

Ανοιχτή υποδομή Internet of Things για online υπηρεσίες περιβάλλοντος Open Internet of Things infrastructure for online environmental services, Open ELIoT

#### ΤΙΤΛΟΣ ΠΑΡΑΔΟΤΕΟΥ

2.5

" Δεύτερη έκδοση διαδικτυακής πλατφόρμας συλλογής, επεξεργασίας και οπτικοποίησης δεδομένων"

ΚΩΔΙΚΟΣ ΕΡΓΟΥ

Т1ЕАК-01613



Αθήνα, 2019

Οι βιβλιογραφικές αναφορές στις εργασίες της παρούσας έκθεσης παρακαλούμε να γίνονται σύμφωνα με τον ακόλουθο τρόπο:

Ex Machina<sup>1</sup>2019. Δεύτερη έκδοση διαδικτυακής πλατφόρμας συλλογής, επεξεργασίας και οπτικοποίησης δεδομένων. Ανοιχτή υποδομή Internet of Things για online υπηρεσίες περιβάλλοντος (Open ELIoT) - 22 σελ.

<sup>&</sup>lt;sup>1</sup> Ομάδα Έργου

### 1. Αντικείμενο, Σκοπός και Ομάδα Έργου

#### 1.1. Αντικείμενο και σκοπός της παρούσας μελέτης

Η παρούσα τεχνική έκθεση συμπεριλαμβάνεται στην ενότητα εργασίας "ΠΕ 2: Ανάπτυξη έξυπνων αισθητήρων ΙοΤ και πλατφόρμας συλλογής, επεξεργασίας και οπτικοποίησης περιβαλλοντικών δεδομένων" και αποτελεί το παραδοτέο Π2.5 με τίτλο "Δεύτερη έκδοση διαδικτυακής πλατφόρμας συλλογής, επεξεργασίας και οπτικοποίησης δεδομένων".

#### 1.2. Ομάδα έργου

#### EXM

Βασίλειος Χρυσός, Μηχανικός Περιβάλλοντος, MSc, EXM Εμμανουήλ Νικηφοράκης, Μηχανικός Ηλεκτρονικών Υπολογιστών, MSc, EXM Ευστράτιος Θεοδώρου, Ηλεκτρολόγος Μηχανικός & Μηχανικός Υπολογιστών, MSc, EXM Νικόλαος Τσιλιγκαρίδης, Μηχανικός Βιομηχανικού Σχεδιασμού, EXM

#### ΠΕΡΙΕΧΟΜΕΝΑ

1.	Αντικ	είμενο, Σκοπός και Ομάδα Έργου	3
1.1	<b>1.</b> A	Αντικείμενο και σκοπός της παρούσας μελέτης	3
1.2	2. 0	Ομάδα έργου	3
ПЕР	IEXO	DMENA	4
2.	Εισαγ	/ωγή	5
3.	Μηχα	ανή Κανόνων (rule engine)	5
3.1	1. N	Μηνύματα μηχανής κανόνων (rule engine message)	5
3.2	2. k	<b>Κόμβος κανόνα (rule node)</b>	6
3.3	3. Σ	Συσχέτιση κόμβου κανόνα (rule node relation)	6
3.4	<b>4.</b> A	Αλυσίδα κανόνων (rule chain)	6
3.5	5. I	Ιαραμετροποίηση κόμβων	7
4.	Γραφ	ικά στοιχεία (widgets)	7
<b>4.</b> ]	1. <i>A</i>	λημιουργία γραφικού στοιχείου	7
5.	Δεύτε	ερη έκδοση πίνακα ελέγχου πλατφόρμας1	0
5.1	1. k	Κανόνας υπολογισμού στάθμης νερού1	0
5.2	2. I	Ιίνακας ελέγχου απομακρυσμένης διαχείρισης συσκευής	5
	5.2.1.	Γραφικό στοιχείο επεξεργασίας παραμέτρων συσκευής1	5
	5.2.2.	Πίνακας ελέγχου απομακρυσμένης διαχείρισης1	7
5.2	2.3.	Πίνακας ελέγχου δεδομένων1	8
6.	Аπоθа	ετήριο κώδικα2	2

### 2. Εισαγωγή

Σε προηγούμενο παραδοτέο (Π2.2) έγινε εκτενής περιγραφή των λειτουργικών χαρακτηριστικών της πλατφόρμας δεδομένων του Open ELIoT, καθώς και των διαδικασιών δημιουργίας οντοτήτων (assets, devices), των μεταξύ τους σχέσεων, και πινάκων ελέγχου στην πλατφόρμα. Επιπλέον, παρουσιάστηκε η πρώτη βασική έκδοση πίνακα ελέγχου της πλατφόρμας, που παρουσιάζει σε πραγματικό χρόνο τις τρέχουσες τιμές μετρήσεων των εγκατεστημένων αισθητήρων.

