

GAO IoT System

For Cloud Native Enterprise Scale IoT Solutions & Local Server-Based Solutions

GAO RFID Inc has developed an RFID System version 3 more than 10 years ago. The system is based on the relational database and Microsoft C# Dot Net technologies. The system has been deployed in more than 4000 companies and government agencies around the world on the cloud and on-premises environment.

GAO RFID Version 3 has evolved into a solid platform for BLE, UHF RFID, HF RFID, LF RFID and active RFID-based systems such as people tracking, asset tracking, access control, parking control, work-inprocess (WIP), fleet management, traceability in a wide range of industries, including construction, manufacturing, healthcare, oil and gas, agriculture, retail, transportation and automotive. GAO RFID Version 3 continues to be a popular choice of software platform for our customers.

In order to provide the market with more efficient, lower cost, and faster response to the customer requirements, GAO IoT system (version 4) is an enhancement of the GAO RFID version 3 system. It is also called GAO BLE RFID IoT Engine. The system is developed with the latest Web 2 technology to provide solutions for RFID industries, integrated with recent development of BLE technologies, HTTP and TCP/IP based RFID readers, handheld devices, Active and passive RFID tags. GAO IoTs provides a RFID solution that is extendable to meet new business requirements and technology changes, to enable system expansion vertically and horizontally.

Technologically, GAO IoT system has the following highlights:

- It is based on the NoSQL on the data layer. Replaced previously structured database and data schema with NoSQL based data model. Fixed structured database solution on the bottom of the data layer restricts the system changes and can not handle the requirement for dynamic data schema of different technical and business use cases.
- It is an application-based web GUI frontend. Replaced previously page-based GUI frontend with Web 2.0 based solution single page application, SPA based GUI frontend has been adopted by the industries and implemented for most of the systems today.
- On the server side, it is based on the Java Spring Boot platform. Replaced previously Windows .Net Platform with Java cross platform solution.
- The mobile app is based on the Flutter/Dart platform which provides common code base for iOS iPhone and Android devices.

System key features:

- 1) The system is extendable and configurable for business modules and reader devices and various operation models without touching core system code/architecture, except to add new code to meet new business requirements and hardware/device interface (device driver, etc.). The design gaol is to make a system that can communicate with tags without deeply coupled with business logics and able to interpreter the various activities based on the configurable business modules as well pluggable devices.
- 2) The system is based on the recent (in the last 10 years) software advancement and provides a solution that is based on HTTP2, HTML5, TypeScript/Angular, NoSQL to address the challenges of the dynamic and distributed/cloud nature of the business and technology. The system can be scalable on the internet, responsive to different devices, interactive to end users and easy to deploy to customer preferred environment.
- 3) The system reuses as much as possible previously version 3 code by starting with device drivers and business logic modules to reduce the time to market and business risks.
- 4) Cross platform support. In the pass decade, we see a growing demand for cloud based solution which mostly is based on the Linux hosting environment. We also see the demand for Windows based deployment for customers who want to host the system in their own environment. For version 4, we provide a system that can support multiple platforms, such as Linux and Windows.
- 5) Simplified deployment. The version 3 system middleware is no longer required. The system combines web services and middleware functions into a single server.
- 6) Horizontal scalability. We have seen the requirement for distributed system with multiple sites deployment with multiple servers and data synchronization for multi-server environment. Also, we have few projects implemented data synchronization on the application level. IoTs version 4 uses data layer synchronization with distributed database technology - primary/secondary database replication technology via VPN or on the cloud.
- 7) Dynamic data schema for tracking entities, location types, readers, and tag attributes. This feature provides system the ability handles dynamic business data and technical properties, such as reader configuration, location property, tracking entity property, etc.

System Architecture

The following system context diagram provides a high-level illustration of the system components and their couplings at runtime. The components connectivity could be on the LAN, Internet and VPN depending on the system performance, data security and complexity. Many enterprises have developed their readers with direct LAN connections with GAO IoT server which provides high system performance, reliability and security while many other enterprises deployed the system on the cloud with fixed readers connect to GAO IoT servers on the cloud environment.

