

GamaFit a “Smart, Simple, Efficient” Health and Fitness System for Tracking Your Daily Activities at Universitas Gadjah Mada

Rabbani Nur Kumoro^{#1}, Ferdio Azka Adhika ^{#1}, Antonius Teddy Kurniawan^{#1}, Mohammad Matim Molayari^{#1}, Mohammad Haryodimas Dewantoro^{#1}, Najma Syifa Ardini^{#1}, and Ananda Salman Alfarezi^{#2}

^{#1}*Department of Computer Sciences and Electronics, Universitas Gadjah Mada Building C, 4th Floor North Sekip, Bulaksumur Yogyakarta Indonesia 55281*

^{#2}*Department of Chemistry, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, North Sekip, Bulaksumur, Yogyakarta 55281*

Abstract - Over the past few years, living a healthy lifestyle has become trendy. Nowadays, being healthy means being beautiful, successful, and fit. People are becoming increasingly aware of diseases caused by obesity and sedentary office work. These and other health-related problems encourage millions of people around the world to adopt healthier lifestyles. This motivates us to create program that could help combat this negative trend.

Keywords – GamaFit, Health-Tracking App, BMI-Counter, Step-Counter, Sleep-Tracker, Program, Computer Science, C++

I. INTRODUCTION

The COVID-19 pandemic that has already started in almost 2 years ago, other than resulting in a devastating disease that kills millions, it also produces other health-related problem that is lowering the healthy lifestyle. A recent study conducted in 2020 has reported that roughly 22% of adults report having gained weight during the COVID-19 pandemic [1]. The same study also shown that the most common factors

are: lack of sleep, decreased physical activity, excessive snacking, and stress-eating. Other than that a social media trend called “quarantine-15” also worsen the crisis that makes people less aware of living healthy [2].

With all of that factors in mind it motivates us to create GamaFit. GamaFit is UGM’s first health tracking app that makes living more simple, efficient, and easy. As a health and fitness system, GamaFit provides an easy way to use and interact. The system will make our life a whole lot easier and better. Think of GamaFit as a tracker for you to store your health and compute the average analysis of users' daily activities throughout their campus life.

The remainder of this report paper is laid out as follows. In Section II, we’ll discuss GamaFit’s code functions in detail. Section III will give an overview of the application and how it should work. Section IV will give conclusion to the paper.

II. SOLUTION AND METHOD

2.1 Overview

GamaFit is a health-tracking app that could help the user count their steps, track their sleep, and count body mass index.

The first step in using GamaFit is to sign-up to GamaFit. The next step is to log in to their GamaFit account with the username and password that they already created. After the necessary steps are completed the user can now access the main menu, in which the user can choose our services:

- **Your Profile:** Displays the user's account information.
- **Step Counter:** Counts the number of steps, then calculates the calories burned and distance.
- **Sleep Tracker:** Tracks the user's sleep quality by inputting sleep time and feelings at that night. After that, it will give sleeping points and feedback on the user's sleep quality.
- **Body Mass Index:** Identifies the user's Body Mass Index by inputting weight, height, and gender. Then, it will display your BMI, show if the user's body is obese, overweight, normal, or underweight, and give recommendations for the user.
- **Help:** Displays the tutorials on how to navigate the GamaFit program.
- **Log-out:** Users are able to log out from GamaFit

After finishing using all the services, the user will log out from GamaFit, and the program will end.

