Meter Data Management

SOFTWARE REQUIREMENTS

MARCH 29TH 2023

Rev 0.3

Prepared By: Shmuel Branover

1 Table of Contents

2 Overview

This document Details the requirements for Satec's MDM software.

This software application is meant to complement Satec's other software offerings, namely PAS and ExpertPower. The main purpose of the software is to continuously poll Satec's Meters storing the data in database allowing the user an easy interface to view the meter data, statistics, and trends.

The ideal software candidate would be branded as a Satec product allowing for Satec to compete in tenders requiring a single solution provider.

While this software can be used for billing purposes by periodically polling energy registers it is not intended for that application and therefore there are no requirements to support advanced billing capabilities such as TOU and bill generation.

While this document may contain images of software screens this is meant for demonstration purposes and has no bearing on the actual application user interface.

3 Initial Setup

3.1 Install Process

The software should not require a technician for installation and the installation process should be straight forward using a single executable file that installs all the required dependencies. This should be done using the standard installation method common on most software packages where the user is showed a series of prompts where he can change some core configurations and click next, at the end of all the prompts the software should be installed.

It is expected that the software will be installed on a recent version of the Windows operating system. A nice to have feature would be to have a Linux version as well. Ideally the software should not require operating system admin privileges to be installed and it should not require those privileges to run.

3.2 Site Setup

A site is the group of all devices being monitored and a site tree is the hierarchical list of all devices being monitored with some devices grouped together.

When setting up the site adding devices should be straight forward allowing the user to add either devices or device groups, with a simple way to configure the communication for each device.

For each group of devices the parent can either be free standing or assigned a device, an example where this is useful is when monitoring the input to a breaker box along with the values on each breaker.

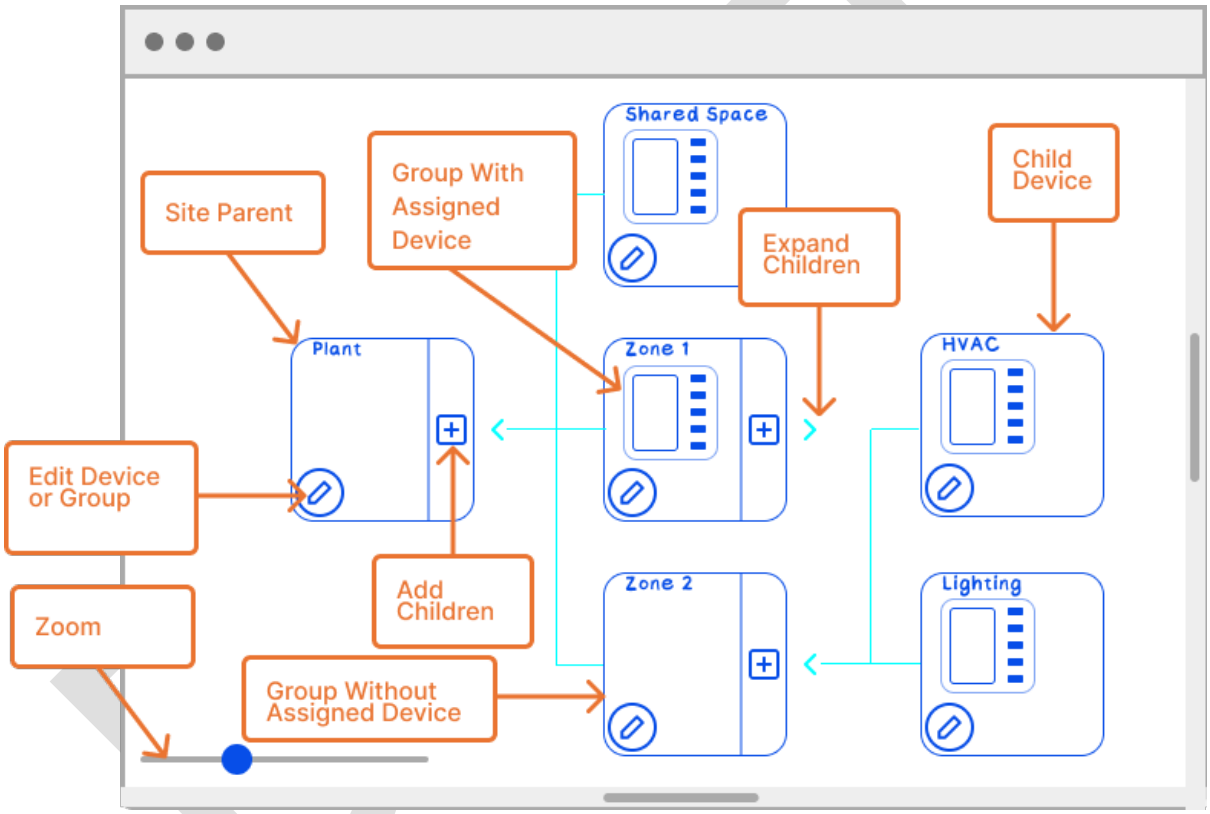

Figure 1 - Sketch of example site tree

Each device should be defined by a human readable, user configurable device name. On top of that that the user can assign labels to a group or a device allowing foe easy filtering.

