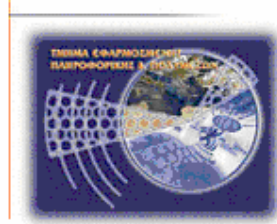


**Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης**  
**Σχολή Τεχνολογικών Εφαρμογών**  
**Τμήμα Μηχανικών Πληροφορικής**



**Πτυχιακή εργασία**

**Τίτλος:**

**Ανάπτυξη portal για τη συλλογή, παρουσίαση και επεξεργασία δεδομένων από αισθητήρες σε πραγματικό χρόνο και ανάπτυξη σχετικής εφαρμογής για παρουσίαση των δεδομένων σε κινητούς χρήστες**

**Παναγιώτης Γιαννικάκης (ΑΜ: 2291)**

**Επιβλέπων καθηγητής : Σπύρος Παναγιωτάκης**

**Επιτροπή Αξιολόγησης :**

**Ημερομηνία παρουσίασης:**

## **Ευχαριστίες**

Θα ήθελα να ευχαριστήσω τον καθηγητή μου κ. Παναγιωτάκη Σπύρο, για την καθοδήγηση, την βοήθεια, που μου προσέφερε, καθώς και την εμπιστοσύνη που μου έδειξε για να φέρω εις πέρας την παρούσα εργασία.

## **Synopsis**

The purpose of this thesis is the implementation of an electronic portal (website) and a mobile application for Android OS, which present in real-time, weather forecast data that are dynamically collected from various sensors. Graphical tools like Grafana and data drawn from Bioklima server of the Technological Institute of Crete are used in order to make weather information beautifully visualised and easily accessible. In addition, the database is developed by using the Influxdb, for the design of the website we chose to use the Bootstrap front-end web framework and many other that are thoroughly described in the next chapters.

## **Σύνοψη**

Σκοπός της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη μιας ηλεκτρονικής πύλης και μίας εφαρμογής για κινητούς χρηστές android που θα συλλέγει σε πραγματικό χρόνο δεδομένα από διάφορους αισθητήρες και θα τα παρουσιάζει σε μια ιστοσελίδα υπό μορφή τιμών και γραφημάτων που θα ανανεώνονται δυναμικά. Για τα γραφήματα θα χρησιμοποιηθεί η πλατφόρμα Grafana, η βάση δεδομένων Influxdb της Influx Data και τα δεδομένα θα αντλούνται από τον διακομιστή bioklima του TEI.

**Πίνακας Περιεχομένων**

<b>1. Εισαγωγή .....</b>	<b>1</b>
1.1. Περίληψη .....	1
1.2. Κίνητρο για την Διεξαγωγή της Εργασίας.....	1
1.3. Σκοπός και Στόχοι Εργασίας .....	1
1.4. Δομή Εργασίας .....	1
<b>2. Διαδίκτυο των πραγμάτων (Internet of Things).....</b>	<b>2</b>
2.1. Ιστορικά στοιχεία .....	3
2.2. Εφαρμογές.....	4
2.3. Λοιπά χαρακτηριστικά .....	6
2.3.1. Νοημοσύνη .....	6
2.3.2. Αρχιτεκτονική.....	6
2.3.3. Αρχιτεκτονική του δικτύου.....	6
2.3.4. Πλαίσιο (Framework) .....	7
2.3.5. Ασφάλεια .....	8
<b>3. BIG DATA.....</b>	<b>10</b>
3.1. Ορισμός.....	11
3.2. Χαρακτηριστικά .....	12
3.3. Αρχιτεκτονική .....	12
3.4. Τεχνολογίες.....	13
3.5. Εφαρμογές.....	14
<b>4. Τεχνολογίες Υλοποίησης.....</b>	<b>16</b>
4.1. HTML5 .....	16
4.2. JavaScript .....	17
4.3. jQuery.....	18
4.4. CSS3.....	19
4.5. Responsive Design .....	19
4.6. Bootstrap .....	20
4.7. InfluxDB .....	21
4.8. Grafana.....	23
4.9. Android Studio .....	24
4.10. Apache HTTP Server .....	25
4.11. Netbeans IDE.....	26
<b>5. Ανάλυση Portal .....</b>	<b>27</b>
5.1. Απαιτήσεις συστήματος .....	27
5.2. Σενάρια.....	28
<b>6. Υλοποίηση Portal .....</b>	<b>32</b>
6.1. Αρχικοποίηση συστήματος .....	32
6.2. Εγκατάσταση της Influxdb .....	35
6.3. Εγκατάσταση του Grafana .....	41
6.4. Εγκατάσταση της Java .....	45
6.5. Εγκατάσταση Netbeans.....	52
6.6. Εγκατάσταση Android Studio.....	56
6.7. Εγκατάσταση του Apache Server .....	64
6.8. Κατέβασμα αρχείων Bootstrap .....	66
6.9. Εισαγωγή δεδομένων στην influxdb.....	67
6.9.1. Εισαγωγή από Web Admin Interface.....	67
6.9.2. Εισαγωγή δεδομένων από HTTP API.....	70

	Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής	
6.9.3.	Εισαγωγή δεδομένων από το Command Line Interface .....	72
6.9.4.	Εισαγωγή δεδομένων με τη βιβλιοθήκη της Java.....	73
6.10.	Δημιουργία γραφημάτων με το Grafana.....	76
6.11.	Δημιουργία ηλεκτρονικής πύλης με το Bootstrap .....	81
6.12.	Δημιουργία εφαρμογής για κινητούς χρηστές με το Android Studio.....	99
<b>7.</b>	<b>Συμπεράσματα .....</b>	<b>123</b>
	<b>Βιβλιογραφία .....</b>	<b>124</b>

**Πίνακας Εικόνων**

Εικόνα 2-1: Internet of Things.....	2
Εικόνα 2-2: 200 Billion objects by 2020.....	3
Εικόνα 2-3: Evolution.....	4
Εικόνα 2-4: Applications.....	5
Εικόνα 2-5: Intelligence.....	6
Εικόνα 2-6: 5G.....	7
Εικόνα 2-7: Safety.....	8
Εικόνα 3-1: Big Data.....	10
Εικόνα 3-2: 3V.....	11
Εικόνα 3-3: Architecture.....	12
Εικόνα 3-4: Data created every minute of the day in 2016.....	14
Εικόνα 4-1: HTML5.....	16
Εικόνα 4-2: JavaScript.....	17
Εικόνα 4-3: jQuery.....	18
Εικόνα 4-4: CSS3.....	19
Εικόνα 4-5: Bootstrap.....	20
Εικόνα 4-6: InfluxDB.....	21
Εικόνα 4-7: Grafana.....	23
Εικόνα 4-8: Android Studio.....	24
Εικόνα 4-9: Apache HTTP Server.....	25
Εικόνα 4-10: Netbeans IDE.....	26
Εικόνα 5-1: Εισαγωγή δεδομένων στην influxdb από HTTP/API.....	29
Εικόνα 5-2: Δημιουργία γραφήματος στο Grafana.....	30
Εικόνα 5-3: Σενάριο Προβολής Πληροφοριών.....	30
Εικόνα 5-4: Προβολή Γραφημάτων στο Grafana.....	31
Εικόνα 6-1: Ubuntu 16 virtual machine.....	32
Εικόνα 6-2: Ubuntu static IP.....	33
Εικόνα 6-3: Ubuntu system update.....	34
Εικόνα 6-4: Ubuntu upgrade command.....	35
Εικόνα 6-5: Add influxdb to repository.....	36
Εικόνα 6-6: Influxdb repository confirmation.....	37
Εικόνα 6-7: Installed influxdb.....	38
Εικόνα 6-8: Service Influxdb status.....	39
Εικόνα 6-9: Influxdb HTTP interface.....	40
Εικόνα 6-10: Grafana add repository.....	41
Εικόνα 6-11: Confirm Grafana repository.....	42
Εικόνα 6-12: Grafana installation.....	43
Εικόνα 6-13: Grafana login interface.....	44
Εικόνα 6-14: Add Oracle repository.....	45
Εικόνα 6-15: Accept agreement.....	46
Εικόνα 6-16: Accept license.....	47
Εικόνα 6-17: Downloading Java 8 JDK.....	48
Εικόνα 6-18: Java version.....	49
Εικόνα 6-19: Java path.....	50
Εικόνα 6-20: Java home.....	51
Εικόνα 6-21: Netbeans Download.....	52
Εικόνα 6-22: Installation wizard.....	53
Εικόνα 6-23: Netbeans Java specified.....	54
Εικόνα 6-24: Netbeans installation finished.....	55
Εικόνα 6-25: Android Studio download page.....	56
Εικόνα 6-26: Android Studio extract.....	57
Εικόνα 6-27: Android studio initiation install.....	58
Εικόνα 6-28: Android Studio no import settings.....	59
Εικόνα 6-29: Android Studio wizard.....	60
Εικόνα 6-30: Android Studio components to download.....	61

Εικόνα 6-31: Android Studio Downloading .....	62
Εικόνα 6-32: Android Studio startup screen.....	63
Εικόνα 6-33: Install Apache Server.....	64
Εικόνα 6-34: Apache Server.....	65
Εικόνα 6-35: Bootstrap download .....	66
Εικόνα 6-36: Influxdb web admin interface .....	67
Εικόνα 6-37: NetBeans csv output .....	68
Εικόνα 6-38: Influxdb http api.....	70
Εικόνα 6-39: Influxdb CLI.....	72
Εικόνα 6-40: NetBeans new Java Project.....	73
Εικόνα 6-41: Netbeans influxdb jar.....	74
Εικόνα 6-42: Grafana first login.....	77
Εικόνα 6-43: Grafana add data source.....	78
Εικόνα 6-44: Grafana select graph type .....	78
Εικόνα 6-45: Graph available settings.....	79
Εικόνα 6-46: Graph general tab.....	79
Εικόνα 6-47: Graph metrics tab.....	79
Εικόνα 6-48: Graph axes tab .....	80
Εικόνα 6-49: Graph display tab .....	80
Εικόνα 6-50: Graph time range tab.....	80
Εικόνα 6-51: Dashboard update time .....	81
Εικόνα 6-52: Grafana share option.....	81
Εικόνα 6-53: Bootstrap extract .....	82
Εικόνα 6-54: Portal.....	97
Εικόνα 6-55: Portal bioklima page .....	97
Εικόνα 6-56: Portal about page .....	98
Εικόνα 6-57: Android Studio App.....	99
Εικόνα 6-58: Android project name .....	99
Εικόνα 6-59: Android activity choose.....	100
Εικόνα 6-60: Android activity name .....	101
Εικόνα 6-61: Android project refactor.....	102
Εικόνα 6-62: Android app main activity .....	121
Εικόνα 6-63: Android app widget .....	122



## Λίστα Πινάκων

Πίνακας 5-1: Προσδιορισμός απαιτήσεων .....	27
Πίνακας 5-2: Περιπτώσεις Χρήσης .....	28
Πίνακας 5-3: Σενάριο Διαχείρισης Συστήματος.....	28

# 1. Εισαγωγή

Μια σύντομη εισαγωγή για την πτυχιακή εργασία και τι θα ακολουθήσει στα επόμενα κεφάλαια.

## 1.1. Περίληψη

Σκοπός της πτυχιακής είναι η ανάπτυξη ενός portal (ηλεκτρονική πύλη) και μίας εφαρμογής που θα συλλέγει σε πραγματικό χρόνο δεδομένα από διάφορους αισθητήρες (π.χ. μετεωρολογικά και δορυφορικά δεδομένα) ή web cameras και θα τα παρουσιάζει σε μια ιστοσελίδα υπό μορφή τιμών και γραφημάτων που θα ανανεώνονται δυναμικά. Κατόπιν συζήτησης με τον επιβλέποντα καθηγητή η βιβλιοθήκη γραφημάτων η οποία θα χρησιμοποιηθεί είναι η Grafana, η βάση δεδομένων Influxdb και τα δεδομένα θα αντλούνται από τον διακομιστή bioklima του TEI.

Η ηλεκτρονική πύλη και η εφαρμογή θα πρέπει να είναι φιλική προς το χρήστη, να είναι συμβατή με το υπάρχον portal του TEI, να επιτρέπεται η διασύνδεση των δεδομένων μεταξύ της παλαιάς, της νέας πύλης και της εφαρμογής, να είναι συμβατό με κινητές συσκευές και τα γραφήματα να υλοποιηθούν με το πλαίσιο εφαρμογής Grafana. Η εφαρμογή για τις κινητές συσκευές να είναι λειτουργική σε Android.

## 1.2. Κίνητρο για την Διεξαγωγή της Εργασίας

Κίνητρο μου για την διεξαγωγή της παρούσας διπλωματικής εργασίας είναι η εξοικείωση μου με τεχνολογίες αιχμής και η απόκτηση δεξιοτήτων για τον σχεδιασμό και ανάπτυξη ηλεκτρονικών πυλών και εφαρμογών για κινητές συσκευές. Πιο συγκεκριμένα θα εξοικειωθώ με τεχνολογίες όπως το πλαίσιο εφαρμογής Bootstrap, το CSS3, την γλώσσα HTML5, το Grafana, την Java, την JavaScript, το jQuery, την βάση δεδομένων Influxdb, το περιβάλλον ανάπτυξης NetBeans IDE και τον εξυπηρετητή Apache HTTP Server.

## 1.3. Σκοπός και Στόχοι Εργασίας

Σκοπός της πτυχιακής εργασίας είναι η δημιουργία μια ηλεκτρονικής πύλης που θα παρουσιάζει τα δεδομένα τα οποία συλλέγονται από αισθητήρες που διαθέτει το TEI Κρήτης, και θα τα παρουσιάζει με τρόπο κατανοητό και ευανάγνωστο σε χρήστες, σταθερούς ή κινητούς και θα απαντά σε ερωτήματα όπως, «τι καιρό κάνει σήμερα», «πόσο φυσάει σήμερα», «έβρεξε εχθές;» κτλ.

## 1.4. Δομή Εργασίας

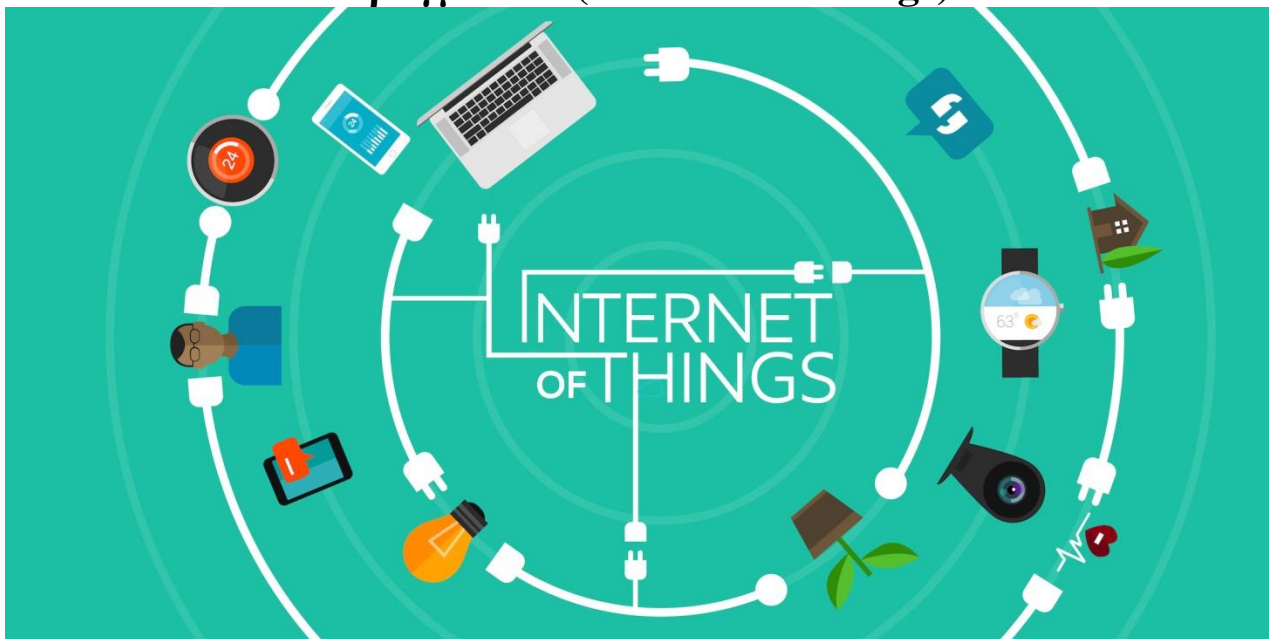
Στο δεύτερο και τρίτο κεφάλαιο παρουσιάζονται το Internet of Things και τα Big Data αντίστοιχα. Περιληπτικά ιστορική ανάδρομη, ορισμός, διάφορα χαρακτηριστικά, τεχνολογίες και τέλος διάφορες εφαρμογές που έχουν στο σήμερα και μέλλον.

Στο τέταρτο κεφάλαιο παρουσιάζονται οι τεχνολογίες οι οποίες χρησιμοποιήθηκαν για την υλοποίηση της πτυχιακής εργασίας. Πιο συγκεκριμένα χρησιμοποιήθηκε το πλαίσιο εφαρμογής Bootstrap, CSS3, η γλώσσα HTML5, το Grafana, η JavaScript, η jQuery, η βάση δεδομένων Influxdb και το περιβάλλον ανάπτυξης NetBeans IDE, Android Studio και τέλος τον Apache Server.

Στο πέμπτο κεφάλαιο υπάρχει μια βασική ανάλυση της ηλεκτρονικής πύλης, state of the art, απαιτήσεις συστήματος και σενάρια χρήσης

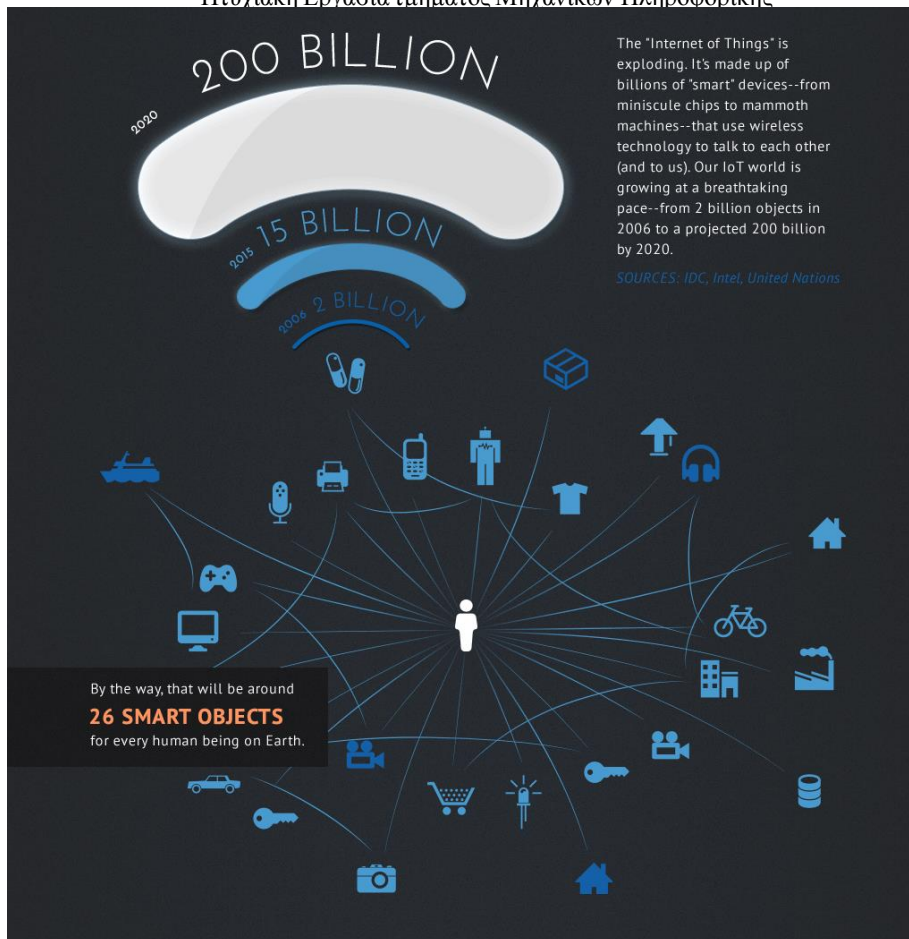
Στο έκτο κεφάλαιο παρουσιάζονται όλες οι φάσεις υλοποίησης της ηλεκτρονικής πύλης και της εφαρμογής, με τέτοιο τρόπο ώστε ο καθένας να μπορεί να την αναπαραγάγει εύκολα. Το έβδομο κεφάλαιο είναι το τελευταίο. Σε αυτό παρουσιάζονται συμπεράσματα και προτείνονται βελτιώσεις και μελλοντικές προεκτάσεις.

## 2. Διαδίκτυο των πραγμάτων (Internet of Things)



Εικόνα 2-1: Internet of Things

Το Διαδίκτυο των πραγμάτων (IoT) είναι η διαδικτύωση των φυσικών συσκευών, οχημάτων, κτιρίων, και άλλων αντικειμένων (που αναφέρονται επίσης ως «έξυπνες συσκευές») με ενσωματωμένα ηλεκτρονικά (λογισμικό, αισθητήρες) συνδεδεμένα στο διαδίκτυο ώστε να γίνεται συλλογή και ανταλλαγή δεδομένων. Το 2013 η Παγκόσμια Αρχή Προτύπων για το διαδίκτυο των πραγμάτων (IoT-Global Standards Initiative) ορίζει το IoT ως «η υποδομή της κοινωνίας της πληροφορίας». Το ιντερνέτ των πραγμάτων επιτρέπει στα αντικείμενα να αναγνωρίζονται ή/και να ελέγχονται εξ' αποστάσεως σε όλη την υπάρχουσα υποδομή του δικτύου, δημιουργώντας ευκαιρίες για πιο άμεση ενσωμάτωση του φυσικού κόσμου σε συστήματα που βασίζονται σε υπολογιστές, με αποτέλεσμα τη βελτίωση της αποτελεσματικότητας, την ακρίβεια και το οικονομικό όφελος. Τυπικά, το IoT αναμένεται να προσφέρει προηγμένη συνδεσιμότητα συσκευών, συστημάτων και υπηρεσιών που πηγαίνει πέρα από την επικοινωνία από συσκευή σε συσκευή (Machine2Machine) και καλύπτει μια πληθώρα πρωτοκόλλων, τομέων και εφαρμογών. Η διασύνδεση αυτών των έξυπνων συσκευών αναμένεται να εγκαινιάσει την αυτοματοποίηση σε όλους σχεδόν τους τομείς, ενώ παράλληλα επιτρέπει την ανάπτυξη προηγμένων εφαρμογών όπως ένα έξυπνο δίκτυο, και την επέκταση σε τομείς όπως οι έξυπνες πόλεις. Το 2011 η CISCO εκτιμούσε ότι το Ιντερνέτ των πραγμάτων θα αποτελείται από σχεδόν 50 δισεκατομμύρια αντικείμενα μέχρι το 2020, ενώ το 2016 η Intel εκτιμάει πως ο αριθμός θα είναι τέσσερις φορές μεγαλύτερος.



Εικόνα 2-2: 200 Billion objects by 2020

## 2.1. Ιστορικά στοιχεία

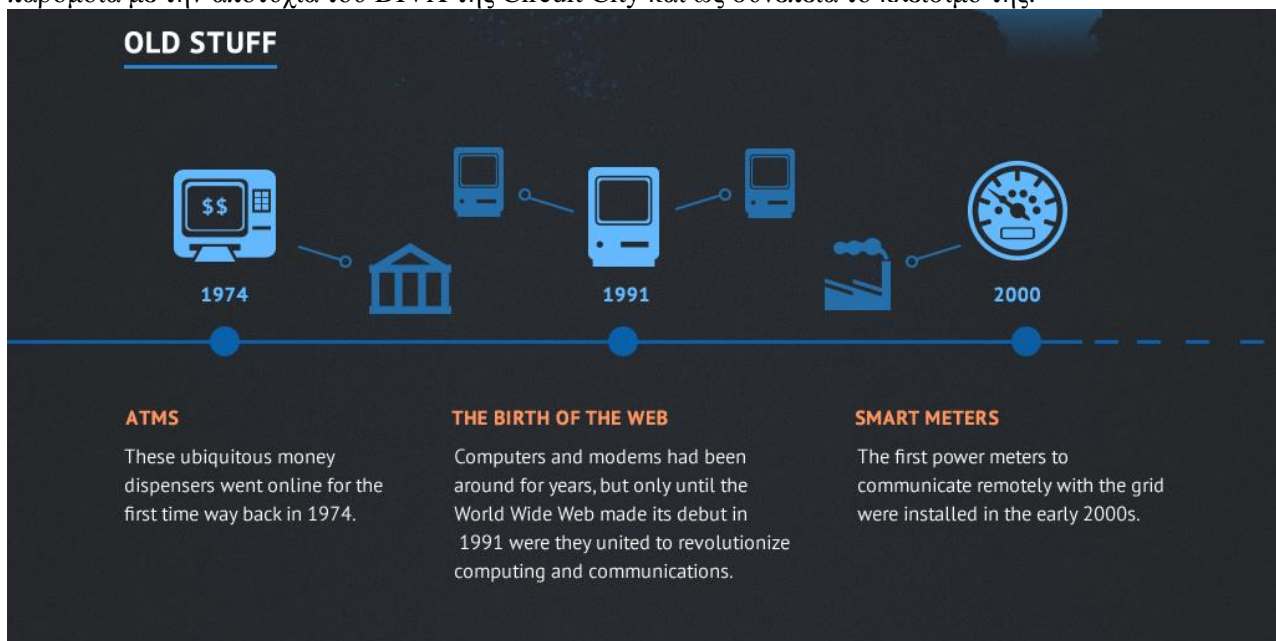
Η έννοια του δικτύου των έξυπνων συσκευών συζητήθηκε ήδη από το 1982, με ένα τροποποιημένο μηχάνημα αυτόματου πωλητή, στο Πανεπιστήμιο Carnegie Mellon, να γίνεται η πρώτη συσκευή με δυνατότητα σύνδεσης στο ιντερνέτ, και να είναι σε θέση να αναφέρει το τι διαθέτει, καθώς και το πόσα αλλά και πια από αυτά τα ποτά είναι κρύα. Το 1991 ο Mark Weiser στο βιβλίο του «Ο υπολογιστής του 21ου αιώνα», περιγράφει την έννοια Ubiquitous computing(ubicomp) δηλαδή τη δυνατότητα να χρησιμοποιήσεις οποιαδήποτε συσκευή, σε οποιαδήποτε θέση και μορφή, και στη ουσία παράγεται το σύγχρονο όραμα του IoT. Οι βασικές τεχνολογίες για την υποστήριξη της «πανταχού υπολογιστικής παρουσίας» περιλαμβάνει το διαδίκτυο, προηγμένο μεσισμικό(middleware), το λειτουργικό σύστημα, τον κινητό κώδικα, αισθητήρες, μικροεπεξεργαστές, νέες συσκευές εισόδου-εξόδου, διεπαφές χρήστη, δίκτυα, κινητά πρωτόκολλα, τη θέση και την τοποθέτηση καθώς και νέα υλικά. Το 1994, ο Reza Raji περιέγραψε την έννοια στο περιοδικό IEEE Spectrum ως «(κινούμενα) μικρά πακέτα δεδομένων σε ένα μεγάλο σύνολο από κόμβους, έτσι ώστε να ενσωματώσουν και να αυτοματοποιήσουν τα πάντα, από οικιακές συσκευές μέχρι ολόκληρα εργοστάσια». Μεταξύ του 1993 και 1996, πολλές εταιρείες πρότειναν λύσεις, όπως η Microsoft το Microsoft at Work ή η Novell το NEST. Ωστόσο, μόνο το 1999 ο τομέας του IoT ξεκινάει να επιταχύνεται. Ο Bill Joy οραματίστηκε την επικοινωνία συσκευής προς συσκευή (Device2Device) ως μέρος του πλαισίου του «Six Webs», που παρουσιάστηκε στο Παγκόσμιο Οικονομικό Φόρουμ στο Νταβός το 1999.

Ο όρος «Διαδίκτυο των Πραγμάτων» επινοήθηκε από τον Peter T.Lewis σε ομιλία του το 1985 στην αμερικανική Ομοσπονδιακή Επιτροπή Επικοινωνιών (Federal Communications Commission). Στην ομιλία του, ο ίδιος αναφέρει ότι «Το Ιντερνέτ των πραγμάτων, ή IoT, είναι η ένταξη των ατόμων, των διαδικασιών και της τεχνολογίας με δυνατότητα σύνδεσης συσκευών και αισθητήρων με σκοπό την ενεργοποίηση της απομακρυσμένης παρακολούθησης, την κατάσταση, το χειρισμό και την αξιολόγηση αυτών των συσκευών».

Η έννοια του ιντερνέτ των πραγμάτων έγινε δημοφιλής το 1999, μέσω του εργαστηρίου Auto-ID στο M.I.T. και τις σχετικές δημοσιεύσεις τους για την ανάλυση της αγοράς. Η ταυτοποίηση μέσω ραδιοσυχνότητων (Radio-frequency identification-RFID) θεωρήθηκε από τον Kevin Ashton (ένας από τους

Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής  
ιδρυτές του Auto-ID) ως προϋπόθεση για το διαδίκτυο των πραγμάτων, εκείνη τη χρονική στιγμή. Ο Ashton προτιμά τη φράση «Το διαδίκτυο για τα πράγματα». Αν όλα τα αντικείμενα και οι άνθρωποι στην καθημερινή ζωή ήταν εξοπλισμένοι με αναγνωριστικά, οι υπολογιστές θα μπορούσαν τους καταγράψουν και να τους διαχειριστούν. Εκτός από τη χρήση RFID, η σήμανση των πραγμάτων μπορεί να επιτευχθεί μέσω τέτοιων τεχνολογιών όπως η επικοινωνία κοντινού πεδίου(Bluetooth) , barcodes , κωδικών QR(Quick Response Code) και ψηφιακή υδατογράφιση.

Στην αρχική ερμηνεία της, μία από τις πρώτες συνέπειες της εφαρμογής του ιντερνέτ των πραγμάτων, με τον εξοπλισμό όλων των αντικείμενων στον κόσμο, με μικροσκοπικές συσκευές εντοπισμού και μοναδικών αναγνωριστικών θα ήταν να αλλάξει ριζικά την καθημερινή ζωή. Για παράδειγμα, άμεσος και αδιάκοπος έλεγχος για τον κάθε άνθρωπο και τις επιλογές του καθώς και την καταγραφή αυτών των επιλογών. Η ικανότητα ενός ατόμου να αλληλεπιδρά με αντικείμενα, που θα μπορούσαν να τροποποιηθούν εξ' αποστάσεως με βάση τις άμεσες ή παρούσες ανάγκες του, σύμφωνα με τις υπάρχουσες συμφωνίες του τελικού χρήστη. Για παράδειγμα, μια τέτοια τεχνολογία θα μπορούσε να χορηγήσει στους εκδότες οπτικοακουστικού υλικού πολύ περισσότερο έλεγχο πάνω στον τελικό χρήστη και τις συσκευές αναπαραγωγής του με την επιβολή περιορισμών πνευματικών δικαιωμάτων και διαχείρισης ψηφιακών δικαιωμάτων. Δηλαδή η ικανότητα ενός πελάτη που αγόρασε ένα δίσκο Blu-ray, για να παρακολουθήσει την ταινία, θα μπορούσε να εξαρτηθεί από την απόφαση του κατόχου των πνευματικών δικαιωμάτων, παρόμοια με την αποτυχία του DIVX της Circuit City και ως συνέπεια το κλείσιμο της.



Εικόνα 2-3: Evolution

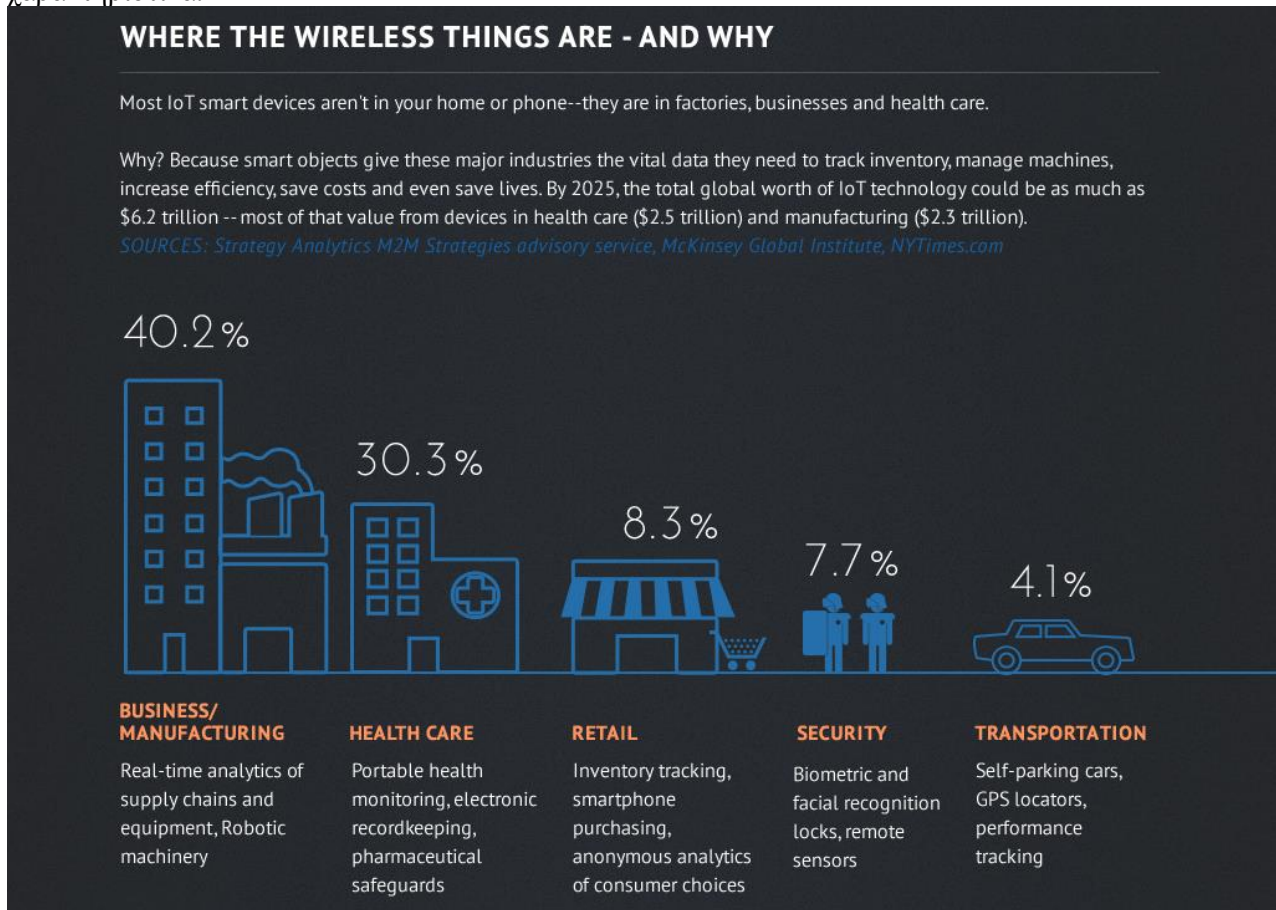
## 2.2. Εφαρμογές

Σύμφωνα με μια έρευνα του 2014 και τη μελέτη που έγινε από το περιοδικό Pew Research Internet Project, μια μεγάλη πλειοψηφία των εμπειρογνομόνων της τεχνολογίας και αφοσιωμένους χρήστες του διαδικτύου οι οποίοι ανταποκρίθηκαν με 83 τοις εκατό, συμφώνησαν με την ιδέα ότι το διαδίκτυο των πραγμάτων, θα έχει ευρεία αποδοχή και επωφεληή αποτελέσματα μέχρι το 2025. Ως εκ τούτου, είναι σαφές ότι το IoT θα αποτελείται από ένα πολύ μεγάλο αριθμό συσκευών να συνδέεται με το ιντερνέτ. Σε μια δραστηκή κίνηση για να φιλοξενήσει νέες και αναδυόμενες τεχνολογικές καινοτομίες, η Κυβέρνηση του Ηνωμένου Βασιλείου, στον προϋπολογισμό της το 2015, διέθεσε £ 40.000.000 για έρευνα γύρο από το διαδίκτυο των πραγμάτων. Ο πρώην Βρετανός υπουργός οικονομικών George Osborne, έθεσε ευθέως ότι το διαδίκτυο των πραγμάτων είναι το επόμενο στάδιο της επανάστασης της πληροφορίας και ανέφερε τη διασύνδεση των πάντων, από τις αστικές μεταφορές, σε ιατρικές συσκευές και οικιακές συσκευές.

Η ενοποίηση με το διαδίκτυο σημαίνει ότι οι συσκευές θα χρησιμοποιούν μια διεύθυνση IP ως μοναδικό αναγνωριστικό. Ωστόσο, λόγω του περιορισμένου χώρου διευθύνσεων του IPv4 (η οποία επιτρέπει 4.300.000.000 μοναδικές διευθύνσεις), τα αντικείμενα στο IoT θα πρέπει να χρησιμοποιούν το IPv6 για να φιλοξενηθεί ο εξαιρετικά μεγάλος αριθμός, που απαιτείται. Τα αντικείμενα του IoT δεν θα είναι μόνο οι συσκευές με ικανότητα αίσθησης όπως η θερμοκρασία, η υγρασία κτλ, αλλά και να παρέχουν δυνατότητες απομακρυσμένης ενεργοποίησης π.χ. κλειδαριές, καφετιέρες ψυγεία κτλ. Σε μεγάλο βαθμό, το μέλλον του διαδικτύου των πραγμάτων δεν θα είναι δυνατό χωρίς την υποστήριξη του IPv6, και κατά συνέπεια, η

Πτυχιική Εργασία τμήματος Μηχανικών Πληροφορικής  
παγκόσμια υιοθέτηση του IPv6 στα επόμενα χρόνια, που θα είναι κρίσιμα για την επιτυχή ανάπτυξη του  
Ιντερνέτ των πραγμάτων στο μέλλον.

Η δυνατότητα να δικτυωθούν συσκευές με περιορισμένη επεξεργαστική ισχύ (CPU), μνήμη και τη συνολική διάθεση πόρων σημαίνει ότι IoT βρίσκει εφαρμογές σε σχεδόν κάθε τομέα. Τα συστήματα αυτά θα μπορούσαν να είναι υπεύθυνα για τη συλλογή πληροφοριών σε πληθώρα εφαρμογών από φυσικά οικοσυστήματα, έως κτίρια και εργοστάσια, βρίσκοντας έτσι εφαρμογές στους τομείς της περιβαλλοντικής ανίχνευσης και πολεοδομικού σχεδιασμού. Από την άλλη πλευρά, τα συστήματα του IoT θα μπορούσαν επίσης να είναι υπεύθυνα για την εκτέλεση δράσεων, όχι μόνο για περισυλλογή δεδομένων. Για παράδειγμα ευφυή συστήματα για ψώνια θα μπορούσαν να παρακολουθούν τις αγοραστικές συνήθειες συγκεκριμένων χρηστών σε ένα κατάστημα με την παρακολούθηση των κινητών τηλεφώνων τους. Αυτοί οι χρήστες θα μπορούσαν στη συνέχεια να λάβουν ειδικές προσφορές για τα αγαπημένα τους προϊόντα, ή ακόμα και την τοποθεσία των αντικειμένων μέσα στο κατάστημα που έχουν ανάγκη, όπου το ψηγείο τους, θα τους έχει στείλει αυτόματα στο τηλέφωνο. Επιπρόσθετα παραδείγματα ανίχνευσης και ενεργοποίησης έχουμε σε εφαρμογές που ασχολούνται με τη θερμότητα, ηλεκτρική ενέργεια, τη διαχείριση ενέργειας, καθώς και συστήματα πλοήγησης. Άλλες εφαρμογές που το διαδίκτυο των πραγμάτων μπορεί να παρέχει, είναι εκτεταμένα χαρακτηριστικά ασφαλείας για το σπίτι και συνολικά στον οικιακό αυτοματισμό. Η έννοια της δημιουργίας ενός «Ιντερνέτ των πραγμάτων διαβίωσης» έχει προταθεί για να περιγράψει τα δίκτυα με βιολογικούς αισθητήρες που θα μπορούσαν να χρησιμοποιήσουν το σύννεφο (cloud) για αναλύσεις έτσι ώστε οι χρήστες να μελετήσουν το DNA ή άλλα μόρια. Ωστόσο, η εφαρμογή του IoT δεν περιορίζεται μόνο σε αυτές τις περιοχές. Μπορεί επίσης να υπάρχουν άλλες περιπτώσεις για εξειδικευμένη χρήση του IoT. Με βάση το πεδίο εφαρμογής, τα προϊόντα IoT μπορούν γενικά να ταξινομηθούν σε πέντε διαφορετικές κατηγορίες: έξυπνη φορητή συσκευή, έξυπνο σπίτι, έξυπνη πόλη, έξυπνο περιβάλλον και έξυπνη επιχείρηση. Τα προϊόντα του IoT και οι λύσεις σε κάθε μία από αυτές τις αγορές έχουν διαφορετικά χαρακτηριστικά.



