



# ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

## Τμήμα Πληροφορικής

Η πλατφόρμα Οικιακών Αυτοματισμών openHAB  
Περιγραφή – Εγκατάσταση – Διαμόρφωση – Χρήση

### (Συνοδευτικό Εγχειρίδιο Άσκησης 2)

Διδάσκων: Δημήτρης Ζεϊναλιπούρης

Υπεύθυνος Εργαστηρίου: Παύλος Αντωνίου

<http://www.cs.ucy.ac.cy/courses/EPL421>

## 1. Εισαγωγή στο openHAB

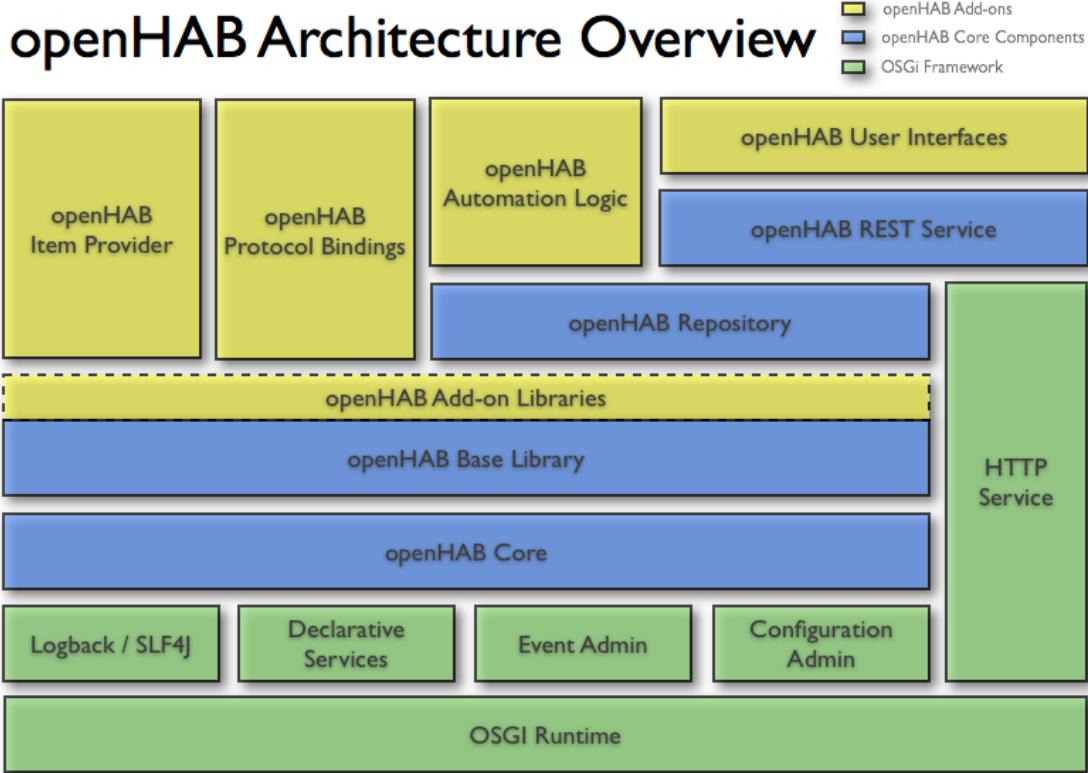
Το openHAB αποτελεί ακρωνύμιο του open Home Automation Bus και είναι μία ανοιχτό ύπερ-εγκατάσταση για την υλοποίηση οικιακών αυτοματισμών. Το λογισμικό αυτό ενσωματώνει διάφορα συστήματα, συσκευές και τεχνολογίες οικιακών αυτοματισμών σε μία ενιαία λύση. Προσφέρει ομοιόμορφες διεπαφές χρήστη, και μία κοινή προσεγγιση σε κανόνες αυτοματισμού σε ολόκληρο το σύστημα, ανεξάρτητα από τον αριθμό των κατασκευαστών και των υποσυστημάτων που το απαρτίζουν.

### 1.1 Εγκατάσταση openHAB

Παρακαλώ προβείτε στην εγκατάσταση του openHAB (version 2) προτού διαβάσετε τις πιο κάτω ενότητες. Ειδικότερα, ακολουθείστε τις εντολές που φαίνονται στο APPENDIX 1 για να γίνει η εγκατάσταση στο VM σας. Στη συνέχεια, επιχειρήστε να δείτε την υπηρεσία σας στο <http://VMIP:8080/> αφού πρώτα έχετε ανοίξει την πόρτα του firewall σας με το iptables. Μπορείτε επίσης να αλληλοεπιδράσετε λίγο με το κινητό σας συνδεδεμένο στο <http://VMIP:8080/> προτού εμβαθύνετε στους ορισμούς.

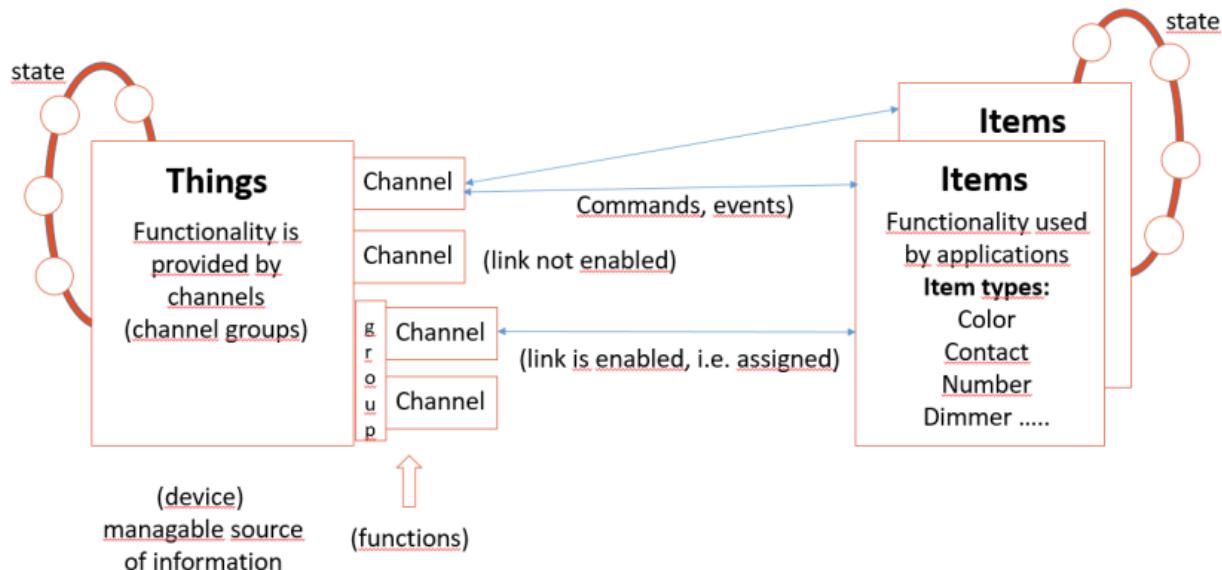
### 1.2 Δομή openHAB

Το OpenHAB είναι ανεπτυγμένο σε Java και κυρίως βασισμένο στο Eclipse SmartHome framework. Χρησιμοποιεί Apache Karaf μαζί με Eclipse Equinox για να δημιουργήσει το περιβάλλον εκτέλεσης Open Services Gateway initiative (OSGi). Το Jetty χρησιμοποιείται σαν εξυπηρετητής HTTP. Το OpenHAB, καθώς αποτελεί ένα «οπονδυλωτό» (modular) λογισμικό, μπορεί να επεκταθεί με την χρήση “Add-ons” (bindings). Τα Add-ons δίνουν στο openHAB ένα ευρύ φάσμα δυνατοτήτων, από διεπαφές χρήστη, μέχρι την ικανότητα να διαδρά με ένα μεγάλο και αυξανόμενο αριθμό φυσικών αντικειμένων.



Εικόνα 1: Αρχιτεκτονική openHAB.

### 1.3 Βασικές Έννοιες



Εικόνα 2: Βασικές έννοιες στο openHAB.

Το Eclipse SmartHome διαφοροποιείται αυστηρά μεταξύ της φυσικής και της λειτουργικής άποψης του συστήματος. Ενώ η φυσική σκοπιά απαιτείται για την εγκατάσταση, διαμόρφωση, εντοπισμό προβλημάτων κ.λπ. του συστήματος, η

λειτουργική καλύπτει τις πληροφορίες που έχουν σημασία στις εφαρμογές, όπως οι διεπαφές χρήστη και η λογική αυτοματισμού.

## 1.4 Things, Channels, Items and Links

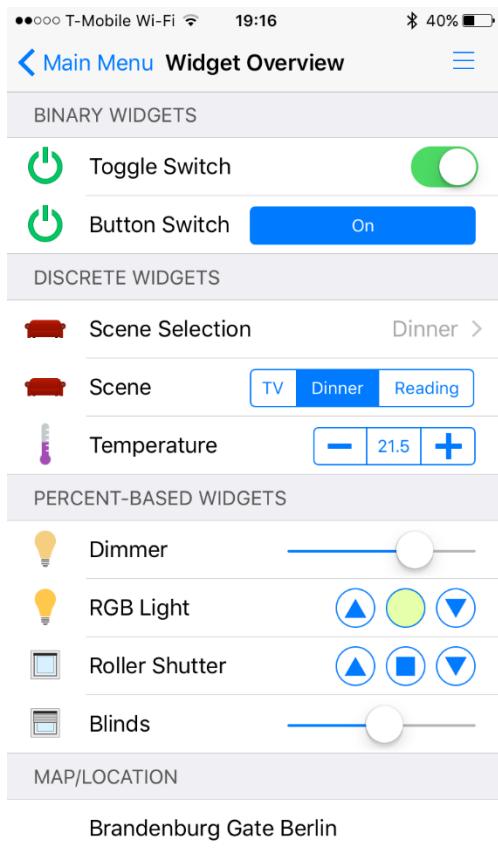
Με τον όρο Things αναφερόμαστε στις οντότητες οι οποίες μπορούν να προστεθούν φυσικά σε ένα σύστημα και δυνητικά παρέχουν πολλές λειτουργίες ταυτόχρονα. **Είναι σημαντικό να σημειωθεί πως τα Things δεν είναι ανάγκη να είναι συσκευές του πραγματικού κόσμου** (π.χ. ένας λαμπτήρας), αλλά μπορούν επίσης να αναπαριστούν μία διαδικτυακή υπηρεσία ή οποιαδήποτε άλλη διαχειρίσιμη πηγή πληροφορίας και λειτουργικότητας (π.χ. μια υπηρεσία που μας δίνει πληροφορίες για τις καιρικές συνθήκες). Από την προοπτική του χρήστη, τα things σχετίζονται με τη ρύθμιση και την διαδικασία διαμόρφωσης (configuration) του συστήματος αλλά όχι για την λειτουργία του. Τα Things παρέχουν την λειτουργικότητά τους μέσω μίας σειράς καναλιών (Channels). Τα κανάλια είναι παθητικά και μπορούν να θεωρηθούν ως μία δήλωση ενός Thing για το τι μπορεί να προσφέρει. Για παράδειγμα, ένας λαμπτήρας μπορεί να έχει ένα dimer channel και ένα color channel, όπου και τα δύο παρέχουν την λειτουργικότητα του λαμπτήρα (Thing) στο σύστημα. Με τον όρο Items αναφερόμαστε στις οντότητες που αναπαριστούν μία λειτουργικότητα ή οποία χρησιμοποιείται από εφαρμογές, όπως οι διεπαφές χρήστη ή την λογική αυτοματισμού. Τα Items βρίσκονται σε μία κατάσταση (state), και μπορούν να λάβουν εντολές προκειμένου αυτή να αλλάξει. Το openHAB παρέχει διάφορους τύπους Items όπως, Switch, Contact, Number κ.α. Αυτό που ενώνει τα Things και τα Items είναι οι σύνδεσμοι (Links). Οι σύνδεσμοι είναι ενώσεις μεταξύ ακριβώς ενός Thing Channel και ενός Item. Αν ένα κανάλι είναι συνδεδεμένο με ένα Item, τότε το Item αυτό είναι ενεργοποιημένο μέσα από το openHAB, που σημαίνει ότι η λειτουργικότητα που το Item αναπαριστά, είναι διαχειρίσιμη μέσω του δοθέντος καναλιού. Τα κανάλια μπορούν να συνδεθούν με πολλαπλά Items και τα Items με πολλαπλά κανάλια.

