
 -> laboratory(username, monster_inventory_data, user_data,
monster_data)
 except
 output("Input harus berupa bilangan bulat")
 call delay()
 -> laboratory(username, monster_inventory_data, user_data,
monster_data)

```

```

function level_price(level: integer) -> integer
 if level = 1 then
 price <- 300
 else if level = 2 then
 price <- 500
 else if level = 3 then
 price <- 800
 else if level = 4 then

```

```

 price <- 1000
 else if level >= 5 then
 price <- 9999
 -> price

```

```

function display_monster_inventory(user_id: string,
monster_inventory_data: list of dictionary, monster_data: list of
dictionary) -> list of tuple
 output("===== MONSTER LIST =====")
 index <- 1
 monster_list <- []

 temp <- []
 for each monster in monster_data do
 append(temp, {'id': monster['id'], 'type': monster['type'],
'atk_power': monster['atk_power'], 'def_power': monster['def_power'],
'hp': monster['hp']})

 for each data in monster_inventory_data do
 if user_id = data['user_id'] then
 level <- int(data['level'])
 monster_id <- data['monster_id']
 for each subdata in temp do
 if monster_id = subdata['id'] then
 monster_name <- subdata['type']
 output(index, ". ", monster_name, " (Level: ",
level, ")")
 append(monster_list, (monster_id, monster_name,
level))
 index <- index + 1

 -> monster_list

```

```

procedure display_upgrade_price()
 output("===== UPGRADE PRICE =====")
 output("1. Level 1 -> Level 2: 300 OC")
 output("2. Level 2 -> Level 3: 500 OC")
 output("3. Level 3 -> Level 4: 800 OC")
 output("4. Level 4 -> Level 5: 1000 OC")

```

## 11. F-12 Shop Management

```

function monster_not_in_shop(monster_shop_data: list of dictionary,
monster_data: list of dictionary) -> list of dictionary
 monster_not_in_shop_list <- []
 id_monster_in_shop <- []

```

```

for each data in monster_shop_data do
 append data['monster_id'] to id_monster_in_shop
for each data in monster_data do
 if data['id'] not in id_monster_in_shop then
 append data to monster_not_in_shop_list
-> monster_not_in_shop_list

```

**function** item\_not\_in\_shop(item\_shop: list of dictionary) -> list of string

```

item_list <- ['healing', 'resilience', 'strength']
item_type_in_shop <- []
for each data in item_shop do
 append data['type'] to item_type_in_shop
item_not_in_shop_list <- []
for each item in item_list do
 if item not in item_type_in_shop then
 append item to item_not_in_shop_list
-> item_not_in_shop_list

```

**function** item\_shop\_arr(item\_shop\_data: list of dictionary, potion\_data: list of dictionary) -> list of dictionary

```

item_shop_list <- []
for each data in potion_data do
 for each subdata in item_shop_data do
 if data['potion_name'] is equal to subdata['type'] then
 append {'id': data['id'], 'type': subdata['type'],
'stock': subdata['stock'], 'price': subdata['price']} to
item_shop_list
-> item_shop_list

```

**procedure** tampilan\_awal(username: string, item\_shop: list of dictionary, monster\_shop\_data: list of dictionary, monster\_data: list of dictionary, potion\_data: list of dictionary) -> None

```

 output("Halo ", username, ", Selamat datang kembali!")
 output("~"*22)
 output("Terdapat beberapa pilihan aksi :")
 memilih(username, item_shop, monster_shop_data, monster_data,
potion_data)

```

**function** memilih(username: string, item\_shop\_data: list of dictionary, monster\_shop\_data: list of dictionary, monster\_data: list of dictionary, potion\_data: list of dictionary) -> None

```

 monster_not_in_shop_list <- monster_not_in_shop(monster_shop_data,
monster_data)
 item_not_in_shop_list <- item_not_in_shop(item_shop_data)
 item_shop_list <- item_shop_arr(item_shop_data, potion_data)
 output("[1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar ")
 pilihan <- input("Mau pilih aksi yang mana? : ")
 if pilihan is "lihat" or pilihan is "1" then

```

```

// Menu Lihat
while True do
 jenis <- input("Mau lihat apa? (monster/potion) : ")
 if jenis is "monster" then
 lihat_monster(monster_shop_data, monster_data)
 break
 else if jenis is "potion" then
 lihat_potion(item_shop_list)
 break
 else
 output("Ups, Pilih antara monster atau potion!")
else if pilihan is "tambah" or pilihan is "2" then
 // Menu Tambah
 while True do
 jenis <- input("Mau tambah apa? (monster/potion) : ")
 if jenis is "monster" then
 if length of monster_not_in_shop_list > 0 then
 tambah_monster(monster_not_in_shop_list,
monster_shop_data, monster_data)
 else
 output('Semua monster telah ada di shop.')
 break
 else if jenis is "potion" then
 if length of item_not_in_shop_list > 0 then
 tambah_potion(item_not_in_shop_list,
item_shop_list, item_shop_data, potion_data)
 else
 output('Semua item telah berada di shop.')
 break
 else
 output("Ups, Pilih antara monster atau potion!")
else if pilihan is "ubah" or pilihan is "3" then
 // Menu Ubah
 while True do
 jenis <- input("Mau ubah apa? (monster/potion) : ")
 if jenis == "monster" then
 if length of monster_shop_data > 0 then
 ubah_monster(monster_shop_data, monster_data)
 else
 output('Tidak bisa mengubah, data tidak ada')
 break
 else if jenis is "potion" then
 if length of item_shop_list > 0 then
 ubah_potion(item_shop_list, item_shop_data)
 else
 output('Tidak bisa mengubah, data tidak ada')
 break
 else
 output("Ups, Pilih antara monster atau potion!")
else if pilihan is "hapus" or pilihan is "4" then
 // Menu Hapus
 while True do
 jenis <- input("Mau hapus apa? (monster/potion) : ")

```

```

 if jenis == "monster" then
 if length of monster_shop_data > 0 then
 monster_shop_data <-
hapus_monster(monster_shop_data, monster_data)
 else
 output('Tidak bisa menghapus, data tidak ada')
 break
 else if jenis == "potion" then
 if length of item_shop_list > 0 then
 item_shop_data <- hapus_potion(item_shop_list,
item_shop_data)
 else
 output('Tidak bisa menghapus, data tidak ada')
 break
 else
 output("Ups, Pilih antara monster atau potion!")
 else if pilihan is "keluar" or pilihan is "5" then
 keluar(username)
 else
 output('Input anda salah! Ulangi.')
 memilih(username, item_shop_data, monster_shop_data,
monster_data, potion_data)

```

```

procedure lihat_monster(monster_shop_data: list of dictionary,
monster_data: list of dictionary) -> None
 list_of_len <- [count_char_max(monster_shop_data, 'monster_id',
'ID'),
 count_char_max(monster_data, 'type', 'Name/Type'),
 count_char_max(monster_data, 'atk_power', 'ATK
Power'),
 count_char_max(monster_data, 'def_power', 'DEF
Power'),
 count_char_max(monster_data, 'hp', 'HP'),
 count_char_max(monster_shop_data, 'stock', 'Stok'),
 count_char_max(monster_shop_data, 'price',
'Harga')]

 output(f"{'ID':<{list_of_len[0]}} |
{'Name/Type':<{list_of_len[1]}} | {'ATK Power':<{list_of_len[2]}} |
{'DEF Power':<{list_of_len[3]}} | {'HP':<{list_of_len[4]}} |
{'Stok':<{list_of_len[5]}} | {'Harga':<{list_of_len[6]}}")
 output("-"*60)
 for data in monster_shop_data :
 for subdata in monster_data:
 if subdata['id'] == data['monster_id']:
 output(f"{subdata['id']:<{list_of_len[0]}} |
{subdata['type']:<{list_of_len[1]}} |
{subdata['atk_power']:<{list_of_len[2]}} |
{subdata['def_power']:<{list_of_len[3]}} |
{subdata['hp']:<{list_of_len[4]}} | {data['stock']:<{list_of_len[5]}}
| {data['price']}"")

```

```

procedure lihat_potion(item_shop_list: list of dictionary) -> None
 list_of_len <- [count_char_max(item_shop_list, 'id', 'ID'),
 count_char_max(item_shop_list, 'type',
 'Name/Type'),
 count_char_max(item_shop_list, 'stock', 'Stok'),
 count_char_max(item_shop_list, 'price', 'Harga')]

 output(f"{'ID':<{list_of_len[0]}} |
{'Name/Type':<{list_of_len[1]}} | {'Stok':<{list_of_len[2]}} |
{'Harga':<{list_of_len[3]}}}")
 output("-"*40)
 for data in item_shop_list :
 output(f"{'data['id']':<{list_of_len[0]}} |
{'data['type']':<{list_of_len[1]}} | {'data['stock']':<{list_of_len[2]}} |
{'data['price']':<{list_of_len[3]}}}")

function tambah_monster(monster_not_in_shop_list: list of dictionary,
monster_shop_data: list of dictionary, monster_data: list of
dictionary) -> None
 list_of_len <- [count_char_max(monster_shop_data, 'monster_id',
 'ID'),
 count_char_max(monster_data, 'type', 'Name/Type'),
 count_char_max(monster_data, 'atk_power', 'ATK
Power'),
 count_char_max(monster_data, 'def_power', 'DEF
Power'),
 count_char_max(monster_data, 'hp', 'HP')]

 output("Memuat data Monster yang belum ada di shop")
 output(f"{'ID':<{list_of_len[0]}} |
{'Name/Type':<{list_of_len[1]}} | {'ATK Power':<{list_of_len[2]}} |
{'DEF Power':<{list_of_len[3]}} | {'HP':<{list_of_len[4]}}}")
 output("-"*50)
 for data in monster_not_in_shop_list :
 output(f"{'data['id']':<{list_of_len[0]}} |
{'data['type']':<{list_of_len[1]}} |
{'data['atk_power']':<{list_of_len[2]}} |
{'data['def_power']':<{list_of_len[3]}} |
{'data['hp']':<{list_of_len[4]}}}")

 ids_list <- []
 for data in monster_not_in_shop_list:
 ids_list.append(data['id'])

 monster_id <- get_numeric_input("Masukkan id monster yang ingin
ditambah ke dalam shop : ")

 if monster_id not in ids_list:
 output('Tidak tersedia id tersebut, pilih id yang lain.')
 delay()
 -> tambah_monster(monster_not_in_shop_list, monster_shop_data,
monster_data)

```



```

 stok_baru <- get_numeric_input("Masukkan stok baru monster : ")
 harga_baru <- get_numeric_input("Masukkan harga baru monster : ")

 monster_shop_data.append({'monster_id': monster_id, 'stock':
stok_baru, 'price': harga_baru})
 sort_data(monster_shop_data, 'monster_id')

 for data in monster_data:
 if data['id'] == monster_id:
 nama <- data['type']

 output(f"Proses menambahkan {nama} ke dalam shop telah berhasil!")

```

```

function tambah_potion(item_not_in_shop_list: list of dictionary,
item_shop_list: list of dictionary, item_shop_data, potion_data: list
of dictionary) -> None
 list_of_len <- [count_char_max(item_shop_list, 'id', 'ID'),
count_char_max(item_shop_list, 'type', 'Name/Type')]
 ids_list <- []
 output("Memuat data Potion yang belum ada di shop")
 output(f"{'ID':<{list_of_len[0]}} |
{'Name/Type':<{list_of_len[1]}}")
 for data in item_not_in_shop_list :
 for subdata in potion_data:
 if subdata['potion_name'] == data:
 output(f"{'subdata['id']':<{list_of_len[0]}} |
{'subdata['potion_name']':<{list_of_len[1]}}")
 ids_list.append(subdata['id'])