Εικόνα 2-4: Applications

## 2.3. Λοιπά χαρακτηριστικά

### 2.3.1. Νοημοσύνη

**AND STUFF YET TO COME**

**MAN-MACHINE MIND MELD**

IoT devices are now able to collect data on the human body, but new research is testing our ability to control machines--and even each other--with our minds. This new wireless frontier could yield life-changing, and life-saving advances.

**WEIGH IN ON MIND CONTROL**

*How far off are we from controlling a computer with our minds as fast as we can control it with our hands?*

**WE, ROBOTS**

Someday soon, connected robots will have the ability to learn from each other and work in teams to increase efficiency and solve scientific problems.

**WEIGH IN ON ROBOTS**

*Do you own a personal robot, like the housecleaning Roomba? If not, why not?*

The infographic features a dark blue background with white and light blue text. It includes icons of a robotic arm, a human head with neural connections, and three stylized robot figures in business suits. The text is arranged in a structured layout with clear section headers and sub-headers.

Εικόνα 2-5: Intelligence

Οι συσκευές θα διαθέτουν περιβάλλουσα νοημοσύνη και αυτόνομο έλεγχο ώστε να λειτουργούν από κοινού για να υποστηρίζουν τους ανθρώπους στην εκτέλεση καθημερινών δραστηριοτήτων, τις δουλειές τους με έναν εύκολο, φυσικό τρόπο με τη χρήση πληροφοριών και στοιχείων που είναι κρυμμένα στο δίκτυο που συνδέει τις συσκευές αυτές. Οι δυο αυτές έννοιες, οι οποίες δεν απαιτούν κάποια ξεχωριστή υποδομή δεν αποτελούν μέρος της αρχικής ιδέας του διαδικτύου των πραγμάτων. Ωστόσο, υπάρχει μια τάση στον τομέα της έρευνας για την ενσωμάτωση των εννοιών του διαδικτύου των πραγμάτων και του αυτόνομου ελέγχου, με κατεύθυνση ότι τα αντικείμενα είναι η κινητήρια δύναμη για εάν αυτόνομο IoT.

### 2.3.2. Αρχιτεκτονική

Το σύστημα πιθανώς θα στηρίζεται σε μια αρχιτεκτονική με βάση τα συμβάντα και το πλαίσιο των διαδικασιών και λειτουργιών, σε πραγματικό χρόνο.

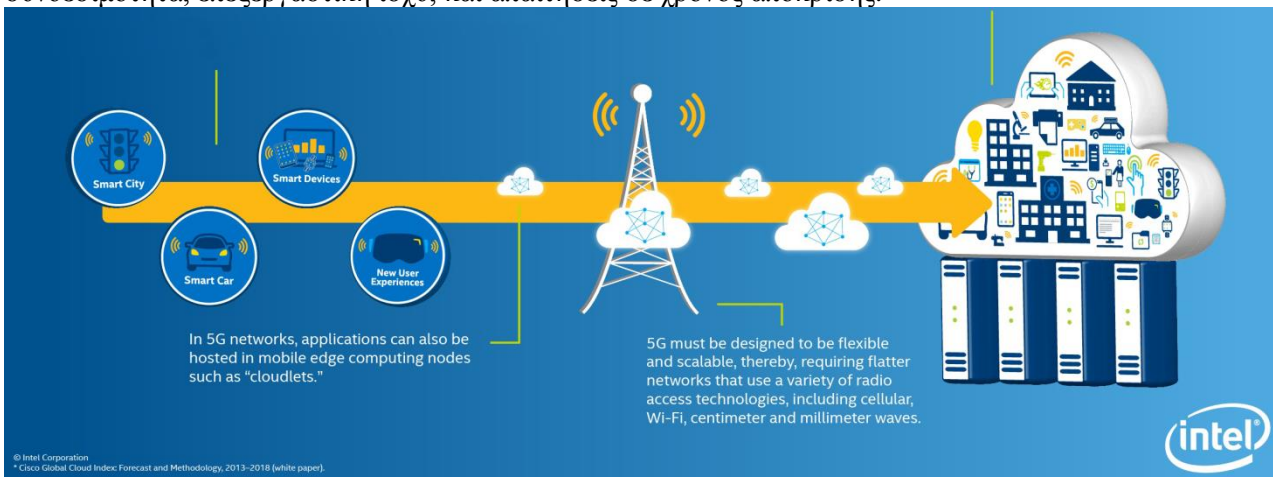
### 2.3.3. Αρχιτεκτονική του δικτύου

Το Διαδίκτυο των πραγμάτων απαιτεί τεράστια επεκτασιμότητα στο χώρο του δικτύου για να χειριστεί το εύρος των συσκευών. Το 6LoWPAN ( προφέρεται slow pan, είναι ένα αρκτικόλεξο του IPv6 πάνω από χαμηλής ισχύος ασύρματα δίκτυα) της IETF (Internet Engineering Task Force) θα μπορούσε να χρησιμοποιηθεί για τη σύνδεση συσκευών σε IP δίκτυα. Με τα δισεκατομμύρια των συσκευών να

Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής  
προστίθεται στο χώρο του Internet, το IPv6 θα διαδραματίσει σημαντικό ρόλο στο χειρισμό της κλιμάκωσης επίπεδο δικτύου. Παράλληλα αναπτύσσεται και η έννοια της «υπολογιστικής ομίχλης». Είναι ένα μοντέλο στο οποίο τα δεδομένα, η επεξεργασία και οι εφαρμογές συγκεντρώνονται σε συσκευές στο άκρο του δικτύου από το είναι εξ' ολοκλήρου στο σύννεφο. Μπορεί να υπάρχουν τεράστιες ποσότητες δεδομένων που προέρχονται από αυτές τις συσκευές. Για παράδειγμα, μια μηχανή τζετ μπορεί να παράγει 10Terabyte των δεδομένων σχετικά με την απόδοση και την κατάσταση της σε μόλις 30 λεπτά, σύμφωνα με τη Cisco. Είναι συχνά χάσιμο χρόνου και εύρος ζώνης να μεταφερθούν όλα τα δεδομένα από τις συσκευές του IoT σε ένα σύννεφο και στη συνέχεια οι απαντήσεις να μεταδοθούν πίσω. Αυτό σημαίνει, ότι τα δεδομένα μπορούν να επεξεργαστούν τοπικά σε έξυπνες συσκευές, αντί να αποστέλλονται στο σύννεφο για επεξεργασία. Η «ομίχλη υπολογιστών» είναι μια προσέγγιση για την αντιμετώπιση των απαιτήσεων του συνεχώς αυξανόμενου αριθμού των ιντερνέτ των πραγμάτων.

Με το λανσάρισμα του Bluetooth 5, η τεχνολογία Bluetooth συνεχίζει να εξελίσσεται για να καλύψει τις ανάγκες της βιομηχανίας, ως το παγκόσμιο ασύρματο πρότυπο για απλή, ασφαλή συνδεσιμότητα. Με τετραπλάσιο εύρος, διπλάσια ταχύτητα και οκταπλάσια ικανότητα για μετάδοση του μηνύματος από τον προκάτοχο του, οι βελτιώσεις του Bluetooth 5 εμφανίζουν αύξηση της λειτουργικότητας, του Bluetooth για το IoT. Αυτά τα χαρακτηριστικά, σε συνδυασμό με τη βελτίωση της διαλειτουργικότητας και της συνύπαρξης με άλλες ασύρματες τεχνολογίες, συνεχίζουν να βελτιώνουν την εμπειρία του IoT, επιτρέποντας απλές και εύκολες αλληλεπιδράσεις σε όλη την τεράστια γκάμα των συνδεδεμένων συσκευών.

Το 5G δίκτυο τηλεπικοινωνιών δεν θα είναι απλά για την αύξηση της ταχύτητας και της χωρητικότητας, αλλά θα είναι επίσης για την ευφυΐα σε όλο το δίκτυο ώστε να επιτρέπει στις συσκευές και το δίκτυο να επικοινωνούν πιο αποτελεσματικά, να μεταφέρουν δεδομένα και περιεχόμενο πιο γρήγορα, και τέλος να μοιράζονται υπολογιστικούς πόρους. Οι συσκευές και το δίκτυο θα πρέπει επίσης να συνεργάζονται μεταξύ τους, για τη δημιουργία ευφυών υπηρεσιών για την επίγνωση ή αντίληψη μιας κατάστασης ή γεγονότος στο δίκτυο. Θα πρέπει να δοθεί προτεραιότητα σε δράσεις, με βάση το επίπεδο της σημασίας, να είναι αρκετά ευέλικτο για να φιλοξενήσει πολλές συσκευές ταυτόχρονα, οι οποίες θα έχουν πολύ διαφορετική συνδεσιμότητα, επεξεργαστική ισχύ, και απαιτήσεις σε χρόνος απόκρισης.



Εικόνα 2-6: 5G

### 2.3.4. Πλαίσιο (Framework)

Το πλαίσιο μπορεί να συμβάλει στη στήριξη της αλληλεπίδρασης μεταξύ των «πραγμάτων» και να επιτρέψει την δημιουργία πολύπλοκων δομών, όπως καταναμημένα υπολογιστικά συστήματα και την ανάπτυξη των καταναμημένων εφαρμογών. Επί του παρόντος, ορισμένα πλαίσια φαίνεται να επικεντρώνονται σε καταγραφή δεδομένων πραγματικού χρόνου, προσφέροντας κάποια βάση για τα «πράγματα» ώστε συνεργάζονται και να αλληλεπιδρούν μεταξύ τους. Οι νεότερες πλατφόρμες που αναπτύσσονται, προσθέτουν περισσότερη ευφυΐα στο IoT, η IBM ανακοίνωσε ότι συνδυάζει τη νοημοσύνη και τη μάθηση, με μοντέλα συγκεκριμένα για τη βιομηχανία και ακόμη για επεξεργασία της φυσικής γλώσσας. Το ίδρυμα προτύπων XMPP Standards Foundation (XSF) δημιουργεί ένα τέτοιο πλαίσιο σε ένα πλήρως ανοικτό πρότυπο που δεν εξαρτάται από οποιαδήποτε εταιρεία, ούτε με οποιαδήποτε υπηρεσία σύννεφου (cloud). Η πρωτοβουλία ονομάζεται Chatty Things. Το XMPP παρέχει το σύνολο των απαιτούμενων δομικών εκείνων στοιχείων και μία αποδεδειγμένα καταναμημένη λύση που μπορεί να κλιμακωθεί με υψηλά επίπεδα ασφάλειας.

Το REST (Representational state transfer) είναι μια επεκτάσιμη αρχιτεκτονική που επιτρέπει στα



Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής  
πράγματα να επικοινωνούν μέσω Hypertext Transfer Protocol. Είναι μία εύκολα υιοθετήσιμη λύση για εφαρμογές IoT ώστε παρέχεται επικοινωνία από μια συσκευή σε έναν κεντρικό εξυπηρετητή.

Το MQTT (MQ Telemetry Transport) είναι μία αρχιτεκτονική πάνω από το TCP/IP που επιτρέπει την αμφίδρομη επικοινωνία ανάμεσα σε μία συσκευή και ένα μεσίτη MQTT.

Από την άλλη η Nokia και η Intel συνεργάζονται μεταξύ τους, ώστε να καθορίσουν τις προδιαγραφές του συστήματος 5G, τις απαιτήσεις λογισμικού, και να υλοποιήσουν τον απαραίτητο εξοπλισμό για δοκιμές και επιδείξεις του σχεδίου τους. Ως μέρος αυτής της προσπάθειας, η λύση της Nokia για κέντρα δεδομένων ονόματι AirFrame, θα χρησιμοποιεί επεξεργαστές της Intel με υψηλή απόδοση, ως βάση για μια πιο ευέλικτη και κατανομημένη αρχιτεκτονική σύννεφου(cloud), για να καθιστάει ικανή τη τεχνολογία 5G. Επιπλέον, με βάση τις υφιστάμενες συνεργασίες στον τομέα της τεχνολογίας ασύρματης πρόσβασης, οι δύο εταιρείες σκοπεύουν να αναπτύξουν ένα πλαίσιο από κοινού, για ένα διαλειτουργικό περιβάλλον για τις τηλεπικοινωνίες, ώστε να αποδείξουν τις επιδόσεις 5G, που αποσκοπούν στις περιπτώσεις ευρυζωνικών κινητών χρηστών.

### 2.3.5. Ασφάλεια



Εικόνα 2-7: Safety

Ανησυχίες έχουν διατυπωθεί ότι το διαδίκτυο των πραγμάτων αναπτύσσεται ταχύτατα χωρίς την ανάλογη εξέταση των προκλήσεων ασφαλείας που εμπλέκονται, καθώς και τις αναγκαίες ρυθμιστικές αλλαγές που μπορεί να χρειάζονται. Σύμφωνα με την έρευνα του Business Insider Intelligence που διεξήχθη κατά το τελευταίο τρίμηνο του 2014, το 39% των ερωτηθέντων είπε ότι η ασφάλεια είναι η μεγαλύτερη ανησυχία στην υιοθέτηση της τεχνολογίας για το διαδίκτυο των πραγμάτων. Συγκεκριμένα, καθώς το διαδίκτυο των πραγμάτων εξαπλώνεται ευρέως, επιθέσεις στον κυβερνοχώρο είναι πιθανό να γίνονται όλο ένα και πιο πραγματικές (και όχι απλώς εικονικές) απειλές. Σε άρθρο του, για την ασφάλεια στον κυβερνοχώρο το Γενάρη του 2014 στο Forbes, ο αρθρογράφος Joseph Steinberg απαριθμεί πολλές συσκευές συνδεδεμένες στο διαδίκτυο που μπορούν ήδη να «κατασκοπεύουν τους ανθρώπους στα σπίτια τους», συμπεριλαμβανομένων τηλεοράσεων, λευκών συσκευές, φωτογραφικών μηχανών και θερμοστάτες. Οι συσκευές που ελέγχονται από υπολογιστή σε αυτοκίνητα, όπως τα φρένα, ο κινητήρας, οι κλειδαριές, καπό, ασφάλειες για ρυμουλκούμενα φορτία, κόρνα, αισθητήρα θερμότητας, και το ταμπλό έχουν αποδειχθεί ότι είναι ευάλωτες, σε εισβολείς που έχουν πρόσβαση στο δίκτυο του οχήματος. Σε ορισμένες περιπτώσεις μάλιστα, όπου τα συστήματα ηλεκτρονικών υπολογιστών στο όχημα που είναι συνδεδεμένα στο ιντερνέτ, μπορούν να εκτεθούν σε απομακρυσμένες επιθέσεις. Μέχρι το 2008 οι ερευνητές ασφαλείας είχαν δείξει τη δυνατότητα του απομακρυσμένου ελέγχου βηματοδοτών, χωρίς οι χρήστες να το γνωρίζουν. Επίσης χάκερ επιδεικνύουν τη δυνατότητα έλεγχου εξ' αποστάσεως αντλιών ινσουλίνης και εμφυτεύσιμων απινιδωτών. Ο David Rogue (Αμερικανός συγγραφέας τεχνολογίας, επιστήμης και τηλεοπτικός

Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής  
παρουσιαστής) έγραψε στο τεύχος Νοεμβρίου του 2015 του Scientific American, ότι ορισμένες από τις πρόσφατες δημοσιευμένες εκθέσεις για τους χάκερ με την ικανότητα απομακρυσμένου τηλεχειρισμού ορισμένων λειτουργιών των αυτοκινήτων, ότι δεν ήταν τόσο σοβαρές όσο θα μπορούσε κανείς να σκεφτεί, λόγω των διάφορων ελαφρυντικών περιστάσεων: όπως ένα σφάλμα(bug) που επέτρεψε την εισβολή, το οποίο έχει διορθωθεί πριν από τη δημοσίευση της έκθεσης, ή ότι οι ερευνητές ασφαλείας έπρεπε να έχουν φυσική πρόσβαση στο αυτοκίνητο προηγουμένως, και να έχουν προετοιμαστεί για αυτό.

Το Εθνικό Συμβούλιο Πληροφοριών των ΗΠΑ σε μια έκθεση του, υποστηρίζει ότι θα ήταν δύσκολο να αρνηθεί «την πρόσβαση σε δίκτυα αισθητήρων και τον απομακρυσμένο έλεγχο αντικειμένων που ανήκουν σε εχθρούς των Ηνωμένων Πολιτειών, εγκληματίες, και γενικά σε τρομοκρατικούς φορείς. Μια ανοικτή αγορά για συγκεντρωτικά δεδομένα αισθητήρων θα μπορούσε να εξυπηρετήσει τα συμφέροντα του εμπορίου και της ασφάλειας, όχι σε λιγότερο βαθμό από ότι βοηθά τους εγκληματίες και τους κατασκόπους στον εντοπισμό ευάλωτων στόχων. Με αυτόν τον τρόπο, η μαζική παράλληλη αύξηση αισθητήρων, μπορεί να υπονομεύσει την κοινωνική συνοχή, εάν αποδειχθεί ότι είναι θεμελιωδώς ασυμβίβαστη με τέταρτη τροπολογία, η όποια αποτελεί εγγύηση κατά τις αλόγιστης παρακολούθησης». Σε γενικές γραμμές, οι υπηρεσίες πληροφοριών βλέπουν το διαδίκτυο των πραγμάτων, ως μία πλούσια πηγή δεδομένων.

Ως απάντηση στις αυξανόμενες ανησυχίες σχετικά με την ασφάλεια, ιδρύθηκε στις 23 Σεπτεμβρίου του 2015 το Ίδρυμα Ασφάλειας για το Ιντερνέτ των Πραγμάτων (IoT Security Foundation). Το IoTSF έχει μια αποστολή, να εξασφαλίσει το διαδίκτυο των πραγμάτων, με την προώθηση των γνώσεων και βέλτιστων πρακτικών. Το ιδρυτικό διοικητικό της συμβούλιο αποτελείται από παρόχους τεχνολογίας και τηλεπικοινωνιακών εταιρειών, συμπεριλαμβανομένων της BT, Vodafone, Intel, Imagination Technologies και Pen Test Partners.

Το 2016, μια κατανεμημένη επίθεση άρνησης παροχής υπηρεσιών που δημιουργήθηκε από συσκευές IoT οι οποίες «έτρεχαν» το Mirai ένα κακόβουλο λογισμικό, έθεσαν εκτός λειτουργίας έναν πάροχο DNS και αρκετές μεγάλες ιστοσελίδες.

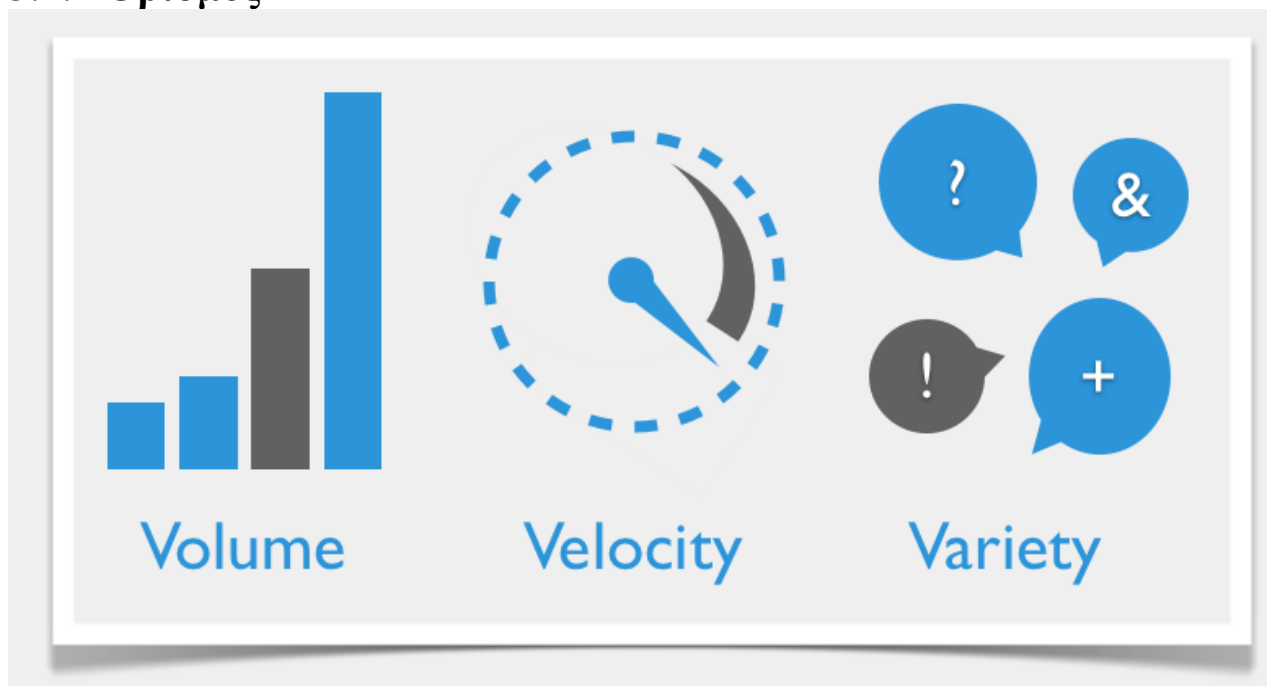
### 3. BIG DATA



Εικόνα 3-1: Big Data

Ο όρος Big Data αναφέρεται σε σύνολα δεδομένων που είναι τόσο μεγάλα ή σύνθετα που οι συνήθεις εφαρμογές για επεξεργασία δεδομένων είναι ανεπαρκείς για την αντιμετώπισή τους. Οι προκλήσεις περιλαμβάνουν την ανάλυση, τη σύλληψη, επιμέλεια δεδομένων, αναζήτηση, κοινή χρήση, αποθήκευση, μεταφορά, απεικόνιση, ερωτήματα, την ενημέρωση και την προστασία πληροφοριών. Ο όρος «μεγάλα δεδομένα» αναφέρεται συχνά σε χρήσεις όπως ανάλυση προγνωστικών, ανάλυση της συμπεριφοράς των χρηστών, ή σε ορισμένες άλλες προηγμένες μεθόδους αναλύσεων δεδομένων που εξάγουν πληροφορίες από τα δεδομένα, και σπανίως σε ένα συγκεκριμένο μέρος του συνόλου δεδομένων. Σε έρευνά τους το Σεπτέμβριο του 2011, η Kate Crawford του πανεπιστημίου New South Wales και η Danah Boyd ερευνήτρια για τη Microsoft, με τίτλο έξι μεγάλες προκλήσεις για τα Big Data (Six Provocations for Big Data) αναφέρουν: «Δεν υπάρχει αμφιβολία ότι οι ποσότητες των δεδομένων που είναι τώρα διαθέσιμες είναι πράγματι μεγάλες, αλλά αυτό δεν είναι το σημαντικότερο χαρακτηριστικό του νέου αυτού οικοσυστήματος των δεδομένων».

### 3.1. Ορισμός



Εικόνα 3-2: 3V

Ο όρος ήταν σε χρήση από τη δεκαετία του 1990, με μερικούς να δίνουν τα εύσημα στον John Mashey για την επινόηση ή τουλάχιστον ότι τον κατέστησε δημοφιλή. Ο όρος Big Data περιλαμβάνει συνήθως σύνολα δεδομένων με μεγέθη πέρα από την ικανότητα των συνήθων εργαλείων λογισμικού που χρησιμοποιούνται για τη σύλληψη, οργάνωση, διαχείριση και επεξεργασία των δεδομένων μέσα σε ανεκτό χρόνο. Το τι εννοούμε ως «μέγεθος» των Big Data αλλάζει συνεχώς, από το 2012 κυμαίνεται από μερικά δεκάδες terabytes σε πολλά petabytes δεδομένων. Τα Big Data απαιτούν μια σειρά από τεχνικές και τεχνολογίες με νέες μορφές ολοκλήρωσης για να αποκαλύψουν στοιχεία που προέκυψαν από σύνολα δεδομένων που είναι ποικίλα, σύνθετα, και σε τεράστια κλίμακα.

Σε έκθεση του το 2001 και σχετικών διαλέξεων, ο Doug Laney, αναλυτής της τότε ομάδας META Group (τώρα Gartner) όρισε τις προκλήσεις της ανάπτυξης των δεδομένων και των ευκαιριών σε τρεις διαστάσεις, δηλαδή, την αύξηση του όγκου (ποσότητα των δεδομένων-Volume), ταχύτητα (ταχύτητα των δεδομένων μέσα και έξω-Velocity), και ποικιλία (γκάμα των τύπων των δεδομένων και των πηγών-Variety). Η Gartner, και ένα μεγάλο μέρος της βιομηχανίας, συνεχίζουν να χρησιμοποιούν αυτό το μοντέλο των «3Vs» για την περιγραφή των «μεγάλων δεδομένων». Το 2012, η Gartner αναβάθμισε τον ορισμό της ως εξής: «Τα Big Data είναι μεγάλα σε όγκο, σε ταχύτητα, και / ή σε πολλά ποικίλα πληροφοριακά στοιχεία που απαιτούν νέες μορφές επεξεργασίας για να καταστεί δυνατή η ενισχυμένη λήψη αποφάσεων, η διορατική ανακάλυψη και η βελτιστοποίηση διαδικασιών». Ο ορισμός της Gartner για τα «3V» εξακολουθεί να χρησιμοποιείται ευρέως, και σε συνύπαρξη με ένα συναινετικό ορισμό που αναφέρει ότι «τα Big Data αντιπροσωπεύουν εκείνα τα πληροφοριακά στοιχεία που χαρακτηρίζονται από πολύ υψηλό όγκο(High Volume), ταχύτητα(Velocity) και ποικιλία(Variety) που απαιτούν ειδική τεχνολογία και αναλυτικές μεθόδους για τη μετατροπή τους σε αξία(Value)». Ο νέος αυτός ορισμός προσθέτει ένα επιπλέον «V-Veracity- πιστότητα/ακρίβεια-», όπου ορισμένες οργανώσεις έχουν αποδεχτεί, ωστόσο η αναθεώρηση αυτή αμφισβητήθηκε από ορισμένους, υποστηρίζοντας πως τα «3V» είναι αρκετά για την περιγραφή τους.

Τα «3V» έχουν επεκταθεί και σε άλλα συμπληρωματικά χαρακτηριστικά των Big Data:

- **Όγκος:** τα Big Data δεν δειγματοληπτούν, απλά παρατηρούν και παρακολουθούν τι συμβαίνει.
- **Ταχύτητα:** τα Big Data είναι συχνά διαθέσιμα σε πραγματικό χρόνο
- **Ποικιλία:** τα Big Data αποθηκεύουν από κείμενο, εικόνες, ήχο, βίντεο, συν του ότι συμπληρώνουν κομμάτια που λείπουν μέσω της «συγχώνευσης δεδομένων (data fusion)».
- **Μηχανική μάθηση:** τα Big Data συχνά δεν «κάνουν» ερωτήσεις και απλά εντοπίζουν μοτίβα.
- **Ψηφιακό αποτύπωμα:** τα Big Data είναι συχνά ένα υποπροϊόν της ψηφιακής αλληλεπίδρασης χωρίς κάποιο κόστος.

Η αυξανόμενη ωριμότητα της έννοιας σκιαγραφεί τη διαφορά μεταξύ των «μεγάλων δεδομένων» και της «επιχειρησιακής νοημοσύνης», δηλαδή:

- Η επιχειρησιακή νοημοσύνη χρησιμοποιεί δεδομένα με υψηλή πυκνότητα πληροφοριών για τη δημιουργία περιγραφικών στατιστικών στοιχείων που σκοπό έχουν τη μέτρηση διαφόρων πραγμάτων, την ανίχνευση τάσεων, κλπ.
- Τα Big Data χρησιμοποιούν επαγωγικά στατιστικά στοιχεία και έννοιες από την αναγνώριση μη γραμμικών συστημάτων για τη δημιουργία συμπερασματικών νόμων (παλινδρομήσεις, μη γραμμικές σχέσεις, και αιτιώδους συνάφειας), από μεγάλα σύνολα δεδομένων με χαμηλή πυκνότητα των πληροφοριών, για να αποκαλύψουν σχέσεις και εξαρτήσεις, ή να κάνουν προβλέψεις εκβάσεων και συμπεριφορών.

## 3.2. Χαρακτηριστικά

Τα Big Data μπορούν να περιγραφούν με τα ακόλουθα χαρακτηριστικά:

1) **Όγκος(Volume)**: Η ποσότητα των παραγόμενων και αποθηκευμένων δεδομένων. Το μέγεθος των δεδομένων καθορίζει την αξία, τις πιθανές πληροφορίες και αν μπορούν πράγματι να θεωρηθούν ως Big Data ή όχι.

2) **Ποικιλία(Variety)**: Ο τύπος και η φύση των δεδομένων, βοηθά τους ανθρώπους που τα αναλύουν, να χρησιμοποιούν αποτελεσματικά τις προκύπτουσες πληροφορίες.

3) **Ταχύτητα(Velocity)**: Εδώ εννοείται ως η ταχύτητα, με την οποία τα δεδομένα παράγονται και επεξεργάζονται ώστε να ικανοποιήσουν τις απαιτήσεις και τις προκλήσεις που βρίσκονται στην αύξηση του όγκου και της εξέλιξης τους.

4) **Μεταβλητότητα(Variability)**: Η ασυνέπεια του συνόλου δεδομένων, μπορεί να παρεμποδίσει τις διαδικασίες για το χειρισμό και της διαχείρισή τους.

5) **Πιστότητα(Veracity)**: Η ποιότητα των δεδομένων που έχουν καταγραφεί μπορεί να ποικίλλει σε μεγάλο βαθμό, με αποτέλεσμα να επηρεάζουν την ακριβή ανάλυση.

## 3.3. Αρχιτεκτονική



Εικόνα 3-3: Architecture

Το 2000, η Seisint Inc. (τώρα ως όμιλος LexisNexis) ανέπτυξε ένα πλαίσιο διαμοιρασμού δεδομένων με βάση την C++ για την αποθήκευση και τη δημιουργία ερωτημάτων. Το σύστημα αποθηκεύει και διανέμει δομημένα, ημι-δομημένα και αδόμητα δεδομένα σε πολλαπλούς εξυπηρετητές. Οι χρήστες μπορούν να χτίσουν ερωτήματα σε C++ διάλεκτο που ονομάζεται Extensible Computer Language. Η ECL χρησιμοποιεί μια νέα μέθοδο που ονομάζεται «εφαρμογή σχήματος κατά την ανάγνωση-schema on read» για να συμπεράνει τη δομή των αποθηκευμένων δεδομένων, όταν ερωτάται, αντί του συνηθισμένου τρόπου που χρησιμοποιείται κατά κόρον δηλαδή πρώτα να δημιουργήσουμε ένα σχήμα των δεδομένων και μετά αποθηκεύουμε τα δεδομένα σύμφωνα με αυτό το σχήμα. Στην ουσία με την εφαρμογή ενός σχήματος κατά την ανάγνωση είναι σαν τραβάμε τα δεδομένα από μία «λίμνη δεδομένων» και ανάλογα τα

ερωτήματα και τις ανάγκες, δημιουργείται το αντίστοιχο σχήμα για την παρουσίασή τους. Το 2004, η LexisNexis απέκτησε Seisint Inc. και το 2008 απέκτησε την ChoicePoint Inc. καθώς και την υψηλής ταχύτητας παράλληλης επεξεργασίας, πλατφόρμα τους. Οι δύο πλατφόρμες συγχωνεύτηκαν το 2011 σε ένα σύμπλεγμα υπολογιστών υψηλής απόδοσης (High-Performance Computing Cluster ή HPCC) το οποίο είναι ανοιχτού κώδικα υπό την άδεια Apache v2.0 . Το σύστημα αρχείων Quantcast ήταν διαθέσιμο την ίδια περίπου περίοδο.

Το 2004, η Google δημοσίευσε ένα έγγραφο σχετικά με μια διαδικασία που ονομάζεται MapReduce η οποία χρησιμοποιεί μια παρόμοια αρχιτεκτονική. Η MapReduce παρέχει ένα μοντέλο παράλληλης επεξεργασίας, και μια σχετική εφαρμογή που κυκλοφόρησε για την επεξεργασία τεράστιων ποσοτήτων από δεδομένα. Με τη MapReduce, τα ερωτήματα χωρίζονται και διανέμονται σε παράλληλους κόμβους και επεξεργάζονται παράλληλα (το βήμα Map). Τα αποτελέσματα στη συνέχεια, συγκεντρώνονται και παραδίνονται (το βήμα Reduce). Το πλαίσιο ήταν πολύ επιτυχημένο, έτσι ώστε οι πολλοί ήθελαν να αναπαράγουν τον αλγόριθμο. Ως εκ τούτου, το πλαίσιο της εφαρμογής MapReduce υιοθετήθηκε από ένα έργο ανοικτού πηγαίου κώδικα υπό την άδεια Apache ονομάστηκε Hadoop .

Η MIKE2.0 (Method for an Integrated Knowledge Environment) είναι μια ανοιχτού κώδικα προσέγγιση για τη διαχείριση πληροφοριών, που αναγνωρίζει την ανάγκη για αναθεώρηση η οποία οφείλεται στις επιπτώσεις των Big Data, οι οποίες προσδιορίζονται σε ένα άρθρο με τίτλο «Big Data Solution Offering». Η μεθοδολογία απευθύνεται στον χειρισμό μεγάλων δεδομένων ως προς τη διαχείριση των πηγών των δεδομένων, την πολυπλοκότητα στις αλληλεξαρτήσεις, και τη δυσκολία στην διαγραφή (ή τροποποίηση) σε μεμονωμένες εγγραφές.

Μελέτες που έγιναν το 2012 έδειξαν ότι μια αρχιτεκτονική πολλαπλών επιπέδων, είναι μία λύση για την αντιμετώπιση των ζητημάτων που παρουσιάζουν τα Big Data. Μια κατανομημένη παράλληλη αρχιτεκτονική διανέμει τα δεδομένα σε πολλαπλούς διακομιστές· αυτά τα παράλληλα περιβάλλοντα εκτέλεσης μπορούν να βελτιώσουν δραματικά την ταχύτητα επεξεργασίας των δεδομένων. Αυτό το είδος αρχιτεκτονικής, εισάγει τα δεδομένα ένα παράλληλο σύστημα διαχείρισης βάσεων δεδομένων (DBMS), η οποία υλοποιεί τη χρήση των MapReduce και Hadoop πλαισίων. Αυτό το είδος αρχιτεκτονικής, απομακρύνει από τον τελικό χρήστη τον τρόπο της εξεργασίας των δεδομένων με την βοήθεια ενός εμπρόσθιου εξυπηρετητή εφαρμογών (front-end application server).

Η μέθοδος αποθήκευσης δεδομένων ως μια «λίμνη δεδομένων» επιτρέπει σε έναν οργανισμό να στρέψει την προσοχή του από τον συγκεντρωτικό έλεγχο σε ένα κοινόχρηστο μοντέλο για να ανταποκριθεί στις συνεχείς αλλαγές της διαχείρισης των πληροφοριών. Με αυτόν τον τρόπο επιτρέπεται ο γρήγορος διαχωρισμός των στοιχείων στη «λίμνη» δεδομένων, μειώνοντας έτσι την χρονική επιβάρυνση.

### 3.4. Τεχνολογίες

Μια έκθεση του 2011 από το McKinsey Global Institute χαρακτηρίζει τα κύρια συστατικά και το οικοσύστημα των Big Data ως εξής:

- Τεχνικές για την ανάλυση των δεδομένων, όπως A/B testing , μηχανική μάθηση και επεξεργασία φυσικής γλώσσας
- Τεχνολογίες Big Data, όπως η επιχειρηματική ευφυΐα , το σύννεφο (cloud computing) και τις βάσεις δεδομένων
- Οπτικοποίηση, όπως χάρτες, διαγράμματα και άλλες απεικονίσεις των δεδομένων.

Πολυδιάστατα μεγάλα δεδομένα μπορούν επίσης να αναπαρασταθούν ως τανυστές , τα οποία μπορούν να είναι πιο αποτελεσματικά με υπολογισμούς με πολυγραμμική άλγεβρα. Επιπλέον τεχνολογίες που εφαρμόζονται σε μεγάλα δεδομένα περιλαμβάνουν: τη μαζική παράλληλη επεξεργασία (Massive Parallel Processing) σε βάσεις δεδομένων, ευρετικές εφαρμογές, «εξόρυξη» δεδομένων, κατανομημένα συστήματα αρχείων , κατανομημένες βάσεις δεδομένων , υποδομές με βάση το σύννεφο (εφαρμογές, αποθήκευση και υπολογιστικούς πόρους) και το διαδίκτυο.