Στο παρόν κείμενο θα αναλυθούν επιπλέον χαρακτηριστικά της πλατφόρμας. Συγκεκριμένα, θα γίνει περιγραφή της διαδικασίας δημιουργίας κανόνων (rules) και ενσωμάτωσής τους στην αλυσίδα κανόνων (rule chain), καθώς και της διαδικασίας δημιουργίας γραφικών στοιχείων (widgets) που προσθέτουν επιπλέον λειτουργικότητα στους πίνακες ελέγχου.

Έπειτα θα παρουσιαστεί η δεύτερη, ολοκληρωμένη έκδοση του πίνακα ελέγχου, που παρουσιάζει, επιπλέον των τρεχουσών τιμών μετρήσεων, αναλυτικά δυναμικά ιστορικά γραφήματα με τα οποία ο χρήστης μπορεί να ανατρέξει σε παρελθοντικές τιμές και να κάνει αναλύσεις.

### 3. Μηχανή Κανόνων (rule engine)

Η μηχανική κανόνων της πλατφόρμας είναι ένα παραμετροποιήσιμο σύστημα για σύνθετη επεξεργασία εισερχομένων μηνυμάτων. Ως μήνυμα νοείται οποιαδήποτε εισερχόμενη επικοινωνία μιας διασυνδεδεμένης συσκευής. Με τη μηχανή κανόνων είναι δυνατό το φιλτράρισμα, ο εμπλουτισμός και ο μετασχηματισμός των εισερχομένων μηνυμάτων που πηγάζουν από συσκευές ΙοΤ και τα σχετιζόμενα με αυτές assets. Επίσης, είναι δυνατό να να πυροδοτηθούν ποικίλες ενέργειες, όπως για παράδειγμα η παραγωγή ειδοποιήσεων (notifications) αν μια τιμή μέτρησης υπερβαίνει ένα προκαθορισμένο όριο, ή επικοινωνία με εξωτερικά τρίτα συστήματα.

### 3.1. Μηνύματα μηχανής κανόνων (rule engine message)

Ένα μήνυμα της μηχανής κανόνων είναι μια σειριοποιήσιμη και αμετάβλητη δομή δεδομένων που αναπαριστά διάφορα μηνύματα που διατρέχουν το σύστημα. Παραδείγματα μηνυμάτων αποτελούν τα παρακάτω:

- Εισερχόμενη τηλεμετρία, αλλαγή χαρακτηριστικού (attribute) ή απομακρυσμένη κλήση διαδικασίας (RPC) μιας συσκευής
- Δραστηριότητα σχετική με τον κύκλο ζωής μιας οντότητας (asset, device): δημιουργία, επεξεργασία, διαγραφή, κλπ
- Δραστηριότητα σχετική με την κατάσταση μιας συσκευής: σύνδεση, αποσύνδεση, ενεργοποίηση, απενεργοποίηση, κλπ
- Άλλες δραστηριότητες του συστήματος

Ένα μήνυμα της μηχανής κανόνων περιέχει την παρακάτω πληροφορία:

- Message ID: μοναδικό αναγνωριστικό, βασισμένο στο χρόνο δημιουργίας του μηνύματος
- Originator: μοναδικό αναγνωριστικό της οντότητας από την οποία προήλθε το μήνυμα (device, asset ή άλλο)
- Type: τύπος μηνύματος, όπως π.χ. "Post telemetry" αν αφορά δεδομένο τηλεμετρίας, κλπ
- Payload: το περιεχόμενο του μηνύματος, κωδικοποιημένο σε μορφή JSON

Metadata: λίστα με ζευγάρια κλειδιών-τιμών με επιπρόσθετα δεδομένα σχετικά με το μήνυμα

## 3.2. Κόμβος κανόνα (rule node)

Ο κόμβος κανόνα είναι η βασική δομική και λογική μονάδα της μηχανής κανόνων, η οποία επεξεργάζεται ένα, και μόνο ένα, εισερχόμενο μήνυμα κάθε φορά και παράγει ένα ή περισσότερα εξερχόμενα μηνύματα. Ένας κόμβος μπορεί να φιλτράρει, να εμπλουτίσει, να μετασχηματίσει ένα εισερχόμενο μήνυμα, να πραγματοποιήσει κάποια ενέργεια ή να επικοινωνήσει με κάποιο εξωτερικό σύστημα.