GAO IoT system (v4) includes GUI, Web server, IoT Engine, embedded reader application and Handheld Reader application. The system data layer is based on the NoSQL which enables the system with capability to handle dynamic data schema and semi-structured data schema, it simplifies and relives the system from direct coupling with fixed data schema in the traditional SQL and avoid code changes for different data schema required for various and unknow business logics.



GAO IoT system context diagram

IoT Web Server is a Java Spring Boot based service that provides runtime interfaces for RFID fixed readers, business modules and data synchronization, this module is directly linked with reader drivers and business modules in the runtime. The web server connects to database server either in the LAN or VPN environment. The component supports REST APIs for system administration related objects, such as users, devices, locations, tags, managed objects (people, assets), report query data and Data Synchronization APIs. The component provides local data for embedded and handheld reader devices with the latest required information such as location information, reader configuration or user data.

GAO IoTs user interface is based on Google Angular. The widely acceptance of Angular based web user interface has become the industry standard to evaluate web applications. Beside providing a responsive user interface, the use of Angular framework can detect errors in compile-time, as TypeScript is a statically typed language, instead of runtime like in JavaScript, this feature will reduce the risk of runtime errors and provides the stability and quality for the new system.

The GUI component communicates with IoTs REST APIs on the enterprise network via Internet or LAN. The component enables system administrators to manage RFID server, monitor system activities, users, configure RFID devices, tags, and access system reports from web browsers. The user interface component provides system configuration, managing system users, RFID tags, deployment locations, reader and antenna configurations.



Login Panel

RFID fixed readers, handheld devices and BLE gateway readers are running on different platforms, such as Android. iOS, etc. – today, most of the new devices can be deployed with HTTP protocol to communicate with the server. Some of the RFID readers are based on the TCP/IP protocol, such as GAO SKU 223026 reader. Server implements TCP based drivers based the current version 3 .NET code. Every TCP based devices have their specific modules deployed (enabled by configuration) in the Version 4 server.

In order to communicate with IoT Server, RFID readers require device drivers to communicate with RFID web server REST APIs. On the IoT server, a device specific adapter may be deployed to process messages from the RFID reader. The system should be able to configure/plugin RFID reader adapters without code changes.

We refer the software that installed on the RFID reader as device driver and the software deployed on the IoT server as the adapter. Between device drivers and adapters, they understand HTTP messages from RFID readers and supports features that are specific to the reader, e.g. some RFID readers support GPS information, as long as the reader has been configured with a URL for the IoT server.

GAO IoTs Dashb	oard Logout								
Dashboard				RFID	Device Managemer	nt ×			
Manage Devices									_
🕏 Manage Tags					New Device				
Manage Location		Name	Model	Antenna	MAC Address	IP Address	FQDN	Registration Date	
Tracking Entities		Reader at the front door	RFID UHF Model 1	4 🌣	33-15-5D-BC-0E-CC			Jan 12, 2023, 7:00:00 PM	x
Schedules		R2	BLE Gateway	0 🌣	12-75-5D-BC-0E-CC			Dec 17, 2022, 11:11:35 AM	x
Reports 🗸		UHF Fixed Reader in building 2	RFID UHF Model 1	4 🌣	10-15-5D-BC-0E-C101	12312312321	office.gaorfid.com	Dec 23, 2022, 1:42:17 PM	×
System Configuration		Location 1, floor 3, door 101 Reader	Speedway 20232	2 🌣	00-15-5D-BC-0E-CC	sfaf		Jan 27, 2023, 7:43:44 AM	x
→Logout		Test Reader for the customer at NYC	RFID UHF Model 2	4 🌣	00-15-5D-BC-0E-C1	192.123.123.101		Jan 27, 2023, 11:46:39 AM	x