## 2. Διαμόρφωση (configurations)

Το openHAB, λοιπόν, είναι ένα σύστημα που εγκαθίσταται και εκτελείται από τον χρήστη και τρέχει ανεξάρτητα σε κάποια μηχανή με αποτέλεσμα ο τελικός χρήστης να έχει τον πλήρη έλεγχο πάνω σε κάθε όψη του οικιακού αυτοματισμού. Καθώς το openHAB είναι το κέντρο ενός οικιακού αυτοματισμού, όλες οι ιδιότητες και ικανότητες των συσκευών είναι διαθέσιμες στο user frontend και άλλα μέρη και συνδεδεμένα συστήματα, μέσω αυτού. Όμως κάθε συσκευή συνδεδεμένη στο openHAB είναι λειτουργικά και λογικά διαφορετική. Για την αναπαράσταση, λοιπόν, όλων αυτών των συσκευών, το openHAB ορίζει τα ακόλουθα βασικά μέρη:

- Bindings – Διάφορα Add-ons για την επικοινωνία με τις διαφορετικές συσκευές ή άλλες διαδικτυακές υπηρεσίες.
- Things – Αναπαράσταση των συσκευών στο openHAB.
- Items – Ιδιότητες και ικανότητες των συσκευών.

- Groups – Συλλογές ή κατηγορίες που περιέχουν Items.
- Sitemaps – Διεπαφή που καθορίζεται από τον χρήστη για την διάταξη των Groups, Items κ.α
- Transformations – Βοηθητικές συναρτήσεις που μετατρέπουν τα δεδομένα.
- Persistence – Υπηρεσίες για την αποθήκευση δεδομένων με το πέρασμα του χρόνου.
- Rules – Λογική αυτοματισμού, κανόνες που ενεργοποιούνται από events, δηλαδή, την αλλαγή κατάστασης κάποιου Item.
- JSR223 Scripting – Ορισμός κανόνων και runtime objects με την χρήση Javascript, Jython ή Groovy.



Η εικόνα 3 δείχνει πληροφορίες για κάποιες ομαδοποιημένες συσκευές ενός ορόφου μιας οικίας. Η διάταξη των πληροφοριών πάνω στο dashboard αυτό καθορίζεται από κάποιο sitemap file.

Για την διαμόρφωση (configuration) του openHAB υπάρχει αρκετά μεγάλη ευελιξία καθώς παρέχονται αρκετές διεπαφές για την τροποποίηση των ρυθμίσεων, την διαχείριση των Things και των Items, την ανάπτυξη κανόνων και την πρόσβαση στο Sitemap. Η διαμόρφωση μπορεί να γίνει είτε γραφικά είτε με χρήση text editor. Τα Things και τα Items μπορούν είτε να οριστούν και να τύχουν διαχείριση μέσω των configuration files ή από το Paper UI σε μία βάση δεδομένων από τη μεριά του συστήματος. Το openHAB μας δίνει τη δυνατότητα χρήσης και των δύο μεθόδων ταυτόχρονα, αν το επιθυμούμε. Στην παρούσα άσκηση, θα χρησιμοποιήσουμε τη γραφική διεπαφή (Paper UI) για τις διαμορφώσεις.

Εικόνα 3: Το Sitemap καθορίζει πως διατάσσουμε σε μια οθόνη (dashboard) πληροφορίες για συσκευές ενός ορόφου μιας οικίας. Η εικόνα αυτή λήφθηκε από την εφαρμογή openHAB για κινητά τηλέφωνα.

ορίζουμε και τις υπηρεσίες που χρησιμοποιούμε και τα χαρακτηριστικά τους.

## Configuration Files

Configuration files είναι τα αρχεία στα οποία ορίζουμε τις συσκευές μας σαν things ή items και τους κανόνες που τα διέπουν, ορίζουμε την εμφάνιση τους στο User Interface και τις διάφορες συναρτήσεις μετατροπής των δεδομένων. Ακόμα,

- Ορισμός Things/Things Files

Όπως είπαμε τα Things αναπαριστούν το φυσικό επίπεδο σε ένα openHAB σύστημα. Από την άποψη διαμόρφωσης, τα Things λένε στο openHAB ποιες φυσικές οντότητες (συσκευές, web services, information sources, κ.λπ.) είναι για να διαχειριστούν από το σύστημα. Τα Things είναι συνδεδεμένα στο openHAB μέσω των bindings. **Για να προσθέσει κάποιος ένα Thing στο σύστημα, πρέπει πρώτα να βρει και να εγκαταστήσει το κατάλληλο binding** το οποίο θα παρέχει τον τόπο του Thing που θες να προσθέσεις. Για παράδειγμα, πριν προσθέσουμε σαν Thing μία συσκευή π.χ. Daikin Aircondition, το Daikin binding πρέπει να είναι εγκατεστημένο. Άρα για να ορίσουμε ένα Thing πρέπει να ακολουθήσουμε την εξής διαδικασία:

- Εύρεση κατάλληλου binding
- Εγκατάσταση binding
- Ορισμός και διαμόρφωση Thing
- Εύρεση των καναλιών του Thing
- Προσθήκη Items και σύνδεση με το/τα κανάλι/α (επεξήγηση στον ορισμό Items)

Σε αυτό το σημείο (μετά την προσθήκη) τα Items μπορούν να χρησιμοποιηθούν για τον έλεγχο του Thing ή για την παροχή της πληροφορίας του thing στο sitemap ή τους κανόνες. Για τον ορισμό ενός Thing υπάρχουν δύο τρόποι. Είτε με την **αυτόματη «ανακάλυψη»** του στο Paper UI, όπου το Thing εντοπίζεται αυτόματα από το openHAB και οι ιδιότητές του μπορούν να διαμορφωθούν από τον χρήστη μέσω της διεπαφής είτε με τον **manual** τρόπο όπου ορίζουμε το Thing στο αντίστοιχο configuration text file. Για να ορίσουμε ένα Thing χρησιμοποιώντας ένα configuration text file, πρέπει πρώτα να δημιουργήσουμε το αρχείο στον φάκελο things που βρίσκεται στο openHAB configuration path /etc/openhab2/things. Το αρχείο θα πρέπει να έχει την κατάληξη .things. Η σύνταξη για τον ορισμό ενός Thing στο things configuration text file θα πρέπει να είναι η ως εξής:

Thing : <binding\_id>:<thing\_id> "Label" @ "Location" [ <parameters> ]

- **Ορισμός Items/Items Files**

Τα Items αναπαριστούν όλες τις ιδιότητες και ικανότητες του οικιακού αυτοματισμού ενός χρήστη. Τα Items μπορούν να είναι τόπου String, Number, Switch ή κάποιου από τους άλλους βασικούς τύπους που παρέχει η πλατφόρμα. Ένα αξιόλογο χαρακτηριστικό που μιας παρέχει το openHAB είναι ότι τα Items έχουν τη δυνατότητα να συνδεθούν με τον «έξω κόσμο» μέσω των Bindings, καθώς ένα Item δεν αποθηκεύει απλά πληροφορία που καθορίζεται από κάποιο software, αλλά η πληροφορία αυτή μπορεί να οριστεί από διάφορες ενέργειες που λαμβάνουν χώρα μέσα στο σύστημα. Τα Items, λοιπόν, είναι βασικοί τύποι δεδομένων και έχουν μία κατάσταση η οποία μπορεί να διαβαστεί αλλά και να αλλάξει. Μπορούν επίσης να είναι συνδεδεμένα με ένα Binding channel για να

μπορούν να δρούν με το υπόλοιπο σύστημα. Για παράδειγμα ένα Item δεσμευμένο (linked) σε ένα αισθητήρα λαμβάνει ενημερώσεις από αυτόν και ένα Item συνδεδεμένο στο dimmer channel ενός λαμπτήρα, μπορεί να ρυθμίσει την φωτεινότητα της λάμπας. Όπως και στα Things, υπάρχουν δύο τρόποι για να ορίσουμε ένα Item. Είτε χρησιμοποιώντας την διεπαφή του Paper UI, είτε με την χρήση configuration text files. Για τον ορισμό ενός Item με configuration text file, δημιουργούμε πρώτα το αρχείο στον αντίστοιχο φάκελο ο οποίος είναι `/etc/openhab2/items` και του δίνουμε την κατάληξη `.items`. Μπορούμε να έχουμε πάνω από ένα αρχείο τύπου `.items` αρκεί κάθε item που ορίζουμε σε αυτά να είναι μοναδικά. Η σύνταξη ενός Item μέσα στο configuration text file θα πρέπει είναι η ως εξής:

```
itemtype itemname "labeltext [stateformat]" <iconname> (group1) ["tag1"]
{bindingconfig}
```

Τα πεδία itemtype και itemname είναι υποχρεωτικά, μπορούν να υπάρξουν παραπάνω από ένα tag και group.

- **Sitemaps Files**

Στο openHAB μία συλλογή Things και Items αναπαριστά φυσικά ή λογικά αντικείμενα στο σύστημα. Τα sitemaps χρησιμοποιούνται για να επιλέξουν (ομαδοποίησουν) και να ετοιμάσουν τη διάταξη αυτών των στοιχείων προκειμένου να συνθέσουν μία φιλική προς τον χρήστη παρουσίαση αυτής της εγκατάστασης για διάφορες διεπαφές, συμπεριλαμβανομένου του Basic UI, της Android εφαρμογής κ.α. Τα sitemaps είναι configuration text files με την κατάληξη `.sitemap`, και πρέπει να τα δημιουργήσουμε στον φάκελο `/etc/openhab2/sitemaps`. Η σύνταξη ενός sitemap αρχείου θα μπορούσε είναι η εξής:

```
Sitemap demo label="My home automation" {
    Frame label="Date" {
        Text item="Date"
    }
    Frame label="Demo"{
        Switch item=Lights icon="light"
        Text item = LR_Temp label = "Room [%.1 °C]"
    }
}
```

- **Transformations Files**

Οι μετατροπές χρησιμοποιούνται για την μετάφραση δεδομένων από μία εξειδικευμένη ορολογία ή συντόμευση σε μία φιλική προς τον χρήστη μορφή. Για παράδειγμα μετατρέπουν εισερχόμενες τιμές ενός αισθητήρα για να μπορεί να τις

καταλάβει ο χρήστης ή ακόμα και να μετατρέπει τιμές ενός Item και να τις κάνει κατανοητές για κάποιο άλλο Item. Τα αρχεία για τις μετατροπές map (επεξήγηση στα Addons/transformation services) τα αποθηκεύουμε στον φάκελο /etc/openhab2/transform και τους δίνουμε την κατάληξη .map. Σημαντικό είναι να έχουμε εγκατεστημένα transformation services (πχ Map). Η σύνταξη και η λογική των αρχείων καθορίζεται από αυτήν. Παράδειγμα σύνταξης transformation map file:

0=No Connection  
1=Bad Connection  
2=Medium Connection  
3=Good Connection  
4=Strong Connection