Υπάρχουν οι εξής διαθέσιμοι τύποι κόμβων κανόνων:

- Filter Nodes, που χρησιμοποιούνται για φιλτράρισμα και αναδρομολόγηση μηνυμάτων
- Enrichment Nodes, που χρησιμοποιούνται για εμπλουτισμό των μεταδεδομ΄νων (metadata) του εισερχόμενου μηνύματος
- Transformation Nodes, που χρησιμοποιούνται για την τροποποίηση των πεδίων (Originator, Type, Payload, Metadata, κλπ) του εισερχόμενου μηνύματος
- Action Nodes, που εκτελούν διάφορες ενέργειες με βάση το εισερχόμενο μήνυμα
- External Nodes, που χρησιμοποιούνται για την αλληλεπίδραση με εξωτερικά συστήματα

## 3.3. Συσχέτιση κόμβου κανόνα (rule node relation)

Κάθε κόμβος μπορεί να συσχετιστεί με άλλους κόμβους. Κάθε σχέση έχει έναν "τύπο", μια ετικέτα που χρησιμοποιείται για να χαρακτηρίσει το λογικό νόημα της σχέσης. Όταν ένας κόμβος παράγει ένα εξερχόμενο μήνυμα, τότε απαραίτητα ορίζει τον τύπο της σχέσης που πρέπει να χρησιμοποιηθεί για να δρομολογήσει το μήνυμα στον επόμενο κόμβο.

Τυπικές σχέσεις μεταξύ κόμβων είναι η "επιτυχία" (success) ή "αποτυχία (failure). Κόμβοι που αναπαριστούν λογικές λειτουργίες μπορεί να χρησιμοποιήσουν "true" ή "false" αντίστοιχα. Άλλοι κόμβοι δύναται να χρησιμοποιήσουν εντελώς διαφορετικούς, προκαθορισμένους ή μη τύπους σχέσεων.

### 3.4. Αλυσίδα κανόνων (rule chain)

Μια αλυσίδα κανόνων είναι μια λογική ομάδα από κόμβους κανόνων και μεταξύ τους σχέσεις. Για παράδειγμα η παρακάτω αλυσίδα κανόνων:



- Αποθηκεύει τα μηνύματα τηλεμετρίας στη βάση δεδομένων
- Παράγει ειδοποίηση τύπου "High Temperature Alarm" αν το πεδίο της θερμοκρασίας (temperature) εντός του μηνύματος έχει τιμή πάνω από 50 βαθμούς
- Παράγει ειδοποίηση τύπου "Low Temperature Alarm" αν το πεδίο της θερμοκρασίας (temperature) εντός του μηνύματος έχει τιμή κάτω από -40 βαθμούς
- Καταγράφει την αποτυχία να ελέγξει την τιμή της θερμοκρασίας στο εισερχόμενο μήνυμα, σε περίπτωση λογικού ή συντακτικού σφάλματος

#### 3.5. Παραμετροποίηση κόμβων

Κάθε κόμβος κανόνα δύναται να έχει συγκεκριμένες παραμέτρους που μπορούν να παραμετροποιηθούν. Για παράδειγμα ο κόμβος "Filter - script" μπορεί να παραμετροποιηθεί, με χρήση κώδικα σε γλώσσα JavaScript, για επεξεργασία του εισερχόμενου μηνύματος. Ο κόμβος "External - send email" μπορεί αντίστοιχα να παραμετροποιηθεί με τις απαραίτητες ρυθμίσεις σύνδεσης του διακομιστή ηλεκτρονικής αλληλογραφίας (mail server).

#### 4. Γραφικά στοιχεία (widgets)

Τα γραφικά στοιχεία (widgets) της πλατφόρμας είναι μονάδες διεπαφής χρήστη (UI) που μπορούν να ενσωματωθούν σε πίνακες ελέγχου (dashboards) και παρέχουν στον τελικό χρήστη επιπρόσθετες δυνατότητες, όπως οπτικοποίηση δεδομένων, απομακρυσμένος έλεγχος συσκευών, διαχείριση ειδοποιήσεων, ακόμα και να δείξουν στατικό περιεχόμενο HTML. Κάθε widget αναπαριστά ένα συγκεκριμένο τύπο (widget type), ανάλογα με τις δυνατότητες που παρέχει στο χρήστη.

#### 4.1. Δημιουργία γραφικού στοιχείου

Για τη δημιουργία ενός νέου widget, πλοηγούμαστε στην επιλογή "Widget Library". Έπειτα επιλέγουμε μια από τις υπάρχουσες συλλογές γραφικών στοιχείων (widget bundles) ή δημιουργούμε μια νέα. Εντός του επιλεγμένου bundle, πατάμε στο κουμπί "+" και έπειτα επιλέγουμε τον τύπο γραφικού στοιχείου που

θέλουμε να δημιουργήσουμε.