RFID Reader Panel

RFID Device Management $ imes$								
Device: Reader at the front door - Model: RFID UHF Model 1								
Anten	nas							Select a location and assign to an antenna
ID	Status	Con	trol Mode	GOP Port	Transmit Power	Sensitivity	Location	
ant1	Enabl	led	0	-1	30	-70	Assigned	> Markville Mall 🗮
ant2	Enabl	led	0	-1	30	-70	Assigned	somewhere in New York
ant3	Enabl	led	0	-1	30	-70	Assigned	
ant4	- Each	lod	0	-1	30	-70	Assigned	> Ikea center
	Enabl	lea					Assigned	3 Time Square
				Back	Save			GUNTUR 📶
								india
								> wqueiwuqeioq

Reader Antenna Configuration

Selected RFID drivers:

Name	URL	Communication Protocol
BLE Gateway	<u>3G/4G/LTE/Wi-Fi/Ethernet</u>	HTTP with basic authentication,
217030/217031	BLE Active RFID Outdoor	supports data backlog forwarding
	Gateway (gaorfid.com)	in case of communication failures.
UHF Fixed Reader	860 - 960 MHz for Multi-	The reader is Android based.
236020	Region RFID Reader Writer	Embedded code on the reader to
	(gaorfid.com)	forward RFID messages to GAO IoT
		server.
		Both driver and adapter are
		required

HF Fixed Reader	13.56 MHz RFID Reader w/	TCP only. Server needs to open a
223026	LED Buzzer GAO RFID Inc.	dedicated TCP port, default port
		2168.
Handheld Reader	Pistol Grip All-in-One	
223034	Handheld RFID Reader Writer	
	(gaorfid.com)	
GAO BPN App	No web page yet	BLE based Position and Navigation
		mobile app communicates with cell
		phone device and GAO IoT server
		when BLE beacons are scanned

The IoT Web Server is implemented with Java Spring Boot Framework. The component provides runtime interfaces for RFID readers (for readers that support HTTP/HTTPS), data synchronization to communicate with embedded on reader application and handheld readers, services for business modules and RFID reader drivers that support device specific messages to process HTTP and TCP messages.

On the server side, the system is partitioned with various controllers (RestController), services and data models. For each entity, such as RFID Reader, Tag, Location, System user. A controller will responsible to handle specific REST APIs that are required for the entity.

When the system started, it will load the reader driver code and business logic modules based on the system configuration. The subsystem provides server-side features for web user interface tier, REST APIs for RFID readers and third-party components integration. It is a Spring Boot based server-side component and could be deployed in Tomcat Servlet engine or other web containers in Windows and/or Linux environment.

The server component has direct connections to database server that persists system configuration data and operational configuration and activity data, such as RFID reader data that are transmitted via HTTP/HTTP messages from RFID readers, device specific configuration with driver code and parameters based on the deployed reader devices in the system.

The data layer is managed by MongoDB for structured, semi-structured and non structured data. IoT Web Server provides REST APIs and for frontend web user interface component, third party integration and mobile device communications. It uses SMTP server for out bond email messages.

The IoT Web Server has the following major packages:

Controller - process request from the web. The package includes controller class for user login controller, reader management controller, tag management controller, location management controller, etc. Controller should extend Spring REST Controller interface and supports required CRUD APIs for the entity.

Service – component for various business modules, such as access control, ABC company asset tracking, etc.

Device driver – component for device driver specific code, could be TCP based or HTTP based driver

Supports CRUD operations packages. System User, Location Type and Location, Tag, Parameter, RFID Reader Model and RFID Reader and IoT Tracking Object and its entities. For semi structured data, system should provide dynamic data schema for RFID tracking objects (such as location, people, asset, etc.), metadata schema supports the following data types:

- String
- Number (real and integer)
- Date
- Array / Selection
- Boolean

The server WAR file could be deployed in the same Tomcat server with frontend angular component. The relative web path is ./rfid, REST APIs will be started with ./rfid/api and the web frontend path is /iots

Client Applications	Web Browser Interface	Mobile Apps	3 rd Party
Client-side framework	Angular/REST Client	REST Client	Applications

Subsystems	IoT Web Server	IoT Engine	Reader Device Drivers	Business Modules

Foundation LayerSQL Data ManagerData Classes	NoSQL Manager Data classes	Report Manager System Audit Security Manager	REST Gateway	SMTP API
---	-------------------------------	--	--------------	-------------

Server-Side	Java	Java EE	Spring Boot with Spring Security (REST, JPA, MVC) and NoSQL Driver, SMTP API, etc.	
Frameworks	API	API		
System platforms	Windows or Linux operating systems			

GAO IoT System Components and Layers

IOT TCP Service is deployed in the same Spring Boot server container with GAO IoTs server environment. TCP based fixed RFID reader uses TCP service adapter. On the server side, every TCP service component has to open a dedicated inbound TCP port in order to communicate with the server.