- Rules Files

Οι κανόνες χρησιμοποιούνται για διαδικασίες αυτοματισμού· κάθε κανόνας μπορεί να ενεργοποιηθεί, ο οποίος με τη σειρά του επικαλείται ένα script που εκτελεί κάθε είδους δουλειάς, πχ αλλαγή κατάστασης κάποιου Item, μαθηματικές πράξεις, έναρξη χρονομετρητή κ.α. Το openHAB συμπεριλαμβάνει μία αρκετά ολοκληρωμένη, ελαφριά αλλά και δυνατή μηχανή κανόνων. Ο ορισμός των κανόνων γίνεται σε ένα αρχείο που δημιουργούμε στην τοποθεσία /etc/openhab2/rules και με κατάληξη .rules. Ένα .rules αρχείο μπορεί να περιέχει πολλαπλούς κανόνες. Όλοι οι κανόνες ενός αρχείου μπορούν να έχουν πρόσβαση στις μεταβλητές μεταξύ τους και να τις ανταλλάσσουν. Το Eclipse SmartHome Designer προσφέρει υποστήριξη για την ανάπτυξη κανόνων. Περιέχει συντακτικό έλεγχο και χρωματισμό, επικύρωση με δείκτες σφάλματος (error markers), βοήθεια περιεχομένου (content assist) κ.α. Η σύνταξη των κανόνων είναι βασισμένη στην Xbase και κατ' επέκταση μοιράζεται πολλές λεπτομέρειες με την Xtend, η οποία είναι επίσης χτισμένη πάνω στην Xbase. Τέλος, ένα αρχείο κανόνων είναι ένα αρχείο κειμένου με την εξής μορφή:

- Imports
- Variable Declarations
- Rules

Στο τμήμα των Imports εμπεριέχονται, ακριβώς όπως στη Java, τα import statements τα οποία επιθυμούμε. Όπως στην Java, οι εισαγόμενοι τύποι μας είναι διαθέσιμοι χωρίς να χρειάζεται να γράψουμε το πλήρες όνομα τους. Μερικά imports έχουν ήδη γίνει οπότε δεν χρειάζεται να τα κάνουμε εμείς. Το τμήμα της δήλωσης μεταβλητών (Variable Declaration) μπορεί να χρησιμοποιηθεί για να δηλώσουμε μεταβλητές που θέλουμε να είναι προσβάσιμες από όλους τους κανόνες του αρχείου. Μπορούμε να δηλώσουμε μεταβλητές με ή χωρίς αρχικές τιμές και τροποποιήσιμες ή μόνο για ανάγνωση. Το τμήμα των κανόνων περιέχει μία λίστα από κανόνες. Κάθε κανόνας μπορεί να έχει πάνω από μία συνθήκες

ενεργοποίησης. Κάτω από αυτές τις συνθήκες είναι το script block, όπου υπάρχει ο κώδικας που θα εκτελεστεί αν ενεργοποιηθούν οι συνθήκες. Υπάρχουν διάφορες κατηγορίες συνθηκών για την ενεργοποίηση των κανόνων. Αυτές είναι οι Item(-Event) – based, δηλαδή αντιδράσεις σε συμβάντα στο openHAB event bus, πχ εντολές και ενημέρωση καταστάσεων των Items, οι time – based, που αντιδρούν συγκεκριμένες ώρες, πχ μεσάνυχτα, κάθε ώρα κ.λπ., οι system – based, ενεργοποιούνται σε συγκεκριμένες καταστάσεις του συστήματος και οι thing – based, που αντιδρούν στις καταστάσεις των Things. Η σύνταξη και η δομή ενός rule file είναι κάπως έτοι:

```
//Import section
Import java.net.URI
//Variable Declaration section
var counter = 0
val msg = "This is a Message"
var Number x
//Rule section
rule "rule name"
when
    <trigger_condition1> or
    <trigger_condition2> or
    <trigger_condition3>...
then
    <script_block>
End
```

- Services Files

Τα services files βρίσκονται στον φάκελο /etc/openhab2/services και είναι τα αρχεία των εγκατεστημένων bindings. Σε αυτά τα αρχεία τροποποιούμε τα πεδία που μας ενδιαφέρουν έτοι όπως εμείς επιθυμούμε για την ορθή λειτουργία του server. Παράδειγμα, όταν εγκαταστήσουμε το mail binding (για να στέλνει email το openHAB), μπορούμε να πάμε σε αυτό τον φάκελο και θα βρούμε το αρχείο mail.cfg, όπου μπορούμε να ρυθμίσουμε τον mail (SMTP) server στον οποίο θα στέλνουμε emails, δηλαδή την IP διεύθυνση του, σε ποια πόρτα ακούει, αν απαιτεί ασφαλή σύνδεση (SSL/TLS) κ.α.

### 3. Πρόσθετα (add-ons)

Τα Add-ons είναι ένα κομμάτι εξίσου σημαντικό με τα configuration files για την διαμόρφωση του συστήματός μας, καθώς για κάθε συσκευή, πρωτόκολλο και υπηρεσία που θέλουμε να χρησιμοποιήσουμε πρέπει να προσθέσουμε στο σύστημά μας το αντίστοιχο Addon. Τα Add-ons διακρίνονται στους εξής τύπους:

- Bindings
- User Interfaces

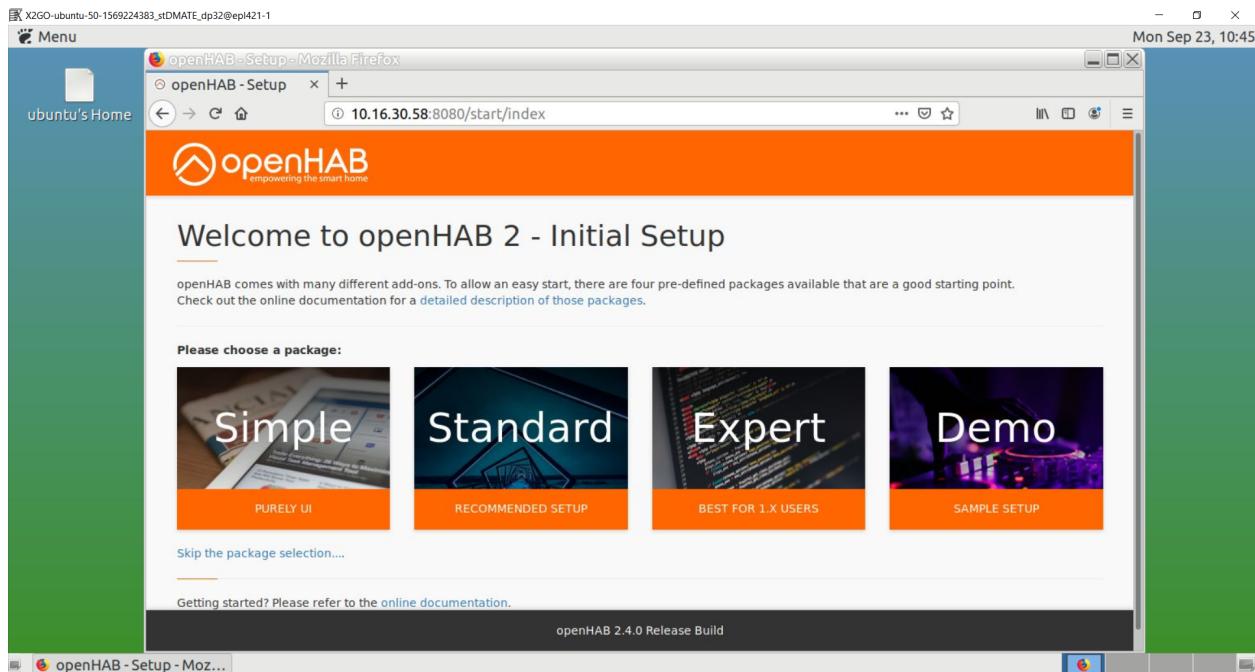
- Persistence
- Actions
- Transformations
- Voice Services
- 3rd Party System Integration

Τα Bindings ενσωματώνουν φυσικό hardware, εξωτερικά συστήματα και web services στο openHAB. Υπάρχουν Bindings για τις περισσότερες smart συσκευές, αλλά και bindings όπως το MQTT binding και άλλων πρωτοκόλλων για την επίτευξη της επικοινωνίας μεταξύ του server και των μικροελεγκτών που έχουμε προγραμματίσει για την εξ' αποστάσεως λειτουργία IoT συσκευών. Το OpenHAB προσφέρει διάφορα User Interfaces όπως έχουμε προαναφέρει και από το πεδίο των πρόσθετων μπορούμε να κατεβάσουμε αυτά που χρειαζόμαστε. Οι διεπαφές αυτές είναι:

- CLASSIC UI - Διεπαφή για πρόσβαση στο Sitemap.
- REST API - Πρόσβαση στο Rest API.
- BASIC UI - Διεπαφή για πρόσβαση στο Sitemap (πιο μοντέρνα γραφικά).
- PAPER UI - Διεπαφή για την τροποποίηση/διαχείριση των Items και των Things, προσφέρει Add-ons management και σύνδεση καναλιών με Items.
- HABMIN - Διεπαφή που παρέχει λειτουργίες χρήστη και διαχείρισης.
- HABPANEL - Διεπαφή dashboard, ενδείκνυται για tablets και εντοιχισμένες συσκευές.
- openHAB apps - Android/iOS/Windows 10 εφαρμογές για την διαχείριση του συστήματος από smartphone και Windows PC, χωρίς την ανάγκη για browser.

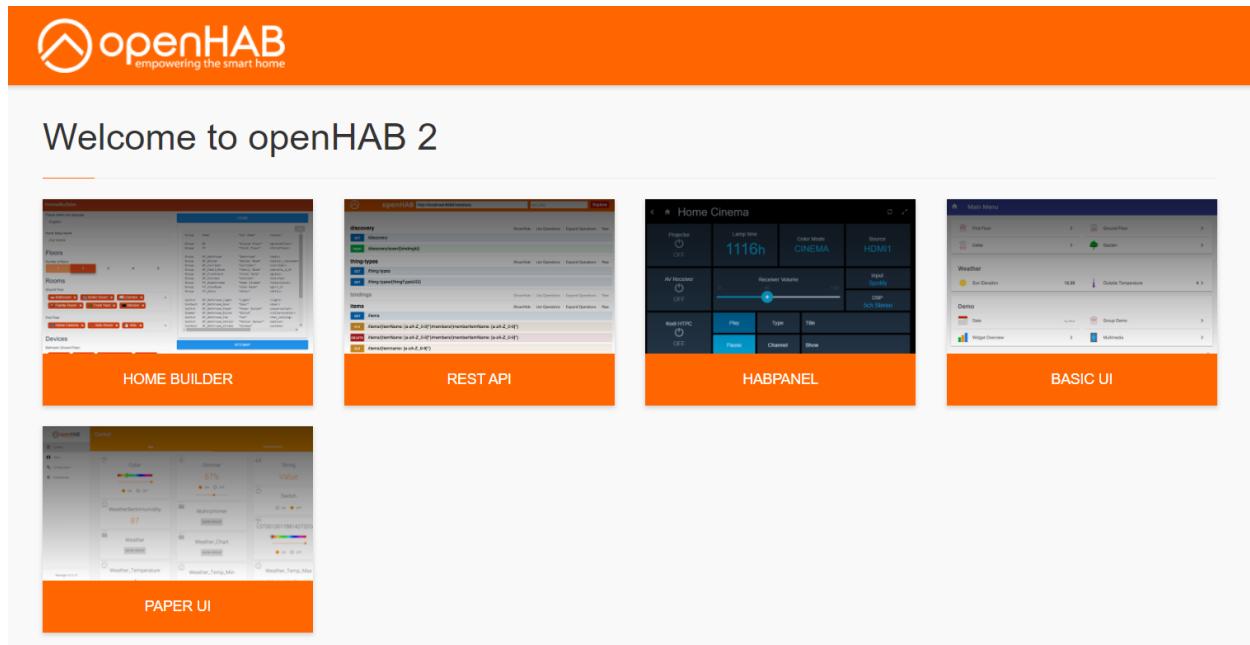
Για να δείτε το dashboard μαζί με τα UIs που προσφέρει το openHAB θα πρέπει να ενωθείτε με χρήση φυλλομετρητή στη θύρα 8080 της εικονικής σας μηχανής. Αν θέλετε να έχετε πρόσβαση στην εικονική σας μηχανή με χρήση γραφικού περιβάλλοντος ακολουθείστε τις εντολές στο APPENDIX 2.