 Id <- get_numeric_input("Masukkan id potion yang ingin ditambahkan
ke dalam shop : ")

 if Id not in ids_list:
 output('Tidak tersedia id tersebut, silahkan pilih id yang
lain.')
 delay()
 -> tambah_potion(item_not_in_shop_list, item_shop_list,
item_shop_data, potion_data)

 stok_baru <- get_numeric_input("Masukkan stok baru potion : ")
 harga_baru <- get_numeric_input("Masukkan harga baru potion : ")

 for data in potion_data:
 if data['id'] == Id:
 nama_item <- data['potion_name']

 item_shop_data.append({'type': nama_item, 'stock': stok_baru,
'price': harga_baru})
 output(f"Proses menambahkan {nama_item} ke dalam shop telah
berhasil!")

```

```

function ubah_monster(monster_shop_data: list of dictionary,
monster_data: list of dictionary) -> None
 output("Memuat keadaan Monster saat ini...")

 lihat_monster(monster_shop_data, monster_data)

 ids_list <- []
 for data in monster_shop_data:
 ids_list.append(data['monster_id'])

 Id <- get_numeric_input("Masukkan id monster yang ingin diubah :
")

 if Id not in ids_list:
 output('Tidak tersedia id tersebut, pilih id yang lain')
 delay()
 -> ubah_monster(monster_shop_data, monster_data)

 stok_baru <- get_numeric_input_boleh_kosong("Masukkan stok baru
monster : ", allow_empty=True)
 harga_baru <- get_numeric_input_boleh_kosong("Masukkan harga baru
monster : ", allow_empty=True)

 if stok_baru != "" and harga_baru != "" :
 for subdata in monster_shop_data:
 if subdata['monster_id'] == Id:
 subdata['stock'] <- str(stok_baru)
 subdata['price'] <- str(harga_baru)

 for data in monster_data:
 if data['id'] == Id:
 nama <- data['type']

 output("Pembaharuan data berhasil dilakukan pada monster ",
nama, " dengan stok baru berjumlah ", stok_baru, " dan harga baru ",
harga_baru)

 elif harga_baru != "" :
 for subdata in monster_shop_data:
 if subdata['monster_id'] == Id:
 subdata['price'] <- str(harga_baru)

 for data in monster_data:
 if data['id'] == Id:
 nama <- data['type']

 output("Pembaharuan data berhasil dilakukan pada monster ",
nama, " dengan harga baru ", harga_baru)

 elif stok_baru != "":
 for subdata in monster_shop_data:
 if subdata['monster_id'] == Id:
 subdata['stock'] <- str(stok_baru)

```

```

 for data in monster_data:
 if data['id'] == Id:
 nama <- data['type']

 output("Pembaharuan data berhasil dilakukan pada monster ",
nama, " dengan stok baru ", stok_baru)
 else:
 output('Tidak ada perubahan')
 delay()

```

```

function ubah_potion(item_shop_list: list of dictionary,
item_shop_data: list of dictionary) -> None
 lihat_potion(item_shop_list)

 ids_list <- []
 for data in item_shop_list:
 ids_list.append(data['id'])

 Id <- get_numeric_input("Urutan ke berapa yang ingin diganti? : ")

 if Id not in ids_list:
 output('Tidak tersedia id tersebut, pilih id yang lain')
 delay()
 -> ubah_potion(item_shop_list, item_shop_data)

 stok_baru <- get_numeric_input_boleh_kosong("Masukkan stok baru
potion : ", allow_empty=True)
 harga_baru <- get_numeric_input_boleh_kosong("Masukkan harga baru
potion : ", allow_empty=True)

 if stok_baru != "" and harga_baru != "" :
 for data in item_shop_list:
 if data['id'] == Id:
 name_potion <- data['type']
 for subdata in item_shop_data:
 if name_potion == subdata['type']:
 nama_potion <- subdata['type']
 subdata['stock'] <- str(stok_baru)
 subdata['price'] <- str(harga_baru)

 output("Pembaharuan data berhasil dilakukan pada potion jenis
", nama_potion, " dengan stok baru berjumlah ", stok_baru, " dan harga
baru ", harga_baru)

 elif harga_baru != "" :
 for data in item_shop_list:
 if data['id'] == Id:
 name_potion <- data['type']
 for subdata in item_shop_data:
 if name_potion == subdata['type']:
 nama_potion <- subdata['type']

```

```

 subdata['price'] <- str(harga_baru)
 output("Pembaharuan data berhasil dilakukan pada potion jenis
", nama_potion, " dengan harga baru ", harga_baru)

 elif stok_baru != "" :
 for data in item_shop_list:
 if data['id'] == Id:
 name_potion <- data['type']
 for subdata in item_shop_data:
 if name_potion == subdata['type']:
 nama_potion <- subdata['type']
 subdata['stock'] <- str(stok_baru)

 output("Pembaharuan data berhasil dilakukan pada monster ",
nama_potion, " dengan stok baru ", stok_baru)

 else:
 output('Tidak ada perubahan')
 delay()

```

```

function hapus_monster(monster_shop_data: list of dictionary,
monster_data: list of dictionary) -> monster_shop_data: list of
dictionary
 lihat_monster(monster_shop_data, monster_data)

 ids_list <- []
 for data in monster_shop_data:
 ids_list.append(data['monster_id'])

 Id <- get_numeric_input("Pilih ID monster yang ingin dihapus : ")

 if Id not in ids_list:
 output('Tidak tersedia id tersebut, pilih id yang lain.')
 delay()
 -> hapus_monster(monster_shop_data, monster_data)

 for data in monster_shop_data:
 for subdata in monster_data:
 if data['monster_id'] == Id and subdata['id'] == Id:
 type_monster <- subdata['type']
 id_monster <- subdata['id']

 yes_no <- input(f"Anda yakin ingin menghapus {type_monster} dari
shop? (y/n) : ").lower()
 if yes_no == "y" :
 monster_shop_data_update <- []
 for data in monster_shop_data:
 if data['monster_id'] != id_monster:
 monster_shop_data_update.append(data)

 monster_shop_data <- monster_shop_data_update
 output(f"Done, {type_monster} telah berhasil dihapus dari

```

```

shop.")
 delay()
 -> monster_shop_data

 elif yes_no == "n" :
 output(f"{type_monster} dibatalkan untuk dihapus dari shop.")
 delay()
 else:
 output('Input anda salah, ulangi!')
 delay()
 -> hapus_monster(monster_shop_data, monster_data)

```

```

function hapus_potion(item_shop_list: list of dictionary,
item_shop_data: list of dictionary) -> item_shop_data: list of
dictionary
 lihat_potion(item_shop_list)

 Id <- get_numeric_input("Pilih ID potion yang ingin dihapus : ")

 for data in item_shop_list:
 if data['id'] == str(Id):
 type_potion <- data['type']

 yes_no <- input(f"Anda yakin ingin menghapus {type_potion} dari
shop? (y/n) : ").lower()
 if yes_no == "y" :
 item_shop_data_update <- []
 for data in item_shop_data:
 if data['type'] != type_potion:
 item_shop_data_update.append(data)

 item_shop_data <- item_shop_data_update
 output(f"Done, {type_potion} telah berhasil dihapus dari
shop.")
 delay()
 -> item_shop_data

 elif yes_no == "n":
 output(f"{type_potion} dibatalkan untuk dihapus dari shop.")
 delay()
 else:
 output('Input anda salah, ulangi!')
 delay()
 -> hapus_potion(item_shop_list, item_shop_data)

```

```

procedure keluar(username: string)
 output "Sampai jumpa lagi, " + username + "..."
 delay()
 ->

```

```
function count_char_max(data_list: list of dictionary, kolom: string,
header: string) -> integer
 char_max <- length(header)
 for i <- 0 to length(data_list) - 1 do
 if length(data_list[i][kolom]) > char_max then
 char_max <- length(data_list[i][kolom])
 -> char_max
```

```
procedure sort_data(data: list of dictionary, sortby: string)
 n <- length(data)
 for i <- 0 to n - 1 do
 for j <- 0 to n - i - 2 do
 if int(data[j][sortby]) > int(data[j+1][sortby]) then
 swap data[j] with data[j+1]
```

```
procedure delay()
 sleep for 1 second
 if operating system is Windows then
 clear screen
 else
 clear screen
```

```
function get_numeric_input(prompt: string) -> string
 while True do
 output prompt
 read input from user
 try
 value <- parse input as integer
 if value >= 0 then
 -> string(value)
 else
 output "Input harus berupa bilangan positif! Ulangi!"
 except ValueError:
 output "Input harus berupa angka. Silakan coba lagi."
```

```
function get_numeric_input_boleh_kosong(prompt: string, allow_empty:
boolean) -> string
 while True do
 output prompt
 read input from user
 if allow_empty and input is empty then
 -> input
 try
 value <- parse input as integer
 if value > 0 then
 -> string(value)
 else
 output "Input harus berupa bilangan positif! Ulangi!"
 except ValueError:
```

```
output "Input harus berupa angka. Silakan coba lagi."
```

## 12. F-13 Monster Management

```
function delay() -> None:
 time.sleep(0.5)
 if os.name == 'nt' then
 os.system('cls')
 else
 os.system('clear')

function tampilan_awal(monster_list: list of dictionaries) -> None:
 output "Selamat Datang Para Agen"
 output "Di sini adalah tempat database para monster."
 output "-----"
 -> pilihan_monster_management(monster_list)

function pilihan_monster_management(monster_list: list of
dictionaries) -> None:
 output ". . ."
 output "[1] Tampilkan semua monster yang ada"
 output "[2] Tambahkan monster baru"
 output "[3] Untuk kembali ke menu admin"
 pil <- input ("Anda ingin pilih aksi mana (1/2/3)? : ")
 if pil == '1' then
 -> lihat_monster(monster_list)
 -> pilihan_monster_management(monster_list)
 elif pil == '2' then
 -> tambah_monster_baru(monster_list)
 elif pil == '3' then
 output "Keluar dari Monster Management"
 output ". . ."
 -> None
 else
 output "Input anda salah, ulangi!"
 delay()
 -> pilihan_monster_management(monster_list)

function cek_kesamaan_nama(monster_list: list of dictionaries, nama:
string) -> boolean:
 for each data in monster_list do
 if data['type'] == nama then
 -> True
 -> False

function tambah_monster_baru(monster_list: list of dictionaries) ->
None:
 output "... Proses pembuatan monster baru dimulai ..."
```

```

nama <- input("Nama/Type Monster Baru : ")
while cek_kesamaan_nama(monster_list, nama) do
 output "Ups.. Nama Monster sudah terdaftar..."
 output "Silakan masukkan nama lain!"
 nama <- input("Nama/Type Monster Baru : ")
valid_int <- set of "0123456789"
atk <- input("ATK Power Monster Baru : ")
while not all(char in valid_int for char in atk) do
 output "ATK Power harus dalam angka yang bernilai positif..."
 output "Silakan coba lagi!"
 atk <- input("ATK Power Monster Baru : ")
Atk <- convert atk to integer
while True do
 output "DEF Power Monster Baru (0-50) : "
 try
 defense <- convert input to integer
 if 0 <= defense <= 50 then
 break
 else
 output "Def Power Monster harus bernilai 0-50..."
 output "Silakan coba lagi!"
 except ValueError do
 output "Input harus berupa bilangan bulat!"
while True do
 output "Nilai HP Monster Baru: "
 try
 hp <- convert input to integer
 if hp >= 0 then
 break
 else
 output "HP Monster harus bernilai bilangan positif"
 output "Silakan coba lagi!"
 except ValueError do
 output "Input harus berupa bilangan bulat!"
monster_id <- length of monster_list + 1
monster_baru <- {'id': convert monster_id to string, 'type': nama,
'atk_power': convert Atk to string, 'def_power': convert defense to
string, 'hp': convert hp to string}
pilihan <- input("Ingin menambahkan Monster baru ke database?
(Y/N) : ")
if pilihan.lower() = "y" then
 output "Monster baru berhasil ditambahkan ke database!"
 append monster_baru to monster_list
 -> pilihan_monster_management(monster_list)
elif pilihan.lower() = "n" then
 output "Monster baru gagal ditambahkan ke database!"
 -> pilihan_monster_management(monster_list)
else
 output 'Perintah anda salah, pilih (Y atau N)'
 -> tambah_monster_ke_database(monster_list, monster_baru)

function count_char_max(data_list: list of dictionaries, kolom:
string, header: string) -> integer:

```