A HTTP based reader communicates with the server via HTTP protocol through the device adapter with its specific messages and format (e.g. JSON). A reader device adapter is an abstraction layer with common APIs for active and passive RFID readers that need to connect to the RFID web server during the runtime. A vender specific reader can implement a reader abstract class to interact with other RFID components. Device driver will pass the received RFID messages to the business module.

Handheld Component provides user interface based on business specific specification. Each handheld device will have a local database with required information for supported business modules. The local

database is updated with a configured server with wireless or USB port. The device stores RFID transactions to the handheld database and uploads transactions to the RFID server whenever the device is connected to the server.

Business Modules

Business Module provides a business logic abstraction layer that enables various business specific modules to plug into the RFID system at runtime by implementing a business module abstract class. Business module translates RFID reader activity data to the GAO IoT server with business logic specific information such as asset tracking, access control and device position use cases.

GAO BPN system

GAO BPN (Bluetooth based Position and Navigation) System provides position and navigation solutions based on the Bluetooth Low Energy and cellphone technologies. The position of a user/asset identified by the cellphone device is determined by the beacon positions in the location map. The device position in the location is determined by the scanned BLE messages and measured signal strength, aka received signal strength indicator (RSSI). The business module supports 1) consumers as a position and location mobile application which allows users to locate where they are and help them navigate to the targeted point of interest. 2) event tracking system which allows participants to register to events and their activities will be reported to the business owners. Event management requirements are a subset of the GAO PBN system, which has features for event configuration, assign event to locations and event report.

GAO BLE system can be used to provide indoor location positioning, navigation guidance to users on smartphone devices and user position tracking for events and trade shows. The system visualizes indoor floor plans and points of Interests with customizable images for floor plans, points of interests and direction pointers.

System administrators can manage and configure the system, view user activity report on the cloud with web browser interface.

GAO BPN system is designed to track device position, such as a location for a specific event and navigation services with Bluetooth Low Energy technology with BLE beacons deployed in the targeted locations. A BLE beacon's position in the location will be used to estimate the user position, i.e., the cellphone device position in the current location. System uses beacons' position information and received beacon signal to calculate the device physical coordinate on the location and display the user location on the location floor plan.



GAO BPN System Context

Beacon system timer will trigger broadcast every 20 (or configuration with specific environment) ms with MAC address and RSSI. BLE broadcast can be received by mobile device with Bluetooth enabled.

End users:

- Download and install GAO BPN on smartphone a one off activity the installation package contains a configuration with the server URL
- User registration
- Select location select a location from the location list panel. A system may be configured with many locations plans with map images
- Display position start the app and show the current position in the location map.
- User gets location specific information, such as map, name, details, zoom in with lower level on the smartphone.
- Navigate display a navigation map with directions.
- Every time the device scans the beacon messages, it will send a short message to the server with the device ID, beacon ID and RSSI a feature required for event tracking.
- Logout system

System administrator:

- Location management, event Management, populate location parameters, building, floor, set images for location.
- Beacon management by assigning BLE beacons to location and event, broadcast interval, coordinates, relative position in the location map.
- Read reports, Data Analytics.

View business reports with registered user information based on the device ID, where and when

 this could be used for event user tracking, attendance, event participation, administrator
 should be able to view event report, such as how many users been to an event, with name,
 date/time, etc.



GAO BPN Main Panel

GAO BPN mobile app allows users to login the system either with a registered user id (mobile user has mobile user role) or login with social media account such as Google ID with OAuth 2.0 Authorization framework with the following roles:

- Resource Owner: the end-user.
- Resource Server: IoTs system.
- Client: GAO BPN App.
- Authorization Server: Google OAuth server or other social media system.