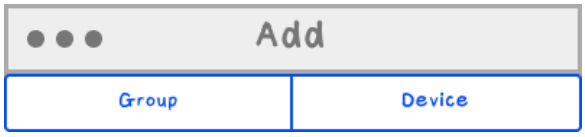

Figure 2 - Add Child Dialog

3.2.1 Predefined Modbus Maps

The software should be provided to the end user with predefined Modbus register maps for all Satec devices allowing for Satec devices to be added to the site tree with minimal configuration.

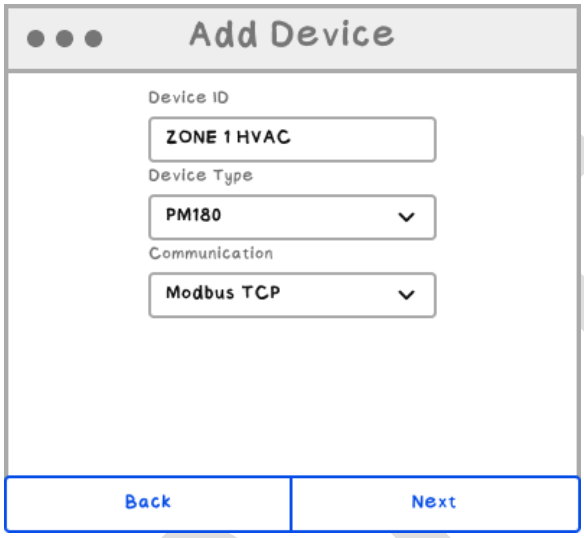

Figure 3 - Add Device Dialog

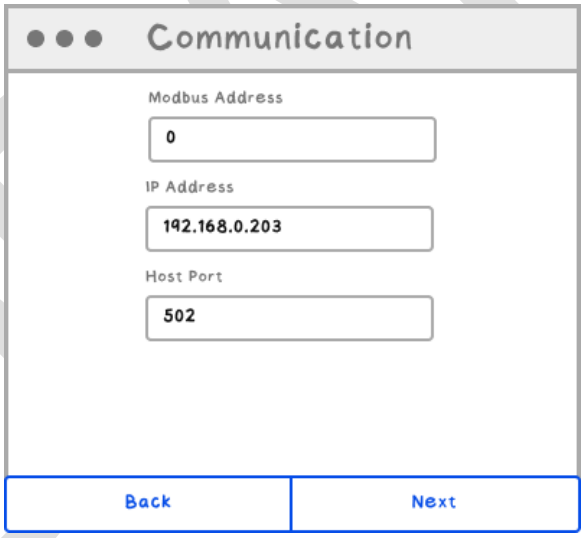

Figure 4 - Communication Setup Dialog

As Satec continuously develops new devices, the software should have the capability of easily adding new devices to future updates without much backend development effort.

3.2.2 Generic Devices

The software should also allow for Modbus maps of generic devices to be defined and added to the device tree. The software should allow the user to save the custom configurations so that the user can reuse the map when adding similar devices to the site.

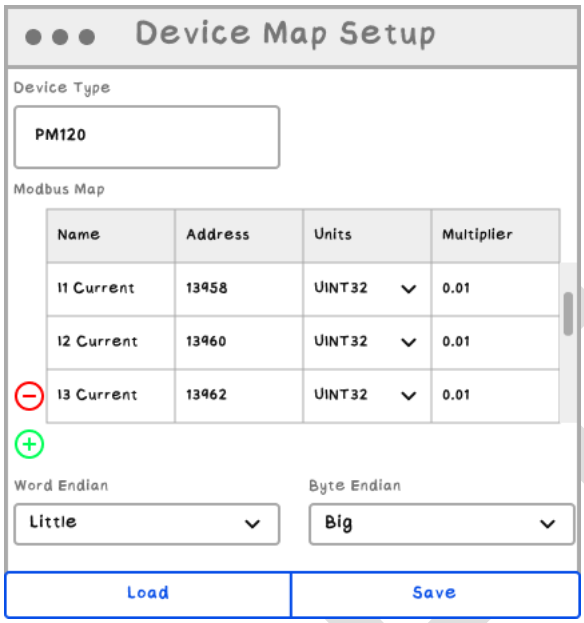

Figure 5 - Modbus Map Setup

The user should have the ability to use an existing device's Modbus map as a starting point to modify the existing map and then save it with a different name. This can also be used to add generic Modbus addresses to the existing device for expanded monitoring, for example monitoring IOs on the device.

3.2.3 Thresholds

For each monitored value the user should be able to define the minimum and maximum value this can be used have a scaled range. This should allow for devices with different ranges to be compared to each other.

