**I N T R O D U C T I O N**

Now a day’s peoples are very busy in their daily work they don’t care about their health. They even don’t know the about the amount of water they should consume in a day for a good health. That’s why we have developed an application which reminds us to take a glass of water frequently. This application is very useful to busy people of now days. It contains various features like notifying the users to drink water, setting the amount of water you would like to consume in a day and alerting users to drink water

**O B J E C T I V E S**

* To keep people healthy.
* To prevent them from diseases
* Maintain the water retention of water
* To remind the people about the importance of water in their life
* To make peoples habit to drink water

SYSTEM ANALISIES

Existing System

When I analyzed that if user somehow forget to set reminder on the software it will be unable to remind him to take water in time that its disadvantage

Proposed System

As we have analyzed

* It make easy for us to know the importance of water
* It make us healthy
* The fear of kidney stone will be less

SOFTWARE & HARDWARE USED

**SOFTWARE** **USED**

DEV C++, Turbo C

**HARDWARE USED**

**MINIMUM REQUIREMENT**

* Processor-intel core i5 11600KF 4.2ghz
* RAM -8GB
* Hard Disk - minimum space available 2-3 Gb(512g SSD used

**MAXIMUM REQUIREMENT**

* NO LIMIT.

**PROGRAMMING LANGUAGES USED**

* C language

**PLATFORM USED**

1. Windows 11

**M O D U L E S**

We have categorized the modules in the two modules

* **ADMIN**

The admin has the unique login and password which completely controls the working.

1. Login
2. Manage scheduling

* **USER**

1. Registration
2. Login
3. Setting reminders
4. Modifying reminders
5. Exit

**C O D I N G**

#include <iostream>

#include <Windows.h>

using namespace std;

int main()

{

const double WATER\_PER\_KG = 0.033; // Liters of water per kilogram of body weight

const int MINIMUM\_AGE = 12; // Minimum age considered for calculation

const int MINIMUM\_WEIGHT = 30; // Minimum weight considered for calculation

const int MINIMUM\_SLEEP\_HOURS = 6; // Minimum hours of sleep considered for calculation

// Variables

double weight, waterIntake;

int age, sleepHours;

// Input

cout << "Enter your weight in kilograms: ";

cin >> weight;

cout << "Enter your age: ";

cin >> age;

// cout << "Enter the number of hours you slept last night: ";

// cin >> sleepHours;

// Check if input meets minimum requirements

if (weight < MINIMUM\_WEIGHT || age < MINIMUM\_AGE)

{

cout << "Sorry, we cannot calculate water intake with the provided information." << endl;

return 1;

}

// Calculate water intake

waterIntake = weight \* WATER\_PER\_KG;

// Output

cout << "Your recommended water intake for today is: " << waterIntake << " liters." << endl;

int count = 1;

int interval;

int waterAmount;

cout << "Enter reminder interval (in seconds):";

cin >> interval;

// cout<<"Enter amount of water to drink (in milliliters):";

// cin>>waterAmount;

while (waterIntake > 0)

{

// Wait for the specified interval

Sleep(interval \* 1000);

// Display reminder message

cout << "Reminder: Drink " << waterIntake << " liters of water." << endl;

// Generate beep sound

Beep(750, 500);

// Decrement water intake

waterIntake -= WATER\_PER\_KG;

