SMART POOL MAINTENANCE AND SOLAR MONITORING PROJECT

GROUP ANAS

SUN POOL MANAGER

SUN POOL MANAGER

INTRODUCTION

Sun Pool Manager is an intelligent, IoT-driven solution designed to meticulously manage and monitor swimming pool parameters and solar power utilization. This system employs the Blynk platform to automate pool maintenance tasks by continuously tracking pH levels, water temperature, and water levels, alongside enhancing security with motion detection. It also monitors solar energy production, harnessing solar power for sustainable pool management.

OBJECTIVE

To maintain ideal swimming pool conditions, ensure perimeter security, and optimize solar energy usage. The system automates pool maintenance processes, allowing for remote monitoring and management through the Blynk application.

SCOPE

- **Design and Development**: Develop an IoT infrastructure tailored for pool monitoring, including sensor integration for pH, temperature, and water levels, as well as motion detection and solar energy assessment.
- **Blynk Integration**: Seamlessly connect the system to the Blynk platform for easy access to monitoring and control features.
- **Data Analysis**: Analyze sensor data to provide actionable insights on pool conditions and solar panel output.
- **Testing and Validation**: Conduct extensive testing to validate the reliability and precision of the system.
- **Deployment**: Implement the Sun Pool Manager system and instruct users on interacting with the Blynk app interface.

METHODOLOGY

- **Hardware Setup**: Deploy microcontrollers with a suite of sensors and actuators to regulate pool conditions and solar panel operations.
- **Software and Analytics**: Craft software solutions for real-time data processing and Blynk integration, utilizing analytics for solar output optimization.
- **Blynk Interface**: Design a customized Blynk dashboard for comprehensive monitoring and control of the pool and solar systems.

TECHNOLOGY STACK

- **Microcontrollers**: Arduino, ESP32
- **Sensors**: pH sensor, DS18B20 Temperature Sensor, Water Level Sensor, PIR Motion Sensor, ACS712 Current Sensor
- **Actuators**: Relay modules
- **IoT Platform**: Blynk
- **Programming**: Arduino IDE, sensor and Blynk libraries in C/C++

EXPECTED OUTCOMES

- An automated and integrated pool monitoring system.
- Remote real-time control and updates via the Blynk app.
- Enhanced operational efficiency and solar energy utilization.
- Improved safety with motion-sensing capabilities.

BUDGET ESTIMATE COMPONENTS AND PRICES

Component Function	Component Name	Component Price	Quantity	Resistor Needed
Microcontroller	Arduino Mega or ESP32	~16 JOD	1	N/A
pH Sensor for Pool Monitoring	pH Sensor Module for Arduino	~20 JOD	1	N/A
Temperature Monitoring	DS18B20 Temperature Sensor for Arduino	~4 JOD	1	4.7kΩ each
Water Level Monitoring	Water Level Sensor for Arduino	~2 JOD	3	N/A
Water Pumping	12V Water Pump	~21 JOD	1	N/A
Wi-Fi Connectivity (if using Arduino Uno)	ESP8266 Wi-Fi Module	~12 JOD	1	N/A
Circuit Prototyping	Breadboard for Arduino	~2 JOD	1	N/A
Circuit Connections	Jumper Wires (Assorted)	~3 JOD for a set	2	N/A
Relay for Pumps	Relay Module for Arduino	~6 JOD	1	N/A

CONNECTIVITY GUIDE

Microcontroller (Arduino Mega or ESP32):

- Power via a 5V USB power supply or through the DC-DC converter connected to your solar power setup.
- Ground (GND) to a common ground bus on your breadboard or distribution board.

Wi-Fi Module (ESP8266, if using Arduino Uno):

- VCC to 3.3V on the Arduino.
- Ground (GND) to common ground.
- TX to RX and RX to TX for serial communication with the Arduino.

pH Sensor Module for Arduino:

- VCC to 5V on the Arduino.
- Ground (GND) to common ground.
- Analog signal output to an analog input pin on the Arduino (e.g., A0).

DS18B20 Temperature Sensor for Arduino:

- VCC to 5V on the Arduino.
- Ground (GND) to common ground.
- Data line to a digital pin on the Arduino (e.g., D2).
- A 4.7kΩ pull-up resistor between VCC and the data line.

Water Level Sensor for Arduino:

- VCC to 5V on the Arduino.
- Ground (GND) to common ground.
- Analog signal output to another analog input pin on the Arduino (e.g., A1).
-
-

12V Water Pump:

- Power directly from the 12V solar power supply or battery through a relay.
- Relay module's control pin connected to a digital pin on the Arduino (e.g., D4).
- Ground of the relay module to common ground.

Breadboard:

- Used for prototyping the circuit, allowing for easy adjustments and testing.

Jumper Wires (Assorted):

- Used to make all the connections on the breadboard and between the components and the microcontroller.

Prototype structure
Figure