'Επειτα, η σελίδα επεξεργασίας γραφικού στοιχείου ("Widget Editor") θα ανοίξει, με τμήματα προπληρωμένα με κώδικα σε γλώσσα HTML/JavaScript/CSS, ανάλογα με τον επιλεγμένο τύπο.



Αφού προστεθεί η απαιτούμενη λειτουργικότητα, το γραφικό στοιχείο μπορεί να δοκιμαστεί με χρήση του κουμπιού "Run" και να αποθηκευτεί, δίνοντάς του ένα όνομα, ώστε να μπορεί να προστεθεί σε ένα dashboard.

#### 5. Δεύτερη έκδοση πίνακα ελέγχου πλατφόρμας

Χτίζοντας πάνω στην πρώτη, βασική έκδοση του πίνακα ελέγχου, με οπτικοποίηση των δεδομένων τηλεμετρίας των αισθητήρων σε πραγματικό χρόνο, προχωρήσαμε στη δημιουργία της δεύτερης, ολοκληρωμένης έκδοσης της πλατφόρμας.

Η δεύτερη έκδοση περιλαμβάνει όλα τα χαρακτηριστικά της πρώτης, και επιπλέον τα εξής:

- Πολλαπλές καταστάσεις / επίπεδα οπτικοποίησης (dashboard states), με ειδικό state για πληροφορίες ανά τοποθεσία σε πραγματικό χρόνο, και διαφορετικό state για ιστορικές τιμές
- Ιστορικά γραφήματα για πλοήγηση σε παρελθοντικές τιμές, για όλες τις μετρήσεις
- Πληροφορίες για την κατάσταση των συσκευών αισθητήρων (στάθμη μπαταρίας, ένταση σήματος GSM), τόσο σε πραγματικό χρόνο, όσο και ιστορικά, σε γραφήματα

Επιπλέον, για να εξυπηρετήσουμε την οπτικοποίηση των τιμών στάθμης νερού, απαιτήθηκε η δημιουργία σχετικού κανόνα (rule) στη μηχανή κανόνων, για μετασχηματισμό της πηγαίας τιμής μέτρησης στην πραγματική.

Τέλος, για να εξυπηρετήσουμε την ανάγκη απομακρυσμένης διαχείρισης παραμέτρων των συσκευών μέτρησης, προχωρήσαμε σε δημιουργία κατάλληλου γραφικού στοιχείου (widget) και αντίστοιχου πίνακα ελέγχου (dashboard).

#### 5.1. Κανόνας υπολογισμού στάθμης νερού

Οι υπερηχητικοί αισθητήρες μέτρησης στάθμης νερού πραγματοποιούν μέτρηση της απόστασης από το αισθητήριο έως την επιφάνεια του υδάτινου σώματος πάνω από το οποίο έχει τοποθετηθεί. Αυτό σημαίνει ότι το ύψος της στάθμης του νερού υπολογίζεται με αφαίρεση της τιμής μέτρησης από το ύψος τοποθέτησης του αισθητηρίου.

ύψος στάθμης νερού = ύψος τοποθέτησης - τιμή μέτρησης

Ήταν απαραίτητο, λοιπόν, να δημιουργηθεί ένας κανόνας στη μηχανή κανόνων, που να μετασχηματίζει τις εισερχόμενες τιμές μέτρησης του αισθητήρα στάθμης, στην πραγματική τιμή ύψους στάθμης νερού.



Ο παραπάνω κανόνας δουλεύει ως εξής:

 Κάθε εισερχόμενο μήνυμα (input node) δρομολογείται στον κόμβο "check existence fields" τύπου "Filter", ο οποίος φιλτράρει τα μηνύματα που περιέχουν το κλειδί "s\_wl", το οποίο περιέχει την τιμή μέτρησης του αισθητήρα στάθμης, απορρίπτοντας τα υπόλοιπα μηνύματα.

🔍 🔍 🍵 💎 Open ELloT   Rule chair	in x +	
← → C 伦 openeliot.exm.	<b>gr</b> /ruleChains/4fdbc530-0f98-11ea-8dd0-a15d716746	7a
<b>Open</b> ELloT	<> Rule chains > <> Water Level	Offset Q :: Offset
🔒 НОМЕ	Q Search nodes <	WATER LEVEL
↔ RULE CHAINS	\Xi Filter	Filter - check existence fields
ASSETS		
	C = check relation	Name* Debug mode
ENTITY VIEWS	gps geofencing filter	
WIDGETS LIBRARY	= message type	Message data *
DASHBOARDS		s_wl× Message data
O AUDIT LOGS		You should press "enter" to complete field input.
	- originator type	Message metadata *
		Vou should areas faster to complete field input
	O = script	Check that all selected keys are present
	Q <del>,</del> switch	If selected, checks that all specified keys are present in the message data and metadata.
	≓+ Enrichment	Description
	=+ customer attributes	
	E=+ customer details	
	□ =+ device attributes	
	E=+ originator attributes	

 Τα φιλτραρισμένα μηνύματα, που περιέχουν τιμές στάθμης, δρομολογούνται στον κόμβο "originator attributes" τύπου "Enrichment" για να εμπλουτισμό του εισερχόμενου μηνύματος με την πληροφορία του ύψους τοποθέτησης του αισθητήρα (water\_level\_offset), η οποία έχει αποθηκευτεί στη συσκευή ως attribute.