If the user clicking on the Google Login button on the device, BPN app on the mobile device will send authenticated user ID Token from the Google OAuth server to the IoTs system which will conduct token validation. IoTs Server sends JWT token to the BPN Mobile device upon successfully ID token validation.

GAO BPN mobile app interacts with BLE tags and provides information location specific information to the user. For event management deployment, the system also provides user event participation report with where, when and duration for event organizers.

The main panel provides a list view of locations that been deployed for event tracking or location navigation. The user may use map view to see the events that are marked on the geo map and use navigation function to find the way to the specific event location. BPN App provides navigation guidance on the Map View panel by using Google Map API which provides zoom levels with world, continent, city, streets and buildings.



GAO BPN Map View

Floor map-based navigation feature can help user to locate a specific store in the mall or event booth in the event with a direction polyline which provides the shortest route based on the floor map routes configuration. The system calculates the route is similar as the Google Map API with the following steps:

- 1) Get the source and target point. The source point should be provided by device which sense the BLE beacon location and the target point should be decided by use, where the user wants to go.
- Get route between starting and target points based on the shortest distance, the route is represented by a set points on the map or in mathematical term, the shortest path p = {V0, V1,...Vk}. A special case could only have the starting and target points,¹ or there is no solution to get target point from the source point.
- 3) Draw the polyline with the route points

In order to enable floor map-based navigation, the floor map image has to be configured with points and routes that can be accessed.

¹ The implementation is based on the <u>Dijkstra's algorithm</u>.



Location Map Configuration Browser Interface

Access Control System

This business module could be used for location access control with a HF or LF RFID fixed readers. The system can control the gate access by deploying RFID readers in the entry and exit points. Some readers may control the gate directly with GPO ports with connections to the gates.

System administrators can assign various location control schedule to different user groups based on the entity group. On the RFID device itself, it can be configured with LED and relay controls and system can directly send TCP/IP messages to RFID readers with control commands, such as turn on the gate, beep for alert event, show red or green LED lights, etc.

If a location needs to be secured by the access control schedule(s), system administrator needs to assign the location to a location group which associates with access schedule.

System will assume a location has only one location group, if there is no location group assigned, the location should accessible to everyone.

An entity may be associated an entity group, for example, a user could belong to Security Guard group, or Resident Group, etc.

Name * Grade One Student	Group Type Entity Group
Description * The kids at Grade 1	
Location Group . restricted area	Weekly Schedule Weekly Schedule for Location 1-3
Access Schedules weekly Build 1 - Floor 1- 10 [Calendar Date Control] restricted area [Weekly Schedule for Location 1-3]	Sun Open: 00:00:00 - 23:59 Closed: 17:02 - 23:59 ✓ Enter _ Exit Enter ✓ Exit
	Tue Open: 00:00:00 - 23:59 Wed
	Thu
	Fri Closed: 00:00:00 - 23:59 ✓ Enter ✓ Exit

Entity Group Management Panel

Access control is enabled by assigning entity group to location group with a specific control schedule with permissions for entering and/or exiting. A group may associate with multiple location groups.

One of the requirements for access control is to tracking the people time attendance or asset location time. From the Assessment Management, manager and system administrator may review the daily hours for employees or to adjust the missing/wrong time attendance.

	Assessment Period	Choose a date 2/26/2023 - 2/26/2023	🗉 G 🗙	
Name	✓ Search	items per page: 10	▼ 1 - 10 of 11	< >
2023-02-26 shawn bushe [Hours: 10.	50]			
6 2023-02-26 indian girl [Hours: 8.98]				
2023-02-26 hrdh [Hours: n/a] 0				
2023-02-26 fredshin123123 [Hours: 7	.98]			
2023-02-26 Windows Server 2				
(1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.				
2023-02-26 New Dev				
2023-02-26 Mark [Hours: 16.98]				
	lours: n/a] 🛛 🛛 🛛 🔒			
Event Adjustment				
2023-02-26 04:00:00 rweiurweiruwei ruweior	4234234 Enter			
	s: 10.95]			

Assessment Management

Assessment Management provides entity time-based tracking record with specific period. The module automatically calculates how many hours of each entity been on the premise. The Event Adjustment feature can be used to add missing event transactions in case of hardware failures.

