



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ  
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

## ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

---

Μελέτη και συγγραφή οδηγιών χρήσης  
διαδικτυακής πλατφόρμας έξυπνων δικτύων

---

Δεληγεώργης Αλέξανδρος

A.M.: HN07499

Επιβλέπων Καθηγητής : Χριστοφορίδης Γεώργιος  
Συνεπιβλέπων Υποψήφιος Διδάκτωρ : Φιλιππίδης Σταύρος

(Υπογραφή)

.....

Δεληγεώργης Αλέξανδρος

Ηλεκτρολόγος Μηχανικός Τ.Ε., ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

© 2022 – All rights reserved

## ΠΕΡΙΛΗΨΗ

Ο βασικός στόχος αυτής της πτυχιακής εργασίας είναι η δημιουργία αναλυτικών οδηγιών χρήσης για την πλατφόρμα SMART (<https://smart.ece.uowm.gr>) και επιπλέον χρήσιμων πληροφοριών που βοηθούν στην κατανόηση της λειτουργίας και της χρηστικότητας της.

Η πτυχιακή περιλαμβάνει αναλυτικές πληροφορίες για όλες τις επιλογές της πλατφόρμας SMART, η οποία αποτελείται από τα ακόλουθα τμήματα : βάση δεδομένων μετρήσεων, λήψη και παρουσίαση δεδομένων και λήψη, απόκτηση δεδομένων σε πραγματικό χρόνο από αισθητήρες και API, διεπαφή API, σύστημα εποπτείας SCADA, online εργαλεία για προσομοίωση κυκλωμάτων, προσομοίωση έξυπνου δικτύου, βελτιστοποίηση PV και BESS, πλατφόρμα γραφικών που βασίζεται στη γλώσσα VEGA και δεδομένα εργαστηρίων Power-hardware-in-the-loop & Superconductivity.

Αρχικά στο πρώτο κεφάλαιο παρουσιάζονται όλα τα πακέτα λογισμικού που χρησιμοποιήθηκαν για την δημιουργία της πλατφόρμας. Στο δεύτερο κεφάλαιο παρουσιάζεται το κύριο μέρος δηλαδή η Database, που είναι η πρώτη επιλογή της πλατφόρμας και στην συνέχεια πραγματοποιείται μια λεπτομερή ανάλυση όλων των λειτουργιών με την σειρά που παρατίθενται στο μενού επιλογής της πλατφόρμας.

Η πλατφόρμα SMART δημιουργήθηκε το 2022 για τις ανάγκες των ερευνητικών έργων BERLIN, PV-estia και MOST από τους ερευνητές που εργάστηκαν σε αυτά τα έργα και αναβαθμίζεται σταδιακά με νέες προσθήκες ώστε να περιέχει ακόμη περισσότερες πληροφορίες και περισσότερες δυνατότητες.

## **ABSTRACT**

The main goal of this thesis is to create detailed user instructions for the SMART platform (<https://smart.ece.uowm.gr>) and additional useful information that helps to understand its operation and usability.

The thesis includes detailed information on all options of the SMART platform, which consists of the following sections: measurement database, data acquisition and presentation and download, real-time data acquisition from sensors and APIs, API interface, SCADA supervisory system, online tools for circuit simulation, smart grid simulation, PV and BESS optimization, VEGA language-based graphics platform and Power-hardware-in-the-loop & Superconductivity lab data.

Initially, the first chapter presents all the software packages that were used to create the platform. In the second chapter, the main part is initially presented that is the “Database”, which is the first choice of the platform. A detailed analysis of all functions is also made in the order listed in the platform selection menu.

The SMART platform was created in 2022 for the needs of the BERLIN, PV-estia and MOST research projects by the researchers who worked on these projects and is gradually upgraded with new additions to contain even more information and more features.

# ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ .....	3
ABSTRACT.....	4
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ.....	5
ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ .....	6
ΠΙΝΑΚΑΣ ΠΙΝΑΚΩΝ.....	7
ΕΙΣΑΓΩΓΗ.....	8
ΚΕΦΑΛΑΙΟ 1 ΤΑ ΛΟΓΙΣΜΙΚΑ ΠΑΚΕΤΑ ΤΗΣ ΠΛΑΤΦΟΡΜΑΣ SMART.....	9
1.1.1 Λειτουργικό σύστημα Debian.....	10
1.1.2 Βάση δεδομένων Mongoddb .....	11
1.1.3 Βάση δεδομένων mySQL .....	11
1.1.4 Γλώσσα προγραμματισμού HTML.....	12
1.1.5 Γλώσσα προγραμματισμού Javascript.....	13
1.1.6 Διακομιστής Apache.....	14
1.1.7 Γλώσσα προγραμματισμού PHP.....	15
1.1.8 Γλώσσα προγραμματισμού Python.....	16
ΚΕΦΑΛΑΙΟ 2 ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ Smart.ece.uowm.gr..	17
2.1 Παρουσίαση του τμήματος της Database .....	18
2.1.1 Τι είναι μια βάση δεδομένων (Database).....	18
2.1.2 Single data analysis.....	18
2.1.3 Upload measurements.....	21
2.1.4 Direct data download .....	22
2.1.5 Api code examples.....	24
2.2 Σύστημα εποπτείας Living lab SCADA .....	26
2.2.1 Device status .....	26
2.2.2 Status maps .....	27
2.3 Online Tools.....	29

2.3.1 Προσομοίωση κυκλωμάτων (Circuit simulation).....	29
2.3.2 Stores Tool.....	33
2.3.3.1 Graphing Platform (Γραφική πλατφόρμα Vega) .....	37
2.3.3.2 Παράδειγμα κώδικα διαγραμμάτων VEGA.....	39
ΣΥΜΠΕΡΑΣΜΑΤΑ .....	48
ΒΙΒΛΙΟΓΡΑΦΙΑ .....	49

## ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1 Αρχική οθόνη πλατφόρμας SMART (smart.ece.uowm.gr).....	9
Εικόνα 2 Single data analysis (Smart.ece.uowm.gr) .....	19
Εικόνα 3 Single data analysis τμήμα επιλογής κατηγορίας.....	19
Εικόνα 4 Single data analysis τμήμα επιλογής μετρητικού σταθμού .....	20
Εικόνα 5 Single data analysis τμήμα επιλογής μετρητικού στοιχείου.....	20
Εικόνα 6 Single data analysis αποτέλεσμα επιλογής Get metric analysis.....	21
Εικόνα 7 Upload measurements τμήμα επιλογής σταθμού .....	21
Εικόνα 8 Αρχείο επιλογής Download template file.....	22
Εικόνα 9 Direct data download (Smart.ece.uowm.gr) τμήμα επιλογής σταθμού.....	23
Εικόνα 10 Direct data download τμήμα επιλογής μετρητικών στοιχείων .....	23
Εικόνα 11 Παράδειγμα λειτουργίας ενός API.....	24
Εικόνα 12 Κατάσταση συσκευών Device Status.....	26
Εικόνα 13 PHP-Server monitor .....	27
Εικόνα 14 Status maps (Smart.ece.uowm.gr).....	29
Εικόνα 15 Raspberry PI.....	29
Εικόνα 16 (Smart.ece.uowm.gr) επιλογή προσομείωσης κυκλώματος.....	30
Εικόνα 17 Circuit Simulation επιλογή Draw .....	31
Εικόνα 18 Circuit Simulation επιλογή Scores.....	31
Εικόνα 19 Circuit Simulation τμήμα επιλογών (Options) .....	32
Εικόνα 20 Circuit Simulation – Circuits – Active filters.....	32
Εικόνα 21 Circuit Simulation – Circuits – Flash ADC/TTL Inverter.....	32
Εικόνα 22 Online tools (Smart.ece.uowm.gr) τμήμα Stores tool .....	33
Εικόνα 23 Stores tool επιλογή PV system .....	34
Εικόνα 24 Stores Tool επιλογή Consumption data.....	34
Εικόνα 25 Stores Tool επιλογή Electricity Costs .....	35
Εικόνα 26 Stores Tool επιλογή Storage System .....	35
Εικόνα 27 Stores Tool επιλογή Policy .....	36
Εικόνα 28 Stores tool επιλογή Financial .....	36
Εικόνα 29 Γραμμικό Διάγραμμα – Line Chart .....	42
Εικόνα 30 Διαγράμμα ακτινωτής διάταξης δέντρου - Radial Tree Layout chart .....	47

## **ΠΙΝΑΚΑΣ ΠΙΝΑΚΩΝ**

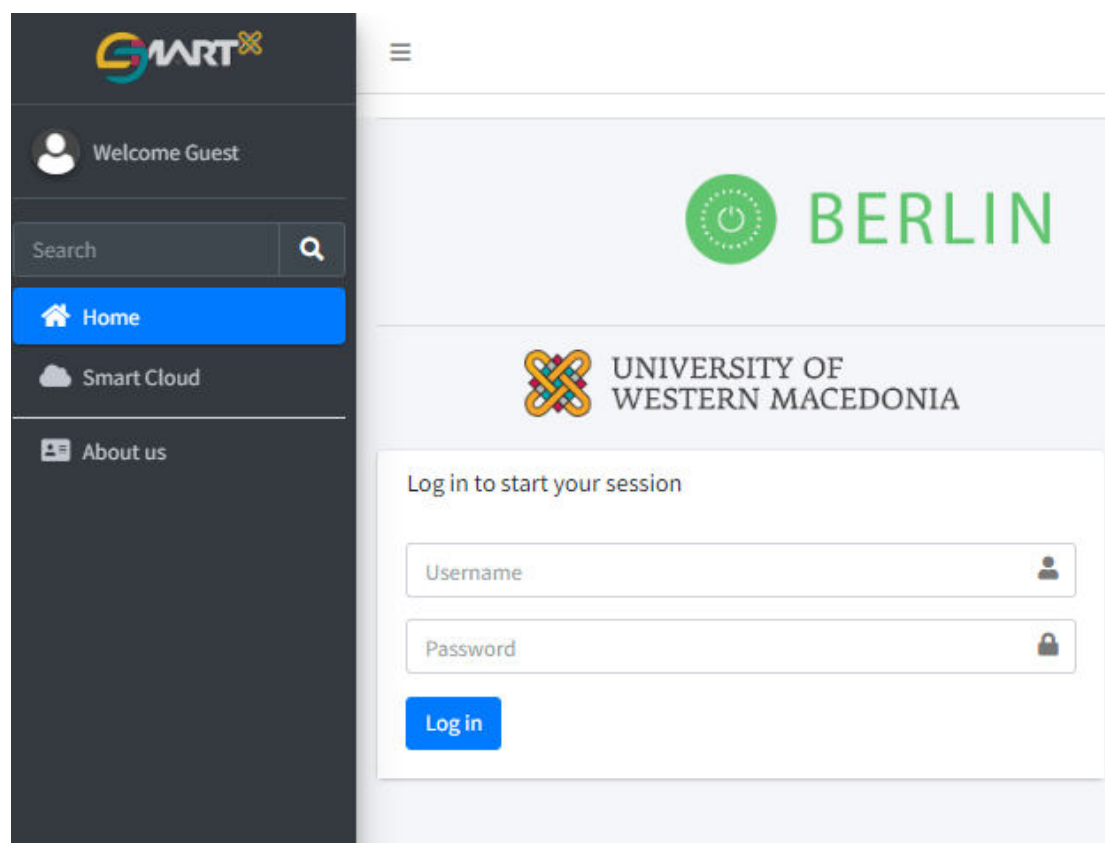
Πίνακας 1 Λογισμικά πακέτα πλατφόρμας SMART.....	10
Πίνακας 2 Λίστα συσκευών Device Status.....	26
Πίνακας 3 Τύποι διαγραμμάτων VEGA.....	37



# ΕΙΣΑΓΩΓΗ

## Εισαγωγή στην πλατφόρμα SMART

Η πλατφόρμα SMART (Εικόνα 1) σχεδιάστηκε για να διευκολύνει τις ανάγκες του ζωντανού εργαστηρίου του Πανεπιστημίου Δυτικής Μακεδονίας το οποίο βρίσκεται στις φοιτητικές εστίες στην περιοχή Κοίλα του Νομού Κοζάνης. Η πλατφόρμα περιλαμβάνει μια βάση δεδομένων όλων των μετρήσεων που καταγράφονται καθημερινά σε πραγματικό χρόνο από αισθητήρες και API που είναι τοποθετημένα μέσα στον χώρο των φοιτητικών εστιών. Μέσω της σελίδας παρέχονται οι δυνατότητες της παρουσίασης και λήψης δεδομένων από τις καταγεραμμένες μετρήσεις. Υπάρχει επίσης ένα σύστημα εποπτείας SCADA και κάποια online εργαλεία όπως προσομοίωση κυκλωμάτων, προσομοίωση έξυπνου δικτύου, βελτιστοποίηση PV και BESS. Στα εργαλεία της πλατφόρμας SMART περιλαμβάνεται επίσης μια πλατφόρμα γραφημάτων που βασίζεται στη γλώσσα απεικόνισης της VEGA τα οποία μπορούν να παρουσιάσουν μια πιο αντιληπτή εικόνα των καταγεραμμένων μετρήσεων. Κάποια άλλα στοιχεία της σελίδας είναι παρουσιάσεις των εργαστηρίων Power-hardware-in-the-loop & Superconductivity και μια βάση δεδομένων προβλέψεων.



Εικόνα 1 Αρχική οθόνη σύνδεσης της πλατφόρμας SMART (Smart.ece.uowm.gr)

# ΚΕΦΑΛΑΙΟ 1 ΤΑ ΛΟΓΙΣΜΙΚΑ ΠΑΚΕΤΑ ΤΗΣ ΠΛΑΤΦΟΡΜΑΣ SMART

Για την κατασκευή της πλατφόρμας SMART έχουν χρησιμοποιηθεί τα παρακάτω ελεύθερα πακέτα λογισμικού:

Πίνακας 1 Λογισμικά πακέτα πλατφόρμας SMART

<b>Debian linux</b>	Λειτουργικό σύστημα
<b>Mongodb</b>	Database των μετρήσεων
<b>mySQL</b>	Database των χρηστών
<b>HTML / javascript</b>	Παρουσίαση της σελίδας
<b>Apache</b>	Ανάρτηση της σελίδας
<b>PHP</b>	Διάφορες λειτουργίες
<b>Python</b>	Λειτουργίες αυτοματοποίησης

Στις επόμενες υποπαραγράφους, παρουσιάζονται πληροφορίες για κάθε ένα από τα λογισμικά πακέτα που χρησιμοποιήθηκαν ξεχωριστά.