```

char_max <- length of header
for each i in range(length of data_list) do
 if length of data_list[i][kolom] > char_max then
 char_max <- length of data_list[i][kolom]
-> char_max

function lihat_monster(monster_data: list of dictionaries) -> None:
 list_of_len <- [count_char_max(monster_data, 'id', 'ID'),
count_char_max(monster_data, 'type', 'Name/Type'),
count_char_max(monster_data, 'atk_power', 'ATK Power'),
count_char_max(monster_data, 'def_power', 'DEF Power'),
count_char_max(monster_data, 'hp', 'HP')]
 output "Memuat data Monster yang belum ada di shop"
 output "ID | Name/Type | ATK Power | DEF Power | HP"
 output "-----"
 for each data in monster_data do
 output data['id'], ' | ', data['type'], ' | ',
data['atk_power'], ' | ', data['def_power'], ' | ', data['hp']

```

### 13. F-14 Load

```

procedure load()
{ Inisialisasi Argument Parser }
parser <- argparse.ArgumentParser(description="Buka folder.")
{ Menambahkan argumen posisional }
parser.add_argument('folder', nargs='?', default=None,
help='mengakses folder csv ')
{ Penguraian Argumen }
args <- parser.parse_args()

{ Pengecekan Argumen }
if args.folder is None then
 output ("Tidak ada nama folder yang diberikan!\nUsage : python
main.py <nama_folder>")
 terminate program

{ Memanggil fungsi untuk memeriksa keberadaan folder }
if not os.path.exists(args.folder) then
 output ("Folder \"" + args.folder + "\" tidak ditemukan!")
 terminate program

```

### 14. F-15 Save

```

procedure save(user: list of dictionary, item_inventories: list of
dictionary, item_shop: list of dictionary, monster: list of
dictionary, monster_shop: list of dictionary, monster_inventory: list
of dictionary)

```

```
{ Memeriksa apakah folder yang dimaksud sudah ada dan akan membuat folder baru jika belum serta menyimpan file csv ke folder tersebut }
```

```
parent_folder <- "data"
folder <- get_valid_input("Masukkan nama folder: ")
folder <- "data/" + folder
output ("Saving...")

if not os.path.exists(parent_folder) then
 output("Membuat folder data")
 os.makedirs(parent_folder)

if not os.path.exists(folder) then
 output("Membuat folder " + folder + ".")
 os.makedirs(folder)

data_save(folder, "user", user)
data_save(folder, "item_inventory", item_inventories)
data_save(folder, "item_shop", item_shop)
data_save(folder, "monster", monster)
data_save(folder, "monster_shop", monster_shop)
data_save(folder, "monster_inventory", monster_inventory)

output("Berhasil menyimpan data di folder " + folder + "!")
```

```
procedure data_save(path: string, file_name: string, data)
{ Menyimpan data dari bentuk list of dictionaries menjadi csv di suatu folder }
```

```
file_path <- path + "/" + file_name + ".csv"

if file_name = "item_inventory" then
 call sort_data(data, "user_id")
else if file_name = "monster" then
 call sort_data(data, "id")
else if file_name = "monster_inventory" then
 call sort_data(data, "user_id")
else if file_name = "monster_shop" then
 call sort_data(data, "monster_id")
else if file_name = "user" then
 call sort_data(data, "id")

data <- csv.join_array(data)

with open(file_path, 'w') as csvfile do
 csvfile.write(data)
```

```
procedure sort_data(data: list of dictionary, sortby: string)
{ Mengurutkan data sesuai urutannya }
```

```

n <- length(data)
for i from 0 to n - 1 do
 for j from 0 to n - i - 1 do
 if int(data[j][sortby]) > int(data[j + 1][sortby]) then
 data[j + 1], data[j]

```

```

function is_valid_name(name: string) -> boolean
{ Memastikan bahwa input nama folder merupakan nama yang valid }
forbidden_characters <- ['\\', '/', ':', '*', '?', '"', '<', '>',
'|']
reserved_names <- [
 "CON", "PRN", "AUX", "NUL",
 "COM1", "COM2", "COM3", "COM4", "COM5", "COM6", "COM7",
"COM8", "COM9",
 "LPT1", "LPT2", "LPT3", "LPT4", "LPT5", "LPT6", "LPT7",
"LPT8", "LPT9"
]
if name in reserved_names then
 -> False
for each char in forbidden_characters do
 if char in name then
 -> False
-> True

```

```

procedure get_valid_input(prompt: string) -> string
{ Meminta user untuk memasukkan nama folder, jika salah maka akan
meminta user untuk mengulangi }
while True do
 user_input <- input(prompt)
 if is_valid_name(user_input) then
 -> user_input
 else
 output("Input invalid, gunakan nama lain!")

```

## 15. F-16 Exit

```

procedure game_exit(username: string, user: list of dictionary,
item_inventories: list of dictionary, item_shop: list of dictionary,
monster: list of dictionary, monster_shop: list of dictionary,
monster_inventory: list of dictionary)
{ Fungsi untuk keluar dari game }

exit_input <- input("Apakah anda mau melakukan penyimpanan file
yang sudah diubah? (y/n): ").lower()

```

```

 if exit_input != 'y' and exit_input != 'n' then
 output("Input anda salah, ulangi")
 -> game_exit(username, user, item_inventories, item_shop,
monster, monster_shop, monster_inventory)
 else if exit_input = 'y' then
 save(user, item_inventories, item_shop, monster, monster_shop,
monster_inventory)
 output("Selamat tinggal agent ", username, "!")
 wait for 3 seconds
 if os.name = 'nt' then
 os.system('cls')
 else
 os.system('clear')
 terminate program
 else if exit_input = 'n' then
 output("Selamat tinggal agent ", username, "!")
 wait for 3 seconds
 if os.name = 'nt' then
 os.system('cls')
 else
 os.system('clear')
 terminate program

```

## TANGKAPAN GAMBAR UJI PROGRAM

### 1. F-01 Register



```
Command Prompt - python main.py data

OWCA

[Login] [Help] [Register] [Menu] [Logout]

[Exit]

Masukkan command (lowercase): register
Masukkan username: banana
Masukkan password: brototo
Silahkan pilih salah satu monster sebagai monster awalmu.
1. Pikachu
2. Bulbu
3. Zeze
4. Zuko
5. Chacha
6. Bimosaurus
7. Arceus
8. Squirex
9. Mewthree
10. Luigi
Monster pilihanmu:
```

Gambar 1.1 Input Username Valid



```
Command Prompt - python main.py data

Selamat Datang di program OWCA!

OWCA

[Login] [Help] [Register] [Menu] [Logout]

[Exit]

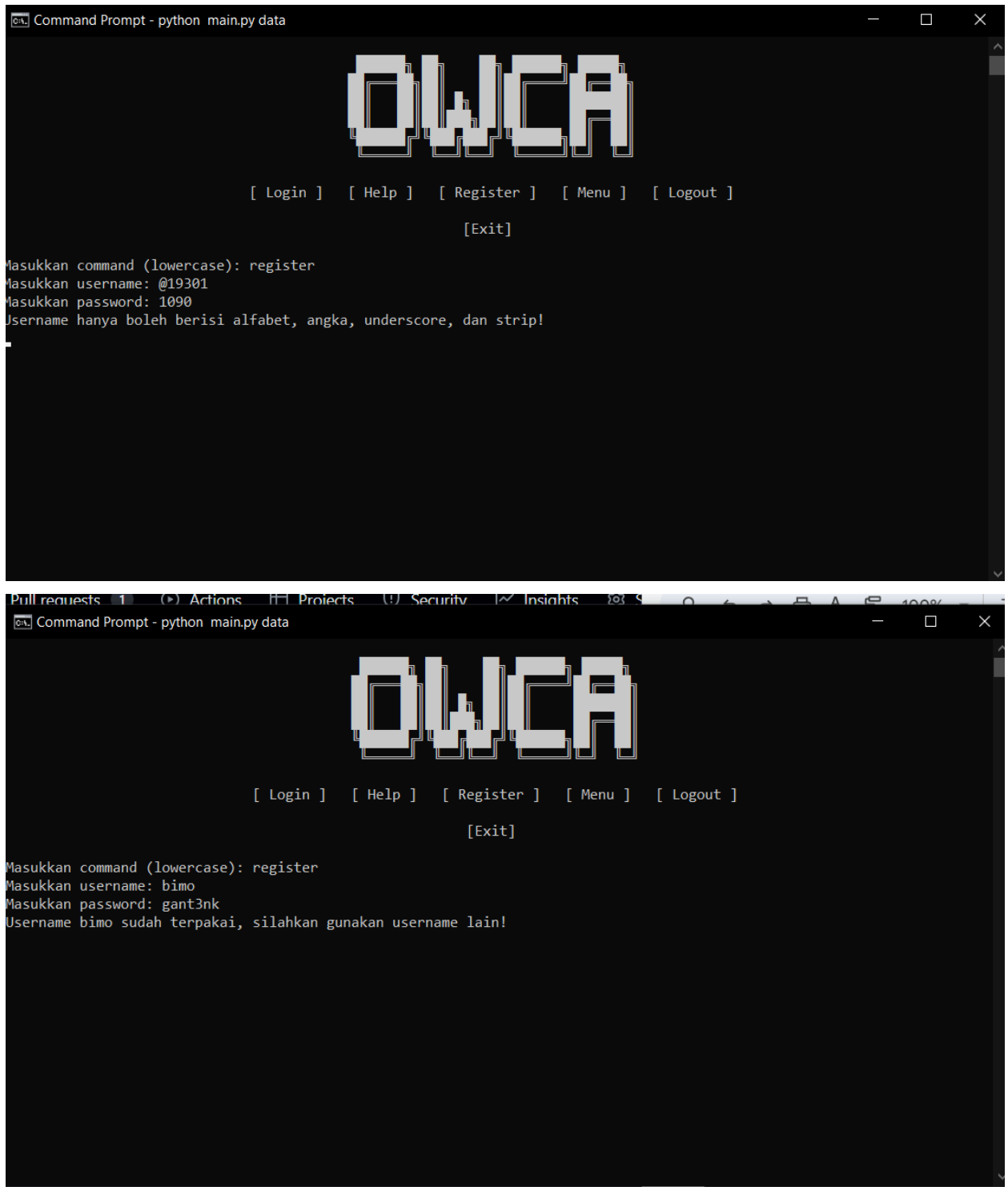
Masukkan command (lowercase): register
Masukkan username: banana
Masukkan password: brototo
Silahkan pilih salah satu monster sebagai monster awalmu.
1. Pikachu
2. Bulbu
3. Zeze
4. Zuko
5. Chacha
6. Bimosaurus
7. Arceus
8. Squirex
9. Mewthree
10. Luigi
Monster pilihanmu: 1901293
Monster pilihan anda tidak ada, silahkan pilih kembali.
Silahkan pilih salah satu monster sebagai monster awalmu.
```

```
Command Prompt - python main.py data
Masukkan password: brototo
Silahkan pilih salah satu monster sebagai monster awalmu.
1. Pikachu
2. Bulbu
3. Zeze
4. Zuko
5. Chacha
6. Bimosaurus
7. Arceus
8. Squirex
9. Mewthree
10. Luigi
Monster pilihanmu: 1901293
Monster pilihan anda tidak ada, silahkan pilih kembali.
Silahkan pilih salah satu monster sebagai monster awalmu.
1. Pikachu
2. Bulbu
3. Zeze
4. Zuko
5. Chacha
6. Bimosaurus
7. Arceus
8. Squirex
9. Mewthree
10. Luigi
Monster pilihanmu: abdul
Input anda salah! Ulangi!
Monster pilihanmu:
```

Gambar 1.2 Jika Input Choose Monster tidak Valid