Open ELIoT   Rule chair	n × +	
← → C ♠ openeliot.exm.g	gr/ruleChains/4fdbc530-0f98-11ea-8dd0-a15d71674	37a 🖹 🖹
<b>Open</b> ELloT	<> Rule chains > <> Water Leve	el Offset Q 🖸 <sup>Tenant administrator</sup> 🗄
🔒 НОМЕ	Q Search nodes <	GET OFFSET
↔ RULE CHAINS	⊤ Filter ^	Enrichment - originator attributes
		DETAILS EVENTS HELP
ASSETS		
		Name* get Offset Debug mode
ENTITY VIEWS	gps geofencing filter	-
WIDGETS LIBRARY	= message type	Tell Failure
DASHBOARDS		If at least one selected key doesn't exist the outbound message will report "Failure".
O AUDIT LOGS		Client attributes
	- originator type	Shared attributes
		Shared attributes
	_ = script	Server attributes
	0 <del>=</del> switch	water_level_offset X Server attributes
		Latest timeseries
		Latest timeseries
	C =+ customer attributes	Description
	C ≡+ customer details	
	E =+ device attributes	
	=+ originator attributes	

3. Έχοντας όλη την πληροφορία που απαιτείται για να γίνει ο υπολογισμός της πραγματικής στάθμης νερού, το μήνυμα δρομολογείται στον κόμβο "script" τύπου "Transformation" για μετασχηματισμό της εισερχόμενης τιμής μέτρησης στην πραγματική. Η έξοδος του κόμβου είναι ένα μήνυμα που περιέχει την πραγματική τιμή στάθμης νερού (Actual Water Level).

← → C i openeliot.exm	n.gr/ruleChains/4fdbc530-0f98-11ea-8dd0-a15d7	67467a	
<b>Open</b> ELloT	<ul> <li>↔ Rule chains</li> <li>&gt;</li> <li>&gt; Water</li> </ul>	Level Offset Q []	
🔒 НОМЕ	Q Search nodes	OFFSET	
↔ RULE CHAINS	∓ Filter ∧		
		DETAILS EVENTS HELP	$\checkmark$
ASSETS			
	□ = check relation	Offset	Debug mode
ENTITY VIEWS	gps geofencing filter		
WIDGETS LIBRARY	- message type	Transform	TIDY 53
DASHBOARDS	= message type switch	1 - var newMsg = {	
() AUDIT LOGS	= originator type	<pre>2 ts: metadata.ts, 3- values: { 4 "Actual Water Level": parseFT 5 cr metan level offect)</pre>	loat(metadata
	originator type switch	5 }	msg.s_wi
	Q = script	7 return {msg: newMsg, metadata: metada	<pre>ita, msgType: msgType};</pre>
	Q = switch	}	
	≓+ Enrichment ^	TEST TRANSFORMER FUNCTION	
	=+ customer attributes	Description	
	=+ customer details		
	=+ device attributes		
	=+ originator attributes		

4. Τέλος, το μήνυμα που περιέχει την πραγματική τιμή της στάθμης νερού δρομολογείται στον κόμβο εξόδου "save timeseries" που αποθηκεύει την τιμή σαν δεδομένο τηλεμετρίας.



Για να ενεργοποιηθεί ο νέος κανόνας, έγινε σύνδεσή του με την υπόλοιπη αλυσίδα κανόνων (root rule chain).

Όπως φαίνεται στην παραπάνω εικόνα, όταν εισέρχεται στο σύστημα ένα μήνυμα που περιέχει δεδομένα τηλεμετρίας (τύπου Post Telemetry), περνάει από τον νέο κανόνα που έχει δημιουργηθεί, ώστε να προστεθεί σε αυτό η τιμή της στάθμης νερού, αν απαιτείται, ακολουθώντας την ροή που περιγράφηκε παραπάνω.