## 1.1.1 Λειτουργικό σύστημα Debian

Το Debian δημιουργήθηκε ως ένας εθελοντικός οργανισμός αφιερωμένος στην ανάπτυξη ελεύθερου λογισμικού και στην προώθηση των ιδεών της κοινότητας του Ελεύθερου Λογισμικού. Το σχέδιο Debian ξεκίνησε το 1993 από τον Ian Murdock ο οποίος εξέδωσε μια ανοιχτή πρόσκληση σε δημιουργούς λογισμικού για να συμβάλλουν σε μια πλήρη και συνεκτική διανομή λογισμικού βασισμένη στον σχετικά νέο τότε πυρήνα Linux. Έτσι δημιουργήθηκε ένα νέο ανοιχτό λειτουργικό σύστημα βασισμένο στο UNIX το οποίο είναι δωρεάν. Το Debian υποστηρίζεται από δωρεές που γίνονται μέσω οργανισμών που προωθούν το ελεύθερο λογισμικό και δε χρηματοδοτείται από κάποια εταιρία αλλά από το Debian Project και τον οργανισμό Software in the Public Interest. Είναι γνωστό για την αφοσίωση του στη φιλοσοφία του Unix και του ελεύθερου λογισμικού, είναι δομημένο για ένα μεγάλο φάσμα συσκευών που περιλαμβάνει φορητούς υπολογιστές, υπολογιστές γραφείου και μικροϋπολογιστές. Όταν δημιουργήθηκε εισήγαγε πολλές δυνατότητες στο Linux που στις σημερινές μέρες είναι πλέον συνηθισμένες. Για παράδειγμα, ήταν η πρώτη διανομή Linux που περιελάμβανε σύστημα διαχείρισης πακέτων για εύκολη εγκατάσταση και αφαίρεση λογισμικού αλλά επίσης και η πρώτη διανομή Linux που μπορούσε να αναβαθμιστεί χωρίς να απαιτείται επανεγκατάσταση. Ένα από τα κύρια

χαρακτηριστικά του Debian είναι ο μεγάλος αριθμός πακέτων λογισμικού αλλά και η πολύ υψηλή ασφάλεια του.

Το Debian είναι επίσης πολύ γνωστό για το σύστημα διαχείρισης πακέτων του και για το APT (Advanced Packaging Tool) δηλαδή ένα προηγμένο εργαλείο πακέτων που διαθέτει συγκεκριμένα χαρακτηριστικά για τις αυστηρές πολιτικές που υιοθετεί ως προς την ποιότητα των πακέτων και των εκδόσεων του αλλά και την ανοιχτή διαδικασία ανάπτυξης ελέγχου που υιοθετεί. Αυτές οι πρακτικές κάνουν πιο εύκολες τις αναβαθμίσεις και την εγκατάσταση ή αφαίρεση πακέτων.

Το χαρακτηριστικό που κάνει το Debian να ξεχωρίζει περισσότερο από άλλες διανομές Linux είναι το σύστημα διαχείρισης πακέτων τα οποία μπορούν επίσης να προστατεύονται μέσω συνεχούς ενημέρωσης. Αυτά τα εργαλεία δίνουν στον διαχειριστή ενός συστήματος Debian πλήρη έλεγχο των πακέτων που είναι εγκατεστημένα σε αυτό το σύστημα, συμπεριλαμβανομένης της δυνατότητας εγκατάστασης ενός μόνο πακέτου ή αυτόματης ενημέρωσης ολόκληρου του λειτουργικού συστήματος. [5],[13]

### **1.1.2 Βάση δεδομένων MongoDB**

Η MongoDB εμφανίστηκε γύρω στα μέσα της δεκαετίας του 2000 και είναι μια βάση δεδομένων NoSQL προσανατολισμένη στα έγγραφα που χρησιμοποιείται για αποθήκευση δεδομένων μεγάλου όγκου. Αντί δηλαδή να χρησιμοποιεί πίνακες και σειρές όπως στις παραδοσιακές σχεσιακές βάσεις δεδομένων, η MongoDB χρησιμοποιεί συλλογές εγγράφων τύπου JSON.

Η MongoDB διαφέρει ως προς το ότι τείνει να είναι λιγότερο αυστηρή για το πώς αυτά τα δεδομένα είναι δομημένα καθώς οι πληροφορίες ομαδοποιούνται αντί να αναλύονται σε σχετικά έγγραφα και πίνακες. Τα έγγραφα αποτελούνται από ζεύγη κλειδιών/τιμών που αποτελούν τη βασική μονάδα δεδομένων της. Είναι επίσης μια βάση δεδομένων χωρίς σχήμα, οπότε δε χρειάζεται να καθορίσουμε τον αριθμό ή τον τύπο στηλών πριν από την εισαγωγή των δεδομένων. Οι συλλογές περιέχουν σύνολα εγγράφων και συναρτήσεις που είναι ισοδύναμες με τους πίνακες σχεσιακών βάσεων δεδομένων. Τα έγγραφα δεν χρειάζεται να είναι δομημένα με συγκεκριμένο τρόπο σύμφωνα με τη συλλογή ή μια ομάδα σχετικών αντικειμένων που είναι οργανωμένα, καθιστώντας έτσι πολύ εύκολο για έναν προγραμματιστή να ξεκινήσει και να τρέξει τη MongoDB. Χρησιμοποιεί το BSON για την αποθήκευση πληροφοριών το οποίο είναι ένας συνδυασμός "δυαδικού" και "JSON" ή JavaScript Object Notation και

μπορεί να θεωρηθεί ως δυαδική ή αριθμητική αναπαράσταση των εγγράφων JSON. Τα αρχεία JSON χρησιμοποιούνται επίσης για τη μετάδοση δεδομένων μεταξύ ενός διακομιστή και μιας εφαρμογής, καθώς και ενός γενικού μέσου αποθήκευσης δεδομένων.

Άλλα χαρακτηριστικά ασφαλείας περιλαμβάνουν τη λίστα επιτρεπόμενων IP ή το VPC Peering, τον έλεγχο ταυτότητας πάντα σε λειτουργία, την κρυπτογράφηση σε κατάσταση ηρεμίας, την κρυπτογράφηση κατά τη μεταφορά, την εξελιγμένη διαχείριση πρόσβασης βάσει ρόλων και πολλά άλλα. Η MongoDB είναι επίσης φιλική προς το χρήστη καθώς δεν απαιτεί ενοποιημένη δομή δεδομένων για όλα τα αντικείμενα της και δεν πρέπει να είναι η ίδια σε όλο το σύστημα, επομένως δεν είναι απολύτως απαραίτητος ένας και μοναδικός τρόπος σκέψης αλλά δίνεται η ελευθερία πολλών πρωτοβουλιών και τροποποιήσεων στην δομή της. [6],[15]

### **1.1.3 Βάση δεδομένων MySQL**

Η MySQL δημιουργήθηκε το 1995 και αποτελεί ένα ανοιχτού κώδικα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων και είναι η πιο δημοφιλής βάση δεδομένων ελεύθερου κώδικα μέχρι και σήμερα κερδίζοντας πολλά βραβεία. Είναι ένα πολύ γρήγορο και αξιόπιστο σχεσιακό σύστημα διαχείρισης βάσεως δεδομένων που χρησιμοποιείται κυρίως για διαδικτυακές εφαρμογές. Χρησιμοποιείται σε πολλούς διαφορετικούς κλάδους και μπορεί να προσαρμοστεί για να ταιριάζει σε οποιαδήποτε ανάγκη. Είναι ένα λογισμικό ανοιχτού κώδικα και δεν υπάρχουν περιορισμοί στον αριθμό των διακομιστών, των χρηστών ή των βάσεων δεδομένων που μπορούν να δημιουργηθούν με αυτή. Επίσης, αποτελεί μια βάση δεδομένων που δίνει τη δυνατότητα να πραγματοποιηθεί αποθήκευση, αναζήτηση, ταξινόμηση και ανάκτηση δεδομένων με αποτελεσματικό τρόπο.

Η MySQL είναι μια σχεσιακή βάση δεδομένων κάτι που σημαίνει ότι τα δεδομένα που βρίσκονται μέσα στη δομή της είναι ικανά να αναγνωρίζουν σχέσεις μεταξύ αποθηκευμένων στοιχείων πληροφοριών. Κάθε βάση δεδομένων περιέχει πίνακες και κάθε πίνακας ξεχωριστά περιέχει μία ή περισσότερες κατηγορίες δεδομένων που είναι αποθηκευμένες σε στήλες. Κάθε σειρά περιέχει ένα μοναδικό κομμάτι δεδομένων για τις κατηγορίες που ορίζονται στις στήλες. Μέσα στις δυνατότητες της συμπεριλαμβάνεται επίσης ο έλεγχος της πρόσβασης στα δεδομένα ώστε να εξασφαλιστεί ότι πολλαπλοί χρήστες μπορούν να τα χρησιμοποιούν ταυτόχρονα, να παρέχει γρήγορη πρόσβαση σε αυτά και να εγγυάται ότι μόνο οι εξουσιοδοτημένοι

χρήστες θα μπορούν να τα προσπελάσουν. Χρησιμοποιεί τη Structured Query Language (SQL), την πιο δημοφιλή γλώσσα ερωτημάτων σε βάσεις δεδομένων και διατίθεται δωρεάν με άδεια ελεύθερου κώδικα ή αντί χαμηλού κόστους με άδεια για επαγγελματική χρήση.

Η MySQL είναι επομένως ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων για την κατασκευή διακομιστών πολλαπλών χρηστών και απεριόριστων λειτουργιών. [1],[3]

#### **1.1.4 Γλώσσα προγραμματισμού HTML**

Η γλώσσα HTML προέρχεται από τα αρχικά των λέξεων HyperText Markup Language και ο κώδικας αυτός γράφεται μέσα σε αρχεία html ή htm τα οποία ανοίγουν με κάποιον περιηγητή ιστού (browser) και εμφανίζουν την ιστοσελίδα που υπάρχει με τον συγκεκριμένο κώδικα. Το 1990 ο Tim Berners-Lee δημιούργησε ένα νέο πρωτόκολλο με το οποίο θα μπορούσαν να μεταφέρονται κάθε είδους αρχείων και αντικειμένων μέσα από το Internet. Το πρωτόκολλο αυτό ονομάστηκε HTTP (HyperText Transfer Protocol) και σηματοδότησε την αρχή του “www” όπως το ξέρουμε σήμερα. Οι σελίδες που ήταν η βάση του “www” ήταν γραμμένες στην πρώτη έκδοση της γλώσσας HTML. Η γλώσσα HTML περιγράφει τη δομή μιας σελίδας και είναι η βασική γλώσσα δόμησης σελίδων του διαδικτύου η οποία καθορίζει τη διάταξη και τη δομή μιας ιστοσελίδας χρησιμοποιώντας μορφοποίηση ετικετών (tags).

Κάθε στοιχείο ιστοσελίδας περιλαμβάνει μια συγκεκριμένη ετικέτα (tag) που λέει στο πρόγραμμα περιήγησης πως να μορφοποιήσει ή να δομήσει αυτό το τμήμα της σελίδας. Είναι μια γλώσσα σήμανσης που χρησιμοποιείται από τους προγραμματιστές ιστοσελίδων για να δημιουργήσουν σχεδόν όλες τις ιστοσελίδες που υπάρχουν στο διαδίκτυο. Λέγεται αλλιώς και γλώσσα σήμανσης υπερκειμένου καθώς επιτρέπει στους χρήστες της να δημιουργούν και να δομούν ενότητες, παραγράφους και συνδέσμους χρησιμοποιώντας στοιχεία, ετικέτες και χαρακτηριστικά. Ωστόσο, αξίζει να σημειωθεί πως δε θεωρείται ακριβώς γλώσσα προγραμματισμού καθώς δεν μπορεί να δημιουργήσει δυναμική λειτουργικότητα. Έχει πολλές εφαρμογές χρήσης με βασικότερη την ανάπτυξη ιστοσελίδων. Οι προγραμματιστές χρησιμοποιούν κώδικα HTML για να σχεδιάσουν πως ένα πρόγραμμα περιήγησης εμφανίζει τα στοιχεία μιας ιστοσελίδας, όπως κείμενο, υπερσυνδέσμους και αρχεία πολυμέσων. Επίσης, χρησιμοποιείται για πλοήγηση στο διαδίκτυο καθώς οι χρήστες μπορούν εύκολα να

πλοηγηθούν και να εισαγάγουν συνδέσμους μεταξύ σχετικών ιστοσελίδων, καθώς η HTML χρησιμοποιείται σε μεγάλο βαθμό για την ενσωμάτωση υπερσυνδέσμων. Μπορούμε πολύ απλά να δούμε τον κώδικα HTML για οποιαδήποτε ιστοσελίδα σε ένα πρόγραμμα περιήγησης απλά κάνοντας δεξί κλικ οπουδήποτε στη σελίδα και επιλέγοντας “Προβολή κώδικα σελίδας” (View page source). [8],[14]

### **1.1.5 Γλώσσα προγραμματισμού Javascript**

Η γλώσσα προγραμματισμού JavaScript δημιουργήθηκε από τον Brendan Eich της εταιρείας Netscape. Η αρχική της ονομασία ήταν Mocha και στην συνέχεια μετονομάστηκε σε LiveScript, και τελικά σε JavaScript, κυρίως επειδή η ανάπτυξη της επηρεάστηκε περισσότερο από τη γλώσσα προγραμματισμού Java. Η JavaScript είναι μια γλώσσα προγραμματισμού υπολογιστών για σεναριοποίηση των διαδραστικών αποτελεσμάτων εντός των περιηγητών ιστού και υποστηρίζεται από όλους τους δημοφιλείς περιηγητές ιστού. Η αρχική έκδοση της Javascript βασίστηκε σε σύνταξη της γλώσσας προγραμματισμού C αλλά πλέον έχει εξελιχθεί ενσωματώνοντας κάποια χαρακτηριστικά από νεότερες γλώσσες.

Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), που ήταν ο περιηγητής ιστού (browser) του χρήστη, και χαρακτηρίστηκε σαν client-side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (server-side γλώσσες προγραμματισμού). Αποτελεί μια από τις πιο ευέλικτες και αποτελεσματικές γλώσσες που μπορεί να χρησιμοποιηθούν από προγραμματιστές.

Κάποια από τα βασικά πλεονεκτήματα της είναι η απλή της χρήση καθώς η σύνταξη αυτής της γλώσσας είναι παρόμοια με τα απλά Αγγλικά και δεν απαιτεί κανέναν ειδικό μεταγλωττιστή ή συντάκτη καθώς το μόνο που χρειάζεται είναι ένας προγραμματιστής, ένα πρόγραμμα επεξεργασίας κειμένου και έναν περιηγητή για να “τρέξει” τον κώδικα JavaScript. Οι επισκέπτες δε χρειάζεται να συμπληρώσουν μια ολόκληρη φόρμα και να την υποβάλλουν για να μάθουν πως υπάρχει κάποιο τυπογραφικό λάθος στο πρώτο πεδίο και ότι θα πρέπει να συμπληρώσουν ολόκληρη τη φόρμα ξανά καθώς με τη JavaScript κάθε πεδίο μπορεί να επαληθεύεται καθώς

συμπληρώνεται από τους χρήστες, γεγονός που παρέχει άμεση ανατροφοδότηση όταν αυτοί κάνουν κάποιο λάθος.

