Here's your fully updated project, integrating Wi-Fi connectivity with your smartphone to replace the GSM module and LCD for customer management. This version ensures seamless interaction via a web-based or mobile application, making the system more flexible and scalable.

# **Smart Water Meter with Wi-Fi-Based Customer Management System**

## **Project Overview**

Water is a crucial resource that requires efficient management to reduce waste and optimize consumption. This Smart Water Meter system integrates RFID-based customer identification with Wi-Fi-enabled real-time data monitoring, replacing GSM and LCD with smartphone interaction. The updated version enhances flexibility by allowing customers to register, check balances, and receive notifications via a web app or mobile platform.

## **Features and Workflow**

### ****1. RFID-Based Customer Identification****

* Each customer has an RFID card containing a unique ID.
* Before using the tap, the customer taps their card on the RFID reader to log in.
* The system records the RFID tag and begins tracking their water usage.

### ****2. Wi-Fi-Based Data Management****

* The ESP32 connects to a local Wi-Fi network, allowing smartphones to access the system.
* Customers interact with the system via a web or mobile app to view usage details and manage their accounts.
* Secure cloud or local database storage ensures data integrity and accessibility.

### ****3. Water Flow Monitoring****

* A flow sensor measures water consumption in real-time.
* The ESP32 processes the data and links it to the respective RFID ID for accurate billing.

### ****4. User Notifications and Alerts****

* The system sends alerts via the web or mobile app instead of SMS.
* Notifications include leak detection, low balance warnings, and usage limits.

### ****5. Automated Water Flow Control****

* A solenoid valve regulates water access based on RFID verification.
* If a customer has available balance, the valve opens; otherwise, access is restricted.

## **Key Components**

### ****1. Hardware****

* **RFID Module (MFRC522)** – Reads customer IDs from their RFID cards.
* **Flow Sensor** – Measures water consumption.
* **ESP32 Microcontroller** – Processes data and manages Wi-Fi communication.
* **Solenoid Valve** – Controls water access based on authentication.
* **LED Indicators** – Show transaction success or warnings.
* **Power Supply** – Battery or mains with backup for reliability.

### ****2. Software****

* **Wi-Fi-Based Web Interface** – Mobile-friendly dashboard for user interaction.
* **RFID Authentication Logic** – Customer identification and data logging.
* **Flow Sensor Data Processing** – Real-time tracking and billing.
* **Database Storage** – Structured logs for consumption and user details.
* **API & Webhooks** – Communication between ESP32 and smartphone for updates.

## **Benefits of the System**

1. **Real-Time Access** – Users check their balance and consumption anywhere via Wi-Fi.
2. **Cost Optimization** – Leak detection helps reduce unnecessary costs.
3. **Improved Security** – Encrypted Wi-Fi-based authentication prevents unauthorized access.
4. **Scalability** – Easily expandable for more users or advanced features.
5. **User-Friendly Interaction** – Eliminates SMS dependency with a modern UI.

## **Challenges & Solutions**

### ****1. Network Dependence****

* **Challenge:** Requires stable Wi-Fi for operation.
* **Solution:** Local fallback storage ensures functionality even during outages.

### ****2. Sensor Accuracy****

* **Challenge:** Precise water measurement.
* **Solution:** Regular calibration and error handling algorithms.

### ****3. Power Management****

* **Challenge:** System reliability during power failures.
* **Solution:** Battery backup and energy-efficient components.

### ****4. Mobile App Integration****

* **Challenge:** Ensuring seamless app functionality.
* **Solution:** Optimized UI with API-based interactions for quick data retrieval.

## **Future Enhancements**

* **IoT Expansion** – Full cloud connectivity for remote monitoring.
* **Prepaid Billing System** – Users pay in advance for water access.
* **AI-Based Leak Detection** – Machine-learning algorithms for efficient problem identification.
* **Solar Power Integration** – Sustainable energy source for long-term operation.

