**D. Y. Patil College of College of Engineering and Technology, Kolhapur**

**Department of Computer Science & Engineering**

**Class: SY-A Subject: AOOC**

**Experiment no: 15**

**Group No. : 15 Mini Project**

**Title of Mini-Project: Alarm Clock For Heavy Sleeper**

**Problem Statement:**

Many heavy sleepers tend to miss or ignore traditional alarm clocks, leading to oversleeping and missed commitments. This project aims to develop an alarm clock application that requires users to solve a puzzle to stop the alarm, ensuring they are mentally alert and awake.

**Introduction:**

Waking up on time is a persistent challenge for heavy sleepers, and conventional alarm clocks often fall short. To address this, we propose a **Java Swing-based Alarm Application** specifically designed to ensure the user is fully awake before disabling the alarm. Unlike typical alarms that can be dismissed with a single tap, this application introduces a cognitive challenge—requiring the user to solve a randomly generated mathematical expression before the alarm can be silenced.

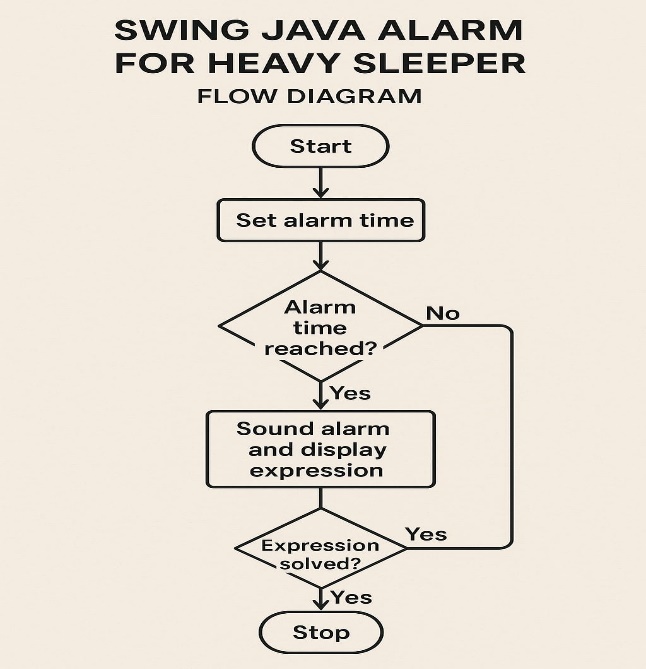
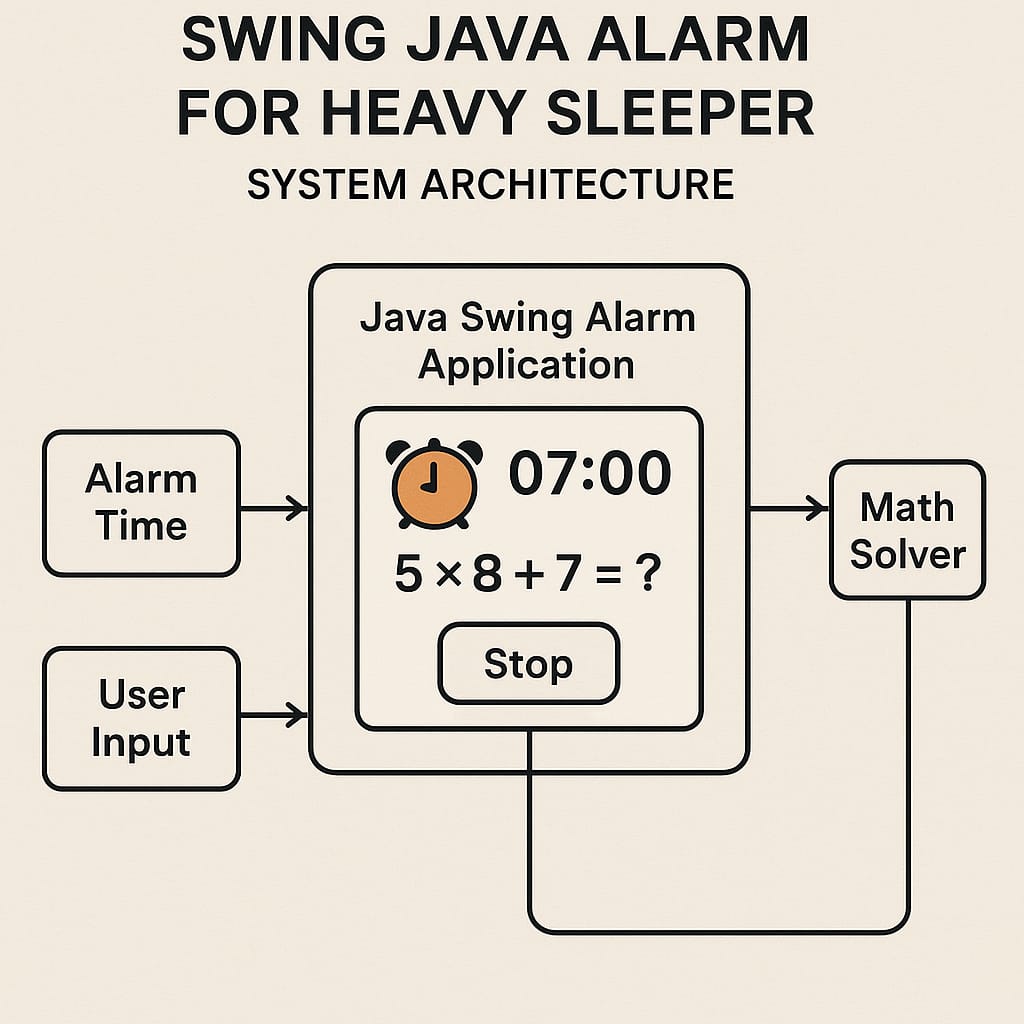
Built using the **Java Swing framework**, this desktop application combines a simple graphical user interface (GUI) with intelligent logic to make waking up more interactive and engaging. It targets users who tend to oversleep, snooze repeatedly, or ignore alarms altogether.