Την πρώτη φορά που θα ενωθείτε θα πρέπει να εγκαταστήσετε ένα από τα διαθέσιμα πακέτα (Simple, Standard, Expert, Demo) όπως φαίνεται πιο κάτω:



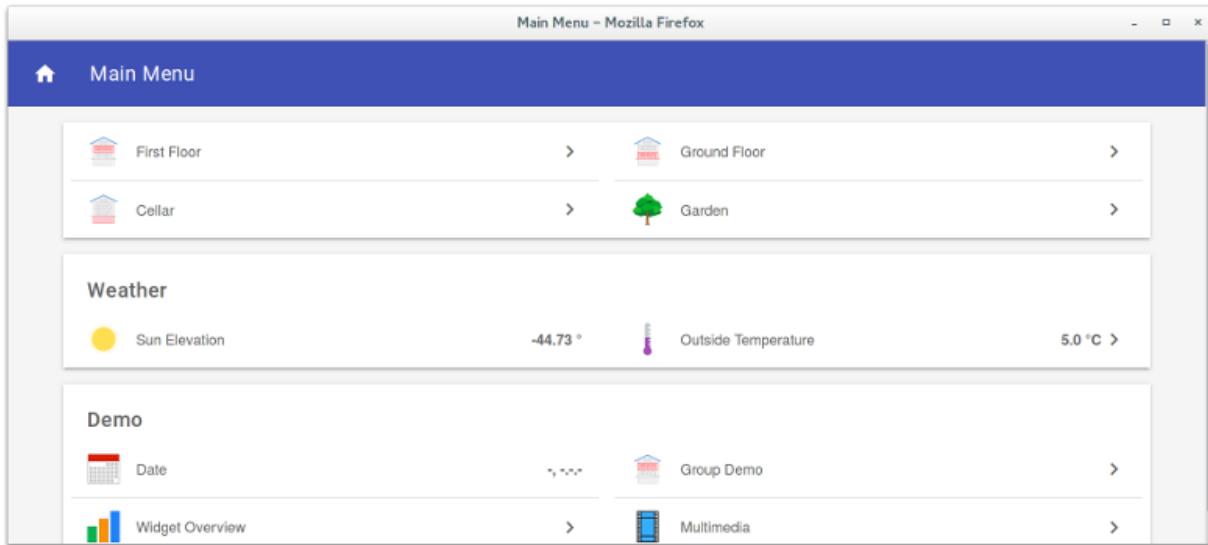
Εικόνα 4: openHAB initial setup.

Αν επλέξετε το Standard πακέτο, όταν εγκατασταθεί θα εμφανιστεί το πιο κάτω. Όπως βλέπετε στην εικόνα, έχει δημιουργηθεί το Home Builder που επιτρέπει τη δημιουργία sitemap, dashboard και γενικά ότι έχει να κάνει με το πως θα εμφανίζονται οι πληροφορίες στον χρήστη, το REST API με το οποίο μπορούμε να λαμβάνουμε πληροφορίες από το openHAB μέσω HTTP calls, το HABPANEL με το οποίο μπορούμε επίσης να δημιουργήσουμε και να διαχειριστούμε dashboards, το BASIC UI το οποίο είναι ένα απλό περιβάλλον απεικόνισης των sitemaps, και το Paper UI μέσω του οποίου θα κάνουμε τα configurations για τα bindings, thing και items.



Εικόνα 5: openHAB UIs.

Πιο κάτω βλέπετε ένα screenshot από το Basic UI.



Εικόνα 6: openHAB Basic UI.

Εδώ βλέπετε τα διάφορα endpoints που υποστηρίζονται από το REST API του openHAB όπως τα παρουσιάζει το REST API UI.

The screenshot shows the openHAB REST API documentation. At the top, there's a header with the openHAB logo, the URL <http://10.16.30.25:8080/rest/swagger.json>, and search fields for 'api\_key' and 'Explore'. Below the header, the title 'openHAB REST API' is displayed. A sidebar on the left lists various API categories: bindings, channel-types, config-descriptions, default, discovery, extensions, habpanel, inbox, and items. Under the 'items' category, several endpoints are listed with their methods, URLs, and descriptions:

- GET /items**: Get all available items.
- PUT /items**: Adds a list of items to the registry or updates the existing items.
- DELETE /items/{itemName}/members/{memberItemName}**: Removes an existing member from a group item.
- PUT /items/{itemName}/members/{memberItemName}**: Adds a new member to a group item.
- DELETE /items/{itemname}**: Removes an item from the registry.
- GET /items/{itemname}**: Gets a single item.

Εικόνα 7: openHAB REST API endpoints.

Το openHAB υποστηρίζει υπηρεσίες που επιτρέπουν την ενσωμάτωση με διάφορες τεχνολογίες, παράδειγμα το Amazon Alexa Skill για την διαχείριση του συστήματος με φωνητικές εντολές μέσω του Amazon Echo Dot και το Dropbox όπου τα αρχεία του server μπορούν να συγχρονιστούν στον λογαριασμό dropbox του χρήστη για να κρατάει back-up.

#### 4. OpenHAB Cloud Connector

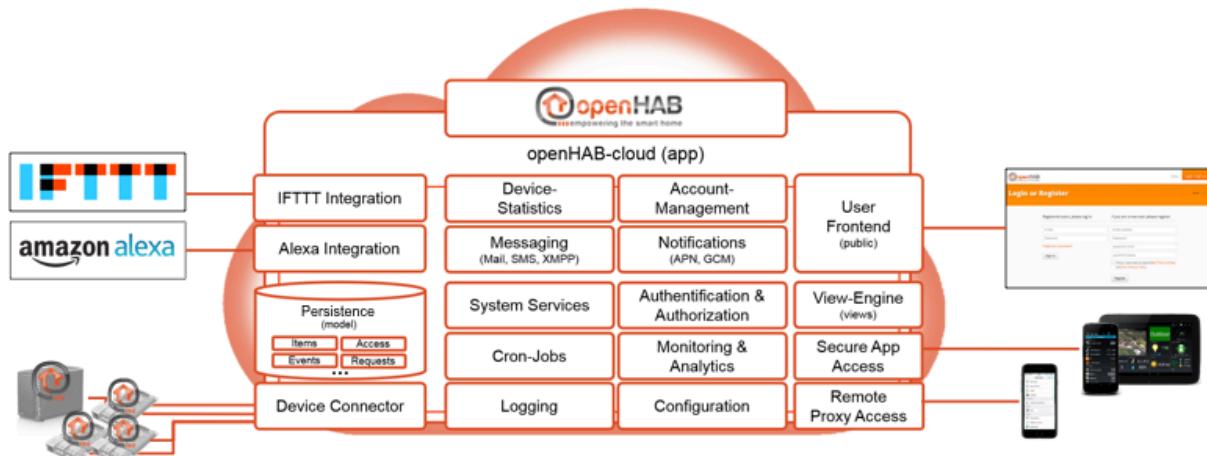
Το openHAB cloud connector είναι μια υπηρεσία που επιτρέπει στον χρήστη να συνδέει το openHAB server που τρέχει πάνω στο VM με κάποιο απομακρυσμένο openHAB Cloud application, όπως το [myopenHAB.org](http://myopenHAB.org), που αποτελεί μια openHAB cloud υπηρεσία που παρέχεται από το openHAB Foundation. Το openHAB Cloud Service (άρα και το connector) είναι χρήσιμο για διάφορες περιπτώσεις όπως:

- Επιτρέπει την απομακρυσμένη πρόσβαση στο τοπικό δίκτυο του openHAB χωρίς χρειάζεται να εκθέσει ports στο διαδίκτυο ή την απαίτηση ρύθμισης πολύπλοκης VPN σύνδεσης.
- Αποτελεί συνδετικός κρίκος στο Google Cloud Messaging (GCM) και Apple Push Notifications (APN) για την αποστολή notifications στις smartphone εφαρμογές.

- Δίνει δυνατότητες ενσωμάτωσης υπηρεσιών που απαιτούν OAuth2 πιστοποίηση έναντι ενός web server, όπως το IFTTT ή το Amazon Alexa Skills.
- Επιτρέπει καταχώρηση επιπλέον απομακρυσμένων χρηστών για το σύστημα.

Το openHAB Cloud είναι κυρίως βασισμένο στα εξής frameworks και τεχνολογίες:

- Node.js - Server - side Javascript - framework
- Express.js - Web application framework για το Node.js
- Nginx - Web server & reverse proxy
- MongoDB - NoSQL database
- redis - Session Manager & data structure server
- Socket.IO - Bi - directional communication μεταξύ web clients και servers



Εικόνα 8: openHAB cloud application.

## 4.1 Πιστοποίηση

Για την πιστοποίηση με το openHAB Cloud application, το τοπικό openHAB (στο VM) δημιουργεί δύο values, τα οποία πρέπει να δοθούν στις ρυθμίσεις λογαριασμού στο openHAB Cloud service. Το πρώτο είναι ένα μοναδικό identifier, που επιτρέπει να αναγνωρίσει openHAB runtime. Αποτελεί κάτι σαν username για την cloud πιστοποίηση. Το δεύτερο είναι ένα τυχαίο μυστικό κλειδί το οποίο εξυπηρετεί σαν κωδικός. Και τα δύο values είναι γραμμένα στο τοπικό file system. Αν χαθούν αυτά τα δύο για κάποιο λόγο, το openHAB θα δημιουργήσει 2 νέα αυτόματα. Σε αυτή την περίπτωση θα πρέπει να αλλάξουν και στις ρυθμίσεις λογαριασμού στο openHAB Cloud Service.

## 4.2 Εγκατάσταση openHAB cloud connector και διασύνδεσή του με το openHAB instance που είναι εγκατεστημένο στο VM

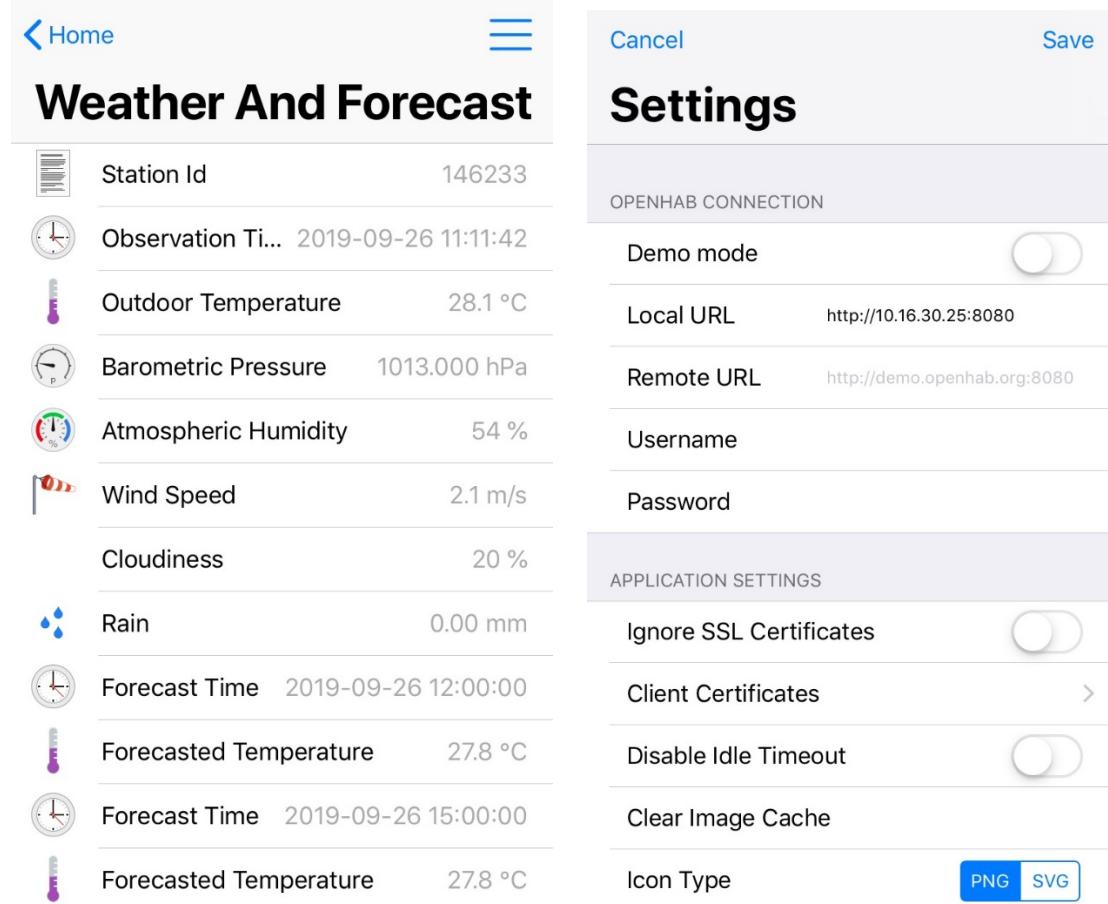
Ακολουθείστε τις οδηγίες στο APPENDIX 3 για να δημιουργήσετε τα προαναφερθέντα κλειδιά έτσι ώστε να διασυνδέσετε το openHAB στο VM σας με το cloud based openHAB application.