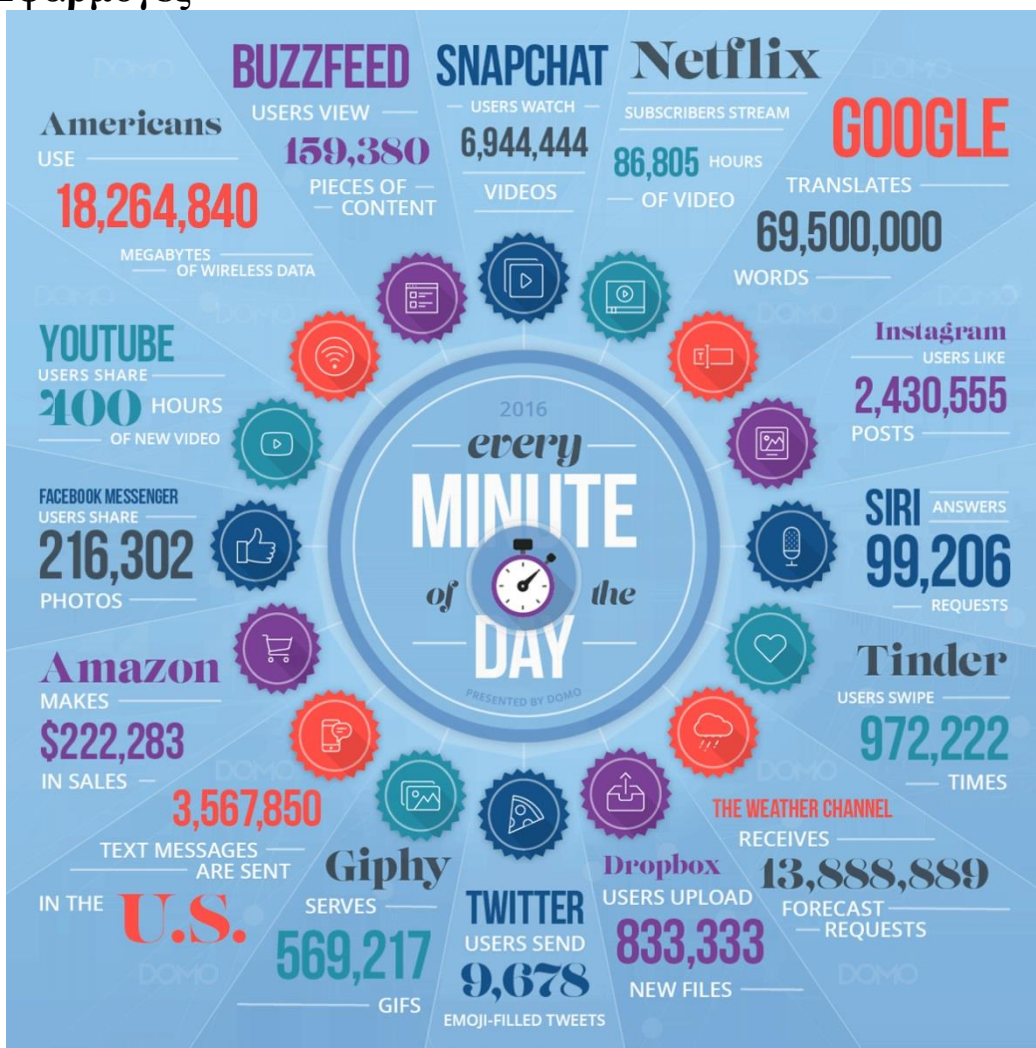
Μερικές, αλλά όχι όλες MPP σχεσιακές βάσεις δεδομένων έχουν την ικανότητα να αποθηκεύουν και να διαχειρίζονται petabytes δεδομένων. Περιορισμένη είναι όμως η ικανότητα να φορτώσουν, να παρακολουθήσουν, να δημιουργήσουν αντίγραφα ασφαλείας και να βελτιστοποιήσουν τη χρήση των μεγάλων πινάκων δεδομένων σε σχεσιακά συστήματα διαχείρισης βάσεων δεδομένων (RDBMS).

Το πρόγραμμα της DARPA για τοπολογική ανάλυση δεδομένων επιδιώκει να βρει μια βασική δομή για μαζικά σύνολα δεδομένων. Η τεχνολογία έγινε κοινή με την έναρξη της εταιρίας Ayasdi το 2008.

Οι επαγγελματίες στην ανάλυση των Big Data είναι γενικά εχθρικοί προς πιο αργή κοινόχρηστη

Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής  
 αποθήκευση, προτιμώντας λύσεις με άμεση σύνδεση στα μέσα αποθήκευσης (ATA, SATA, eSATA, NVMe, SCSI, SAS, USB, USB 3.0, IEEE 1394) στις διάφορες μορφές τους από μονάδες δίσκου στερεάς κατάστασης (solid-state drive) έως υψηλής χωρητικότητας SATA δίσκους ενσωματωμένους σε παράλληλους κόμβους επεξεργασίας. Η αντίληψη των αρχιτεκτονικών κοινής αποθήκευσης, περιοχή αποθήκευσης δικτύου (Storage Area Network) και δικτυακός αποθηκευτικός χώρος (Network Attach Storage), είναι ότι είναι σχετικά αργές, περίπλοκες και δαπανηρές. Συνεπώς δεν είναι κατάλληλες για συστήματα αναλύσεων με Big Data. Πραγματικού ή σχεδόν πραγματικού χρόνου παροχής πληροφοριών είναι ένα από τα καθοριστικά χαρακτηριστικά των αναλύσεων για Big Data. Η καθυστέρηση, επομένως, αποφεύγεται όποτε και όπου αυτό είναι δυνατόν. Τα αποθηκευμένα δεδομένα στη μνήμη είναι καλά ενώ για τα δεδομένα που αποθηκεύονται σε σκληρούς δίσκους σε μια περιοχή αποθήκευσης δικτύου (SAN) στο άλλο άκρο ενός οπτικού καναλιού δεν είναι. Το κόστος ενός SAN στην κλίμακα που απαιτείται για εφαρμογές ανάλυσης είναι υψηλότερο από ότι άλλες τεχνικές αποθήκευσης. Υπάρχουν πλεονεκτήματα όπως και μειονεκτήματα για εφαρμογές αναλύσεων των Big Data σε κοινά μέσα αποθήκευσης αλλά δεν αποτελούν την καλύτερη λύση σύμφωνα με τους ειδικούς.

### 3.5. Εφαρμογές



Εικόνα 3-4: Data created every minute of the day in 2016

Τα Big Data έχουν αυξήσει τη ζήτηση των ειδικών διαχείρισης πληροφοριών τόσο πολύ έτσι ώστε εταιρίες όπως η Software AG, η Oracle Corporation, η IBM, η Microsoft, η SAP, η EMC, η HP και η Dell έχουν δαπανήσει περισσότερα από 15 δισεκατομμύρια δολάρια σε επιχειρήσεις λογισμικού που ειδικεύονται στον τομέα της διαχείρισης των δεδομένων και αναλύσεων. Το 2010, η βιομηχανία αυτή άξιζε περισσότερα από \$100 δις δολάρια και αυξάνεται κατά σχεδόν 10 τοις εκατό το χρόνο δηλαδή περίπου δύο φορές πιο γρήγορα από τις επιχειρήσεις λογισμικού στο σύνολό τους. Οι ανεπτυγμένες οικονομίες χρησιμοποιούν όλο και περισσότερο τεχνολογίες δεδομένων, που σημαίνει αύξηση της πληροφορίας. Η παγκόσμια χωρητικότητα για την ανταλλαγή πληροφοριών μέσω τηλεπικοινωνιακών δικτύων ήταν 281 petabytes το 1986, 471 petabytes το 1993, 2.2 exabytes το 2000, 65 exabytes το 2007 και οι προβλέψεις για

το 2014 και πάνω είναι τουλάχιστον 667 exabytes ετησίως. Σύμφωνα με μια εκτίμηση, μόλις το ένα τρίτο των αποθηκευμένων πληροφοριών παγκόσμια είναι υπό τη μορφή αλφαριθμητικών κειμένων και δεδομένα εικόνας, τα οποία αποτελούν την πιο χρήσιμη μορφή για τις εφαρμογές των Big data. Αυτό δείχνει επίσης τις δυνατότητες ακόμη αχρησιμοποίητα δεδομένα (δηλ, με τη μορφή εικόνας και ήχου). Τα Big data και το Internet of Things λειτουργούν συνδυαστικά. Από τη σκοπιά των μέσων ενημέρωσης, τα δεδομένα είναι το παράγωγο κλειδί της διασύνδεσης συσκευών, επιτρέποντας έτσι, ακριβέστερη στόχευση. Το Ιντερνέτ των πραγμάτων, με τη βοήθεια των μεγάλων δεδομένων, ως εκ τούτου, μετασχηματίζει τη βιομηχανία των μέσων ενημέρωσης, τις εταιρείες, ακόμη και κυβερνήσεις, ανοίγοντας μια νέα εποχή οικονομικής ανάπτυξης και ανταγωνιστικότητας. Η διασταύρωση ανθρώπων, δεδομένων και ευφυών αλγόριθμων έχει εκτεταμένες επιπτώσεις για την αποτελεσματικότητα των μέσων ενημέρωσης. Ο πλούτος των δεδομένων που παράγονται, επιτρέπει ένα περίτεχνο στρώμα στους υπάρχοντες μηχανισμούς στόχευσης της βιομηχανίας.

Ενώ πολλοί προμηθευτές προσφέρουν έτοιμες λύσεις για μεγάλα δεδομένα, οι ειδικοί συστήνουν την ανάπτυξη της εσωτερικών λύσεων που είναι προσαρμοσμένες στο πρόβλημα της κάθε εταιρείας, εάν φυσικά η εταιρεία διαθέτει προσωπικό με τις επαρκείς τεχνικές ικανότητες.

Οι αναλύσεις από σύνολα δεδομένων μπορούν να βρουν νέες εφαρμογές σε επιχειρηματικούς τομείς, την πρόληψη των ασθενειών, την καταπολέμηση της εγκληματικότητας και ούτω καθεξής. Συχνά οι επιστήμονες, στελέχη επιχειρήσεων, επαγγελματίες της ιατρικής, της διαφήμισης και κυβερνήσεις συναντούν δυσκολίες με μεγάλα σύνολα δεδομένων σε τομείς όπως η αναζήτηση στο διαδίκτυο, τη χρηματοδότηση, την αστική πληροφορική και την πληροφορική επιχειρήσεων. Οι επιστήμονες αντιμετωπίζουν συχνά περιορισμούς στην εργασία τους γύρω από ηλεκτρονική-επιστήμη(e-Science), συμπεριλαμβανομένου της μετεωρολογίας, της γονιδιωματικής, βιολογικών νευρωνικών δικτύων, πολύπλοκες προσομοιώσεις φυσικής, της βιολογίας και της περιβαλλοντικής έρευνας. Η επεξεργασία των δεδομένων πρέπει να γίνεται με προηγμένα εργαλεία (αναλύσεις και αλγόριθμοι) για να αποκαλύψουν σημαντικές πληροφορίες. Για παράδειγμα, για να διαχειριστεί ένα εργοστάσιο πρέπει κανείς να εξετάσει τόσο ορατά και αόρατα θέματα με διάφορα σημεία του εργοστασίου. Οι αλγόριθμοι δημιουργίας πληροφοριών πρέπει να ανιχνεύουν και να αντιμετωπίζουν αόρατα θέματα όπως η υποβάθμιση μηχανημάτων, στοιχεία φθοράς, κλπ στο χώρο του εργοστασίου.



## 4. Τεχνολογίες Υλοποίησης

Bootstrap, CSS 3, HTML5, responsive templates, Grafana, JavaScript, Influx dB, Android Studio  
Σε αυτήν την ενότητα παρουσιάζονται οι τεχνολογίες οι οποίες χρησιμοποιήθηκαν για την υλοποίηση της πτυχιακής εργασίας. Πιο συγκεκριμένα χρησιμοποιήθηκε το πλαίσιο εφαρμογής bootstrap, το CSS3, η γλώσσα HTML5, το Grafana, η JavaScript, η jquery, η βάση δεδομένων influxdb και το περιβάλλον ανάπτυξης eclipse ide.

### 4.1. HTML5



Εικόνα 4-1: HTML5

Η **H**yper **T**ext **M**arkup **L**anguage HTML, γλώσσα σήμανσης υπερκειμένου, είναι η γλώσσα η οποία χρησιμοποιείται για την δημιουργία και σχεδιασμό ιστοσελίδων. Η HTML5 είναι η πέμπτη και η τρέχουσα έκδοση του προτύπου HTML η οποία είχε δημοσιευθεί τον Οκτώβριο του 2014 από την η Κοινοπραξία του Παγκόσμιου Ιστού (W3C). Οι περιηγητές (για παράδειγμα Internet Explorer, Mozilla, Chrome κ.α.) αναγνωρίζουν και μεταφράζουν τα στοιχεία της HTML. Όμως, όπως αναφέρθηκε και νωρίτερα στο κείμενο, ο όρος HTML5 αναφέρεται σε ένα σύνολο τεχνολογιών και όχι απλά σε μια νέα έκδοση της γλώσσας HTML. Το σημαντικό με αυτήν την έκδοση όμως είναι ότι περιλαμβάνει λεπτομερή μοντέλα επεξεργασίας για να ενθαρρύνει περισσότερο τις διαλειτουργικές εφαρμογές, επεκτείνεται, βελτιώνει και εκλογικεύει τη σήμανση που διατίθενται για τα έγγραφα, και εισάγει τη σήμανση και την εφαρμογή διεπαφών προγραμματισμού (API) για σύνθετες εφαρμογές web. Για τους ίδιους λόγους, η HTML5 είναι επίσης κατάλληλη για εφαρμογές ανεξαρτήτου πλατφόρμας, διότι περιλαμβάνει λειτουργίες που έχουν σχεδιαστεί με γνώμονα συσκευές οι οποίες έχουν λιγοστούς πόρους. Τέλος σημαντικό είναι να αναφέρουμε πως ενώ ορισμένα χαρακτηριστικά της HTML5 είναι συχνά συγκρίνονται με το Adobe Flash, οι δύο τεχνολογίες είναι πολύ διαφορετικές.

## 4.2. JavaScript



# JavaScript

Εικόνα 4-2: JavaScript

Η JavaScript(JS) είναι διερμηνευμένη γλώσσα προγραμματισμού για ηλεκτρονικούς υπολογιστές. Αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται. Η JavaScript είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική, με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δε σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστραφές, προστακτικό και συναρτησιακό προγραμματισμό. Επίσης χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων όπως είναι τα έγγραφα PDF, μικροεφαρμογές επιφάνειας εργασίας (desktop widgets) κτλ. Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript (όπως το Node.js) έχουν κάνει τη γλώσσα πιο δημοφιλή για την ανάπτυξη εφαρμογών ιστού στην πλευρά του διακομιστή.

### 4.3. jQuery



Εικόνα 4-3: jQuery

Η jQuery είναι μια βιβλιοθήκη της JavaScript σχεδιασμένη να απλοποιήσει την υλοποίηση σεναρίων (scripting) στη πλευρά του πελάτη (client-side) της HTML και υποστηρίζει πολλαπλούς φυλλομετρητές ιστού. Είναι από τις πιο δημοφιλείς βιβλιοθήκες της JavaScript και κυκλοφόρησε τον Ιανουάριο του 2006 από τον John Resig.

Μερικά Χαρακτηριστικά:

- **Document Object Model** element επιλογές χρησιμοποιώντας την ανοιχτού κώδικα μηχανή επιλογής πολλαπλών φυλλομετρητών Sizzle.
- DOM διάσχιση και τροποποίηση (υποστηρίζοντας CSS 1-3).
- χειρισμός DOM βασισμένος σε CSS επιλογείς που χρησιμοποιεί αναγνωριστικά(id) και κλάσεις (class) σαν κριτήρια για να κατασκευάσει επιλογείς.
- Συμβάντα
- Εφέ και κινητά στοιχεία
- JSON
- AJAX
- Επεκτασιμότητα μέσω προσθέτων.
- Εργαλεία όπως ανίχνευση χαρακτηριστικών.
- Μεθόδους συμβατότητας που είναι εγγενώς διαθέσιμες σε σύγχρονα προγράμματα περιήγησης.
- Υποστήριξη πολλαπλών φυλλομετρητών.

## 4.4. CSS3



Εικόνα 4-4: CSS3

Η Cascading Style Sheets (CSS) είναι μια γλώσσα που χρησιμοποιείται για την περιγραφή της παρουσίασης ενός εγγράφου γραμμένο σε μια γλώσσα σήμανσης. Αν και πιο συχνά χρησιμοποιείται για να ρυθμίσει το οπτικό στυλ των ιστοσελίδων και των διεπαφών χρηστών που είναι γραμμένες σε HTML και XHTML, η γλώσσα μπορεί να εφαρμοστεί σε οποιοδήποτε έγγραφο XML, συμπεριλαμβανομένων των απλό XML, SVG και XUL. Μπορεί επίσης να ρυθμιστεί κατάλληλα ώστε να συνθέσει ομιλία, ή άλλα μέσα. Μαζί με την HTML και JavaScript, η CSS αποτελεί τον ακρογωνιαίο λίθο της τεχνολογίας που χρησιμοποιείται από περισσότερες ιστοσελίδες για να δημιουργήσουν μια οπτικά ελκυστική ιστοσελίδα, διεπαφές χρήστη για εφαρμογές διαδικτύου, και για κινητές εφαρμογές

## 4.5. Responsive Design

Ο ανταποκρίσιμη ή «δυναμική» σχεδίαση (Responsive Design) πήρε την ονομασία του από την λέξη «Respond» και σημαίνει ότι μια ιστοσελίδα προσαρμόζεται δυναμικά και άμεσα (on the fly) σε κάθε διάσταση και προσανατολισμό οθόνης. Αυτό σημαίνει ότι όταν κάποιος πλοηγηθεί σε μια ιστοσελίδα η οποία χρησιμοποιεί responsive template, για παράδειγμα ένας κινητός χρήστης, η ιστοσελίδα θα διαμορφωθεί με τέτοιο τρόπο ώστε να μην χρειάζεται ο χρήστης να κάνει μεγέθυνση ή πλάγια κύλιση για να διαβάσει με ευκολία το περιεχόμενό της. Αυτό είναι πάρα πολύ σημαντικό καθώς τα τελευταία χρόνια έχει παρατηρηθεί μια ραγδαία αύξηση της περιήγησης στο διαδίκτυο από κινητές συσκευές.

Πλεονεκτήματα δυναμικής σχεδίασης:

- Διευκόλυνση του επισκέπτη της ιστοσελίδας: Καθώς ο επισκέπτης της ιστοσελίδας δεν χρειάζεται να κάνει συνεχώς μεγέθυνση για να διαβάσει το περιεχόμενο της ιστοσελίδας, του προσφέρεται έτσι ένα πιο ευχάριστο, καλαισθητό και γρήγορο περιβάλλον για να την εξερευνήσει.
- Εύκολη ενημέρωση. Σε αντίθεση με το να υπάρχει μία ιστοσελίδα για ηλεκτρονικούς υπολογιστές και μία εφαρμογή ή μια άλλη έκδοση ιστοσελίδας για κινητές συσκευές, η χρήση δυναμικής σχεδίασης διευκολύνει το διαχειριστή της με την έννοια ότι δεν χρειάζεται να ενημερώνει δύο ιστοσελίδες αλλά μόνο την κεντρική. Κατά συνέπεια αυτό τον γλυτώνει από πολύ χρόνο.
- Αύξηση χρηστικότητας και χρόνου παραμονής στην ιστοσελίδα: Όταν ένας χρήστης

Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής  
επισκεφτεί την ιστοσελίδα από μια κινητή συσκευή, δεν θα παραμείνει πολύ ώρα στην ιστοσελίδα αν αυτή δεν τον διευκολύνει. Ενώ αν η ιστοσελίδα είναι σχεδιασμένη «δυναμικά» οι πιθανότητες να περιηγηθεί σε όλες τις υπηρεσίες ή τα προϊόντα είναι πολύ υψηλότερες.

- Προσαρμοστικότητα: Μία δυναμική ιστοσελίδα θα προσαρμοστεί σε οποιαδήποτε ανάλυση, η οποία μπορεί να μην είναι ακόμα διαθέσιμη και να μην γνωρίζουμε.
- Κόστος: Η δυναμική σχεδίαση είναι οικονομικότερη από τη δημιουργία ξεχωριστών ιστοσελίδων ή εφαρμογών για κάθε συσκευή.

## 4.6. Bootstrap



Εικόνα 4-5: Bootstrap

Το Bootstrap είναι μια συλλογή εργαλείων ανοιχτού κώδικα για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών. Περιέχει HTML και CSS για τις μορφές τυπογραφίας, κουμπιά πλοήγησης και άλλων στοιχείων του περιβάλλοντος, καθώς και προαιρετικές επεκτάσεις JavaScript. Αναπτύχθηκε από τους Mark Otto και τον Jacob Thornton, προγραμματιστές στο Twitter ως ένα πλαίσιο για την ενθάρρυνση της συνέπειας στη δημιουργία εσωτερικών εργαλείων. Πριν από το Bootstrap, διάφορες βιβλιοθήκες χρησιμοποιήθηκαν για την ανάπτυξη της διεπαφής χρήστη του Twitter, η οποίες οδήγησαν σε αστάθειες και υψηλό κόστος συντήρησης.

Είναι συμβατό με τις τελευταίες εκδόσεις των Google Chrome , Firefox , Internet Explorer , Opera Safari και Edge φυλλομετρητών, αν και ορισμένα από αυτά τα προγράμματα περιήγησης δεν υποστηρίζονται σε όλες τις πλατφόρμες. Από την έκδοση 2.0 και μετά υποστηρίζει επίσης δυναμικό σχεδιασμό ιστοσελίδων. Αυτό σημαίνει ότι η διάταξη των ιστοσελίδων προσαρμόζεται δυναμικά, λαμβάνοντας υπόψη τα χαρακτηριστικά της συσκευής που χρησιμοποιείται. Ξεκινώντας με την έκδοση 3.0, το Bootstrap υιοθέτησε τη φιλοσοφία δυναμικής σχεδίασης, δίνοντας έμφαση στο δυναμικό σχεδιασμό από προεπιλογή. Η τρέχουσα έκδοση είναι η 3.3.7. Η νέα έκδοση 4.0 alpha προστεθεί υποστήριξη για Sass και flexbox.

Το Bootstrap είναι «σπονδυλωτό» και αποτελείται ουσιαστικά από μια σειρά στυλ εμφάνισης Less stylesheets που εφαρμόζουν τα διάφορα στοιχεία της εργαλειοθήκης. Ένα στυλ που ονομάζεται «Bootstrap Less» περιλαμβάνει όλα stylesheets. Οι προγραμματιστές μπορούν να προσαρμόσουν το ίδιο το Bootstrap αρχείο, επιλέγοντας τα στοιχεία που επιθυμούν να χρησιμοποιήσουν στα έργα τους. Προσαρμογές είναι εφικτές αλλά σε περιορισμένο βαθμό, μέσω ενός κεντρικού στυλ διαμόρφωσης. Πιο βαθιές αλλαγές είναι δυνατόν να γίνουν με από το δηλώσεις τύπου Less. Η χρήση της γλώσσας στυλ Less επιτρέπει τη χρήση μεταβλητών, λειτουργιών και χειρισμών, ένθετες επιλογείς, καθώς και mixins (παρόμοια λειτουργία με τις αφηρημένες κλάσεις στη Java-abstract). Από την έκδοση 2.0, η διαμόρφωση του Bootstrap έχει επίσης μία ειδική επιλογή που ονομάζεται «Προσαρμογή». Έτσι, ο σχεδιαστής του έργου επιλέγει από μια φόρμα τα επιθυμητά στοιχεία και τα προσαρμόζει ανάλογα τις ανάγκες του. Στη συνέχεια δημιουργείται ένα πακέτο που περιλαμβάνει ήδη το «προ-κτισμένο» στυλ CSS. Η τέταρτη έκδοση του Bootstrap θα χρησιμοποιεί στυλ τύπου Sass αντί του τωρινού Less.

- Stylesheets: Το Bootstrap παρέχει ένα σύνολο από στυλ που περιέχουν όλα τα βασικά στοιχεία HTML. Αυτά παρέχουν μια ομοιόμορφη, μοντέρνα εμφάνιση για τη μορφοποίηση του κειμένου, πινάκων και στοιχείων φόρμας.
- Επαναχρησιμοποιήσιμα συστατικά: Σε αντίθεση με τυπικά στοιχεία HTML, το Bootstrap περιέχει και άλλα κοινώς χρησιμοποιούμενα στοιχεία διεπαφής. Τα συστατικά εφαρμόζονται ως κλάσεις

Πτυχιακή Εργασία τμήματος Μηχανικών Πληροφορικής  
CSS, οι οποίες πρέπει να εφαρμόζονται σε ορισμένα στοιχεία HTML σε μια σελίδα.

- JavaScript: το Bootstrap έρχεται με πολλά στοιχεία JavaScript, με τη μορφή προσθέτων jQuery. Αυτά με τη σειρά τους παρέχουν επιπλέον στοιχεία διεπαφής χρήστη, όπως παράθυρα διαλόγου, επεξηγήσεις, και «καρουσέλ εικόνων». Μπορούν επίσης να επεκτείνουν τη λειτουργικότητα ορισμένων υφιστάμενων στοιχείων διεπαφής, συμπεριλαμβανομένου, για παράδειγμα μια λειτουργία αυτόματης συμπλήρωσης για πεδία εισαγωγής.

#### **Bootstrap 4:**

Το 2015 Bootstrap προγραμματιστές έχουν ανακοινώσει ότι εργάζονται σε μια νέα έκδοση πλαίσιο με πολλαπλές ενημερώσεις:

- Άλλαξε σύνταξη από Less σε Sass. Ταχύτερη σύνταξη για Bootstrap χάρη στην Libsass και Sass-σύνταξη.
- Ενημερωμένο σύστημα πλέγματος. Νέα σημασιολογία mixins και στόχευση κινητών συσκευών.
- Υποστήριξη flexbox. Περισσότερα στοιχεία απόκρισης, χάρη στην flexbox από το HTML5 .
- Εγκατάλειψη πλαισίων, μικρογραφιών και πάνελ. Έχουν αντικατασταθεί με πλαίσια περιεχομένου που ονομάζονται «κάρτες».
- Όλες οι επαναφορές HTML μετατέθηκαν στην ενότητα «επανεκκίνηση».
- Προσαρμογή. Από την έκδοση 4 είναι ευκολότερη η προσαρμογή στυλ για τα πρότυπα, δεδομένου ότι όλες οι επιλογές αποθηκεύονται σε μεταβλητές Sass. Μετά την μεταροπή του αρχείου σε εκτελέσιμο ο προγραμματιστής παίρνει ένα ενημερωμένο CSS αρχείο.
- Έκλεισε υποστήριξη τον IE8. Οι προγραμματιστές που χρειάζονται υποστήριξη για IE8, θα πρέπει να συνεχίσουν να χρησιμοποιούν Bootstrap 3.
- Οι μονάδες άλλαξαν από pixels σε μονάδες rem και em.
- Επανεγγραφή όλων των προσθέτων της JavaScript.
- Βελτίωση επεξηγήσεων.
- Βελτιωμένη αναζήτηση και τεκμηρίωση.

Το Bootstrap είναι ανοικτού κώδικα και διατίθεται στο GitHub. Οι προγραμματιστές ενθαρρύνονται να συμμετάσχουν στο πρόγραμμα και να κάνουν τη δική τους συμβολή στην πλατφόρμα.

## **4.7. InfluxDB**



**Εικόνα 4-6: InfluxDB**

Η InfluxDB είναι ένα ανοικτού κώδικα βάση δεδομένων χρονολογικής σειράς που αναπτύχθηκε από την InfluxData. Είναι γραμμένη σε Go και βελτιστοποιημένη για γρήγορη αποθήκευση και ανάκτηση των χρονολογικών σειρών δεδομένων σε τομείς όπως είναι η παρακολούθηση ενεργειών, μετρήσεις εφαρμογών, δεδομένα αισθητήρων του ιντερνέτ των πραγμάτων, και αναλύσεις πραγματικού χρόνου. Η InfluxDB δεν εξαρτάται από άλλα προγράμματα και παρέχει ένα γλώσσα προγραμματισμού παρόμοια με την SQL με ενσωματωμένες χρονολογικές συναρτήσεις για τη δημιουργία ερωτημάτων, σε δομές δεδομένων αποτελούμενες από τις μετρήσεις, σειρές, και τα σημεία. Κάθε σημείο αποτελείται από πολλά ζεύγη κλειδιών-τιμών ονομάζεται «fieldset» και έχει χρονική σήμανση. Όταν συγκεντρώνονται από ένα σύνολο από ζεύγη κλειδιού-τιμής ονομάζεται «tagset», αυτά καθορίζουν μια σειρά. Τέλος, οι σειρές ομαδοποιούνται από ένα αναγνωριστικό συμβολοσειράς για να σχηματίσουν μια μέτρηση. Οι τιμές μπορούν να είναι ακέραιοι 64-bit, 64-bit δείκτες κινητής υποδιαστολής, συμβολοσειράς, και αληθής ή ψευδής. Εισαγωγή δεδομένων μπορεί να γίνει με τα πρωτόκολλα HTTP, UDP και TCP. Επίσης διαθέτει και βιβλιοθήκες για διάφορες γλώσσες όπως η Go, Haskell, Java, JavaScript/Node.js, Lisp, .Net, Perl, PHP,

Κάποια από τα βασικά χαρακτηριστικά της είναι:

- Ειδικά κατασκευασμένη για αποθήκευση δεδομένων χρονολογικής σειράς.
- Έχε γραφτεί εξολοκλήρου σε μια γλώσσα(Go), που σημαίνει πως ότι χρησιμοποιεί μια και μόνο βιβλιοθήκη για τη μεταγλώττιση της.
- Απλή με υψηλής απόδοσης ερωτήματα με την HTTP διεπαφή.
- Υποστήριξη πρόσθετων για άλλα πρωτόκολλα εισαγωγής δεδομένων όπως Graphite,collectd και OpenTSDB.
- Δυνατότητα αποθήκευσης δισεκατομμυρίων σημείων.
- Δυνατότητα αποθήκευσης χιλιάδων χρονοσειρών
- Συγχώνευση πολλαπλών χρονοσειρών
- Ενσωματωμένο περιβάλλον διαχείρισης
- Παρόμοια γλώσσα με την SQL για τη δημιουργία ερωτημάτων.

## 4.8. Grafana



Εικόνα 4-7: Grafana

Το Grafana είναι μια κορυφαία εφαρμογή ανοιχτού κώδικα για την οπτικοποίηση δεδομένων μέτρησης μεγάλης κλίμακας. Παρέχει ένα ισχυρό και κομψό τρόπο για τη δημιουργία, διαμοιρασμό και εξερεύνηση δεδομένων και πινάκων από διαφορετικές βάσεις δεδομένων σας, είτε με την ομάδα σας ή τον κόσμο. Χρησιμοποιείται συχνά για εφαρμογές διαδικτύου και αναλύσεων, αλλά πολλοί το χρησιμοποιούν σε άλλους τομείς, συμπεριλαμβανομένου των βιομηχανικών αισθητήρων, οικιακού αυτοματισμού, τον καιρό, και τον έλεγχο και παρακολούθηση διαδικασιών. Το Grafana διαθέτει δυνατότητα προσαφαίρεσης πάνελ και πηγών δεδομένων που επιτρέπει έτσι εύκολη επεκτασιμότητα του. Έχει υπάρξει πλούσια υποστήριξη για βάσεις δεδομένων όπως τις Graphite, InfluxDB, OpenTSDB, KairosDB, Prometheus, HP Vertica, Monitoring Art, OpenNMS, OSISoft-PI, Blueflood κ.τ.λ. Ακόμα υπολογίζεται και η προσθήκη της SQL σε κάποια στιγμή. Τέλος διαθέτει μεγάλη ποικιλία από πάνελ συμπεριλαμβανομένου ενός πλήρως εξοπλισμένου πίνακα με γράφημα και επιλογές απεικόνισης στα οποία γίνονται συνεχώς νέες προσθήκες. Η δεύτερη έκδοση κυκλοφόρησε τον Απρίλιο του 2015 η οποία έκτος από πολλές αλλαγές και χαρακτηριστικά προσθέτει και τον δικό της διακομιστή (backend server). Χιλιάδες επιχειρήσεις, μικρές και μεγάλες βασίζονται στο Grafana, και υπάρχει μια ζωντανή και ενεργή κοινότητα που προσθέτει συνεχώς νέα χαρακτηριστικά και βελτιώσεις. Το Grafana είναι ανοικτού κώδικα με την άδεια του Apache License 2.0.

Τα κυριότερα χαρακτηριστικά της Grafana είναι τα παρακάτω:

- Γρήγορα και ευέλικτα γραφήματα τα οποία δημιουργούνται στην πλευρά του χρήστη με μία σειρά από επιλογές.
- Δυνατότητα επιλογής συγκεκριμένης περιοχής για μεγέθυνση.
- Διαθέτει πολλές μορφές απεικόνισης δεδομένων όπως μπάρες, γραμμές, σημεία κ.τ.λ.
- Προβολή ή μεταβολή των γραφημάτων σε πλήρης οθόνη.
- Πλήρης έλεγχος στην σχεδίαση της κάθε χρονοσειράς.
- Δυνατότητα ανάμιξης διαφορετικών τύπων γραφημάτων σε μία οθόνη.
- Δυνατότητα εξαγωγής του γραφήματος σε εικόνα τύπου png.



## 4.9. Android Studio



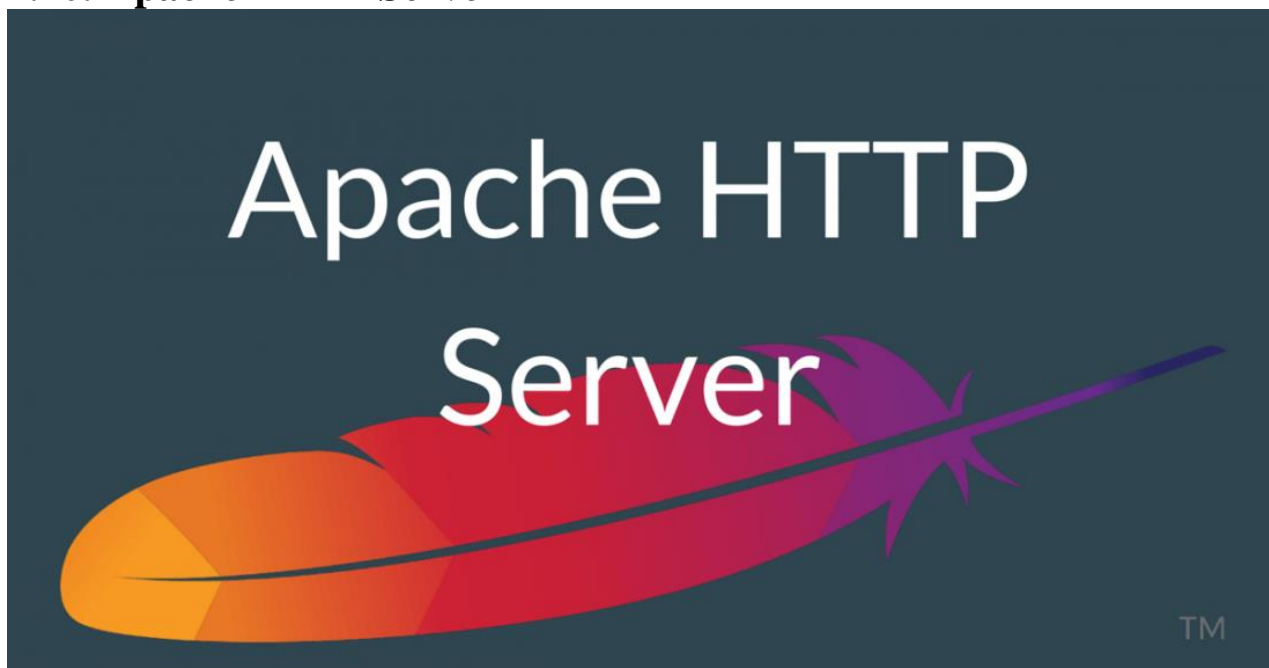
Εικόνα 4-8: Android Studio

Το Android Studio είναι ένα ολοκληρωμένο προγραμματιστικό περιβάλλον (IDE) για ανάπτυξη εφαρμογών στην πλατφόρμα Android. Ανακοινώθηκε επίσημα στις 16 Μαΐου του 2013 στο συνέδριο της Google I/O από την Google Product Manager, Katherine Chou. Το Android Studio είναι διαθέσιμο ελεύθερα με την άδεια Apache License 2.0. Το Android Studio ήταν διαθέσιμο σε πρώιμο στάδιο για προεπισκόπηση ξεκινώντας από την έκδοση 0.1 τον Μάιο του 2013, και έπειτα ξεκίνησε το δοκιμαστικό στάδιο από την έκδοση 0.8 που κυκλοφόρησε τον Ιούνιο του 2014. Η πρώτη «σταθερή» έκδοση κυκλοφόρησε το Δεκέμβριο του 2014, ξεκινώντας με την έκδοση 1.0. Είναι βασισμένο στο λογισμικό της JetBrains IntelliJ IDEA, το οποίο σχεδιάστηκε αποκλειστικά για προγραμματισμό Android. Είναι διαθέσιμο για Windows, Mac OS X και Linux, και αντικατέστησε το Eclipse Android Development Tools (ADT) ως το πλατφόρμα της Google για ανάπτυξη εφαρμογών Android.

Μερικά βασικά χαρακτηριστικά του είναι τα ακόλουθα:

- Διαθέτει ειδικό εργαλείο (Lint) για να μετρήσει την απόδοση, τη χρηστικότητα, να ελέγξει έκδοση συμβατότητας και να βρει άλλα πιθανά προβλήματα που μπορεί να υπάρχουν.
- Έτοιμα πρότυπα για τη δημιουργία νέων εφαρμογών Android.
- Ένα πλούσιο πρόγραμμα επεξεργασίας που επιτρέπει στους χρήστες να προσθέτουν εύκολα νέα στοιχεία με «drag-and-drop», και επιλογή για προεπισκόπηση της εφαρμογής σε πολλαπλές διαμορφώσεις οθόνων.
- Υποστήριξη για την κατασκευή Android Wear εφαρμογών.
- Ενσωματωμένη υποστήριξη για την πλατφόρμα Google Cloud, επιτρέποντας την ενσωμάτωση με το Google Cloud Messaging και App Engine.
- Μια εικονική μηχανή που χρησιμοποιείται για να τρέξει και να δοκιμάσει εφαρμογές.

## 4.10. Apache HTTP Server



Εικόνα 4-9: Apache HTTP Server

Ο Apache HTTP γνωστός και απλά σαν Apache είναι ένας εξυπηρετητής του παγκόσμιου ιστού (web). Όποτε ένας χρήστης επισκέπτεται ένα ιστότοπο το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν διακομιστή (server) μέσω του πρωτοκόλλου HTTP, ο οποίος παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης. Ο Apache είναι ένας από τους δημοφιλέστερους εξυπηρετητές ιστού, εν μέρει διότι λειτουργεί σε διάφορες πλατφόρμες όπως Windows, Linux, Unix και Mac OS. Κυκλοφόρησε υπό την άδεια λογισμικού Apache και είναι λογισμικό ανοιχτού κώδικα. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation). Η πρώτη του έκδοση, γνωστή ως NCSA HTTPd, δημιουργήθηκε από τον Robert McCool και κυκλοφόρησε το 1993. Θεωρείται ότι έπαιξε σημαντικό ρόλο στην αρχική επέκταση του παγκόσμιου ιστού. Ήταν η πρώτη βιώσιμη εναλλακτική επιλογή που παρουσιάστηκε απέναντι στον εξυπηρετητή http της εταιρείας Netscape και από τότε έχει εξελιχθεί στο σημείο να ανταγωνίζεται άλλους εξυπηρετητές βασισμένους στο Unix σε λειτουργικότητα και απόδοση. Η τελευταία σταθερή έκδοση είναι η 2.4.25 η οποία δημοσιεύτηκε στις 20 Δεκεμβρίου 2016. Από τον Ιούλιο του 2016, ο Apache υπολογίστηκε για να εξυπηρετήσει το 46.41% όλων των ενεργών δικτυακών τόπων και το 43,18% από τις κορυφαίες ιστοσελίδες.

Κάποια από τα χαρακτηριστικά του:

- Χειρίζεται εύκολα πάνω από 10.000 ταυτόχρονες συνδέσεις
- Υποστήριξη .htaccess
- Πολλαπλούς μηχανισμούς εξισορρόπησης φορτίου
- Ανοχή σφαλμάτων και ανακατεύθυνσης με αυτόματη ανάκτηση
- Υποστήριξη WebSocket, FastCGI, SCGI, AJP και uWSGI με προσωρινή αποθήκευση
- Δυναμική διαμόρφωση
- Υποστήριξη TLS/SSL με SNI και OCSP, μέσω του OpenSSL .
- Συμβατότητα με IPv6
- Υποστήριξη πρωτοκόλλου HTTP/2
- Συμπίεση και αποσυμπίεση με gzip
- Εύρεση τοποθεσίας με βάση την IP
- Ενσωματωμένο κώδικα Perl, PHP και Lua
- Υποστήριξη CGI
- Υποστήριξη XML
- Προβολή κατάστασης σε πραγματικό χρόνο.