```
Monster pilihan anda tidak ada, silahkan pilih kembali.
Silahkan pilih salah satu monster sebagai monster awalmu.
1. Pikachu
2. Bulbu
3. Zeze
4. Zuko
5. Chacha
6. Bimosaurus
7. Arceus
8. Squirex
9. Mewthree
10. Luigi
Monster pilihanmu: abdul
Input anda salah! Ulangi!
Monster pilihanmu: 2
Selamat datang agent banana. Mari kita mengalahkan Dr. Asep Spakbor dengan Bulbu!
```

Gambar 1.3 Jika Input Choose Monster Valid

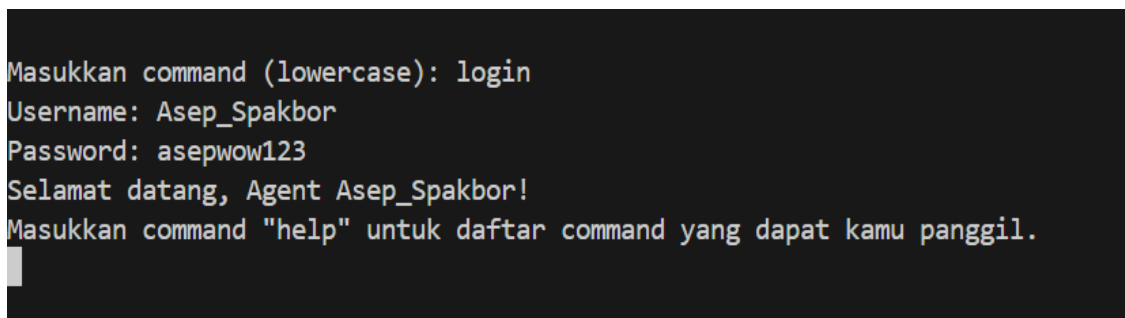


Gambar 1.4 Jika Input Username Tidak Valid



Gambar 1.5 Jika Register setelah Register

## 2. F-02 Login



Gambar 2.1: Pengujian saat login yang dilakukan berhasil





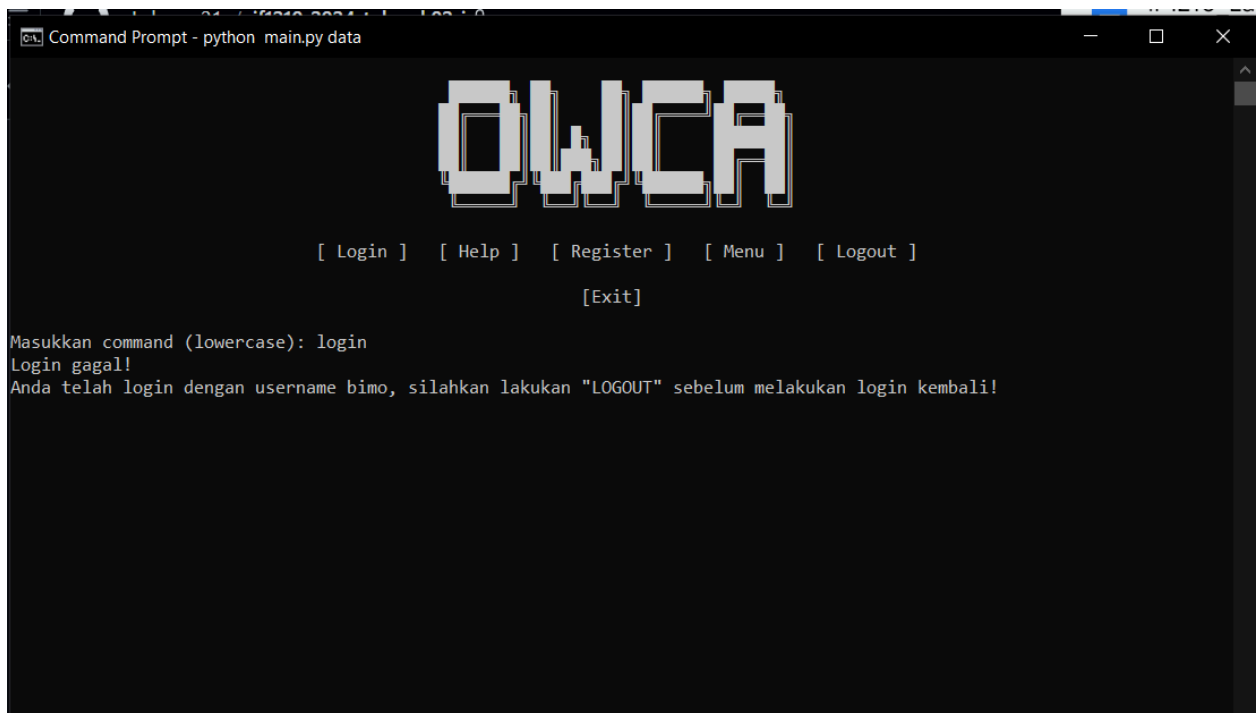
Gambar 2.2 Uji Login Username dan Password Tidak Valid



Gambar 2.3 Uji Password Salah



Gambar 2.2 Uji Username tidak Valid tapi Password Valid



Gambar 2.3 Uji Login Setelah Login

### 3. F-03 Logout



Gambar 3.1 Jika Logout Tidak Valid



Gambar 3.2 Logout Valid

#### 4. F-04 Menu & Help



Gambar 4.1 Help untuk User yang Belum Login

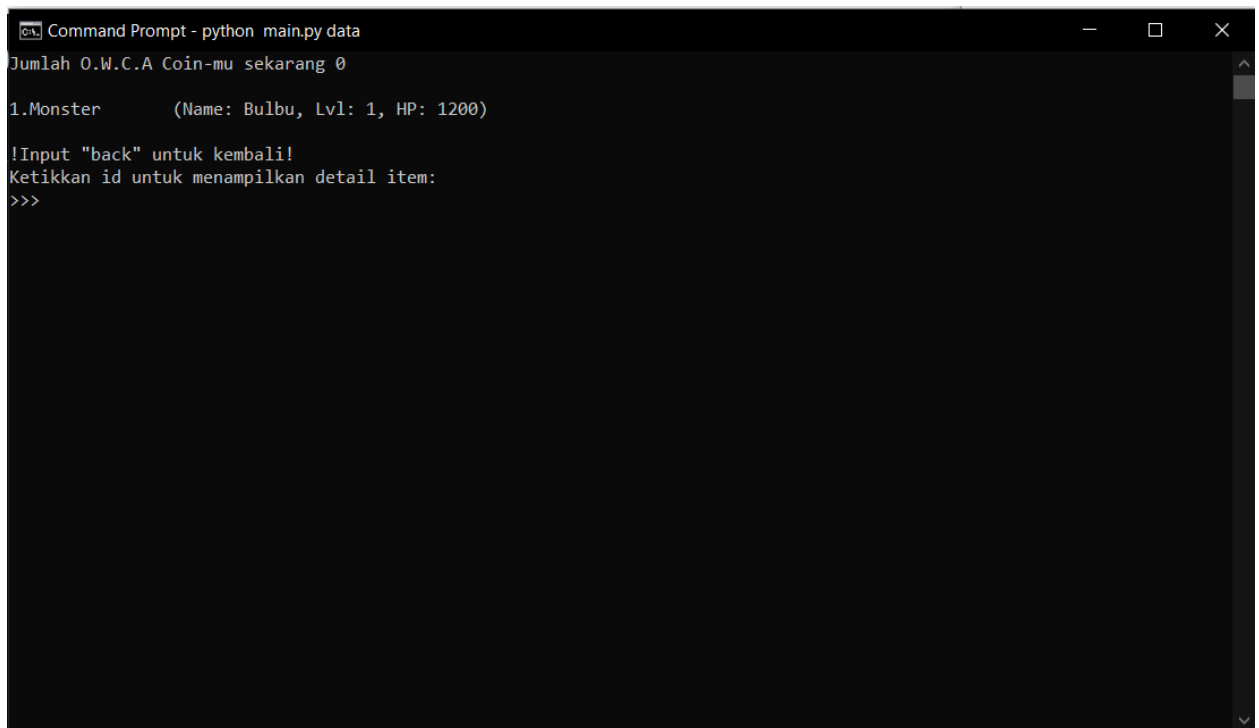


Gambar 4.2 Help untuk User yang Telah Login



Gambar 4.3 Help untuk Admin

## 5. F-07 Inventory



Gambar 5.1 Tampilan Awal Inventory

```
Command Prompt - python main.py data
Jumlah O.W.C.A Coin-mu sekarang 0

1.Monster (Name: Bulbu, Lvl: 1, HP: 1200)

!Input "back" untuk kembali!
Ketikkan id untuk menampilkan detail item:
>>> 2
Id item tidak ada di inventory, gunakan Id lain.
!Input "back" untuk kembali!
Ketikkan id untuk menampilkan detail item:
>>> ahdiajweod
Id item tidak ada di inventory, gunakan Id lain.
!Input "back" untuk kembali!
Ketikkan id untuk menampilkan detail item:
>>>
```

Gambar 5.2 Input Tidak Valid di Inventory

```
Command Prompt - python main.py data
Jumlah O.W.C.A Coin-mu sekarang 9999

1.Monster (Name: Pikachu, Lvl: 1, HP: 600)
2.Monster (Name: Squirex, Lvl: 3, HP: 650)
3.Potion (Type: strength, Qty: 5)
4.Potion (Type: resilience, Qty: 3)

!Input "back" untuk kembali!
Ketikkan id untuk menampilkan detail item:
>>> 1
Monster
Name : Pikachu
ATK Power : 125
DEF Power : 10
HP : 600
Level : 1
!Input "back" untuk kembali!
Ketikkan id untuk menampilkan detail item:
>>> 3
Potion
Type : strength
Quantity : 5
!Input "back" untuk kembali!
Ketikkan id untuk menampilkan detail item:
>>>
```

Gambar 5.3 Input Valid di Inventory

## 6. F-08 Battle

```
RAWRRR, Monster Mewthree telah muncul !!!
```

```
 /\-----/_
 / \
 | | o o | //|
 | | .www.| //|
 | |~~~~~|// /
 | |~~~~~|// /
 | |~~~~~|// /
 | |~~~~~|// /
 |_____|___|\
 |_____|___|\

Mewthree HP: [██] 260/260
Name : Mewthree
ATK Power : 130
DEF Power : 65
HP : 260
Level : 3

===== MONSTER LIST =====
1. Pikachow, Level = 1
2. Squirex, Level = 3
Pilih monster untuk bertarung: █
```

Gambar 6.1: Muncul monster musuh secara acak

```
RAWRRR, Monster Mewthree telah muncul !!!
```

```
 /\-----/_
 / \|
 | | o o ||
 | | .vvv..|| /
 // | | || /
 // | ~~~~~|//
 // | | || /
 |// | | ||
 |// |___| ||
 \ | | |
 \ | | |
 . _\._\
```

Mewthree HP : [ ] 260/260  
Name : Mewthree  
ATK Power : 130  
DEF Power : 65  
HP : 260  
Level : 3

===== MONSTER LIST =====  
1. Pikachu, Level = 1  
2. Squirex, Level = 3  
Pilih monster untuk bertarung: 3  
Pilihan monster tidak tersedia! Coba lagi!

Pilih monster untuk bertarung:

Gambar 6.2: Pengujian input pilihan monster (user) yang tidak tersedia di monster list





```
Pilih perintah: 1
```

```
===== MONSTER LIST =====
SKADIDOD00, Pikachow menyerang Mewthree !!!
```

```
Mewthree HP: [██] -----] 233/260
```

```
Name : Mewthree
```

ATK Power : 130

DEF Power : 65

```
Penjelasan: ATT: 41, Reduced by: 34, ATT Results: 27
```

```
===== TURN 1 (Mewthree) =====
```

===== ENEMY TURN =====

SKADLIDODOR, Mewthree menyerang Pikachow !!!

Pikachow HP: [████████████████████] ----] 552/600

```
Name : Pikachow
```

ATK Power : 125

```
DEF Power : 10
```

HP : 552

```
Level : 1
```

```
Penjelasan: ATT: 49, Reduced by: 1, ATT Results: 48
```

```
===== TURN 2 (Pikachow) =====
```

```
===== USER TURN =====
```

## 1. Attack

```
2. Use Potion
```

```
3. Quit
```

Pilih perintah:

Gambar 6.5: Pengujian saat user memilih perintah 1 (*attack*)