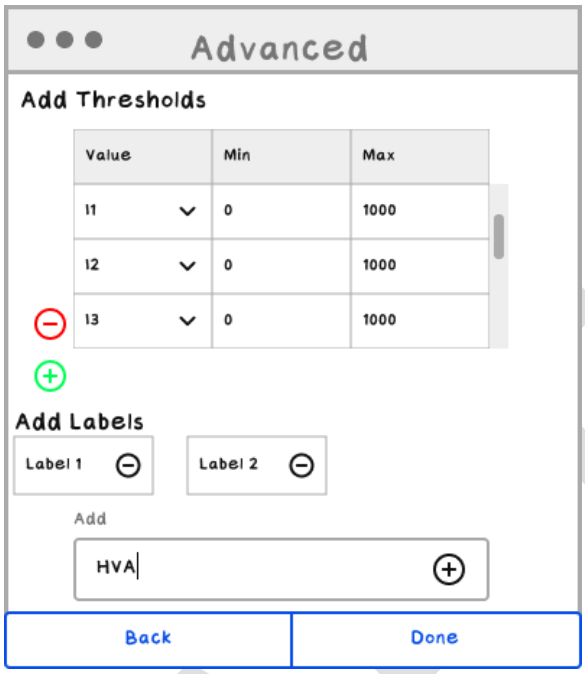

Figure 6 – Threshold and Label Setup Dialog

3.3 Communication Capabilities

The software should communicate with the devices using Modbus either over TCP/IP or Serially over COM ports.

The software should analyze the communication setup and notify the user of any contradictions. If devices that share the same COM port have different baud rates the software should notify the user of this contradiction but should **not** block the configuration.

Ideally the software would also support communication using ExpertPower Client.

3.3.1 Modbus Addressing over TCP/IP

When using TCP/IP Modbus addressing should still be supported, allowing for multiple slaves to share the same IP address. In the cases where multiple slaves do share the same IP address and port the software should not have multiple sockets opened simultaneously but rather it should either a) read all devices with addresses that associated with a single IP using a single opened socket or b) close the socket after reading the values.

In general, it is important that the software closes sockets whenever they aren't in use as some of the devices are limited with the number of simultaneous open sockets.

3.3.2 Satec ETCs

Some devices will be wired through our ETC (see links below). The software should allow for a configuration that uses such a device.

ETC-II

ETC ONE

3.3.3 Other Protocols

While not necessary, adding the ability to communicate using different protocols such as IEC 61850, DNP 3, BACnet, DLMS would be a nice to have additional feature.

4 Polling

At its core the software continuously polls and stores values from multiple devices.

4.1 Scheduling

The software should allow the user to customize the polling frequency of the devices. While not strictly necessary the ability to modify the schedule per device and per value would be a nice to have feature.

Aside from polling the measured values a separate polling schedule will be needed for datalogs and device configuration values as these are generally read less.

To make sure the storage requirements are deterministic the user should have the ability to configure the amount of time the data is retained, where the oldest polled values are deleted. The software should also display to the user the expected storage requirements for the specific configuration. The user should have the option to define the max storage size and the software should estimate the total log time for that configuration.

4.2 Polled Values

The primary function of the software is to poll and store the metered data as such the Modbus map for the following values should be included for all Satec devices:

4.2.1 Basic Readings

Voltage – Per Phase

Current – Per Phase (Including I4 and Neutral where applicable)

THD Voltage and Current – Per Phase

Active Power, Reactive Power, Apparent Power, and Power Factor – Per Phase and total

Frequency

Present Volt Demand – Per Phase

Present Ampere Demand – Per Phase (Including I4 where applicable)

kW, kvar, kVA demands – Total

4.2.2 Energies

On devices that support phase energies those should be logged as well

kWh and kvarh – import

kWh and kvarh - export

kWh and kvarh - net

kWh, kvarh and kVAh - total

4.2.3 Harmonics

Fundamental Magnitude Values for - V1, V2, V3, V4, I1, I2, I3, I4, kW L1, kW L2, kW L3, kvar L1, kvar L2, kvar L3, kVA L1, kVA L2, kVA L3, PF L1, PF L2, PF L3

Fundamental Phase Angles for - V1, V2, V3, V4, I1, I2, I3, I4s

*Unless otherwise specified all values are 1 second average values.

4.2.4 Future Modifications

As market demands evolve, the software should be implemented in a way that would allow for the Modbus maps to be easily updated.

4.3 Data Log Retrieval

Satec devices implement a proprietary method for storing datalogs. The retrieval of datalogs is done over Modbus using a block of addresses for requests from the software to the device and a block of addresses for the response. Using these blocks there is a packet structure that can be used to retrieve the data.

Each metered value has a unique ID called a point ID used to identify the value in the datalog, the software should support a subset of the point ID's relevant to the application and display an error when encountering an unknown point ID.

The datalogs are necessary for the following situations 1) when there are communication errors to fill in missing data 2) some values need accurate time stamped values 3) for application specific logs.

4.3.1 Communication Errors

In the event of a lapse in communication the software should be able to fill in the missing data (if available) from the data log.

4.3.2 Timestamped Values

In some cases, accurate time stamped values are necessary, as examples single cycle min/max, maximum demands, and energy where regulation requires that. In those cases, the

communication latency might be problematic especially if the polling interval is not often, for those cases it is important to be able to pull time stamped data from the device.

