SECTION 250000 INTEGRATED AUTOMATION

1.1 INTEGRATED AUTOMATION

A. Vendor shall provide a single, comprehensive Platform that enables team members across the enterprise; including facility managers, operations staff, engineering, executives, and tenants to drive action from building data and guide them in operations, planning, and engagement.

B. Approved Platforms & Companies

1. Company: KODE Labs
   Platform: KODE OS
   Website: kodelabs.com
   Contact: Edi Demaj
   edi@kodelabs.com
   +1.248.495.4353

C. SaaS Platform Summary

1. Platform must be a Software as a Service (SaaS) Platform (i.e., web-based software tool hosted by the Vendor) that can be accessed from any computer (desktop or laptop) or tablet (e.g., iPad) via any modern web browser (Google Chrome, Firefox, Safari etc.) by all individuals with user credentials.
2. The Platform must not require software installation on users¢ computers, or installation on onsite servers

D. Automatic Upgrades

1. Platform must include automatic upgrades at no additional cost to the Client as released as part of the subscription.
2. These upgrades should happen in the background, not disrupt service, and not require any action by users.

E. Scalable

1. The Platform must be able to easily scale from one to thousands of buildings and users.
2. Adding buildings and data in the future must be straightforward and cost effective.

F. Additional Features

1. Platform must be able to easily add additional features to their subscription without system redesign.

G. Mobile Applications

1. Platform must have a native mobile application that can be accessed from any mobile operating systems (iOS, Android) by all individuals with user credentials.

H. Application Functions

1. The mobile application must provide all available user functions for monitoring and control as provided through web-browser.

I. Additional Features

1. Supplemental mobile applications must be available for specific customized user experiences for users such as Tenants.

J. The Vendor has provided evidence that their implementation team has the bandwidth to take on a project of this size.

K. The Vendor has provided evidence that their implementation team has successfully implemented projects of this size, scope, and complexity in a reasonable timeframe.

L. The Vendor must be able to provide, or partner with an organization that can provide on-site installation services if required.

1. Vendor to provide a designated project manager who will be responsible for the following:
   a. Maintain outlined project schedule
   b. On-site coordination with all associated trades and subcontractors
   c. Management of required Client Integration Checklists
   d. Coordination with Client IT to manage and assign, as required, all related device IP and BACnet or Modbus device address and networks
   e. Authorized to accept and execute orders or instructions from Client
   f. Attend project meetings as scheduled by the Client to avoid conflicts and delays
   g. Make necessary field decisions relating to this scope of work
   h. Advise Client that system designs may not achieve expected results
   i. Coordination/single point of contact.

M. The vendor must provide a service / technical support help desk with a guaranteed response time of no more than (1) hour.

1. Vendor must provide telephone support from a dedicated support team.
2. Vendor must provide easy access to support from a dedicated support team via email and phone during defined hours.

N. Professional Escalation:

1. Vendor must make engineering professionals available via established escalation protocols. Escalation process and list of contacts to be provided with the response to this document.

O. Custom Colors & Logos

1. Platform will allow customization with Client's branding, logo, or images.
2. Vendor will be able to modify the colors associated with the user interface, so long as they do not interfere with the Platform visibility and functionality.
3. Vendor must have ability to provide branded Android and IOS apps.

P. Clean Minimalistic Design

1. The Platform must be professionally designed while considering visual appeal, user interaction, and task completion efficiency.

Q. Search Function

1. The Platform must allow users to search for any valid Platform device or point, regardless of where they currently are within the Platform.

R. Navigation

1. A map view must be provided for a graphical representation of buildings and their locations.
2. The Platform must allow for search of any building without the need to scroll up/down.

S. Clear and Simple Terminology

1. All textual descriptions and names should use simple language so that it is familiar to users coming from non-technical backgrounds, too.

T. Loading times

1. The Platform must be able to load a large amount of data at a rapid pace (within one - two seconds) to allow for a fast access and operation with the Platform by its users.

U. The Platform should encompass a well-rounded management of multiple building systems, tenants, amenities, building personnel, data comparison, and energy consumption (i.e., electricity, water, steam, gas, chilled water, etc.).

V. The presented Platform should be modular in nature and adaptable to building systems that are available in a building, whilst maintaining a holistic view and access of all building systems at once.

W. A portfolio map should function as the home page where each building should be clearly identifiable, searchable and selectable for further operations specific to its own function

1. Upon selection, key building system features should be displayable as separate widgets with the ability to customize based on user preference.

2. Images of the buildings which will serve as de-facto profile pictures should be uploadable.
3. Each building should have a building card which identifies it.

    a. Building Name and Address.
    b. Offline Devices and Active Events.

X. The Platform should feature a street level map view which:

1. Will show each building clearly pinpointed based on its physical address.
2. Building markers should serve as quick status updates for the users showing active events or down devices as identified by a unique color.
3. When down devices and active events occur simultaneously, the latter should take precedence in color identification.