Η Javascript χρησιμοποιείται κυρίως για να βελτιώσει τις ιστοσελίδες και για να προσφέρει μια πιο φιλική προς τον χρήστη εμπειρία. Αυτές περιλαμβάνουν δυναμική ενημέρωση ιστοσελίδων, βελτιώσεις διεπαφής χρήστη, όπως μενού και πλαίσια διαλόγου, κινούμενα σχέδια, γραφικά 2D και 3D, διαδραστικούς χάρτες, συσκευές αναπαραγωγής βίντεο και άλλα. [12],[19]

### **1.1.6 Διακομιστής Apache**

Ο Apache HTTP Server είναι ένας διακομιστής ιστού (web) ανοιχτού κώδικα που δημιουργήθηκε από τον Αμερικανό προγραμματιστή λογισμικού Ρόμπερτ ΜακΚουλ. Ο διακομιστής Apache HTTP ("httpd") κυκλοφόρησε το 1995 και είναι ο πιο δημοφιλής διακομιστής ιστού στο Διαδίκτυο από τον Απρίλιο του 1996. Το λογισμικό του διαχειρίζεται το εμπορικό σήμα και τις υπηρεσίες του. Ως διακομιστής ιστού, ο Apache είναι υπεύθυνος για την αποδοχή αιτημάτων καταλόγου (HTTP) από χρήστες του Διαδικτύου και την αποστολή των πληροφοριών που επιθυμούν με τη μορφή αρχείων και ιστοσελίδων.

Υπάρχουν πολλά πλεονεκτήματα για τον διακομιστή HTTP του Apache. Το πιο αξιοσημείωτο από αυτά αποτελεί ότι είναι εντελώς δωρεάν τόσο για ιδιωτικές όσο και για εμπορικές χρήσεις. Μεγάλο μέρος του λογισμικού και του κώδικα του ιστού έχει σχεδιαστεί για να λειτουργεί μαζί με τις δυνατότητες του Apache. Παρόλο που είναι ένα δωρεάν και συνεχώς ενημερωμένο προϊόν, δε μειώνει τις λειτουργίες του. Στην πραγματικότητα, είναι ένας από τους πιο γνωστούς διαδικτυακούς διακομιστές HTTP με πλήρεις δυνατότητες, ο οποίος είναι ένας άλλος λόγος που είναι τόσο δημοφιλής. Οι προγραμματιστές που εργάζονται σε εφαρμογές συνήθως χρησιμοποιούν μια αρχική έκδοση του Apache για προ επισκόπηση και δοκιμή κώδικα.

Ο Apache διαθέτει επίσης μια ασφαλή δυνατότητα κοινής χρήσης αρχείων, η οποία επιτρέπει στους χρήστες να τοποθετούν αρχεία στον ριζικό κατάλογο του λογισμικού τους και να τα μοιράζονται με άλλους χρήστες. Ο server αναπτύσσεται και συντηρείται από μια ανοιχτή κοινότητα προγραμματιστών υπό την ASF και συνήθως τρέχει σε Linux. Ο Apache τρέχει περίπου στο 46% του συνόλου των ιστοσελίδων σε ολόκληρο τον κόσμο. Το Apache HTTP Server Project είναι μια προσπάθεια ανάπτυξης και διατήρησης ενός διακομιστή HTTP ανοιχτού κώδικα για σύγχρονα

λειτουργικά συστήματα, συμπεριλαμβανομένων των UNIX και Windows. Ο στόχος αυτού του έργου είναι να παρέχει έναν ασφαλή, αποτελεσματικό και επεκτάσιμο διακομιστή που παρέχει υπηρεσίες HTTP σε συγχρονισμό με τα τρέχοντα πρότυπα HTTP. [9],[10]

### **1.1.7 Γλώσσα προγραμματισμού PHP**

Η PHP είναι μια γλώσσα σεναρίων για τον διακομιστή, η οποία έχει σχεδιαστεί συγκεκριμένα για το διαδίκτυο. Δημιουργήθηκε το 1994 από τον Rasmus Lerdorf και υιοθετήθηκε από πολλούς ανθρώπους περνώντας από πολλά στάδια μέχρι να φτάσει στο τελικό της σημερινό επίπεδο. Αρχικά σήμαινε Personal Home Page, αλλά στη συνέχεια άλλαξε σημασία ώστε να συμμορφωθεί με τη σύμβαση ονομάτων GNU (Gnu's Not Unix) και πλέον σημαίνει PHP Hypertext Preprocessor.

Είναι μία πολύ διαδεδομένη γλώσσα προγραμματισμού γενικής χρήσης, που είναι κατάλληλη για προγραμματισμό διαδικτυακών εφαρμογών και μπορεί να εισαχθεί σε HTML. Η PHP είναι μια γλώσσα προγραμματισμού δημιουργημένη ειδικά για την κατασκευή δυναμικών ιστοσελίδων, σελίδων δηλαδή που μπορούν να τροποποιούνται από το διαχειριστή τους online ή να διαφοροποιούνται, ανάλογα με τα χαρακτηριστικά του χρήστη που τις προβάλλει. Ο τρόπος χρήσης της δεν είναι για την αισθητική διαμόρφωση μιας σελίδας, αλλά κυρίως για τον χειρισμό των λειτουργιών και εργασιών που θα διεκπεραιώνει. Συνεπώς, ο κώδικας που γράφεται για μια ιστοσελίδα σε γλώσσα PHP δε γίνεται άμεσα αντιληπτός αλλά μετά από την επέμβαση του χρήστη στην ιστοσελίδα. Μέσα σε μια σελίδα HTML μπορεί να γίνει η ενσωμάτωση του κώδικα PHP που θα εκτελείται κάθε φορά που κάποιος χρήστης επισκέπτεται τη σελίδα.

Ο κώδικας της γλώσσας μπορεί να ενσωματωθεί μέσα στον κώδικα μιας HTML σελίδας ή να αποθηκεύεται στον διακομιστή ως ξεχωριστό αρχείο κατάληξης php. Όταν η σελίδα ζητηθεί από κάποιο πρόγραμμα περιήγησης ο διακομιστής επεξεργάζεται και εκτελεί τον κώδικα php. Ο διακομιστής συνδυάζει τα δεδομένα της HTML σελίδας μαζί με τα δυναμικά δημιουργημένα αποτελέσματα από την εκτέλεση της PHP και τα παραδίδει στην εφαρμογή του χρήστη που τα ζήτησε σε μορφή HTML κώδικα. Επειδή σχεδιάστηκε για χρήση στο διαδίκτυο, ενσωματώνει πολλές συναρτήσεις για την εκτέλεση διάφορων χρήσιμων εργασιών για το διαδίκτυο μαζί με έμφυτες συνδέσεις που διατίθενται σε πολλά συστήματα βάσεων δεδομένων. Σύμφωνα με έρευνα που πραγματοποιήθηκε το 2015 από την google η PHP



χρησιμοποιούταν τη συγκεκριμένη χρονική περίοδο για τη λειτουργία περισσότερων από το 75% των ιστοσελίδων ολόκληρου του κόσμου.

Συνοψίζοντας η γλώσσα προγραμματισμού PHP αποτελεί ένα έργο ανοιχτού κώδικα που σημαίνει πως έχουμε πρόσβαση στον πηγαίο κώδικα της αλλά και την απεριόριστη ελευθερία τόσο για τη χρήση όσο και την αλλαγή της καθώς και την αναδιανομή της. [3],[11]

### **1.1.8 Γλώσσα προγραμματισμού Python**

Η Python είναι μια γλώσσα προγραμματισμού που αναπτύχθηκε από τον Guido van Rossum του Εθνικού Ινστιτούτου Ερευνών για Μαθηματικά και Πληροφορική στο Άμστερνταμ και κυκλοφόρησε επίσημα το 1991. Έγινε γρήγορα μία από τις δημοφιλέστερες γλώσσες προγραμματισμού παγκοσμίως ειδικά στην εκπαιδευτική και την επιστημονική πληροφορική προσπερνώντας πρόσφατα τη γλώσσα προγραμματισμού R ως η πιο δημοφιλής γλώσσα προγραμματισμού στην επιστήμη δεδομένων.

Η Python είναι μια γλώσσα προγραμματισμού που διαβάζεται πολύ ευκολότερα από πολλές άλλες δημοφιλείς γλώσσες και χρησιμοποιείται ευρέως στον τομέα της εκπαίδευσης. Είναι μια ελεύθερη γλώσσα ανοιχτού κώδικα που είναι διαθέσιμη δωρεάν και βελτιώνει την παραγωγικότητα των προγραμματιστών με την βοήθεια εκτενών βιβλιοθηκών ανοιχτού κώδικα από τρίτους έτσι ώστε οι προγραμματιστές να μπορούν να γράφουν κώδικα ταχύτερα και να εκτελούν πολύπλοκες εργασίες με ελάχιστες εντολές.

Ένας από τους κύριους λόγους της δημοτικότητας της είναι ότι μπορεί να διαβαστεί και να γίνει εύκολα κατανοητή ακόμη και από ερασιτέχνες προγραμματιστές. Η Python, είναι εξαιρετικά ευέλικτη και χρησιμοποιείται για διάφορες απαιτητικές εργασίες. Χρησιμοποιείται ευρέως σε πάρα πολλούς τομείς, από την τεχνητή νοημοσύνη στον κβαντικό υπολογισμό και τη ρομποτική. Υποστηρίζει επίσης πολλά πρότυπα δημοφιλούς προγραμματισμού όπως διαδικαστικού, συναρτησιακού, αντικειμενοστραφούς, ανακλαστικού. Ένα ακόμα πλεονέκτημα της είναι πώς έχει γίνει δημοφιλής στον κλάδο της τεχνητής νοημοσύνης η οποία μετέρχεται εκρηκτικής αύξησης κυρίως λόγω της ειδικής σχέσης της με την επιστήμη δεδομένων.

Με το πέρασμα του χρόνου αποτελεί μια από τις δημοφιλέστερες γλώσσες προγραμματισμού καθώς έχει πάρα πολλά πλεονεκτήματα σε σχέση με αντίστοιχες γλώσσες προγραμματισμού όπως η C, η C++ και η Java. Η κοινότητα

προγραμματιστών rython είναι μία από τις πιο ενεργές κοινότητες κάτι που σημαίνει πως εάν κάποιος χρήστης της αντιμετωπίσει οποιοδήποτε πρόβλημα μπορεί να λάβει άμεση καθοδήγηση οποιαδήποτε στιγμή, πράγμα σπουδαίο αφού μειώνει τις καθυστερήσεις στην παράδοση οποιοδήποτε project. [2],[4]

## **ΚΕΦΑΛΑΙΟ 2 ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ ΒΑΣΗΣ**

### **ΔΕΔΟΜΕΝΩΝ Smart.ece.uowm.gr**

Η πλατφόρμα SMART είναι προσβάσιμη στον σύνδεσμο <https://smart.ece.uowm.gr>. Ο χρήστης αρχικά πρέπει να πραγματοποιήσει είσοδο με το όνομα και τον κωδικό του λογαριασμού του και στη συνέχεια του δίνεται η δυνατότητα να επιλέξει πιο από τα τμήματα της πλατφόρμας θέλει να χρησιμοποιήσει. Η πλατφόρμα χωρίζεται στα εξής τμήματα: Database”, “Living Lab SCADA”, “Online tools”, “Laboratory facilities”, “Documentation”, “About us” και στις επόμενες παραγράφους, παρουσιάζονται αναλυτικές πληροφορίες για κάθε ένα από τα τμήματα ξεχωριστά.

### **2.1 Παρουσίαση του τμήματος της Database**

#### **2.1.1 Τι είναι μια βάση δεδομένων (Database)**

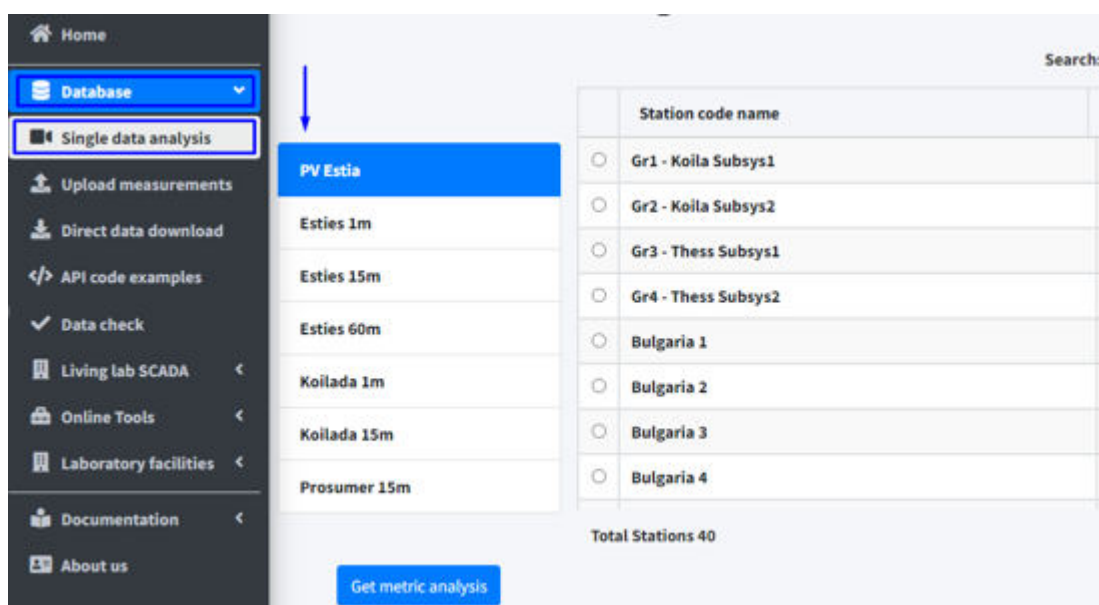
Μια βάση δεδομένων είναι ένας οργανωμένος τρόπος αποθήκευσης πληροφοριών που επιτρέπει σε πολλούς χρήστες ταυτόχρονα να έχουν γρήγορη και ασφαλή πρόσβαση στα δεδομένα που περιέχει. Είναι ουσιαστικά ένα εργαλείο για την συλλογή, την οργάνωση και την μελέτη δεδομένων. Μια database αποθηκεύει πληροφορίες για διάφορες μετρήσεις όπως ενεργειακές, προϊόντων, ατομικών προτιμήσεων, ή οτιδήποτε άλλο μπορεί να έχει χρηστική αξιοποίηση. Οι βάσεις δεδομένων μπορεί να είναι πολύ απλές στη μορφή τους όπως μια λίστα σε επεξεργαστή κειμένου ή ένα υπολογιστικό φύλλο αλλά και πιο σύνθετες και περιεκτικές με την χρήση γραφημάτων για καλύτερη αναπαράσταση των δεδομένων. Το λογισμικό μιας βάσης δεδομένων χρησιμοποιείται για τη δημιουργία, την επεξεργασία και την διατήρηση αρχείων και εγγραφών επιτρέποντας την ευκολότερη ενημέρωση και την αναφορά. Υπάρχουν πολλοί διαφορετικοί τύποι βάσεων δεδομένων. Η καταλληλότερη βάση δεδομένων για έναν συγκεκριμένο οργανισμό εξαρτάται από τον τρόπο με τον οποίο ο οργανισμός σκοπεύει να χρησιμοποιήσει τα δεδομένα.