# 5.2. Πίνακας ελέγχου απομακρυσμένης διαχείρισης συσκευής

# 5.2.1. Γραφικό στοιχείο επεξεργασίας παραμέτρων συσκευής

Για να είναι δυνατή η επεξεργασία των παραμέτρων των συσκευών μετρήσεων (water nodes), όπως ο ρυθμός δειγματοληψίας και ο ρυθμός αποστολής δεδομένων, ήταν απαραίτητη η δημιουργία σχετικού γραφικού

#### στοιχείου.

<b>Open</b> ELloT	😫 Widgets Bundles > 📑 ELIoT Widgets > 🕕 Eliot Device Widget 🚦 🕄 🤤 <sup>Tenant administrator</sup> 🚦
🔒 НОМЕ	Eliot Device Widget Latest values 👻 🕨 RUN 🍽 UNDO 📑 SAVE 🗟 SAVE AS 🕄 TOGGLE FULLSCREEN
··· > RULE CHAINS	RESOURCES HTML CSS SETTINGS SCHEMA DATA KEY SETTINGS SCHEMA
	1 - <pre>cform class="ota-config-form" 2 name="esp320taUpdateConfigForm" TIDY [] 1 {} TIDY []</pre>
ASSETS	<pre>i 3 ng-submit="updateOtaConfig()"&gt; 4 - <md-input-container></md-input-container></pre>
	<pre>5 <label>Last config data ID: {{ data_id }}</label></pre>
ENTITY VIEWS	7 < md-input-container> 8 <label>Last activity time: {{</label>
WIDGETS LIBRARY	lastActivityTime }} 9
DASHBOARDS	10- diversity - Production 15nv 0."
	3 let types;       13         4 let utils;       Last config data ID:         5 let settings;       Last activity time:         8       Scope = self.ctx.\$scope;         10       \$scope = self.ctx.\$scope;         11       toxt = \$scope.\$injector.get('toxt');         12       types_i
	<pre>13 utils = \$scope.\$injector.get('utils'); 14 settings = angular.copy(self.ctx.settings)    {} Weather station measure interval (mins) ; 15 datasource = self.ctx.datasources[0]; 16 subscription = self.ctx.defaultSubscription;</pre>
	17     \$scope.settings = settings;     Call home interval (mins)       18     attributeService = \$scope.\$injector.get     ('attributeService');
	20 \$scope.updateOtaConfig = updateOtaConfig; 21 \$scope.WAKEUP_SCHEDULE_VALID_VALUES = [0, 1, 2, 3, 4, 5, 10, 15, 20, 30, 60, 120, 240];

Η αναλυτική περιγραφή του κώδικα που υλοποιεί την επιθυμητή συμπεριφορά ξεφεύγει από τα όρια αυτού του κειμένου. Η λειτουργικότητα του γραφικού στοιχείου συνοψίζεται ως εξής:

- Ο χρήστης, μέσα από το widget, ρυθμίζει τις επιθυμητές τιμές παραμέτρων (π.χ. ρυθμός δειγματοληψίας και αποστολής δεδομένων)
- Οι παράμετροι αποθηκεύονται ως χαρακτηριστικά (attributes) στην οντότητα της συσκευής
- Η φυσική συσκευή συνδέεται στην πλατφόρμα μέσω πρωτοκόλλου ΗΤΤΡ, με χρήση του REST API της πλατφόρμας, και ζητάει από αυτή τις επικαιροποιημένες τιμές των παραμέτρων
- Η συσκευή αντικαθιστά τοπικά στην ενσωματωμένη μνήμη της (Flash memory) τις παλιές παραμέτρους και τις χρησιμοποιεί στο εξής τις νέες τιμές

Μέσα από το γραφικό στοιχείο ο χρήστης μπορεί να ορίζει τις παρακάτω εντολές και παραμέτρους:

- Sensors measure interval: ο ρυθμός δειγματοληψίας, σε λεπτά, δηλαδή ο ρυθμός με τον οποίον θα ξυπνάει η συσκευή για να πάρει μετρήσεις από του αισθητήρες της
- Call home interval: ο ρυθμός αποστολής, σε λεπτά, δηλαδή ο ρυθμός με τον οποίον θα ξυπνάει η συσκευή για να επικοινωνήσει με την πλατφόρμα (αποστολή δεδομένων, λήψη νέων ρυθμίσεων/εντολών και εφαρμογή)
- OTA Update: εντολή για λήψη νέου λογισμικού, εφαρμογή του και εκκίνηση σε αυτό
- Reboot: εντολή για επανεκκίνηση της συσκευής
- Format SPIFFS: εντολή για διαμόρφωση (format) της εσωτερικής μνήμης flash

#### 5.2.2. Πίνακας ελέγχου απομακρυσμένης διαχείρισης

Για να γίνει χρήση του γραφικού στοιχείου που δημιουργήθηκε πρέπει αυτό να ενσωματωθεί σε έναν πίνακα ελέγχου. Δημιουργήθηκε λοιπόν ένας νέος πίνακας ειδικού σκοπού, αποκλειστικά και μόνο για τον απομακρυσμένο έλεγχο των παραμέτρων της συσκευής.