Y. Buildings should have the ability to be sorted based on available criteria (i.e., down devices, active events, name, etc.).

Z. A global search function should be made available, on all screens, to allow users to directly search for buildings, floors, devices, points, events, etc.

AA. All available modules should be instantly available, directly from the home page.

BB. The main page of the Platform should have notifications feature to notify users regarding events related to their assigned buildings.

CC. All users should have the ability to submit feedback directly to the Vendor for bug reporting and/or further improvements.

DD. The Platform should also have a profile page where users can set their picture and basic information.

EE. The Platform must provide weather integration though an external API service.

EE-2. Data to be collected should include but not be limited to:

1. Outdoor Air Temperature and Humidity
2. Outdoor Air Enthalpy
3. Current Conditions (i.e., Cloud, Rain, Snow, etc.)
4. 24 hours to 10 Day Forecast
5. Degree Days

FF. The Platform shall have a Network Status widget to identify the following:

1. Network status of Edge Devices
2. Status of all building devices and points
3. Integrity of all points (Stale, Override, Event)

GG. An Events Overview widget will provide a list of the most recent active, unacknowledged events reported by the Platform.

1. The Summary widget will allow for a customized display of either Devices and/or Systems as direct links while including up to four (4) selectable points from each Device or System to display.

HH. The Platform shall provide the ability to connect points from different devices, to be viewed as a single System.

II. Buildings shall be able to have specific Area(s), or floor(s) assigned to them to display specific data and devices.

JJ. An Area should allow for the following information to be displayed

1. Uploadable floor plan in an SVG format
2. Ability to add devices and temperatures in their appropriate locations
3. Ability to add additional layers such as Fire and Lighting
4. Clustering of like data values to show Minimum, Maximum and Average values

KK. Selection of devices on the floor plan will provide to options

1. Action window to adjust connected points
2. Navigation to specific Device Details

LL. The Platform shall be able to perform configuration updates of multiple selected devices simultaneously based on applied filters.

MM. The batch update functions include but are not limited to

1. Area
2. Location
3. Equipment Tag(s)
4. Device Reference(s)

NN. Virtual Devices

1. Platform shall have the ability to create Virtual Devices which:
    a. Represent physical devices based on a specific selection of hard-wired (analog and digital I/O) and variable points (set points, calculated variables, control loops).
    b. Maintain original device addressing to perform active read/write control.
    c. Can be applied to Platform Templates as a typical network device.

OO. An embedded wizard tool within the platform will allow the Vendor or System Admin to discover network devices available in the assigned Edge Devices with the following features at minimum:

1. Select Edge Device
2. Select Devices and Points
3. Batch update of devices and points

PP. The Platform shall be able to perform configuration updates of multiple selected points simultaneously based on applied filters.

QQ. Batch updates should be made possible for:

1. Naming of points,
2. Units of measure,
3. Tags (adding and/or removing)
4. Minimum and maximum values depending on point type,
5. COV tolerance for history collection
6. Point precision.

RR. The Platform shall be capable of using pre-defined Templates to update all device points as part of a single operation.

SS. The Platform shall provide the ability for individual device selection.

1. The Platform should showcase all inputs, variables, and outputs of the device as available or discovered on each device.

TT. Two (2) Device Detail views shall be available:

1. Default View arranged by Inputs, Variables, Outputs
2. Dashboard View
3. Table View with ability to export time series via CSV files from the device(s) for selected points.
4. All data history should be displayed and made filterable for specific time periods as required by the user.

TT-2. Read/Write/Command functions are available on all data points defined as writeable.

TT-3. Users should be allowed to employ the use of custom tagging on devices as points, as needed.

UU. Each Device, as part of its Device Detail, will offer the ability to display all devices which have been linked as part of the Device Batch Update.

X. The referenced device(s) will display in a split screen format for viewing two (2) devices simultaneously. BBB. The Platform should make possible attaching of various documents related to device functionality including

1. Photos
2. Documents
3. Notes

WW. In addition to the default Device Detail view, a secondary Dashboard View should be provided. This Dashboard View should merge the functions of data point display with historical data for all points. And provide the following:

1. Sort all points by Type
2. Ability to select each individual point for display as part of the associated chart
3. Apply predefined and custom time periods
4. Minimize groups of points by Type

XX. Schedules

1. Schedules should be managed for all devices in a simple and efficient manner, such that they allow for each user to select times and dates of device operation and non-operation based on custom parameters.
2. Schedules should be able to synchronize with local BAS schedules where applicable to maintain both systems up to date.
3. Schedules should be able to be created and linked to associated devices from the Platform.
4. Platform should have the ability to add/modify calendar schedules and special events.
5. Platform should have the ability to batch schedule updates across multiple buildings within a portfolio.
6. All available schedules within a building or portfolio should be searchable and discoverable through a designated function. The user should be allowed to include or omit the schedules as needed.
7. All selected schedules should be writable, editable, and inclusive of the following information:

    a. Name
    b. Location/Area
    c. Point Type
    d. Default Value

1.2 Notifications

A. The platform must support various notification channels including in-app, email, SMS, and push notifications.

B. Users should have the ability to configure and customize alerts based on specific events or triggers within the building.