Όσο το μέγεθος της βάσης δεδομένων μεγαλώνει εμφανίζονται περισσότερα προβλήματα όπως ασυνέπειες και υπάρχουν δυσκολίες στην διαχείριση. Τα δεδομένα που εμπεριέχει είναι πολλές φορές δύσκολο να κατανοηθούν και υπάρχουν περιορισμένοι τρόποι αναζήτησης ή ανάκτησης υποσυνόλων δεδομένων για επαλήθευση.

Μια ηλεκτρονική βάση δεδομένων είναι ουσιαστικά ένα δοχείο για πολλές χρήσιμες πληροφορίες οι οποίες αποτελούν τμήμα ανατροφοδότησης για άλλες διεργασίες ή μπορούν να παρουσιαστούν στον χρήστη με την μορφή πινάκων και γραφημάτων για την καλύτερη και ευκολότερη κατανόηση τους.

### 2.1.2 Single data analysis

Στο τμήμα της πλατφόρμας Single data analysis (Εικόνα 2) ο χρήστης αρχικά επιλέγει από ποια κατηγορία θέλει να λάβει μετρήσεις (Εικόνα 3), στην συνέχεια από ποιόν μετρητικό σταθμό θέλει να είναι αυτές (Εικόνα 4) και έπειτα για ποιο ηλεκτρικό στοιχείο τους συγκεκριμένα.



Εικόνα 2 Single data analysis (Smart.ece.uowm.gr)

PV Estia
Esties 1m
Esties 15m
Esties 60m
Koilada 1m
Koilada 15m
Prosumer 15m

Εικόνα 3 Single data analysis τμήμα επιλογής κατηγορίας

	Station code name	inMap
<input type="radio"/>	Gr1 - Koila Subsys1	
<input type="radio"/>	Gr2 - Koila Subsys2	
<input type="radio"/>	Gr3 - Thess Subsys1	
<input type="radio"/>	Gr4 - Thess Subsys2	
<input type="radio"/>	Bulgaria 1	
<input type="radio"/>	Bulgaria 2	
<input type="radio"/>	Bulgaria 3	
<input type="radio"/>	Bulgaria 4	

Total Stations 40

Εικόνα 4 Single data analysis τμήμα επιλογής μετρητικού σταθμού

	Metric code name	inUnit
<input type="radio"/>	Aggregate data	
<input type="radio"/>	Ambient Humidity	%
<input type="radio"/>	Ambient Temperature	oC
<input type="radio"/>	Battery Current	A
<input type="radio"/>	Battery SoC	%
<input type="radio"/>	Battery Voltage	V
<input type="radio"/>	Consumption Power	W
<input type="radio"/>	Direct PV Consumption	W


Total metrics 21

Εικόνα 5 Single data analysis τμήμα επιλογής μετρητικού στοιχείου


Ο χρήστης μπορεί να επιλέξει το χρονικό διάστημα που μεσολαβεί ανάμεσα στις μετρήσεις με τις επιλογές του να είναι ανά ένα λεπτό, ανά δεκαπέντε λεπτά και κάθε μια ώρα.

Αφού έχει γίνει η επιλογή κατηγορίας, σταθμού και στοιχείου ο χρήστης πρέπει να επιλέξει ανάμεσα σε ποιο χρονικό διάστημα/ημερομηνίες θέλει να λάβει τις πληροφορίες αυτές.

Select from date

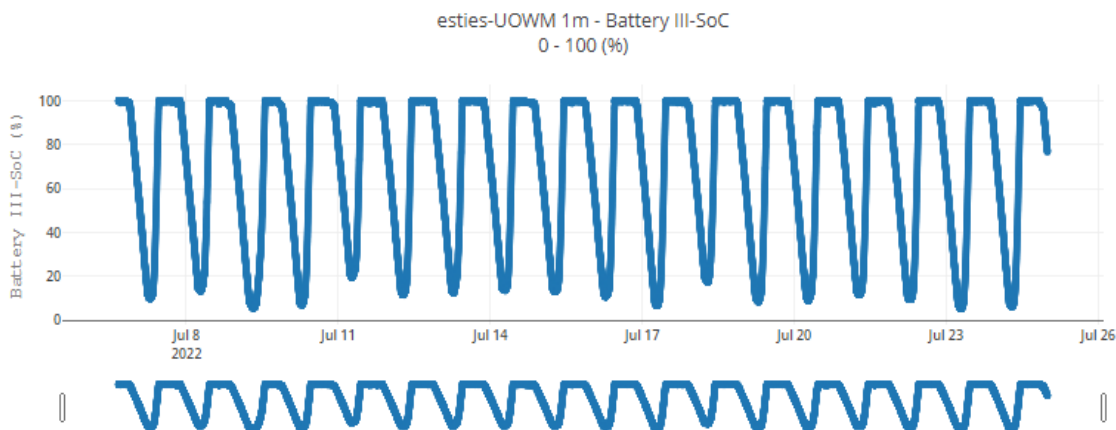
01/07/2022 

Select to date

24/07/2022 

Μόλις ο χρήστης επιλέξει το χρονικό διάστημα και το στοιχείο που επιθυμεί πατώντας το Get metric analysis εμφανίζεται ένα διάγραμμα με τις μετρήσεις που έχουν καταγραφεί στον συγκεκριμένο χρόνο (Εικόνα 3).

[Get metric analysis](#)

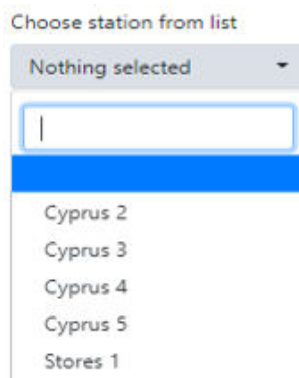


Εικόνα 6 Single data analysis αποτέλεσμα επιλογής Get metric analysis

### 2.1.3 Upload measurements

Σε αυτό το τμήμα της πλατφόρμας οι χρήστες ανάλογα με τα δικαιώματα που έχει ο λογαριασμός τους μπορούν να ανεβάζουν μετρήσεις στους ανάλογους μετρητικούς σταθμούς.

Ο χρήστης για να ανεβάσει χειροκίνητα δεδομένα στην πλατφόρμα Smart κάνει κλικ στο Upload measurements από το μενού “Database” και στην συνέχεια πραγματοποιεί την επιλογή σταθμού :



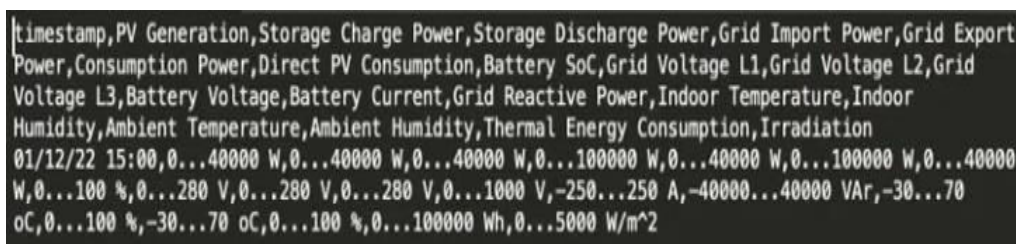
Εικόνα 7 Upload measurements τμήμα επιλογής σταθμού

Αφού επιλεγθεί ένας σταθμός στο πάνω δεξιά μέρος της σελίδας υπάρχουν πληροφορίες για τον επιλεγμένο σταθμό, οι οποίες είναι ο κωδικός ID του σταθμού και το χρονικό βήμα των μετρήσεων το οποίο στο παρακάτω παράδειγμα (Εικόνα 8) είναι 15 δεκαπέντε λεπτά.

Στη συνέχεια ο χρήστης επιλέγει το αρχείο με τις μετρήσεις που θέλει να ανεβάσει και πατώντας την επιλογή Send file ξεκινάει η διαδικασία ανεβάσματος (uploading) των δεδομένων. Κάτω από την επιλογή Send file μπορούμε να δούμε το μέγιστο

επιτρεπτό μέγεθος αρχείου και τις μορφές αρχείων που είναι αποδεκτές από το σύστημα.

Ακριβώς δίπλα υπάρχει η επιλογή download template file το οποίο είναι ένα αρχείο csv (Εικόνα 8) το οποίο ο χρήστης μπορεί να κατεβάσει και στο πάνω μέρος του περιέχει τις στήλες που δέχεται ο συγκεκριμένος σταθμός και ακριβώς από κάτω το εύρος τιμών που μπορούν να έχουν οι μετρήσεις τη συγκεκριμένη χρονική στιγμή για το κάθε μετρητικό στοιχείο ξεχωριστά. Στην περίπτωση που ο χρήστης δώσει μεγαλύτερες τιμές από τις επιτρεπόμενες θα εμφανιστεί σφάλμα και δε θα προχωρήσει το ανέβασμα του αρχείου στην πλατφόρμα.



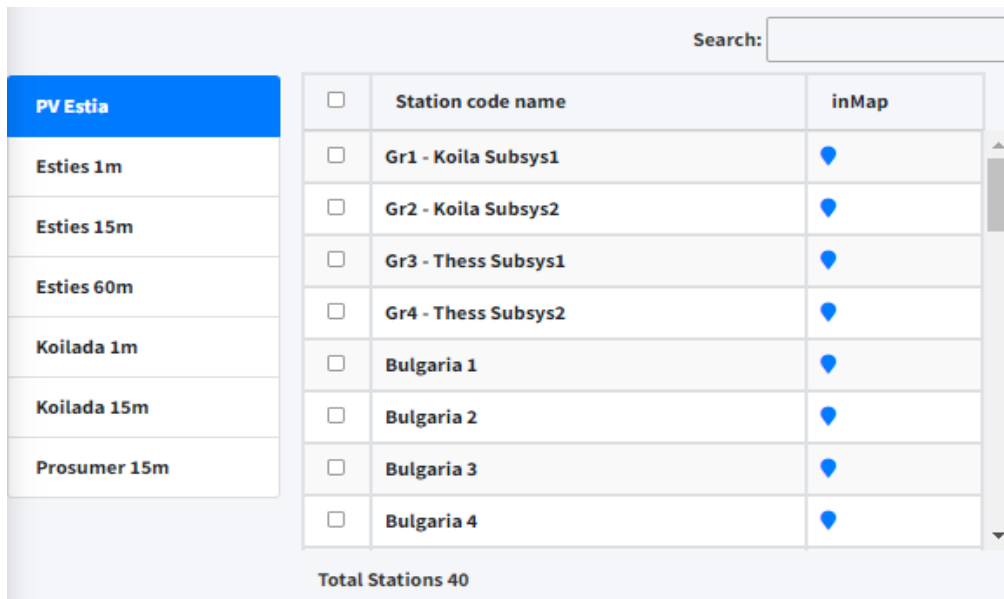
```
timestamp,PV Generation,Storage Charge Power,Storage Discharge Power,Grid Import Power,Grid Export Power,Consumption Power,Direct PV Consumption,Battery SoC,Grid Voltage L1,Grid Voltage L2,Grid Voltage L3,Battery Voltage,Battery Current,Grid Reactive Power,Indoor Temperature,Indoor Humidity,Ambient Temperature,Ambient Humidity,Thermal Energy Consumption,Irradiation  
01/12/22 15:00,0...40000 W,0...40000 W,0...40000 W,0...100000 W,0...40000 W,0...100000 W,0...40000 W,0...100 %,0...280 V,0...280 V,0...280 V,0...1000 V,-250...250 A,-40000...40000 VAR,-30...70 oC,0...100 %, -30...70 oC,0...100 %,0...100000 Wh,0...5000 W/m^2
```

Εικόνα 8 Αρχείο επιλογής Download template file

#### 2.1.4 Direct data download

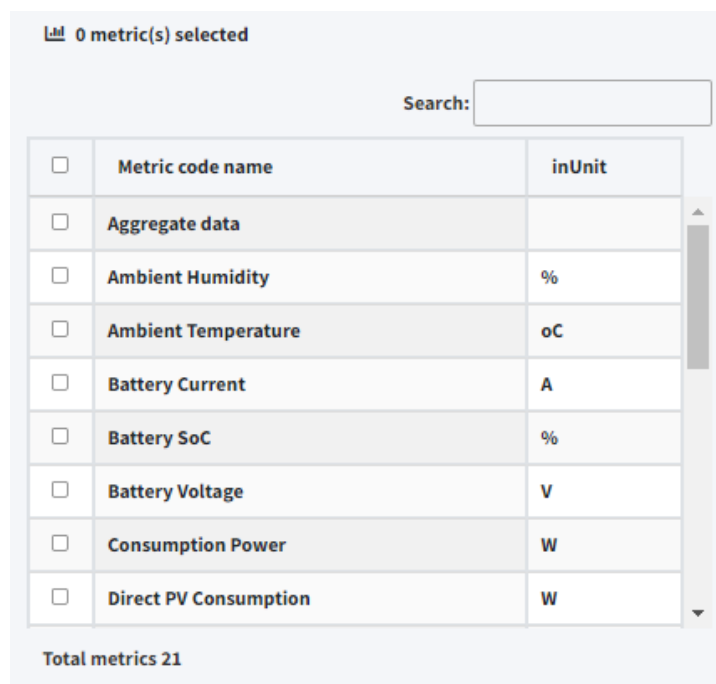
Το τμήμα “Direct data download” της πλατφόρμας SMART παρουσιάζει πολλές ομοιότητες με το τμήμα “Single data analysis” το οποίο αναλύθηκε σε προηγούμενο κεφάλαιο, αλλά η βασική διαφορά είναι πως στο τμήμα Direct data download μας δίνεται η επιλογή να επιλέξουμε πολλαπλά ή όλα τα μετρικά στοιχεία ενός σταθμού ταυτόχρονα και να λάβουμε τις αντίστοιχες μετρήσεις σε ξεχωριστά αρχεία csv για το καθένα.

Όπως και στο Single data analysis έτσι και εδώ η διαδικασία που πρέπει να ακολουθήσει ο χρήστης είναι η ίδια. Αρχικά στο τμήμα επιλογής σταθμού (Εικόνα 9) ο χρήστης μπορεί να επιλέξει την κατηγορία σταθμού που επιθυμεί και έπειτα από ποιόν συγκεκριμένο μετρητικό σταθμό θέλει να αντλήσει τις πληροφορίες (Εικόνα 9).



Εικόνα 9 Direct data download (Smart.ece.uowm.gr) τμήμα επιλογής σταθμού

Στην συνέχεια ο χρήστης επιλέγει από ποια μετρικά στοιχεία θέλει να λάβει πληροφορίες (Εικόνα 10) και τέλος το χρονικό διάστημα που μεσολαβεί ανάμεσα στις καταγεγραμμένες οι μετρήσεις.



Εικόνα 10 Direct data download τμήμα επιλογής μετρητικών στοιχείων

Αφού έχει γίνει η επιλογή όλων των παραπάνω, πατώντας την επιλογή Start new session ξεκινάει η διαδικασία δημιουργίας ενός αρχείου csv το οποίο θα περιέχει τις μετρήσεις των επιλεγμένων μετρικών στοιχείων ανάμεσα στις ημερομηνίες που έχει θέσει ο χρήστης.