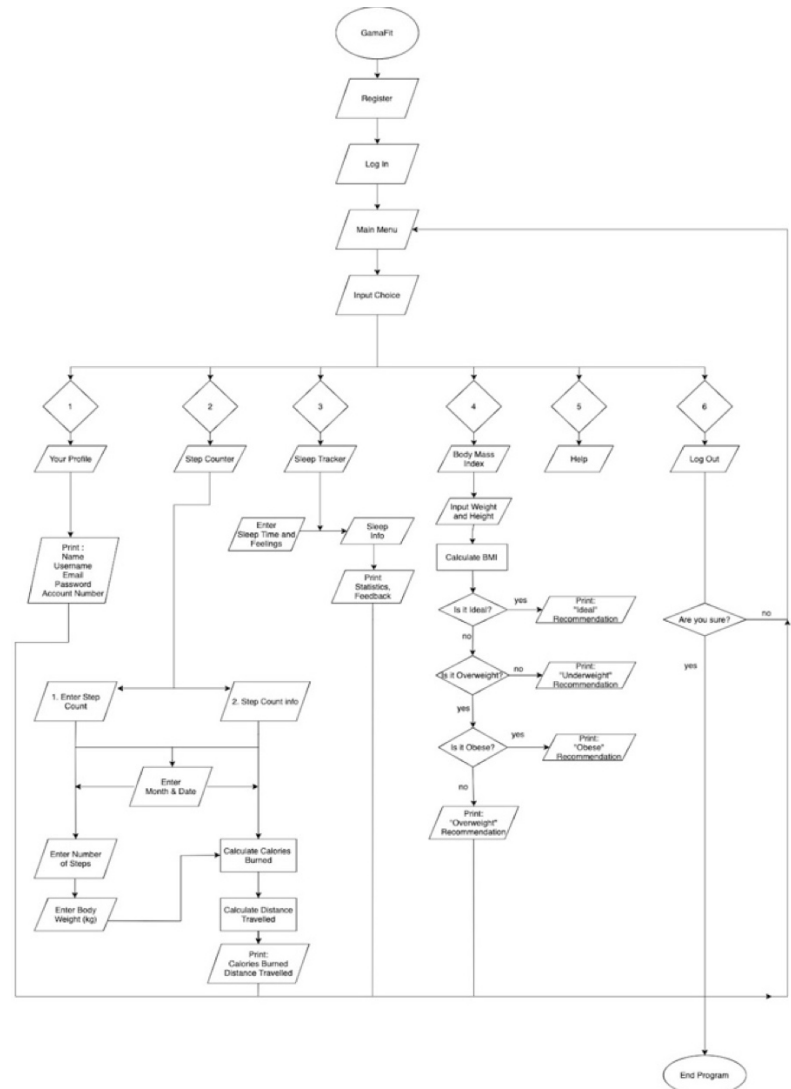


Fig. 1.1. GamaFit's Flowchart

III. IMPLEMENTATION AND RESULTS

To Implement C++ for making a health tracking app we need to create some function and struct to achieve the goal desired goal

3.1 Code Functions, Struct, and Constants

Below are the functions that is used for some features that GamaFit offers.

Here are the functions that we use:

1. void bmi_info

Outputs the user's BMI and feedback from the data that they already input before.

```
void bmi_info(bmi_count &bmi)
{
    string key;
    int i,j;
    double bmscore;
    char gen;
    cout << "\nBMI (Body Mass Index) Info\n" << "-----\n\n";
    cout << "Please Select The Month (1-12): ";
    cin >> j;
    cout << "Please Select The Date (1-31): ";
    cin >> i;

    if (i<1||i>31||j<1||j>12)
    {
        cout << "Invalid Month or Date!\n\n";
        return bmi_info(bmi);
    }

    i--;j--;
    bmi.bmscore[j][i]=bmi.weight[j][i]/(bmi.height[j][i]*bmi.height[j][i]);
    gen=bmi.gender[j][i];
    bmscore=bmi.bmscore[j][i];
    cout << endl;
    cout << "User Weight: " << bmi.weight[j][i] << " kg\n";
    cout << "User Height: " << bmi.height[j][i] << " cm\n";
    if (gen=='F'||gen=='f')
    {
        cout << "User Gender: Female\n";
    }

    else
    {
        cout << "Unknown\n";
    }

    cout << "User BMI: " << bmscore << endl;
    if (bmscore >= 0.0018 && bmscore <= 0.0025)
    {
        cout << "BMI Category : Normal" << endl;
        cout << "Recommendation : Stay Fit and Healthy." << endl;
    }
    else if (bmscore >= 0.0025 && bmscore <= 0.0030)
    {
        cout << "BMI Category : Overweight" << endl;
        cout << "Recommendation : Increase Your Protein Consumption To Help Build Muscle." << endl;
    }
    else if (bmscore > 0.0030)
    {
        cout << "BMI Category : Obesity" << endl;
        cout << "Recommendation : Eat More Vegetables and Fruits, Also Add More Grain." << endl;
    }
    else if (bmscore < 0.0018)
    {
        cout << "BMI Category : Underweight" << endl;
        cout << "Recommendation : Increase Your Carbs and Protein Consumption." << endl;
    }

    cout << "\n\nEnter any key(s) to exit: ";
    cin >> key;
}
```