## 5. Android & iOS Applications

Το openHAB προσφέρει και εφαρμογές για iOS και Android που χρησιμοποιούν το REST API με σκοπό να κάνουν render τα sitemaps που έχει δημιουργήσει ο χρήστης. Τα χαρακτηριστικά που προσφέρουν οι εφαρμογές είναι τα εξής

- Έλεγχος του openHAB και του openHAB Cloud
- Λήψη notifications από το openHAB Cloud
- Αλλαγή των Items μέσω NFC tags
- Αποστολή φωνητικών εντολών στο openHAB
- Ανακάλυψη συσκευών και προσθήκη αυτών σαν Items
- Υποστηρίζει εντοιχισμένα tablets

Μπορείτε να κατεβάσετε την εφαρμογή ([Android](#), [iOS App Store](#)) και να την εγκαταστήσετε για να διαχειρίζεστε το openHAB που είναι εγκατεστημένο στο VM σας. Πιο κάτω βλέπετε κάποια screenshots από την εφαρμογή.



Εικόνα 9: Screenshots από το openHAB mobile app.

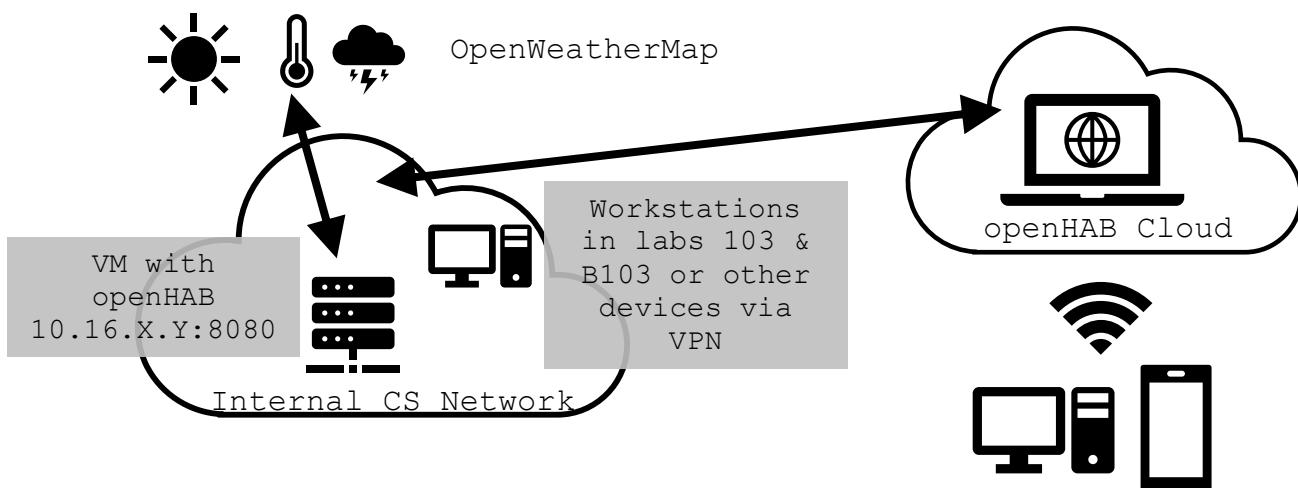
### Σύνδεση με OpenHAB Server

Προκειμένου να επιτευχθεί η σύνδεση της εφαρμογής με τον server του συστήματός μας πρέπει να πάτε στις ρυθμίσεις της εφαρμογής και να καταχωρίσετε την IP διεύθυνση του

server (αν η κινητή σας συσκευή είναι στο ίδιο δίκτυο με το openHAB server). Εναλλακτικά, για διαχείριση του openHAB μέσω κινητού από οποιοδήποτε σημείο πρέπει να δώσετε στην εφαρμογή τη διεύθυνση του cloud instance (<https://myopenhab.org>) και επιπλέον να καταχωρίσετε τα credentials σας (username & password) που δημιουργήσατε όταν κάνατε register στο cloud.

## 6. Ανάλυση Συστήματος Άσκησης

Σε αυτή την ενότητα θα παρουσιάσουμε το σύστημα που θα χρησιμοποιήσουμε στην άσκηση αυτή. Για σκοπούς απλοποίησης δεν θα χρησιμοποιήσουμε κάποιο hardware (π.χ. μικρο-ελεγκτή) που να στέλνει πληροφορίες στο openHAB εκτός από το κινητό μας τηλέφωνό ή/και τον προσωπικό μας υπολογιστή το οποίο θα χρησιμοποιηθεί για έλεγχο του openHAB μέσω openHAB cloud.



Εικόνα 10: Αρχιτεκτονική Συστήματος άσκησης.

Στην εικόνα 10 περιγράφεται η αρχιτεκτονική του συστήματος που θα υλοποιήσουμε. Κεντρικός κόμβος του συστήματος αυτοματισμού είναι το VM που έχει τον ρόλο του εξυπηρετητή με το openHAB server εγκατεστημένο. Επειδή μέσα στο κλειστό δίκτυο του Τμήματος Πληροφορικής δεν μπορούμε να έχουμε οποιαδήποτε άλλη πραγματική IoT συσκευή (hardware) για να τη διαχειρίζομαστε μέσω openHAB σαν “thing” θα χρησιμοποιήσουμε ένα πάροχο υπηρεσιών καιρού όπως είναι το [OpenWeatherMap](#). Τα βήματα για την εγκατάσταση και τη λήψη δεδομένων από το OpenWeatherMap είναι τα εξής:

A) **Εύρεση binding** →  
<https://www.openhab.org/addons/bindings/openweathermap/>

Το συγκεκριμένο binding υποστηρίζει 2 Things:

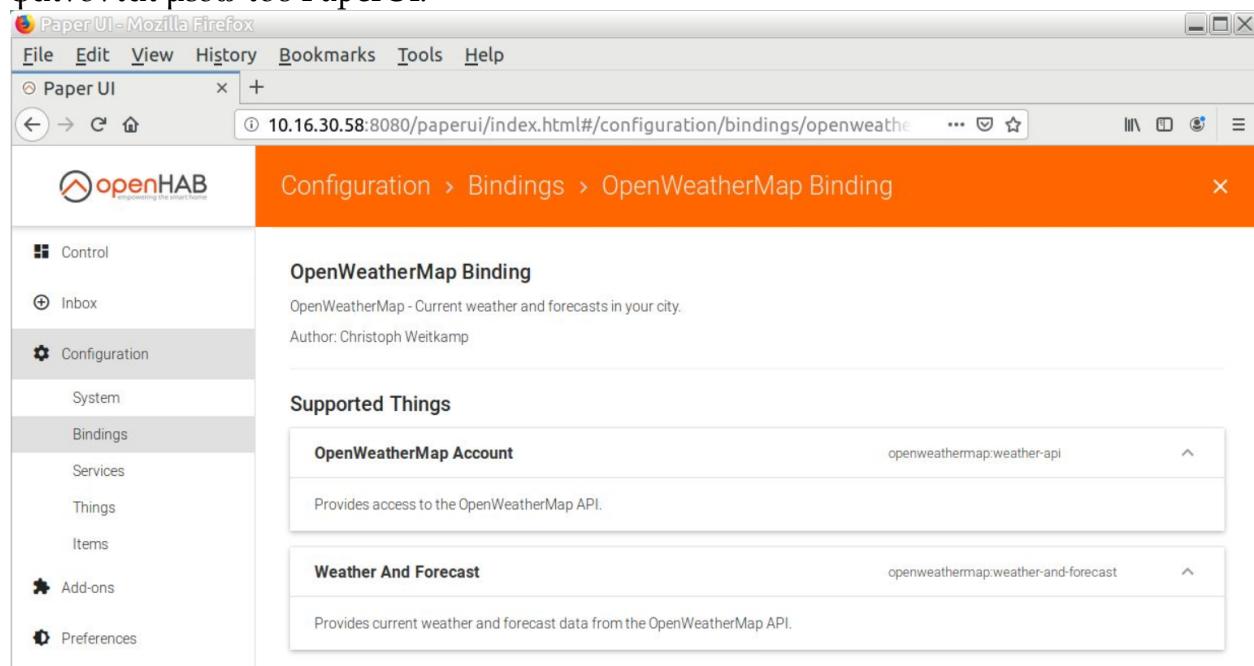
- OpenWeatherMap Account: Το πρώτο thing είναι το weather-api που αντιπροσωπεύει το λογαριασμό OpenWeatherMap. Πρέπει να δηλωθεί

υποχρεωτικά το API key<sup>1</sup> για να αποκτήσετε πρόσβαση στο OpenWeatherMap API.

- Current Weather And Forecast: Το δεύτερο thing είναι το weather-and-forecast που υποστηρίζει τις δωρεάν υπηρεσίες [current weather](#) και [5 day / 3 hour forecast](#) καθώς και την επί πληρωμή υπηρεσία [16 day / daily forecast](#) για μια συγκεκριμένη τοποθεσία. Απαιτούνται οι γεωγραφικές συντεταγμένες (longitude, latitude) της τοποθεσίας. Μπορούμε να προσθέσουμε όσα weather-and-forecast things θέλουμε για κάθε τοποθεσία ξεχωριστά.

## B) Εγκατάσταση binding

Το binding μπορεί να εγκατασταθεί στο openHAB μέσω της διαδικασίας που περιγράφεται στο APPENDIX 4 και μετά την εγκατάσταση τα 2 πιο πάνω things φαίνονται μέσω του PaperUI:



Εικόνα 11: Things που υποστηρίζονται από το OpenWeatherMap.

## Γ) Ορισμός και διαμόρφωση Thing

Αφού εγκαταστήσαμε το OpenWeatherMap binding στο openHAB, το επόμενο βήμα είναι να διαμορφώσουμε (configure) τα 2 things όπως για παράδειγμα καταχώρηση API key στο OpenWeatherMap Account και καταχώρηση πόλεων για συλλογή δεδομένων στο OpenWeatherMap Weather and Forecast.