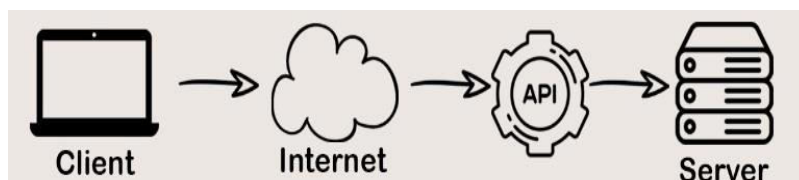


## 2.1.5 Api code examples

Το API είναι το ακρωνύμιο του (Application Programming Interface), το οποίο είναι ένα ενδιάμεσο λογισμικό που επιτρέπει σε δύο εφαρμογές να ανταλλάσσουν μεταξύ τους πληροφορίες. Ουσιαστικά ένα API είναι ένας σύνδεσμος ο οποίος περιέχει πληροφορίες ενός server ή μιας συσκευής.

Όταν χρησιμοποιούμε μια εφαρμογή στο κινητό ή τον υπολογιστή μας, η εφαρμογή συνδέεται στο διαδίκτυο και στέλνει δεδομένα σε έναν διακομιστή. Στη συνέχεια, ο διακομιστής ανακτά αυτά τα δεδομένα, τα ερμηνεύει, εκτελεί τις απαραίτητες ενέργειες και τα στέλνει πίσω στην συσκευή (Εικόνα 11). Έπειτα, η εφαρμογή ερμηνεύει αυτά τα δεδομένα και μας παρουσιάζει τις πληροφορίες που αναζητούμε με ευανάγνωστο τρόπο. Αυτό είναι ουσιαστικά ένα API και όλες οι παραπάνω ενέργειες συμβαίνουν μέσω διεργασιών διαφορετικών API.

Τα API προσφέρουν επίσης ένα επίπεδο ασφάλειας καθώς το καθένα επικοινωνεί με μικρά πακέτα δεδομένων, μοιράζοντας μόνο ό,τι είναι απαραίτητο. Υπάρχουν δημόσια API αλλά και περιορισμένης προσβασιμότητας τα οποία μπορούν να ρυθμιστούν με διάφορους παραμέτρους από τους προγραμματιστές έτσι ώστε να αποδώσουν στο ευρύ κοινό μόνο τις απαραίτητες πληροφορίες. [18]



Εικόνα 11 Παράδειγμα λειτουργίας ενός API

### Παράδειγμα από το την επιλογή API code examples της πλατφόρμας Smart

Η επιλογή API code examples διαθέτει δύο διαφορετικά παραδείγματα κώδικα γραμμένα στην γλώσσα προγραμματισμού Python, τα οποία υπάρχουν ώστε να μπορέσει ο χρήστης να πραγματοποιήσει αυτοματισμούς με την χρήση του API.

Παρακάτω αναλύονται οι εντολές του πρώτου παραδείγματος:

#### #Example 1

Εδώ εμφανίζεται το 1<sup>ο</sup> παράδειγμα από το API code examples

```
Import json, urllib, request
```

```
Import pandas as pd
```

Με τις πρώτες δύο εντολές εισάγονται οι απαραίτητες βιβλιοθήκες που θα χρησιμοποιηθούν παρακάτω στον κώδικα. Εν προκειμένω, αυτές οι βιβλιοθήκες

επιτρέπουν τη λήψη μετρήσεων από ένα web API, την αποθήκευσή και επεξεργασία των δεδομένων σε αρχεία json και τέλος την επεξεργασία τους ως πίνακες με τη βιβλιοθήκη pandas.

```
getdata = urllib request.urlopen
```

```
('https://smart.ece.uowm.gr/api/?station=1&metric=1&from=01/02/20&api_key=1234567890123456789012345678901234567890').read()
```

```
apidata=json.load(getdata)
```

Η παραπάνω εντολή ανοίγει το σύνδεσμο `https://[..]` μέσω της βιβλιοθήκης `urllib` και αποθηκεύει τα δεδομένα που επιστρέφει ο σύνδεσμος στη μεταβλητή `getdata`. Στο σύνδεσμο διακρίνεται ότι σε αυτό το παράδειγμα θα ληφθούν δεδομένα από τον πρώτο σταθμό, το πρώτο μετρικό, ξεκινώντας στις 01/02/20 έως και σήμερα.

```
list = []
```

Έπειτα δημιουργούμε μία κενή λίστα, η οποία ορίζεται στη μεταβλητή `list`.

```
for val in apidata['data']:
```

```
list.append([val['timestamp'],val['value']])
```

Με την παραπάνω δομή επανάληψης, κάθε τιμή από τα δεδομένα τοποθετείται στην κενή λίστα που δημιουργήθηκε νωρίτερα. Τα δεδομένα σε αυτό το παράδειγμα είναι η στήλη `timestamp` με τις ημερομηνίας και η στήλη `value` που θα λάβει το όνομα και τις τιμές του μετρικού 1, που για παράδειγμα μπορεί να είναι η παραγωγή από τα φωτοβολταϊκά.

```
#Initialize a dataframe from list
```

```
df = pd.DataFrame(list,columns=['timestamp', apidata['metric']['label']])
```

Τα δεδομένα έπειτα μεταφέρονται από τη λίστα σε ένα πίνακα δεδομένων (`dataframe`) τον οποίο μπορεί να χειρίζεται εύκολα η βιβλιοθήκη `pandas`.

```
#Set column as index
```

```
df = df.set_index('timestamp')
```

Στον πίνακα δεδομένων ορίζεται ότι η πρώτη στήλη του, που είναι η χρονολογία, θα χρησιμοποιείται ως δείκτης για τις υπόλοιπες στήλες.

```
#Optional print dataframe
```

```
print ('\nDataFrame with column as index\n',df)
```

```
df.to_excel ('apidata.xls')
```

Τέλος, τα δεδομένα εκτυπώνονται στο τερματικό για εποπτεία και γίνεται η μετατροπή του πίνακα δεδομένων σε ένα αρχείο `excel`, το οποίο αποθηκεύεται στο σκληρό δίσκο.

## 2.2 Σύστημα εποπτείας Living lab SCADA

### 2.2.1 Device status

Στο Device status (Εικόνα 10) της πλατφόρμας SMART μπορούμε να δούμε σε ζωντανό χρόνο το σύνολο των συσκευών που είναι τοποθετημένες στο ζωντανό εργαστήριο και να ενημερωθούμε για την κατάσταση τους, δηλαδή το πότε ήταν η τελευταία φορά που φάνηκαν να είναι ενεργές στο σύστημα, επίσης πότε ήταν η τελευταία φορά που βρέθηκαν απενεργοποιημένες ή εκτός σύνδεσης αλλά και την ταχύτητα απόκρισης.

#	Last online	Last offline	Latency
15U1 - Αντιστροφέας 2 - KOSTAL - 10.0.22.34	3 minutes ago	Never	0.0139649s
4U1 - Αντιστροφέας 1 - KOSTAL - 10.0.22.33	3 minutes ago	Never	0.0135980s
BYD Batteries - Fronius - 10.0.22.63	3 minutes ago	July 1 (5 minutes)	0.0000000s
BYD Batteries - Kostal Αριστέρα - 10.0.22.61	3 minutes ago	July 1 (9 minutes 18 seconds)	0.0000000s
BYD Batteries - Kostal δεξιά - 10.0.22.62	3 minutes ago	July 1 (9 minutes 15 seconds)	0.0000000s
CHARGER 1 - φορτιστής αυτοκινήτου - 10.0.22.91	2 minutes ago	Never	0.0426731s
Fronius GEN24 (Berlin) - 10.0.22.53	3 minutes ago	Never	0.0420561s

Εικόνα 12 Κατάσταση συσκευών Device Status

Πίνακας 2 Λίστα συσκευών Device Status

Αντιστροφέας/Inverter (Kostal)
Μπαταρίες (Kostal)
Φορτιστής αυτοκινήτου (Legrand)
Μετρητές ηλεκτρικής ενέργειας (Janitza)
Μετρητές θερμότητας
Controller
Κεντρικό ρούτερ της database
Μετεωρολογικός σταθμός

Το User Interface της σελίδας του Device status είναι δημιουργημένο με τη βοήθεια ενός λογισμικού που ονομάζεται PHP Server Monitor και συγκεκριμένα της έκδοσης v3.5.2. Το PHP Server Monitor (Εικόνα 11) είναι ένα πακέτο κώδικα το οποίο αφού τροφοδοτηθεί με πληροφορίες μέσω ενός εσωτερικού δικτύου πραγματοποιεί ping

στις συσκευές με τις οποίες είναι συνδεδεμένο ή ανοίγει κάποιες από τις σελίδες των συσκευών και ελέγχει πως οι συσκευές αυτές που του έχουν ανατεθεί λειτουργούν φυσιολογικά. Αν παρατηρηθεί από το λογισμικό ότι κάποια από τις συσκευές δε λειτουργεί όπως πρέπει ή δεν έχει ανταποκριθεί μέσα σε έναν συγκεκριμένο χρονικό διάστημα που ορίζεται από τον χρήστη και στην περίπτωση της σελίδας Smart Device status είναι τα πέντε λεπτά, προχωρεί σε ενημέρωση του χρήστη μέσω ειδοποιητικού μηνύματος email.

The screenshot displays the PHP-Server monitor interface with the following components:

- Service Overview:**
  - Service: 15U1 - Αντιστροφέας 2 - KOSTAL - 10.0.22.34
  - Domain/IP: 10.0.22.34
  - Port: 80
  - Status: on
  - Latency: 0.0607 seconds
- Settings:**
  - Type: Service
  - Warning threshold: 1
  - Timeout: 10 seconds
- Status:**
  - Last online: about a minute ago
  - Last offline: September 18
  - Last check: about a minute ago
- Output:**
  - Last error: Connection timed out
  - Last positive output:
  - Last error output: Could not get headers. probably HTTP 50x error.
- Monitoring:**
  - Monitoring:
  - Email:
  - SMS:
  - Pushover:
  - Telegram:
  - Jabber:

Εικόνα 13 PHP-Server monitor

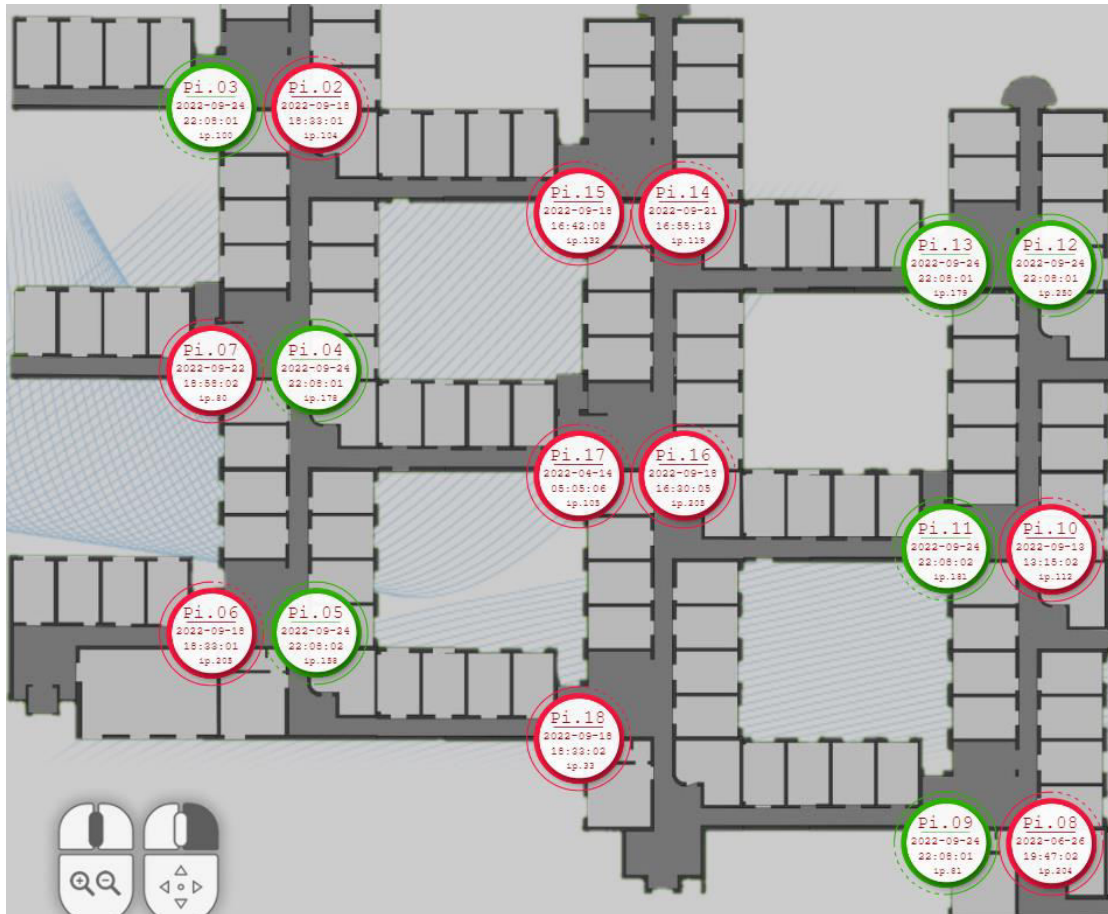
## 2.2.2 Status maps

Στο Status Maps (Εικόνα 14) ο χρήστης μπορεί να παρατηρήσει την κατάσταση των συσκευών που δουλεύουν με ασύρματο δίκτυο μέσα στο ζωντανό εργαστήριο του Πανεπιστημίου. Στον χώρο των φοιτητικών εστιών υπάρχουν τοποθετημένοι δεκαοκτώ microcontroller Raspberry Pi όπως φαίνονται και στον παρακάτω χάρτη του Status maps. Το Raspberry Pi (Εικόνα 15) είναι ένας υπολογιστής τσέπης ο οποίος είναι εξοπλισμένος με κάρτα wifi, κάρτα SD, δύο έως οκτώ gigabyte μνήμης ram, έναν απλό τετραπύρηνo επεξεργαστή και λειτουργεί με λογισμικό Linux.

Διαθέτει τέσσερις εξόδους usb μαζί με δύο εξόδους mini hdmi, υποδοχή καλωδίου ethernet και υποδοχές για οθόνη και κάμερα. Στους microcontroller Raspberry Pi έχει συνδεθεί ένα θερμόμετρο DHT11 και μαζί έχουν τοποθετηθεί στα κεντρικά σημεία της κάθε πτέρυγας του κτηρίου των φοιτητικών εστιών. Χρησιμοποιούνται για την καταγραφή της εσωτερικής θερμοκρασίας αλλά και του ποσοστού της υγρασίας.

