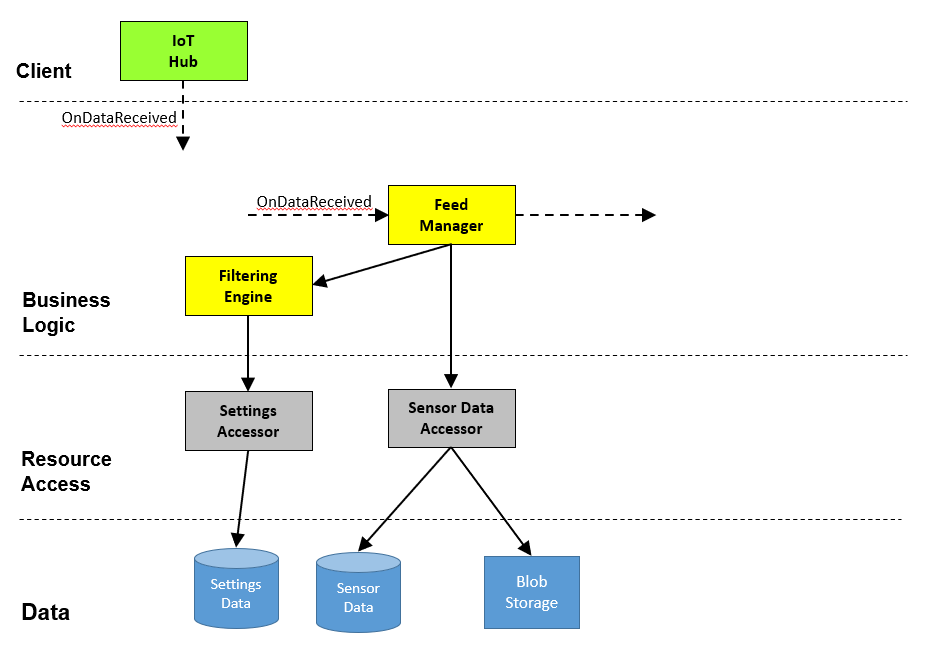
# Device sends data (reading, status, alarm, health, …)



## Description

Data from individual IoT devices will be sent through the local network to the gateway device. The data will then be sent to the IoT Hub which will initiate this call chain for every message received. This message will handle all message types from the system: Status updates, Alarms, Reading Data, and Health check-ins. Data will be stored in the relational database and into cold storage for archiving.

## Component Roles

* IoT Hub
  + Sends data from the configured devices
* Feed Manager
  + Receives the messages from the configured devices and manages the workflow
* Filtering engine
  + Determines message type and reformats for use through the system
* Settings accessor
  + Returns the settings data for the message type that was received
* Sensor Data accessor
  + Stores the message data/type to the sensor database and cold storage

## Use Case Sequence

1. IoT hubs sends message
2. Feed Manager receives new message
3. Feed Manager calls filtering engine
4. Filtering engine retrieves settings from the settings accessor based on the message type
5. Filtering engine returns formatted data to the feed manager
6. Feed Manager sends formatted data to the sensor data accessor for storage
7. Feed Manager sends message to queue for next process in the chain

## Services

### Feed Manager

|  |  |
| --- | --- |
| OnDataReceived | HubMessage, Object |
| SendQueue | SensorMessage, Object |

### Filtering Engine

|  |  |
| --- | --- |
| FilterDeviceSettings | CustomerId, long  DeviceId, string |

### Settings Accessor

|  |  |
| --- | --- |
| LoadDeviceSettings | CustomerId, long  DeviceId, string |

### Sensor Data Accessor

|  |  |
| --- | --- |
| SaveSensorData | CustomerId, long  DeviceId, string  SensorData, Object |

## DTO Definition

List and explain the DTOs needed for this call chain. Listing all properties of the DTO is not required.

### DTO Definition

|  |  |
| --- | --- |
| HubMessage | Contains device sensor data sent from the Hub |
| SensorMessage | Contains properties to define a sensor message to send to the queue |
| DeviceSetting | Contains properties to define the Device Settings |
| SensorData | Contains properties needed for storing sensor data table and cold storage |

# System monitors incoming messages

# 

## Description

Messages received by the IoT Hub and process by Feed Manager will be sent to this call chain for analysis. The call chain needs to produce alarms in real time and trigger notifications when needed.

## Component Roles

* Analysis Manager
  + Manages the workflow of analyzing events and sending notifications
* Alarming Engine
  + Retrieves alarm settings and returns the properly formatted settings
* Settings Accessor
  + Returns the settings data for the event type that was received
* Customer Accessor
  + Accesses customer notification settings for alarm types and subscriptions
* Events Accessor
  + Store the event information
* Notification Manager
  + Manages the workflow of sending configured notifications to the proper clients

## Use Case Sequence

1. Analysis Manager receives new message
2. Analysis Manager calls Alarming engine
3. Alarming engine retrieves settings from the settings accessor
4. Alarming engine determines if alarm trigger is needed
5. Alarming engine returns alert to Analysis Manager
6. Analysis Manager calls Customer accessor to retrieve subscription data based on alarm type
7. Analysis Manager calls Event accessor to save event data
8. Analysis Manger sends message to queue for next process in the chain – If Needed

## Services

### Analysis Manager

|  |  |
| --- | --- |
| ReceiveMessage | SensorMessage, Object |
| SendNotification | EventId, long |

### Alarming Engine

|  |  |
| --- | --- |
| FindAlarm | CustomerId, long  DeviceId, String |

### Settings Acessor

|  |  |
| --- | --- |
| LoadDeviceSettings | CustomerId, long  DeviceId, string |