est		Device N	lanagement 🔻	□ Pikrodafni Node 🕓 Rea	altime - last day 👤 🛟
Create new configuration	🏚 Target configuratio	on (outdoing)		🗯 Device attributes (incoming	)) ;
ast config data ID: 157/721222	Pikrodafni Node			Pikrodafni Node	
ast coming data ID. 1574721252	Remote control dat	a ID	1574721232	FW Version	106.00
ast activity time: Tue Nov 26 2019 16:	Calling home interv	al	30.00	Calling home interval (mins)	30.00
	Water sensors mea	sure interval (mins)		System time	Tue, 26 Nov 201
Water sensors measure interval (mins)	Water sensors mea	isure interval (mins)	30.00	Weather station read interva	d (mins)
	OTA Update		false	Water sensors read interval	(mins)
30 -					
	:≡ Logs				Q
Weather station measure interval (mins)	Realtime - last 7 day	s			
*	Timestamp V	Log			Raw
	2019-11-26 16:52:14	Sensor data submission er	rors (Total requests: 2	- Failed requests: 1)	6,2,1
Call home interval (mins)					
30	2019-11-26 16:52:14	Sensor data submitted (To	tal entries: 11 - CRC fa	ilures: 0)	5,11,0
	2019-11-26 16:51:49	GSM: RSSI (-56)			101-56.0
	2019-11-26 16:51:49	Battery gauge (Voltage: 40	49 mV - Pct: 81%)		18,4049,81
onrapadie	2010 11 26 16 51 24	File sustem analos (Llosdi 2	000 B. Eroo: 1272460	D)	47 2000 1272460
	2019-11-20 10.31.24	File system space (osed. 2	000 B - Flee. 1372400	5)	47,2000,1372400
Reboot	2019-11-26 16:51:24	Internal env. sensor (Press	ure: 1002hPa - Alt: 88r	n)	17,1002,88
-					
	2019-11-26 16:51:24	Internal env. sensor (Temp	erature: 15C - Rel. Hun	hidity: 98%)	16,15,98
Format SPIFFS	2019-11-26 16:51:23	Battery gauge (Voltage: 38	30 mV - Pct: 73%)		15,3830,73
		,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,			
SAVE	2019-11-26 16:51:23	Calling home			3,0,0
	2019-11-26 16:51:15	Waking up (Reason: Water	- Weather)		1330
	2017112010.01.10	Making up (neubon: Mater	freduciery		10,0,0
	2019-11-26 16:41:26	Going to sleep (Time awak	e: 11s - Sleeping for: 0	s)	12,11,0

Στον συγκεκριμένο πίνακα ελέγχου, εκτός του γραφικού στοιχείου ελέγχου των παραμέτρων, έχουν προστεθεί γραφικά στοιχεία για προβολή των τρεχουσών τιμών των παραμέτρων της επιλεγμένης συσκευής, καθώς και πλήρες ιστορικό των μηνυμάτων αποσφαλμάτωσης (debug logs) που στέλνουν οι συσκευές σε τακτά χρονικά διαστήματα.

Πατώντας "Save" στο γραφικό στοιχείο ελέγχου των παραμέτρων, οι ρυθμίσεις αποθηκεύονται και θα ληφθούν από τη συσκευή στη επόμενη επικοινωνία της με την πλατφόρμα. Αφού οι ρυθμίσεις ληφθούν και εφαρμοστούν από τη συσκευή, τα αποτελέσματά τους μπορούν να προβληθούν στα debug logs.

#### 5.2.3. Πίνακας ελέγχου δεδομένων



Στην πρώτη έκδοση του πίνακα ελέγχου υπήρχε οπτικοποίηση της τοποθσίας των αισθητήρων σε χάρτη, λίστα με τη συνδεδεμένη συσκευή κάθε τοποθεσίας και κάρτες με τις τρέχουσες τιμές μετρήσεων για τα διάφορα μετρούμενα μεγέθη.

Για μεγαλύτερη λεπτομέρεια στις τρέχουσες επικρατούμενες συνθήκες, αρχικά προστέθηκε ένα δεύτερο state, στο οποίο γίνεται πλοήγηση πατώντας στο όνομα της τοποθεσίας από τη λίστα, ή στην "πινέζα" της στο χάρτη.