2. void bmi_calculate

Inputs the user's weight, height, and gender-based on a certain date and month and stores it in bmi_count struct.

```
void bmi_calculate(bmi_count &bmi)
{
    string key;
    int i,j;
    char gen;
    cout << "\nBody Data Input\n" << "-----\n\n";
    cout << "Please Select The Month (1-12): ";
    cin >> j;
    cout << "Please Select The Date (1-31): ";
    cin >> i;
    if (i<1||i>31||j<1||j>12)
    {
        cout << "Invalid Month or Date!\n\n";
        return bmi_calculate(bmi);
    }

    i--;j--;

    cout << "\nPlease Enter The Following Info: \n\n";
    cout << "Input Weight (kg) : ";
    cin >> bmi.weight[j][i];
    cout << "Input Height (cm) : ";
    cin >> bmi.height[j][i];
    cout << "Input Gender (M/F): ";
    cin >> gen;
    gen=bmi.gender[j][i];
    cout << "Enter any key(s) to exit: ";
    cin >> key;
}
```

3. void bmi_menu

Displays the user menu to navigate features within the BMI counter.

```
void bmi_menu(bmi_count &bmi)
{
    char choice_3;
    string key;

    cout << "\nBMI (Body Mass Index)\n";
    cout << "-----\n\n";
    cout << "1. Enter Body Data\n" << "2. BMI Info\n" << "3. Return to Main Menu\n";
    cout << "Please Choose The Option (1-3): ";
    cin >> choice_3;

    switch (choice_3)
    {
        case '1':
            bmi_calculate(bmi);
            bmi_menu(bmi);
            break;
        case '2':
            bmi_info(bmi);
            bmi_menu(bmi);
            break;
        case '3':
            return;
        default:
            cout << "Invalid input! " << endl;
            bmi_menu(bmi);
    }
}
```

4. void sleep_tracker

Inputs the user's sleeping hours, sleeping minutes, and their feeling based on a certain date and month and stores it in sleep_track struct.

```
void sleep_tracker(sleep_track &sleep)
{
    string key;
    int i,j,k;
    cout << "\nSleep Tracker Input\n" << "-----\n\n";
    cout << "Please Select The Month (1-12): ";
    cin >> j;
    cout << "Please Select The Date (1-31): ";
    cin >> i;
    if (i<1||i>31||j<1||j>12)
    {
        cout << "Invalid Month or Date!\n\n";
        return sleep_tracker(sleep);
    }

    i--;j--;

    string feelings[4]={"Restful","Neutral","Restless","Irritated"};

    cout << "\nPlease Enter The Following Info: \n\n";
    cout << "Input Hour of Sleep Time (0-24): ";
    cin >> sleep.sleep_hour[j][i];
    cout << "Input Minute of Sleep Time (0-59): ";
    cin >> sleep.sleep_minute[j][i];
}
```

```

if (sleep.sleep_hour[j][i]<0||sleep.sleep_hour[j][i]>24||sleep.sleep_minute[j][i]>59)
{
    cout<<"The Entered Value is Invalid, Please Try Again! \n";
    sleep_tracker(sleep);
}
cout << "What Do You Feel That Night? (1. Restful 2. Neutral 3. Restless 4. Irritate)";
cin >> k;
if (k==0||k>4)
{
    cout << "The Entered Value is Invalid, Please Try Again! \n";
    sleep_tracker(sleep);
}
sleep.feeling[j][i]=feelings[k-1];
cout << "Enter any key(s) to exit: ";
cin >> key;
}