The alarm system operates based on a scheduled time and, upon activation, triggers an audible alarm along with a math problem display. The alarm continues ringing until the user correctly solves the problem. This ensures that the user is mentally alert before the alarm stops, thereby reducing the chances of falling back asleep.

**Key Features:**

* **User-Friendly GUI:** Intuitive interface for setting alarms and viewing the countdown.
* **Persistent Alarm Ringing:** Alarm will not stop until the correct answer is entered.
* **Dynamic Expression Generator:** Random math problems (e.g., multiplication, addition, order of operations) generated each time.

**System Architecture:**

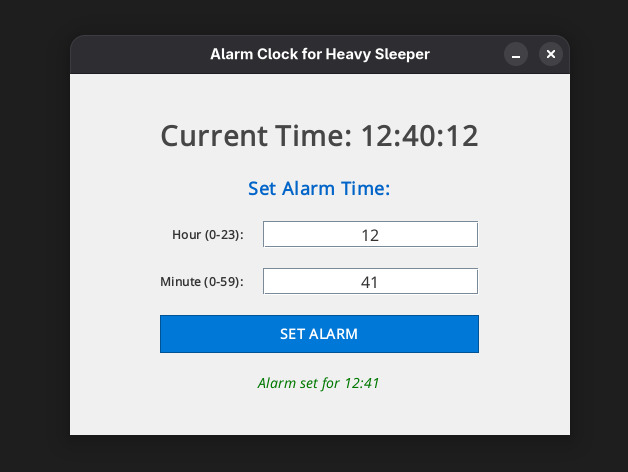
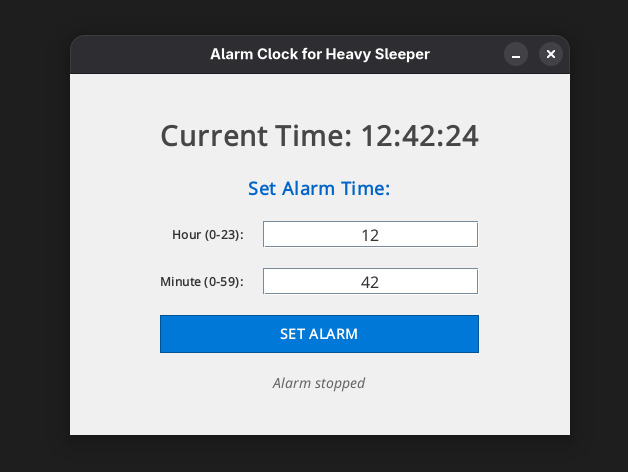
** **