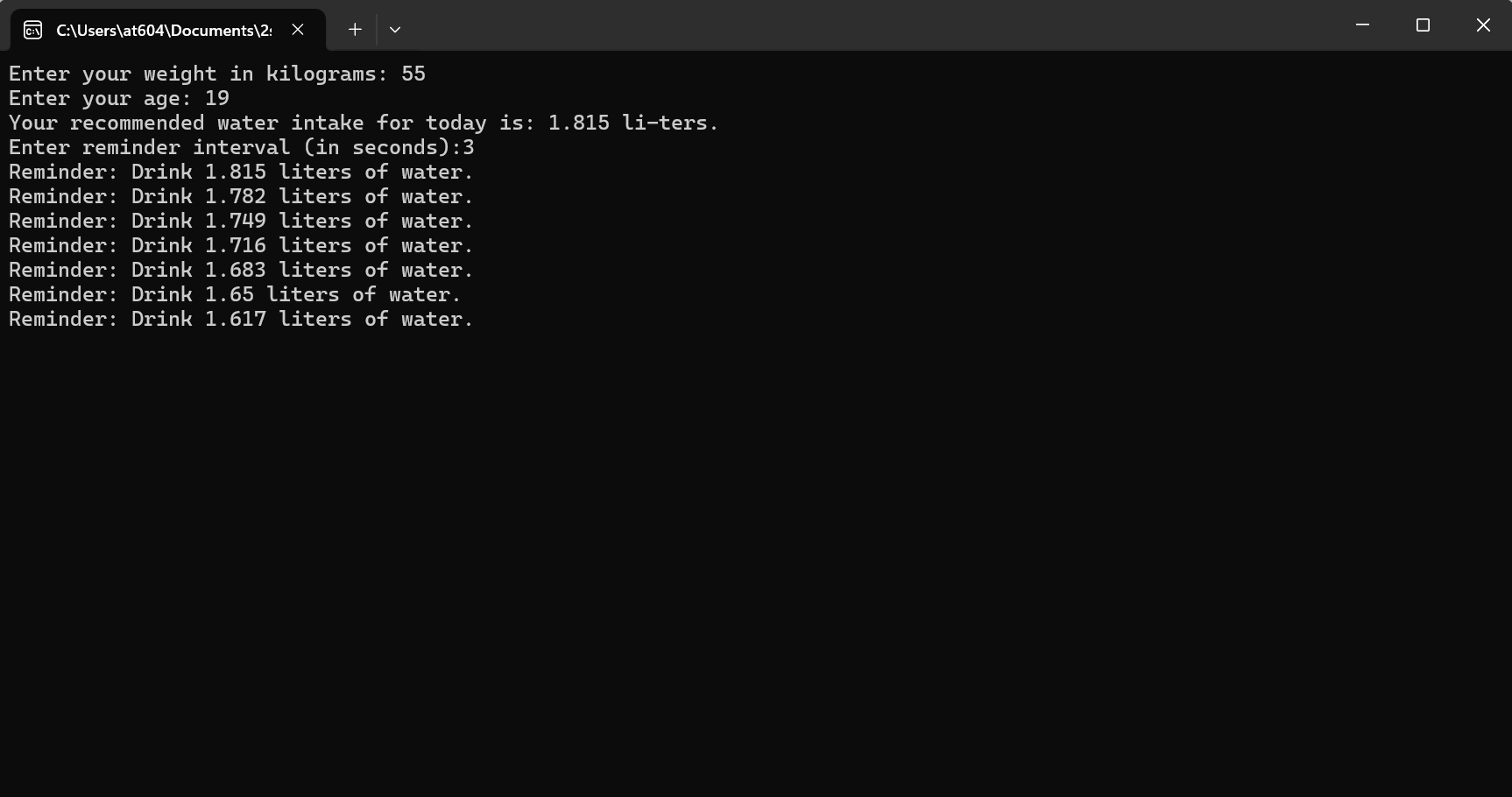
}

cout << "You've finished your daily water intake. Good job!" << endl;

return 0;

}

**O U T P U T**

****

SYSTEM DESIGN

System Analysis & Design

The way that is followed by carrying on with the development of application is as followed

Defining a Problem

Defining a problem is one of the important activities of the project. The objective is to define precisely the objective type problem to be solved and thereby determined the scope of the new system. The phase consists of two main tasks. The fast task within this activity is to review the organization needs to originally initiate the project. The second task is to identify, at an abstract or general level, the expected capability of the new system. Thus, it helps to define the goal to we achieved and the boundaries of your system. A clear understanding of the problem will help us in building a better system and reduce the risk of project failure. It also specifies the resources that have to be made available to the project. Three important factor project goals, project bounce and the resource limits are sometimes called the projects in term of reference.