```

5. void sleep_info

Outputs the user's sleeping time, sleeping points, and feedback from the data that they already input before.

```

void sleep_info (sleep_track &sleep)
{
    string key;
    int i,j,sleep_point=100,minute;
    cout << "\nSleep Tracker Info Info \n" << "-----\n\n";
    cout << "Please Select The Month (1-12): ";
    cin >> j;
    cout << "Please Select The Date (1-31): ";
    cin >> i;

    if (i<1||i>31||j<1||j>12)
    {
        cout << "Invalid Month or Date! \n\n";
        return sleep_info(sleep);
    }

    i-=1;j-=1;
    cout << endl;
    cout << "I Slept For " << sleep.sleep_hour[j][i] << " Hours and "
    << sleep.sleep_minute[j][i] << " Minutes" << endl;
    cout << "While Asleep, I Feel " << sleep.feeling[j][i] << endl;
    minute=(sleep.sleep_hour[j][i]*60)+sleep.sleep_minute[j][i];

    if (minute<48060minute=0)
    {
        sleep_point=100+minute/480;
        cout << "Sleeping Points: " << sleep_point << endl;
        if (minute==480)
        {
            cout << "You Sleep Adequately :, Please Maintain Your Sleeping Habbit ";
        }
        else if (minute<48060minute=360)
        {
            cout << "You Are Slightly Lacking Sleep, Please Try to Aim For 8 Hours";
        }
        else
        {
            cout << "You Are Sleep Deprived :, Please Add Your Sleep Time Immediately and Aim For 8 ";
        }
        cout << "\n\nEnter any key(s) to exit: ";
        cin >> key;
    }
}

```

6. void sleep_tracker_menu

Displays the user menu to navigate features within the sleep tracker.

```

void sleep_tracker_menu(sleep_track &sleep)
{
    char choice_2;
    string key;

    cout << "\nSLEEP TRACKER\n";
    cout << "-----\n\n";
    cout << "1. Enter Sleep Time\n" << "2. Sleep Info\n" << "3. Return to Main Menu\n";
    cout << "Please Choose The Option (1-3): ";
    cin >> choice_2;

    switch (choice_2)
    {
        case '1':
            sleep_tracker(sleep);
            sleep_tracker_menu(sleep);
            break;
        case '2':
            sleep_info(sleep);
            sleep_tracker_menu(sleep);
            break;
        case '3':
            return;
        default:
            cout << "Invalid input! " << endl;
            sleep_tracker_menu(sleep);
    }
}

```

7. void step_counter

Inputs the user's number of steps and weight based on a certain date and month, then calculates calories burned and distance traveled from the input and stores it in step_count struct.

```

void step_counter(step_count &step)
{
    string key;
    int i,j;
    cout << "\nStep Counter Input\n" << "-----\n\n";
    cout << "Please Choose The Month (1-12): ";
    cin >> j;
    cout << "Please Choose The Date (1-31): ";
    cin >> i;

    if (i<1||i>31||j<1||j>12)
    {
        cout << "Invalid Month or Date! \n\n";
        return step_counter(step);
    }

    i-=1;j-=1;
    cout << "\nPlease Enter The Following Info: \n\n";
    cout << "Number of Step: ";
    cin >> step.step_num[j][i];
    cout << "Body Weight (kg): ";
    cin >> step.weight[j][i];
    step.calories[j][i]=step.step_num[j][i]*step.weight[j][i]/72.6*0.04;
    step.distance[j][i]=step.step_num[j][i]*0.001524;
    cout << "Enter any key(s) to exit: ";
    cin >> key;
}