C. Immediate notifications must be sent out for critical building events or anomalies to ensure timely action.

D. Users must be able to set their notification preferences, choosing which events they wish to be alerted for and through which channel(s).

1. Users should have the option to set 'silent hours' during which non-critical notifications are muted.

E. The system should provide notifications based on user hierarchy or roles, ensuring the right personnel are alerted for specific events.

1. Ability to send notifications to user groups or departments to ensure coordinated responses to events.

F. The platform should offer customizable notification templates to ensure clarity and relevance of the message content.

G. In the event of multiple similar events, notifications should be batched to avoid flooding the user with repetitive alerts.

H. Administrators should have access to analytics related to notifications, such as response times, frequency, and user engagement

1.3 Event Management

A. Event Summary

1. Users will have access to view all events from all available building systems. Events are to be identified by three (3) distinct priorities as follows:

    a. Critical
    b. Warning
    c. Alert

B. Events should be visible from a single window (preferably through a table view) with the following information:

1. Event name
2. Building name
3. Associated device(s)
4. Event class and type
5. Date, start time and duration.

C. Permitted users should be able to confirm or acknowledge present events.

1. This functionality should serve to inform all other personnel that the user has tagged him/herself as responsible for resolving the event.
2. The event itself will remain active until the condition is resolved.
3. Other users should see the same event as acknowledged
4. In cases of Critical and Warning events, permitted users will be able to receive e- mail, web, and mobile app notifications for any relevant events associated with the building(s) that they manage.

D. Notifications should be configured in a manner that allows for automatic escalation

E. The number of possible escalations should be up to three escalation levels at minimum.

F. All events older than 30 days should be archived and only viewable through using a functionality designed to enable the showing of past events.

G. The class, type and status of events should be filterable via a dedicated function.

H. The Platform will allow System Admins to create new event Classes and Types as required and assign User Roles to the selected Class for Critical, Warning and Alert priority level events.

1.3 Event Details

A. Each Event shall have a device detail page which will identify all information as identified as part of the Event Summary and include:

1. Historical chart showing the time interval in question and indicators of when the event started/ended.
2. List of all Users assigned to the event Class that received this specific event notification.

1.4 Routines

A. Routines are defined as a feature which allows users and administrators to set custom logic including parameters and requirements, to generate events for one or more similar devices.

B. Event routines should be fully customizable for creation of distinct types, including but not limited to temperature high limit, temperature low limit, fire alarm detection, fire alarm fault, pump failures, etc.

C. Configuring a routine should be completed via a graphically represented interface which will allow users to define their own routines using logic blocks as follows:

1. Outputs (i.e., alarm triggers)
2. Inputs (strings, numbers, and Boolean values)
3. Bases (numbers, Boolean, strings, result, and/or point statuses)
4. Arithmetical (i.e., difference, product, quotient, mean, maximum, and minimum)
5. Conditional (If, numeric switches, etc.)
6. Logical (Or, Not, And, Merge, etc.)
7. Comparison rules (greater than, less than, equals, etc.)
8. Timing rules (i.e., delays)

D. Routine events should have the ability to be saved and displayed in a list view for users to select for further editing, duplication, or deletion.

1.5 Device Audit Tool

A. Platform shall provide a wizard to create an audit function to allow for comparison of similar devices and points within a building.

1. Audit generation is built based on selection of devices and points from defined Templates
2. Users should have the ability to create an Audit configuration and:

a. Preview the audit table for further editing
b. Save assigned to a specific building
c. Save as template for use on any building within the portfolio

B. Audit will display live data from the selected devices and points:

1. In a table view format and can export the current data view
2. Enable chart view for graphical representation of history data
3. Points should be filterable based on their integrity, status, and area.

C. All audits must be exportable to CSV.

D. Saved Audit templates should be editable and removable at the user's discretion.

1.6 Building Access

A. Building, portfolio, and/or property managers should have a straightforward way of managing their engineers, control team members, and supporting staff between multiple buildings. This becomes applicable when there are rotations of employees between one or more buildings, substitutions, and filling in for staff that may be on leave.

B. Platform must have the ability to provide different level access to different users including but not limited to:
1. Specific building access
2. View only vs full ability to write/command
C. Platform must have the ability to integrate with any Single Sign-on (SSO)
D. Platform must have the ability to set multi-authentication across all users.

1.7 Administration

A. As an integral feature encompassing other modules mentioned throughout this document should also be an administration section which will govern all basic configuration of the Platform.