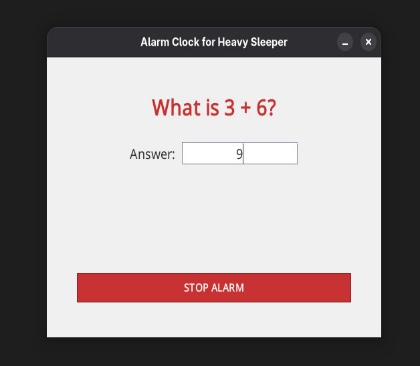
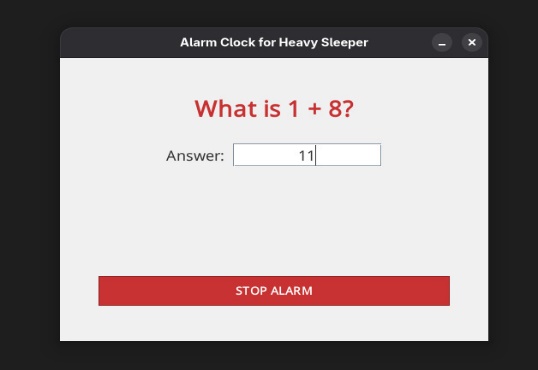
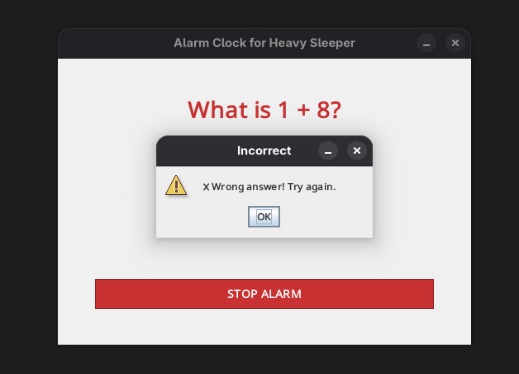


**Module description or working of system:**

* AlarmPanel (Alarm Setting Module):  
  Allows the user to input and set an alarm time. It displays the current system time in real-time and shows the set alarm time.
* AlarmController (Logic and Alarm Control Module):  
  Manages core functionalities such as setting the alarm, checking the current time, triggering the alarm, and switching between panels. It also handles sound playback and puzzle generation.
* PuzzlePanel (Puzzle Verification Module):  
  When the alarm goes off, this panel displays a randomly generated math puzzle (like "What is 7 + 3?"). The user must input the correct answer to stop the alarm sound.
* MainPanel (Navigation Controller):  
  Acts as a container using CardLayout to switch between the AlarmPanel and PuzzlePanel based on the state of the alarm.

**Screenshots:**

** **

****

**Group Members:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Unique id** | **Roll No** | **Name of Student** | **Sign** |
| En23219227 | 62 | Harsh Pravin Chandankhede |  |
| En23159346 | 64 | Sumedh Rupeshkumar kamble |  |
| En23193312 | 65 | Prathmesh Rajaram Gosavi |  |
| En23147793 | 67 | Mandar Balasaheb Patil |  |
| En23178287 | 71 | Aditya Bhimrao Kamble |  |