```

8. void step_counter_info

Outputs the user's number of steps, weight, calories burned and distance traveled based on the data that they already input before, according to the desired date.

```

void step_counter_info(step_count &step)
{
    string key;
    int i,j;
    cout << "\nStep Counter Info \n" << "-----\n\n";
    cout << "Please Select The Month (1-12): ";
    cin >> j;
    cout << "Please Select The Date (1-31): ";
    cin >> i;

    if (i<1||i>31||j<1||j>12)
    {
        cout << "Invalid Month or Date! \n\n";
        return step_counter_info(step);
    }

    i-=1;j-=1;
    cout << endl;
    cout << "Number of Steps: " << step.step_num[j][i] << endl;
    cout << "User Body Weight: " << step.weight[j][i] << " kg" << endl;
    cout << "Calories Burned: " << step.calories[j][i] << endl;
    cout << "Distance Travelled: " << step.distance[j][i] << " km" << endl;
    cout << "Enter any key(s) to exit: ";
    cin >> key;
}

```

9. void step_counter_menu

Displays the user menu to navigate features within the step counter.

```

void step_counter_menu(step_count &step)
{
    char choice;
    string key;

    cout << "\nSTEP COUNTER\n";
    cout << "=====\n\n";
    cout << "1. Enter Step Count\n" << "2. Step Count Info\n" << "3. Return to Main Menu\n";
    cout << "Please Choose the Option (1-3): ";
    cin >> choice;

    switch (choice)
    {
        case '1':
            step_counter(step);
            step_counter_menu(step);
            break;
        case '2':
            step_counter_info(step);
            step_counter_menu(step);
            break;
        case '3':
            return;
        default:
            cout << "Invalid Input! " << endl;
            step_counter_menu(step);
    }
}

```

10. void userprofile

Displays the user's account information to the console.

```

void userprofile(userid user_info)
{
    string key;
    cout << "\nYour Profile \n-----\n\n";
    cout << "Name: " << user_info.name<<endl;
    cout << "Username: " << user_info.username<<endl;
    cout << "Email: " << user_info.email<<endl;
    cout << "Password: " << user_info.password<<endl;
    cout << "Account Number: " <<
        user_info.account_no<<endl;
    cout << "Enter any key(s) to exit: ";
    cin >> key;
}

```

11. void help_screen

Displays the information regarding the navigation options in the main menu.

```

void help_screen()
{
    char temp;
    cout<<"\nHelp Screen\n";
    cout<<"=====\n\n";
    cout << "1. Sign Up and Log In" << endl;
    cout << "2. Sign up to create your GamaFit Account by filling the required data." << endl;
    cout << "3. Log in to your GamaFit account by filling your Username / Email and Password." << endl;
    cout << endl;
    cout << "4. Main Menu" << endl;
    cout << "5. Your Profile: Displays Your Account Information." << endl;
    cout << "6. Step Counter: Displays Your Step Counts, Distance Travelled, and Calories Burned." << endl;
    cout << "7. Sleep Tracker: Displays Your Sleep Time, Sleep Points, Feelings, and Feedback." << endl;
    cout << "8. Body Mass Index: Displays Your Weight, Height, BMI Results and Category, also a Feedback." << endl;
    cout << "9. Log Out: Ends the Program." << endl;
    cout << endl;
    cout<<"Enter any key(s) to continue: ";
    cin>>temp;
}