Λειτουργούν με έναν αλγόριθμο της γλώσσας προγραμματισμού Python και αποθηκεύουν τις μετρήσεις τους σε αρχεία μορφής csv τα οποία στη συνέχεια προωθούνται στην database της πλατφόρμας μέσω ενός ακόμη αλγορίθμου Python που υπάρχει ενσωματωμένος σε καθένα από αυτά. Στον χώρο των φοιτητικών εστιών είναι τοποθετημένες επίσης εκατόν πενήντα “έξυπνες” πρίζες της εταιρείας Ham, οι οποίες δε φαίνονται προς το παρόν στο Status maps αλλά θα αποτελέσουν μια μελλοντική αναβάθμιση του. Στο κάθε δωμάτιο της κεντρικής πτέρυγας έχουν τοποθετηθεί τρεις έξυπνες πρίζες οι οποίες υπολογίζουν όλα τα μεγέθη του ηλεκτρικού ρεύματος που τις διαρρέει αποδίδοντας έτσι τη συνολική κατανάλωση ρεύματος που είναι το βασικό μας ζητούμενο. Η βασική διαφορά των έξυπνων πριζών με τους microcontroller Raspberry Pi είναι πως αν χάσουν την σύνδεση τους με το ίντερνετ δεν μπορούν να αποθηκεύσουν τις μετρήσεις τους σε αντίθεση με τα Raspberry Pi που μπορούν να αποθηκεύουν τις μετρήσεις σε μια εσωτερική κάρτα SD που διαθέτουν ακόμη και σε λειτουργία εκτός δικτύου. Όλοι οι αισθητήρες είναι συνδεδεμένοι μέσω ενός switch με έναν κεντρικό υπολογιστή ο οποίος βρίσκεται τοποθετημένος σε ένα δωμάτιο κεντρικού ελέγχου όπου με την βοήθεια του απαραίτητου λογισμικού και ενός εσωτερικού VPN (Virtual Private Network) δικτύου γίνεται η καταγραφή των συλλεγμένων μετρήσεων σε μια database η οποία στην συνέχεια προωθείται με τη μορφή αρχείου csv και είναι προσβάσιμη στους χρήστες της πλατφόρμας Smart. Με το πράσινο χρώμα απεικονίζονται οι αισθητήρες που έχουν συνδεθεί πρόσφατα στο wifi ενώ με κόκκινο αυτοί που ενώ λειτουργούν κανονικά καταγράφοντας τις μετρήσεις τους λόγω κάποιου τεχνικού προβλήματος έχουν αρκετό καιρό να αποστείλουν τα δεδομένα τους μέσω του δικτύου wifi.

Έξω από τις φοιτητικές εστίες έχει τοποθετηθεί ένας σταθμός φόρτισης ηλεκτρικών αυτοκινήτων της εταιρείας Legrand Premium Green'up Station 22kW, 11/22 kVA ο οποίος είναι επίσης συνδεδεμένος με το κεντρικό VPN δίκτυο και μας δίνει πληροφορίες από τα μεγέθη της κατανάλωσης του. Πάνω από τις φοιτητικές εστίες υπάρχει επίσης τοποθετημένος συνδεδεμένος με το εσωτερικό δίκτυο ένας μετεωρολογικός σταθμός ο οποίος έχει την δυνατότητα να μετράει τα μεγέθη της κατεύθυνσης και την έντασης του ανέμου, την βροχόπτωση, την ηλιοφάνεια αλλά και την εξωτερική θερμοκρασία.



Εικόνα 14 Status maps (Smart.ece.uowm.gr)



Εικόνα 15 Raspberry PI

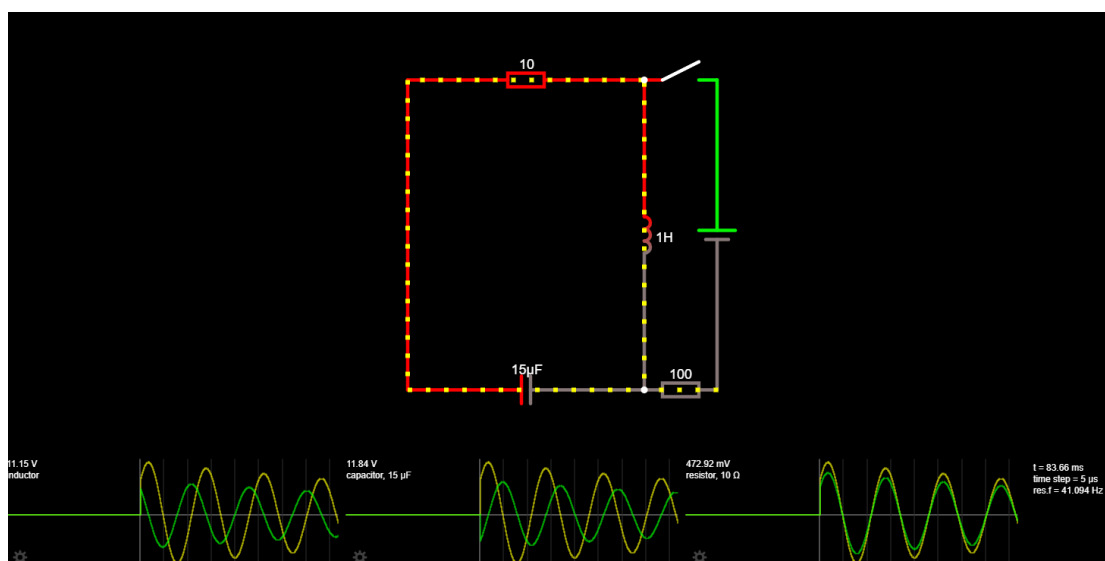
## 2.3 Online Tools

### 2.3.1 Προσομοίωση κυκλωμάτων (Circuit simulation)

Το εργαλείο του προσομοιωτή κυκλώματος Circuit simulation (Εικόνα 16) προσφέρει στον χρήστη πολλές δυνατότητες για την κατασκευή ενός απλού αλλά και σύνθετου ηλεκτρικού κυκλώματος. Το κύκλωμα μπορεί να κατασκευαστεί αποτελούμενο από όλα τα γνωστά ηλεκτρικά εξαρτήματα τα οποία στη συνέχεια μπορούν να ρυθμιστούν

με συγκεκριμένες τιμές. Έτσι μπορούν να συγκριθούν και να ελεγχθούν τα παραγόμενα αποτελέσματα ή να εντοπιστούν τυχόν λάθη.

Υπάρχουν αρκετές διαφορετικές ρυθμιστικές επιλογές ανάλογα με τις προτιμήσεις του χρήστη σχετικά με τον χρωματισμό και το μέγεθος προβολής. Προσφέρεται επίσης μια μεγάλη ποικιλία από έτοιμα κυκλώματα όπως κυκλώματα RLC, ψηφιακά κυκλώματα λογικών πυλών, κυκλώματα φίλτρων και πολλά άλλα, τα οποία ο χρήστης μπορεί να επεξεργαστεί πλήρως για την κατασκευή μεγαλύτερων συνδυαστικών κυκλωμάτων.



Εικόνα 16 (Smart.ece.uowm.gr) επιλογή προσομείωσης κυκλώματος

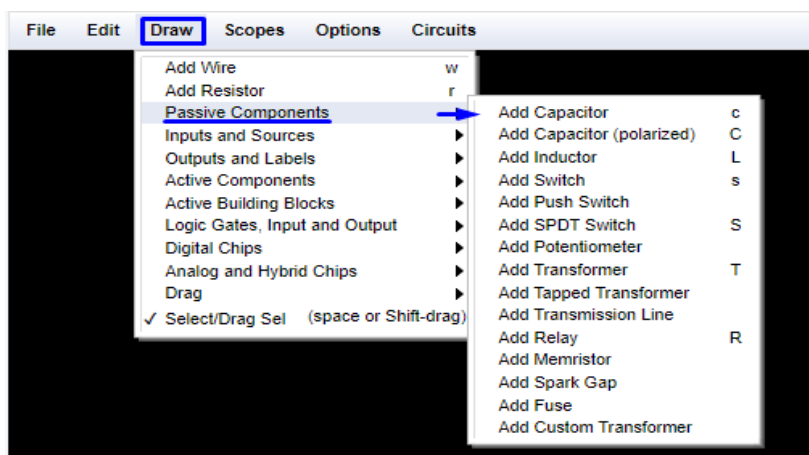
Αρχικά στον χρήστη δίνεται η δυνατότητα εισαγωγής αρχείου ενός κυκλώματος που υπάρχει ήδη στον υπολογιστή του αλλά και η εισαγωγή κώδικα αρχείου ενός προϋπάρχοντος κατασκευασμένου κυκλώματος ή ενός αρχείου που έχει ληφθεί μέσω του Dropbox.

Μετά την ολοκλήρωση της κατασκευής ενός κυκλώματος δίνεται η δυνατότητα της εξαγωγής του κυκλώματος αυτού μέσω: ενός συνδέσμου, κειμένου κώδικα του προγράμματος, ενός στιγμιότυπου - φωτογραφίας αλλά και με της δυνατότητας εκτύπωσης του κυκλώματος.

Στην επιλογή της επεξεργασίας παρέχονται στον χρήστη κάποιες τυπικές δυνατότητες επεξεργασίας όπως: αποκοπή, αντιγραφή, διπλασιασμός, κεντράρισμα του κυκλώματος στο μέσω της οθόνης και μεγέθυνση ή σμίκρυνση στο επιθυμητό μέγεθος προβολής, που είναι πλέον συνηθισμένες σε όλα τα σύγχρονα προγράμματα.

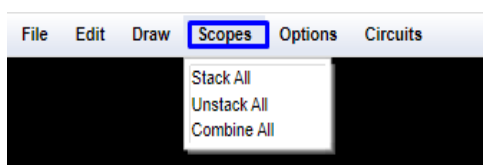
Ο σχεδιασμός είναι το βασικό εργαλείο του χρήστη για την κατασκευή ενός κυκλώματος. Υπάρχουν επιλογές με όλα τα γνωστά ηλεκτρικά εξαρτήματα από πολύ

απλά έως και τα πιο σύνθετα (Εικόνα 17), τα οποία είναι στη διάθεση του ανάλογα με το πρόγραμμα που θέλει να κατασκευάσει. Έπειτα μπορεί να ρυθμίσει το κάθε εξάρτημα με την επιθυμητή τιμή προσαρμόζοντας το κύκλωμα στις δικές του επιθυμίες και ανάγκες. Κάθε κατηγορία υλικών έχει υποκατηγορίες με όλα τα σχετικά ηλεκτρικά εξαρτήματα που προκύπτουν από αυτό.



Εικόνα 17 Circuit Simulation επιλογή Draw

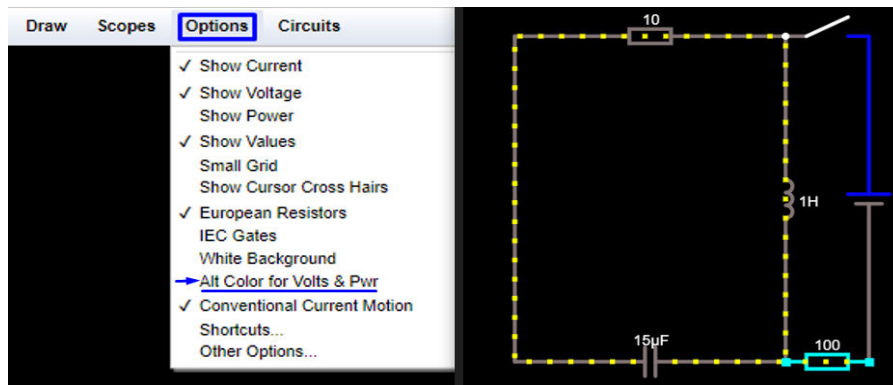
Επίσης, στον χρήστη δίνεται η δυνατότητα εναλλαγής των μετρητικών κυματομορφών σε τρεις διαφορετικές προβολές (Εικόνα 18): σε ξεχωριστά πλαίσια, σε παράλληλη παράθεση και σε σύγκριση όλων των κυματομορφών μαζί.



Εικόνα 18 Circuit Simulation επιλογή Scopes

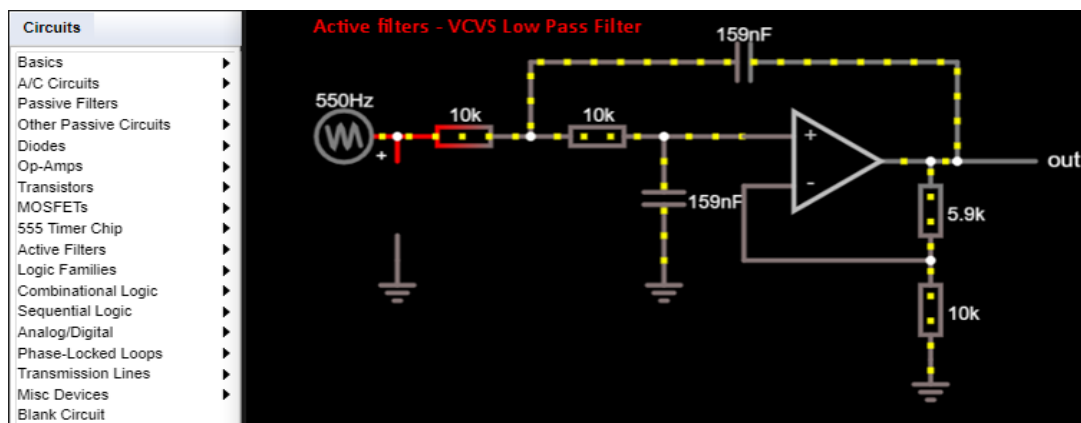
Στις επιλογές (Εικόνα 19) υπάρχουν διάφορες ενδείξεις σχετικά με τα στοιχεία του κυκλώματος που προβάλλονται. Ο χρήστης ανάλογα με τις ανάγκες του μπορεί να επιλέξει να προβληθούν περισσότερα ή λιγότερα αποτελέσματα του κυκλώματος όπως η ένταση του ρεύματος, η τάση, η ισχύς αλλά και ακόμα περισσότερες λεπτομέρειες όπως τα χρώματα προβολής του κάθε στοιχείου ή το μέγεθος του κάτι που κάνει κάθε κύκλωμα ξεχωριστό, ευανάγνωστο και πολύ πιο εύκολο να κατανοηθεί από ένα τρίτο πρόσωπο. Υπάρχουν επίσης επιπρόσθετες ρυθμίσεις δίπλα στο κύκλωμα που αφορούν την ταχύτητα της προσημείωσης και του ρεύματος αλλά και την εύκολη αλλαγή στις τιμές των ηλεκτρικών στοιχείων του κυκλώματος.