1.8 Buildings

A. A sub-menu where all buildings can be added, removed, and edited should be present as a part of the Administration menu.

B. All buildings that have already been added as a part of the existing portfolio should be visible and browsable by any user that has administrative permissions.

C. A form to add new buildings should be viewable upon clicking the function when in the building's sub-menu.

D. The information required to create a new building(s) should include:

1. Building Name

2. Address
3. Latitude and Longitude (used to display the building on the map)
4. Primary function
5. Year of construction
6. Total Square Feet
7. Building Image

E. Upon saving, the building should be added to the existing portfolio of buildings as well as appear on the map as described in the Portfolio Map section of this document.

F. User management can be accomplished in two (2) ways:

1. Users are assigned within the building, under the user tab
2. Buildings are assigned to users, under the building tab.

1.9 Edge Device Management

A. Edge Devices should be able to collect and store data coming from devices and their respective points. Furthermore, edge devices should be able to initiate and maintain secure connections to the Platform and an eventual offsite storage location. The device should be an embedded mini-PC with an SSD drive.

B. The devices shall be managed via an Ethernet connection and a built-in web interface. Through this connection, the device should be configured, managed, and monitored as needed.

C. Configuration of devices should include:

1. Password change
2. Restart and Shutdown
3. Save Database function
4. IP Configuration
5. Adding and removal of sites
6. Inclusion and/or exclusion of device ranges.

D. Monitoring of onsite device should include:

1. Hardware status
2. Archiving statistics
3. Connection statistics
4. Memory and storage usage
5. Network Status

1.10 Sensor Data Collection, Processing and Storing

A. Overview

1. The system must collect history for all points previously discovered with no need to do custom configuration.
2. History collection for new added devices and points must start as soon as those devices are discovered/added on the Platform.
3. History collection is mandatory to ensure data quality and provide the ability for advanced analytics, fault detection and alarms.
4. History must be collected not only for points but also for their devices and edge devices/controllers (i.e., JACE) to be used for up-time/down-time analytics.

B. Collection

1. Collection interval must be 30 seconds or less
2. The number of collected points and frequency should not affect the Platform and database performance.

C. Data Cleaning & Normalization

1. Collector must detect and remove faults and incorrect data points such as spikes.
2. Cleaning must be available - in a manual way if needed.

D. Status Change Updates

1. System must detect status and state changes on all controllers, devices, and points.
2. All status transitions must be saved and reported.
3. All status changes must be stored to provide an aggregated view of all controllers, devices and points.
4. Status change alarms for ok-to-down status must detect the level in hierarchy
    a. If the connector is down one alarm should be sent for that specific connector and not all devices and points on that connector.

5. If a status change update occurs in upper levels in hierarchy (i.e., communication to a building is down), all devices and points for that building must get the same status as the building.

E. Storing

1. Collected data should be stored in a specialized analytics database together with other metadata.
2. Databases must be able to operate even if reports require scanning of billions of sensor data.
3. All raw data must be stored in a compressed way to minimize storage size.
4. All raw data must be available for querying from analytics apps or batch data processing.
5. Data retention must be at least 3 years.
6. Continuous backups should be enabled at least every two hours and it must provide the ability to do back in time recovery.

F. Real-time / WebSockets

1. During the processing time the system must publish all collected values to subscribers using WebSockets.

G. Scalability

1. Collector must scale up-and-down based on the number of buildings, devices, and sensors.
2. Increasing the number of buildings in the Platform must not affect application and database performance.
3. Increasing the number of connected subscribers on real-time data using WebSockets must not affect the performance of the app.

1.11 Users and Permissions

A. Unlimited Users

1. Software licensing should allow for an unlimited number of users.

B. User Permissions

1. Vendor will supply a set of predefined Roles that each user can be assigned to.
2. Vendor will supply added custom roles as required by the Client.
3. Each Administrator user role will have the ability to assign added users to the supplied roles.

C. User Profile

1. Each user shall have the ability to modify the following items within their profile:

    a. Update Contact Information
    b. Reset Credentials as required
    c. View Active Sessions

2. Administrative users shall be able to perform the following actions for users:

    a. Add/Remove Users
    b. Enable/Disable Account
    c. Update Contact information
    d. Reset Credentials
    e. View Active Sessions
    f. Assign Buildings to Users

D. Attribute Based Access Control (ABAC)

1. Every Platform action should be authorized using ABAC to provide a higher level of flexibility for assigning user permissions for each resource (i.e., floor, device, point, etc.) as opposed to Role Based Access Control where a write/read permission is assigned on a role basis.

1.12 Ontology

A. The platform must support the integration of established and widely-accepted ontologies such as Google DBO, BrickSchema, or equivalents.

B. Through the use of ontology, the system should be able to accurately and uniquely identify all devices and sensors present within the smart building environment.