```

12. void mainscreen

Displays the main menu for the user to navigate between Gamafit's services.

```

void mainscreen(userid &user_info, step_count &step, sleep_track &sleep, bmi_count &bmi)
{
    char option,option_quit;
    string EnterStr;
    cout << endl << "MAIN MENU\n";
    cout << "=====\n\n";
    cout << "1. Your Profile" << endl;
    cout << "2. Step Counter" << endl;
    cout << "3. Sleep Tracker" << endl;
    cout << "4. Body Mass Index" << endl;
    cout << "5. Help" << endl;
    cout << "6. Log Out" << endl;
    cout << "\nSelect Options (1-5): ";
    cin >> option;
}

```

```

switch (option)
{
    case '1':
        userprofile(user_info);
        mainscreen(user_info, step, sleep, bmi);
        break;
    case '2':
        step_counter_menu(step);
        mainscreen(user_info, step, sleep, bmi);
        break;
    case '3':
        sleep_tracker_menu(sleep);
        mainscreen(user_info, step, sleep, bmi);
        break;
    case '4':
        bmi_menu(bmi);
        mainscreen(user_info, step, sleep, bmi);
        break;
    case '5':
        help_screen();
        mainscreen(user_info, step, sleep, bmi);
        break;
    case '6':
        cout << "\nAre you sure you want to quit this program? (Y/N)\n";
        cout << "Press (y) to quit or press any key(s) to cancel: ";
        cin >> option_quit;
        if (option_quit=='y')
        {
            cout << endl;
            cout << "Thank You For Using Our Service, Stay Healthy and Stay Safe (Y^)" << endl;
            break;
        }
    }
}

```

```

    }
    else
        mainscreen(user_info, step, sleep, bmi);
default:
    cout << "Invalid Input" << endl;
    mainscreen(user_info, step, sleep, bmi);
    break;
}
}

```

13. void login

Input the user's login information in order to use the service.

```

void login(userid &user_info)
{
    bool check=false;
    string username_input,pass_input;
    cout << "\nPlease Login \n";
    cout << "=====\n\n";
    while (!check)
    {
        cout << "Username or Email: ";
        cin >> username_input;
        cout << "Password: ";
        cin >> pass_input;

        if((username_input==user_info.username||username_input==user_info.email)&&pass_input==user_info.password)
            check=true;
        else
        {
            cout << "Your Input of Username or Email or Password Is Wrong!\n\n";
            login(user_info);
        }
        break;
    }
}

```

14. void signup

Allows the user to create a new account in order to use Gamafit.

```
void signup (userid &user_info)
{
    cout << "\nWelcome to GamaFit, Please Register to Enter \n";
    cout << "----- \n\n";
    cout << "Enter Name: ";
    getline(cin, user_info.name);
    cout << "Enter Username: ";
    cin >> user_info.username;
    cout << "Enter Email: ";
    cin >> user_info.email;
    cout << "Enter Password: ";
    cin >> user_info.password;

    user_info.account_no=(rand()%900000000)+1000000000;
    login(user_info);
}
```

15. int terminate_program

Terminates the program.

```
int terminate_program()
{
    cout<<"Bye";
    return 0;
}
```

Here are the struct that we use:

1. struct userid

Saves the user's name, username, email, and password in string data type, and a randomly generated account_no in integer data type.

```
struct userid
{
    string name;
    string username;
    string email;
    string password;
    int account_no;
};
```

2. struct step_count

Saves the user's step count, weight, calories, distance travelled in an MD array according to a specific month and date.

```
struct step_count
{
    int step_num[month][date];
    int weight[month][date];
    double calories[month][date];
    double distance[month][date];
};
```

3. struct sleep_track

Saves the user's sleep hour, sleep minute, and feeling in an MD array according to a specific month and date.

```
struct sleep_track
{
    int sleep_hour[month][date];
    int sleep_minute[month][date];
    string feeling[month][date];
};
```

4. struct bmi_count

Saves the user's weight, height, gender, and BMI Score in and MD array according to a specific month and date.

```
struct bmi_count
{
    double weight[month][date];
    double height[month][date];
    char gender[month][date];
    double bmiscore[month][date];
};
```

Here are some constants that we use:

1. const int month

The above constant is set to 12.