## **System Design for Web & Wi-Fi-Based Interface**

### ****1. Displaying User Info on Mobile UI****

Instead of an LCD, the smartphone interface will show:

* Customer ID and transaction status.
* Real-time water consumption (in liters).
* Alerts for leaks and low balance.
* Payment history and updates.

#### ****Implementation:****

* **Mobile Web Dashboard (React.js or Flask-based server)** for data access.
* **REST API to ESP32** for live data updates.
* **Cloud or Local Database (MySQL, Firebase)** for storage.

### ****2. Wi-Fi-Based User Authentication & Billing Updates****

* Users check balance and update their accounts via a web app instead of GSM SMS.
* Admins can credit water balance remotely via their phone.

#### ****Implementation:****

* **User sends request via the mobile app**
  + Example query: “Check Balance”
  + System responds with usage details.
* **Admin updates balance remotely via the app**
  + Example request: “Add 50L to Customer 1234”
  + System updates balance and notifies the user.

### ****3. RFID-Based Water Control & Smart Restrictions****

* Customers authenticate via RFID cards before water access.
* If balance exists, the solenoid valve opens; otherwise, the system denies access.
* Visual and app-based notifications inform users of their status.

### ****4. Data Storage Format for ESP32****

To store user data efficiently, structured logs will include:

* **Customer ID**
* **Water usage (liters consumed)**
* **Remaining balance**
* **Last update timestamp**

#### ****Storage Format Options:****

* **JSON** – Ideal for web-based processing.
* **CSV** – Compatible with external applications.
* **Cloud Database** – Real-time tracking and retrieval.

## **Updated ESP32 Wiring Layout**

### ****1️⃣ Power & Ground Connections****

| **ESP32 Pin** | **Function** | **Connected Device** |
| --- | --- | --- |
| 3V3 | Power | ESP32, RFID, Sensors |
| GND | Ground | Common Ground |
| VIN | External Power | Power Supply |

✅ Stable power distribution to components

### ****2️⃣ RFID Module (SPI Connection)****

| **ESP32 Pin** | **Function** | **Connected Device** |
| --- | --- | --- |
| D5 | SPI SS (SDA) | RFID Module |
| D18 | SPI Clock(SCK) | RFID Module |
| D23 | SPI MOSI | RFID Module |
| D19 | SPI MISO | RFID Module |
| D4 | RFID Reset | RFID Module |
| D15 | IRQ (Interrupt Request) | RFID Module |

✅ **Notes:**

* **SDA** on your RFID module functions as **SS (Chip Select)** in SPI mode.
* **IRQ** is optional and used for interrupts if needed.
* **RST** must be connected to ensure proper initialization of the RFID module.
* **Ensure stable 3.3V power**, as the RFID module typically **does not tolerate 5V directly**.

✅ Correct SPI wiring for RFID authentication

### ****3. 📌 LED Pinout****

| **ESP32 Pin** | **Connected LED** | **Function** |
| --- | --- | --- |
| D32 | Blue LED (Power) | System is running |
| D27 | White LED (Leak) | Blinks rapidly for leaks |
| D33 | Red LED (Rejected) | RFID authentication failed |
| D25 | Yellow LED (Water Flow) | Water is actively running |
| D26 | Green LED (Accepted) | RFID authentication successful |

✅ **LED indicators provide clear feedback for customers**

### ****4. 📌 Solenoid Valve (Relay Control)****

| **ESP32 Pin** | **Function** | **Connected Device** |
| --- | --- | --- |
| D16 | Relay Control | Solenoid Valve |

✅ **Relay ensures controlled activation of the solenoid valve for water flow**

### ****5. 📌 Water Flow Sensor****

| **ESP32 Pin** | **Function** | **Connected Device** |
| --- | --- | --- |
| D14 | Pulse Signal | Water Flow Sensor |

✅ **Monitors real-time water flow through pulse detection**

🚀 **This setup keeps all functions properly assigned without conflicts!** Let me know if you want any further adjustments. 🔧🔥