## 4.11. Netbeans IDE



# NetBeans

Εικόνα 4-10: Netbeans IDE

Το NetBeans είναι μια πλατφόρμα ανάπτυξης λογισμικού γραμμένη σε Java. Η πλατφόρμα NetBeans επιτρέπει στις εφαρμογές που πρέπει να αναπτυχθούν από ένα σύνολο αρθρωτών στοιχείων λογισμικού που ονομάζονται ενότητες. Οι εφαρμογές που βασίζονται στην πλατφόρμα NetBeans, συμπεριλαμβανομένου και του ολοκληρωμένου περιβάλλοντος ανάπτυξης (IDE) του Netbeans, μπορούν να επεκταθούν από τρίτους προγραμματιστές. Το NetBeans IDE προορίζεται κυρίως για την ανάπτυξη σε Java, αλλά υποστηρίζει επίσης και άλλες γλώσσες, ιδίως PHP, C/C++ και HTML5. Το NetBeans είναι πλατφόρμα που υποστηρίζει Microsoft Windows, Mac OS X, Linux, Solaris και άλλες πλατφόρμες που υποστηρίζουν ένα συμβατό **Java Virtual Machine**. Η ομάδα του NetBeans υποστηρίζει ενεργά το προϊόν και αναζητάει προτάσεις από την ευρύτερη κοινότητα. Σε κάθε έκδοση έχει προηγηθεί δόκιμη και ανατροφοδότηση από την κοινότητα.

Το NetBeans ξεκίνησε το 1996 ως Xelfi (λογοπαίγνιο της γλώσσας προγραμματισμού Delphi) ως ένα έργο φοιτητή για Java IDE υπό την καθοδήγηση του Τμήματος Μαθηματικών και Φυσικής στο Πανεπιστήμιο του Καρόλου στην Πράγα. Το 1997, ο Roman Stanek σχηματίζεται μια επιχείρηση γύρω από το έργο και παράγει εμπορικές εκδόσεις του NetBeans IDE έως ότου αγοράστηκε από την Sun Microsystems το 1999. Η Sun έκανε το NetBeans ανοικτού κώδικα, τον Ιούνιο του επόμενου έτους. Από τότε, η κοινότητα του NetBeans συνέχισε να αυξάνεται. Το 2010, η Sun (και, συνεπώς, το NetBeans) εξαγοράστηκε από την Oracle Corporation. Σύμφωνα με την Oracle, το NetBeans ανταγωνίστηκε με το JDeveloper, ένα δωρεάν λογισμικό IDE που ιστορικά υπήρξε προϊόν της εταιρείας. Τον Σεπτέμβριο του 2016 η Oracle υπέβαλε πρόταση να δωρίσουν το έργο NetBeans στο Ίδρυμα Ελεύθερου Λογισμικού Apache, δηλώνοντας ότι «το άνοιγμα του μοντέλου NetBeans θα δώσει στους συμμετέχοντες ουσιαστικά μεγαλύτερη φωνή για κατεύθυνση του έργου και τη μελλοντική επιτυχία του, μέσα από την επερχόμενη έκδοση της Java 9 και NetBeans 9 και ακόμα παραπέρα». Η κίνηση αυτή εγκρίθηκε τον δημιουργό της Java, James Gosling. Η τελευταία έκδοση του NetBeans IDE είναι η 8.2 και κυκλοφόρησε στις 3 Οκτωβρίου το 2016.

## 5. Ανάλυση Portal

Σκοπός της πτυχιακής είναι να φτιαχτεί μια ηλεκτρονική πύλη που θα συλλέγει σε πραγματικό χρόνο δεδομένα από διάφορους μετεωρολογικούς αισθητήρες και θα τα παρουσιάζει σε μια ιστοσελίδα υπό μορφή τιμών και γραφημάτων, οι οποίες θα ανανεώνονται δυναμικά. Κατόπιν συζήτησης με τον επιβλέποντα καθηγητή η βιβλιοθήκη γραφημάτων η οποία θα χρησιμοποιηθεί είναι η Grafana, η βάση δεδομένων η influxdb και τα δεδομένα θα αντλούνται από τον διακομιστή bioklima5 του TEI. Κατόπιν θα δημιουργηθεί και εφαρμογή για κινητούς χρήστες η θα παρουσιάζει της τρέχουσες τιμές.

### 5.1. Απαιτήσεις συστήματος

Η ηλεκτρονική πύλη και η εφαρμογή θα πρέπει να είναι φιλικές προς το χρήστη, να είναι συμβατές με το υπάρχον portal του Τ.Ε.Ι., να επιτρέπεται η διασύνδεση των δεδομένων μεταξύ της παλαιάς, της νέας πύλης και της εφαρμογής, να είναι συμβατό με κινητές συσκευές και να τα γραφήματα να υλοποιηθούν με το πλαίσιο εφαρμογής Grafana. Η εφαρμογή για τις κινητές συσκευές να είναι σχεδιασμένη στην πλατφόρμα Android.

Διερεύνηση state of the art της ηλεκτρονικής πύλης:

- Web-based συστήματα διαχειριστή. Είναι η περιοχή από την οποία ο υπεύθυνος του καταστήματος οργανώνει-διαχειρίζεται-ελέγχει όλες τις παραμέτρους του συστήματος.
- Web-based σύστημα χρήστη. Κάθε χρήστης μπορεί να ενημερώνεται για τα μετεωρολογικά δεδομένα καθώς και το ιστορικό τους σε γράφημα. Οι κινητοί χρήστες έχουν και την επιλογή της ξεχωριστής εφαρμογής για Android.
- Απαιτούνται γνώσεις προγραμματισμού από τον διαχειριστή του back-end συστήματος για την οργάνωση και τη διαχείριση της βάσης δεδομένων.
- Εύκολη διαμόρφωση γραφημάτων χωρίς να υπάρχει κανένας περιορισμός .
- Προστασία του συστήματος διαχείρισης με κωδικό πρόσβασης.

Λειτουργικές Απαιτήσεις	
Κωδικός Απαίτησης	Προσδιορισμός/Τίτλος Απαίτησης
Requirement 1	Web-based σύστημα διαχειριστή/χρήστη.
Requirement 2	Διαμόρφωση γραφημάτων ανάλογα με τις ανάγκες
Μη Λειτουργικές Απαιτήσεις	
Requirement 3	Προστασία του συστήματος διαχείρισης με κωδικό
Requirement 4	Δημιουργία ξεχωριστής εφαρμογής για κινητούς χρήστες

Πίνακας 5-1: Προσδιορισμός απαιτήσεων

#### Κωδικός Απαίτησης

Requirement 1 Functional Requirement

#### Προσδιορισμός/Τίτλος Απαίτησης

Web-based σύστημα διαχειριστή/χρήστη

#### Περιγραφή Απαίτησης

1. Το σύστημα διαχειριστή. Είναι η περιοχή από την οποία ο υπεύθυνος του συστήματος οργανώνει-διαχειρίζεται-ελέγχει όλες τις παραμέτρους του συστήματος.
2. Απαιτούνται γνώσεις προγραμματισμού για τον διαχείριση και οργάνωση της βάσης δεδομένων.
3. Το σύστημα πελάτη. Κάθε πελάτης μπορεί να ενημερώνεται τις καιρικές συνθήκες που επικρατούν με τη βοήθεια των γραφημάτων.

#### Κωδικός Απαίτησης

Requirement 2 Functional Requirement

#### Προσδιορισμός/Τίτλος Απαίτησης

### Περιγραφή Απαίτησης

1. Ο διαχειριστής έχει την δυνατότητα να προσθέσει νέα γραφήματα, να αφαιρέσει υπάρχουσα γραφήματα, να διαμορφώσει τον τρόπο απεικόνισης γραφημάτων, να δώσει την δυνατότητα στους χρήστες να δουν ή όχι και εάν μπορούν να τα απεργαστούν.
2. Ο χρήστης μπορεί να επιλέξει χρονική περίοδο για τα γραφήματα και να εναλλάσσει τα διαθέσιμα ταμπλό γραφημάτων.

### Κωδικός Απαίτησης

Requirement 3 Non Functional Requirement

### Προσδιορισμός/Τίτλος Απαίτησης

Προστασία του συστήματος διαχείρισης με κωδικό

### Περιγραφή Απαίτησης

1. Η περιοχή του διαχειριστή είναι προστατευμένη με κωδικό πρόσβασης

Κωδικός Απαίτησης

Requirement 4 Non Functional Requirement

Προσδιορισμός/Τίτλος Απαίτησης

Δημιουργία ξεχωριστής εφαρμογής για κινητούς χρήστες

Περιγραφή Απαίτησης

1. Στους κινητούς χρήστες παρέχεται η δυνατότητα να έχουν ξεχωριστή εφαρμογή για αμεσότερη προβολή των καιρικών συνθηκών.

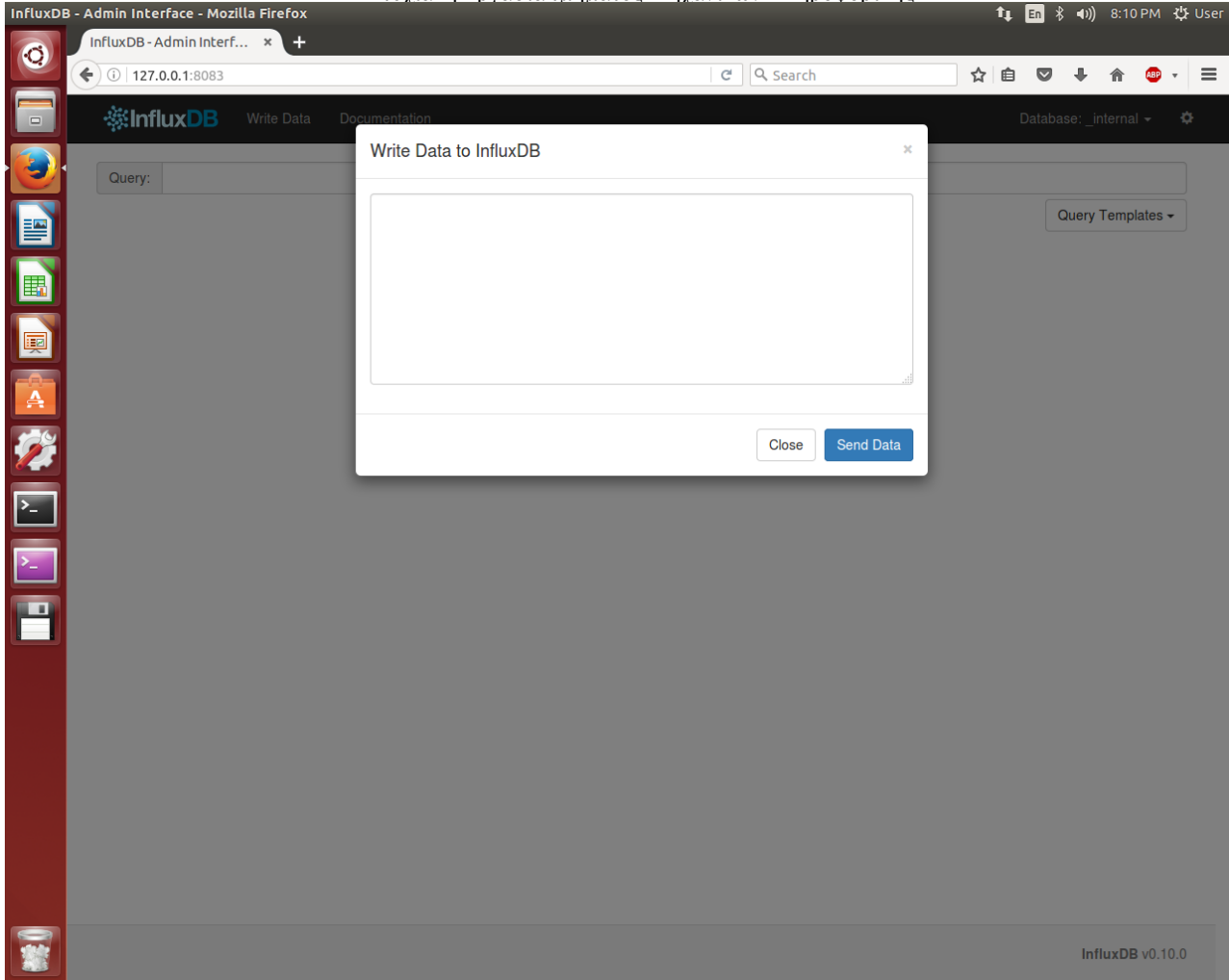
## 5.2. Σενάρια

Περίπτωση Χρήσης	Σενάρια
Διαχείριση Συστήματος	➤ Εισαγωγή νέων πηγών δεδομένων στη βάση.
Προβολή Πληροφοριών	➤ Προβολή Γραφημάτων ή Τρεχουσών Τιμών

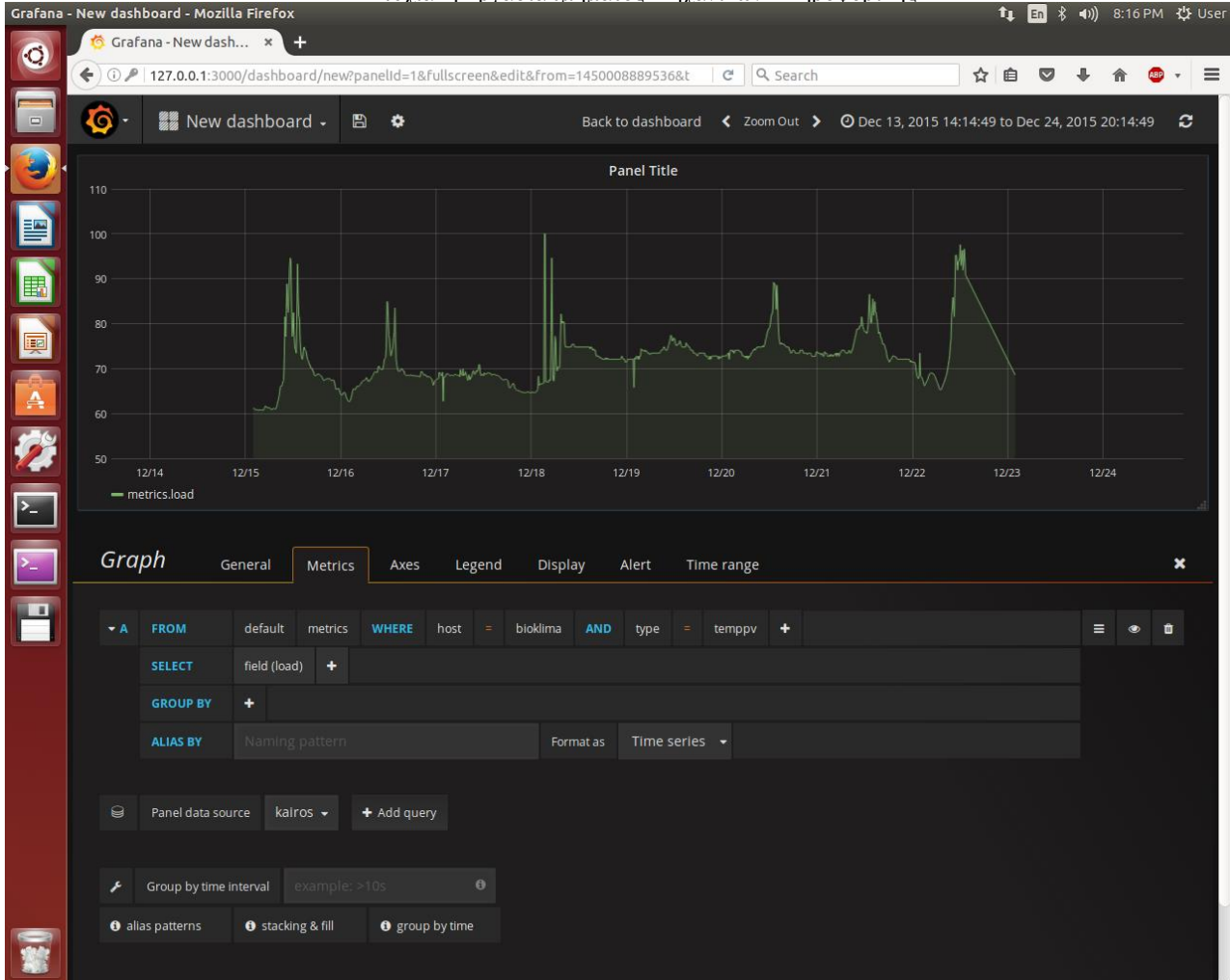
Πίνακας 5-2: Περιπτώσεις Χρήσης

<b>Use Case Name:</b> Διαχείριση Συστήματος	<b>Importance Level:</b> Υψηλό
<b>Primary Actor:</b> Διαχειριστής	<b>Use Case Type:</b> Ουσιώδης
<b>Stakeholders and Interests:</b> Διαχειριστής back end	
<b>Brief Description:</b> Ο Διαχειριστής πραγματοποιεί ελέγχους κατά την λειτουργία της εφαρμογής (έλεγχος για την ενημέρωση της εφαρμογής με νέα δεδομένα). Είναι παθητικός υποχρεωτικός δράστης.	
<b>Trigger:</b> Συντήρηση συστήματος ή υπάρχουν νέα καιρικά δεδομένα.	
<b>Relationships:</b> Συσχέτιση (association) : Περιλαμβάνει (include) : Δημιουργία, Εύρεση, Τροποποίηση, Διαγραφή Επεκτείνει (extend) : Εξειδικεύει (generalization) :	
<b>Normal Flow of Events:</b> 1. Επιλογή υπάρχουσας ή δημιουργία νέας βάσης στη influxdb. 2. Προσθήκη νέων δεδομένων στην βάση. 3. Δημιουργία Τροποποίηση, Διαγραφή Γραφημάτων.	
<b>Sub-flows:</b>	
<b>Alternate/Exceptional Flows:</b> Σε περίπτωση που εισαχθεί νέος μετεωρολογικός σταθμός απαιτείται εγγραφή κώδικα.	

Πίνακας 5-3: Σενάριο Διαχείρισης Συστήματος



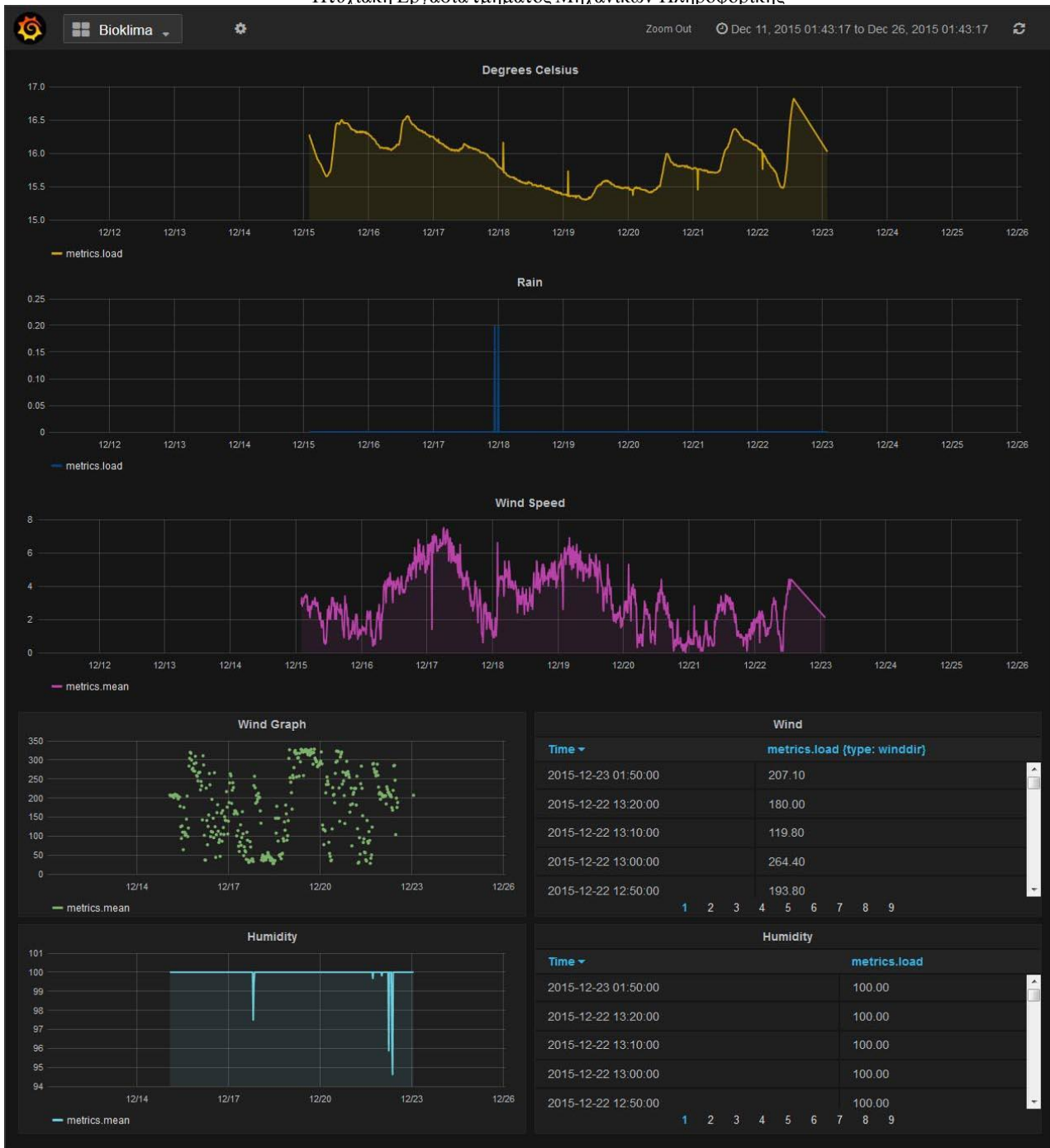
Εικόνα 5-1: Εισαγωγή δεδομένων στην influxdb από HTTP/API



Εικόνα 5-2: Δημιουργία γραφήματος στο Grafana

<b>Use Case Name: Προβολή Πληροφοριών</b>	<b>Importance Level: Υψηλό</b>
<b>Primary Actor: Διαχειριστής/Χρήσης</b>	<b>Use Case Type: Ουσιώδης</b>
<b>Stakeholders and Interests: Διαχειριστής/Χρήστης front end</b>	
<b>Brief Description:</b> Η εφαρμογή επιτρέπει στον χρήστη αναζητάει καιρικά δεδομένα. Ο χρήστης αναζητεί καιρικά δεδομένα βάσει μετεωρολογικού σταθμού και είτε βάσει ημερομηνιών είτε βάσει καιρικών δεδομένων είτε και των δύο μαζί.	
<b>Trigger:</b> Το portal να λειτουργεί εύρυθμα και με καιρικά δεδομένα.	
<b>Relationships:</b> Συσχέτιση (association) : Περιλαμβάνει (include) : Προβολή, Εύρεση Επεκτείνει (extend) : Εξειδικεύει (generalization) :	
<b>Normal Flow of Events:</b> 1. Επιλογή ταμπλό. 2. Επιλογή ημερομηνιών. 3. Προβολή γραφημάτων.	
<b>Sub-flows:</b>	
<b>Alternate/Exceptional Flows:</b> Οι χρήστες της εφαρμογής Android βλέπουν μόνο τις τελευταίες τιμές.	

Εικόνα 5-3: Σενάριο Προβολής Πληροφοριών

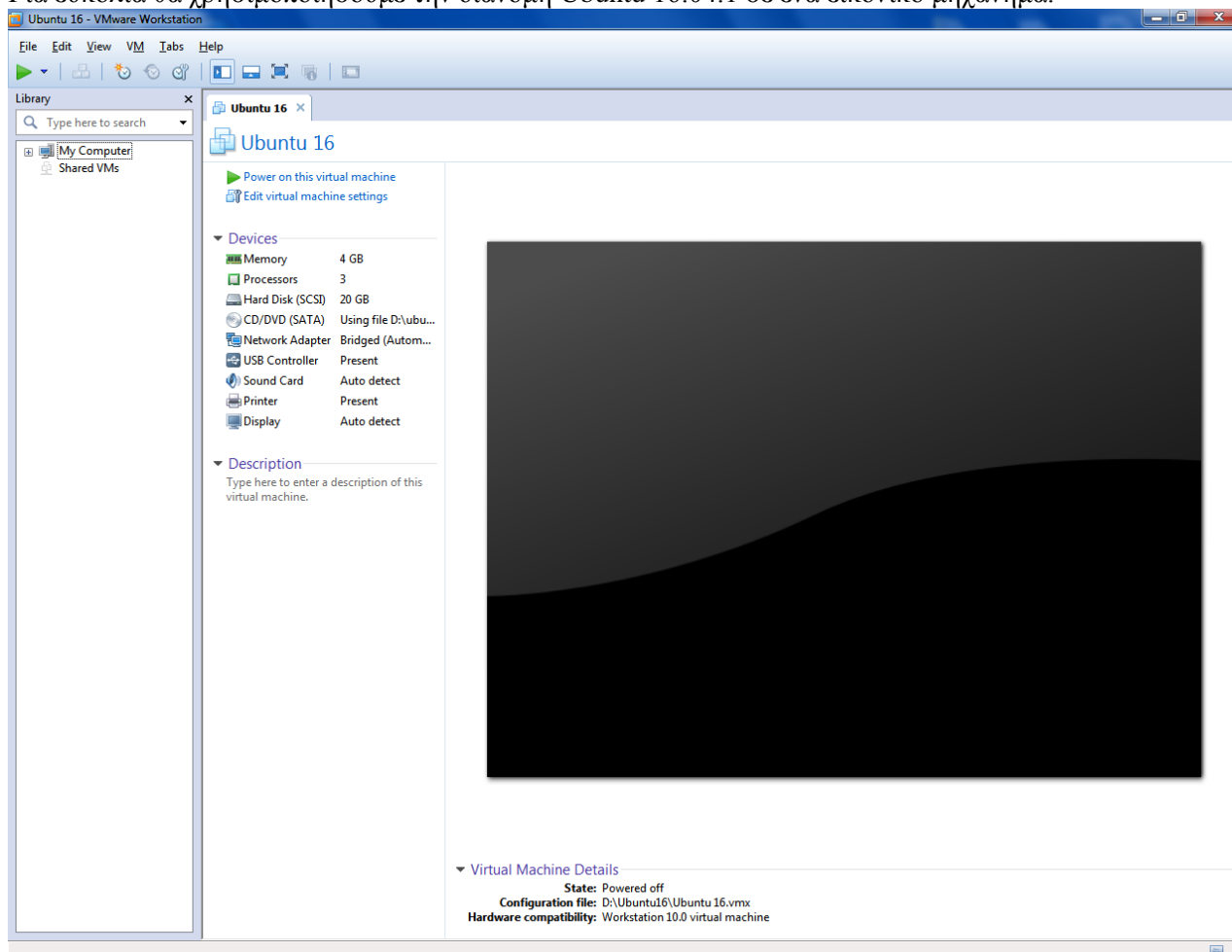


Εικόνα 5-4: Προβολή Γραφημάτων στο Grafana



## 6. Υλοποίηση Portal

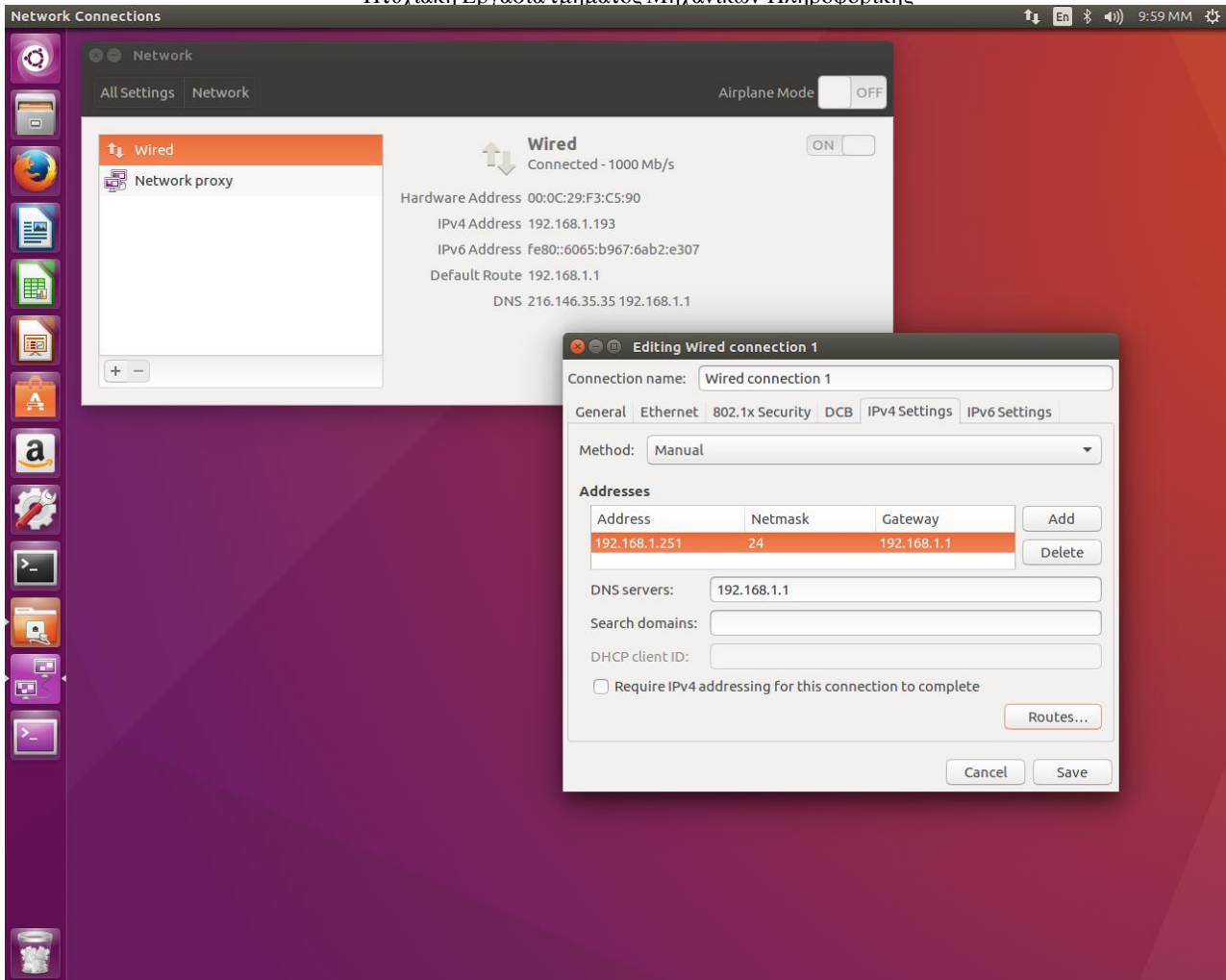
Σε αυτό το κεφάλαιο θα περιγράψουμε αναλυτικά βήμα προς βήμα με εικόνες την υλοποίηση της ηλεκτρονικής πύλης. Η υλοποίηση προϋποθέτει ότι υπάρχει εγκατεστημένο λειτουργικό σύστημα Linux. Για ευκολία θα χρησιμοποιήσουμε την διανομή Ubuntu 16.04.1 σε ένα εικονικό μηχάνημα.



Εικόνα 6-1: Ubuntu 16 virtual machine

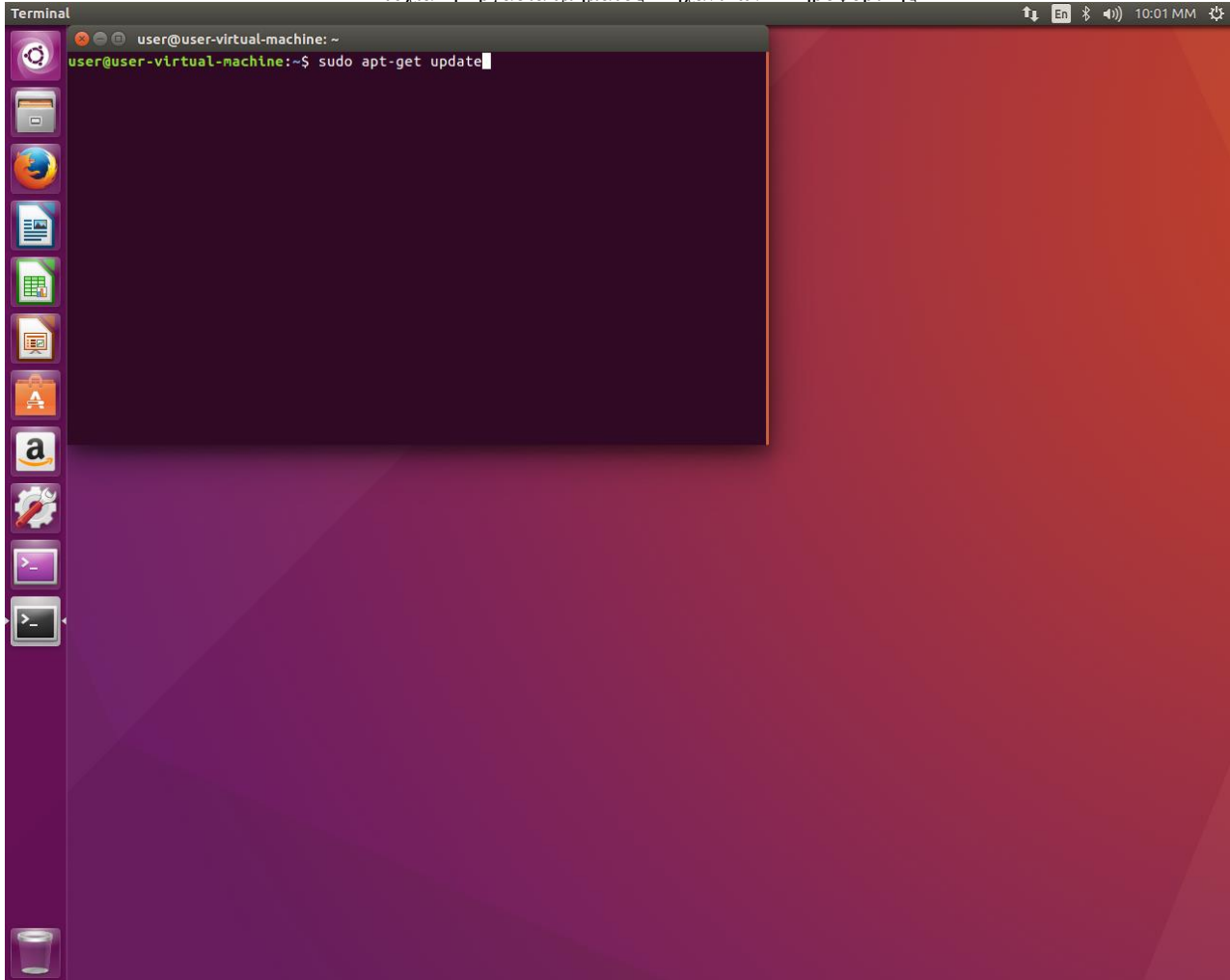
### 6.1. Αρχικοποίηση συστήματος

Το πρώτο βήμα που πρέπει να κάνουμε είναι να εισάγουμε μια στατική διεύθυνση δικτύου στο σύστημα μας. Από την μπάρα αριστερά στην επιφάνεια εργασίας επιλεγούμε το System settings (εικονίδιο με το γρανάζι) και στο νέο παράθυρο που ανοίγει πατάμε στο εικονίδιο με το όνομα Network και ανοίγει ένα νέο υπό-μενού που κάτω δεξιά έχει την επιλογή options, στη οποία κάνουμε ένα αριστερό κλικ. Στο νέο παράθυρο επιλεγούμε την καρτέλα με το όνομα IP v4 settings (εάν έχουμε δίκτυο που υποστηρίζει IP v6 τότε επιλεγούμε εκείνη την καρτέλα) και συμπληρώνουμε μια στατική διεύθυνση όπως παρακάτω.



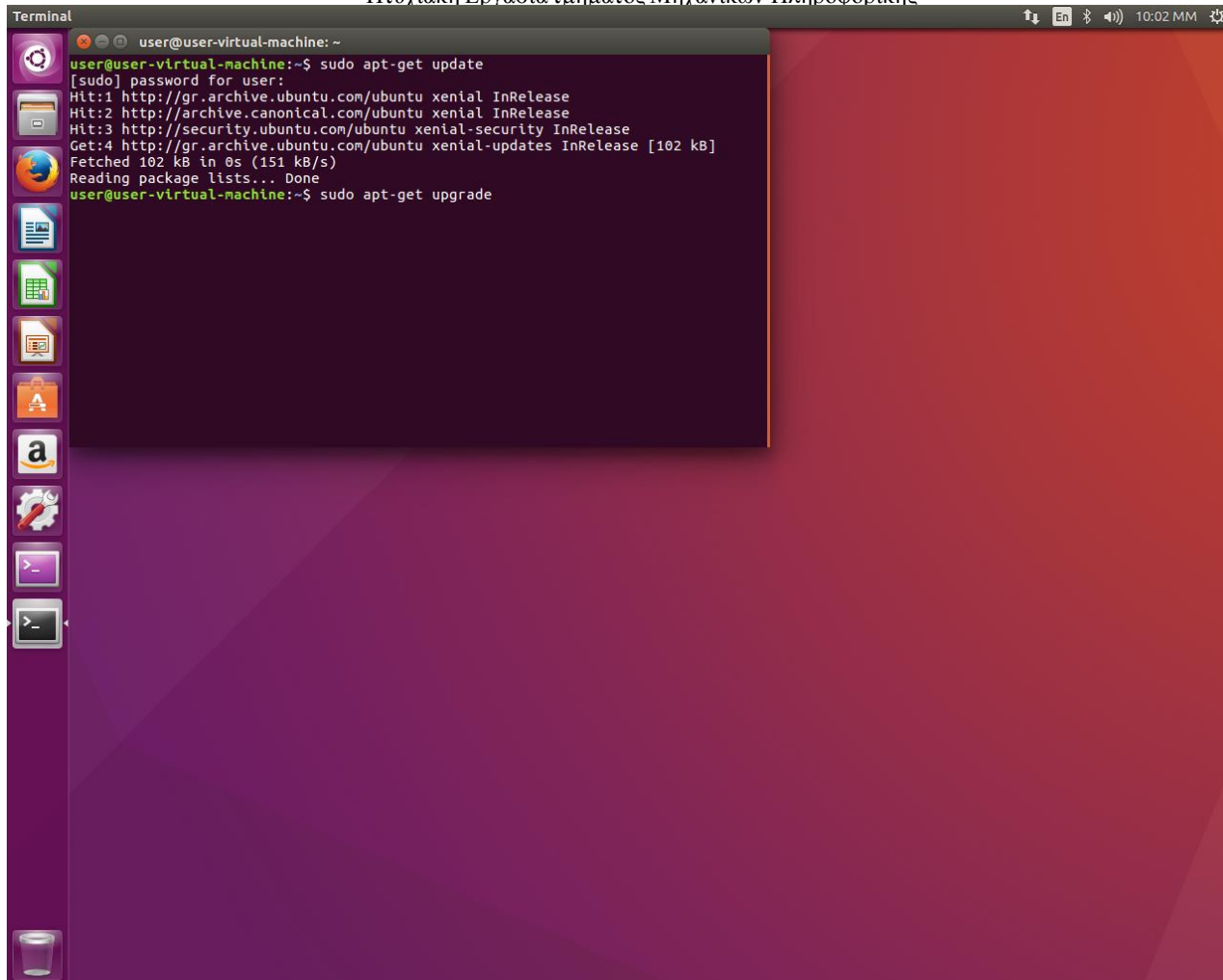
Εικόνα 6-2: Ubuntu static IP

Μετά πατάμε save και κλείνουμε τα παράθυρα. Έπειτα με τον συνδυασμό πλήκτρων CTRL+ALT+T ανοίγει ένα νέο παράθυρο γραμμής εντολών όπου πληκτρολογούμε `sudo apt-get update` και πατάμε enter.



