Smart Health Monitoring System - Java Code

Team: Java Tech

- Jatin Gaur
- Devranjan Pradhan
- Priyam Shrivastava
- Piyush Kumar Sharma

SerialReader.java

```
import com.fazecast.jSerialComm.SerialPort;
import java.util.Scanner;

public class SerialReader {
    public static void main(String[] args) {
        SerialPort comPort = SerialPort.getCommPorts()[0];
        comPort.openPort();
        Scanner scanner = new Scanner(comPort.getInputStream());
        while(scanner.hasNextLine()) {
            try {
                String line = scanner.nextLine();
                System.out.println("Received: " + line);
            } catch(Exception e) { e.printStackTrace(); }
        }
        comPort.closePort();
    }
}
```

HealthDashboard.java

```
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Label;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;

public class HealthDashboard extends Application {
    @Override
    public void start(Stage primaryStage) {
        Label heartRateLabel = new Label("Heart Rate: -- BPM");
        Label tempLabel = new Label("Body Temp: -- °C");
        VBox layout = new VBox(10, heartRateLabel, tempLabel);
        Scene scene = new Scene(layout, 300, 200);
        primaryStage.setTitle("Smart Health Monitor");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

DatabaseManager.java

```
import java.sql.*;

public class DatabaseManager {
    private Connection conn;

    public void connect() {
        try {
            conn = DriverManager.getConnection("jdbc:sqlite:health.db");
            System.out.println("Connected to DB.");
        } catch(SQLException e) {
            e.printStackTrace();
        }
    }

    public void insertVital(int userId, int heartRate, double temp) {
        try {
              String sql = "INSERT INTO vitals(user_id, heart_rate, temperature, timestamp) VALUES (?, ?, ?,
CURRENT_TIMESTAMP)";
            PreparedStatement stmt = conn.prepareStatement(sql);
            stmt.setInt(1, userId);
            stmt.setInt(2, heartRate);
            stmt.setDouble(3, temp);
            stmt.executeUpdate();
        } catch(SQLException e) {
            e.printStackTrace();
        }
    }
}
```

AuthManager.java

```
import org.mindrot.jbcrypt.BCrypt;

public class AuthManager {
    public static String hashPassword(String password) {
        return BCrypt.hashpw(password, BCrypt.gensalt());
    }

    public static boolean checkPassword(String password, String hashed) {
        return BCrypt.checkpw(password, hashed);
    }
}
```