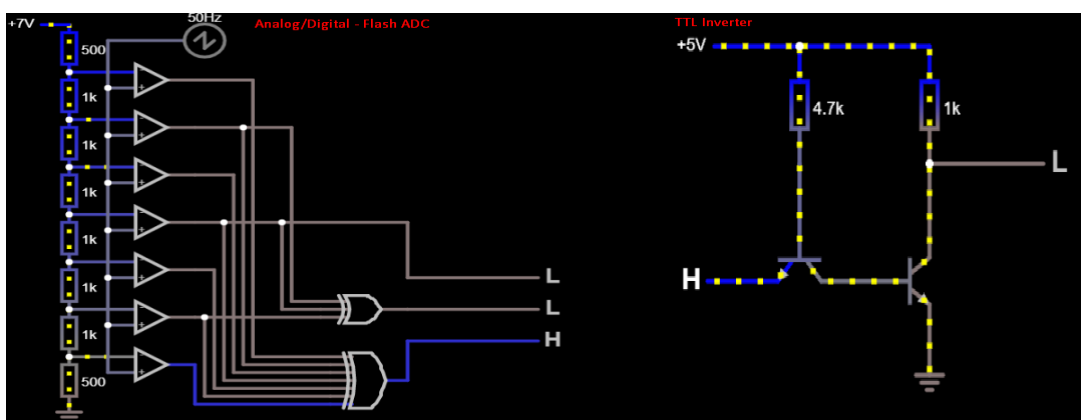


Εικόνα 19 Circuit Simulation τμήμα επιλογών (Options)

Στην επιλογή κυκλώματα υπάρχουν πάρα πολλά έτοιμα κυκλώματα τα οποία είναι πλήρως κατασκευασμένα αλλά δίνουν επίσης τη δυνατότητα στον χρήστη να πραγματοποιήσει ανάπτυξη πάνω τους ή σε συνδυασμό μεταξύ διαφορετικών κυκλωμάτων. Παρακάτω υπάρχουν παραδείγματα τέτοιων κυκλωμάτων (Εικόνα 20,21):



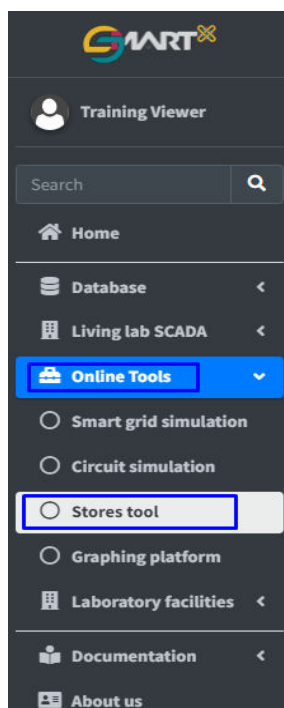
Εικόνα 20 Circuit Simulation – Circuits – Active filters



Εικόνα 21 Circuit Simulation – Circuits – Flash ADC/TTL Inverter

## 2.3.2 Stores Tool

Το εργαλείο Stores Tool (Εικόνα 22) της πλατφόρμας SMART έχει ως βασικό στόχο του τον υπολογισμό του βέλτιστου μεγέθους ενός υβριδικού φωτοβολταϊκού και συστήματος αποθήκευσης ενέργειας ως προς την καθαρή παρούσα αξία της επένδυσης. Τα απαιτούμενα στοιχεία που θα πρέπει να διαθέτει ο χρήστης είναι η κατανάλωση ηλεκτρικής ενέργειας, η ηλιακή ακτινοβολία της συγκεκριμένης περιοχής, το μέγεθος του φωτοβολταϊκού πάρκου και της μπαταρίας αλλά και το κόστος της ηλεκτρικής ενέργειας. Στην συνέχεια πραγματοποιείται μια οικονομική ανάλυση για μια περίοδο περίπου 20 ετών, λαμβάνοντας υπόψη τεχνικές και οικονομικές παραμέτρους και διάφορες επιλογές ενεργειακών πολιτικών. Το εργαλείο μπορεί να χρησιμοποιηθεί τόσο από επαγγελματίες όσο και από μη εξειδικευμένους χρήστες.



Εικόνα 22 Online tools (Smart.ece.uowm.gr) τμήμα Stores tool

Τα βήματα που θα πρέπει να ακολουθήσει ο χρήστης είναι τα εξής:

Στην αρχική επιλογή PV System (Εικόνα 23) θα πρέπει να επιλέξει μία από τις ακόλουθες δύο επιλογές :

Εάν υπάρχουν διαθέσιμες οι μετρήσεις ισχύος ενός φωτοβολταϊκού πάρκου μιας ανάλογης εγκατάστασης, ο χρήστης μπορεί να εισαγάγει αυτά τα δεδομένα σε ένα αρχείο .csv και να τα ανεβάσει κάνοντας κλικ στην επιλογή «Μεταφόρτωση».

Ενώ εάν δεν υπάρχουν διαθέσιμες μετρήσεις ισχύος ανάλογου φωτοβολταϊκού πάρκου τότε ο χρήστης μπορεί να υποδείξει τις συντεταγμένες της θέσης εγκατάστασης και τον προσανατολισμό των πινάκων ώστε να λάβει το σύστημα τα απαραίτητα κλιματικά δεδομένα για τη συγκεκριμένη τοποθεσία.

Εικόνα 23 Stores tool επιλογή PV system

Στην συνέχεια στην επιλογή Consumption Data (Εικόνα 24) θα πρέπει να εισαχθεί η συνολική κατανάλωση της εγκατάστασης. Σε αυτό το σημείο ο χρήστης επιλέγει την χρονική περίοδο της κατανάλωσης για την οποία διαθέτει τις αντίστοιχες μετρήσεις έπειτα την χώρα στην οποία βρίσκεται, την χρονολογία και το ποσό της ενέργειας σε μορφή kWh με την δυνατότητα να ανεβάσει ένα προσαρμοσμένο προφίλ της δικής του εγκατάστασης με όλα τα παραπάνω στοιχεία.

Εικόνα 24 Stores Tool επιλογή Consumption data

Στο επόμενο τμήμα Electricity Costs (Εικόνα 25) απαιτούνται τα δεδομένα από την διακύμανση της τιμής του ρεύματος και ο χρήστης μπορεί να επιλέξει μεταξύ των δύο επιλογών Flat pricing και Dynamic pricing δηλαδή σταθερών χρεώσεων ηλεκτρικής ενέργειας καθ' όλη τη διάρκεια της ημέρας, ή ενός συστήματος «δυναμικής τιμολόγησης» όπου ισχύουν περισσότερες από μία ζώνες ώρας.

Country	Production Cost (€/kWh)	Network Cost (€/kWh)	Taxes (€/kWh)	VAT (%)	Fixed Cost (€/year)
Greece	0.10252	0.02657	0.02494	13	31.15

Εικόνα 25 Stores Tool επιλογή Electricity Costs

Στην επιλογή Storage System (Εικόνα 26) ζητούνται κάποια από τα χαρακτηριστικά του συστήματος αποθήκευσης ενέργειας δηλαδή της μπαταρίας. Ο χρήστης σε αυτό το σημείο θα πρέπει να επιλέξει τον συνολικό αριθμό κύκλων φόρτισης και εκφόρτισης, τα μέγιστα ποσοστά φόρτισης/εκφόρτισης της ονομαστικής χωρητικότητας της μπαταρίας αλλά και το εύρος μπαταρίας (μέγιστο/ελάχιστο) σε μορφή kWh όπως δηλώνεται από τον κατασκευαστή της μπαταρίας.

Parameter	Value	Range
Overall number of charge/discharge cycles	8000	
Charge rate (%)	67	[1 - 100]
Discharge rate (%)	67	[1 - 100]
Usable capacity (%)	80	[1 - 100]
Minimum size	Select minimum battery size [kWh]	[0 - 500]
Maximum size	Select maximum battery size [kWh]	[0 - 500]

Εικόνα 26 Stores Tool επιλογή Storage System

Στο επόμενο βήμα το οποίο είναι το τμήμα Policy (Εικόνα 27) ο χρήστης καλείτε να επιλέξει μεταξύ των τριών πολιτικών από το μενού Επιλογή πολιτικής με την πρώτη επιλογή του να είναι πως η πλεονάζουσα παραγόμενη ενέργεια από το φωτοβολταϊκό πάρκο θα εξάγεται στο δίκτυο και δεν αποζημιώνεται καθόλου. Στην δεύτερη επιλογή η πλεονάζουσα ενέργεια του φωτοβολταϊκού που θα εξάγεται στο δίκτυο και θα υπάρχει αποζημίωση ανάλογης τιμής. Η τρίτη και τελευταία επιλογή η οποία αναφέρεται σε μια πολιτική «net-metering», ο αγοραστής χρεώνεται στην καθαρή καταναλωμένη ενέργεια με την τιμή κόστους παραγωγής, στο τέλος κάθε περιόδου τιμολόγησης. Εάν υπάρχει πλεόνασμα παραγόμενης ενέργειας κατά τη διάρκεια μιας περιόδου χρέωσης, αυτή η ποσότητα ενέργειας μεταφέρεται με τη μορφή πιστώσεων ανανεώσιμης ενέργειας στην επόμενη περίοδο χρέωσης. Στο τέλος κάθε έτους, τυχόν πιστώσεις για ανανεώσιμες πηγές ενέργειας διαγράφονται και ο καταναλωτής χρεώνεται στην εισαγόμενη ενέργεια από τα κόστη δικτύου και τους φόρους που δηλώνονται στην καρτέλα “Κόστος ηλεκτρικής ενέργειας”.

The screenshot shows the 'Policy' selection interface. At the top, there are tabs for 'PV System', 'Consumption Data', 'Electricity Costs', 'Storage System', 'Policy' (selected), and 'Financial'. Below the tabs, there is a 'Select Policy' dropdown menu with three options: 'Pure self-consumption', 'Self-consumption', and 'Partial net-metering'. Three red arrows point from the 'Added cost(€/year)', 'Added income(€/year)', and 'Price for selling excess energy(€/kWh)' input fields to the dropdown menu. The 'Price for selling excess energy(€/kWh)' field has a value of '0'.

Εικόνα 27 Stores Tool επιλογή Policy

Τέλος στην επιλογή Financial (Εικόνα 28) ο χρήστης πρέπει να καθορίσει το συνολικό κόστος αγοράς του εξοπλισμού του μαζί με το κόστος λειτουργίας με την τιμή της kWh αλλά και το κόστος συντήρησης.

The screenshot shows the 'Financial' selection interface. At the top, there are tabs for 'PV System', 'Consumption Data', 'Electricity Costs', 'Storage System', 'Policy', and 'Financial' (selected). Below the tabs, there are two input fields: 'PV system and inverter cost\* (€/kWp)' with a value of 1600, and 'BESS cost\* (€/kWh)' with a value of 200.

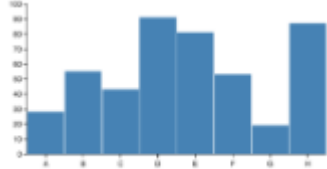
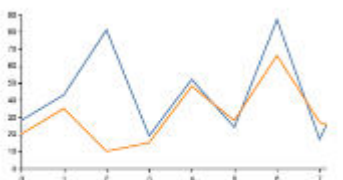

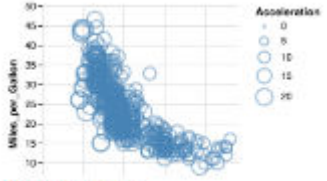
Εικόνα 28 Stores tool επιλογή Financial

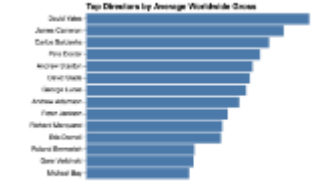

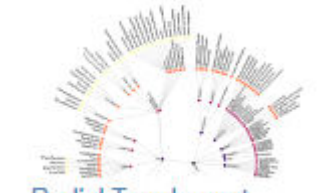

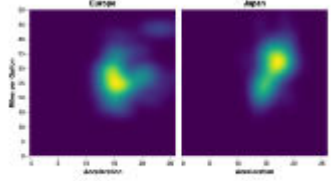


### 2.3.3.1 Graphing Platform (Γραφική πλατφόρμα Vega)

Η γραφική πλατφόρμα της ιστοσελίδας VEGA η οποία είναι μια δηλωτική γλώσσα για τη δημιουργία, την αποθήκευση και την κοινή χρήση σχεδίων διαδραστικής απεικόνισης που παράλληλα προσφέρει μια μεγάλη ποικιλία από προγραμματιζόμενα διαγράμματα. Τα διαγράμματα μπορούν να μεταφερθούν εύκολα στην σελίδα του <https://smart.ece.uowm.gr> μέσω ενός αρχείου JSON στο οποίο περιέχεται ολόκληρος ο κώδικας του διαγράμματος μέσω του οποίου μπορούν να γίνουν όλες οι αλλαγές και τροποποιήσεις που επιθυμεί ο χρήστης. [17]

Τα διαγράμματα χωρίζονται στις εξής κατηγορίες :

Πίνακας 3 Τύποι διαγραμμάτων VEGA

Διάγραμμα με μπάρες ποσοστών (Bar chart)	 <p>Bar Chart</p>
Γραμμικό διάγραμμα (Line chart)	 <p>Line Chart</p>
Καμπυλωτό διάγραμμα (Circular chart)	 <p>Donut Chart</p>
Διάγραμμα ποσοτικής διασποράς (Scatter chart)	 <p>Scatter Plot</p>

<p>Βαθμιδωτό διάγραμμα (Distribution chart)</p>	 <p>Top K Plot</p>
<p>Διάγραμμα γεωγραφικού χάρτη (Geographic chart)</p>	 <p>World Map</p>
<p>Διάγραμμα μορφής “δέντρου” (Tree diagram)</p>	 <p>Radial Tree Layout</p>
<p>Διαδικτυακό διάγραμμα (Network diagram)</p>	 <p>Force Directed Layout</p>
<p>Διαγράμματα διαφόρων μορφών (Other chart types)</p>	 <p>Density Heatmaps</p>
<p>Διαγράμματα κατασκευασμένα από χρήστες (Custom designed diagrams)</p>	 <p>Falkensee Population</p>
<p>Διαδραστικά διαγράμματα (Interaction diagrams)</p>	 <p>Interactive Legend</p>

### 2.3.3.2 Παράδειγμα κώδικα διαγραμμάτων VEGA

Παρακάτω αναλύονται δύο παραδείγματα αρχείων JSON από διαγράμματα της VEGA.