4.3.3 Application Specific Datalogs (Optional Module)

Some devices have application specific datalogs, the software should periodically poll these logs and generate reports. The following features are required:

- Power Quality (PQ) Event Log
- Power Quality Compliance Reports, Statistics Report, and Harmonics Reports supporting both IEC50160 and IEEE1159
- Ability to export PQ reports in industry-standard file formats such as PQDIF and Comtrade
- Event Log
- Sequence of events (SOE) log
- Fault (DFR) Log with distance to fault calculations

4.3.4 Waveforms

On devices that support waveforms all waveform logs should be polled and stored. The retrieval mechanism is like other datalogs, but each record contains a waveform instead of individual logged values.

Some waveforms correlate to events in PQ, SOE, or DFR logs, in which case a method of linking from the specific event in the respective record to the corresponding waveform needs to be added to the log viewer.

4.4 Configuration Information

As mentioned above aside from polling the devices for measured values the software should also poll the device for its configuration values. This will allow for validation that the configuration was not modified.

4.4.1 Device Information

The following device information should be logged periodically.

Device model ID – This should be compared to the configured device and a warning should be indicated if the values don't match.

Device Serial Number – A notification should be sent if this value has changes.

Device Firmware version and build number.

4.4.2 Device Basic Setup

The following configurations should be logged allowing for future validation of the data.

Wiring mode, PT ratio, CT primary current, Nominal line frequency, Phase order, Power demand period, Number of demand periods in a sliding window, Power calculation mode.

For users with elevated privileges there should be the ability to modify basic setup. The software should have the ability to enter the device passwords to modify these settings. The software should store the device password in a secure manner and only allow users with elevated privileges to make use of this.

4.4.2.1 RTC Synchronization

Since each device has a clock RTC which has some drift, the software should poll the RTC and store the drift giving the user the ability to view timestamps on the server.

The software should offer the ability to periodically update the on device time. This should make use of the time zone registers on the device.

4.4.3 Device Faults

The software should also monitor the device for faults and diagnostic flags alerting the user in the event of a fault.

5 Viewing Data

5.1 Grouping and Hierarchy

When viewing the data, the device should be displayed in a hierarchical manner, where children devices can be minimized or expanded at the parent node in the tree.

The software should allow filtering by labels where devices with the selected labels can be hidden or displayed. For example, if the user only wants to view meters monitoring the HVAC system or if he wishes to view only meters on the third floor in building 6.

The user should have the option to save the different display configurations to a file allowing the user to reuse the view later.

5.2 Graphical View

The default view should be a graphical dashboard of the entire site with key parameter displayed for each device or group.

The views should be user modifiable allowing for dashboard creating in a user friendly manner.

5.2.1 Site View

The user should have the ability to add small widgets that can be displayed for each device, these can be a list of select values, a small trend chart, or a gauge. If thresholds are defined the user can select between absolute values or scaled values.