Asset/People Tracking System

Asset Tracking system provides functions to digitally track the location of assets that have been physically tagged. The system stores asset information and inventory data, and can send alert when asset stock is low or important assets are moved.

Some assets can only be allocated to specific locations and some assets can not been moved to certain locations.



Version 3 Asset Tracking Panel

System Deployment

For a specific deployment, the system can be configured with a set of supported device drivers and business modules depending on the client requirement. The system deployment property is defined in the backend server in the application.properties file (Spring Boot default property file):

For example:

business.modules=com.gaorfid.busmodule.AccessControl; com.gaorfid.busmodule.AssetTracking

device.drivers=com.gaorfid.driver.Gao217031; com.gaorfid.driver.Gao232023

The configuration process will also provide device and business logic specific parameters for the targeted deployment with reader model (s) and business module (s). There are two types of system configurations:

1) the system metadate (a list attributes with different data types) for RFID device models, location types and reader models. One reader model could be configurated with antenna (s). The location parameters and tracking object properties are business logic specific.

2) configuration for specific object entities, for example, a location, a reader, a tracking object for a specific asset. For example, a deployed reader is an instance of BLE Gateway 1234 with MAC Address 00-42-38-D3-B9-40.

3) System parameters for system roles, access control actions, system title, logo, etc. For a RFID based access control application, the system may be configuration with various access control actions, such as "lock" and "unlock" for some time ranges for specific locations.

The following parameter names are reserved:

- SYSTEM_TITLE and SYSTEM_SUB_TITLE used for system title on the home page.
- SYSTEM_ROLES used for system to define system user role(s), by default, a new system user has no role, system administrators (with ADMIN role) can assign role(s) to the user.
- ACCESS_CONTROL_PERMISSIONS for example, open, closed, lock, unlock, etc. Access Control Permission parameters are used to define access schedule based on the date and time.
- Reader and Antenna Control Parameter SCAN, for tracking if entity been scanned in the location; Enter, to indicate if the reader or antenna is positioned as the entry point; Exit, to indicate if the reader or antenna is positioned as the exit point.

4) Scheduled Tasks² – system can be configured with a list of scheduled tasks. For example, daily.reports property is reserved for system to send end of day report to the registered emails.

One of the non-functional requirements, the web interface should be mobile friendly. For Version 4 release 1, we are focused mainly on the major use cases, but it will have a mobile friendly interface as one the key features for the release 2.

Subsystem	Technology	IDE/Tools	Runtime environment
Web User Interface	Angular Framework	Visual Studio Code	Chrome and Edge
IoT Web server	Java Servlet/Spring	Eclipse with STS bundle,	Windows and Linux
	Framework/Spring	Tomcat in production or	
	Boot	microservice	
IoT Engine	Java Console	Eclipse	Windows and Linux
	Application		
SQL Server	MS SQL Server Express	SQL Management Studio	Windows and Linux
NoSQL Server	MongoDB	MongoDB Compass	Windows and Linux
Version Control	Subversion or GitHub?	should talk about this	Cloud?
Bug Tracking	Bugzilla	N/A	Cloud

We suggest the following development environment for the system:

² In application.property file, the scheduled tasks is defined in schedule.cron.expression, e.g. 0 0 1 * * ? , job start at 1:00 AM daily.

The system dashboard component provides entry points with a common system menu (sub menu contains system configuration, reader configuration, location configuration, tag configuration, user management, tracking object configuration, etc.).

System Configuration allows system administrators to configure the system for application domain specific object types, metadata and parameters.