C. The platform must allow for the integration of new or custom ontologies, ensuring compatibility with future developments or industry-specific requirements.

D. An intuitive interface should be provided to allow users to map or align existing building device and sensor data to the ontology, aiding in data normalization and identification.

E. The Ontology should encompass standardized units or unit groupings for each measurable sensor/data point, ensuring clarity in data interpretation by precisely denoting the measurement type and scale for any given sensor value, thereby promoting consistent data analytics and visualization across the platform.

F. The platform should be able to interpret and utilize the relationships defined by the ontology. This includes understanding hierarchies, equivalencies, and interconnections between devices and sensors.

G. Utilizing the ontology, the system should auto-tag devices and sensors with standardized labels, ensuring consistency and ease of data analysis.

H. The platform should provide visualization tools to showcase the relationship between different devices and sensors as defined by the ontology.

I. Users should be able to search and query devices and sensors based on ontology-defined attributes, ensuring quick access to relevant information.

J. Using the ontology definitions, the platform must be capable of identifying anomalies or faults in device behavior, ensuring optimal performance and quick problem resolution.

K. Every feature of the platform should be intrinsically linked and operational based on the integrated ontology. This ensures that the ontology not only serves as a foundational layer but actively assists in automation, ease of integration, and the efficient functioning of all platform capabilities. The ontology's presence should enhance user experience, streamline operations, and promote seamless integration of devices, sensors, and other smart building components.

L. Comprehensive documentation on the integrated ontologies should be provided, and training should be available to ensure operators and users can effectively utilize ontology-driven features.

1.13 Tenant Module

A. A section of the Platform shall be designed for management of tenants and management of select building aspects by tenants.

B. Building managers should be able to register and assign tenants to their rented locations. This should provide the tenants with access to the Platform as developed by the Vendor.

C. In the Platform, the tenant shall be able to:

1. Request After-Hours Services

    a. For HVAC and lighting operation

    b. Calculate overtime cost for used services

    c. Send notifications to tenant and building management as needed

2. Work Order Management

    a. Create and announce work orders by tenants and building management

    b. include description and pictures

    c. Work Order status and assignee tracking

    d. Analytics of work order causes and forms

3. Send notifications to tenant and building management as needed

D. Tenant Controls

1. Tenants should have intuitive controls for devices linked to their spaces – adjusting temperatures, lighting levels, and other environment parameters.

2. Each tenant should have a personalized dashboard that provides real-time and historical data visualizations pertinent to their space.

3. Tenants should be able to view metrics on air quality, including factors such as CO2 levels, humidity, and particulate matter, ensuring a healthy working/living environment.

4. Tenants should have access to real-time occupancy data for their spaces, aiding in decisions related to space utilization, meeting room availability, and energy efficiency.

5. Provide tenants with insights into their energy usage patterns, helping them make informed decisions to optimize consumption.

6. While tenants have control over their devices, the building operator should be able to set maximum or minimum thresholds (e.g., temperature ranges) to ensure overall building efficiency.

    a. Ensure robust security measures to prevent unauthorized access to a tenant's controls, maintaining the privacy and integrity of each space

1.14 Fault Detection Analytics

A. The Platform shall host the FDD software with off-site servers that shall require no maintenance or management by the building personnel.

B. FDD should be able to notify building engineers about the occurrence of the fault, type of fault, and its location.

C. FDD should be an automated process that detects faults and provides continuous feedback.

D. FDD should be able to provide a report on a faulty device with details of the fault and the steps needed to be taken to address the fault.

E. Minimum Required Capabilities:

1. Determine the stability of control devices (valves/actuators/speed drives).
2. Compare sensor readings to set points and identify out-of-range errors.
3. Evaluate and rank faults based on various impact categories, including comfort, energy, environmental, safety, property, ventilation, heating and cooling, as well as urgency.
4. Create an index for all devices based on their performance.
5. Identify simultaneous heating and cooling in a system.
6. Identify best and worst performing devices by their ability to maintain process variables (temperature, pressure, flow) within set point.
7. Detecting leaking valves.
8. Detecting stuck valves and dampers.
9. Undersized cooling and heating coil.
10. Track and report on devices detected as experiencing multiple resets.
11. Track and report on cycling control loops.

1.15 Functional Testing Tool

A. The Platform will offer a Functional Testing Tool (FTT) which will allow the Users to initiate manually or schedule a functional test of the components related to a unitary device such as a heat pump, VAV or Network Thermostat.

B. This tool will follow a defined sequence based on the unitary device type to drive commands and verify status of the following:

1. Fans
2. Dampers
3. Heating and Cooling Valves/Stages

C. Based on completion and status of these functional tests a basic score will be applied to components that have either passed or failed the testing. This device score will be used in

combination with other scores from all unitary devices within the building/site to provide a general building equipment health score.