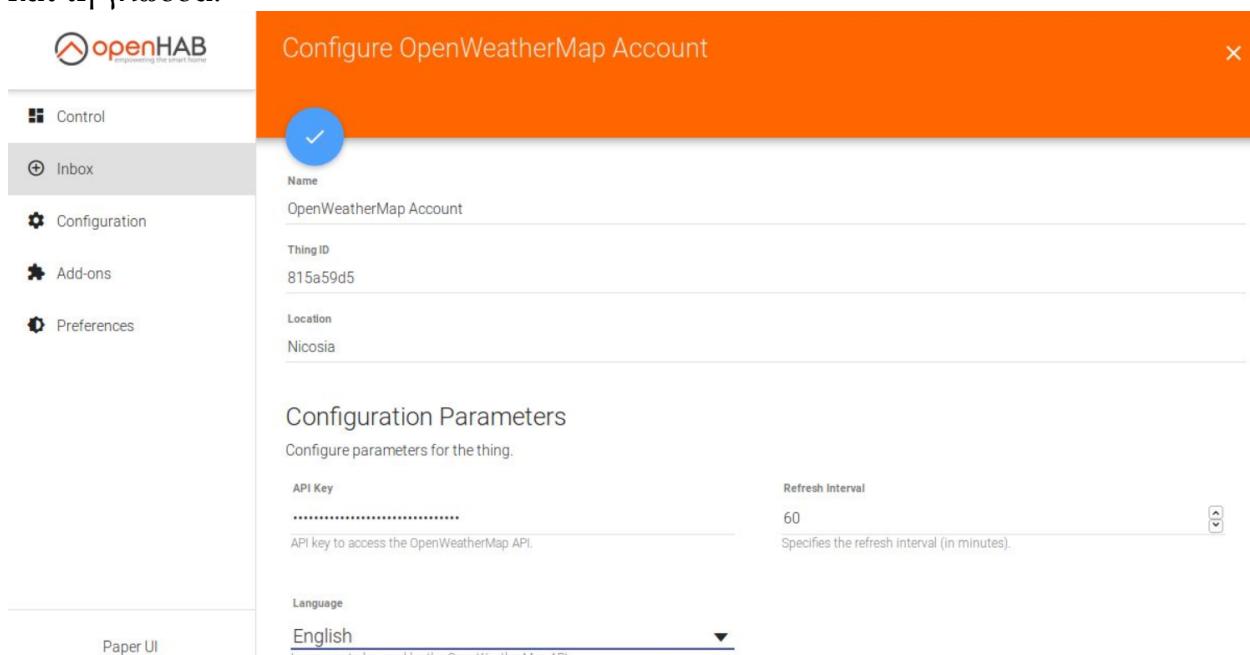
Αυτό γίνεται και πάλι μέσω του PaperUI, από το μενού Configuration → Things :

<sup>1</sup> Εγγραφείτε [εδώ](#) για έναν δωρεάν λογαριασμό με το OpenWeatherMap. Μόλις εγγραφείτε, θα λάβετε ένα APIID, ουσιαστικά το κλειδί για τη χρήση της υπηρεσίας. Χωρίς APIID, δεν θα μπορείτε να έχετε πρόσβαση σε κανένα από τα APIs. Χρειάζεται έως και 1 ώρα για να ενεργοποιήσετε το κλειδί API. Θα λάβετε ένα μήνυμα ηλεκτρονικού ταχυδρομείου επιβεβαίωσης, καθώς το κλειδί API είναι έτοιμο για εργασία.



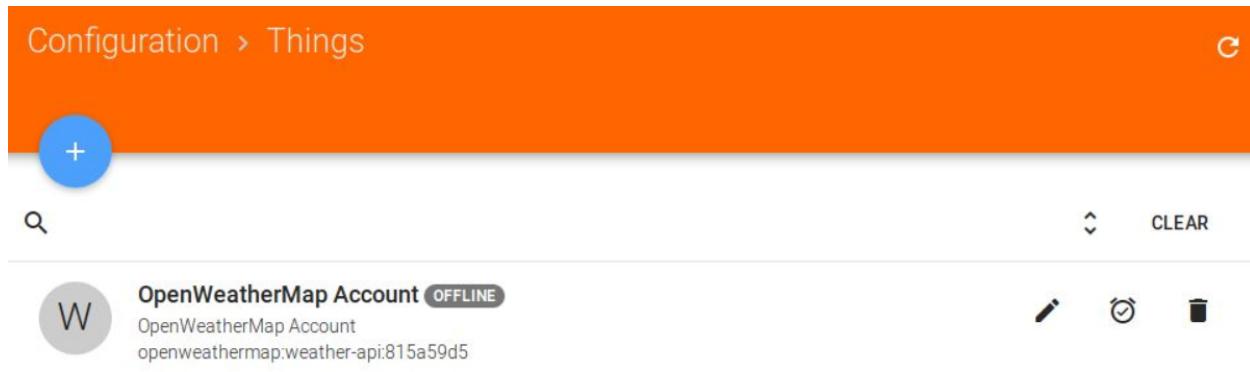
Εικόνα 12: Προσθήκη thing στο openHAB.

Στη συνέχεια, πατούμε το κουμπί + ή το ADD THINGS και επιλέγουμε το OpenWeatherMap Account και εισάγουμε την τοποθεσία μας (του openHAB), το API key και τη γλώσσα.



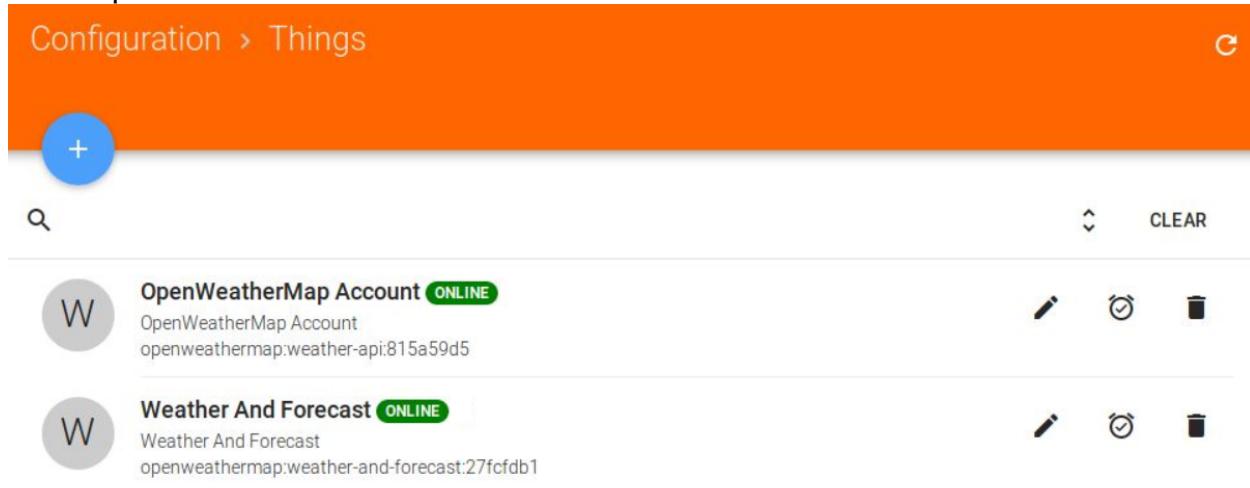
Εικόνα 13: Διαμόρφωση του OpenWeatherMap Account thing.

Μετά την αποθήκευση επιστρέφουμε στην οθόνη με τα εγκατεστημένα things και βλέπουμε αυτό που μόλις εγκαταστήσαμε.



Εικόνα 14: OpenWeatherMap Account thing.

Στη συνέχεια, προσθέτουμε και το 2<sup>o</sup> thing (Weather and Forecast), όπου στην τοποθεσία προσθέτουμε Nicosia καθώς και τις συντεταγμένες της περιοχής για την οποία θέλουμε τον καιρό.



Εικόνα 15: OpenWeatherMap Account & Weather Forecast things.

#### Δ) Εύρεση των καναλιών του Thing

Το επόμενο βήμα είναι η εύρεση των καναλιών που υποστηρίζει το κάθε thing. Το OpenWeatherMap Account δεν υποστηρίζει κάποια κανάλια. Το Weather And Forecast όμως υποστηρίζει αρκετά κανάλια όπως για παράδειγμα Station Id, Station Name, Outdoor Temperature κτλ. τα οποία μπορούμε να δούμε επιλέγοντας (όπως φαίνεται στην εικόνα 12) το αντίστοιχο thing:

Εικόνα 16: Διαθέσιμα κανάλια κάτω από το Weather Forecast thing.

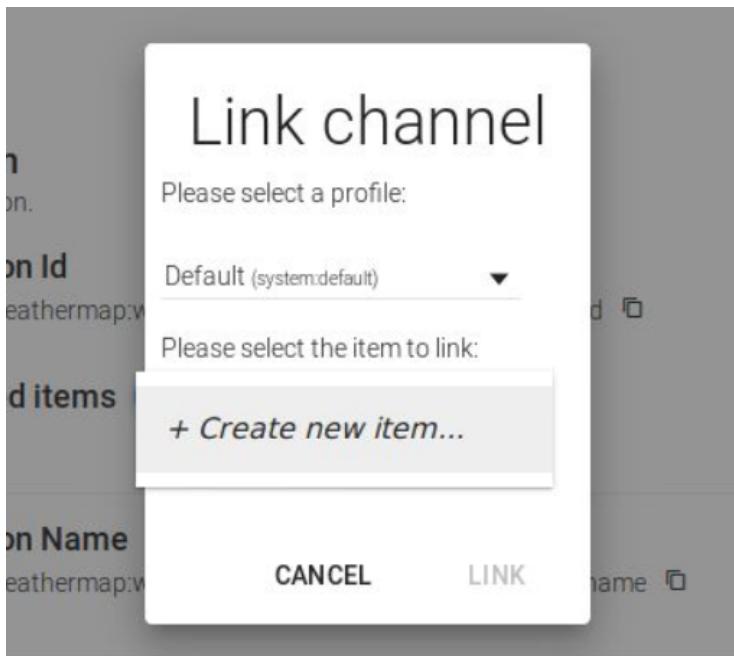
#### Ε) Προσθήκη Items και σύνδεση με το/τα κανάλι/α (επεξήγηση στον ορισμό Items)

Τα items μπορούν να συνδεθούν με ένα κανάλι (channel) για αλληλεπίδραση με τον έξω κόσμο. Μέσα από την ιστοσελίδα της προηγούμενης εικόνας, μπορούμε να δημιουργήσουμε items πάνω στα channels.

Για παράδειγμα πατώντας στο μπά στα δεξιά του κάθε channel εμφανίζεται η δυνατότητα προσθήκης item πά στο channel.

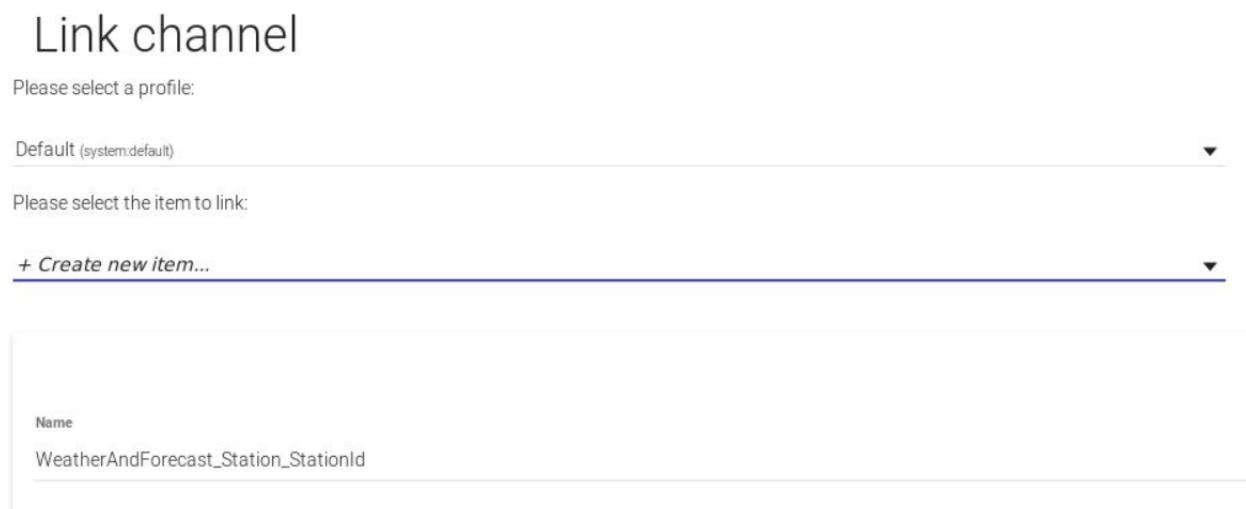
Εικόνα 17: Προσθήκη item πάνω σε channel.

Πατώντας το + εμφανίζεται ένα pop up παράθυρο με το οποίο δημιουργείται ένα νέο item.



Εικόνα 18: Δημιουργία νέου item.

Μπορούμε να αφήσουμε το νέο item με τις default τιμές που φαίνονται στην πιο κάτω εικόνα.