**T E S T I N G**

Testing System testing is designed to uncover the weaknesses that were not found in earlier test. In the testing phase, the program is executed with the explicit intention of finding errors. This includes forced system failures and validation of the system, as its user in the operational environment will implement it. For this purpose, test cases are developed. When a new system replaces the old one, such as in the present case, the organization can extract data from the old system to test them on the new. Such data usually exist in sufficient volume to provide sample listings and they can create a realistic environment that ensures eventual system success. Regardless of the source of test data, the programmers and analyst will eventually conduct four different types of tests.

SECURITY TESTING OF THE PROJECT

Unit testing focuses verification efforts on the smallest unit of the software design, the module. This is also known as Module Testing. The modules are tested separately. This testing is carried out during programming stage itself.

### Validation Testing

Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably expected by the users. After validation test has been conducted one of the two possible conditions exists

1. The function or the performance characteristics confirm to specification and are accepted.
2. A deviation from specification is uncovered and a deficiency list is created.

### Output Testing

After performing the validation testing the next step is output testing of the proposed system since no system is useful if it does not produce the required output in the specific format. The outputs generated or displayed by the system under consideration are tested by asking the users about the formats required by them.

### User Acceptance Testing

User acceptance of a system is a key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system users at the time development and making changes whenever required.

### System verification and validation

System verification and validation is done to check the quality of the software in simulated and live environment. A number of different transactions are used to perform verification. Validation is the process of demonstrating that the implemented software does satisfy the system requirements. One aspect of software validation is to statistically analyze the program without resorting to actual execution. The system validation done in such-a-way that the system response time will not cause any hardship to the user.

### White Box Testing

White box testing is a test case design method that uses the control structure of the procedural design to derive test cases. Using white box testing methods, we can derive test cases that

* Guarantee that all independent paths within a module have been exercised at least once.
* Exercise all logical decisions on their true and false sides.
* Execute all loops at their boundaries and within their operational bounds.
* Exercise internal data structures to ensure their validity.

### Black Box Testing

Black box testing methods focus on the functional requirements if the software. That is, black box testing enables us to derive sets of input conditions that will fully exercise all functional requirements of the program. Black box testing attempts to find errors in following categories:

* Incorrect or missing functions
* Interface errors
* Performance errors
* Initialization and termination error.

**FUTURE SCOPE**

As now the project/software will work upon command prompt in future we will make GUI (graphical User Interface) or web-based software/system so that every user or employ can access it remotely.

In future we can add custom ringtones to make it more interactive. This software can be easily implemented under various situations. We can add new features as and when we require. It is reusable and flexible in all manners.

This software is extendable in ways that its original developers may not expect. We can also update as many times as we need.

Performance will be improved day to day users increment.

In future we can convert it in android application (APP).

### BIBLIOGRAPHY

* <https://www.javatpoint.com/c-preprocessor-include>
* <https://www.javatpoint.com/printf-scanf>
* <https://en.cppreference.com/w/c/language/main_function>
* <https://www.javatpoint.com/data-types-in-c>
* <https://www.javatpoint.com/variables-in-c>
* <https://www.javatpoint.com/keywords-in-c>
* <https://www.javatpoint.com/c-switch>
* <https://www.javatpoint.com/c-loop>
* <https://www.javatpoint.com/c-break>
* <https://www.javatpoint.com/c-continue>
* <https://www.javatpoint.com/c-if-else>
* <https://www.javatpoint.com/c-format-specifier>
* <https://www.javatpoint.com/functions-in-c>
* <https://www.javatpoint.com/c-strings>

**Flow Chart**

A flow chart is a diagram of the steps in a process and their sequences. Also known as a process flow diagram or process map.

THE POP UP WILL APPER WITH THE AMOUNT OF WATER AT EVERY GIVEN INTERVAL

SET THE TIME INTERVAL

START

USER