1.16 Optimized Start/Stop

A. The platform should have available a feature (referred below as OSS) designed to predictively and autonomously control the start and stop times of HVAC equipment based on various parameters, ensuring optimal occupant comfort while maximizing energy efficiency.

B. The OSS should leverage Machine Learning algorithms to analyze historical data, current weather conditions, and device performance, predicting optimal start and stop times for HVAC equipment.

   1. The system should consider real-time external parameters like outside temperature, wind, humidity, and other relevant factors to ensure accurate predictions.

   2. OSS should prioritize energy savings, preventing unnecessary early start-ups or extended run times beyond the building's operational hours.

   3. E.g. By the start of the building's schedule (e.g., 8AM), the OSS must ensure that temperature setpoints are achieved for optimal occupant comfort.

   4. The system should predict and gradually reduce HVAC operations as the end of the building's schedule approaches, ensuring energy isn’t wasted after occupants have left.

C. Operators should have the flexibility to configure OSS settings, enabling or disabling the feature for specific devices as required.

D. OSS should work at a device level, determining unique start and stop times for individual HVAC devices based on their performance and location within the building.

E. Beyond just suggesting optimal times, the OSS should have the capability to autonomously control devices, turning them on or off as needed.

   1. While the OSS operates automatically, users should have the option to override predictions and manually set start and stop times if necessary.

   2. OSS should always ensure that any autonomous decisions made do not compromise the safety or longevity of the equipment.

F. The system should continuously learn from any discrepancies between its predictions and actual optimal times, refining its algorithms for improved accuracy over time.

G. Detailed analytics should be available, showcasing energy savings, prediction accuracy, and other relevant metrics to assess the performance of the OSS feature.

1.27 BI Tool

A. The platform should have an integrated BI tool that serves as a data visualization engine, enabling the creation, customization, and sharing of dashboards that visually represent time-series data and custom calculations from various building sensors.

B. The tool should provide diverse visualization widgets, including but not limited to line charts, bar charts, tables, heatmaps, and combined chart types.

C. Dashboards should support interactive elements, allowing users to drill down into data, filter views, or pivot information based on their needs.

D. Dashboards should be more than just static visuals. They should facilitate click-through navigation, allowing users to delve deeper into detailed pages such as device information or directly command and control points, enhancing the interactive and operational capabilities from a single interface.

E. Visualization elements should support dynamic coloring based on sensor values, offering an immediate indication of 'good' or 'bad' conditions to building operators.

F. The BI tool should ensure real-time data visualization and reporting, enabling users to see the most current building data and swiftly respond to emerging situations or changes.

G. The tool must be adaptable to various data types, ensuring a comprehensive analysis across all building systems.

H. Users should have the capability to create and define custom metrics, highlighting the status and performance of crucial devices in the building.

I. Building custom dashboards should be straightforward, with functionalities to share them among different users or roles within the organization.

    a. Sharing permissions should be configurable, allowing access to specific dashboards or data views based on user roles.

J. The tool should come equipped with a range of predefined templates tailored to smart building needs, representing various device types and typical scenarios.

    a. Beyond the default templates, operators should have the freedom to design and store their custom dashboard templates.

K. The BI tool should provide versatile data exporting options, allowing users to extract information in various formats, including CSV, PDF, and potentially others for ease of analysis and sharing.

L. The BI tool should be seamlessly integrated with the smart building platform, ensuring consistent data flow and interpretation.

1.18 System Performance Monitoring

A. The platform should be able to use its BI tool and other features and functionalities to provide insights into how effectively the integrated systems function and where improvements are needed.

B. The System Performance Monitoring metrics should allow building operators to ensure optimal performance, extend equipment longevity, save costs, and enhance the overall efficiency of the smart building infrastructure.

C. System Performance Monitoring should offer but not limited to these metrics:

   a. Uptime/Downtime Metrics: Monitor the operational status of every device and equipment, detailing how long they've been active or inactive.

   b. State Duration Tracking: Assess how long equipment remains in specific states (e.g., standby, active, or fault) to help pinpoint inefficiencies or malfunctions.

   c. Occupancy Metrics: In connection with performance, assess how building occupancy (or lack thereof) impacts system efficiency.

   d. Load Analysis: Measure and analyze the load on different systems, ensuring they operate within safe and optimal limits.

   e. Historical Analysis: Archive past performance metrics to analyze trends over time, facilitating predictive maintenance and strategic improvements.

   f. Energy Consumption Metrics: Monitor the energy usage of devices and systems in real-time, aiding in optimizing energy consumption and identifying wastage.

   g. Performance Benchmarks: Set target performance metrics based on industry standards or past building performance and compare real-time data against these benchmarks.

D. The BI Tool should be able to display the above-mentioned performance metrics on interactive dashboards, allowing users to dive deeper into specific data points or timeframes.