Εικόνα 6-3: Ubuntu system update

Θα ζητήσει κωδικό για τον χρηστή root ή administrator. Εάν δεν έχουμε ορίσει μπορούμε να ορίσουμε ένα με την εντολή `sudo passwd root`. Προσοχή! Αυτόν τον κωδικό πρέπει να τον θυμόμαστε. Μόλις τελειώσει πληκτρολογούμε την εντολή `sudo apt-get upgrade`.



Εικόνα 6-4: Ubuntu upgrade command

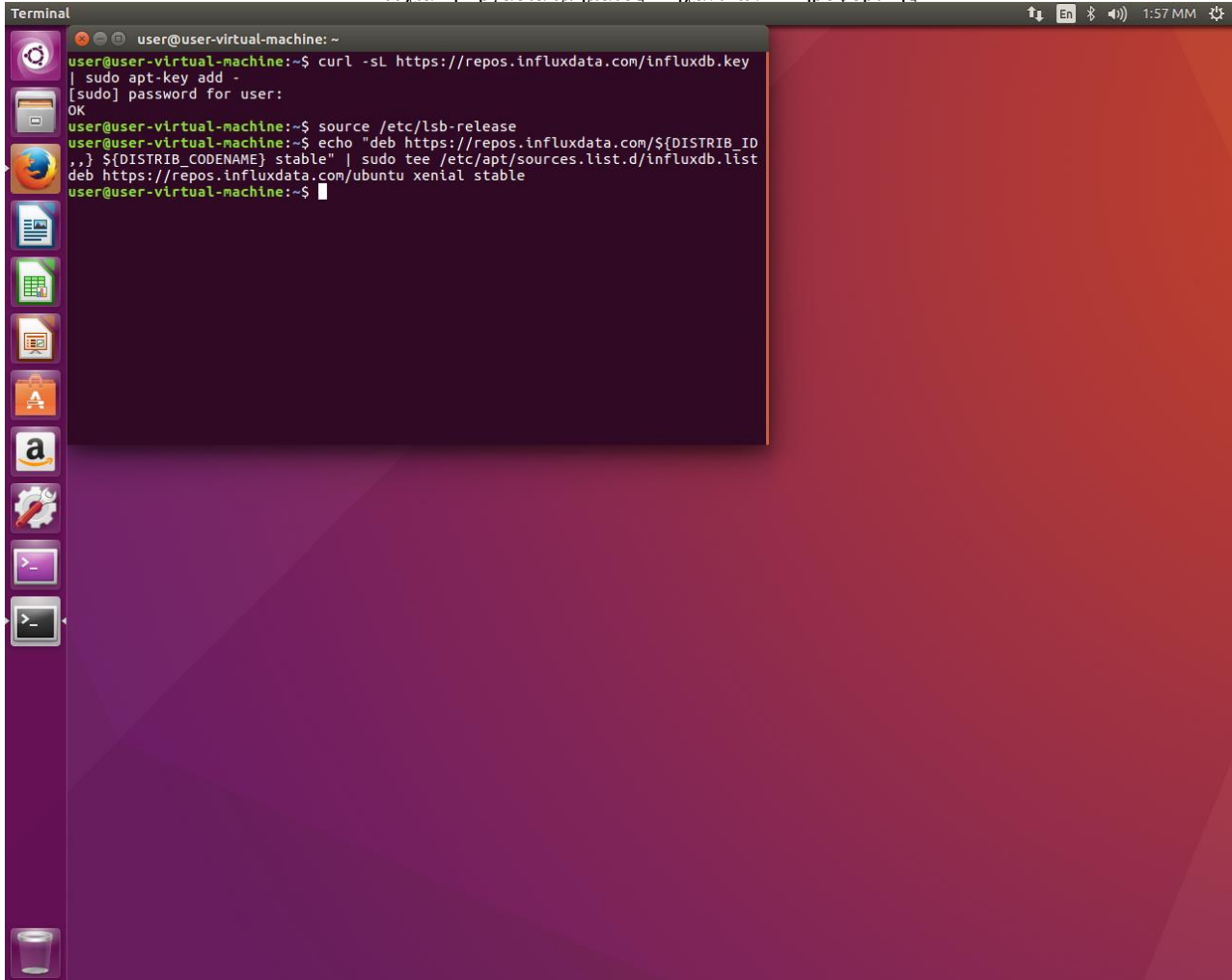
Τώρα που το σύστημα ενημερώθηκε είμαστε έτοιμοι να ξεκινήσουμε την εγκατάσταση των εργαλείων που θα φτιάξουμε το portal.

## 6.2. Εγκατάσταση της Influxdb

Η influxdb είναι βάση δεδομένων του συστήματος που θα αποθηκεύονται τα καιρικά δεδομένα και είναι μέρος του back end συστήματος. Από εκεί θα τα αντλεί το Grafana για τα γραφήματα καθώς και η εφαρμογή για τους κινητούς χρηστές.

Ανοίγουμε ένα νέο παράθυρο Terminal (Ctrl+Alt+T) και πληκτρολογούμε τις παρακάτω εντολές:

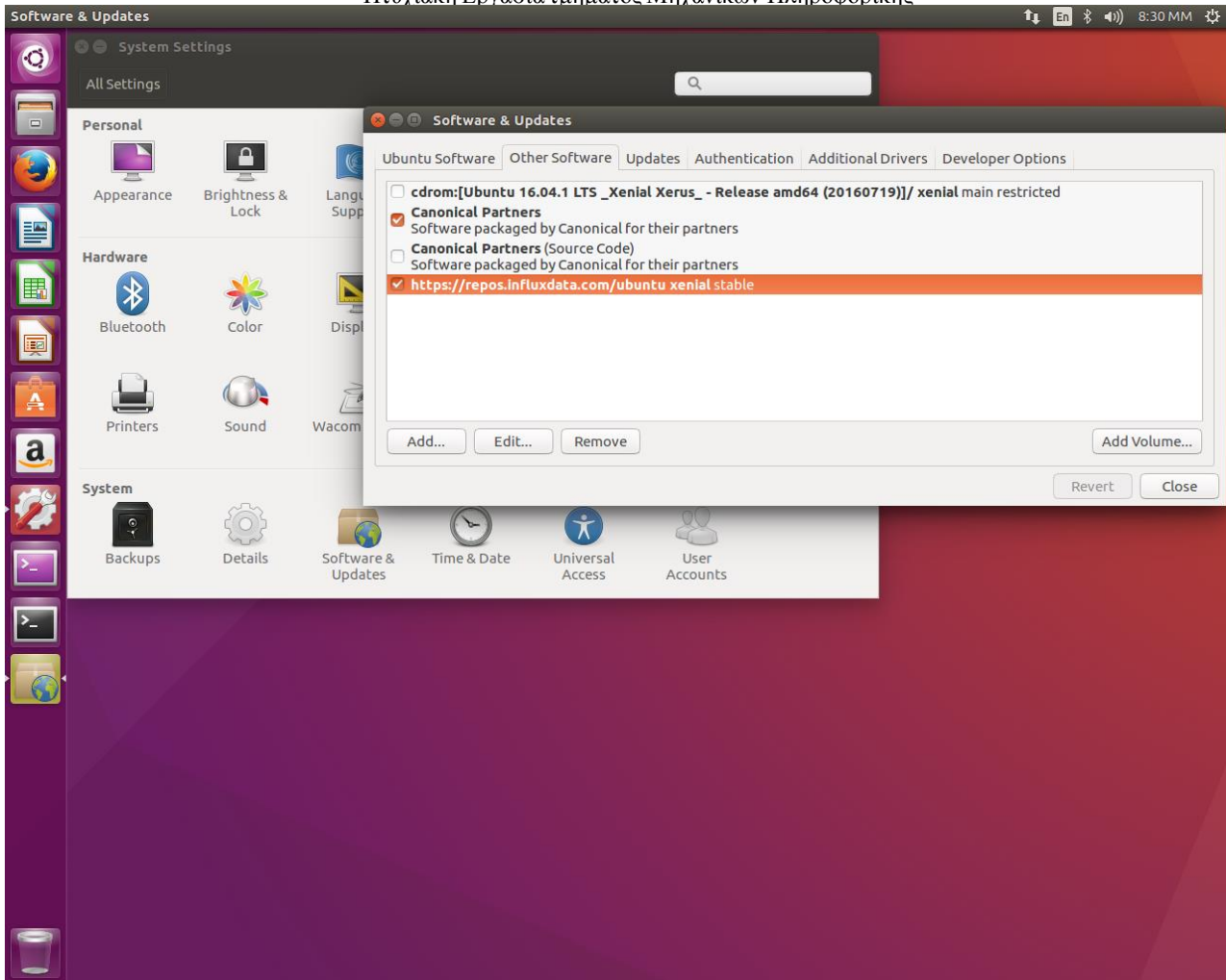
1. `curl -sL https://repos.influxdata.com/influxdb.key | sudo apt-key add -`
2. `source /etc/lsb-release`
3. `echo "deb https://repos.influxdata.com/${DISTRIB_ID,,} ${DISTRIB_CODENAME} stable" | sudo tee /etc/apt/sources.list.d/influxdb.list`

A terminal window on a Linux desktop environment. The terminal shows the following commands and output:

```
user@user-virtual-machine:~  
user@user-virtual-machine:~$ curl -sL https://repos.influxdata.com/influxdb.key  
| sudo apt-key add -  
[sudo] password for user:  
OK  
user@user-virtual-machine:~$ source /etc/lsb-release  
user@user-virtual-machine:~$ echo "deb https://repos.influxdata.com/${DISTRIB_ID  
,,} ${DISTRIB_CODENAME} stable" | sudo tee /etc/apt/sources.list.d/influxdb.list  
deb https://repos.influxdata.com/ubuntu xenial stable  
user@user-virtual-machine:~$
```

The desktop background is a dark red gradient. On the left side, there is a vertical dock with several application icons: Dash, Home, Files, LibreOffice Writer, LibreOffice Calc, LibreOffice Impress, Amazon, Settings, Terminal, and a trash can.

Εικόνα 6-5: Add influxdb to repository

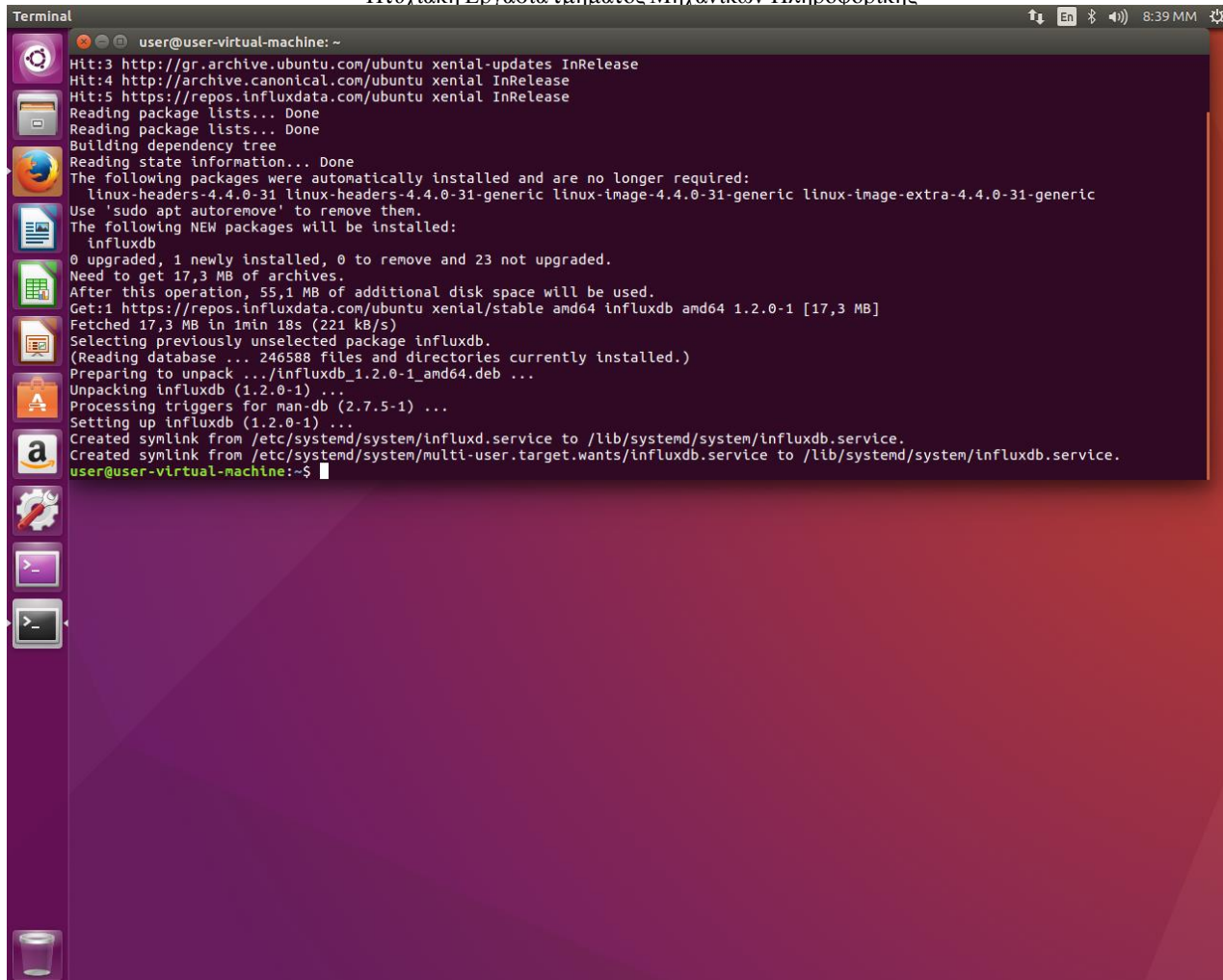


**Εικόνα 6-6: Influxdb repository confirmation**

Με αυτόν τον τρόπο προσθέσαμε την influxdb στο repository του Ubuntu και από εκεί θα κατεβάσουμε την τελευταία έκδοση καθώς και θα κάνει τις ενημερώσεις σε νέες εκδόσεις την βάσης.

Τώρα στο terminal θα πληκτολογίσουμε τις παρακάτω εντολές για κατεβάσουμε και να εγκαταστήσουμε την τελευταία έκδοση (1.2 για τώρα) της influxdb.

```
sudo apt-get update && sudo apt-get install influxdb
```



Εικόνα 6-7: Installed influxdb

Για να εκκινήσουμε την υπηρεσία έχουμε 2 τρόπους:

Εάν έχουμε έκδοση Ubuntu 15.04+ πληκτρολογούμε την παρακάτω εντολή

```
sudo systemctl start influxdb
```

Διαφορετικά

```
sudo service influxdb start
```

Για να δούμε την κατάσταση της υπηρεσίας πληκτρολογούμε:

```
service influxdb status
```

Αφού έχει ξεκινήσει η υπηρεσία επιτυχώς πληκτρολογούμε:

```
influxd config
```

και μας εμφανίζει τις ρυθμίσεις της influxdb.

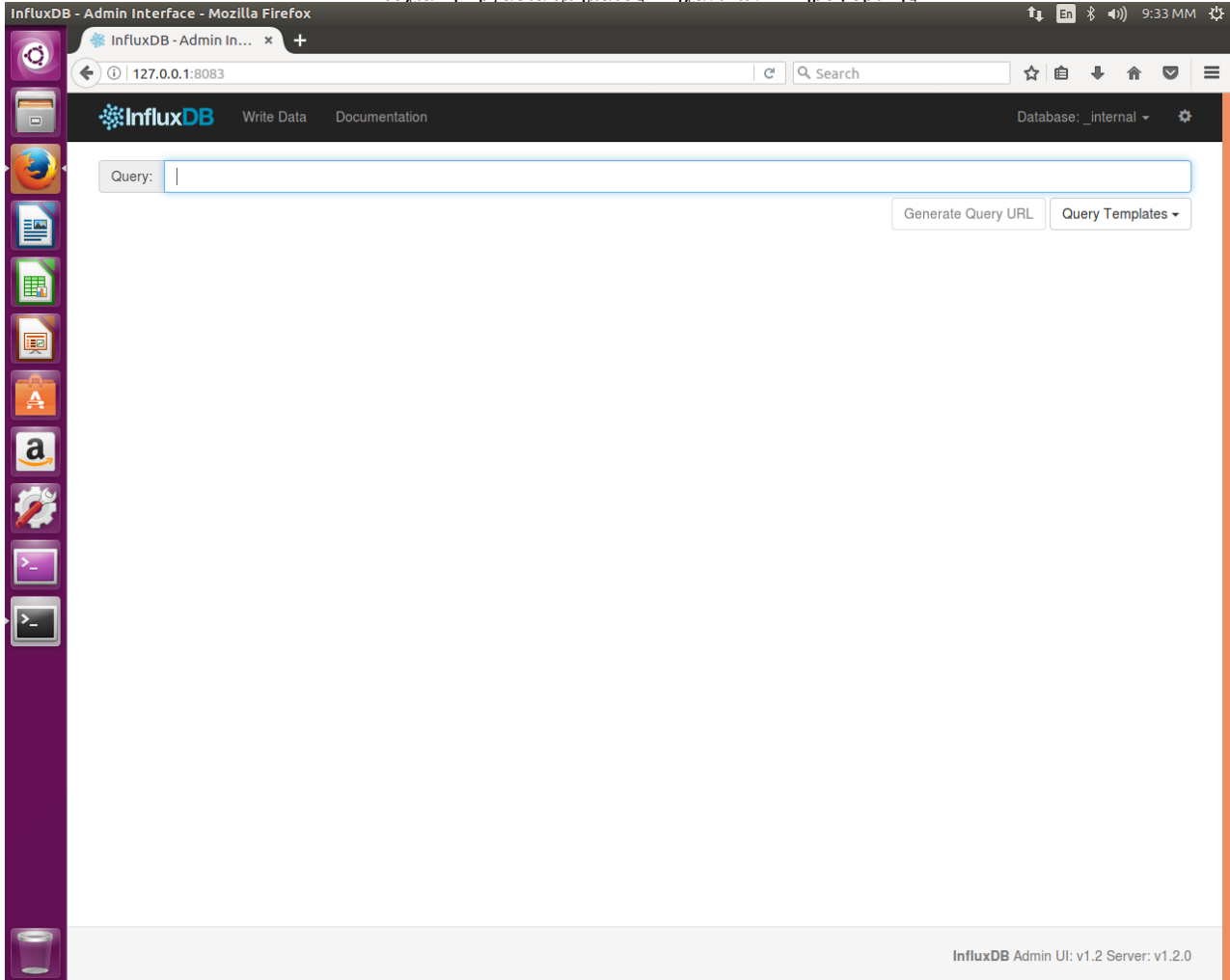
```
Terminal
user@user-virtual-machine:~
Unpacking influxdb (1.2.0-1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up influxdb (1.2.0-1) ...
Created symlink from /etc/systemd/system/influxd.service to /lib/systemd/system/influxdb.service.
Created symlink from /etc/systemd/system/multi-user.target.wants/influxdb.service to /lib/systemd/system/influxdb.service.
user@user-virtual-machine:~$ sudo systemctl start influxdb
user@user-virtual-machine:~$ service influxdb status
● influxdb.service - InfluxDB is an open-source, distributed, time series database
   Loaded: loaded (/lib/systemd/system/influxdb.service; enabled; vendor preset: enabled)
   Active: active (running) since Δευ 2017-02-06 20:47:49 EET; 1min 16s ago
     Docs: https://docs.influxdata.com/influxdb/
    Main PID: 18449 (influxd)
   CGroup: /system.slice/influxdb.service
           └─18449 /usr/bin/influxd -config /etc/influxdb/influxdb.conf

0εβ 06 20:47:55 user-virtual-machine influxd[18449]: [tcp] 2017/02/06 20:47:55 tcp.Mux: handler not registered: 71. Connection from
0εβ 06 20:47:55 user-virtual-machine influxd[18449]: [tcp] 2017/02/06 20:47:55 tcp.Mux: handler not registered: 71. Connection from
0εβ 06 20:47:55 user-virtual-machine influxd[18449]: [tcp] 2017/02/06 20:47:55 tcp.Mux: handler not registered: 71. Connection from
0εβ 06 20:47:55 user-virtual-machine influxd[18449]: [tcp] 2017/02/06 20:47:55 tcp.Mux: handler not registered: 71. Connection from
0εβ 06 20:47:55 user-virtual-machine influxd[18449]: [tcp] 2017/02/06 20:47:55 tcp.Mux: handler not registered: 71. Connection from
0εβ 06 20:47:55 user-virtual-machine influxd[18449]: [tcp] 2017/02/06 20:47:55 tcp.Mux: handler not registered: 71. Connection from
0εβ 06 20:47:55 user-virtual-machine influxd[18449]: [tcp] 2017/02/06 20:47:55 tcp.Mux: handler not registered: 71. Connection from
0εβ 06 20:48:11 user-virtual-machine influxd[18449]: [I] 2017-02-06T18:48:11Z SHOW DATABASES service=query
0εβ 06 20:48:11 user-virtual-machine influxd[18449]: [httpd] 127.0.0.1 - - [06/Feb/2017:20:48:11 +0200] "GET /query?db=&q=SHOW+DATA
user@user-virtual-machine:~$
```

Εικόνα 6-8: Service Influxdb status

Τώρα εάν ανοίξουμε ένα περιηγητή ιστοσελίδων στη διεύθυνση 127.0.0.1:8083 θα πρέπει να φαίνεται το παρακάτω: (Εάν δεν εμφανίζεται, ανατρέξτε στο κεφάλαιο, [9.9 Εισαγωγή δεδομένων στην Influxdb](#) για περισσότερες λεπτομερές)





**Εικόνα 6-9: Influxdb HTTP interface**

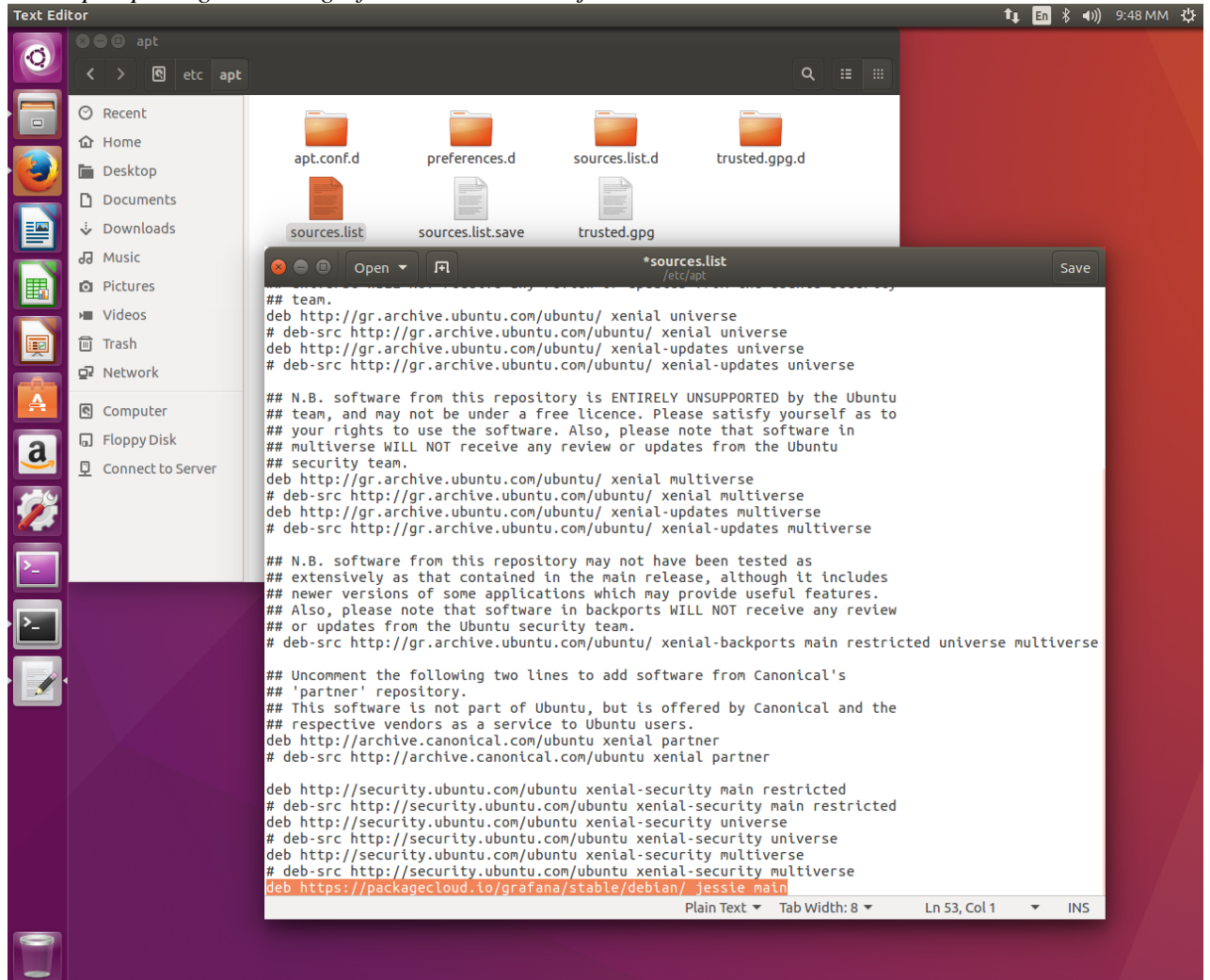
Υπάρχει και δεύτερος τρόπος εγκατάστασης της influxdb. Ανοίγοντας ένα terminal πληκτρολογούμε:

```
wget https://dl.influxdata.com/influxdb/releases/influxdb_1.2.0_amd64.deb  
sudo dpkg -i influxdb_1.2.0_amd64.deb
```

Υπόψη ότι με αυτόν το τρόπο η αναβάθμιση στη τελευταία έκδοση θα πρέπει να γίνεται χειροκίνητα από τον διαχειριστή του συστήματος.

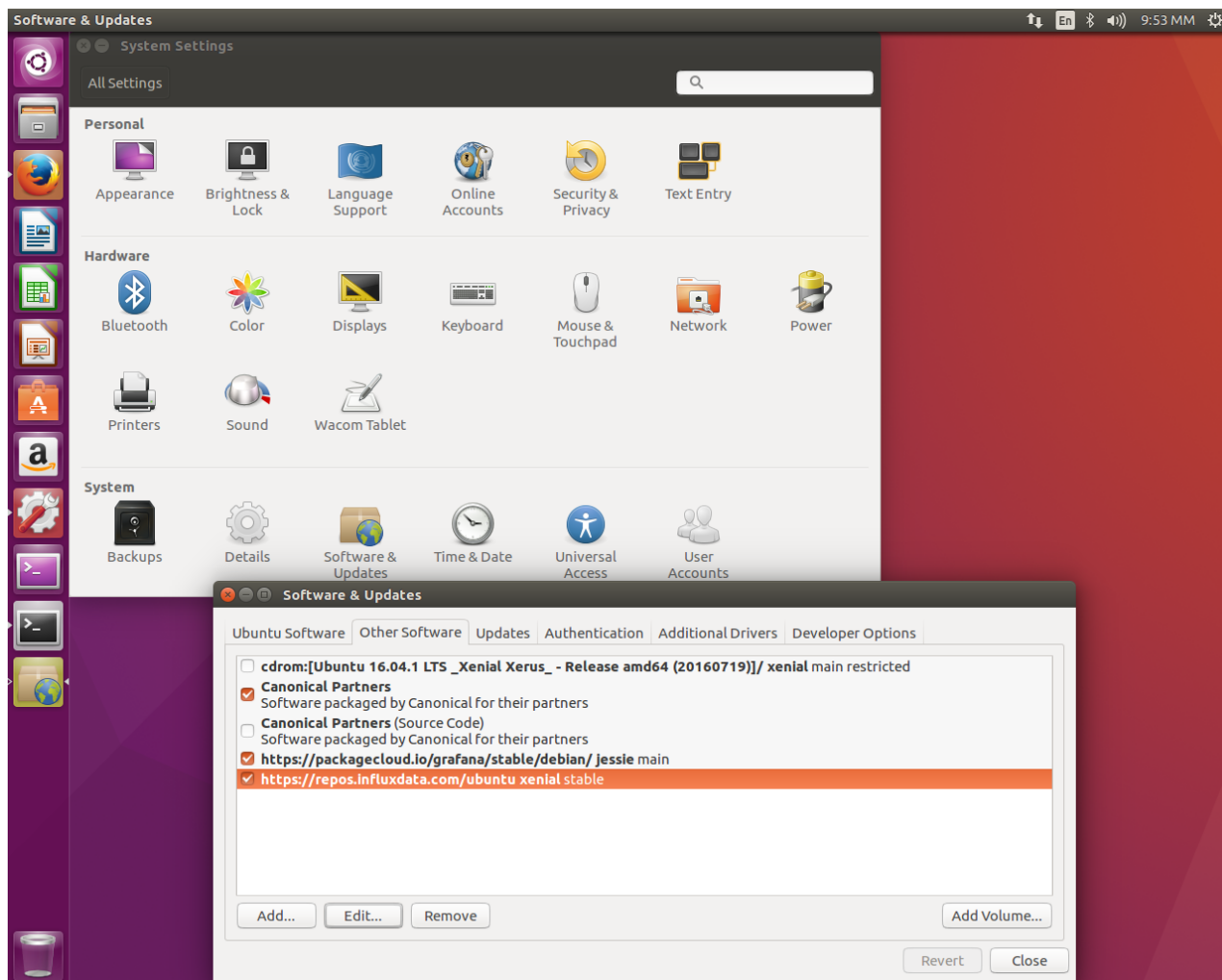
### 6.3. Εγκατάσταση του Grafana

Για την εγκατάσταση του Grafana πρέπει πρώτα να πάμε στη περιοχή /etc/apt και εκεί να βρούμε το αρχείο sources.list και να προσθέσουμε στο τέλος την παρακάτω γραμμή:  
`deb https://packagecloud.io/grafana/stable/debian/ jessie main`



Εικόνα 6-10: Grafana add repository

Και το επιβεβαιώνουμε:



Εικόνα 6-11: Confirm Grafana repository

Τώρα πρέπει να προσθέσουμε το δημόσιο κλειδί του Grafana για μπορούμε να εγκαταστήσουμε τις υπογεγραμμένες ψηφιακά εκδόσεις του. Συνεπώς στο terminal πληκτρολογούμε:

```
curl https://packagecloud.io/gpg.key | sudo apt-key add -
```

Τώρα πληκτρολογούμε τα παρακάτω για το εγκαταστήσουμε και να ξεκινάει στην εκκίνηση :

```
sudo apt-get update
sudo apt-get install grafana
```

Για να πούμε στο σύστημα να ξεκάνει το Grafana κάθε φορά που ξεκινάει και το ίδιο πληκτρολογούμε:

```
systemctl daemon-reload
sudo systemctl enable grafana-server.service
```

Για παλαιότερα συστήματα από την έκδοση 15.04 και κάτω πληκτρολογούμε:

```
sudo update-rc.d grafana-server defaults
```

Για να δούμε την κατάσταση της υπηρεσίας πληκτρολογούμε:

```
systemctl status grafana-server
```

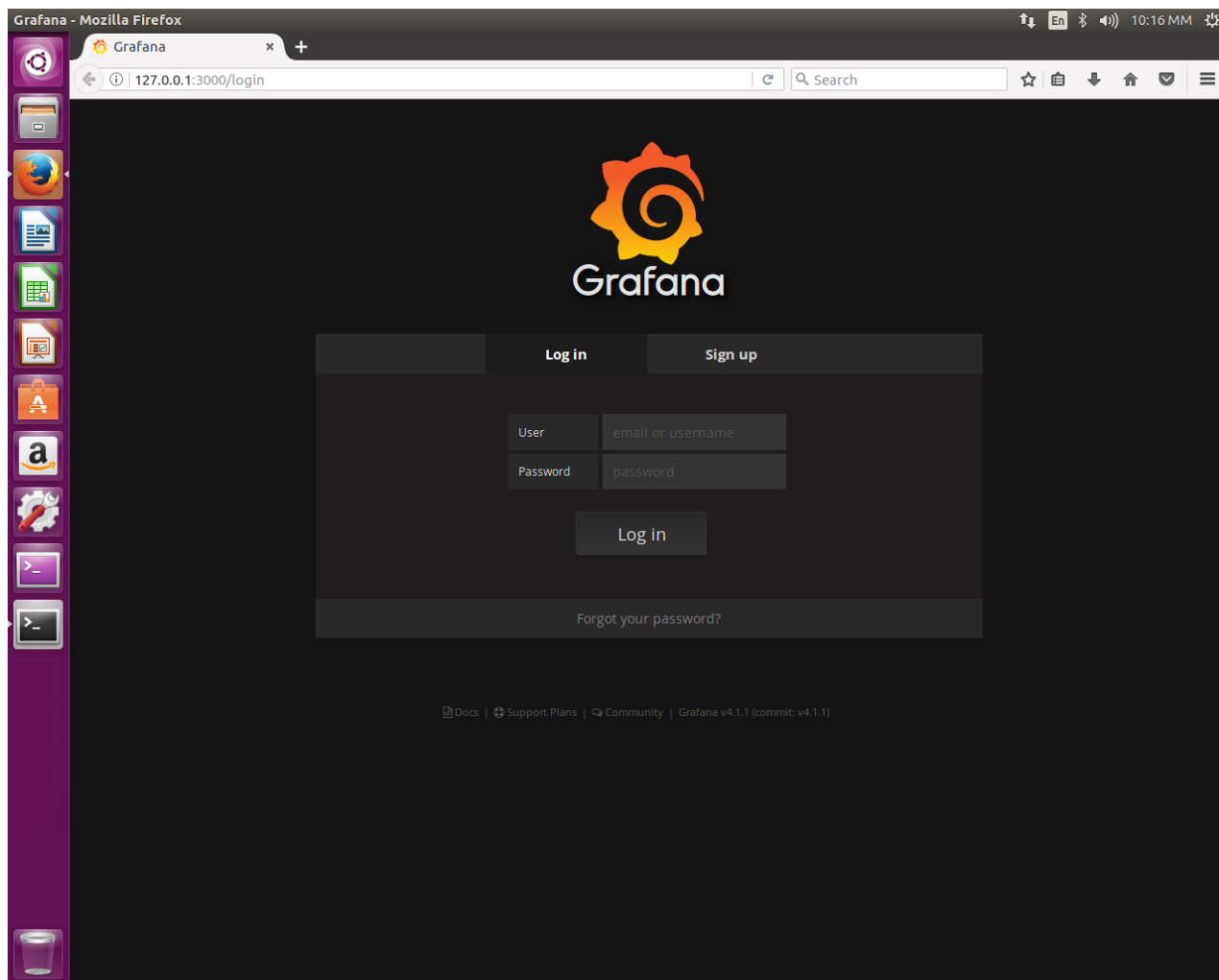
```

Terminal
user@user-virtual-machine: ~
The following NEW packages will be installed:
grafana
0 upgraded, 1 newly installed, 0 to remove and 30 not upgraded.
Need to get 45,2 MB of archives.
After this operation, 132 MB of additional disk space will be used.
Get:1 https://packagecloud.io/grafana/stable/debian jessie/main amd64 grafana amd64 4.1.1-1484211277 [45,2 MB]
Fetched 45,2 MB in 1min 57s (385 kB/s)
Selecting previously unselected package grafana.
(Reading database ... 246614 files and directories currently installed.)
Preparing to unpack .../grafana_4.1.1-1484211277_amd64.deb ...
Unpacking grafana (4.1.1-1484211277) ...
Processing triggers for systemd (229-4ubuntu16) ...
Processing triggers for ureadahead (0.100.0-19) ...
Setting up grafana (4.1.1-1484211277) ...
Adding system user `grafana' (UID 121) ...
Adding new user `grafana' (UID 121) with group `grafana' ...
Not creating home directory `/usr/share/grafana'.
### NOT starting on installation, please execute the following statements to configure grafana to start automatically using syst
end
sudo /bin/systemctl daemon-reload
sudo /bin/systemctl enable grafana-server
### You can start grafana-server by executing
sudo /bin/systemctl start grafana-server
Processing triggers for systemd (229-4ubuntu16) ...
Processing triggers for ureadahead (0.100.0-19) ...
user@user-virtual-machine:~$ systemctl daemon-reload
user@user-virtual-machine:~$ systemctl start grafana-server
user@user-virtual-machine:~$ systemctl status grafana-server
grafana-server.service - Grafana instance
Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; disabled; vendor preset: enabled)
Active: active (running) since Δευ 2017-02-06 22:13:53 EET; 8s ago
Docs: http://docs.grafana.org
Main PID: 4071 (grafana-server)
CGroup: /system.slice/grafana-server.service
└─4071 /usr/sbin/grafana-server --config=/etc/grafana/grafana.ini --pidfile= cfg:default.paths.logs=/var/log/grafana

0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Executing migration" logger=
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Executing migration" logger=
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Executing migration" logger=
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Created default admin user:
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Starting plugin search" logg
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=warn msg="Plugin dir does not exist" l
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Plugin dir created" logger=p
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Initializing CleanUpService"
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Initializing Alerting" logge
0εβ 06 22:13:54 user-virtual-machine grafana-server[4071]: t=2017-02-06T22:13:54+0200 lvl=info msg="Initializing HTTP Server" lo
:1...skipping...
    
```

Εικόνα 6-12: Grafana installation

Για να ελέγξουμε ότι το Grafana τρέχει πληκτρολογούμε σε έναν browser την διεύθυνση 127.0.0.1:3000



Εικόνα 6-13: Grafana login interface

Package details:

- Installs binary to /usr/sbin/grafana-server
- Installs Init.d script to /etc/init.d/grafana-server
- Creates default file (environment vars) to /etc/default/grafana-server
- Installs configuration file to /etc/grafana/grafana.ini
- Installs systemd service (if systemd is available) name grafana-server.service
- The default configuration sets the log file at /var/log/grafana/grafana.log
- The default configuration specifies a sqlite3 db at /var/lib/grafana/grafana.db

Εναλλακτικά μπορούμε με τις παρακάτω εντολές:

```
wget https://grafanarel.s3.amazonaws.com/builds/grafana_4.1.1-1484211277_amd64.deb
sudo apt-get install -y adduser libfontconfig
sudo dpkg -i grafana_4.1.1-1484211277_amd64.deb
sudo systemctl start grafana-server
```

Απλά όπως και με την influxdb οι αναβαθμίσεις θα πρέπει να γίνονται από τον διαχειριστή του συστήματος.

Το αρχείο των ρυθμίσεων βρίσκεται στη διαδρομή /etc/grafana/grafana.ini

## 6.4. Εγκατάσταση της Java

Αν και η διανομή Ubuntu προσφέρει την Java για εγκατάσταση, δεν προσφέρει την κατάλληλη έκδοση για ανάπτυξη της εφαρμογής στο Android Studio που θα εγκαταστήσουμε αργότερα αλλά και για το Netbeans που επίσης θα χρειαστούμε αργότερα.