Entity Type interface for creating RFID Object attribute/metadata:

≡ GAO loTs Dashb	oard Logout			
Dashboard Manage Devices	t	estuser ogged in as [ROLE_USER,ROLE_ADMIN]		
✗ Manage Tags ➡ Manage Location ✗ Tracking Entities		Manage Devices Manage fixed and handheld devices	Ŷ	Manage Tags *
Schedules Reports ✓ System Configuration		Manage Locations Configurate location type & property	ā	Tracking Entities Configure entities, assets and users
[→Logout		Schedule Management Setup schedules for access control		Daily Reports I. View daily and history reports
		Real Time Reports Time and location based reports	Ŭ	System Configurations System setup and configurations

GAO IoTs Dashboard

Operational functions (such as device, tag, location management, etc.) allow user to manage defined object types and properties to generate concreate objects, such as tags, readers, assets and locations.

System Settings-> Email and SMTP Settings

Should support SMTP server configuration for out bond emails,

System Setting for System Parameters, such as system title for specific project and client, log images, etc.

System Settings->Tracking Object, Reader Model, Location Type

Provide interface for administrator to configure RFID devices, the system tracking object and attributes. For example, if the target system will be used as an asset tracking system, the component will be able to

define the asset properties (metadata) based on the customer requirement. The specific attributes will be used later to populate asset object for tracking.

Location Management

This component provides structured/layered and grouped location such as a physical location with address, a building, a floor with floor plan image, a gate/door are floor or building, a room, etc. System provides business specific metadata for location types. For example, a deployment may require location been structured with building, floor and door location object types. A specific location type may be associated with a set of data attributes.

Description			11
First Name	Text	~	Required >
Last Name	Text	~	Required >
Department	Text	~	Required >
Job Title	Text	~	Required >
Field name	Text	~	Required >
Add Row Save	Text Date Number Member Selections		Back

Tracking Entity Type Configuration

RFID Reader Management

RFID reader models have many similar properties such as MAC address, IP, deployed location and every RFID reader model will have its unique specific configuration parameters, some readers may be configured with two antennas and some readers may have four antennas.

Version 4 system will start with GAO 217031 BLE Gateway (Ref 217031-UserManual), GAO 223026 and GAO 236020 UHF gen 2 passive RFID reader to make a generic RFID Cloud based reader management user interface.

Basically, the RFID reader management module provides user interface for RFID reader configuration and location association, such as a reader deployed at building 1, floor 3 at gate 3 or room 102. Also, should consider the antennas, for example, a fixed reader may connect to 8 antennas and the system should identify the locations by antenna deployment.

RFID Tag management

Tag in the system could be a regular RFID tag or a BLE beacon. This module provides user interface for RFID tag registration and tag association with the target object, for example a tag may bind with a person, employee or an asset. Some use cases may also require a tag to bind with a specific location, for example, the application to use BLE beacon to track people position, i.e., GAO BPN system where the BLE beacons will be deployed in the points of interests.

A tag can only be assigned to a single entity at a time, and it can be reassigned. System tracks tag assignments.

System report

The system provides a set of fixed reports, such as system access report, tag assignment report, tag scan report, etc. For domain and application specific requirement, we should use the version 3 code as the template and migrate to the system.

For example, the location visitors report, provides information about user visiting data based on the event and location for the real time or with specific period of times.

Event Monitor

The IoTs Event Monitor is designed for monitoring real time event by selecting a location. The system updates RFID reader and tag event with location coordinates.



System Realtime Activity

The solid line displays the current period data and the dashed line displays previous data. By clicking on the REAL-TIME REPORT, the system will display report the current location with the current activities (Current means now to now - 5 minutes).

Access Control Schedule Configuration

System uses access control schedule to send system alert messages, trigger system-controlled activities and daily report to administrators and/or registered email addresses. Schedule may also be used for time-based access control to some locations (e.g. main entrance at building A), some use cases need to grant user access based on day time, evening, and weekend, etc.

There two kinds of access control schedules: 1) Weekly Schedule, system can configure access actions (parameters specified in the system parameter for ACCESS_CONTROL_ACTIONS), such as weeks with open, closes, no access, and weekend control schedules, etc. and 2) Calendar Schedule which allows administrators to configure access permissions based on specific calendar date.

