Key Algorithm List

This page includes a list of key algorithms/ functions that are currently in use for the latest release of the product, which requires extra explanations for its logic.

Algorithm / Function Name	Location	Used For	Description
fetchAttributes()	AQ-Redback/src/AquaTerra/AquaTerra/Views/ProfileView.swift	Retrieve user attribute from AWS Cognito	Import AWSMobileClient to call the built-in function: *getUserAttributes()*
editSensor()	AQ-Redback/src/AquaTerra/AquaTerra/Models/SensorListApi.swift	Dashboard receives response date	Create the sensorUpdateData dictionary and Convert sensorUpdateData to JSON data
isWarningPresented	AQ-Redback/src/AquaTerra/AquaTerra/ViewModels/DashboardViewModel.swift	moistures alarm	Trigger alarm mechanism
updateChart()	AQ-Redback/src/AquaTerra/AquaTerra/ViewModels/DashboardViewModel.swift	Update chart data	Create corresponding DateMoistureChartItem according to different humidity depths (50, 100, 150), temperature, and battery.
makeUIView(context: Context) -> MKMapView	AQ-Redback/src/AquaTerra/AquaTerra/MainTab/Views/GRegistration/GRMapView.swift	Initialization settings for MapUI	Set the initial map area, map zoom, and add a tap gesture recognizer
registerField	AQ-Redback/src/AquaTerra/AquaTerra/Views/Farm&Fields/ViewModels/FarmsViewModel.swift	Registering new field in the backend	It first checks if the farm exists. If it doesn't exist, it will register the farm first and then the field.
performRequest	AQ-Redback/src/AquaTerra/AquaTerra/Views/Farm&Fields/ViewModels/FarmNetwork.swift	Execute HTTP request	The core of the FarmNetwork class as all other API call methods rely on performRequest to complete the actual network request.
copyFromZone(_ zone: Zone) -> ZoneEditable	/Users/youzhou/Desktop/Project/SPRINT3/AQ-Redback/src/AquaTerra/AquaTerra/Views/Field&Zones/Model/Zone.swift	Create a new ZoneEditable object from an existing Zone object	When the user wants to edit the properties of a certain area, the original data is first copied to an editable structure, then modified on this structure, and finally the modified data is Sent back to the server.
createSensor()	AQ-Redback/src/AquaTerra/AquaTerra/Models/SensorListApi.swift	Create V2 Sensor	It can create a V2 sensor by passing the sensorID, fieldID, and coordinate.
submitSensorV1()	AQ-Redback/src/AquaTerra/AquaTerra/Models/SensorListApi.swift	Create V1 Sensor	It can create a V1 sensor by passing the sensorId, gatewayId, fieldId and coordinate.
fetchGatewaySensors()	AQ-Redback/src/AquaTerra/AquaTerra/Models/SensorListApi.swift	Fetch Gateway Sensors	It can get the Sensors with corresponding gatewayId by passing the gatewayIds.
checkGatewayAPI()	AQ-Redback/src/AquaTerra/AquaTerra/Models/SensorListApi.swift	Check gateway status of the field	Before creating a V1 sensor for specific fields, it will check the gateway status first.