Label	Category
Station Id	
Type	String
Parent groups	<b>OPEN ITEM</b>
autoupdate - Auto Update	

Εικόνα 19: Default τιμές νέου item πάνω στο Station Id channel.

Το νέο item θα δημιουργηθεί και θα διασυνδεθεί με το channel πατώντας το κουμπί Link στο κάτω μέρος της ιστοσελίδας. Το νέο item φαίνεται και από τη σελίδα των channels:

**Weather Station** ×  
 This is a weather station.

**Station Id** ×  
 openweathermap:weather-and-forecast:27fcfdb1:station#id ✖  
 String

**Linked items** +

S Station Id (WeatherAndForecast\_Station\_StationId) ✎ ✖

Εικόνα 20: Station Id item διασυνδεδεμένο με το Station Id channel.

αλλά και από τη σελίδα που δείχνει όλα τα items μέσα στο PaperUI.

Configuration > Items

**Station Id**  
 WeatherAndForecast\_Station\_StationId ✖  
 String

Εικόνα 21: openHAB items.

## 7. openHAB REST API

Εκτός από τις γραφικές διεπαφές του openHAB (π.χ. PAPERUI, openHAB cloud, openHAB mobile apps), κάποιος μπορεί να αλληλοεπιδράσει με την πλατφόρμα μέσω του [openHAB REST API](#).

Μπορούμε να αλληλεπιδράσουμε μέσω του openHAB REST API χρησιμοποιώντας τον φυλλομετρητή ή το εργαλείο curl από το terminal. Όλα τα διαθέσιμα endpoints για το REST API μπορείτε να τα δείτε μέσω του φυλλομετρητή όπως είδαμε στην εικόνα 7.

Για παράδειγμα τα διαθέσιμα things του openHAB φαίνονται από το endpoint: [http://VM\\_IP\\_ADDR:8080/rest/things](http://VM_IP_ADDR:8080/rest/things) ή μέσω της εντολής:

```
curl -v http://VM\_IP\_ADDR:8080/rest/things
```

```
ubuntu@epl421-1:~$ curl -v 10.16.30.58:8080/rest/things
*   Trying 10.16.30.58...
* TCP_NODELAY set
* Connected to 10.16.30.58 (10.16.30.58) port 8080 (#0)
> GET /rest/things HTTP/1.1
> Host: 10.16.30.58:8080
> User-Agent: curl/7.58.0
> Accept: /*
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Transfer-Encoding: chunked
< Server: Jetty(9.4.11.v20180605)
<

[{"statusInfo": {"status": "ONLINE", "statusDetail": "NONE"}, "editable": true, "label": "Weather And Forecast", "bridgeUID": "openweathermap:weather-apib815a59d5", "configuration": {"forecastDays": 0, "location": "35.1856,33.3823", "forecastHours": 24}, "properties": {}, "UID": "openweathermap:weather-and-forecast:27fcfdb1", "thingTypeUID": "openweathermap:weather-and-forecast", "channels": [{"linkedItems": ["WeatherAndForecast_StationId"], "uid": "openweathermap:weather-and-forecast:27fcfdb1:station#id", "id": "station#id", "channelTypeUID": "openweathermap:station-id", "itemType": "String", "kind": "STATE", "label": "Station Id", "description": "Id of the weather station or the city.", "defaultTags": [], "properties": {}, "configuration": {}}, {"linkedItems": [{"uid": "openweathermap:weather-and-forecast:27fcfdb1:station#name", "id": "station#name", "channelTypeUID": "openweathermap:station-name", "itemType": "String", "kind": "STATE", "label": "Station Name", "description": "Name of the weather station or the city.", "defaultTags": [], "properties": {}, "configuration": {}}, {"linkedItems": [{"uid": "openweathermap:weather-and-forecast:27fcfdb1:station#location", "id": "station#location", "channelTypeUID": "system:location", "itemType": "Location", "kind": "STATE", "label": "Location", "description": "Location of the weather station or the city.", "defaultTags": [], "properties": {}, "configuration": {}}, {"linkedItems": [{"uid": "openweathermap:weather-and-forecast:27fcfdb1:current#time-stamp", "id": "current#time-stamp", "channelTypeUID": "openweathermap:time-stamp", "itemType": "DateTime", "kind": "STATE", "label": "Observation Time", "description": "Time of data observation.", "defaultTags": [], "properties": {}, "configuration": {}}, {"linkedItems": [{"uid": "openweathermap:weather-and-forecast:27fcfdb1:current#condition", "id": "current#condition", "channelTypeUID": "openweathermap:condition", "itemType": "String", "kind": "STATE", "label": "Weather Condition", "description": "Current weather condition.", "defaultTags": [], "properties": {}, "configuration": {}}, {"linkedItems": [{"uid": "openweathermap:weather-and-forecast:27fcfdb1:current#icon", "id": "current#icon", "channelTypeUID": "openweathermap:condition-icon", "itemType": "Image", "kind": "STATE", "label": "Icon", "description": "Icon representing the weather condition.", "defaultTags": [], "properties": {}, "configuration": {}}, {"linkedItems": [{"uid": "openweathermap:weather-and-forecast:27fcfdb1:current#icon", "id": "current#icon", "channelTypeUID": "openweathermap:condition-icon", "itemType": "Image", "kind": "STATE", "label": "Icon", "description": "Icon representing the weather condition."}, {"linkedItems": [{"uid": "openweathermap:weather-and-forecast:27fcfdb1:current#icon", "id": "current#icon", "channelTypeUID": "openweathermap:condition-icon", "itemType": "Image", "kind": "STATE", "label": "Icon", "description": "Icon representing the weather condition."}]]}]]}
```

Εικόνα 22: Ανάκτηση των things μέσω REST API call από το terminal μέσω curl.

The screenshot shows a JSON viewer interface with the URL `10.16.30.58:8080/rest/things`. The JSON structure is as follows:

```
JSON Raw Data Headers
Save Copy Collapse All Expand All Filter JSON
▼ 0:
  ▼ statusInfo:
    status: "ONLINE"
    statusDetail: "NONE"
    editable: true
    label: "Weather And Forecast"
    bridgeUID: "openweathermap:weather-api:815a59d5"
  ▼ configuration:
    forecastDays: 0
    location: "35.1856,33.3823"
    forecastHours: 24
    properties: {}
    UID: "openweathermap:weather-and-forecast:27fcfdb1"
    thingTypeUID: "openweathermap:weather-and-forecast"
  ▼ channels:
    ▼ 0:
      ▼ linkedItems:
        0: "WeatherAndForecast_Station_StationId"
      ▼ uid: "openweathermap:weather-and-forecast:27fcfdb1:station#id"
        id: "station#id"
        channelTypeUID: "openweathermap:station-id"
        itemType: "String"
        kind: "STATE"
        label: "Station Id"
        description: "Id of the weather station or the city."
        defaultTags: []
        properties: {}
        configuration: {}
```

Εικόνα 23: Ανάκτηση των things μέσω REST API call από τον φυλλομετρητή.

Ομοίως και για τα items, το REST API endpoint είναι είναι:

[http://VM\\_IP\\_ADDR:8080/rest/items](http://VM_IP_ADDR:8080/rest/items)

```

0:
  link: "http://10.16.30.58:8080/rest/items/WeatherAndForecast_Current_WindSpeed"
  state: "4.75 m/s"
  stateDescription:
    pattern: "%.1f %unit%"
    readOnly: true
    options: []
    editable: true
    type: "Number:Speed"
    name: "WeatherAndForecast_Current_WindSpeed"
    label: "Wind Speed"
    category: "Wind"
    tags: []
    groupNames: []

1:
  link: "http://10.16.30.58:8080/rest/items/WeatherAndForecast_Current_OutdoorTemperature"
  state: "28.08 °C"
  stateDescription:
    pattern: "%.1f %unit%"
    readOnly: true
    options: []
    editable: true
    type: "Number:Temperature"
    name: "WeatherAndForecast_Current_OutdoorTemperature"
    label: "Outdoor Temperature"
    category: "Temperature"
    tags: []
    groupNames: []

```

Εικόνα 24: Ανάκτηση των items μέσω REST API call από τον φυλλομετρητή.

Όπως βλέπετε πω πάνω, το κάθε item έχει διάφορες παραμέτρους όπως link, state, type, name, label, category κτλ. Το link είναι το REST endpoint που μας δίνει πληροφορίες για το συγκεκριμένο item. Αν πιάσουμε για παράδειγμα το 1<sup>o</sup> item, και καλέσουμε το REST endpoint του θα πάρουμε:

The screenshot shows a JSON editor interface with the URL `10.16.30.58:8080/rest/items/WeatherAndForecast_Current_WindSpeed`. The JSON structure is as follows:

```
JSON Raw Data Headers
Save Copy Collapse All Expand All Filter JSON
{
  "link": "http://10.16.30.58:8080/rest/items/WeatherAndForecast_Current_WindSpeed",
  "state": "4.75 m/s",
  "stateDescription": {
    "pattern": "%.1f %unit%",
    "readOnly": true,
    "options": [],
    "editable": true,
    "type": "Number:Speed",
    "name": "WeatherAndForecast_Current_WindSpeed",
    "label": "Wind Speed",
    "category": "Wind",
    "tags": [],
    "groupNames": []
  }
}
```

Εικόνα 25: REST endpoint για το item WeatherAndForecast\_Current\_WindSpeed.

Επίσης, για την ανάκτηση της τιμής (state) του πιο πάνω item το REST endpoint είναι:

`http://VM_IP_ADDR:8080/rest/items/  
WeatherAndForecast_Current_OutdoorTemperature/state`

The screenshot shows a JSON editor interface with the URL `10.16.30.58:8080/rest/items/WeatherAndForecast_Current_WindSpeed/state`. The state value is displayed as `4.75 m/s`.

Εικόνα 26: REST endpoint για το state του item WeatherAndForecast\_Current\_WindSpeed.

## APPENDIX 1 – openHAB installation instructions for Ubuntu

### Prerequisites

Please download and install the Java 8 version<sup>2</sup> of the JVM. openHAB is known not to work on Java 9 and Java 10.

```
sudo apt install openjdk-8-jdk-headless
```

Check which JDK-Versions are available on your system perform:

```
sudo update-java-alternatives --list
```

you get something like this:

```
ubuntu@ep1421-1:/$ sudo update-alternatives --list
update-alternatives: --list needs <name>

Use 'update-alternatives --help' for program usage information.
ubuntu@ep1421-1:/$ sudo update-java-alternatives --list
java-1.11.0-openjdk-amd64      1111      /usr/lib/jvm/java-1.11.0-openjdk-amd64
java-1.8.0-openjdk-amd64       1081      /usr/lib/jvm/java-1.8.0-openjdk-amd64
```

Make version 8 your default Java:

```
sudo update-alternatives --config java
```

you get something like this:

```
There are 2 choices for the alternative java (providing /usr/bin/java).
Selection    Path                                Priority  Status
-----*
* 0          /usr/lib/jvm/java-11-openjdk-amd64/bin/java   1111    auto mode
  1          /usr/lib/jvm/java-11-openjdk-amd64/bin/java   1111    manual mode
  2          /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java  1081    manual mode

Press <enter> to keep the current choice[*], or type selection number: 2
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java to provide /usr/bin/java (java) in manual mode
ubuntu@ep1421-1:/$ java -version
openjdk version "1.8.0_222"
OpenJDK Runtime Environment (build 1.8.0_222-8u222-b10-1ubuntu1~18.04.1-b10)
OpenJDK 64-Bit Server VM (build 25.222-b10, mixed mode)
```

Pay attention to the asterisk before the numbers on the left and if the correct one is not set choose the correct one by typing the number of it and pressing enter. Enter '2' to select the java-8-openjdk-amd64.