```
Pilih perintah: 2

===== POTION LIST =====
1. Strength Potion (Qty: 5) - Increases ATK Power
2. Resilience Potion (Qty: 3) - Increases DEF Power
3. Healing Potion (Qty: 0) - Restores Health
4. Cancel
Pilih perintah:
```

Gambar 6.6: Pengujian saat user memilih perintah 2 (*use potion*)

```
===== POTION LIST =====
1. Strength Potion (Qty: 5) - Increases ATK Power
2. Resilience Potion (Qty: 3) - Increases DEF Power
3. Healing Potion (Qty: 0) - Restores Health
4. Cancel
Pilih perintah: 1
Name : Pikachu
ATK Power : 131
DEF Power : 10
HP : 552
Level : 1

Penjelasan: Attack power monster bertambah sebesar 5% menjadi: 131
===== TURN 2 (Pikachu) =====
===== USER TURN =====
1. Attack
2. Use Potion
3. Quit
Pilih perintah:
```

Gambar 6.7: Pengujian saat user memilih potion 1 (*strength potion*)

```
===== POTION LIST =====
1. Strength Potion (Qty: 4) - Increases ATK Power
2. Resilience Potion (Qty: 3) - Increases DEF Power
3. Healing Potion (Qty: 0) - Restores Health
4. Cancel
Pilih perintah: 2
Name : Pikachu
ATK Power : 131
DEF Power : 10
HP : 552
Level : 1

Penjelasan: Defense power monster bertambah sebesar 5% menjadi: 10
===== TURN 2 (Pikachu) =====
===== USER TURN =====
1. Attack
2. Use Potion
3. Quit
Pilih perintah:
```

Gambar 6.8: Pengujian saat user memilih potion 2 (*resilience potion*)

```
===== POTION LIST =====
1. Strength Potion (Qty: 4) - Increases ATK Power
2. Resilience Potion (Qty: 2) - Increases DEF Power
3. Healing Potion (Qty: 0) - Restores Health
4. Cancel
Pilih perintah: 3
Wah, kamu sedang tidak memiliki ramuan ini, silahkan pilih ramuan lain!
===== TURN 2 (Pikachow) =====
===== USER TURN =====
1. Attack
2. Use Potion
3. Quit
Pilih perintah: █
```

Gambar 6.9: Pengujian saat user memilih potion dengan stok yang kosong

```
===== POTION LIST =====
1. Strength Potion (Qty: 4) - Increases ATK Power
2. Resilience Potion (Qty: 2) - Increases DEF Power
3. Healing Potion (Qty: 0) - Restores Health
4. Cancel
Pilih perintah: 4
===== TURN 2 (Pikachow) =====
===== USER TURN =====
1. Attack
2. Use Potion
3. Quit
Pilih perintah: █
```

Gambar 6.10: Pengujian saat user memilih perintah *cancel*/keluar dari pemilihan potion

```
===== POTION LIST =====
1. Strength Potion (Qty: 4) - Increases ATK Power
2. Resilience Potion (Qty: 2) - Increases DEF Power
3. Healing Potion (Qty: 0) - Restores Health
4. Cancel
Pilih perintah: 5
Perintah yang diberikan tidak tersedia, coba ulangi!
===== TURN 2 (Pikachow) =====
===== USER TURN =====
1. Attack
2. Use Potion
3. Quit
Pilih perintah: █
```

Gambar 6.11: Pengujian saat user memilih perintah yang tidak tersedia saat pemilihan potion

```
SKADIDOD00, Pikachu menyerang Mewthree !!!

 /\----/_
 / \ /|
 | | 0 0 | / |
 | | .vvvv. | / /
 / | | | | /
 / | | ^^^^^ | /
| /| | | | /
 / | | | | /
 | | | |
 \ | | |
 _ \ | |
 _ \ | |

Mewthree HP: [-----] 0/260
Name : Mewthree
ATK Power : 130
DEF Power : 65
HP : 0
Level : 3

Penjelasan: ATT: 90, Reduced by: 36, ATT Results: 57

Selamat anda mengalahkan monster Mewthree !!!
Total OC yang diperoleh: 9
Tekan apapun untuk keluar: █
```

Gambar 6.12: Pengujian ketika user berhasil mengalahkan musuh



```

 /\----/_
 / \ /|
 | | 0 0 | /|
 | | .www. | /
 / | | | /
 / | | | /
 | /| | /
 / | | | /
 | | | |
 \ | | |
 _ \ | _ \

```

```
===== TURN 1 (Pikachow) =====
===== USER TURN =====
1. Attack
2. Use Potion
3. Quit
```

100

```

 /\-----/_
 / \ /|
 | | 0 0 | / |
 | | .vvvv. | / /
 / | | | | | /
 / | | ~~~~~ /
| / | | | | /
| / | | | | /
| | | | | |
 \ | | | | |
 \ | | | | |
 _| _| _|

```

SKADIDODOO, Pikachu menyerang Mewthree !!!

Mewthree HP: [-----] 0/200  
 Name : Mewthree  
 ATK Power : 100  
 DEF Power : 50  
 HP : 0  
 Level : 1

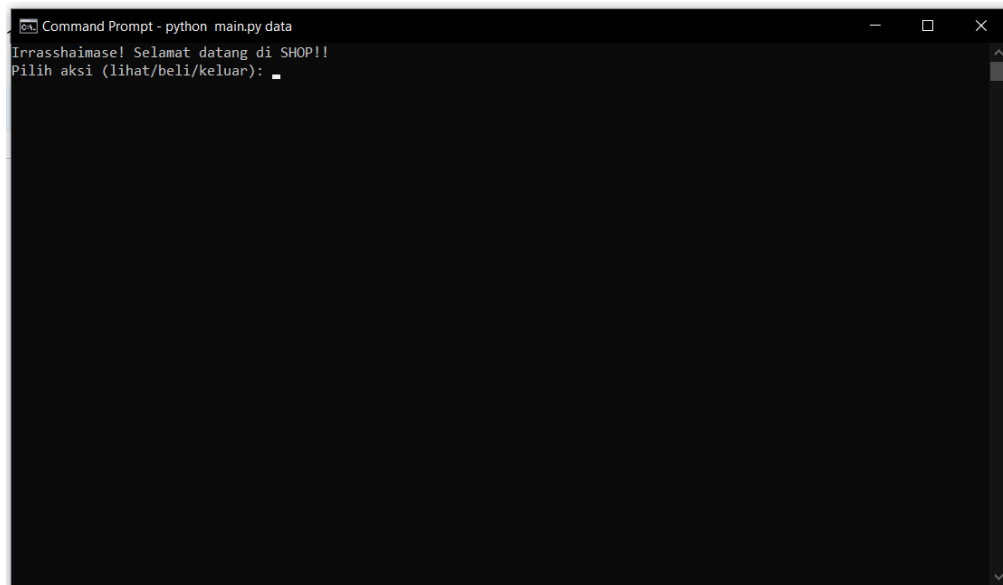
# Penjelasan: ATT: 62, Reduced by: 34, ATT Results: 40  
 Selamat anda mengalahkan monster Mewthree !!!  
 Menuju ke stage berikutnya....

Gambar 7.3: Pengujian ketika user berhasil mengalahkan musuh dan lanjut ke *stage* berikutnya

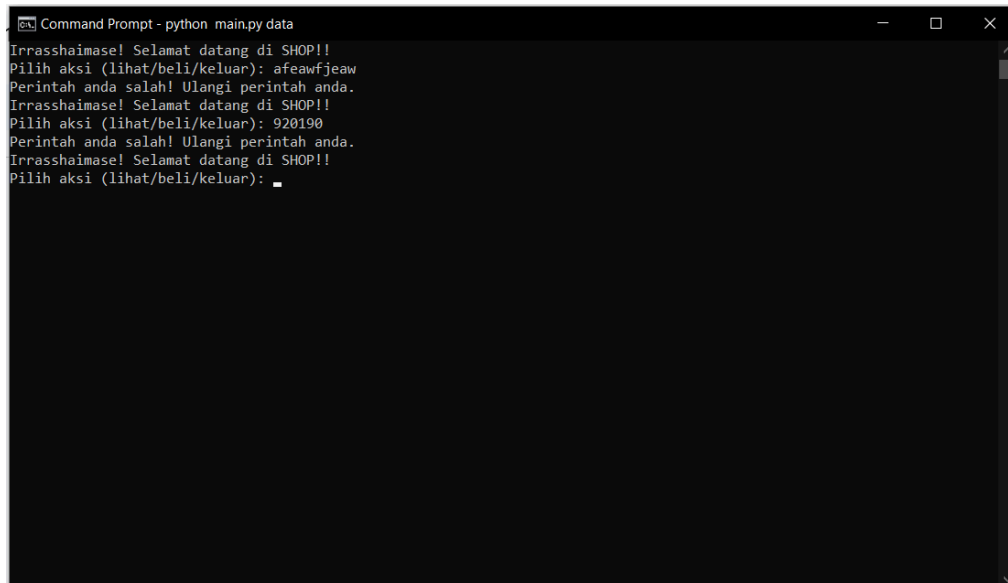
[illegible]

Gambar 7.4: Pengujian ketika user telah dikalahkan oleh musuh dan sesi arena telah selesai

## 8. F-10 Shop & Currency



Gambar 8.1 Tampilan Awal Shop



Gambar 8.2 Input Tidak Valid di Shop



```

Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): afeawfjeaw
Perintah anda salah! Ulangi perintah anda.
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): 920190
Perintah anda salah! Ulangi perintah anda.
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion)?: monster
ID | Name/Type | ATK Power | DEF Power | HP | Stok | Harga

1 | Pikachu | 125 | 10 | 600 | 10 | 500
2 | Bulbu | 50 | 50 | 1200 | 4 | 700
3 | Zeze | 300 | 10 | 100 | 3 | 1000
4 | Zuko | 100 | 25 | 800 | 8 | 550
5 | Chacha | 80 | 30 | 700 | 7 | 600
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion)?: potion
ID | Name/Type | Stok | Harga

1 | strength | 10 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar):

```

Gambar 8.3 Lihat Monster dan Potion di Shop

```

Command Prompt - python main.py data
Mau lihat apa? (monster/potion)?: monster
ID | Name/Type | ATK Power | DEF Power | HP | Stok | Harga

1 | Pikachu | 125 | 10 | 600 | 10 | 500
2 | Bulbu | 50 | 50 | 1200 | 4 | 700
3 | Zeze | 300 | 10 | 100 | 3 | 1000
4 | Zuko | 100 | 25 | 800 | 8 | 550
5 | Chacha | 80 | 30 | 700 | 7 | 600
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion)?: potion
ID | Name/Type | Stok | Harga

1 | strength | 10 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): aefjawppjfea
Perintah anda salah! Ulangi perintah anda.
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): ihat
Perintah anda salah! Ulangi perintah anda.
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion)?: 131031afeieajf
Perintah anda salah! Ulangi perintah anda.

```

Gambar 8.4 Input tidak valid ketika ingin lihat

```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 9999

Mau beli apa? (monster/potion): 3131jiwawjd
Perintah anda salah! Ulangi perintah anda!
Jumlah O.W.C.A coin mu sekarang 9999

Mau beli apa? (monster/potion):
```

Gambar 8.5 Input Tidak Valid ketika Beli

```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion?): monster
ID | Name/Type | ATK Power | DEF Power | HP | Stok | Harga

1 | Pikachu | 125 | 10 | 600 | 10 | 500
2 | Bulbu | 50 | 50 | 1200 | 4 | 700
3 | Zeze | 300 | 10 | 100 | 3 | 1000
4 | Zuko | 100 | 25 | 800 | 8 | 550
5 | Chacha | 80 | 30 | 700 | 7 | 600
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 9999

Mau beli apa? (monster/potion): monster
Masukkan monster id: 9319203
Id tidak tersedia, silahkan coba id yang lain.
```