2. const int date

The above constant is set to 31.

3.2 Results

From the function, struct, and constants that is used to create the program, we can see the result in the form of a console program.

3.2.1 Users can register to create their accounts.

```
Welcome to GamaFit, Please Register to Enter
-----
Enter Name: Ananda Salman Alfarezi
Enter Username: rezi
Enter Email: ananda.salman.alfarezi@mail.ugm.ac.id
Enter Password: testrun1
```

3.2.2 Users can log in to their GamaFit account.

```
Please Login
-----
Username or Email: rezi
Password: testrun1
```

3.2.3 Users can view their profile and account information.

```
Your Profile
-----
Name: Ananda Salman Alfarezi
Username: rezi
Email: ananda.salman.alfarezi@mail.ugm.ac.id
Password: testrun1
Account Number: 104289383
Enter any key(s) to exit:
```

3.2.4 Users can access the main menu and choose one of the six available options.

```
MAIN MENU
-----
1. Your Profile
2. Step Counter
3. Sleep Tracker
4. Body Mass Index
5. Help
6. Log Out
Select Options (1-5):
```

3.2.5 Users can see the step count information.

```
STEP COUNTER
-----
1. Enter Step Count
2. Step Count Info
3. Return to Main Menu
Please Choose the Option (1-3):
```

3.2.6 Users can enter the total number of steps and their weight.

```
Step Counter Input
-----
Please Choose The Month (1-12): 11
Please Choose The Date (1-31): 26
Please Enter The Following Info:
Number of Steps: 500
Body Weight (kg): 58
Enter any key(s) to exit:
```

3.2.7 Users can see their calorie consumption after walking or jogging.

```
Step Counter Info
-----
Please Select The Month (1-12): 11
Please Select The Date (1-31): 26
Number of Steps: 500
User Body Weight: 58 kg
Calories Burned: 15.978
Distance Travelled: 0.762 km
Enter any key(s) to exit:
```

3.2.6 Users can use the sleep tracker.

```
SLEEP TRACKER
-----
1. Enter Sleep Time
2. Sleep Info
3. Return to Main Menu
Please Choose The Option (1-3):
```

3.2.8 Users can enter their sleep time and their feelings.

```
Sleep Tracker Input
-----
Please Select The Month (1-12): 11
Please Select The Date (1-31): 25
Please Enter The Following Info:
Input Hour of Sleep Time (0-24): 8
Input Minute of Sleep Time (0-59): 18
What Do You Feel That Night? (1. Restful 2. Neutral 3. Restless 4. Irritated):
1
Enter any key(s) to exit:
```

3.2.9 Users can see their sleep quality statistics.

```
Sleep Tracker Info Info
-----
Please Select The Month (1-12): 11
Please Select The Date (1-31): 25
I Slept For 8 Hours and 18 Minutes
While Asleep, I Feel Restful
Sleeping Points: 100
You Sleep Adequately :), Please Maintain Your Sleeping Habit
Enter any key(s) to exit:
```

3.2.10 Users can get feedback on their sleep quality.

```
Sleep Tracker Info Info
-----
Please Select The Month (1-12): 11
Please Select The Date (1-31): 24
I Slept For 3 Hours and 20 Minutes
While Asleep, I Feel Restless
Sleeping Points: 41
You Are Sleep Deprived :(, Please Add Your Sleep Time Immediately and Aim For 8 Hours
Enter any key(s) to exit:
```

3.2.11 Users can use Body Mass Index (BMI).

```

BMI (Body Mass Index)
=====
1. Enter Body Data
2. BMI Info
3. Return to Main Menu
Please Choose The Option (1-3): 