Ανοίγοντας ένα terminal πληκτρολογούμε:

```
sudo add-apt-repository ppa:webupd8team/java
sudo apt-get update
sudo apt-get install oracle-java8-installer
```

```
Terminal
user@user-virtual-machine:~
user@user-virtual-machine:~$ java --version
The program 'java' can be found in the following packages:
* default-jre
* gcj-5-jre-headless
* openjdk-8-jre-headless
* gcj-4.8-jre-headless
* gcj-4.9-jre-headless
* openjdk-9-jre-headless
Try: sudo apt install <selected package>
user@user-virtual-machine:~$ sudo add-apt-repository ppa:webupd8team/java
Oracle Java (JDK) Installer (automatically downloads and installs Oracle JDK7 / JDK8 / JDK9). There are no actual Java files in this PPA.

More info (and Ubuntu installation instructions):
- for Oracle Java 7: http://www.webupd8.org/2012/01/install-oracle-java-jdk-7-in-ubuntu-via.html
- for Oracle Java 8: http://www.webupd8.org/2012/09/install-oracle-java-8-in-ubuntu-via-ppa.html

Debian installation instructions:
- Oracle Java 7: http://www.webupd8.org/2012/06/how-to-install-oracle-java-7-in-debian.html
- Oracle Java 8: http://www.webupd8.org/2014/03/how-to-install-oracle-java-8-in-debian.html

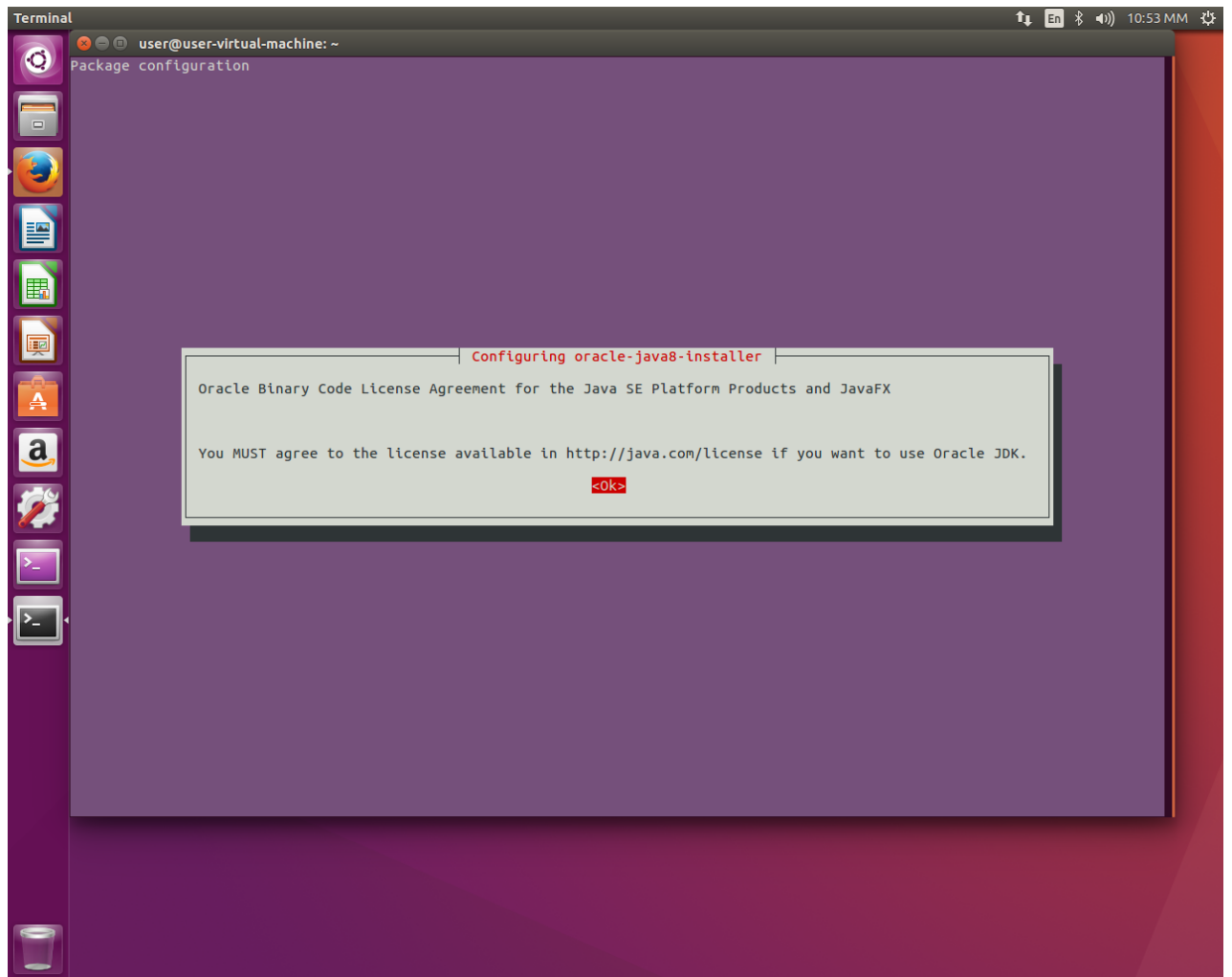
Oracle Java 9 (for both Ubuntu and Debian): http://www.webupd8.org/2015/02/install-oracle-java-9-in-ubuntu-linux.html

For JDK9, the PPA uses standard builds from: https://jdk9.java.net/download/ (and not the Jigsaw builds!).

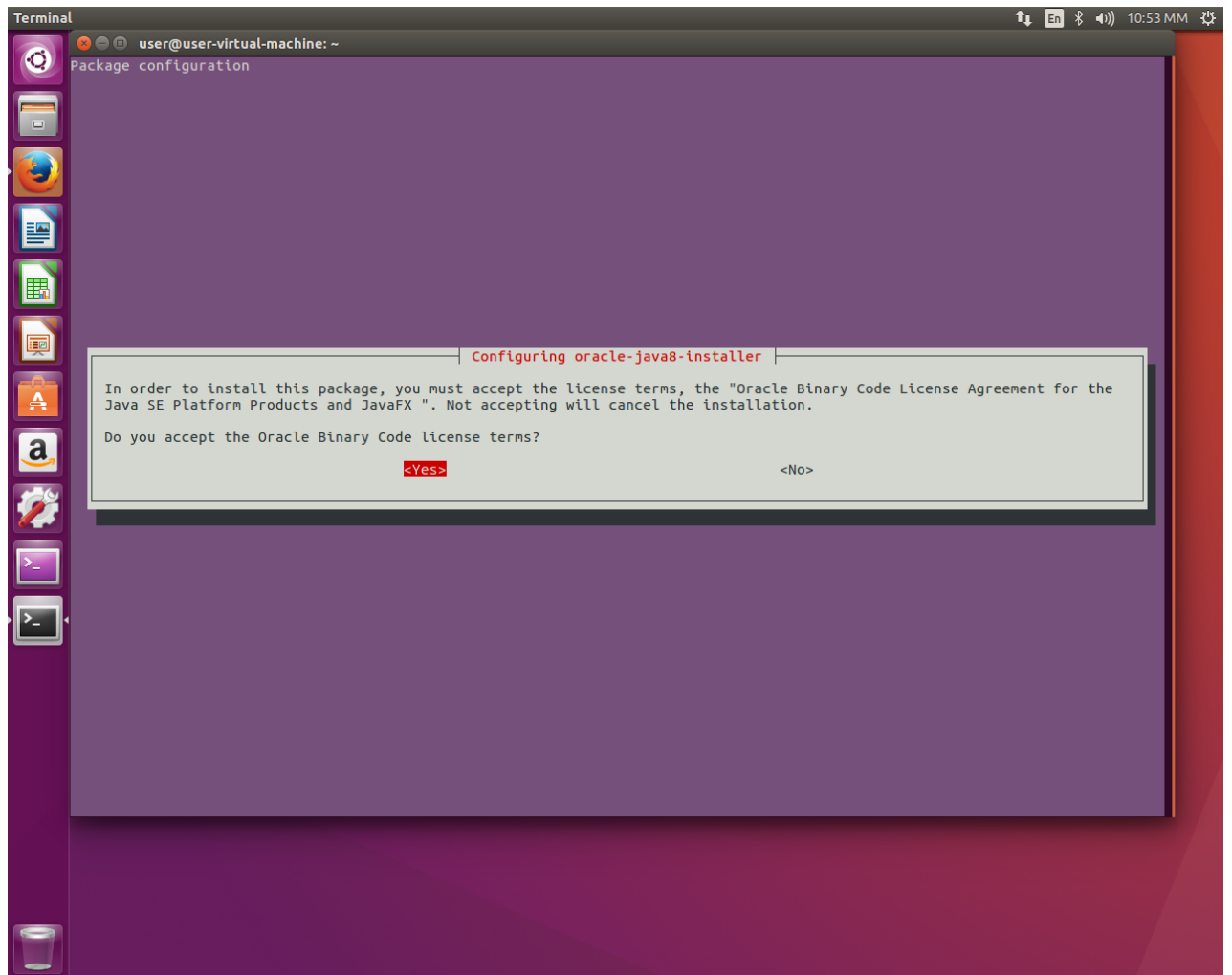
Important!!! For now, you should continue to use Java 8 because Oracle Java 9 is available as an early access release! You should only use Oracle Java 9 if you explicitly need it, because it may contain bugs and it might not include the latest security patches! Also, some Java options were removed in JDK9, so you may encounter issues with various Java apps. More information and installation instructions (Ubuntu / Linux Mint / Debian): http://www.webupd8.org/2015/02/install-oracle-java-9-in-ubuntu-linux.html
More info: https://launchpad.net/~webupd8team/+archive/ubuntu/java
Press [ENTER] to continue or ctrl-c to cancel adding it

gpg: keyring '/tmp/tmpy6fbd0ox/secring.gpg' created
gpg: keyring '/tmp/tmpy6fbd0ox/pubring.gpg' created
gpg: requesting key EEA14886 from hkp server keyserver.ubuntu.com
gpg: /tmp/tmpy6fbd0ox/trustdb.gpg: trustdb created
gpg: key EEA14886: public key "Launchpad VLC" imported
gpg: no ultimately trusted keys found
gpg: Total number processed: 1
gpg:      imported: 1 (RSA: 1)
gpg: OK
user@user-virtual-machine:~$
```

Εικόνα 6-14: Add Oracle repository



Εικόνα 6-15: Accept agreement



Εικόνα 6-16: Accept license



```

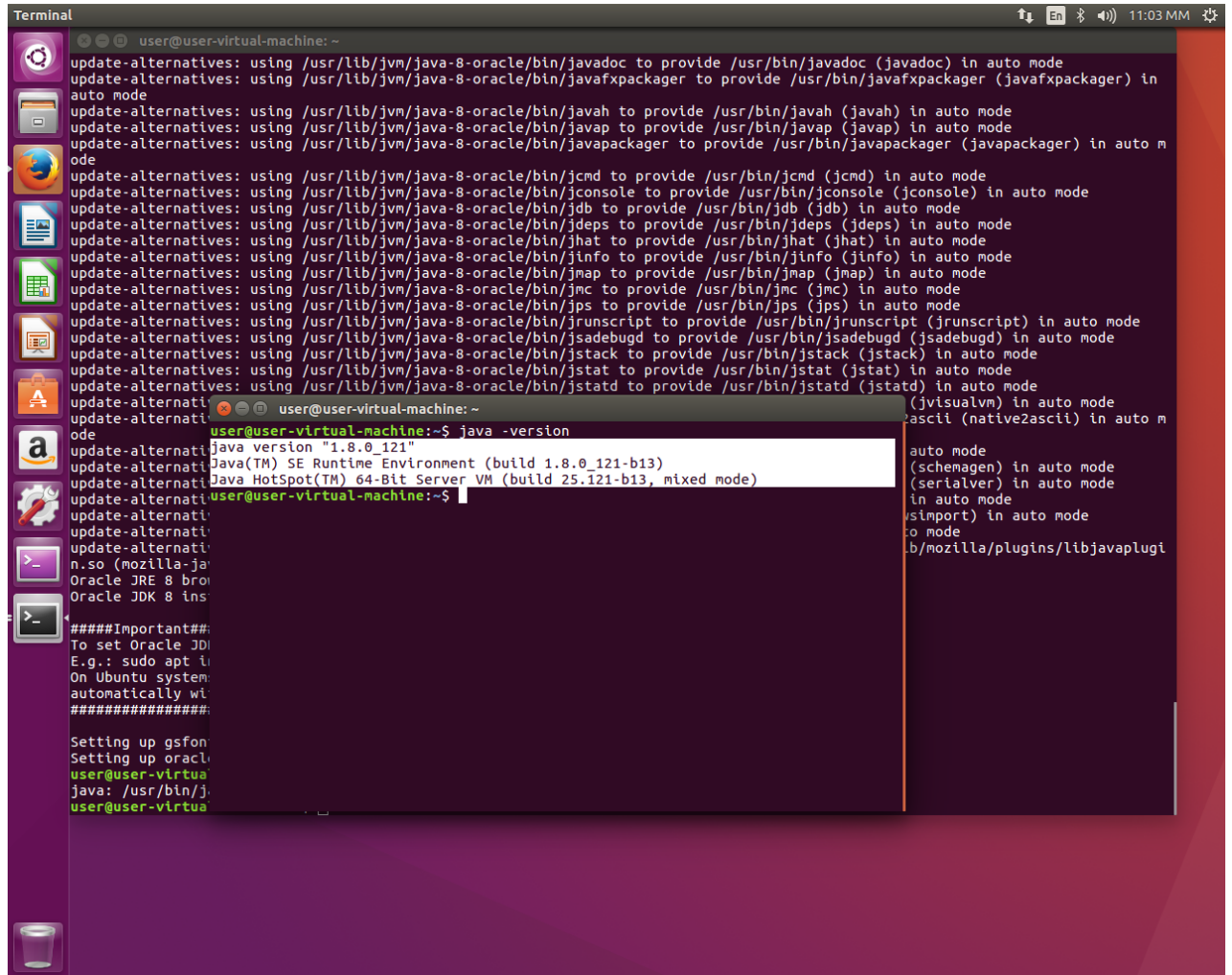
Terminal
user@user-virtual-machine: ~
Preparing to unpack .../oracle-java8-installer_8u121-1-webupd8-0_all.deb ...
Unpacking oracle-java8-installer (8u121-1-webupd8-0) ...
Selecting previously unselected package gsfon
Preparing to unpack .../gsfonts-x11_0.24_all.deb ...
Unpacking gsfon
Selecting previously unselected package oracle-java8-set-default.
Preparing to unpack .../oracle-java8-set-default_8u121-1-webupd8-0_all.deb ...
Unpacking oracle-java8-set-default (8u121-1-webupd8-0) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for shared-mime-info (1.5-2ubuntu0.1) ...
Processing triggers for bamfdaemon (0.5.3-bzr0+16.04.20160824-0ubuntu1) ...
Rebuilding /usr/share/applications/bamf-2.index...
Processing triggers for gnome-menus (3.13.3-6ubuntu3.1) ...
Processing triggers for desktop-file-utils (0.22-1ubuntu5) ...
Processing triggers for mime-support (3.59ubuntu1) ...
Processing triggers for fontconfig (2.11.94-0ubuntu1.1) ...
Setting up java-common (0.56ubuntu2) ...
Setting up oracle-java8-installer (8u121-1-webupd8-0) ...
No /var/cache/oracle-jdk8-installer/wgetrc file found.
Creating /var/cache/oracle-jdk8-installer/wgetrc and
using default oracle-java8-installer wgetrc settings for it.
Downloading Oracle Java 8...
--2017-02-06 22:53:38-- http://download.oracle.com/otn-pub/java/jdk/8u121-b13/e9e7ea248e2c4826b92b3f075a80e441/jdk-8u121-linux-
x64.tar.gz
Resolving download.oracle.com (download.oracle.com)... 62.169.226.23, 62.169.226.24
Connecting to download.oracle.com (download.oracle.com)|62.169.226.23|:80... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://edelivery.oracle.com/otn-pub/java/jdk/8u121-b13/e9e7ea248e2c4826b92b3f075a80e441/jdk-8u121-linux-x64.tar.gz [f
ollowing]
--2017-02-06 22:53:39-- https://edelivery.oracle.com/otn-pub/java/jdk/8u121-b13/e9e7ea248e2c4826b92b3f075a80e441/jdk-8u121-linu
x-x64.tar.gz
Resolving edelivery.oracle.com (edelivery.oracle.com)... 2.18.84.152, 2001:41a8:806:284::2d3e, 2001:41a8:806:280::2d3e
Connecting to edelivery.oracle.com (edelivery.oracle.com)|2.18.84.152|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: http://download.oracle.com/otn-pub/java/jdk/8u121-b13/e9e7ea248e2c4826b92b3f075a80e441/jdk-8u121-linux-x64.tar.gz?Auth
Param=1486414539_9eef36e2afbba3e085f17a7eb7938928 [following]
--2017-02-06 22:53:39-- http://download.oracle.com/otn-pub/java/jdk/8u121-b13/e9e7ea248e2c4826b92b3f075a80e441/jdk-8u121-linux-
x64.tar.gz?AuthParam=1486414539_9eef36e2afbba3e085f17a7eb7938928
Connecting to download.oracle.com (download.oracle.com)|62.169.226.23|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 183246769 (175M) [application/x-gzip]
Saving to: 'jdk-8u121-linux-x64.tar.gz'

 0K ..... 1% 932K 3m9s
3072K ..... 3% 951K 3m4s
6144K ..... 5% 783K 3m12s
9216K ..... 6% 952K 3m5s
    
```

Εικόνα 6-17: Downloading Java 8 JDK

Αφού αποδεχτούμε όλα όσα ζητεί ξεκάνει το κατέβασμα και η εγκατάσταση. Όταν τελειώσει η εγκατάσταση, την επιβεβαιώνουμε πληκτρολογώντας:

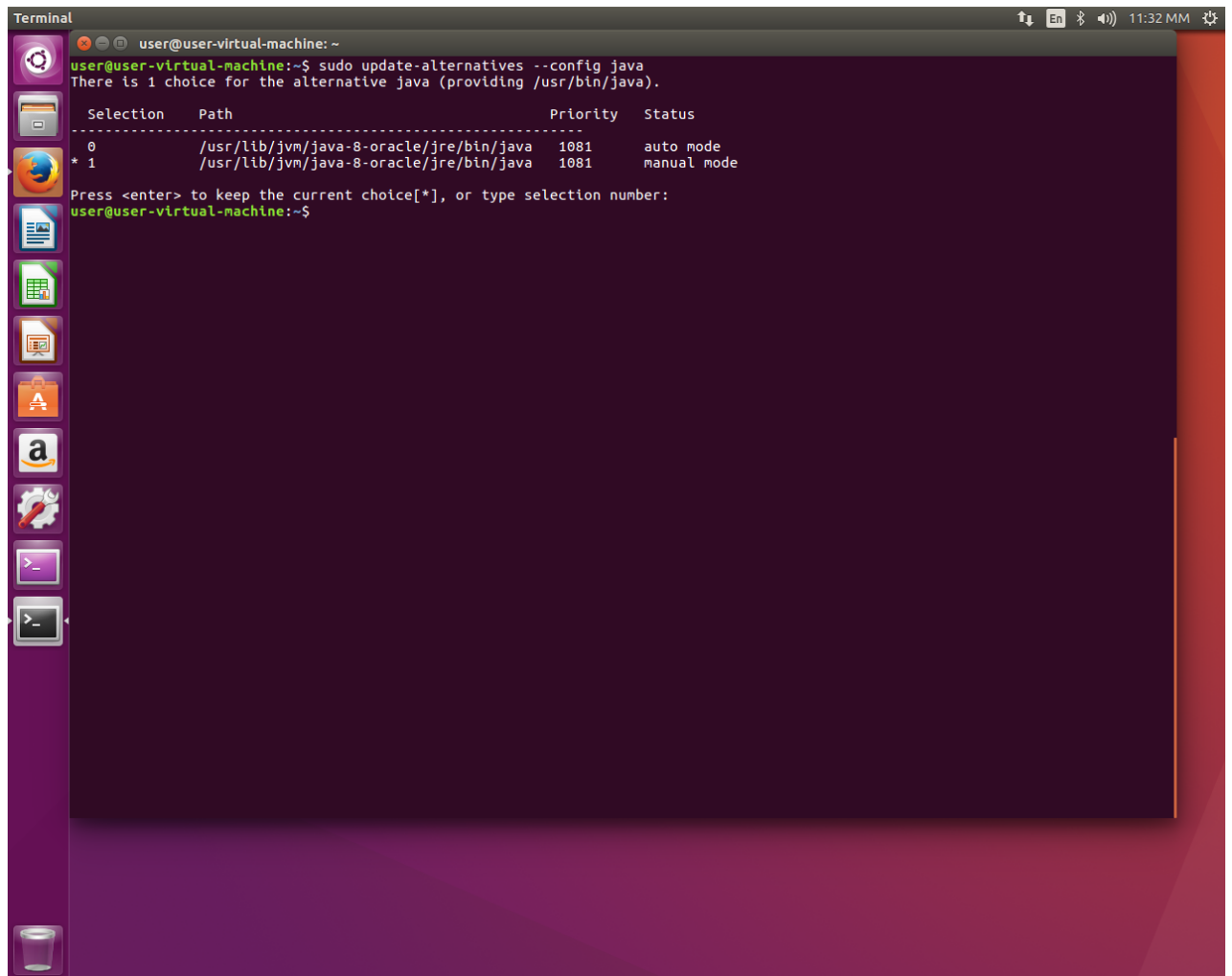
```
java -version
```



Εικόνα 6-18: Java version

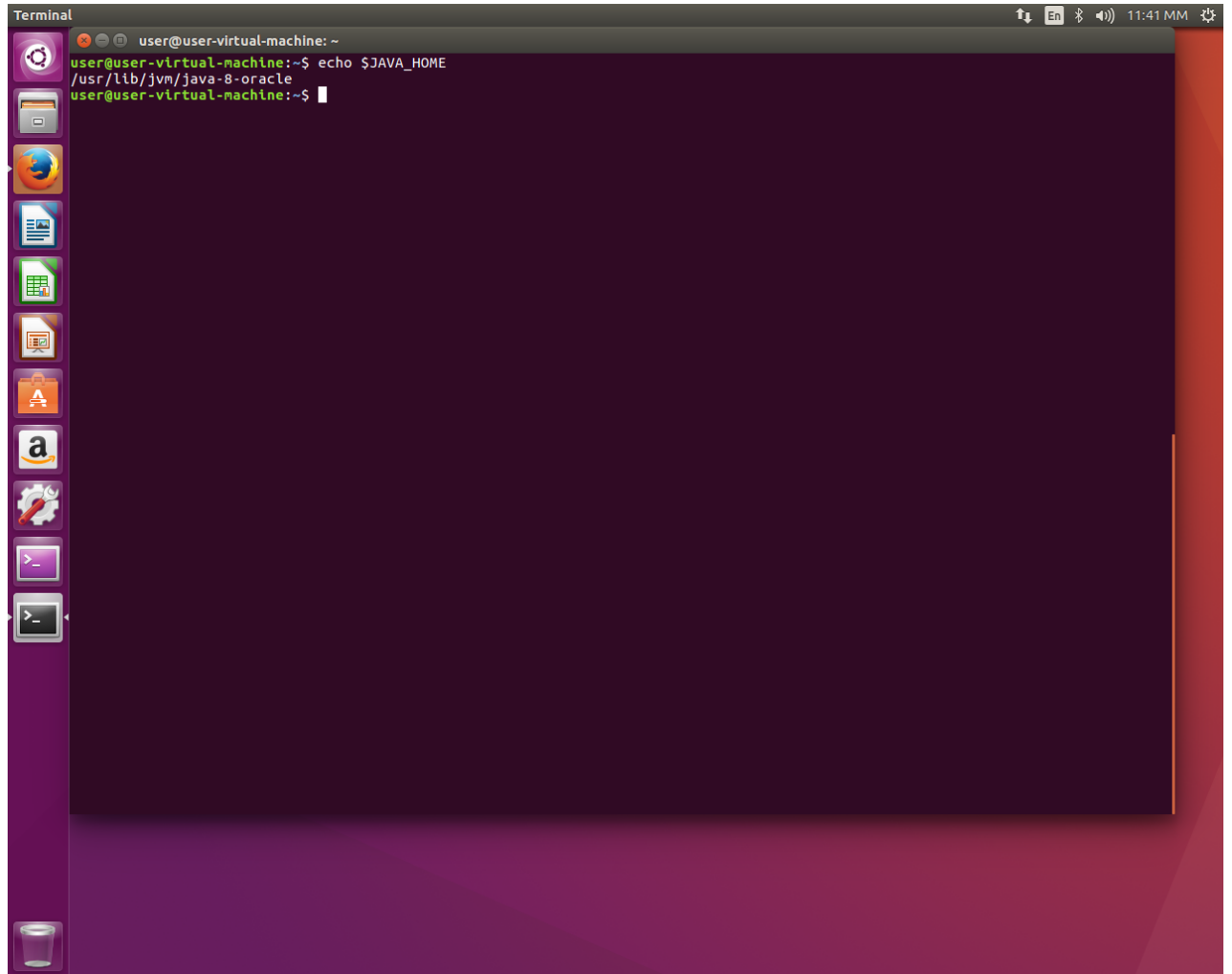
Αυτό που πρέπει να κάνουμε είναι να ορίσουμε την java 8 ως προεπιλεγμένη και να την προσθέσουμε στο μονοπάτι(path) των μεταβλητών περιβάλλοντος. Πληκτρολογούμε την παρακάτω εντολή:

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



Εικόνα 6-19: Java path

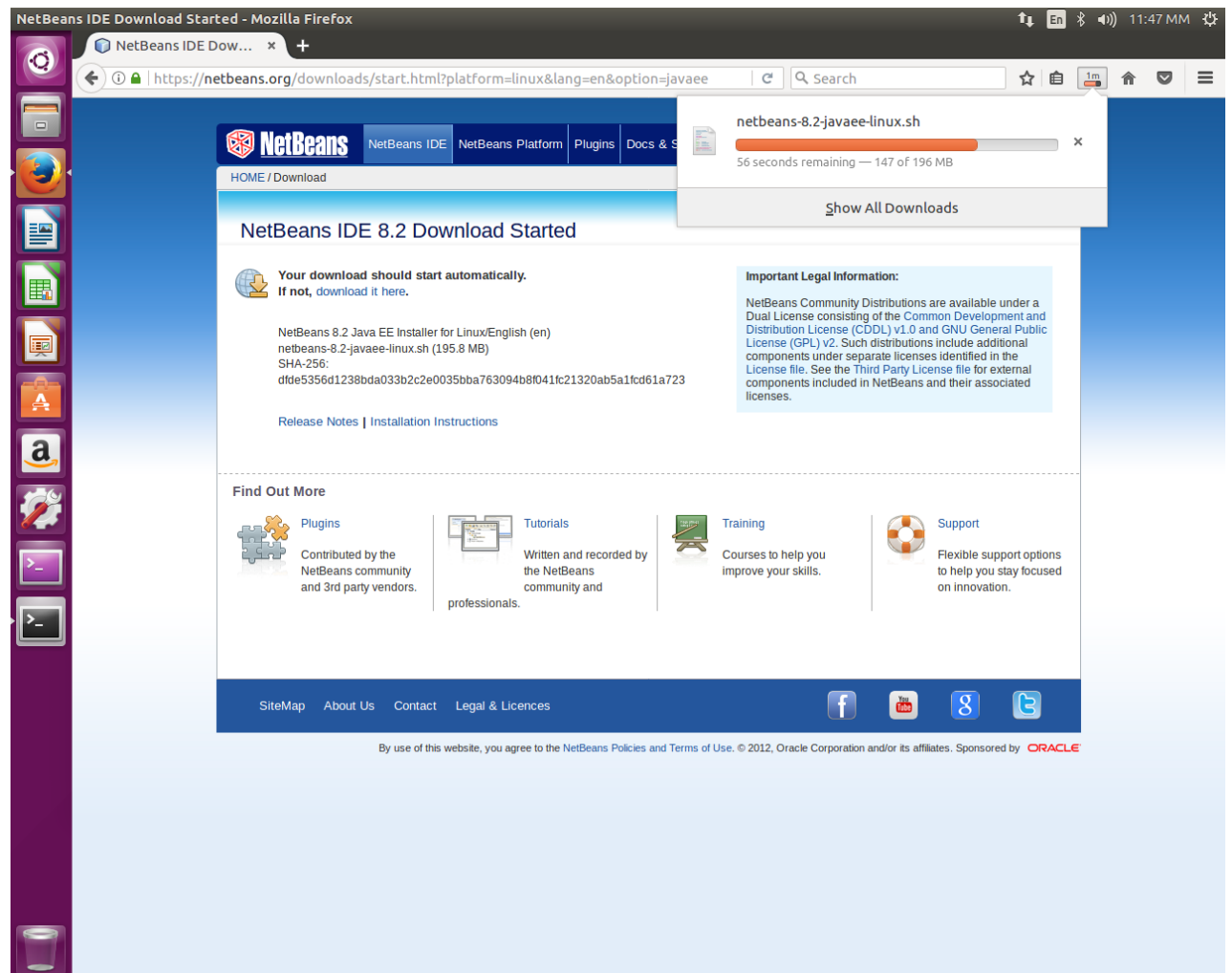
Αφού βρούμε που είναι η διαδρομή της java πληκτρολογούμε:  
`sudo nano /etc/environment`  
και στο τέλος του αρχείου προσθέτουμε  
`JAVA_HOME="/usr/lib/jvm/java-8-oracle"` και αποθηκεύουμε.  
Μετά πληκτρολογούμε:  
`source /etc/environment`  
και επιβεβαιώνουμε με:  
`echo $JAVA_HOME`



Εικόνα 6-20: Java home

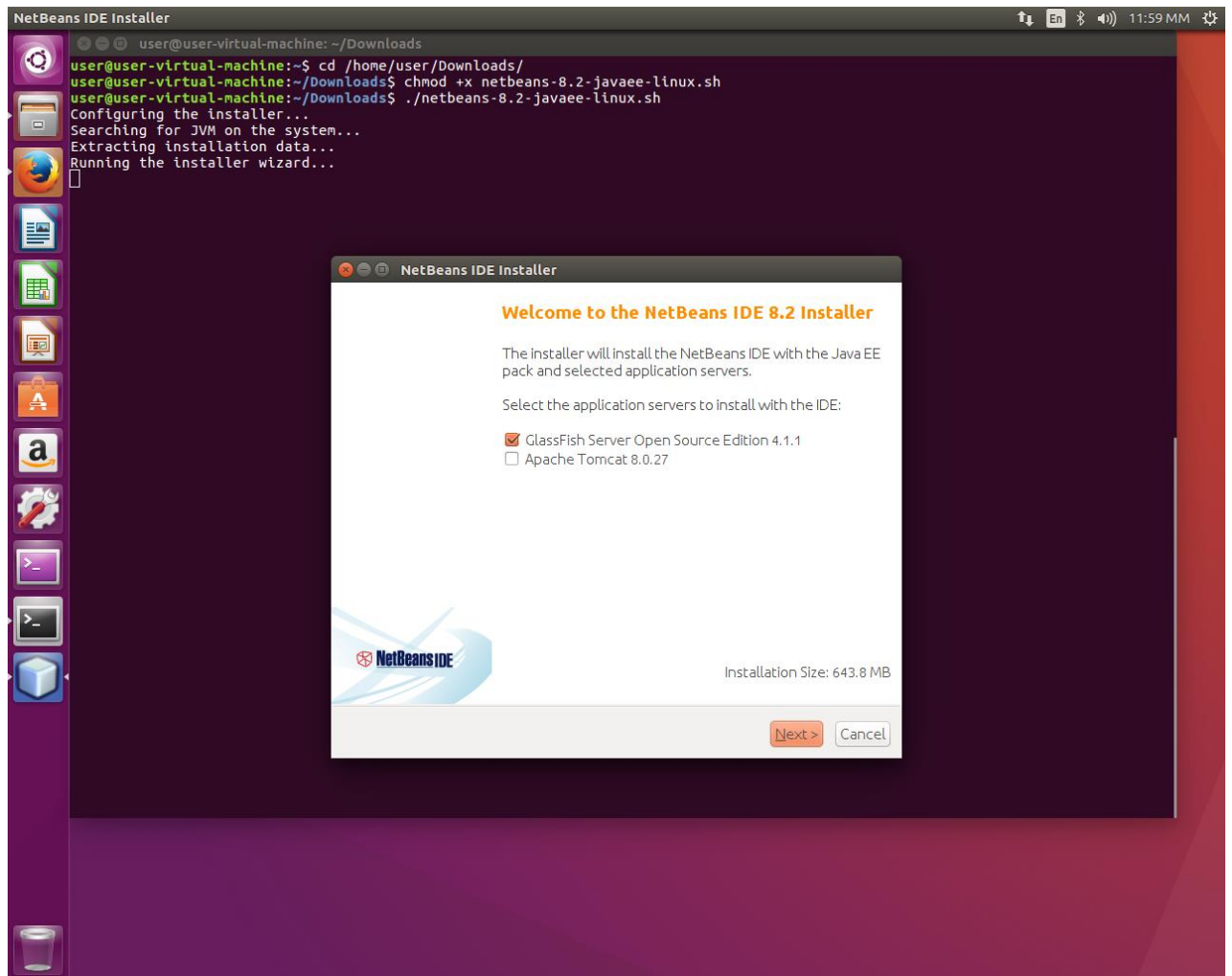
## 6.5. Εγκατάσταση Netbeans

Το Netbeans χρειαζόμαστε για να μετατρέψουμε τα δεδομένα από το βιοκλίμα που είναι σε μορφή csv σε αντίστοιχη κατάλληλη μορφή για την influxdb. Για να εγκαταστήσουμε το Netbeans πληκτρολογούμε τη διεύθυνση <https://netbeans.org/downloads> και κατεβάζουμε την διανομή Java EE.

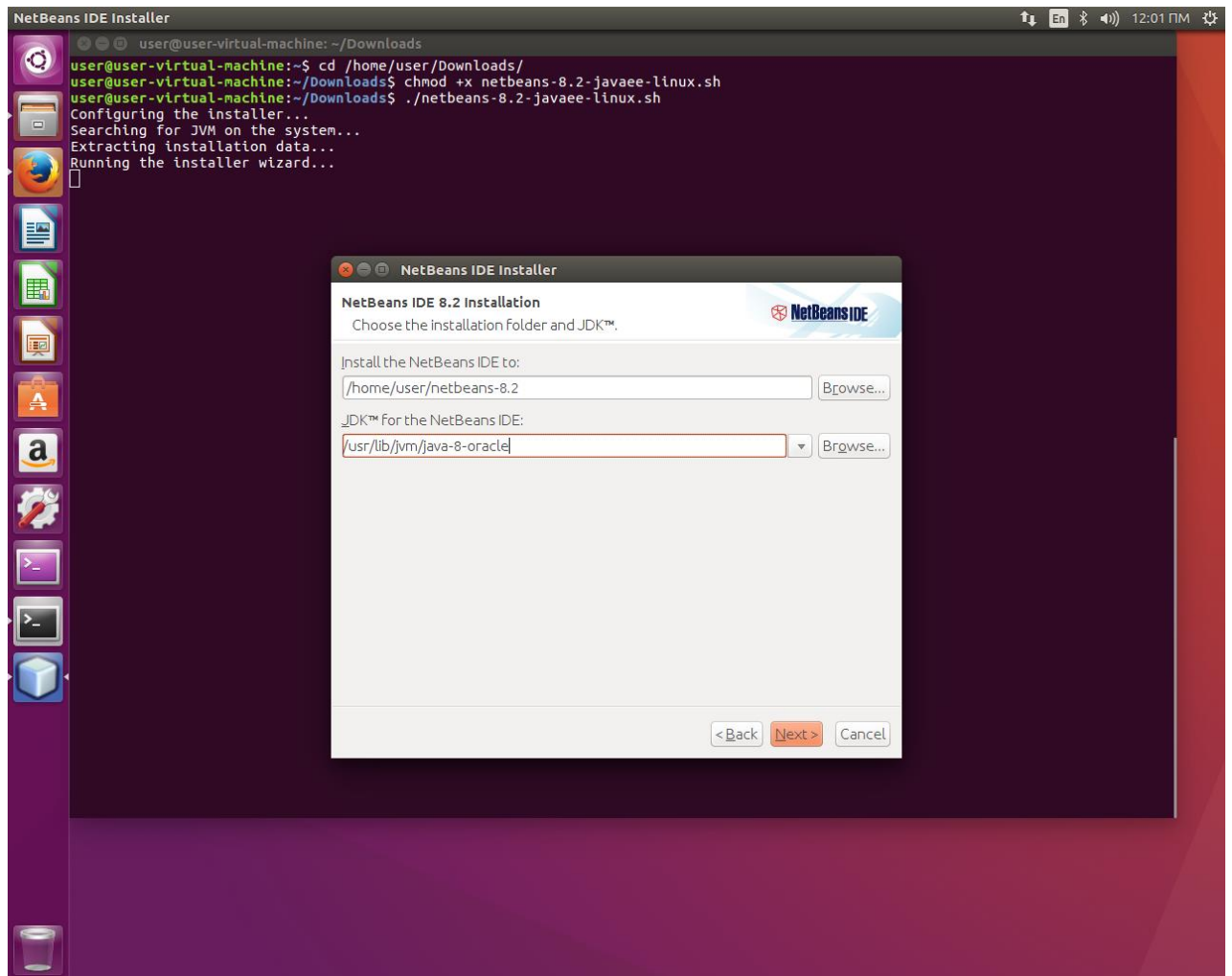


Εικόνα 6-21: Netbeans Download

Μόλις κατέβει, από το terminal πληκτρολογούμε `cd /home/user/Downloads` και μετά `chmod +x <το όνομα του αρχείου που κατεβάσαμε>` δηλαδή `chmod +x netbeans-8.2-javaee-linux.sh` και μετά `./ netbeans-8.2-javaee-linux.sh` για το τρέξουμε.

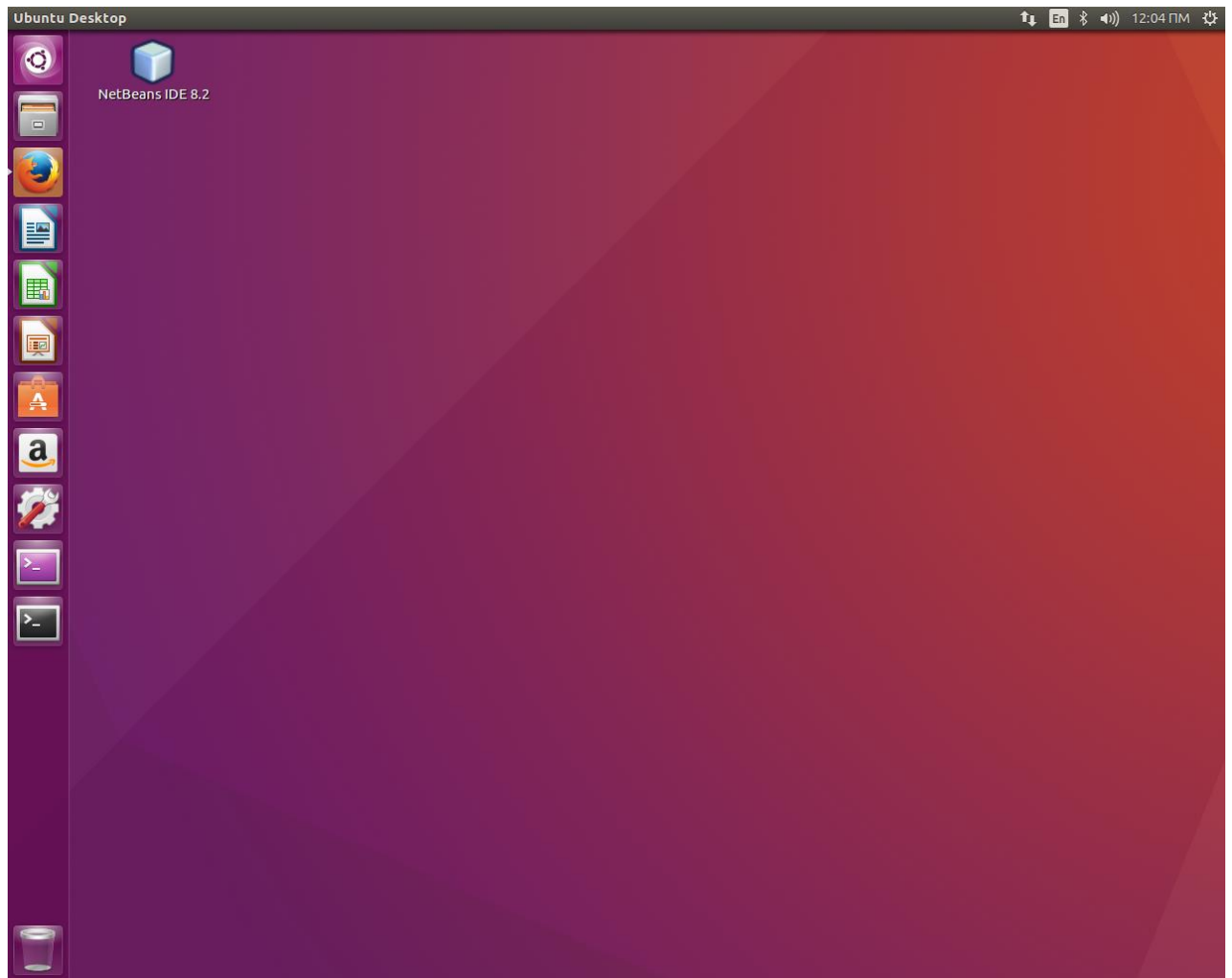


Εικόνα 6-22: Installation wizard



Εικόνα 6-23: Netbeans Java specified

Προσοχή μόνο, όταν ζητήσει το κατάλογο εγκατάστασης της java να του δώσουμε τον σωστό.