Σε αυτό το state φαίνεται, επιπλέον του αρχικού state:

- Η τρέχουσα (πιο πρόσφατη) τιμή για κάθε μέτρηση
- Η χρονική στιγμή της δειγματοληψίας της μέτρησης
- Η μέγιστη και ελάχιστη επιτρεπόμενη τιμή για κάθε μέτρηση
- Χάρτης με το σημείο τοποθέτησης σε μέγιστη μεγέθυνση

Στη συνέχεια, για να εξυπηρετηθεί η ανάγκη οπτικοποίησης των ιστορικών δεδομένων μετρήσεων, δημιουργήθηκε ένα τρίτο state, στο οποίο προστέθηκαν επιπλέον γραφικά στοιχεία γραφημάτων. Σε αυτό το state γίνεται πλοήγηση πατώντας στο όνομα της συσκευής (Water Sensors).



Σε αυτό το state φαίνονται οι τρέχουσες τιμές των μετρήσεων, για πληρότητα, ενώ έχουν προστεθεί σε αυτές οι τρέχουσες τιμές της στάθμης μπαταρίας της συσκευής και της έντασης του σήματος GSM.

Επιπλέον, έχουν προστεθεί γραφήματα με τις ιστορικές τιμές ανά μέτρησης, με ανάλυση 15 λεπτών. Τα γραφήματα είναι δυναμικά, με δυνατότητα μεγέθυνσης και μετακίνησης στο χρόνο, και πλήρως παραμετροποιήσιμα ως προς την ανάλυση των δεδομένων (interval), ενώ μπορεί να γίνει και προβολή τους σε πλήρη οθόνη για μεγαλύτερη λεπτομέρεια, όπως για παράδειγμα στο παρακάτω γράφημα στάθμης νερού του ρέματος της Πικροδάφνης.



Όπως φαίνεται και παραπάνω, σε όλα τα γραφήματα έχουν προστεθεί (κάτω δεξιά) επιπρόσθετες πληροφορίες για τη μέγιστη, ελάχιστη και μέση τιμή της εκάστοτε μέτρησης.

Τα γραφήματα μετακινούνται σε πραγματικό χρόνο όσο έρχονται νέες μετρήσεις, ώστε να περιλαμβάνουν πάντα τις πιο πρόσφατες τιμές. Επιπλέον, χρησιμοποιώντας το στοιχείο ελέγχου του χρόνου, που βρίσκεται πάνω δεξιά στον πίνακα ελέγχου, είναι δυνατό να επιλεγεί το επιθυμητό παράθυρο χρόνου προβολής των δεδομένων, η ανάλυση των μετρήσεων (interval) και η συνάρτηση άθροισης (data aggregation function) των μετρήσεων. Για παράδειγμα, αν επιλεγεί συνάρτηση "Average" και interval 30 λεπτών, τότε αν η δειγματοληψία είναι πιο συχνή (π.χ. 5-10 λεπτά, ή και λιγότερα), τότε το γράφημα θα δείξει μία τιμή ανά 30 λεπτά - δηλαδή μία τιμή για το επιλεγμένο interval - η οποία θα είναι ο μέσος όρος όλων των τιμών εντός του συγκεκριμένου 30λεπτου.

<b>Open</b> ELloT	Dashboards	> 📑 Overview				a 😣		
🔒 НОМЕ	Overview > Pikro	dafni > Pikrodafni \	Water Node			Realtime - last day		:3
··> RULE CHAINS	[		1		REALTIME HISTORY			101 (db)
CUSTOMERS	Water Level (cm)	Temperature (°C)	Conductivity (µS/cm)	pН	Last		Advanced	-52
ASSETS					1 day		• 🔘	-1
	18	17.1	874.9		Data aggregation function			
ENTITY VIEWS	0 26/11, 18:20 300	-10 26/11, 18:20 50	0 26/11, 18:20 2500	0 26	Average		•	-121 N/11, 78:21
WIDGETS LIBRARY	Water Level			Water -	Grouping interval 30 minutes		Advanced	
DASHBOARDS	24 cm			18 °C				
) AUDIT LOGS	23 cm	h	~~~	17 °C		UPDATE	CANCEL	-
	21 cm 20:00 00:00	04:00 08:00	12:00 16:00	15 °C 20	00 00:00 04:00	08:00	12:00 16:00	

Με αυτήν την ολοκληρωμένη έκδοση της πλατφόρμας, είναι πλέον δυνατή η πλήρης οπτικοποίηση των δεδομένων των συνδεδεμένων συσκευών, και ο απομακρυσμένος έλεγχος των παραμέτρων τους.

#### 6. Αποθετήριο κώδικα

Ο κώδικας (σε μορφή JSON) της δεύτερης έκδοσης του πίνακα ελέγχου δεδομένων της πλατφόρμας βρίσκεται στον παρακάτω σύνδεσμο.

https://github.com/exmgr/openeliot-platform/blob/release-2.4/dashboards/platform-dashboard-v2.json