Repeat the previous operation with default Javac:

<sup>2</sup> You should use the 32-bit version of the JVM for ARM platforms, even on 64-bit operating systems if serial bindings need to be installed. Serial connections won't work with a 64-bit JVM, preventing bindings like Z-Wave from functioning. However, serial bindings are outside of the scope of the current assignment.

```
sudo update-alternatives --config javac
```

```
ubuntu@epl421-1:~$ sudo update-alternatives --config javac
There is only one alternative in link group javac (providing /usr/bin/javac): /usr/lib/jvm/java-8-openjdk-amd64/bin/javac
Nothing to configure.
```

## Installation

Add the openHAB 2 Bintray repository key to your package manager and allow Apt to use the HTTPS Protocol:

```
wget -qO - 'https://bintray.com/user/downloadSubjectPublicKey?username=openhab'
| sudo apt-key add -
sudo apt-get install apt-transport-https
```

Add the openHAB 2 Stable Repository to your systems apt sources list:

```
echo 'deb https://dl.bintray.com/openhab/apt-repo2 stable main' | sudo tee
/etc/apt/sources.list.d/openhab2.list
```

Next, resynchronize the package index:

```
sudo apt-get update
```

Now install openHAB with the following command:

```
sudo apt-get install openhab2
```

When you choose to install an add-on, openHAB will download it from the internet on request. If you plan on disconnecting your machine from the internet, then you will want to also install the add-ons package.

```
sudo apt-get install openhab2-addons
```

Configure openHAB to start automatically on system start:

```
sudo /bin/systemctl daemon-reload
```

```
sudo /bin/systemctl enable openhab2.service
```

Start openhab2 by executing:

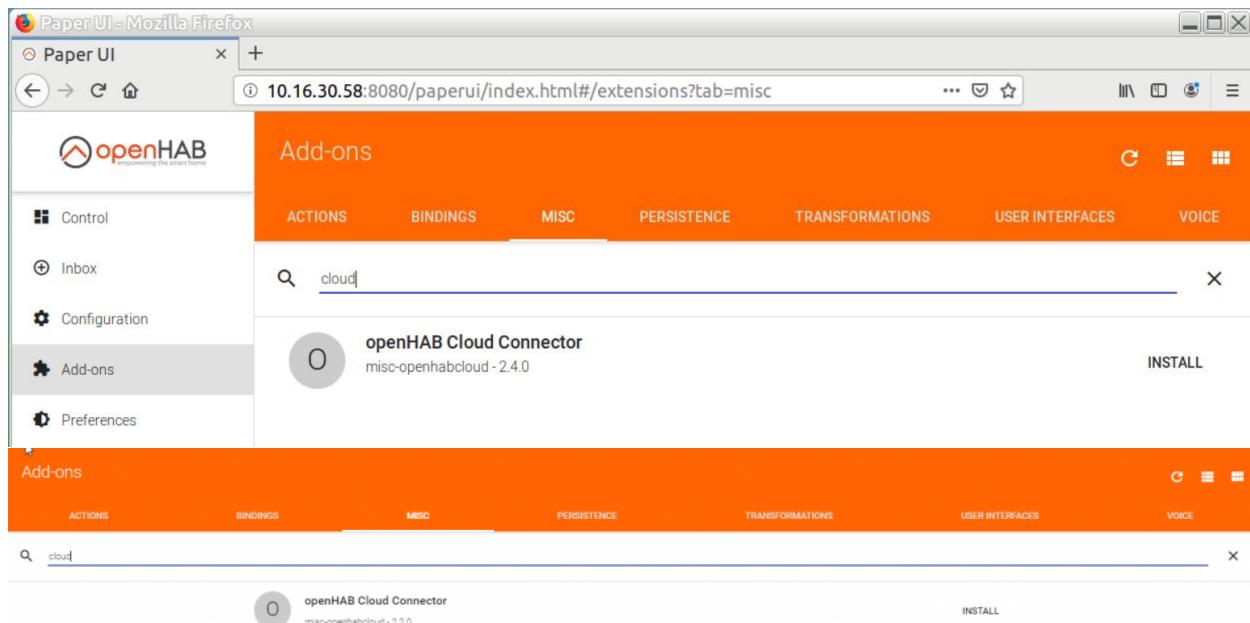
```
sudo /bin/systemctl start openhab2.service
```

## File Locations

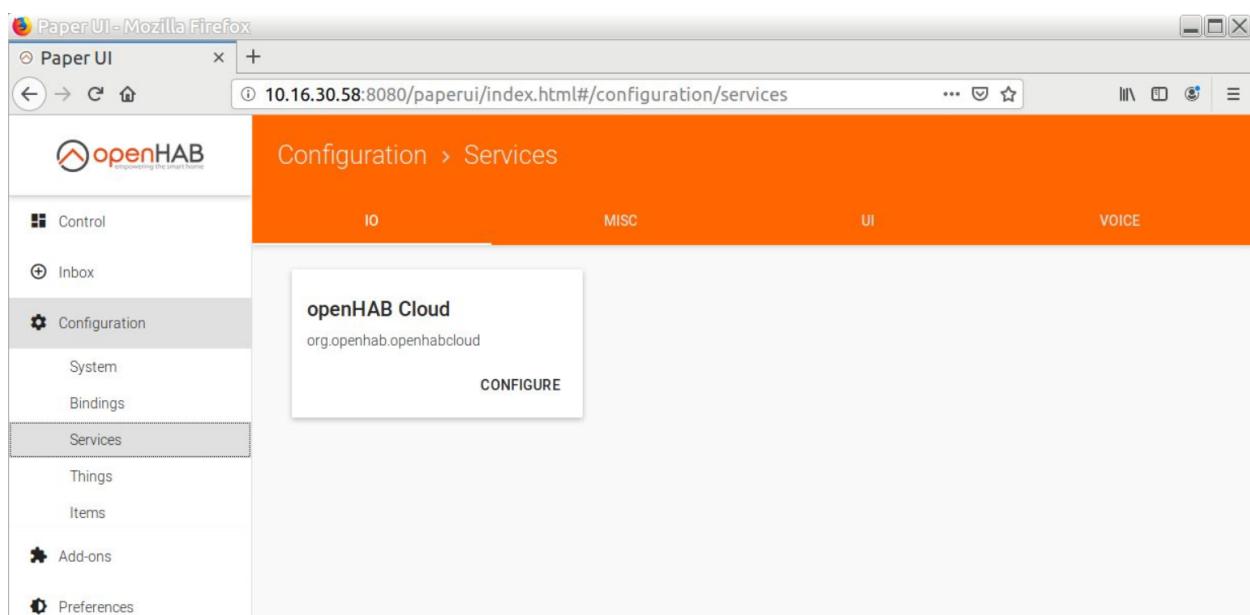
openHAB application	/usr/share/openhab2
Additional add-on files	/usr/share/openhab2/addons
Site configuration	/etc/openhab2
Log files	/var/log/openhab2
Userdata like rrd4j databases	/var/lib/openhab2
Backups folder	/var/lib/openhab2/backup
Service configuration	/etc/default/openhab2

## APPENDIX 2 – Install & configure openHAB cloud connector

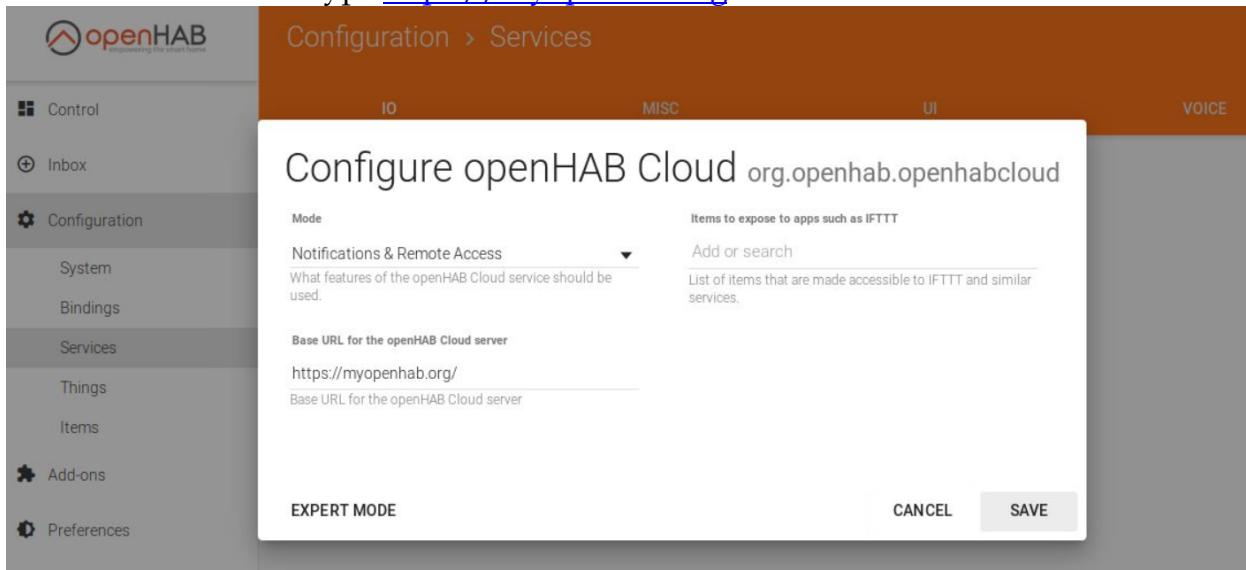
Open PaperUI ([http://VPS\\_name:8080/paperui/index.html#/extensions](http://VPS_name:8080/paperui/index.html#/extensions)) and go to *Add-ons*. \*\*\* *Please make sure that the port is open on your VM with iptables* \*\*\*  
In the section *Misc*, look for OpenHab Cloud Connector and install it. It should take just a few seconds.



Once it is installed, in the left column navigation go to configuration and then the services sub tab. In the openHAB cloud box click configure.



In the mode field click the drop down menu and select Notifications and Remote Access.  
In the base URL field type <https://myopenhab.org> and click save.



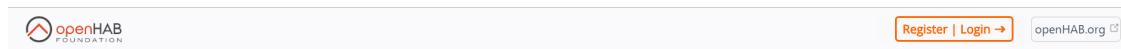
To authenticate with the openHAB Cloud service the openHAB cloud connector add-on generates two values when the add-on is installed. These values need to be entered in your account settings of the openHAB Cloud service. The first one is a unique identifier, which allows to identify your runtime. One can think of it as something similar like a username for the cloud authentication. The second one is a random secret key which serves as a password. Both values are written to the local file system. If you lose these files for some reason, openHAB will automatically generate new ones. You will then have to reconfigure UUID and secret in the openHAB Cloud service under the *My account* section.

Get Secret and UUID from your OpenHab Setup

```
cat /var/lib/openhab2/uuid
```

```
cat /var/lib/openhab2/openhabcloud/secret
```

Go to <https://www.myopenhab.org/> and register using the following 2 information.



myopenHAB is an instance of the [openHAB Cloud service](#), which is hosted by the [openHAB Foundation e.V.](#)



If everything works ok you will see that your openhab (on VM) is online and connected to openhab cloud:

A screenshot of the openHAB 2.x dashboard. At the top, there is a navigation bar with the openHAB logo, Home, Items, Event log, Notifications, and Online. The 'Online' button is highlighted in green. Below the navigation bar is a large orange header with the word 'Home' in white. The main content area is white and contains the text 'You are using openHAB 2.x. Click here to access your openHAB's dashboard' in orange. At the bottom of the page is a footer with the text 'Copyright © 2018 by the openHAB Community and the openHAB Foundation'.

And you can access the dashboard of your openHAB via cloud by clicking to "Click here to access your openHAB's dashboard"

A screenshot of the openHAB 2.x cloud dashboard. The top navigation bar includes the openHAB logo, Home, Items, Event log, Notifications, and Online. Below the navigation bar is a large orange header with the word 'Home' in white. The main content area displays four different UI options: 'HOME BUILDER', 'HABPANEL', 'BASIC UI', and 'PAPER UI', each showing a preview of its respective interface. At the bottom of the page is a footer with the text 'Getting started? Please refer to the [online documentation](#)'.

## APPENDIX 3 – Install OpenWeatherMap Binding

On the PaperUI, select the Add-ons menu, and on the Bindings section (tab) search for openWeatherMap to install the binding.