= OAQ IoTs Dashboard Logout							
Dashboard	-						Edit or view schedules
Manage Devices	Manage	Create access control calendar or weekly schedule danage Calendar Manage Weekly Schedule					Calendar Schedules Weekly Schedules
⊁ Manage Tags		Time range action	ns (00:00AM-11:59PM)				Building1
Hanage Location	0	Area B - building 1	- 10				
2 Tracking Entities		00:00AM	06:00AM	12:00PM	06:00PM	11:59P	weqw
Schedules	Mon	lock 00:00:00 - 2	2:59 unlock 08:00 - 15		2-22:50		third test
Reports \lor	Tur	1000 00.00.00 - 2			0-20.07		second test
System Configuration	Tue						Area B - building 1- 10
[→Logout	wed						
	Thu						
	Fri	closed 00:00:00	- 23:59				

Sample Weekly Access Control Schedule

= GAC IoTs Dashboard Logout						
Dashboard	Access control calendar and weekly schedule					
Manage Devices	Weekly Schedule Manage Calendar					
⊁ Manage Tags						
🖶 Manage Location	Choose a date* 1/24/2023 Clear Save Schedule	Calendar Schedules				
2 Tracking Entities	5	Monday, January 9, 2023 test				
Reports v	Stratul lime* Access Schedule for Building C29	Sunday, January 1, 2023 i qwqw				
System Configuration	Feer * 12:00 AM 0 11:59 PM 0 Control Action * ▼ €	Wednesday, January 11, 2023 qweqweqwqw				
[→Logout		Tuesday, January 17, 2023 wewqeq				
	00:00 - 23:59 unlock X	Tuesday, January 24, 2023 Access Schedule for Building C29				
	18:00 - 23:591 closed X	Thursday, January 26, 2023 Building				
		Saturday, February 25, 2023 rewrewewrwerer				

Sample Calendar Access Control Schedule

System Security

System is secured with user authentication and authorization based on user role (s). By default, the system supports administrator, power user and report viewer roles.

On the user interface, the system menu provides access to two major system features, 1) system settings, accessible for administrator role, and operational functions accessible for administrator and power user roles and system reports, accessible for administrator, power user and report user roles.

The system provides an implementation of Spring SecurityProvider and use Bcrypt³ Hashing Password Encoder for password validation. Hashed passwords will be saved in the database. After use has been successfully authenticated, system will return a signed time sensitive Java Web Token (JWT) to the client agent (could be browser or mobile app), the client will communicate with the server by including the JWT in the request header which will be validated by the server.

System authorization is role based and the system uses SpringSecurityFilterChain to set permission based on the URL, for example, antMatchers("/Administration").hasRole("Administrator"), antMatchers("/api/tags").hasAnyRole("Administrator","User","Tag"), antMatchers("/Help","Home").permitAll()

Be default, system has ADMIN, USER, DEVICE, REPORT, REGISTRATION, and OAUTH roles:

ADMIN – has all the permissions

DEVICE – permission to post RFID tag scan data to the server

USER – has all the permissions, RFID Entity Management, Location Management, Tag Management, etc. except system user management, system parameter management, license management and SMTP server configuration

REPORT – can access various system reports

REGISTRATION – can manage tag registration, a use case could the security office employees to assign tag to visitors, contractors, etc.

OAUTH – social media users who login with OAUTH server, e.g., login from the BPN App with Google Login, the role has read only permission for locations.

³ <u>https://en.wikipedia.org/wiki/Bcrypt</u>



Appendix: System Deployment Environment

	Platform	Hard drive	RAM	CPU	Third party
Version 4	Linux and	5 Gigabytes	Min 8	2 Core	MongoDB 4.x
	Windows 2018		Gigabytes	CPUs	community edition
	or newer				(free)
					JAVA Runtime
					Apache Tomcat
Version 3	Windows 2018	5 Gigabytes	Min 8	2 Core	MS SQL Express or
	or newer		Gigabytes	CPUs	Server
					IIS
BPN App	iOS and	2 Megabyte			Google APIs
	Android	on the device			Flutter/Dart

Copyright @ 2023 GAO RFID Inc. http://www.gaorfid.com