E. Part of the System Performance Monitoring there should be the ability to set up immediate notifications for any deviations from expected performance metrics or when equipment operates outside of designated parameters.

1.19 Energy Monitoring

A. The BI tool, together with other dedicated features and functionalities of the platform should be able to provide comprehensive monitoring and reporting capabilities for the Energy consumption monitoring

   a. The platform should provide a comprehensive overview of the building's total energy consumption, split by various energy types (e.g., electricity, gas, water, renewables).

b. Track peak energy demands in real-time and historically to aid in demand-side management and optimize energy tariffs.

c. Offer a clear representation of the building's carbon emissions, based on its energy consumption, aiding in carbon footprint reduction efforts.

d. If renewable energy sources are available, the platform should showcase the amount of energy generated from renewable sources within the building, such as solar panels or wind turbines, and its contribution to total energy consumption.

e. Compare the building's energy performance against industry benchmarks or similar building types to determine efficiency standing.

f. Correlate occupancy data with energy consumption to identify patterns and implement efficient energy-saving strategies during low occupancy times.

g. Provide insights into historical energy consumption trends to predict future energy needs and plan accordingly.

h. Generate real-time alerts for abnormal energy consumption spikes, helping in quick detection and resolution of potential issues.

i. Implement predictive analytics to forecast energy consumption based on past data and current trends.

j. Provide actionable insights and recommendations to enhance energy efficiency based on real-time data analytics.

B. The platform should provide Sustainability Reporting and allow the creation of customizable dashboards highlighting sustainability metrics, including water reuse, waste recycling rates, and more.

a. It should be able to generate detailed energy and sustainability reports tailored for different stakeholders, from building managers to environmental agencies.

C. The platform and the BI Tool should allow users to set and track custom Key Performance Indicators (KPIs) related to energy and sustainability goals.

D. The platform should have the capabilities to enable seamless integration with other sustainability platforms or tools for expanded metric tracking.

E. Bill Reporting and Integration should be supported to give the building operators a detailed understanding of their energy expenses, enabling more informed decisions on consumption patterns, savings, and efficiency measures.

a. Integrate with utility providers such as Urjanet to automatically import and sync monthly energy bills, ensuring up-to-date cost reporting.

b. Allow users to view a real-time and historical breakdown of energy consumption costs, translating energy metrics into actual dollar values.

c. Provide flexibility in setting custom periods for bill calculations, be it monthly, quarterly, or annually.

d. Enable the creation of virtual meters that can combine data from multiple physical meters or segment a particular physical meter to analyze energy consumption for specific sections or systems in a building.

e. Compare bills over various periods to identify trends, anomalies, and savings opportunities.

f. Allow users to set energy cost budgets for specified periods and monitor expenditures against these benchmarks in real-time.

g. Provide tools for conducting energy audits, creating detailed reports highlighting areas of inefficiencies and their associated costs.

h. If a building uses sub-metering, provide detailed cost breakdowns for individual tenants or departments, ensuring transparency in energy billing.

1.20 Graphics Tool

A. The platform should have the capability to build graphics (referred below as the Graphic Tool) with the aim of providing a visual, interactive, and real-time representation of the building's systems and devices. With the key mission to simplify complex device interconnections into intuitive layouts, helping operators to quickly understand, analyze, and make informed decisions for efficient building management.

B. The tool should offer capabilities for detailed and zoomable floor plans where devices and sensors are overlaid, allowing operators to easily pinpoint locations and identify potential issues or zones of interest.

    a. Allow users to view different layers on a graphic, such as HVAC, lighting, or security, either separately or simultaneously.

C. The Graphic tool should have the ability to render real-time visuals of critical building systems like HVAC, showing the interconnection between devices, flow directions, and relevant sensor values.

    a. Present live sensor values directly on graphics, ensuring that the data is always current and operators can immediately recognize deviations.

D. Allow users to click on visual representations of devices to access deeper information, configuration options, or even control commands directly from the graphic interface.

E. The tool should empower operators or system administrators to customize graphics based on specific needs, adding or removing devices, adjusting layouts, or changing visual themes.

F. The tool should support visual indicators that change color or design based on device status (e.g., operational, fault, standby), making it easier to identify issues at a glance.

G. There should be the ability to replay past data on graphics, allowing operators to visually backtrack and analyze events or anomalies.

H. Graphics should render seamlessly across various devices, whether viewed on a desktop, tablet, or even mobile for on-the-go operators.

1.21 API Integrations

A. The platform must integrate diverse smart devices through their APIs, ensuring all building data is accessible within a single interface, providing the operator with a 'single pane of glass' experience.

    a. To keep up with technological advancements, the platform should be capable of integrating with any new device or sensor type, irrespective of its manufacturer or inherent protocols.

B. Leveraging its Ontology, the platform must dynamically normalize data from integrated devices to ensure consistency and compatibility, allowing seamless interaction with its in-built tools and features.