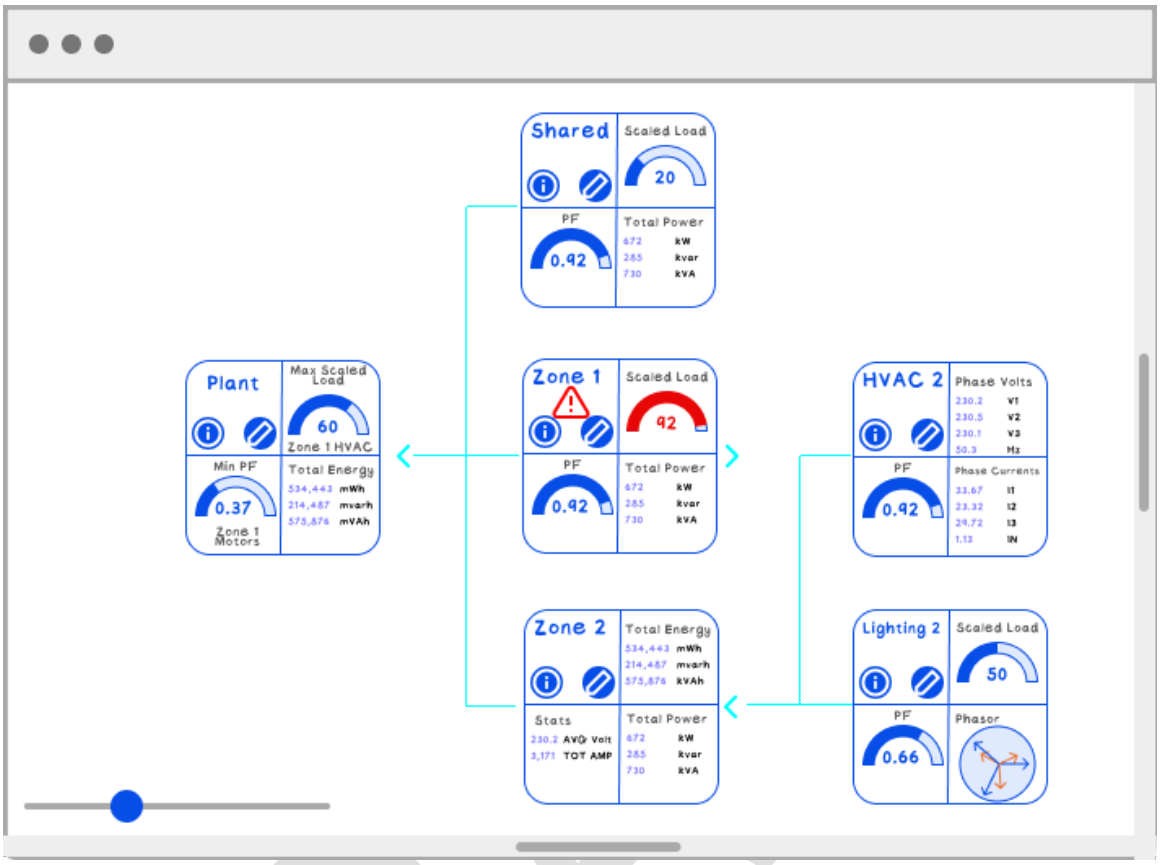

Figure 7- Sketch of a sample site view

When configuring widgets for a group, the user should be able to select if the value being displayed is the average, sum, max or min of the children, when the max or min are viewed the device ID should be specified allowing the user to know which device is associated with the max or min value.

Ideally, the user should have the ability to define a formula using basic arithmatic and not just display a single value.

5.2.2 Device View

When Clicking on a device in the site view the user should be shown the device view. When clicking on a group the values the user can select if the values being displayed are the average, sum, max or min of the children.

The device should allow the user to configure a custom dashboard with different widgets showing the user relevant information for that specific device.

5.2.2.1 Device Info

The basic information about the device

5.2.2.2 History
A user configurable trend plot, histogram, or heat map of the measured values.

5.2.2.3 Diagnostics and Alarms
Diagnostics and Alarms from the device both real time showing if the device is ok or if there are any active faults or errors as well as log of historical alarms and faults.

5.2.2.4 Phasor Diagram
A phasor diagram of all the three phases (I4 when available), current and voltage.

5.2.2.5 Logs
An option to view a list of data logs and the ability to open them.

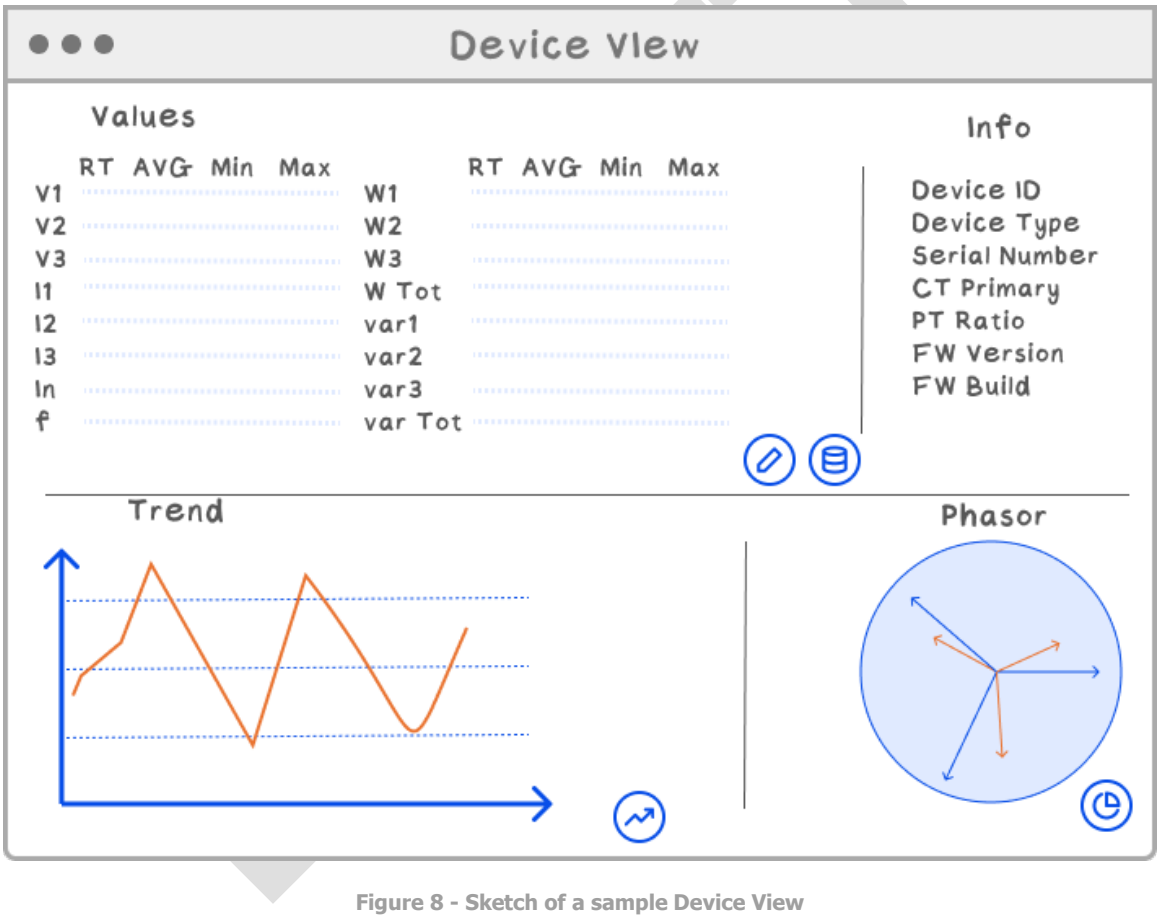

Figure 8 - Sketch of a sample Device View

5.2.3 Trend View
The user should have an easy way to generate graphs and plots from all the measured values, this can be either individual value, a formula combining multiple values, or statistical values.