```

3.2.12 Users can enter their weight and height to calculate their body mass index (BMI).

```

Body Data Input
-----
Please Select The Month (1-12): 11
Please Select The Date (1-31): 26

Please Enter The Following Info:
Input Weight (kg) : 58
Input Height (cm) : 173
Input Gender (M/F): M
Enter any key(s) to exit: 

```

3.2.13 Users can find out their body mass index (BMI). Users can find out if their body is obese, overweight, normal, or underweight. Users can get body mass index (BMI) recommendations.

```

BMI (Body Mass Index) Info
-----
Please Select The Month (1-12): 11
Please Select The Date (1-31): 26

User Weight: 58 kg
User Height: 173 cm
Unknown
User BMI: 0.00193792
BMI Category : Normal
Recommendation : Stay Fit and Healthy.

Enter any key(s) to exit: 

```

3.2.14 Users can ask for help if needed.

```

Help Screen
-----
A. Sign Up and Log In
1. Sign up to create your GamaFit Account by filling the required data.
2. Log in to your GamaFit account by filling your Username / Email and Password.

B. Main Menu
1. Your Profile: Displays Your Account Information.
2. Step Counter: Displays Your Step Counts, Distance Travelled, and Calories Burned.
3. Sleep Tracker: Displays Your Sleep Time, Sleep Points, Feelings, and Feedback.
4. Body Mass Index: Displays Your Weight, Height, BMI Results and Category, also a Feedback.
5. Log Out: Ends the Program.

Enter any key(s) to continue: 

```

3.2.15 Users can return to the main menu.

```

1. Sign up to create your GamaFit Account by filling the required data.
2. Log in to your GamaFit account by filling your Username / Email and Password.

B. Main Menu
1. Your Profile: Displays Your Account Information.
2. Step Counter: Displays Your Step Counts, Distance Travelled, and Calories Burned.
3. Sleep Tracker: Displays Your Sleep Time, Sleep Points, Feelings, and Feedback.
4. Body Mass Index: Displays Your Weight, Height, BMI Results and Category, also a Feedback.
5. Log Out: Ends the Program.

Enter any key(s) to continue: 0

MAIN MENU
=====
1. Your Profile
2. Step Counter
3. Sleep Tracker
4. Body Mass Index
5. Help
6. Log Out

Select Options (1-5): 

```

3.2.16 Users can log off and exit.

```

MAIN MENU
=====
1. Your Profile
2. Step Counter
3. Sleep Tracker
4. Body Mass Index
5. Help
6. Log Out

Select Options (1-5): 6

Are you sure you want to quit this program? ('y'/'n')
Press (y) to quit or press any key(s) to cancel: y

Thank You For Using Our Service, Stay Healthy and Stay Safe (♥)

```

IV. CONCLUSION

We can conclude that GamaFit is a “Smart, Simple, Efficient” health-tracking system for tracking your daily activities that have the ability to track steps, sleep, and count BMI.

It came as a solution for Universitas Gadjah Mada’s students to live healthy by responding to the feedback of the program which gave recommendations on what the students should do. With that feature, our program can encourage students to live a healthy lifestyle.

ACKNOWLEDGEMENTS

We very much appreciate the guidance and advice that Mr. Wahyono, S. Kom., Ph.D., throughout the Programming course in this first semester. Thank you for all the knowledge that you have given to us.

We also are aware of the program drawbacks that it isn’t comparable to other health apps out there that uses complex programming languages and many design keys. But throughout the project completion we are really proud of what we can create and of how it could be used and applied by the users.

REFERENCES

- [1] Z. Zachary *et al.*, “Self-quarantine and weight gain related risk factors during the COVID-19 pandemic,” *Obes. Res. Clin. Pract.*, vol. 14, no. 3, pp. 210–216, 2020, doi: 10.1016/j.orcp.2020.05.004.
- [2] R. L. Pearl, “Weight Stigma and the ‘Quarantine-15,’” *Obesity*, vol. 28, no. 7, pp. 1180–1181, 2020, doi: 10.1002/oby.22850.