C. Post-integration, the platform should immediately reflect the data from new devices in its visualizations, ensuring operators have access to real-time information without delays.

    a. Upon integrating a new device or sensor, the platform should allow operators to customize dashboards, ensuring they can prioritize and visualize data as per their preferences.

D. With multiple devices getting integrated, the platform should prioritize data security, ensuring all data transmissions and storages through API integrations are encrypted and compliant with relevant security standards.

E. The platform should offer built-in tools or environments for testing API integrations, ensuring flawless communication between devices and the platform before going live.

F. The platform should have robust error handling for API integrations, notifying operators in case of integration failures, data mismatches, or compatibility issues.

G. The API integration toolset should be scalable, accommodating a growing number of devices and sensors as the building's needs evolve.

    a. As smart devices get firmware or software updates, the platform's API integration capabilities should adapt without necessitating extensive reconfigurations

H. Given the critical nature of data from various devices, the platform should automatically backup integrated data and provide a straightforward restoration process in case of any data loss scenarios.

1.22 User Activity Log

A. A User Activity Log will be included as part of the Platform which will allow administration level users to view all functions which have been completed or initiated within the Platform. This includes, but not limited to:

   a. Override Action
   b. Auto Action
   c. Set Action
   d. Event Acknowledged
   e. Work Order Complete
   f. User Profile details of Creation and Assignment

B. This log will provide the administration user the ability to filter, and sort, based on all the above-mentioned criteria and export the log as a CSV and PDF report.

1.23 Security Requirements

A. SSL Requirements

   1. Platform must offer SSL for secure transmission of Client data
   2. Platform should feature an auto-renewal of SSL Certificates to ensure a continuous operation and prevent failure in case of a human error caused by non- renewal of an SSL certificate

B. Upload authentication

   1. Uploads of Client data must be authenticated and access-controlled to restrict access to user data

C. Hosted product

   1. Platform must be a hosted SaaS offering so that security is the Vendor's concern, not the Client's

D. Active Sessions

   1. All users and admins must see their active sessions coming from all devices (web and mobile) and have the possibility of deactivating suspicious or unnecessary sessions
   2. Attribute Based Access Control (ABAC)
   3. Every Platform action should be authorized using ABAC to provide a higher level of flexibility for assigning user permissions for each resource (i.e., floor, device, point, etc.) as opposed to Role Based Access Control where a write/read permission is assigned on a role basis.

E. Physical and Virtual Security

   1. Equipment hosting Client data must be located in a secure facility with two-factor authentication for physical access.
   2. Firewall

3. Equipment hosting Client data must be secured by firewall
4. Access control
5. Dashboards must have the option of being password protected to prevent access by unauthorized parties
6. Security monitoring
7. Equipment hosting Client data must be monitored for security vulnerabilities and patched expeditiously

F. Disaster Recovery Management

1. All databases, files, and services should undergo regular replication in multiple data centers located in different regions to ensure high-availability, protection from natural disasters, and unavoidable emergencies.

G. Runtime Security

1. The Platform should be able to identify a service(s) acting maliciously and take proper actions to protect the workloads.

H. Regular Updates

1. The Platform must ensure a constant updating of operating systems to have the latest security patches implemented as soon as they become available.
2. The Platform must ensure a high availability of the operating system even when updates are being conducted.

I. Specialized Operating Systems

1. The Platform must contain within itself specialized operating systems which exclusively ensure the proper functioning of mission critical applications and minimize the risk of exploiting vulnerabilities from outside threats.

J. Audit Logging

1. All Platforms actions undertaken in the server side as well as the Client side must be properly logged, and accessible at all times, for administration purposes and auditing.

K. Networking Policy

1. Services should communicate with one another app-to-app, microservice-to-microservice, and microservice-to-database through a secure and private network setup that uses private IP addresses. The only part that should be exposed to the public should be an authenticated public API.

L. Automated Vulnerability Scanning and Detection

1. The Platform should feature scanning for and detection of common vulnerabilities including, but not limited to cross-site-scripting (XSS), mixed content (HTTP in HTTPS) and outdated and/or insecure libraries. This should enable an early identification and delivery of an extremely low level of false-positive rates.
2. Pre-release scan for vulnerabilities of executable files/images for potential and known vulnerabilities.

M. Standardization and Compliance

1. The Platform must be SOC2 certified.
2. The Platform should be compliant with ISO 27001, ISO 27017, ISO 27108, HIPAA, and PCI-DSS.

1.24 Data Centralization and Access

A. All system data related to users, buildings, meters and other data should be accessible via secure JSON API:

1. API should be accessible over HTTPS
2. API must have advanced security features for authentication and authorization.
3. Resource data must be exportable to CSV

B. All relevant ERP metadata must be centralized including building data, user data, weather data, documents, images, and any other file that is uploadable in the Platform.

END OF SECTION