```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion)?: monster
ID | Name/Type | ATK Power | DEF Power | HP | Stok | Harga

1 | Pikachu | 125 | 10 | 600 | 10 | 500
2 | Bulbu | 50 | 50 | 1200 | 4 | 700
3 | Zeze | 300 | 10 | 100 | 3 | 1000
4 | Zuko | 100 | 25 | 800 | 8 | 550
5 | Chacha | 80 | 30 | 700 | 7 | 600
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 9999
Mau beli apa? (monster/potion): monster
Masukkan monster id: naeidjdaoejf
Id tidak tersedia, silahkan coba id yang lain.
```

Gambar 8.6 Input Monster id Tidak Valid di Beli Monster

```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion)?: monster
ID | Name/Type | ATK Power | DEF Power | HP | Stok | Harga

1 | Pikachu | 125 | 10 | 600 | 10 | 500
2 | Bulbu | 50 | 50 | 1200 | 4 | 700
3 | Zeze | 300 | 10 | 100 | 3 | 1000
4 | Zuko | 100 | 25 | 800 | 8 | 550
5 | Chacha | 80 | 30 | 700 | 7 | 600
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 9999
Mau beli apa? (monster/potion): monster
Masukkan monster id: 4
Berhasil membeli item Zuko. Item sudah masuk ke inventory-mu!
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar):
```

Gambar 8.7 Input Valid di Beli Monster

```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion?): potion
ID | Name/Type | Stok | Harga

1 | strength | 10 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): 1290
Perintah anda salah! Ulangi perintah anda.
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 9399

Mau beli apa? (monster/potion): potion
Masukkan id potion: -13-0
Id tidak tersedia, silahkan coba id yang lain.
```

```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion?): potion
ID | Name/Type | Stok | Harga

1 | strength | 10 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 9399

Mau beli apa? (monster/potion): 2
Perintah anda salah! Ulangi perintah anda!
Jumlah O.W.C.A coin mu sekarang 9399

Mau beli apa? (monster/potion): potion
Masukkan id potion: 2
Masukkan jumlah: 100
Kuantitas barang tidak mencukupi, silahkan ulangi
```

```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion?): potion
ID | Name/Type | Stok | Harga

1 | strength | 10 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 9399

Mau beli apa? (monster/potion): potion
Masukkan id potion: 1
Masukkan jumlah: 0
Input harus bilangan positif dan lebih dari 0!
Masukkan jumlah: -1
Input harus bilangan positif dan lebih dari 0!
Masukkan jumlah:
```

Gambar 8.8 Input Tidak Valid di Beli Potion

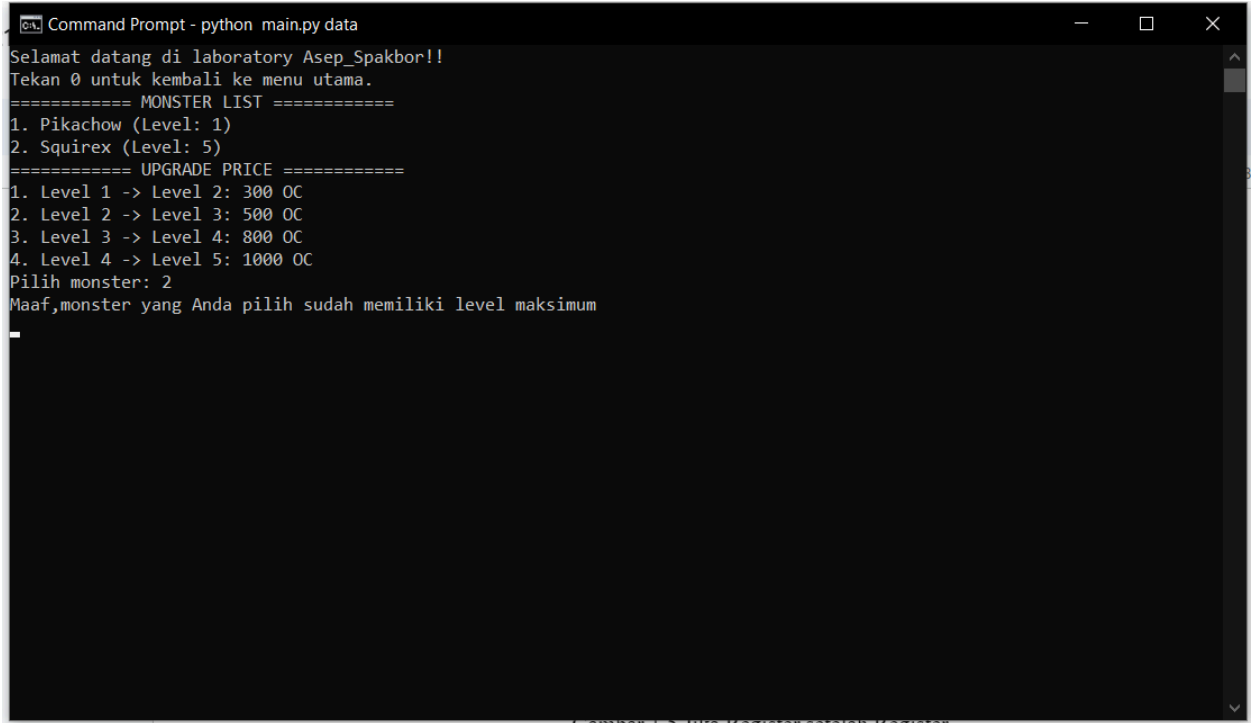
```
Command Prompt - python main.py data
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): lihat
Mau lihat apa? (monster/potion?): potion
ID | Name/Type | Stok | Harga

1 | strength | 1 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
Irrasshaimase! Selamat datang di SHOP!!
Pilih aksi (lihat/beli/keluar): beli
Jumlah O.W.C.A coin mu sekarang 8949

Mau beli apa? (monster/potion): potion
Masukkan id potion: 1
Masukkan jumlah: 1
Berhasil membeli item: 1 strength. Item sudah masuk ke inventory-mu!
```

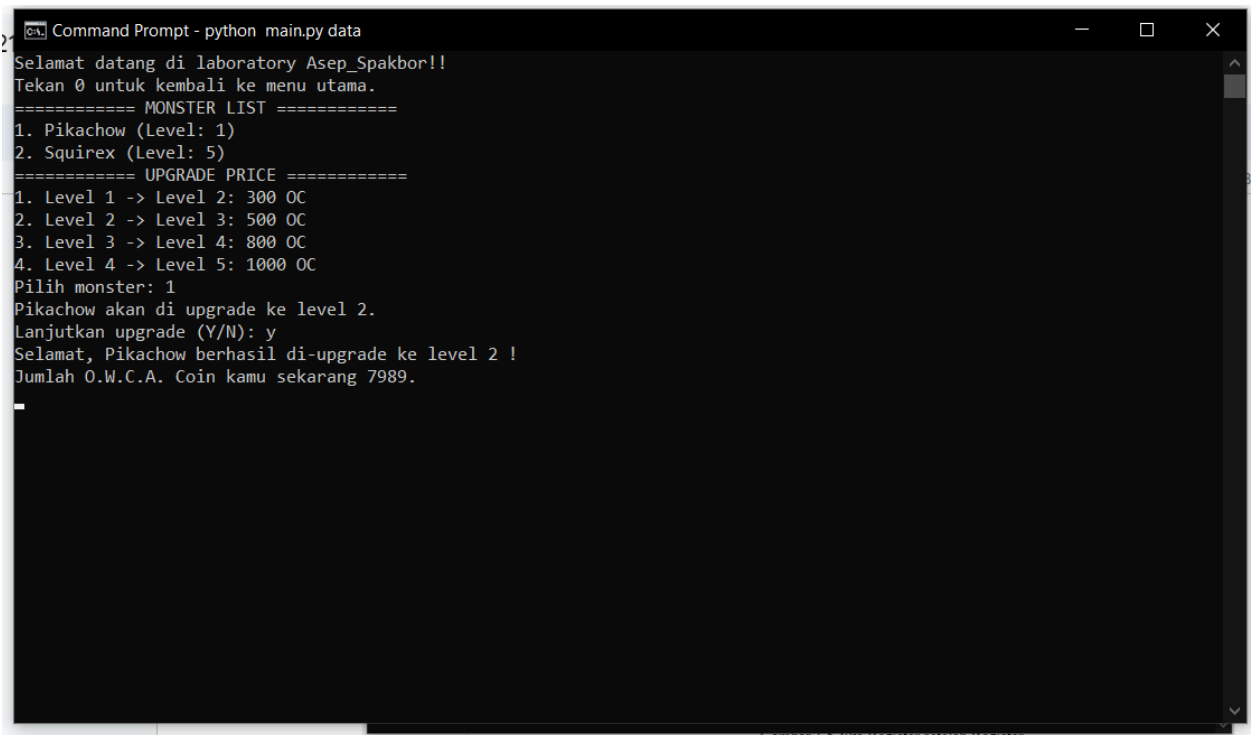
Gambar 8.9 Input Valid Beli Potion

## 9. F-11 Laboratory



```
Command Prompt - python main.py data
Selamat datang di laboratory Asep_Spakbor!!
Tekan 0 untuk kembali ke menu utama.
===== MONSTER LIST =====
1. Pikachow (Level: 1)
2. Squirex (Level: 5)
===== UPGRADE PRICE =====
1. Level 1 -> Level 2: 300 OC
2. Level 2 -> Level 3: 500 OC
3. Level 3 -> Level 4: 800 OC
4. Level 4 -> Level 5: 1000 OC
Pilih monster: 2
Maaf,monster yang Anda pilih sudah memiliki level maksimum
```

Gambar 9.1 Ketika Ingin Upgrade Lvl Max Monster



```
Command Prompt - python main.py data
Selamat datang di laboratory Asep_Spakbor!!
Tekan 0 untuk kembali ke menu utama.
===== MONSTER LIST =====
1. Pikachow (Level: 1)
2. Squirex (Level: 5)
===== UPGRADE PRICE =====
1. Level 1 -> Level 2: 300 OC
2. Level 2 -> Level 3: 500 OC
3. Level 3 -> Level 4: 800 OC
4. Level 4 -> Level 5: 1000 OC
Pilih monster: 1
Pikachow akan di upgrade ke level 2.
Lanjutkan upgrade (Y/N): y
Selamat, Pikachow berhasil di-upgrade ke level 2 !
Jumlah O.W.C.A. Coin kamu sekarang 7989.
```

Gambar 9.2 Input Valid di Laboratory

```
Command Prompt - python main.py data
Selamat datang di laboratory Asep_Spakbor!!
Tekan 0 untuk kembali ke menu utama.
===== MONSTER LIST =====
1. Pikachow (Level: 2)
2. Squirex (Level: 5)
===== UPGRADE PRICE =====
1. Level 1 -> Level 2: 300 OC
2. Level 2 -> Level 3: 500 OC
3. Level 3 -> Level 4: 800 OC
4. Level 4 -> Level 5: 1000 OC
Pilih monster: jawejfap
Input harus berupa bilangan bulat
```

```
2 Command Prompt - python main.py data
Selamat datang di laboratory Asep_Spakbor!!
Tekan 0 untuk kembali ke menu utama.
===== MONSTER LIST =====
1. Pikachow (Level: 2)
2. Squirex (Level: 5)
===== UPGRADE PRICE =====
1. Level 1 -> Level 2: 300 OC
2. Level 2 -> Level 3: 500 OC
3. Level 3 -> Level 4: 800 OC
4. Level 4 -> Level 5: 1000 OC
Pilih monster: 10
Tidak ada monster tersebut
```

Gambar 9.3 Input Tidak Valid di Laboratory

## 10. F-12 Shop Management

```
Halo Asep_Spakbor, Selamat datang kembali!
~~~~~
Terdapat beberapa pilihan aksi :
[1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar
Mau pilih aksi yang mana? : 100
Input anda salah! Ulangi.
█
```