For example: The Average of all the scaled power of all devices labeled *"HVAC"*

5.2.3.1 Delta Values

When viewing accumulated values there is an advantage of being able to view delta values the trend view should have the capability to plot the deltas between each point in the plot. This data is usually presented best as a bar graph where each bar represents the accumulated value for the specified date range.

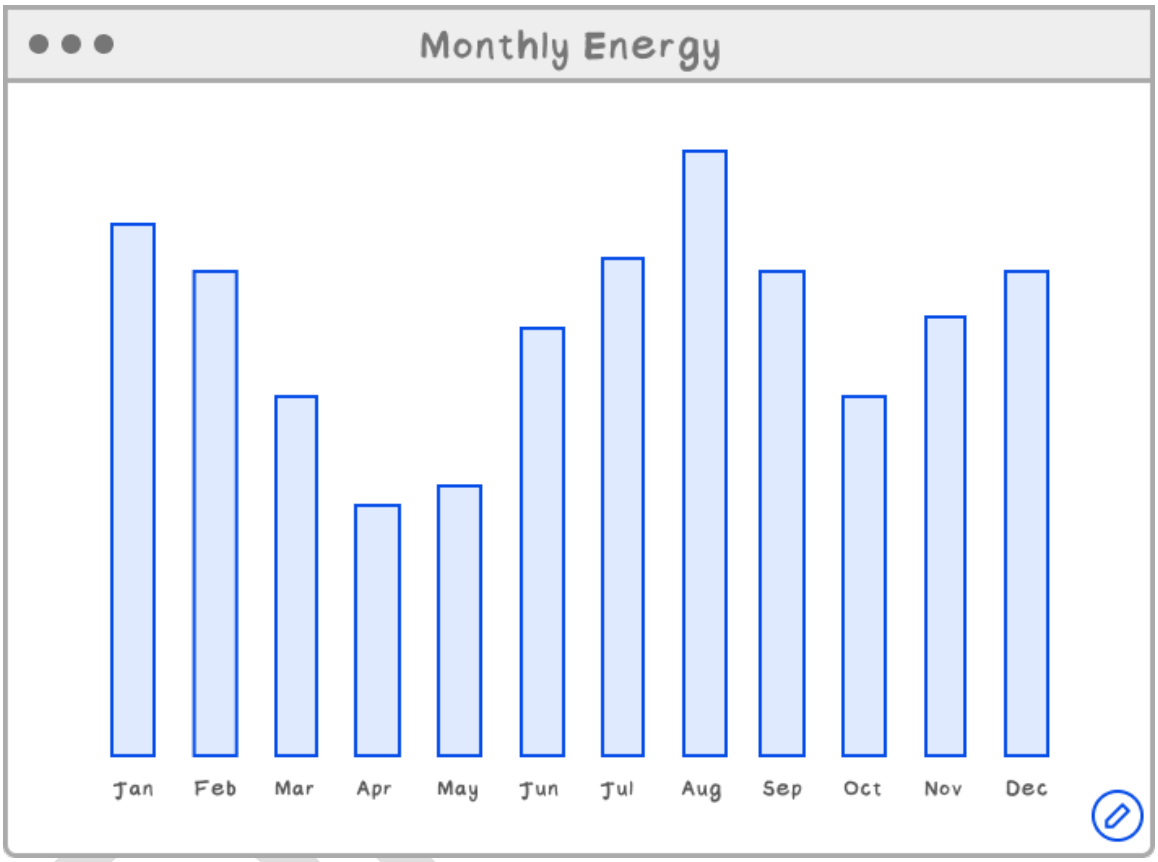

Figure 9 - Sample bar graph of accumulated values

5.2.3.2 Heat Maps

Another useful graphical view is a heat map where values are displayed in a grid with a color scale.

An example grid can be:

- A weekly view with days on the vertical axis and hours on the horizontal axis.
- A monthly view with weeks on the vertical axis and days on the horizontal axis.