**Εικόνα 6-24: Netbeans installation finished**

Όταν τελειώσει η εγκατάσταση βλέπουμε ότι έχει δημιουργήσει αυτόματα και μια συντόμευση στη επιφάνεια εργασίας μας.

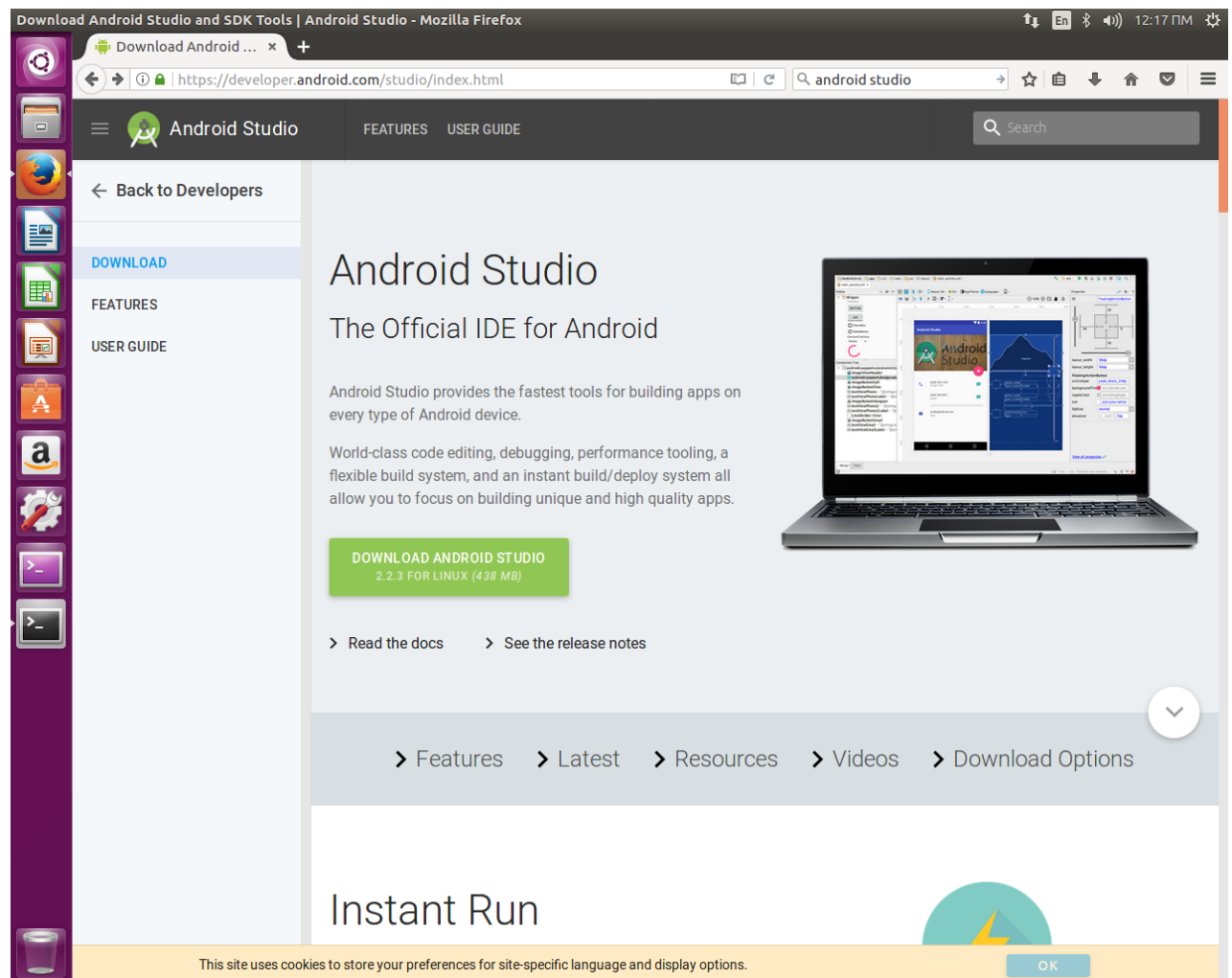


## 6.6. Εγκατάσταση Android Studio

Για να εγκαταστήσουμε το Android Studio θα πρέπει πρώτα να εγκαταστήσουμε κάποιες βιβλιοθήκες που χρειάζεται, όποτε σε ένα terminal πληκτρολογούμε την παρακάτω εντολή:

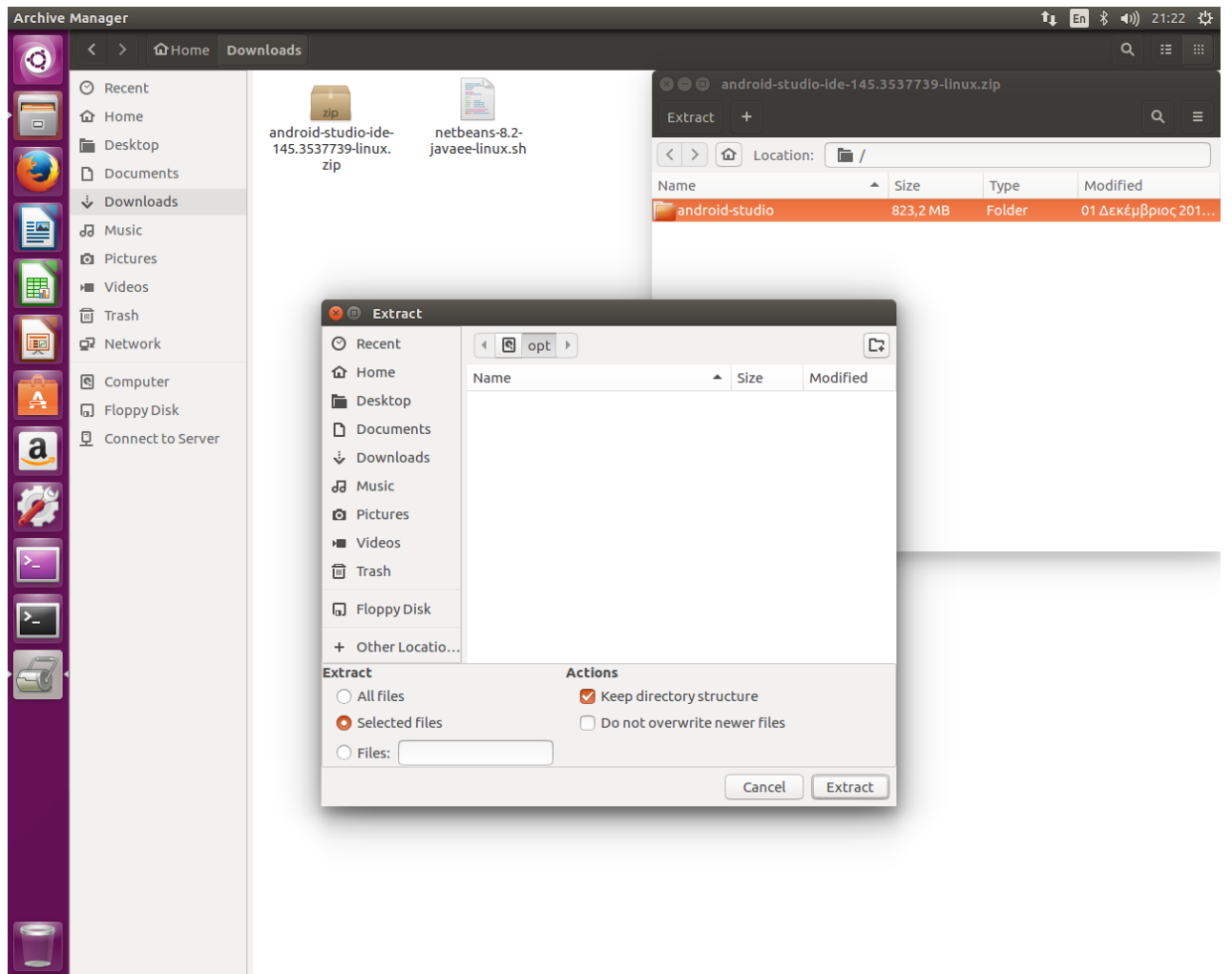
```
sudo apt-get install libc6:i386 libncurses5:i386 libstdc++6:i386 lib32z1
libbz2-1.0:i386
```

Εφόσον τελειώσει η εγκατάσταση των βιβλιοθηκών μεταφερόμαστε στη σελίδα <https://developer.android.com/studio/index.html> και κατεβάζουμε το android studio.



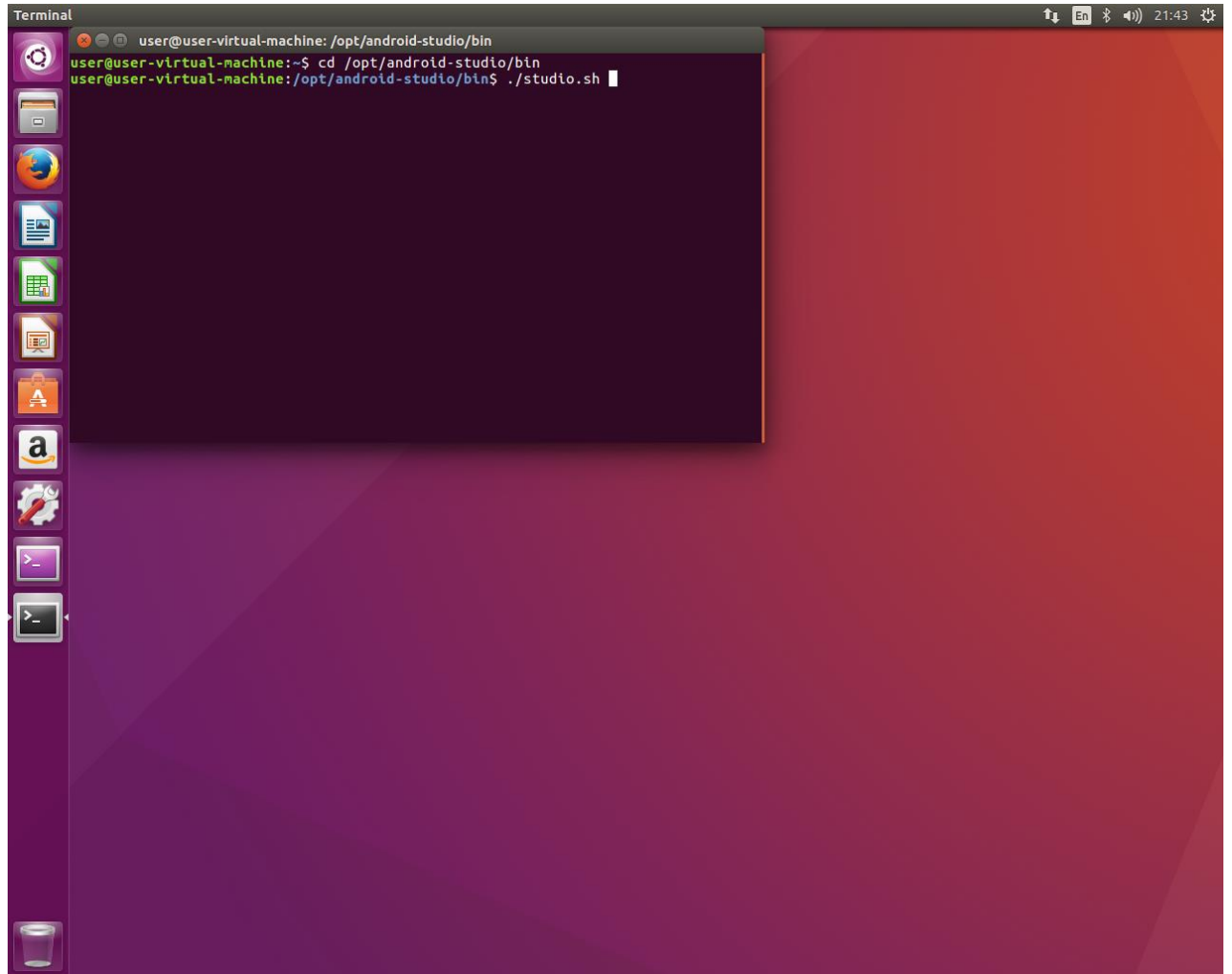
Εικόνα 6-25: Android Studio download page

Αφού κατεβάσουμε το αρχείο το οποίο θα βρίσκεται στη θέση `/home/<username>/Downloads`, το αποσυμπιέζουμε είτε στη θέση `/usr/local` για τον συγκεκριμένο χρήστη είτε στη θέση `/opt` για να είναι κοινό σε όλους τους χρηστές. Στη περίπτωση μας θα το κάναμε εξαγωγή στη θέση `/opt`.

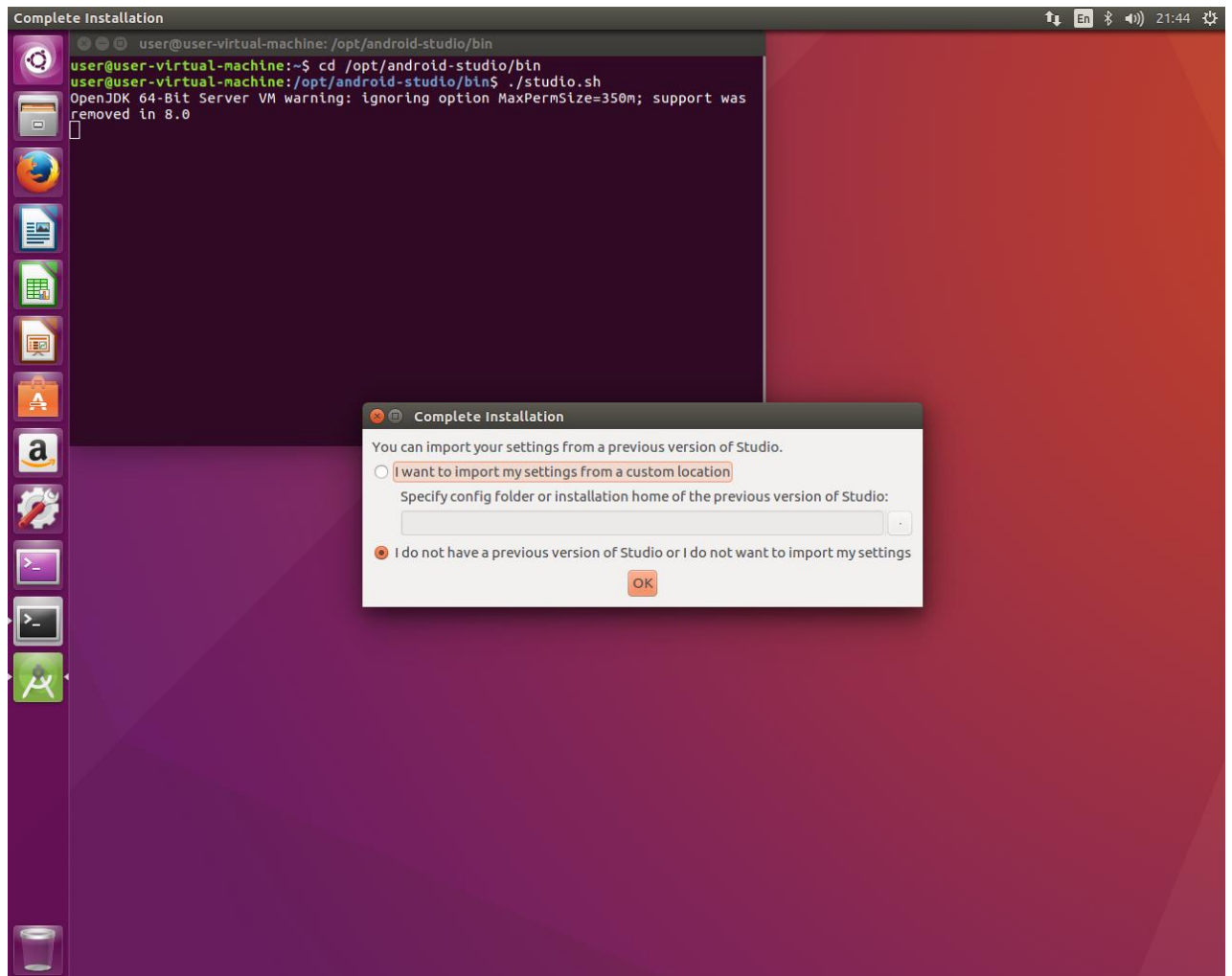


Εικόνα 6-26: Android Studio extract

Έπειτα μεταφερόμαστε στη θέση `/opt/android-studio/bin` με την βοήθεια ενός terminal και από εκεί πληκτρολογούμε `./studio.sh` για να ξεκινήσει η εγκατάσταση.

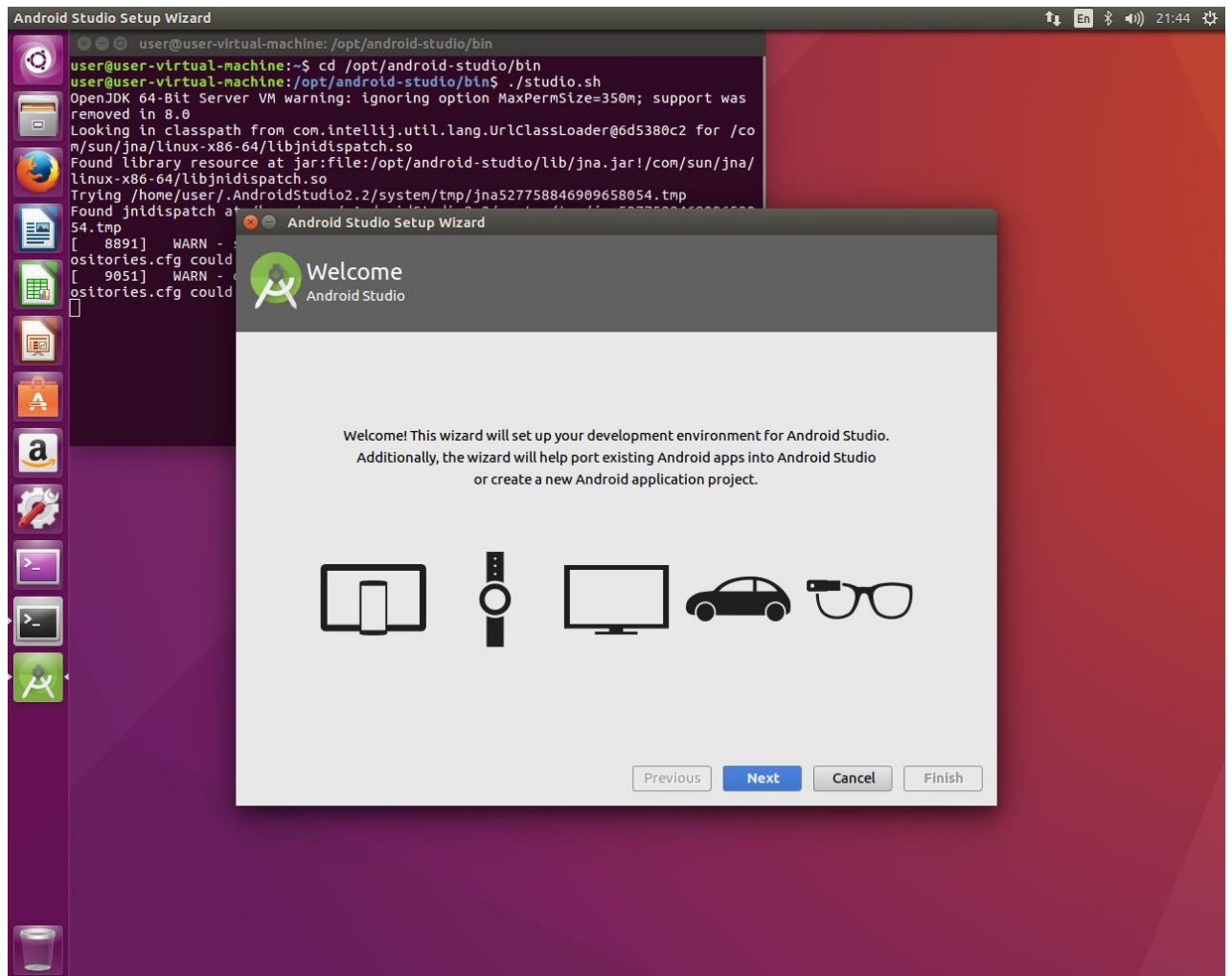


Εικόνα 6-27: Android studio initiation install

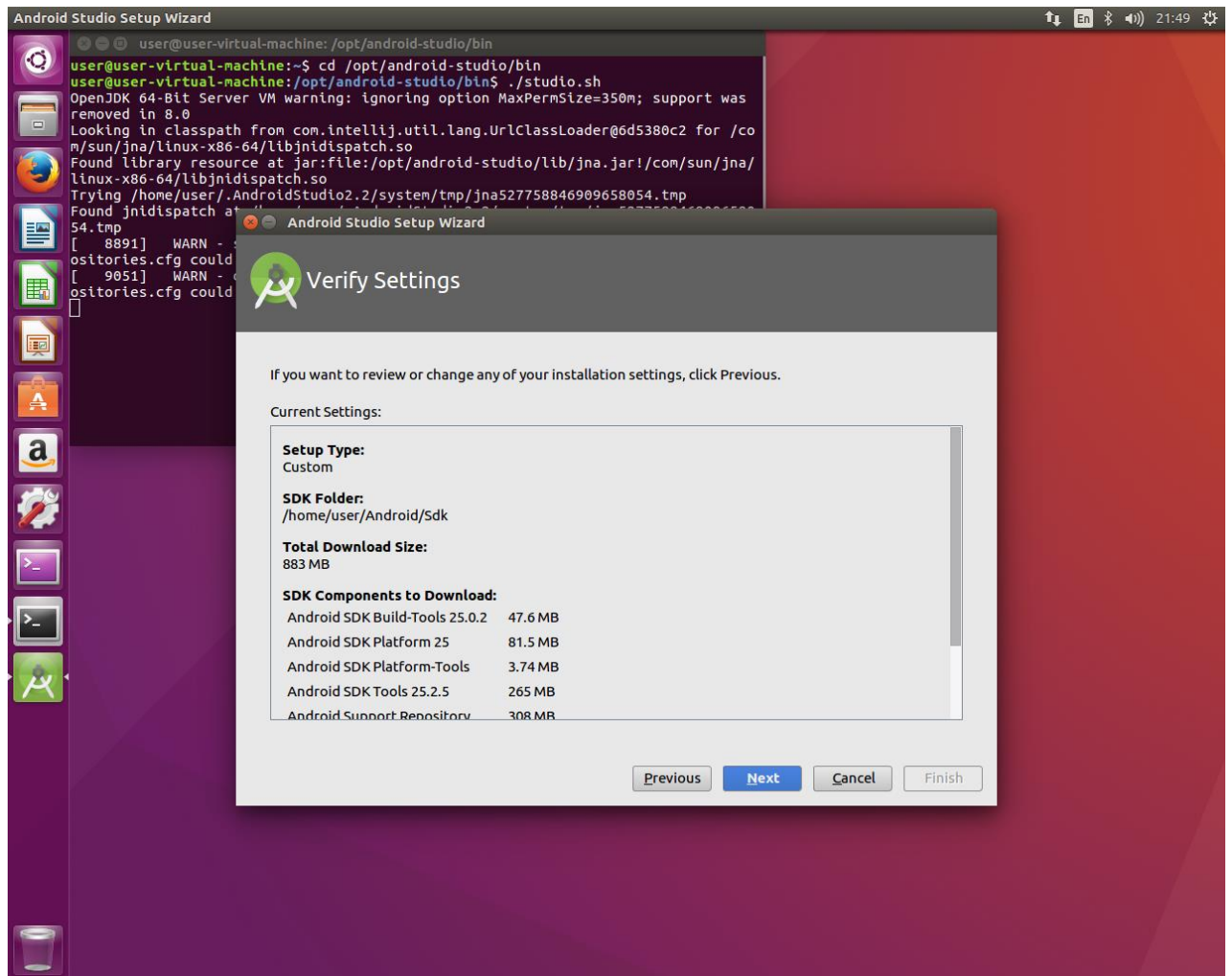


Εικόνα 6-28: Android Studio no import settings

Εδώ διαλέγουμε τη δεύτερη επιλογή για να ξεκινήσει μια νέα καθαρή εγκατάσταση.

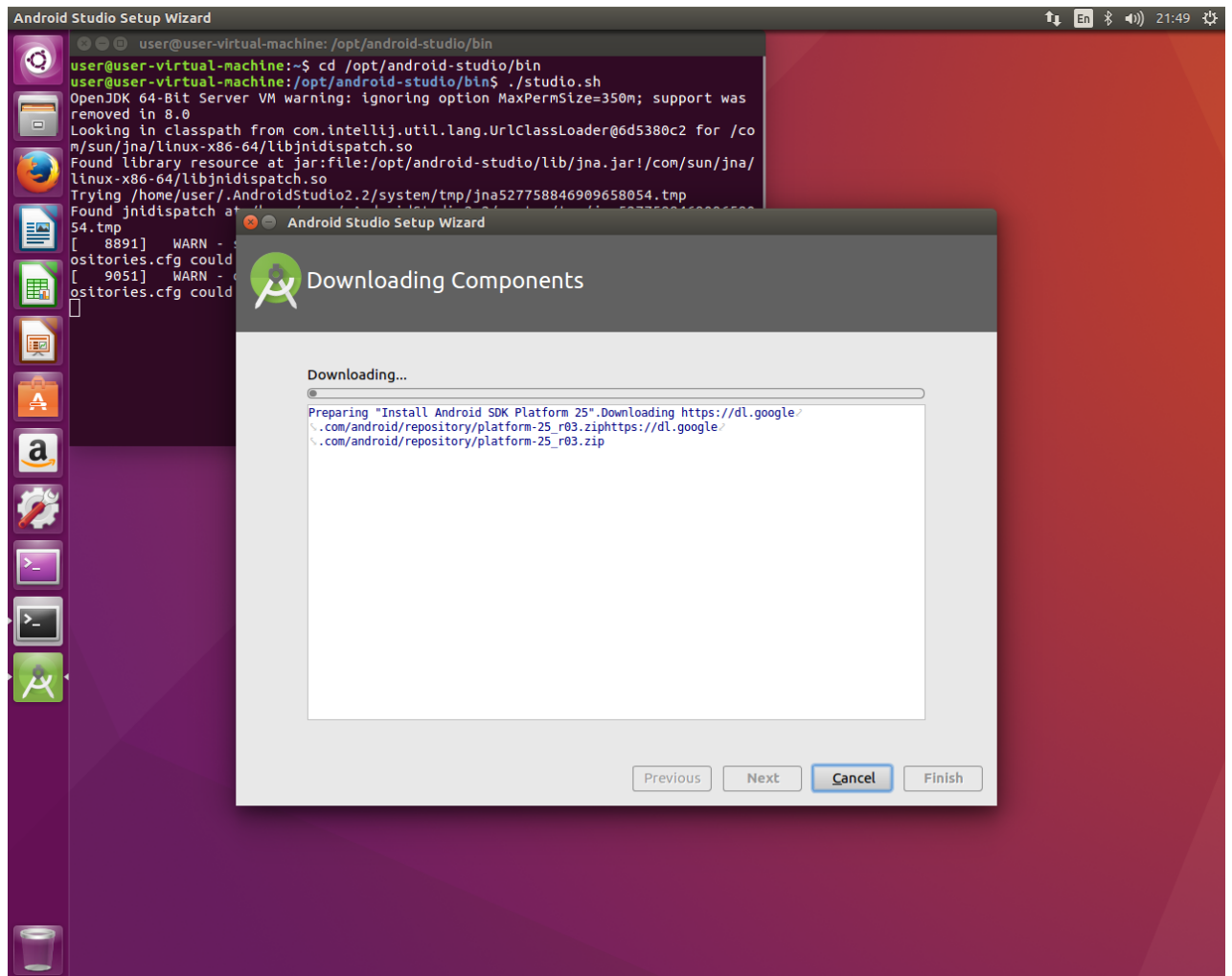


Εικόνα 6-29: Android Studio wizard

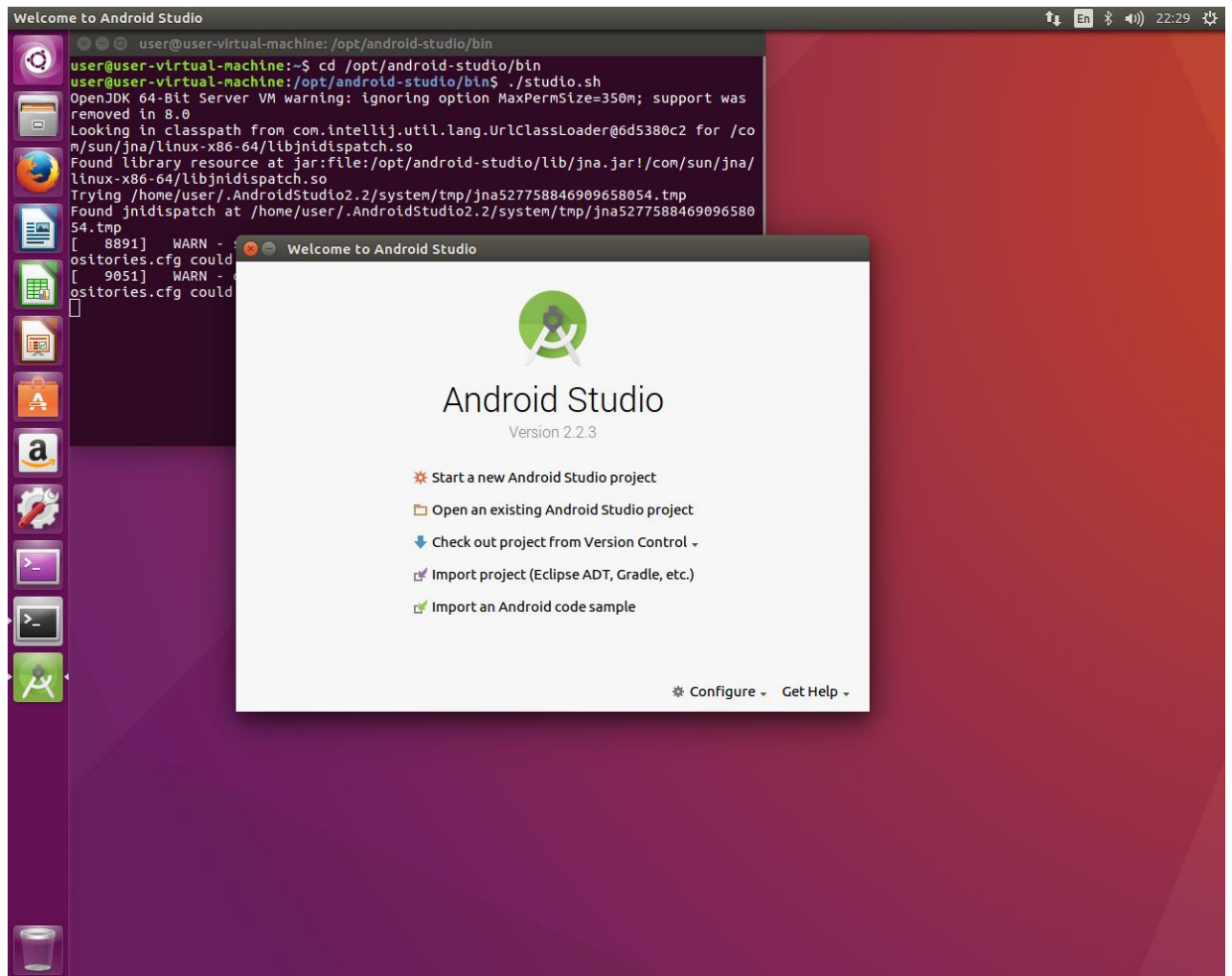


Εικόνα 6-30: Android Studio components to download

Εδώ μας γραφεί αναλυτικά τι θα κατεβάσει.



Εικόνα 6-31: Android Studio Downloading



Εικόνα 6-32: Android Studio startup screen

Μόλις τελειώσει η εγκατάσταση πατάμε finish και εμφανίζεται η αρχική οθόνη. Μόλις τελειώσει η εγκατάσταση, σε terminal πληκτρολογούμε:

```
sudo ln -s /opt/android-studio/bin/studio.sh /usr/local/bin/studio
sudo chmod 777 /usr/local/bin/studio
```

Με αυτόν τον τρόπο, πληκτρολογώντας κάθε φορά studio σε ένα terminal θα ξεκινάει το android studio.

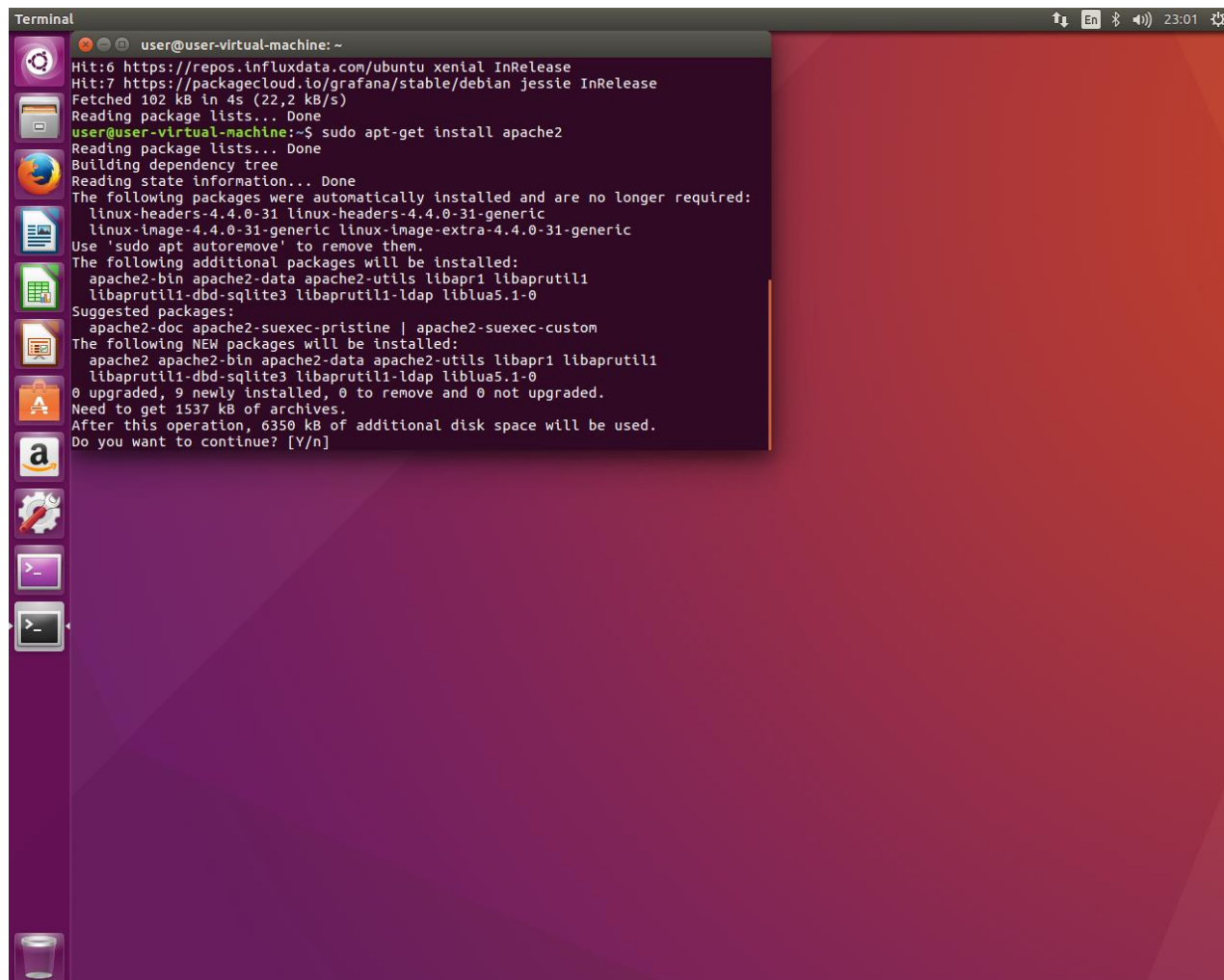


## 6.7. Εγκατάσταση του Apache Server

Με την εγκατάσταση του apache server θα έχουμε τη δυνατότητα να κάνουμε διαθέσιμη την ηλεκτρονική πύλη πρόσβαση από κάθε χρήστη του διαδικτύου, σε ένα terminal πληκτρολογούμε:

```
sudo apt-get update
```