Gambar 10.1: Pengujian fungsi saat input pilihan aksi tidak valid

```
[1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar
Mau pilih aksi yang mana? : 1
Mau lihat apa? (monster/potion) : sesuatu
Ups, Pilih antara monster atau potion!
. . .
Mau lihat apa? (monster/potion) : █
```

Gambar 10.2: Pengujian fungsi saat input pilihan aksi valid tetapi input lihat (monster/potion) tidak valid

| ID                                                     | Name/Type | ATK Power | DEF Power | HP   | Stok | Harga |
|--------------------------------------------------------|-----------|-----------|-----------|------|------|-------|
| 1                                                      | Pikachow  | 125       | 10        | 600  | 10   | 500   |
| 2                                                      | Bulbu     | 50        | 50        | 1200 | 4    | 700   |
| 3                                                      | Zeze      | 300       | 10        | 100  | 3    | 1000  |
| 4                                                      | Zuko      | 100       | 25        | 800  | 8    | 550   |
| 5                                                      | Chacha    | 80        | 30        | 700  | 7    | 600   |
| [1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar |           |           |           |      |      |       |
| Mau pilih aksi yang mana? : █                          |           |           |           |      |      |       |

Gambar 10.3: Pengujian fungsi saat input lihat (monster/potion) adalah monster



```

ID | Name/Type | Stok | Harga
-----
1 | strength | 10 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
[1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar
Mau pilih aksi yang mana? : 

```

Gambar 10.4: Pengujian fungsi saat input lihat (monster/potion) adalah potion

```

Memuat data Monster yang belum ada di shop
ID | Name/Type | ATK Power | DEF Power | HP
-----
6 | Bimosaurus | 175 | 30 | 600
7 | Arceus | 100 | 10 | 1000
8 | Squirex | 250 | 20 | 500
9 | Mewthree | 100 | 50 | 200
10 | Luigi | 150 | 30 | 700
Masukkan id monster yang ingin ditambah ke dalam shop : 

```

Gambar 10.5: Pengujian fungsi saat input pilih aksi adalah tambah

```

Masukkan id monster yang ingin ditambah ke dalam shop : banyak
Input harus berupa angka. Silakan coba lagi.
Masukkan id monster yang ingin ditambah ke dalam shop : 10
Masukkan stok baru monster : banyak
Input harus berupa angka. Silakan coba lagi.
Masukkan stok baru monster : 5
Masukkan harga baru monster : mahal
Input harus berupa angka. Silakan coba lagi.
Masukkan harga baru monster : 500
Proses menambahkan Luigi ke dalam shop telah berhasil!
[1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar
Mau pilih aksi yang mana? : 

```

Gambar 10.6: Pengujian fungsi saat input id, stok\_baru, dan harga\_baru tidak valid atau valid dan masih ada monster/potion yang belum termasuk dalam shop

```

[1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar
Mau pilih aksi yang mana? : 2
Mau tambah apa? (monster/potion) : potion
Semua item telah berada di shop.

```

Gambar 10.7: Pengujian fungsi saat tidak ada monster/potion yang belum termasuk dalam shop

```
Memuat keadaan Monster saat ini...
ID | Name/Type | ATK Power | DEF Power | HP | Stok | Harga
-----
1 | Pikachow | 125 | 10 | 600 | 10 | 500
2 | Bulbu | 50 | 50 | 1200 | 4 | 700
3 | Zeze | 300 | 10 | 100 | 3 | 1000
4 | Zuko | 100 | 25 | 800 | 8 | 550
5 | Chacha | 80 | 30 | 700 | 7 | 600
Masukkan id monster yang ingin diubah : seratus
Input harus berupa angka. Silakan coba lagi.
Masukkan id monster yang ingin diubah : -90
Tidak tersedia id tersebut, pilih id yang lain
█
```

Gambar 10.8: Pengujian fungsi saat pilihan aksi adalah ubah monster/potion dan input id tidak valid

```
ID | Name/Type | Stok | Harga
-----
1 | strength | 10 | 50
2 | resilience | 5 | 30
3 | healing | 3 | 20
Urutan ke berapa yang ingin diganti? : 2
Masukkan stok baru potion : banyak banget
Input harus berupa angka. Silakan coba lagi.
Masukkan stok baru potion : 6
Masukkan harga baru potion : mahal banget bro
Input harus berupa angka. Silakan coba lagi.
Masukkan harga baru potion : 100
Pembaharuan data berhasil dilakukan pada potion jenis resilience dengan stok baru berjumlah
6 dan harga baru 100
█
```

Gambar 10.9: Pengujian fungsi saat stok\_baru dan harga\_baru tidak kosong

```
Masukkan id monster yang ingin diubah : 2
Masukkan stok baru monster : 5
Masukkan harga baru monster :
Pembaharuan data berhasil dilakukan pada monster Bulbu dengan stok baru 5
█
```

Gambar 10.10: Pengujian fungsi saat harga\_baru kosong dan stok\_baru tidak kosong

```
Masukkan id monster yang ingin diubah : 2
Masukkan stok baru monster :
Masukkan harga baru monster : 550
Pembaharuan data berhasil dilakukan pada monster  Bulbu  dengan harga baru  550
█
```

Gambar 10.11: Pengujian fungsi saat harga\_baru tidak kosong dan stok\_baru kosong

```
Urutan ke berapa yang ingin diganti? : 3
Masukkan stok baru potion :
Masukkan harga baru potion :
Tidak ada perubahan
█
```

Gambar 10.12: Pengujian fungsi saat harga\_baru dan stok\_baru yang kosong

```
Pilih ID monster yang ingin dihapus : 4
Anda yakin ingin menghapus Zuko dari shop? (y/n) : y
Done, Zuko telah berhasil dihapus dari shop.
█
```

Gambar 10.13: Pengujian fungsi saat pilihan aksi hapus monster/potion dengan input id valid dan yakin ingin menghapus (y)

```
Pilih ID potion yang ingin dihapus : 1
Anda yakin ingin menghapus strength dari shop? (y/n) : n
strength dibatalkan untuk dihapus dari shop.
█
```

Gambar 10.14: Pengujian fungsi saat pilihan aksi hapus monster/potion dengan input id valid dan tidak yakin ingin menghapus (n)

```
Pilih ID monster yang ingin dihapus : 5
Anda yakin ingin menghapus Chacha dari shop? (y/n) : waduh
Input anda salah, ulangi!
█
```

Gambar 10.15: Pengujian fungsi saat pilihan aksi hapus monster/potion dengan input id valid dan input yakin tidak valid

```
Pilih ID potion yang ingin dihapus : wkwk
Input harus berupa angka. Silakan coba lagi.
Pilih ID potion yang ingin dihapus : -10
Tidak tersedia id tersebut, pilih id yang lain.
█
```

Gambar 10.16: Pengujian fungsi saat input id tidak valid

```
[1] lihat, [2] tambah, [3] ubah, [4] hapus, [5] keluar
Mau pilih aksi yang mana? : keluar
Sampai jumpa lagi, Asep_Spakbor...
█
```

Gambar 10.17: Pengujian fungsi saat pilihan aksi adalah keluar

## 11. F-13 Monster Management

```
Selamat Datang Para Agen
Di sini adalah tempat database para monster.
-----
. . .
[1] Tampilkan semua monster yang ada
[2] Tambahkan monster baru
[3] Untuk kembali ke menu admin
Anda ingin pilih aksi mana (1/2/3)? : wkwkwk
Input anda salah, ulangi!
█
```

Gambar 11.1 Pengujian fungsi saat input awal aksi tidak valid

```
Memuat data Monster yang belum ada di shop
ID | Name/Type | ATK Power | DEF Power | HP
-----
1 | Pikachu | 125 | 10 | 600
2 | Bulbu | 50 | 50 | 1200
3 | Zeze | 300 | 10 | 100
4 | Zuko | 100 | 25 | 800
5 | Chacha | 80 | 30 | 700
6 | Bimosaurus | 175 | 30 | 600
7 | Arceus | 100 | 10 | 1000
8 | Squirex | 250 | 20 | 500
9 | Mewthree | 100 | 50 | 200
10 | Luigi | 150 | 30 | 700
. . .
[1] Tampilkan semua monster yang ada
[2] Tambahkan monster baru
[3] Untuk kembali ke menu admin
Anda ingin pilih aksi mana (1/2/3)? : █
```

Gambarr 11. 2 Pengujian fungsi saat input aksi tampilkan monster (1)

```

[2] Tambahkan monster baru
[3] Untuk kembali ke menu admin
Anda ingin pilih aksi mana (1/2/3)? : 2
... Proses pembuatan monster baru dimulai ...

Nama/Type Monster Baru : Kurama

ATK Power Monster Baru : wkwkwk
ATK Power harus dalam angka yang bernilai positif...
Silakan coba lagi!
ATK Power Monster Baru : 500

DEF Power Monster Baru (0-50) : 100
Def Power Monster harus bernilai 0-50...
Silakan coba lagi!
DEF Power Monster Baru (0-50) : 45
Nilai HP Monster Baru: tinggi
Input harus berupa bilangan bulat!
Nilai HP Monster Baru: 1000

```

Gambar 11.3 Pengujian fungsi saat pilihan aksi adalah tambah monster baru (2) dengan variasi input yang valid dan tidak valid

```

Ingin menambahkan Monster baru ke database? (Y/N) : wkwkwk
Perintah anda salah, pilih (Y atau N)
Ingin menambahkan Monster baru ke database? (Y/N) : 

```

Gambar 11.4 Pengujian fungsi saat semua input valid dan input ingin menambahkan monster ke database tidak valid

```

Ingin menambahkan Monster baru ke database? (Y/N) : y
Monster baru berhasil ditambahkan ke database!

```

Gambar 11.5 Pengujian fungsi saat input nya adalah y

```

Ingin menambahkan Monster baru ke database? (Y/N) : n
Monster baru gagal ditambahkan ke database!

```

Gambar 11.6 Pengujian fungsi saat input nya adalah n

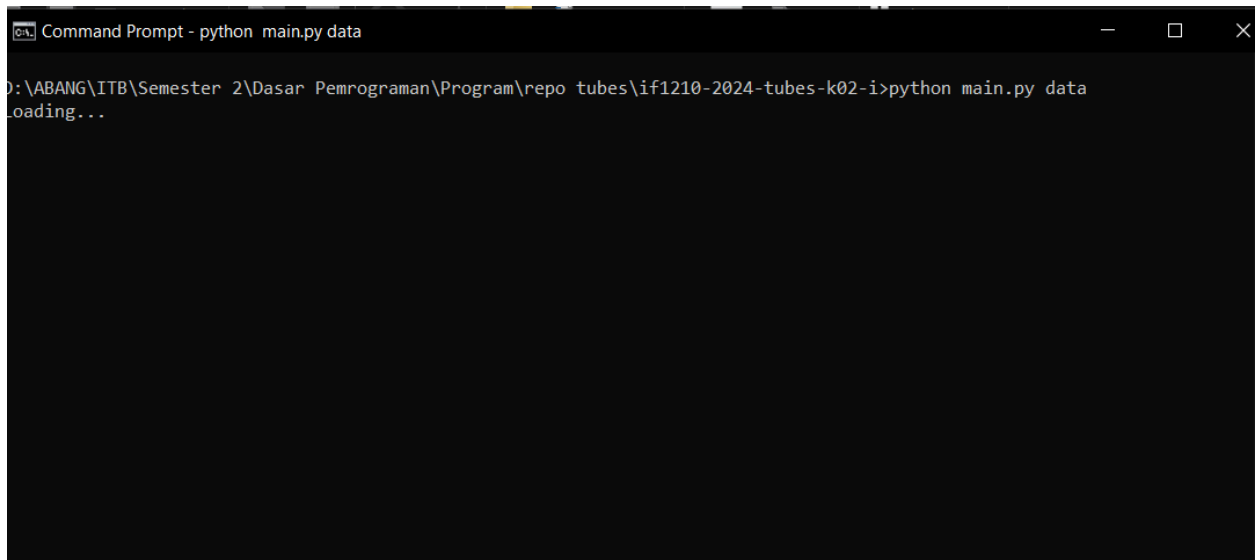
```

. . .
[1] Tampilkan semua monster yang ada
[2] Tambahkan monster baru
[3] Untuk kembali ke menu admin
Anda ingin pilih aksi mana (1/2/3)? : 3
Keluar dari Monster Management
. . .