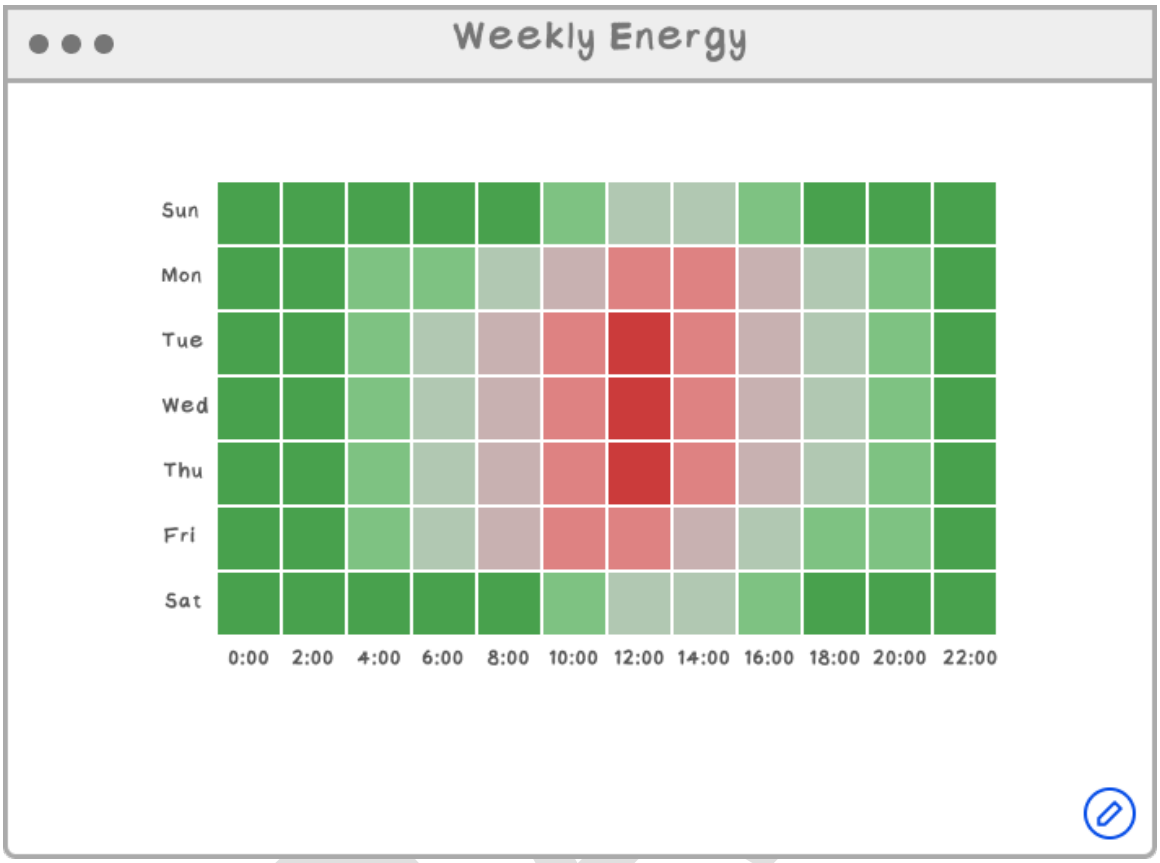

Figure 10 - Example heat map

The value for each box can be the average, min or max or the delta value for that time slot.

5.2.4 Custom Dashboards

The software should allow the user to create custom dashboards like the device view but aggregating data from the entire site.

Aside from the mention widgets the software should have the ability to generate a Sankey Diagram for each parent in the device tree.

5.3 List View

List view essentially a table of user selectable values allowing for easy comparison and custom report generation.

Ideally the list view can be exported to file formats to be used with common spread sheet software.

The software should allow the user to store templates of different list for quick and easy report generation.

5.4 Waveform Viewer

In some cases, mainly when supporting DFR, the meters record waveforms. Ideally the software should have the ability to respond to the fault by polling and storing the waveform (this is done in a similar manner to the datalogs).

The user should have the ability to view multiple waveforms on either a single plot or on multiple plots.

5.5 REPORT TOOL

The software should have a method of generating reports of the polled data, allowing the user to create custom reports from subsets of data. The user should also be able to save a report template to generate repeated reports.

This feature could be offered as a module at an additional cost.

6 Events and Alarms

Alarms can be defined either on the device or in software. The option to support device triggered alarms can be considered a DFR and can be sold as an add on module.

6.1 Device Triggered Alarms (DFR – Extra Module)

Satec meters can allow users to define setpoints and alarms for recording faults with a high precision timestamp. The software should have the capability to respond to the alarm by polling the relevant datalogs as well as initiate a user defined alarm response.

6.1.1 Polled alarms

The software should poll the devices to detect a flag and respond accordingly to that event.

6.1.2 TCP Notification

Some of Satec's meter support a TCP notification mechanism where upon an event the device will access a Modbus server and write values to 16 registers detailing the fault.

To implement this the software would need to implement a Modbus server allowing the device to write to it. The software would then need to respond to the device accordingly.

6.2 Software Triggered Alarms

The software should give the user the ability to define formulas conditions and setpoints to trigger alarms based on the measured values.

6.3 Alarm Response

Aside from logging alarms within the software allowing a user to see previous alarms, the software should also support the ability to send E-mails and SMS in response to alarms.

7 Other Requirements

7.1 Help Files

The software should have built in help files and tutorials showing the user how to use the software.

On top of that it would be helpful if these instructions were available directly from the applicable section (without needing to search).