### Customer Acessor

|  |  |
| --- | --- |
| LoadCustomerSettings | CustomerId, long |

### Events Acessor

|  |  |
| --- | --- |
| SaveEventData | CustomerId, long  Event, Object |

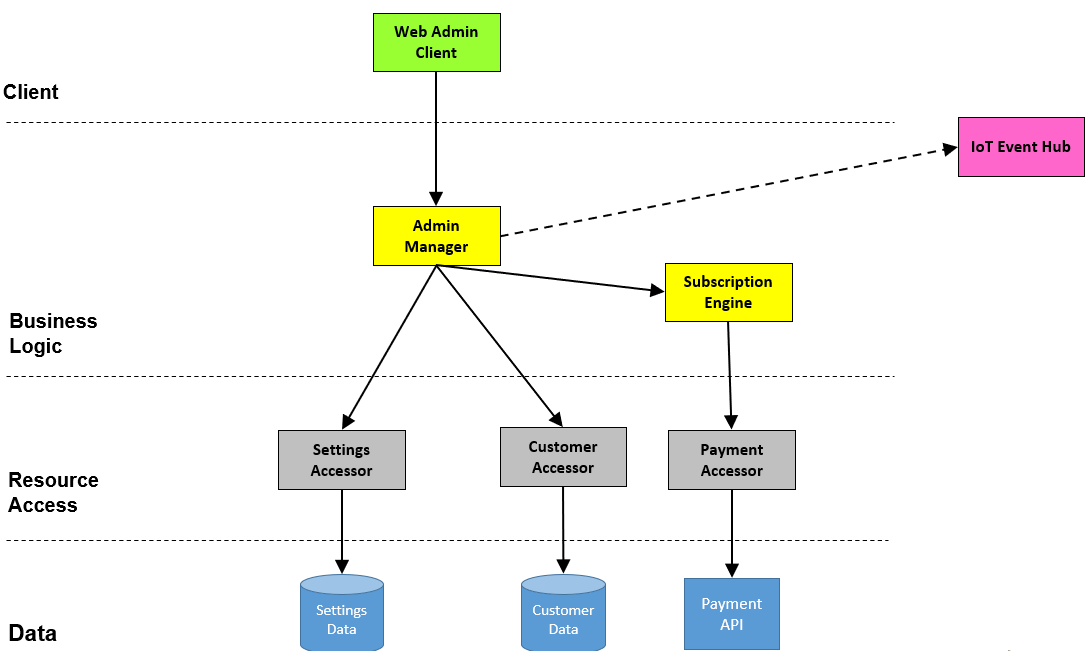
## DTO Definition

List and explain the DTOs needed for this call chain. Listing all properties of the DTO is not required.

### DTO Definition

|  |  |
| --- | --- |
| SensorMessage | Contains properties to define a sensor message pulled from queue |
| DeviceAlertSettings | Contains properties to define the Device Alert Settings |
| CustomerSettings | Contains properties needed for the Customer Subscription and alerts |
| EventData | Contains properties needed for storing event data |

# User configures settings and notifications



## Description

Through the Web Admin client users can manage their settings for their installed system and configure how they want to receive notifications when alarms happen. They can also manage their subscription level and payment information.

## Component Roles

* Web Admin Client
  + User accesses the configuration and notification settings for their installed system
* Admin Manager
  + Manages the workflow of the user configurations, notification settings and subscriptions
* Settings Acessor
  + Manages the CRUD operations of the system settings
* Customer Accessor
  + Manages the CRUD operations of the customer profile settings
* Subscription Engine
  + Manages subscription level and payment information
* Payment Accessor
  + Manages the payment validation for the subscription and calls Payment API
* IoT Event Hub
  + Encapsulates the 3rd party for hub management (adding/removing devices)

## Use Case Sequence

1. User accesses system through Web Admin Client
2. Admin Manager retrieves system settings from Settings accessor
3. Admin Manager retrieves customer data from Customer accessor
4. Admin Manager calls Subscription Engine to validate client subscriptions and payment info
5. Subscription engine calls Payment Accessor to validate payment information through payment API
6. User makes changes through Web Admin Client to system settings
7. Admin Manager updates system settings by calling Settings Accessor
8. Admin Manager updates customer data settings by calling Customer Accessor
9. Admin Manager calls Subscription engine for Payment updates
10. Subscription engine calls Payment Accessor to validate schedule payment information through payment API
11. Admin manager calls IoT hub to manage client devices (add, remove)

## Services

### Admin Manager

|  |  |
| --- | --- |
| LoadSystemProfile | ClientId, long |

### Settings Accessor

|  |  |
| --- | --- |
| LoadClientDeviceSettings | ClientId, long |
| ManageDeviceSettings | Settings, Object  OperationType, Enum |

### Client Accessor

|  |  |
| --- | --- |
| LoadClientData | ClientId, long |
| ManageClientSettings | OperationType, Enum |

### Subscription Engine

|  |  |
| --- | --- |
| ValidateSubscriptions | ClientId, long  Subscription, Object |
| ValidatePayment | ClientId, long  Payment, Object |
| AddPayment | ClientId, long  Payment, Object |

## DTO Definition

List and explain the DTOs needed for this call chain. Listing all properties of the DTO is not required.

### DTO Definition

|  |  |
| --- | --- |
| SystemProfile | Contains properties to define the client Device(s), profile, notification settings and subscription/payment information |
| DeviceSettings | Contains properties to define the Device Settings |
| CustomerSettings | Contains properties needed for the Customer profile and notification settings |
| Subscription | Contains properties to define the subscription information |
| Payment | Contains properties to define the payment information |