```

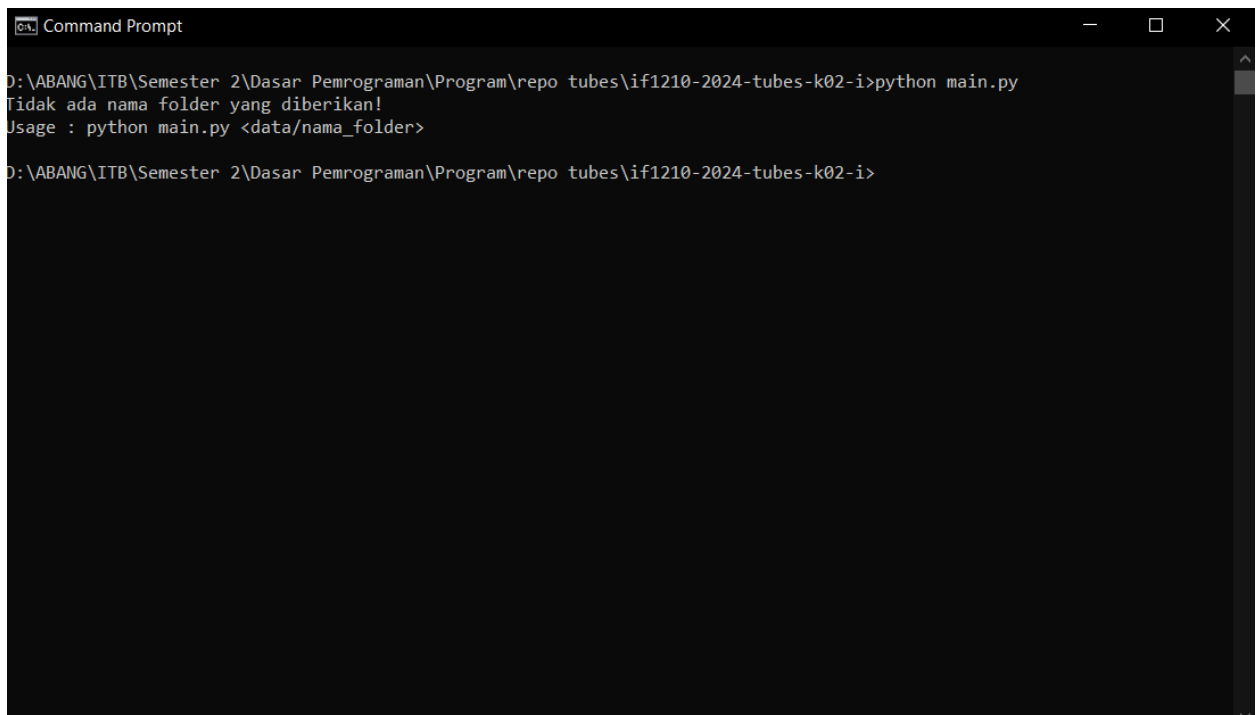
Gambar 11.7 Pengujian fungsi saat input pilihan aksi adalah kembali ke menu admin

## 12. F-14 Load



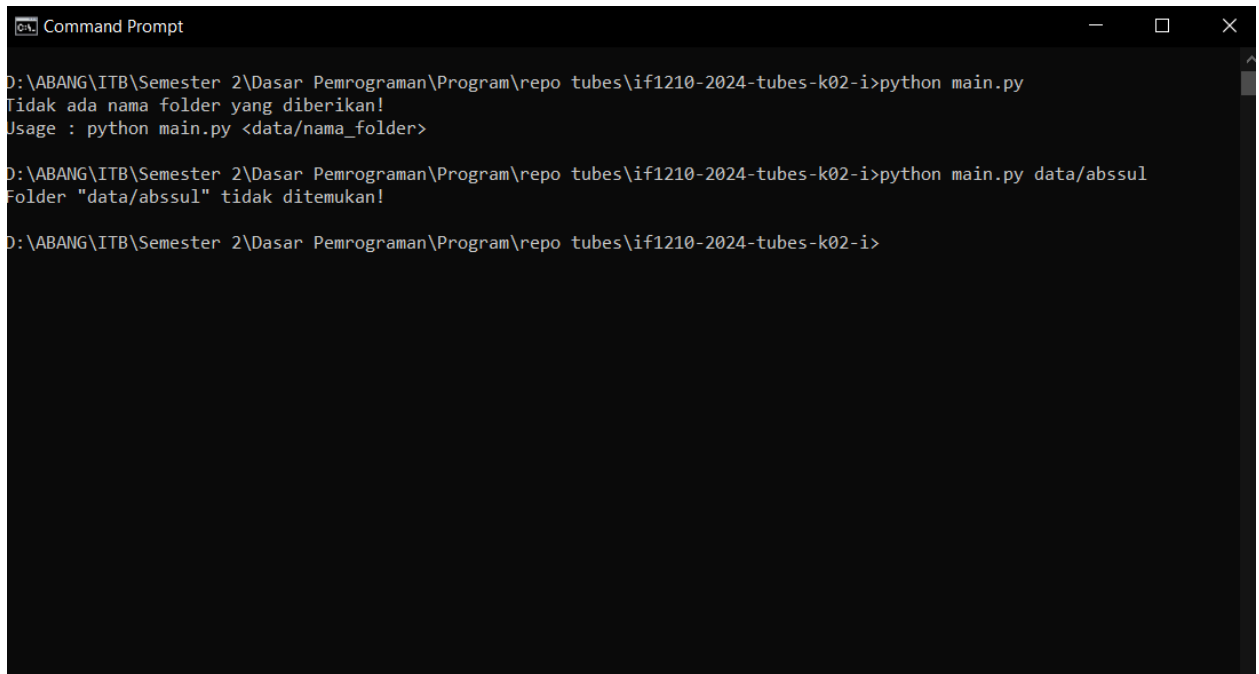
```
Command Prompt - python main.py data
D:\ABANG\ITB\Semester 2\Dasar Pemrograman\Program\repo tubes\if1210-2024-tubes-k02-i>python main.py data
loading...
```

Gambar 12.1 Uji Load



```
Command Prompt
D:\ABANG\ITB\Semester 2\Dasar Pemrograman\Program\repo tubes\if1210-2024-tubes-k02-i>python main.py
Tidak ada nama folder yang diberikan!
Usage : python main.py <data/nama_folder>
D:\ABANG\ITB\Semester 2\Dasar Pemrograman\Program\repo tubes\if1210-2024-tubes-k02-i>
```

Gambar 12.2 Uji Load jika tidak ada Input



```
Command Prompt
D:\ABANG\ITB\Semester 2\Dasar Pemrograman\Program\repo tubes\if1210-2024-tubes-k02-i>python main.py
Tidak ada nama folder yang diberikan!
Usage : python main.py <data/nama_folder>

D:\ABANG\ITB\Semester 2\Dasar Pemrograman\Program\repo tubes\if1210-2024-tubes-k02-i>python main.py data/abssul
Folder "data/abssul" tidak ditemukan!

D:\ABANG\ITB\Semester 2\Dasar Pemrograman\Program\repo tubes\if1210-2024-tubes-k02-i>
```

Gambar 12.3 Uji untuk mengecek apakah terdapat folder atau tidak

### 13. F-15 Save



```
Command Prompt - python main.py data
Selamat Datang di program OWCA!

      OWCA

[ Login ] [ Help ] [ Register ] [ Menu ] [ Logout ]

[Exit]

Masukkan command (lowercase): exit
Apakah anda mau melakukan penyimpanan file yang sudah diubah? (y/n): y
Masukkan nama folder: testing123
Saving...
Membuat folder data/testing123.
Berhasil menyimpan data di folder data/testing123!
Selamat tinggal agent !
```

Gambar 13.1 Uji Input Valid pada Save



```
Command Prompt - python main.py data
Selamat Datang di program OWCA!

      OWCA

[ Login ] [ Help ] [ Register ] [ Menu ] [ Logout ]
[Exit]

Masukkan command (lowercase): exit
Apakah anda mau melakukan penyimpanan file yang sudah diubah? (y/n): y
Masukkan nama folder: \139joqw
Input invalid, gunakan nama lain!
Masukkan nama folder:
```

Gambar 13.2 Uji Save ketika Input Tidak Valid



## 14. F-16 Exit



Gambar 14.1 Uji Untuk Exit



Gambar 14.2 Uji Input Tidak Valid untuk Exit

## LAMPIRAN

### 1. Lampiran MoM Asistensi Pertama

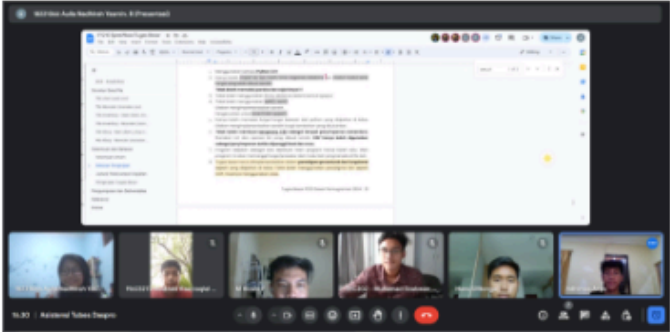
[Link Form MoM Asistensi 1](#)

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

Nomor Asistensi : 1  
No. Kelompok/Kelas : K02-I  
Tanggal asistensi : 3 May 2024

| Anggota kelompok   | NIM / Nama (Hanya yang Hadir)          |
|--------------------|----------------------------------------|
| 1                  | 19623142 / Adhimas Aryo Bimo           |
| 2                  | 19623132 / Hans Joseph B. W. Silitonga |
| 3                  | 19623262 / Muhammad Rizain Firdaus     |
| 4                  | 19623272 / Mudzaki Kaarzaqiel Hakim    |
| 5                  | 16523252 / Muhamad Dzakwan Musfajra    |
| 6                  |                                        |
| Asisten pembimbing | 18221066 / Aulia Nadhirah              |

Catatan Asistensi:

|                                                                                                                                                                       |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Rangkuman Diskusi</b>                                                                                                                                              |
| Membahas batasan-batasan apa saja yang tidak boleh digunakan.<br>Tips agar bisa bagi-bagi tugas secara baik dalam kerja kelompok<br>user_id yang dibuat harus berurut |
| <b>Tindak Lanjut</b>                                                                                                                                                  |
| Mulai membagi tugas<br>Merevisi generate_id agar menghasilkan id yang berurut                                                                                         |
| <b>Dokumentasi</b>                                                                                                                                                    |
|                                                                                   |

## 2. Lampiran MoM Asistensi Kedua


[Link Lampiran MoM Asistensi 2](#)

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

Nomor Asistensi : 2  
No. Kelompok/Kelas : K02-I  
Tanggal asistensi : 12 May 2024

|                    |                                        |
|--------------------|----------------------------------------|
| Anggota kelompok   | NIM / Nama (Hanya yang Hadir)          |
| 1                  | 19623142 / Adhimas Aryo Bimo           |
| 2                  | 19623132 / Hans Joseph B. W. Silitonga |
| 3                  | 19623262 / Muhammad Rizain Firdaus     |
| 4                  | 19623272 / Mudzaki Kaarzaqiel Hakim    |
| 5                  | 16523252 / Muhamad Dzakwan Musfajra    |
| 6                  |                                        |
| Asisten pembimbing | 18221066 / Aulia Nadhirah              |

Catatan Asistensi:

|                                                                                      |
|--------------------------------------------------------------------------------------|
| <b>Rangkuman Diskusi</b>                                                             |
| Merevisi penggunaan batasan dalam kode terutama pada global<br>Membahas progress     |
| <b>Tindak Lanjut</b>                                                                 |
| Mulai menentukan deadline pengerjaan<br>Mulai mengerjakan fungsi-fungsi detail       |
| <b>Dokumentasi</b>                                                                   |
|  |