7.2 Firmware Updates and Device Configuration

I nice to have feature would be the ability to write to the devices this can be used for doing a basic configuration or updating the devices firmware.

It would be extremely useful if this feature allowed batch operation, where the user can select multiple devices and software will run the operation on the all the selected device presenting the user with a report of which devices were successful and for which devices an error was encountered.

7.3 Complete Backup

The software should have the ability to store a complete backup including the entire site configuration along with all previously logged data. In the event of an error or the need to replace the computer this should allow the entire program to be restored to the backed-up configuration.

Additionally, the ability to save backups automatically on a configurable schedule would be extremely useful in the event of a computer crash or a ransomware attack. Ideally there should be an option for the backups to only be snapshots/deltas as not to require full storage for each backup.

This can be used to migrate users from the MDM solution to expert power.

7.4 Data Export

The software should allow the user to export all polled data to common data file formats such as CSV, XML and JSON.

7.5 Site Import/Export

The ability for site configurations to be imported and exported would be useful, this will allow sharing configurations with support. This will also allow to share templates of common configurations allowing the user to implement sites easily.

This file should be stored in text-based format (JSON would be a good example) allowing for version control and diff to be used.

7.6 Run as a Service/Daemon

The software should run as a Service/Daemon where runs in the background, starts automatically at boot, and restarts itself in the event of a crash.

7.7 Encryption and Signed Data (Future)

While not currently necessary, more and more cyber security requirements are becoming common so it is believed that in the future the software will need to support these features. The two main elements believed to be required in the future are encrypted communication and signed data. While not necessary at this point the ideal software candidate would be implemented in a way that would allow these features to be added in the future.

7.7.1 Encryption

The software should have the ability to communicate with Satec devices using an encrypted communication protocol such as Secure Modbus.

7.7.2 Signed Data

The software should have the ability to read and confirm that the read data is coming from the specified device using an authentication protocol. That data should then be able to be shared with third parties while maintaining the authentication.

7.8 Headless Operation

The ideal software candidate would support headless operations where the software is run on a server without an HMI device.

In these cases, the site would be configured on a separate computer and imported into the server. In order to support this the software should have a built-in webserver allowing for a browser interface as well as support a command line interface for control from a terminal.

7.8.1 Web Browser Interface

The software should expose a subset of features using a webserver allowing for remote access using a web browser. This feature should require a login and be limited with user access control. This feature should have the ability to be disabled if the customer has security concerns.

7.8.2 Command Line Interface (CLI)

To allow for remote control of the software the software should offer CLI that can be used by users accessing the PC via SSH or similar methods.

7.9 ExpertPower Client

In some cases, the devices can't be assigned a fixed IP for that Satec has implemented the ExpertPower Client model which allows the device to initiate the communication with software and be identified by its MAC address. After a socket has been established the software will communicate with the device as a standard Modbus client.

7.10 Permissions and Usage Logs

When enabled the software should allow for the creation of different user accounts where each account is given different levels of control.

The following are the different authorization levels and their capabilities.

Admin – Full Access

Privileged – Full access but can't create users or view usage logs.

Operator – Can view everything and modify communication settings, can't add or remove devices.

User – Can only view values.

The software should log the following with timestamps.

- Failed access including username tried.
- Successful Logins and usage time.
- Any changes made and the user who performed them.

7.11 Diagnostics and diagnostic logs

The software should include usage and diagnostic logs so the user and the Satec support team can troubleshoot errors. Having diagnostic records would allow our support team to be more effective.

7.12 Digital Twin (Future)

A growing trend is the use of digital twins to simulate complex systems, after commissioning there is a need to synchronize the physical system to the digital twin for continuous monitoring. While this is not being implemented yet it is believed that in the future the software will need to have the capabilities to synchronize with digital twin platforms.

7.13 Protocol Converter (Future)

The ideal software solution will also serve as protocol converter allowing for other software systems that don't use Modbus to read the device data. These can be local software systems such as BMS software where BACnet is required or cloud software that uses a RESTful API.

Another application would be to integrate with the end users third party softwer for example BI software used for energy efficiency analysis.

While these specifications aren't clearly defined at this point, the software package should be implemented in a way that would allow this to be implemented (relatively) easily.

8 Non-Technical Requirements

8.1 Pricing Model

Since many of our customers wish to purchase the software as part of a CapEx budget and therefor request a perpetual license. It is a requirement the software should be offered as a standalone software package.

The software should have a onetime sale price for the base package as well as additional modules like PQ and DFR. The initial price should include one year of support and upgrades.

After the first year the customer should have the option to purchase an annual maintenance package which would include technical support and software upgrades. If the customer ops-out of the maintenance package no support or upgrades will be offered.

8.2 Support

Satec would be offer Tier 1 support in house but should have the ability to escalate issues to the software provider. It is expected that when issues are escalated the software provider should offer support in a timely manner. As mentioned earlier this support will only be offered in the first year after purchase or to those who purchase an annual maintenance package.

8.2.1 Termination of support

As part of the initial agreement, it is important to clarify what are the terms should the software provider wish to no longer support this package.