**CLIENT REQIREMENT FOR SMART METER DESIGN**

1. GPS  
  2. Remote ON/OFF  
  3. Diagnostic Events - Meter tampering msg to be sent   
  4. Security / Authentication  
  5. Meter Parameters  
  6. MQTT Bridge  
  7. Encryption / Decryption  
  8. Import and Export of energy to grid  
  9. voltage control for smart meter  
  10. Refresh data everyday of 45 days back  
  11. Scheduler report to be sent every  
  12. Optical communication / manual communication or fetching of data  
  13. MQTT channel RF Module. communication module

**1. GPS**

Start → Power On → Initialize GPS Module → Acquire Satellite Signals

↓ (success?)

   → Yes → Fetch Latitude/Longitude → Store in NVM + Send via MQTT → End

  ↓ (Failure)

   → No → Retry after Timeout → End

**2.Remote ON/OFF**

Start → MQTT Subscribe(control ON/OFF) →wait for command

  ↓ (command Received)

→ yes → validate security → switch relay ON/OFF→ACK via MQTT→end

↓(command not received )

→No → Loop listening

**3.Diagnostic Events - Meter tampering msg to be sent**

Start →Monitor tamper sensors (magnetic, cover open, reverse current)→ wait for command

↓(tamper detected)

Yes →log event in memory→encrypt event →send MQTT alert→end

No→continue monitoring

**4. Security / Authentication**

Start → client connection request →receive authentication APDU

↓verify credentials (HLS challenge-response)

Pass → establish secure session → proceed to data exchange

Fail →reject connection → log event

**5. Meter parameters**

Start → client request (get/set parameters)

↓ decode DLMS/COSEM request

↓identify parameter (voltage, Current, power,etc )

↓fetch value or update settings

↓encrypt response

↓send MQTT response → end

**6. MQTT Bridge (client connect to MQTT broker)**

start → initialize MQTT client → connect to broker

↓ subscribe to control topics

↓publish meter data →Encryption data → bridge to cloud/other modules

↓(connection lost?)

Yes → reconnect procedure→retry

no→continue data flow

**7. Encryption / Decryption**

Start→ (data ready for transmission )

↓apply AES/HMAC Encryption

↓send via MQTT

↓on receive → decrypt data →verify integrity

↓pass/fail → process or discard

**8.Import and Export of energy to grid**

Start → measure energy flow direction

↓(import?)

Yes → add to import register

↓(export?)

Yes → add to export register

↓log data → send periodic report via MQTT → end

**9. Voltage control for smart meter**

Start → measure voltage

↓ compare with thresholds

→ if over voltage → log event +alert

→if under voltage → log event + alert

→Else normal operation

10. Refresh data (45 days back)

Start → scheduler (daily trigger)

↓fetch historical data (upto 45 days back)

↓compile report

↓encrypt & send via MQTT

↓if new entry

↓refresh /delete the data from 45 days (oldest data)

↓else keep as it is

↓end

**11.Scheduler Report**

Start → scheduler time interrupt

↓collect meter data snapshot

↓format report (XML/JSON)

↓Encrypt

↓send MQTT Report to server

**12.Optical /Manual communication**

Start → detect optical probe connected

↓handshake with probe

↓request /response data (manual fetch)

↓provide parameters to technician

↓ end

**13. MQTT channel RF Module. communication module**

Start → Initialize RF Module → Establish RF Link

↓configure as MQTT transport layer

↓subscribe + publish to topics

↓forward data between smart meter → broker

↓(RF link lost?)

Yes → retry /reinitialize

no→ continue operation