Project Report on

Menstrual Cycle Tracker in Java

SUBMITTED BY:

Ridima

CS-23411235

2nd Year [B.Tech CSE]

SUBMITTED TO:

Ms. Surabhi Purwar

DATE OF SUBMITTION

22nd November 2024

ABSTRACT

The Menstrual Cycle Tracker project is designed to help users track their menstrual cycle based on the first day of their last period and their average cycle length. This tool can predict the expected start date of future periods, assisting individuals in planning their schedules and health management.

The project uses the Java programming language to implement a console-based application that asks the user for input data and calculates future menstrual cycles.

The purpose of the application is to calculate the expected dates of menstruation, identify fertile windows, and compute average cycle lengths based on user inputs. This tool is particularly useful for individuals who wish to monitor their cycles for health, family planning, or general awareness.

TABLE OF CONTENT

Introduction

A menstrual cycle is an important aspect of a woman's health, and tracking it can help in understanding the regularity, spotting irregularities, and planning for family life or health-related activities. The average menstrual cycle lasts 28 days, though it can range from 21 to 35 days. This project aims to develop a *Menstrual Cycle Tracker* in Java that allows users to input the first day of their last period and their typical cycle length to predict upcoming periods.

The application's core functionality is based on simple date calculations using Java's LocalDate class. By inputting the date of their last period and average cycle length, users can predict their next period date and calculate their fertile window.

Some objectives of this project:

- To help users track their menstrual cycle.
- To predict the expected start of the next menstrual period based on user input.
- To provide insights into how long the next period might be based on average cycle lengths.

Problem Statement:

Accurate menstrual cycle tracking is essential for individuals who wish to monitor their health or plan for pregnancy.

This project addresses this issue by automating the prediction of future periods and fertile windows based on historical data, providing a more accurate, reliable, and user-friendly solution.

Current applications may require complex interfaces or not offer simple predictions based on basic input data. This project seeks to provide an accessible, simple, and effective solution by creating a Java-based application that predicts future periods based on user-provided data.

Literature Review

Several studies and applications have been developed to help track menstrual cycles. These solutions range from basic calendar-based tools to advanced mobile applications with more features such as tracking ovulation, symptoms, and mood. Some key findings from existing literature:

Cycle Prediction Algorithms:

Most modern tracking applications use algorithms based on a typical 28-day cycle, with variations based on user input. These systems help to predict not only the next period but also ovulation dates.

Technologies in Use:

Mobile apps such as Clue, Flo, and Period Tracker leverage user-friendly interfaces and track various health-related aspects. However, many of these apps use proprietary algorithms or are too complex for users seeking simple predictions.

Challenges:

There is an emphasis on the difficulty in building accurate predictions for users with irregular cycles, and while many apps do a good job for average cycles, they may not provide as accurate results for those with unusual patterns.

Methodology

Step-by-step approach for building the Menstrual Cycle Tracker in Java:

1. Requirement Analysis:

- The system should allow users to input the date of the first day of their last period and their average menstrual cycle length.

- It should then calculate and predict the next period date based on the input data.

2. Input Format:

- Date input in dd-mm-yyyy format.

- Cycle length input in days (typically 21 to 35 days).

3. System Design:

- The application will be developed using Java, utilizing the Scanner class for user input and Calendar and Simple Date Format classes for date manipulation and output.

4. Algorithm:

- The system will use a basic algorithm to predict the next period based on adding the average cycle length (in days) to the last period date.

5. Code Development:

- Code will be developed using basic Java constructs.

- The SimpleDateFormat class will be used for date parsing and formatting.

- The Calendar class will handle date arithmetic.

6. Testing:

- The system will be tested with several use cases, including normal cycles, shorter, and longer cycles.

7. Output:

- The system will output the expected start date of the next period.

8. Limitations:

- Assumes regular cycles (users with irregular cycles may find predictions less accurate).

- Only basic predictions based on last period and average cycle length.

Code Implementation

```
java

import java.util.Scanner;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Date;


public class MenstrualCycleTracker {


    //Method to calculate next period date

    public static String calculateNextPeriod(Date lastPeriodDate, int cycleLength) {

        Calendar calendar = Calendar.getInstance();

        calendar.setTime(lastPeriodDate);

        calendar.add(Calendar.DAY_OF_YEAR, cycleLength); // Adds the cycle length to the last period

        Date nextPeriodDate = calendar.getTime();


        SimpleDateFormat sdf = new SimpleDateFormat("dd-mm-yyyy");

        return sdf.format(nextPeriodDate);
```

```
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        //Ask for last period date and cycle length
        System.out.print("Enter the first day of your last period (dd-MM-
yyyy): ");
        String inputDate = scanner.nextLine();
        System.out.print("Enter your average menstrual cycle length in
days (e.g., 28): ");
        int cycleLength = scanner.nextInt();

        try {
            //Parse the input date into a Date object
            SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
            Date lastPeriodDate = sdf.parse(inputDate);

            //Calculate the next period date
            String nextPeriod = calculateNextPeriod(lastPeriodDate,
cycleLength);
```

```
            //Display the next period date
            System.out.println("Your next period is expected to start on: " + nextPeriod);
        } catch (Exception e) {
            System.out.println("Invalid date format. Please use dd-MM-yyyy.");
        } finally {
            scanner.close();
        }
    }
}
```

Results and Discussion

Sample Input:

```
- Last Period Date: 01-10-2024
- Cycle Length: 28 days
```

Sample Output:

```
Your next period is expected to start on: 29-10-2024
```

Discussion:

- The system accurately predicts the next period based on the input date and cycle length.

- It is important to note that the system assumes regular cycles. If a user's cycle is irregular, predictions may not be as accurate.

- This simple approach is effective for users with regular cycles who are looking for a straightforward tool.

Conclusion

The Menstrual Cycle Tracker provides a basic but effective solution for predicting menstrual cycles based on the first day of the last period and the average cycle length. While it is a useful tool for individuals with regular cycles, further improvements can be made to handle irregular cycles, track more health-related data (like ovulation), and add a graphical interface for better user experience.

Future work could include:

- Building a GUI using JavaFX or Swing for a more interactive experience.

- Implementing features to track symptoms, moods, or fertility windows.

- Enhancing the accuracy of predictions with data analytics or machine learning for irregular cycles.

References:

- "Understanding the Menstrual Cycle" – National Institute of Health (NIH).
- *Fertility and Sterility*, 2017. The Importance of Menstrual Cycle Tracking for Health Monitoring.
- Mobile applications for period tracking: Clue, Flo, and Period Tracker.