Κώδικας αρχείου JSON γραμμικού διαγράμματος/line chart (Εικόνα 29) :

```
"$schema": "https://vega.github.io/schema/vega/v5.json"
```

Σύνδεσμος που αντλούνται πληροφορίες από την ιστοσελίδα της VEGA για το συγκεκριμένο παράδειγμα.

```
"description": "A basic line chart example."
```

Απλή και σύντομη περιγραφή του διαγράμματος.

```
"width": 500,
```

```
"height": 200,
```

```
"padding": 5,
```

Με τις παραπάνω εντολές πραγματοποιείται η τροποποίηση των διαστάσεων του διαγράμματος.

```
"signals":
```

```
{
```

```
"name": "interpolate",
```

```
"value": "linear",
```

```
"bind": {
```

```
"input": "select",
```

```
"options": [
```

```
"basis",
```

```
"cardinal",
```

```
"catmull-rom",
```

```
"linear",
```

```
"monotone",
```

```
"natural",
```

```
"step",
```

```
"step-after",
```

```
"step-before"
```

Οι παραπάνω εντολές είναι επιλογές που μπορεί να τροποποιήσει ο χρήστης στο dropdown menu αφού έχει δημιουργήσει το διάγραμμα. Είναι συναρτήσεις που



κάνουν μοντέλα regression βάσει των σημείων που έδωσε ο χρήστης στο αρχείο JSON.

```
"data": [  
  {  
    "name": "table",  
    "values": [  
      {"x": 0, "y": 28, "c": 0}, {"x": 0, "y": 20, "c": 1},  
      {"x": 1, "y": 43, "c": 0}, {"x": 1, "y": 35, "c": 1},  
      {"x": 2, "y": 81, "c": 0}, {"x": 2, "y": 10, "c": 1},  
      {"x": 3, "y": 19, "c": 0}, {"x": 3, "y": 15, "c": 1},  
      {"x": 4, "y": 52, "c": 0}, {"x": 4, "y": 48, "c": 1},  
      {"x": 5, "y": 24, "c": 0}, {"x": 5, "y": 28, "c": 1},  
      {"x": 6, "y": 87, "c": 0}, {"x": 6, "y": 66, "c": 1},  
      {"x": 7, "y": 17, "c": 0}, {"x": 7, "y": 27, "c": 1},  
      {"x": 8, "y": 68, "c": 0}, {"x": 8, "y": 16, "c": 1},  
      {"x": 9, "y": 49, "c": 0}, {"x": 9, "y": 25, "c": 1}  
    ]  
  }  
]
```

Στις παραπάνω εντολές του πίνακα εισάγονται όλα τα δεδομένα που απεικονίζονται στο διάγραμμα δηλαδή οι τιμές από όλες τις γραμμές και τις στήλες.

```
"scales": [  
  {  
    "name": "x",  
    "type": "point",  
    "range": "width",  
    "domain": {"data": "table", "field": "x"}  
  },  
  {  
    "name": "y",  
    "type": "linear",  
    "range": "height",  
    "nice": true,  
    "zero": true,  
    "domain": {"data": "table", "field": "y"}  
  }  
]
```

```
{  
  "name": "color",  
  "type": "ordinal",  
  "range": "category",  
  "domain": {"data": "table", "field": "c"}
```

Στις παραπάνω εντολές ρυθμίζεται ο τρόπος απεικόνισης του διαγράμματος από τους δύο άξονες σχετικά με τον τρόπο αύξησής τους και από ποιο σύνολο δεδομένων αντλούν τα δεδομένα.

```
  "axes": [  
    {"orient": "bottom", "scale": "x"},  
    {"orient": "left", "scale": "y"}  
  ]
```

Οι παραπάνω εντολές αλλάζουν επίσης ρυθμίσεις για τους άξονες και καθορίζουν το πόσα σημεία εμφανίζονται στον οριζόντιο άξονα.

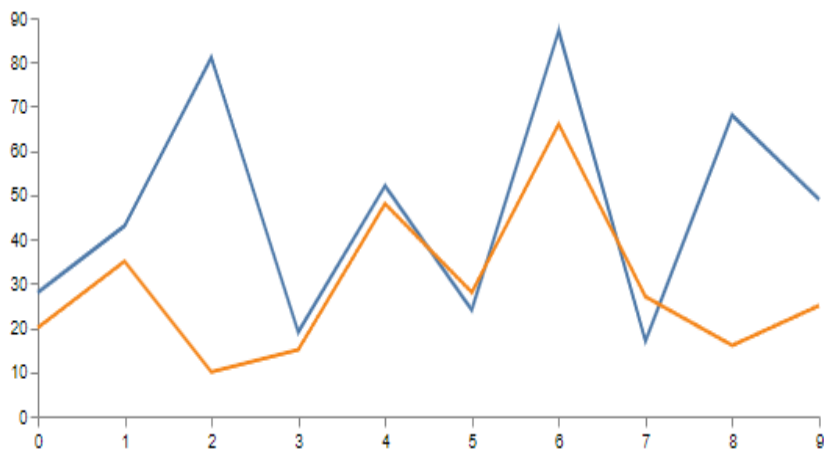
```
  "marks":  
  {  
    "type": "group",  
    "from": {  
      "facet": {  
        "name": "series",  
        "data": "table",  
        "groupby": "c"  
      }  
    },  
    "marks": [  
      {  
        "type": "line",  
        "from": {"data": "series"},  
        "encode": {  
          "enter": {  
            "x": {"scale": "x", "field": "x"},  
            "y": {"scale": "y", "field": "y"},  
            "stroke": {"scale": "color", "field": "c"},
```

```

"strokeWidth": {"value": 2}
}
},
"update": {
"interpolate": {"signal": "interpolate"},
"strokeOpacity": {"value": 1}
},
"hover": {
"strokeOpacity": {"value": 0.5}
}
},

```

Οι παραπάνω είναι ρυθμίσεις που μπορεί να τροποποιήσει ο χρήστης για τα σημεία που απεικονίζονται στο διάγραμμα όπως το χρώμα η διαφάνεια τους.



Εικόνα 29 Γραμμικό Διάγραμμα – Line Chart

Κώδικας αρχείου JSON διαγράμματος ακτινωτής διάταξης δέντρου (Εικόνα 30) :

```

"$schema": "https://vega.github.io/schema/vega/v5.json",

```

Σύνδεσμος που αντλούνται πληροφορίες από την ιστοσελίδα της VEGA για το συγκεκριμένο παράδειγμα.

```

"description": "An example of a radial layout for a node-link diagram of hierarchical data.",

```

Απλή και σύντομη περιγραφή του διαγράμματος.

```

"width": 720,

```

```

"height": 720,

```

```

"padding": 5,

```

```
"autosize": "none",
```

Με τις παραπάνω εντολές μπορεί να γίνει τροποποίηση των διαστάσεων του διαγράμματος.

```
"signals": [  
  {  
    "name": "labels", "value": true,  
    "bind": {"input": "checkbox"}  
  },  
  {  
    "name": "radius", "value": 280,  
    "bind": {"input": "range", "min": 20, "max": 600}  
  },  
  {  
    "name": "extent", "value": 360,  
    "bind": {"input": "range", "min": 0, "max": 360, "step": 1}  
  },  
  {  
    "name": "rotate", "value": 0,  
    "bind": {"input": "range", "min": 0, "max": 360, "step": 1}  
  },  
  {  
    "name": "layout", "value": "tidy",  
    "bind": {"input": "radio", "options": ["tidy", "cluster"]}  
  },  
  {  
    "name": "links", "value": "line",  
    "bind": {  
      "input": "select",  
      "options": ["line", "curve", "diagonal", "orthogonal"]  
    }  
  },  
  {"name": "originX", "update": "width / 2"},  
  {"name": "originY", "update": "height / 2"}  
],
```

Οι παραπάνω εντολές είναι επιλογές που μπορεί να τροποποιήσει ο χρήστης στο dropdown menu αφού έχει δημιουργήσει το διάγραμμα. Είναι συναρτήσεις που κάνουν μοντέλα regression βάσει των σημείων που έδωσε ο χρήστης στο αρχείο JSON.

```
"data": [  
  {  
    "name": "tree",  
    "url": "data/flare.json",  
    "transform": [  
      {  
        "type": "stratify",  
        "key": "id",  
        "parentKey": "parent"  
      },  
      {  
        "type": "tree",  
        "method": {"signal": "layout"},  
        "size": [1, {"signal": "radius"}],  
        "as": ["alpha", "radius", "depth", "children"]  
      },  
      {  
        "type": "formula",  
        "expr": "(rotate + extent * datum.alpha + 270) % 360",  
        "as": "angle"  
      },  
      {  
        "type": "formula",  
        "expr": "PI * datum.angle / 180",  
        "as": "radians"  
      },  
      {  
        "type": "formula",  
        "expr": "inrange(datum.angle, [90, 270])",  
        "as": "leftside"      }  
    ]  
  }  
]
```

```

    },
    {
      "type": "formula",
      "expr": "originX + datum.radius * cos(datum.radians)",
      "as": "x"
    },
    {
      "type": "formula",
      "expr": "originY + datum.radius * sin(datum.radians)",
      "as": "y"
    }
  ],
  "name": "links",
  "source": "tree",
  "transform": [
    {"type": "treelinks"},
    {
      "type": "linkpath",
      "shape": {"signal": "links"}, "orient": "radial",
      "sourceX": "source.radians", "sourceY": "source.radius",
      "targetX": "target.radians", "targetY": "target.radius"
    }
  ]

```

Στις εντολές των δεδομένων εισάγονται όλα τα δεδομένα που απεικονίζονται στο διάγραμμα και στο συγκεκριμένο παράδειγμα υπάρχουν επιπλέον οι εντολές για την κυκλική διάταξη των στοιχείων σε μοίρες.

```

"scales": [
  {
    "name": "color",
    "type": "linear",
    "range": {"scheme": "magma"},
    "domain": {"data": "tree", "field": "depth"}
    "zero": true
  }
]

```

Στις παραπάνω εντολές υπάρχουν οι ρυθμίσεις σχετικά με το σχήμα του διαγράμματος.

```
"marks": [  
  {  
    "type": "path",  
    "from": {"data": "links"},  
    "encode": {  
      "update": {  
        "x": {"signal": "originX"},  
        "y": {"signal": "originY"},  
        "path": {"field": "path"},  
        "stroke": {"value": "#ccc"}  
      }  
    },  
    "type": "symbol",  
    "from": {"data": "tree"},  
    "encode": {  
      "enter": {  
        "size": {"value": 100},  
        "stroke": {"value": "#fff"}  
      },  
      "update": {  
        "x": {"field": "x"},  
        "y": {"field": "y"},  
        "fill": {"scale": "color", "field": "depth"}  
      },  
    },  
  },  
  {  
    "type": "text",  
    "from": {"data": "tree"},  
    "encode": {  
      "enter": {  
        "text": {"field": "name"},  
      },  
    },  
  },  
],
```

Οι παραπάνω εντολές είναι ρυθμίσεις που μπορεί να τροποποιήσει ο χρήστης για τα σημεία που απεικονίζονται στο διάγραμμα όπως το χρώμα και η διαφάνεια τους.

```
"type": "text",  
"from": {"data": "tree"},  
"encode": {  
  "enter": {  
    "text": {"field": "name"},  
  },  
},
```

```

"fontSize": {"value": 9},
"baseline": {"value": "middle"}
},
"update": {
"x": {"field": "x"},
"y": {"field": "y"},
"dx": {"signal": "(datum.leftside ? -1: 1) * 6"},
"angle": {"signal": "datum.leftside ? datum.angle - 180 : datum.angle"},
"align": {"signal": "datum.leftside ? 'right': 'left'"},
"opacity": {"signal": "labels ? 1 : 0"}
}

```

Από τις παραπάνω εντολές καθορίζονται το μέγεθος, η γωνία και ο τρόπος απεικόνισης του διαγράμματος.



Εικόνα 30 Διαγράμμα ακτινωτής διάταξης δέντρου - Radial Tree Layout chart



## ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην πτυχιακή εργασία αρχικά αναπτύχθηκε μια λεπτομερή περιγραφή όλων των λογισμικών πακέτων που χρησιμοποιήθηκαν για την δημιουργία και λειτουργία της πλατφόρμας SMART. Στο δεύτερο κεφάλαιο, αρχικά έγινε μια λεπτομερή επεξήγηση όλων των κύριων τμημάτων της. Έπειτα από μια σύντομη περιγραφή για την γενική έννοια μιας βάσης δεδομένων, ακολούθησε η ανάπτυξη του πρώτου κύριου τμήματος της πλατφόρμας που είναι η Database και συμπεριλαμβάνει τα υποτμήματα Single data analysis, Upload measurements, Direct data download και API codes examples. Στην συνέχεια ακολουθεί το δεύτερο κύριο τμήμα της πλατφόρμας SMART, το σύστημα εποπτείας Living lab SCADA το οποίο χωρίζεται σε δύο υποτμήματα το Device status και το Status maps από τα οποία παίρνουμε πολύ χρήσιμες πληροφορίες για όλες τις μετρητικές συσκευές που έχουν τοποθετηθεί στον χώρο του ζωντανού ερευνητικού εργαστηρίου και τροφοδοτούν με τις αντίστοιχες πληροφορίες την βάση δεδομένων της πλατφόρμας. Το τελευταίο αλλά πολύ χρήσιμο τμήμα της πλατφόρμας που αναπτύσσεται είναι το Online tools το οποίο χωρίζεται στα υποτμήματα Circuit simulation, Stores tool και Graphic platform. Το Circuit simulation όπως φαίνεται και στα παραδείγματα που αναπτύχθηκαν προσφέρει την δυνατότητα ανάπτυξης πολλαπλών έτοιμων κυκλωμάτων αλλά και την δημιουργία νέων πειραματικών κυκλωμάτων για την απόκτηση περισσότερων πληροφοριών. Τέλος το τμήμα Graphic platform προσφέρει έναν πολύ αποδοτικό τρόπο παρουσίασης δεδομένων μέσα από την μεγάλη ποικιλία διαγραμμάτων που περιέχει δίνοντας την δυνατότητα μιας πολύ εύκολης και ευπαρουσίαστης αναπαράστασης των δεδομένων. Η αναβάθμιση της πλατφόρμας SMART (<https://smart.ece.uowm.gr>) είναι συνεχής και σταδιακά υπάρχουν νέες προσθήκες ώστε να περιέχει ακόμη περισσότερες πληροφορίες και δυνατότητες για όλους τους χρήστες της.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Joe Kraynak, Joe Habraken - Internet 6 σε 1  
(Β. ΓΚΙΟΥΡΔΑΣ ΕΚΔΟΤΙΚΗ)
- [2] Paul Deitel, Harvey Deitel – Εισαγωγή στην Python  
(Εκδόσεις Μ.Γκιούρδας)
- [3] Luke Welling, Laura Thomson – Ανάπτυξη web εφαρμογών με PHP και MySQL  
(Εκδόσεις Μ.Γκιούρδας)
- [4] <https://www.python.org/>
- [5] [https://www.debian.org/intro/why\\_debian](https://www.debian.org/intro/why_debian)
- [6] <https://www.mongodb.com>
- [7] <https://www.mysql.com>
- [8] <https://www.javatpoint.com/what-is-html>
- [9] <https://httpd.apache.org/>
- [10] <https://httpd.apache.org>
- [11] [http://users.sch.gr/petros\\_salavasidis/\\_tutorials/cscience/php.pdf](http://users.sch.gr/petros_salavasidis/_tutorials/cscience/php.pdf)
- [12] [https://el.wikipedia.org/wiki/JavaScript#cite\\_note-13](https://el.wikipedia.org/wiki/JavaScript#cite_note-13)
- [13] <https://el.wikipedia.org/wiki/Debian>
- [14] <https://geekolar.com/html>
- [15] <https://el.tipsandtrics.com/how-mongodb-database-can-better-organize-your-data-767816>
- [17] <https://vega.github.io/vega/>
- [18] <https://www.mulesoft.com/resources/api/what-is-an-api>
- [19] <https://stackify.com/learn-javascript-tutorials/>