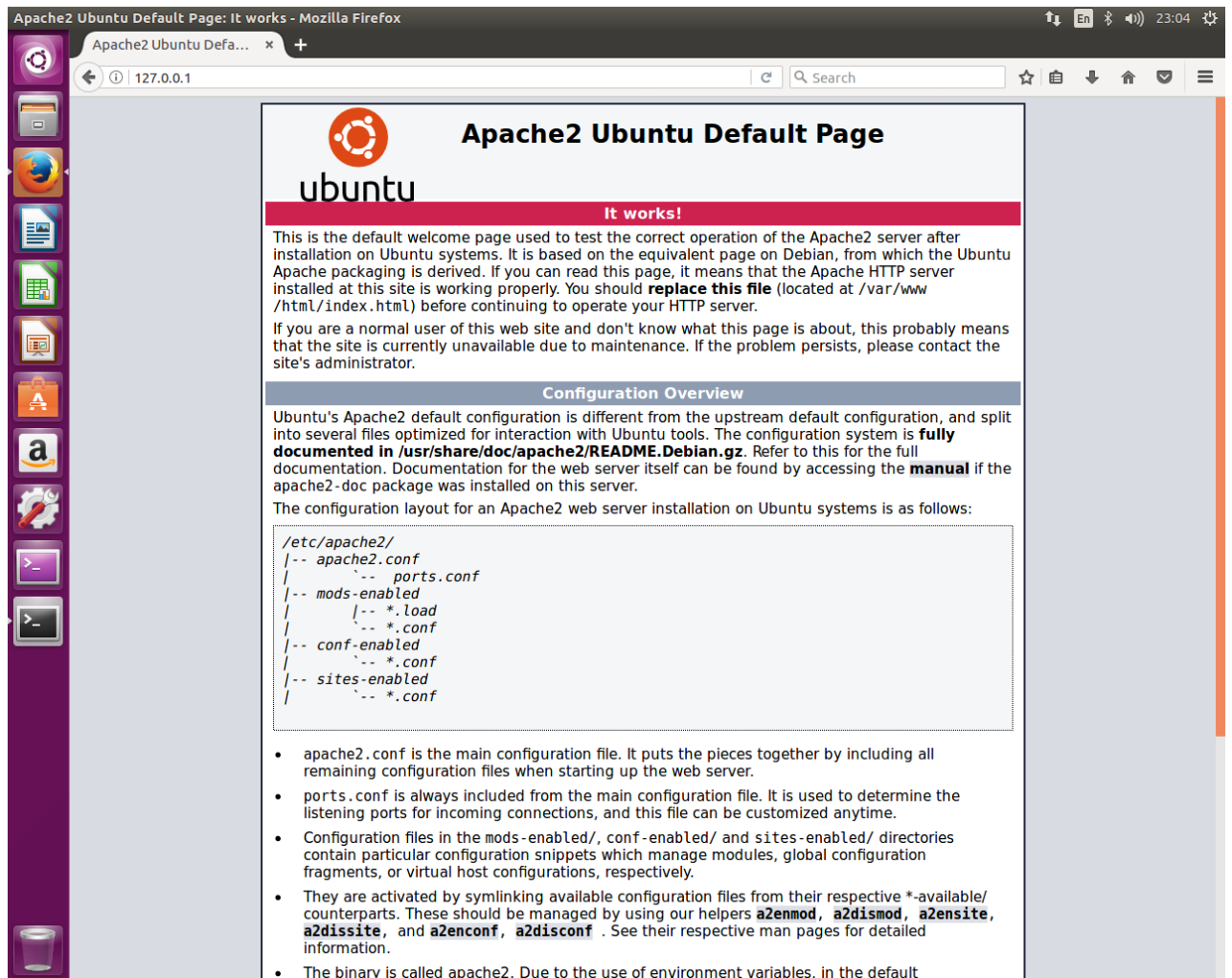
```
sudo apt-get install apache2
```



```
Terminal
user@user-virtual-machine: ~
Hit:6 https://repos.influxdata.com/ubuntu xenial InRelease
Hit:7 https://packagecloud.io/grafana/stable/debian jessie InRelease
Fetched 102 kB in 4s (22,2 kB/s)
Reading package lists... Done
user@user-virtual-machine:~$ sudo apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-4.4.0-31 linux-headers-4.4.0-31-generic
  linux-image-4.4.0-31-generic linux-image-extra-4.4.0-31-generic
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0
0 upgraded, 9 newly installed, 0 to remove and 0 not upgraded.
Need to get 1537 kB of archives.
After this operation, 6350 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Εικόνα 6-33: Install Apache Server

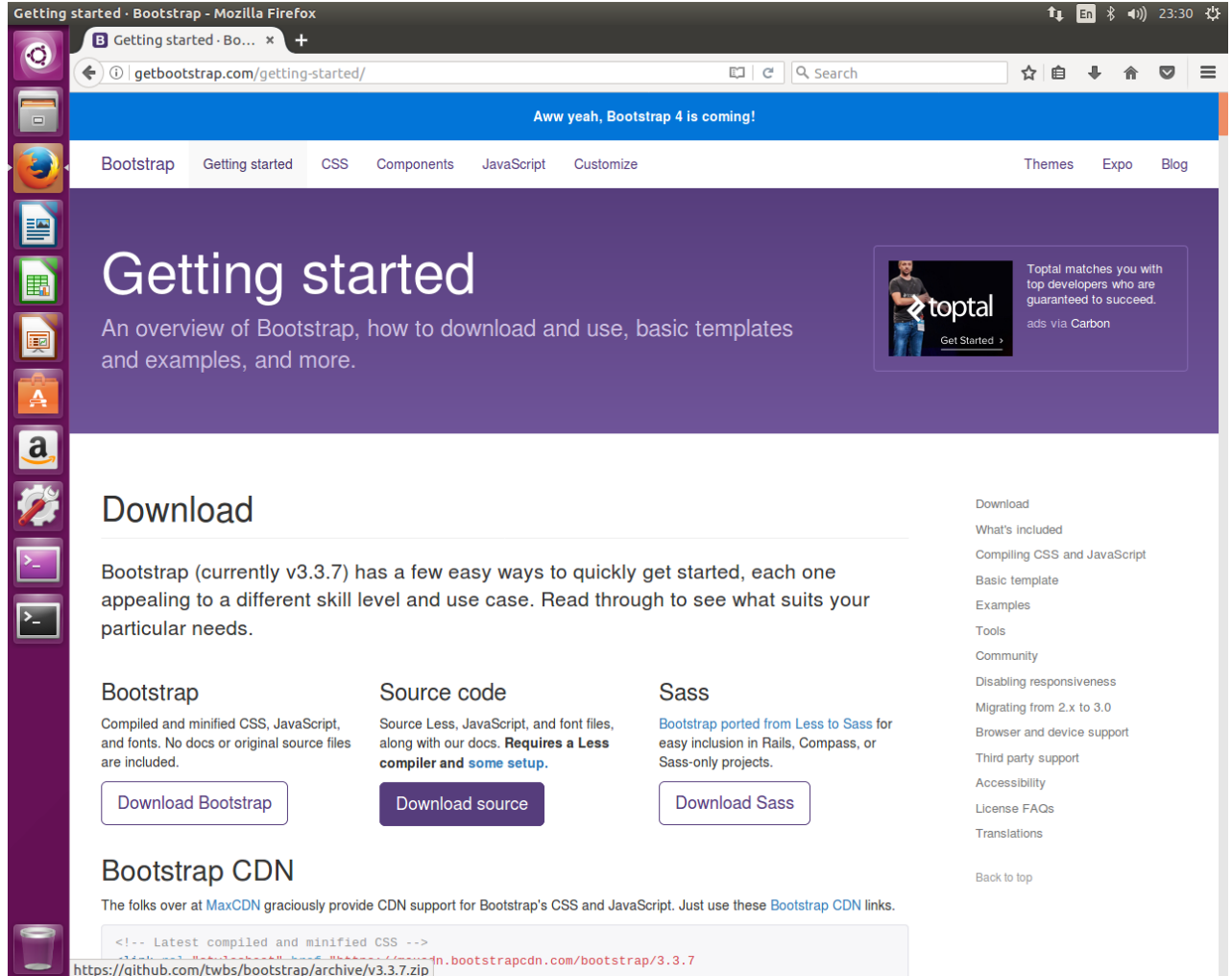
Όταν τελειώσει η εγκατάσταση του apache2 μπορούμε να την επιβιβάσουμε είτε πληκτρολογώντας `systemctl status apache2` είτε ανοίγοντας έναν περιηγητή ιστοσελίδων στην διεύθυνση 127.0.0.1 οπου πρέπει να δούμε κάτι ανάλογο με το παρακάτω.



**Εικόνα 6-34: Apache Server**

## 6.8. Κατέβασμα αρχείων Bootstrap

Χρειαζόμαστε τα αρχεία του bootstrap για να χύσουμε πάνω σε αυτά το portal. Σε έναν browser πληκτρολογούμε την σελίδα <http://getbootstrap.com/getting-started/> και από εκεί στο πεδίο download πατάμε το κουμπί download source για να κατέβει.



Εικόνα 6-35: Bootstrap download

Με βάση τα πρότυπα του bootstrap θα χύσουμε το front end της ηλεκτρονικής πύλης το οποίο θα είναι διαθέσιμο στους χρηστές.

## 6.9. Εισαγωγή δεδομένων στην influxdb

Η influxdb μας παρέχει τη δυνατότητα να εισάγουμε δεδομένα με τέσσερις διαφορετικούς τρόπους:

- Εισαγωγή δεδομένων από την ιστοσελίδα του διαχειριστή (**ΘΑ ΚΑΤΑΡΤΗΘΕΙ**)
- Εισαγωγή δεδομένων από το HTTP API
- Εισαγωγή δεδομένων το Command Line Interface
- Εισαγωγή δεδομένων από διαφορές βιβλιοθήκες Java, .Net, PHP, Python, GO και πολλές άλλες.

Δεν θα επεκταθούμε σε βάθος σε κάθε μια από τις επιλογές άλλωστε υπάρχουν και τα ανάλογα επεξηγηματικά έγγραφα στην ιστοσελίδα της influxdb.

Επειδή η βάση δεν βασίζεται σε κάποιο σχήμα πρέπει να είμαστε προσεκτικοί στη εισαγωγή δεδομένων. Γενικά πρέπει να έχουμε στο νου μας ότι έχουμε επιλέξει την σωστή βάση για να κάνουμε εισαγωγή και τα δεδομένα πρέπει να έχουν μια συγκεκριμένη μορφή, παραδείγματος χάρη:

```
weather,location=us-midwest temperature=82 1465839830100400200
measurement,tag_set field_set timestamp
```

### 6.9.1. Εισαγωγή από Web Admin Interface

Για να ενεργοποιήσουμε το Web admin interface πρέπει να βρούμε το αρχείο `influxdb.conf` το οποίο βρίσκεται στην περιοχή `/etc/influxdb/influxdb.conf` και πρέπει να το τροποποιήσουμε με έναν επεξεργαστή κειμένου ως διαχειριστής ανάλογα με τις ανάγκες μας. Στη περιοχή `[admin]` αλλάζουμε τα παρακάτω :

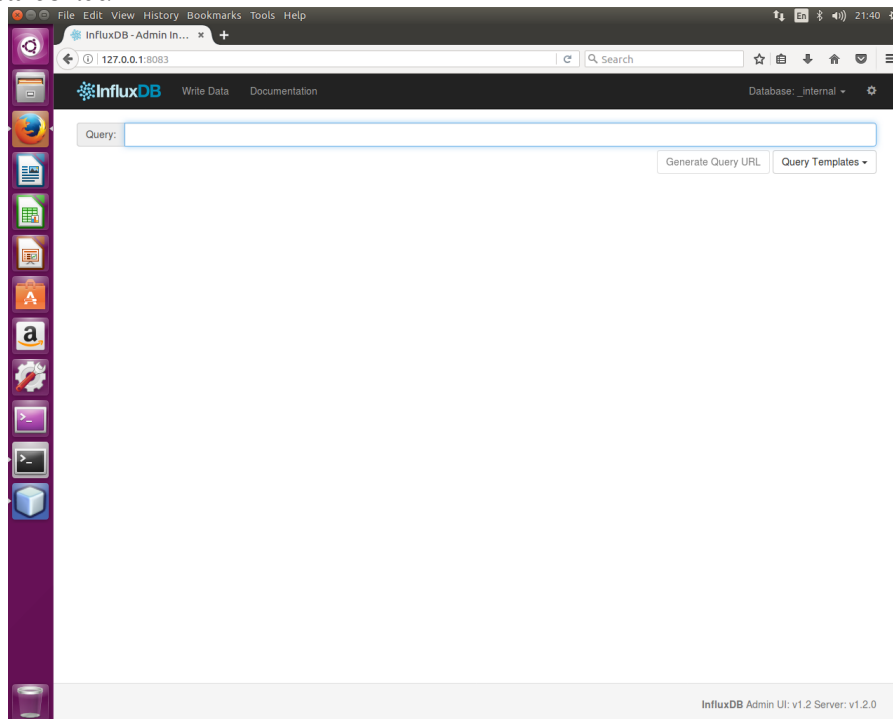
```
enable = true
bind-access = ":8083"
```

και αφαιρούμε τη `#` για να γίνουν ενεργά.

Έπειτα κάνουμε επανεκκίνηση της υπηρεσίας με την εντολή:

```
systemctl restart influxdb
```

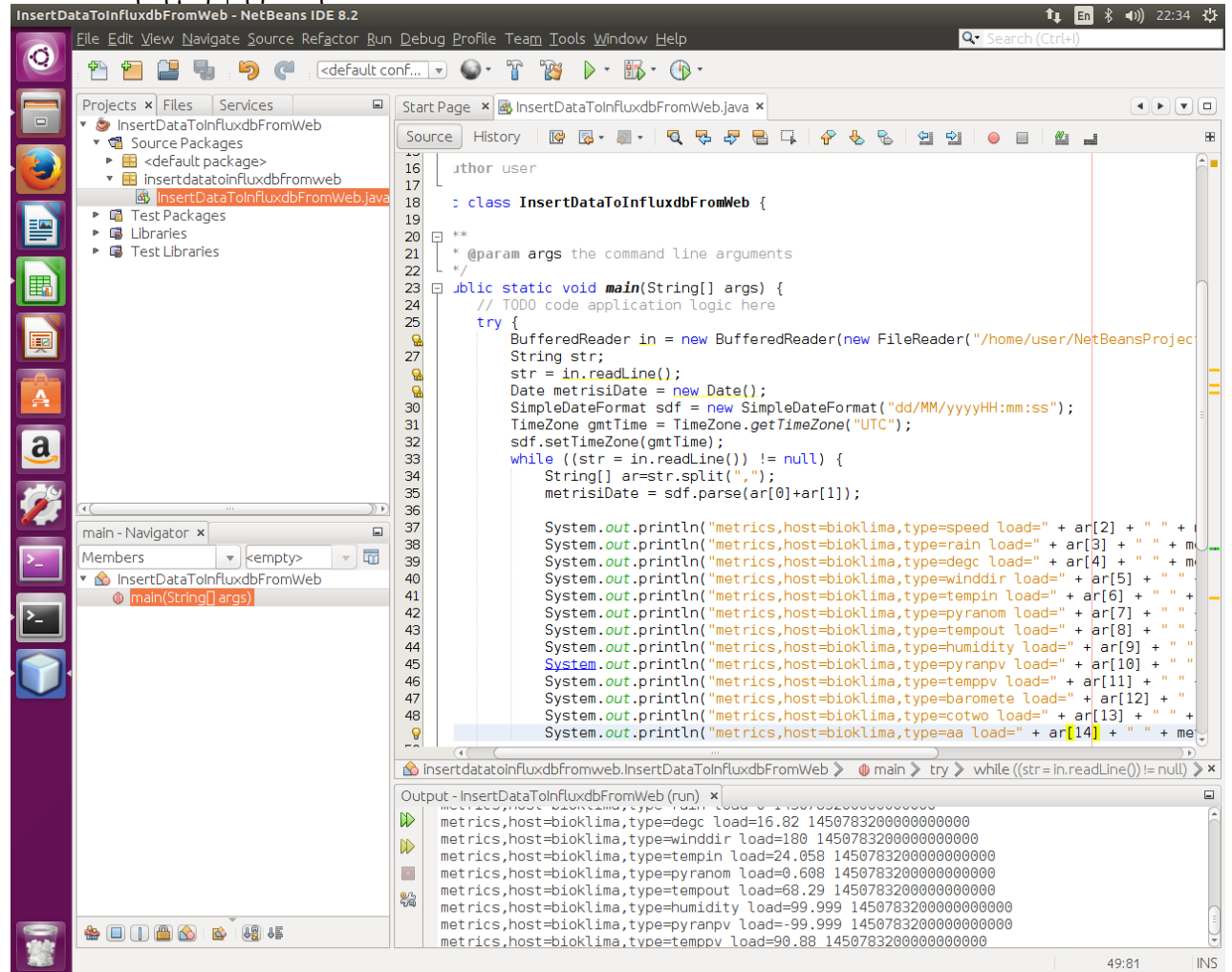
Επιβεβαιώνουμε πληκτρολογώντας σε έναν browser `127.0.0.1:8083` και βλέπουμε την παρακάτω σελίδα



Εικόνα 6-36: Influxdb web admin interface

Στο πεδίο query πληκτρολογούμε CREATE DATABASE kairos και πατάμε enter. Μετά από το κουμπί Database πάνω αριστερά επιλεγούμε την βάση που μόλις φτιάξαμε και από το κουμπί Write Data εισάγουμε τα δεδομένα μας, που πρέπει να είναι την μορφής measurement,tag\_set field\_set timestamp. Το πεδίο tag\_set δεν είναι υποχρεωτικό αλλά βοηθάει και πεδίο timestamp αν δεν το εισάγουμε θα πάρει αυτόματα τιμή από το ρολόι του συστήματος μας.

Για τις δοκιμές κατέβασα από τον server <http://bioklima5.prv3.teiher.gr/wsn/reports.php> ένα αρχείο csv το οποίο έπρεπε να μετατραπεί σε κατάλληλη μορφή για την influxdb.



Εικόνα 6-37: NetBeans csv output

Για να γίνει η μετατροπή, σε ένα νέο java project του Netbeans προσθέτουμε τον παρακάτω κώδικα:

```
public static void main(String [ ] args){
try{
BufferedReader in = new BufferedReader(new
FileReader("/home/user/NetBeansProjects/InsertDataToInfluxdbFromWeb/src/tes
t.csv"));
String str;
str = in.readLine();
Date metrisiDate = new Date();
SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yyyyHH:mm:ss");
TimeZone gmtTime = TimeZone.getTimeZone("UTC");
sdf.setTimeZone(gmtTime);
while ((str = in.readLine()) != null) {
```

```

String[] ar=str.split(",");
metrisiDate = sdf.parse(ar[0]+ar[1]);

System.out.println("metrics,host=bioklima,type=speed load=" + ar[2] + " " +
metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=rain load=" + ar[3] + " " +
metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=degc load=" + ar[4] + " " +
metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=winddir load=" + ar[5] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=tempin load=" + ar[6] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=pyranom load=" + ar[7] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=tempout load=" + ar[8] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=humidity load=" + ar[9] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=pyranpv load=" + ar[10] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=temppv load=" + ar[11] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=baromete load=" + ar[12] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=cotwo load=" + ar[13] + " " +
+ metrisiDate.getTime()*1000000);
System.out.println("metrics,host=bioklima,type=aa load=" + ar[13] + " " +
metrisiDate.getTime()*1000000);
}
in.close();
} catch (Exception e) {
System.out.println("File Read Error");
}
}

```

Προσοχή θέλει να δώσουμε τη σωστή διαδρομή του αρχείου csv για να γίνει η μετατροπή. Το αποτέλεσμα που θα εξάγει κάτω στο πεδίο output θα το αντιγράψουμε και θα εισάγουμε από web admin interface πατώντας το κουμπί write data.

## 6.9.2. Εισαγωγή δεδομένων από HTTP API

Για να κάνουμε εισαγωγή δεδομένων με το HTTP API χρησιμοποιούμε το εργαλείο cURL-curl το οποίο υπάρχει στις περισσότερες διανομές Linux και με τη μέθοδο POST εισάγουμε δεδομένα στην influxdb. Μερικά παραδείγματα:

Δημιουργία νέας βάσης

```
curl -i -XPOST http://localhost:8086/query --data-urlencode "q=CREATE DATABASE mydb"
```

Εμφάνιση αποτελεσματος

```
curl -G 'http://localhost:8086/query?pretty=true' --data-urlencode "q=show databases"
```

Διαγραφή της βάσης που δημιουργήσαμε

```
curl -G 'http://localhost:8086/query?pretty=true' --data-urlencode "q=drop database mydb"
```

```
Terminal
user@user-virtual-machine: ~
user@user-virtual-machine:~$ curl -i -XPOST http://localhost:8086/query --data-urlencode "q=CREATE DATABASE kairos"
HTTP/1.1 200 OK
Connection: close
Content-Type: application/json
Request-Id: da701b34-f04e-11e6-800f-000000000000
X-Influxdb-Version: 1.2.0
Date: Sat, 11 Feb 2017 11:40:14 GMT
Transfer-Encoding: chunked

{"results":[{"statement_id":0}]}
user@user-virtual-machine:~$ curl -G 'http://localhost:8086/query' --data-urlencode "q=show databases"
{"results":[{"statement_id":0,"series":[{"name":"databases","columns":["name"],"values:[["_internal"],["kairos"]}]}]}
user@user-virtual-machine:~$ curl -i -XPOST http://localhost:8086/query --data-urlencode "q=DROP DATABASE kairos"
HTTP/1.1 200 OK
Connection: close
Content-Type: application/json
Request-Id: f05dfc56-f04e-11e6-8011-000000000000
X-Influxdb-Version: 1.2.0
Date: Sat, 11 Feb 2017 11:40:51 GMT
Transfer-Encoding: chunked

{"results":[{"statement_id":0}]}
user@user-virtual-machine:~$ curl -G 'http://localhost:8086/query' --data-urlencode "q=show databases"
{"results":[{"statement_id":0,"series":[{"name":"databases","columns":["name"],"values:[["_internal"]}]}]}
user@user-virtual-machine:~$
```

Εικόνα 6-38: Influxdb http api

Εγγραφή δεδομένων στη βάση

```
curl -i -XPOST 'http://localhost:8086/write?db=mydb' --data-binary 'cpu_load_short,host=server01,region=us-west value=0.64 1434055562000000000'
```

Εγγραφή πολλαπλών δεδομένων

```
curl -i -XPOST 'http://localhost:8086/write?db=mydb' --data-binary 'cpu_load_short,host=server02 value=0.67
cpu_load_short,host=server02,region=us-west value=0.55 1422568543702900257
cpu_load_short,direction=in,host=server01,region=us-west value=2.0
1422568543702900257'
```

Εγγραφή από αρχείο. Έστω το αρχείο cpu\_data.txt με τις παρακάτω καταχωρίσεις

```
cpu_load_short,host=server02 value=0.67  
cpu_load_short,host=server02,region=us-west value=0.55 1422568543702900257  
cpu_load_short,direction=in,host=server01,region=us-west value=2.0  
1422568543702900257
```

**Και εγγραφή στη βάση**

```
curl -i -XPOST 'http://localhost:8086/write?db=mydb' --data-binary  
@cpu_data.txt
```



### 6.9.3. Εισαγωγή δεδομένων από το Command Line Interface

Για να εισάγουμε δεδομένα από το CLI ανοίγουμε ένα terminal και πληκτρολογούμε influx.

```

Terminal
user@user-virtual-machine: ~
user@user-virtual-machine:~$ influx
Connected to http://localhost:8086 version 1.2.0
InfluxDB shell version: 1.2.0
> help
Usage:
connect <host:port> connects to another node specified by host:port
auth prompts for username and password
pretty toggles pretty print for the json format
use <db_name> sets current database
format <format> specifies the format of the server responses: json, csv, or column
precision <format> specifies the format of the timestamp: rfc3339, h, m, s, ms, u or ns
consistency <level> sets write consistency level: any, one, quorum, or all
history displays command history
settings outputs the current settings for the shell
clear clears settings such as database or retention policy. run 'clear' for help
exit/quit/ctrl+d quits the influx shell

show databases show database names
show series show series information
show measurements show measurement information
show tag keys show tag key information
show field keys show field key information

A full list of influxql commands can be found at:
https://docs.influxdata.com/influxdb/latest/query_language/spec/

> show databases
name: databases
-----
 _internal
> use _internal
Using database _internal
>

```

Εικόνα 6-39: Influxdb CLI

Πληκτρολογώντας την εντολή help μας δείχνει κάποιες διαθέσιμες εντολές. Για να εισάγουμε δεδομένα πρέπει πρώτα να επιλέξουμε την βάση που θα εισάγουμε τα δεδομένα, συνεπώς πληκτρολογούμε την εντολή show databases και αφού εμφανίσει τις βάσεις επιλεγούμε με την εντολή use και το όνομα της βάσης.

Για εισαγωγή χρησιμοποιούμε την εντολή insert.

```
insert metrics,host=bioklima,type=degc load=25.
```

Για να εμφανίσει το αποτέλεσμα πληκτρολογούμε

```
select * from metrics where host='bioklima' and type='degc'.
```

Για να διαγράψουμε μια εγγραφή χρησιμοποιούμε τις εντολές delete ή drop.

## 6.9.4. Εισαγωγή δεδομένων με τη βιβλιοθήκη της Java

Για να μπορέσουμε να κάνουμε τη σύνδεση της java με την influxdb επιτυχώς, πρέπει να έχουμε μια επιπλέον βιβλιοθήκη της influx και να την εισάγουμε στο project μας.

Για να βρούμε την βιβλιοθήκη πληκτρολογούμε στον browser μας την διεύθυνση:

[https://docs.influxdata.com/influxdb/v1.2/tools/api\\_client\\_libraries/](https://docs.influxdata.com/influxdb/v1.2/tools/api_client_libraries/)

και επιλεγούμε την Java. Στην σελίδα θα υπάρχουν και άλλες βιβλιοθήκες διαθέσιμες σε

άλλες γλώσσες.

Αφού επιλέξουμε την Java, το πρόγραμμα περιήγησης θα μας κατευθύνει στη διεύθυνση:

<https://github.com/influxdata/influxdb-java>

Ουσιαστικά είναι ένα project που υποστηρίζεται από την influx και από εκεί μπορούμε εφόσον τον κατεβάσουμε και έχουμε στον υπολογιστή μας τα απαιτούμενα προγράμματα (maven 3+ , docker-engine) να το συνθέσουμε με έναν compiler και να εξάγουμε το jar αρχείο που χρειαζόμαστε. Όμως υπάρχει και η δυνατότητα να το κατεβάσουμε έτοιμο από ένα άλλο project που υπάρχει επίσης στο GitHub. Συνεπώς πληκτρολογούμε στο περιηγητή:

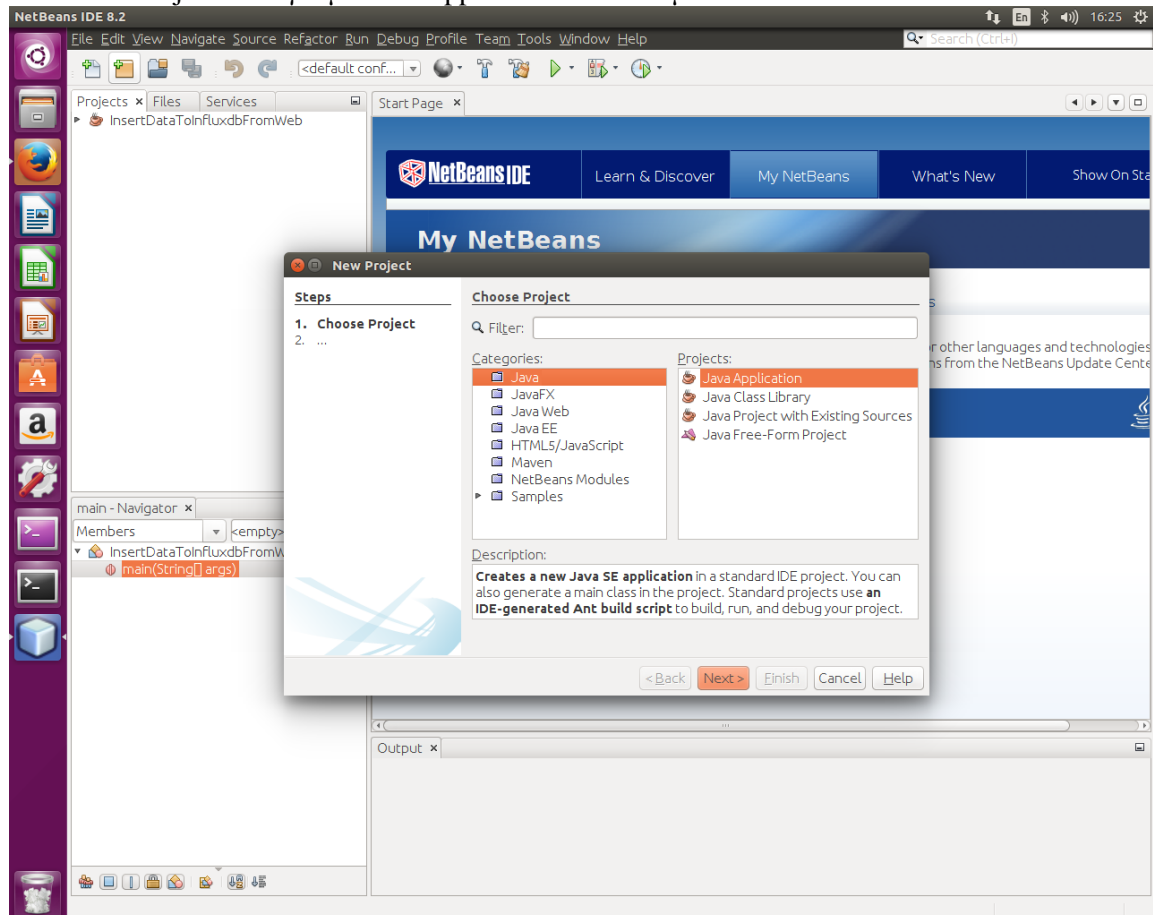
<https://github.com/ashishdoneriya/influxdb-java>

και από έχει στο τέλος της σελίδας την επιλογή:

[https://github.com/ashishdoneriya/influxdb-](https://github.com/ashishdoneriya/influxdb-java/releases/download/2.5.1/influxdb-2.5.1.jar)

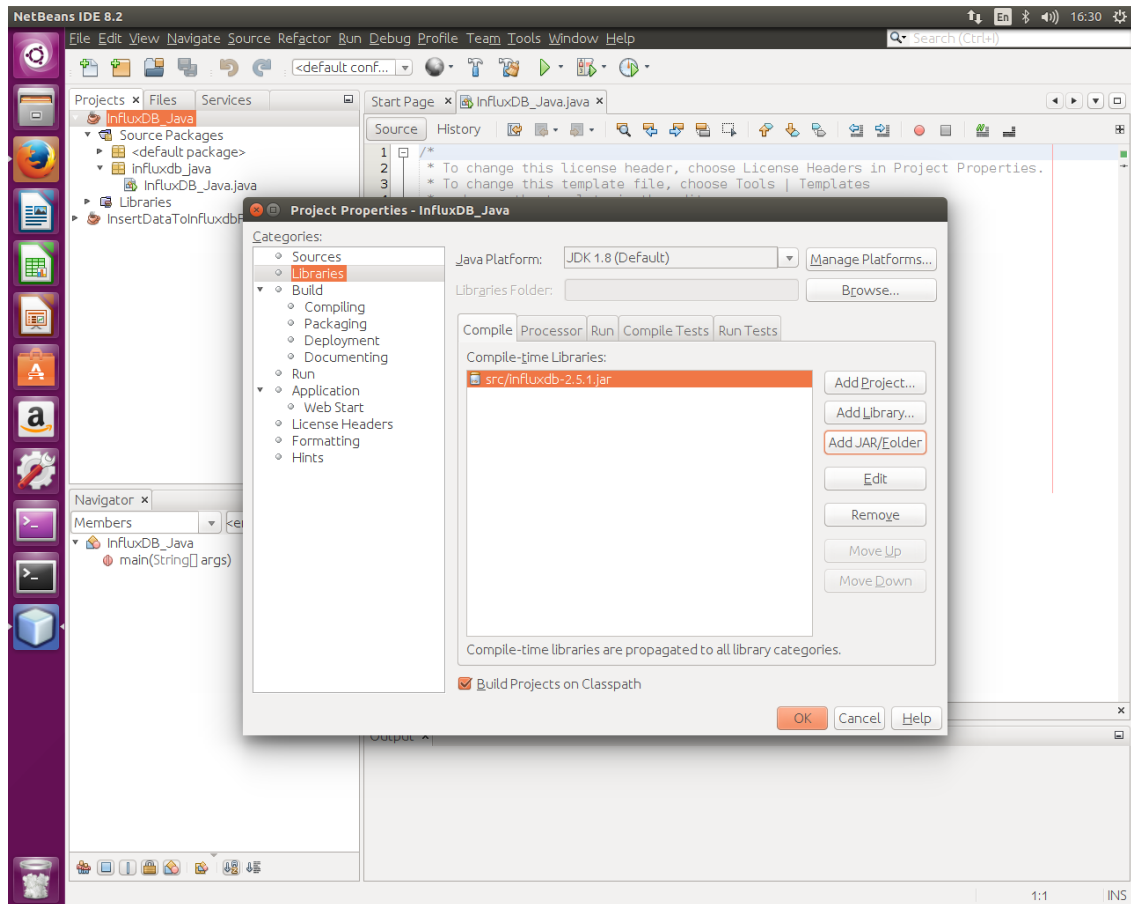
[java/releases/download/2.5.1/influxdb-2.5.1.jar](https://github.com/ashishdoneriya/influxdb-java/releases/download/2.5.1/influxdb-2.5.1.jar)

Τώρα ανοίγοντας το NetBeans δημιουργούμε ένα νέο project από την επιλογή File>New Project επιλεγούμε Java Application και πατάμε next.



Εικόνα 6-40: NetBeans new Java Project

Το αρχείο που κατεβάσαμε νωρίτερα το βάζουμε στο φάκελο /src που βρίσκεται μέσα στο νέο project που μόλις δημιουργήσαμε. Κάνοντας δεξί κλικ πάνω στο φάκελο του project μας μέσα στο NetBeans επιλεγούμε properties από το μενού. Έπειτα αριστερά επιλεγούμε libraries και προσθέτουμε το αρχείο που μόλις κατεβάσαμε.



Εικόνα 6-41: Netbeans influxdb jar

Ένα παράδειγμα μπορεί να είναι το παρακάτω:

```
public class influxdb {
private static Configuration configuration;
private static DataWriter writer;
private static Query query;
private static DataReader dataReader;
private static ResultSet resultSet;
public static void main(String[] args) throws IOException,
URISyntaxException, Exception {
//make connection with the database
configuration = new Configuration("127.0.0.1", "8086", "admin", "admin",
"db_test");
//Create a new object data writer
writer = new DataWriter(configuration);
//initialize the writer
writer.setMeasurement("cpu,");
writer.addTag("host=serverA", "region=us_west");
writer.setTimeUnit(TimeUnit.MILLISECONDS);
writer.setTime(System.currentTimeMillis());
writer.addField("value", 69.0);
//flush the data to the database
writer.writeData();
//Create a new query
query = new Query();
query.setCustomQuery("SELECT * FROM cpu WHERE host ='serverA' AND region
='us_west'");
//make the query
```

```
dataReader = new DataReader(query, configuration);  
//send query  
resultSet = dataReader.getResult();  
//show results  
    System.out.println(resultSet);  
    }  
}
```

## 6.10. Δημιουργία γραφημάτων με το Grafana

Πριν ξεκινήσουμε να προσθέτουμε διαγράμματα στο Grafana θα πρέπει πρώτα να κάνουμε κάποιες αλλαγές στο αρχείο ρυθμίσεων του. Το αρχείο με τις ρυθμίσεις του βρίσκεται στο `/etc/grafana/grafana.ini` και θα το τροποποιήσουμε ανάλογα:

### Paths

```
# Path to where grafana can store temp files, sessions, and the sqlite3 db (if that is used)
data = /var/lib/grafana
# Directory where grafana can store logs
logs = /var/log/grafana
# Directory where grafana will automatically scan and look for plugins
plugins = /var/lib/grafana/plugins
```

### Server

```
# Protocol (http or https)
protocol = http
# The ip address to bind to, empty will bind to all interfaces
http_addr =
# The http port to use
http_port = 3000
# The public facing domain name used to access grafana from a browser
domain = localhost
# The full public facing url you use in browser, used for redirects and emails
# If you use reverse proxy and sub path specify full url (with sub path)
root_url = http://localhost:3000
```

### Security

```
# default admin user, created on startup
admin_user = admin
# default admin password, can be changed before first start of grafana, or in profile settings
admin_password = admin
```

### Users

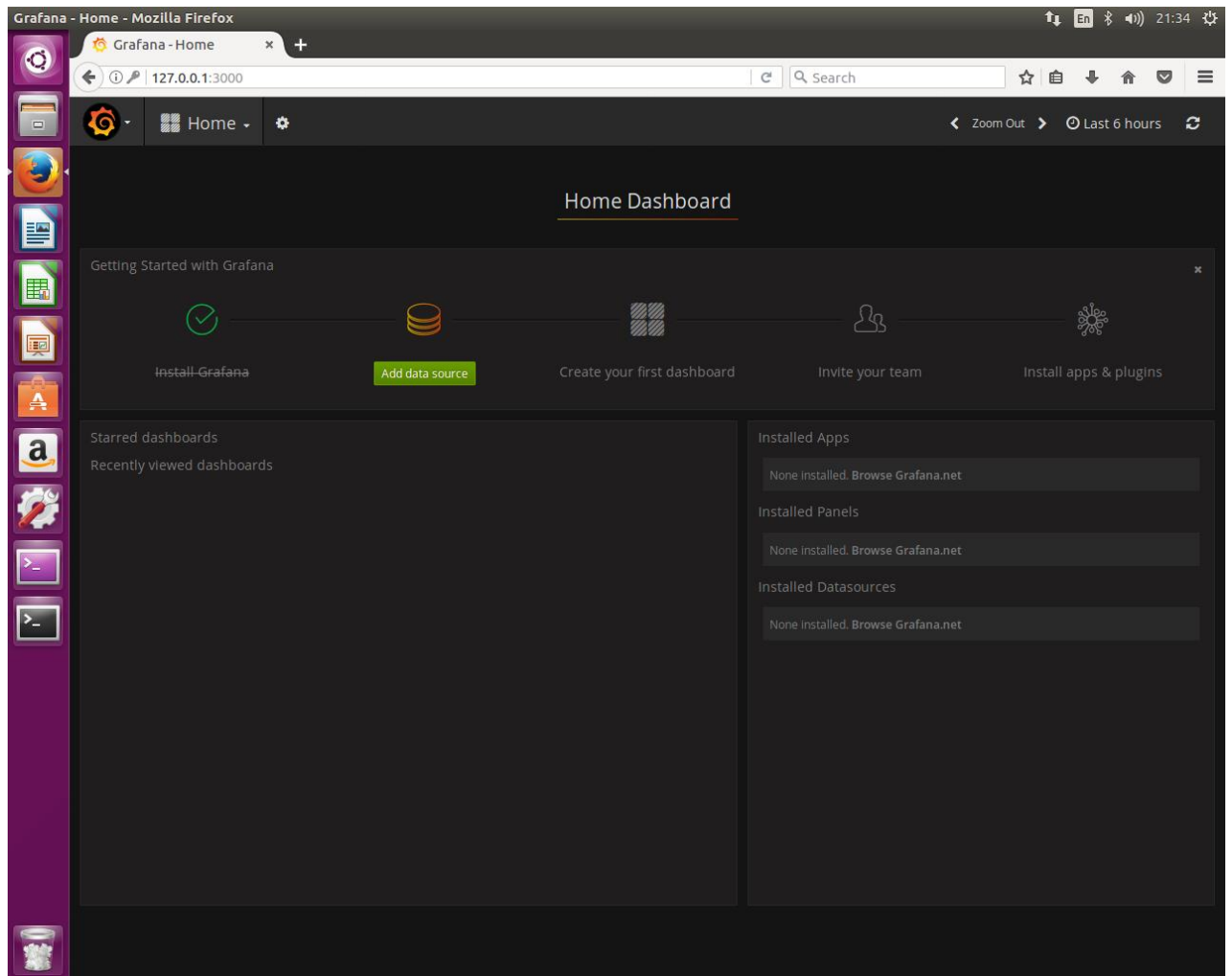
```
# disable user signup / registration
allow_sign_up = false
```

### Authorization anonymous

```
# enable anonymous access
enabled = true
# specify organization name that should be used for unauthenticated users
org_name = Main Org.
# specify role for unauthenticated users
org_role = Viewer
```

Αφού κάνουμε τις αλλαγές και αποθηκεύσουμε το αρχείο κάνουμε και μια επανεκκίνηση της υπηρεσίας με την εντολή `systemctl restart Grafana-server` για να ενεργοποιηθούν οι αλλαγές.

Για να δημιουργήσουμε τα γραφήματα πληκτρολογούμε στον φυλλομετρητή μας την διεύθυνση `http://127.0.0.1:3000/login` και βάζουμε `admin` για όνομα χρήστη και κωδικό.



Εικόνα 6-42: Grafana first login

Βλέπουμε ότι μας εμφανίζει έναν οδηγό, και το πρώτο πράγματα που μας ζητάει είναι να προσθέσουμε μια πηγή δεδομένων. Πατάμε πάνω στο κουμπί Add data source για να προσθέσουμε την influxdb. Στην οθόνη που εμφανίζεται βάζουμε τα ανάλογα στοιχεία που μας ζητάει:

Edit Data Source

Name: kairos

Type: influxdb

Http settings

Url: http://192.168.1.251:8086 εδώ καλό είναι να βάλουμε την κανονική διεύθυνση και όχι localhost ή 127.0.0.1

Access: direct

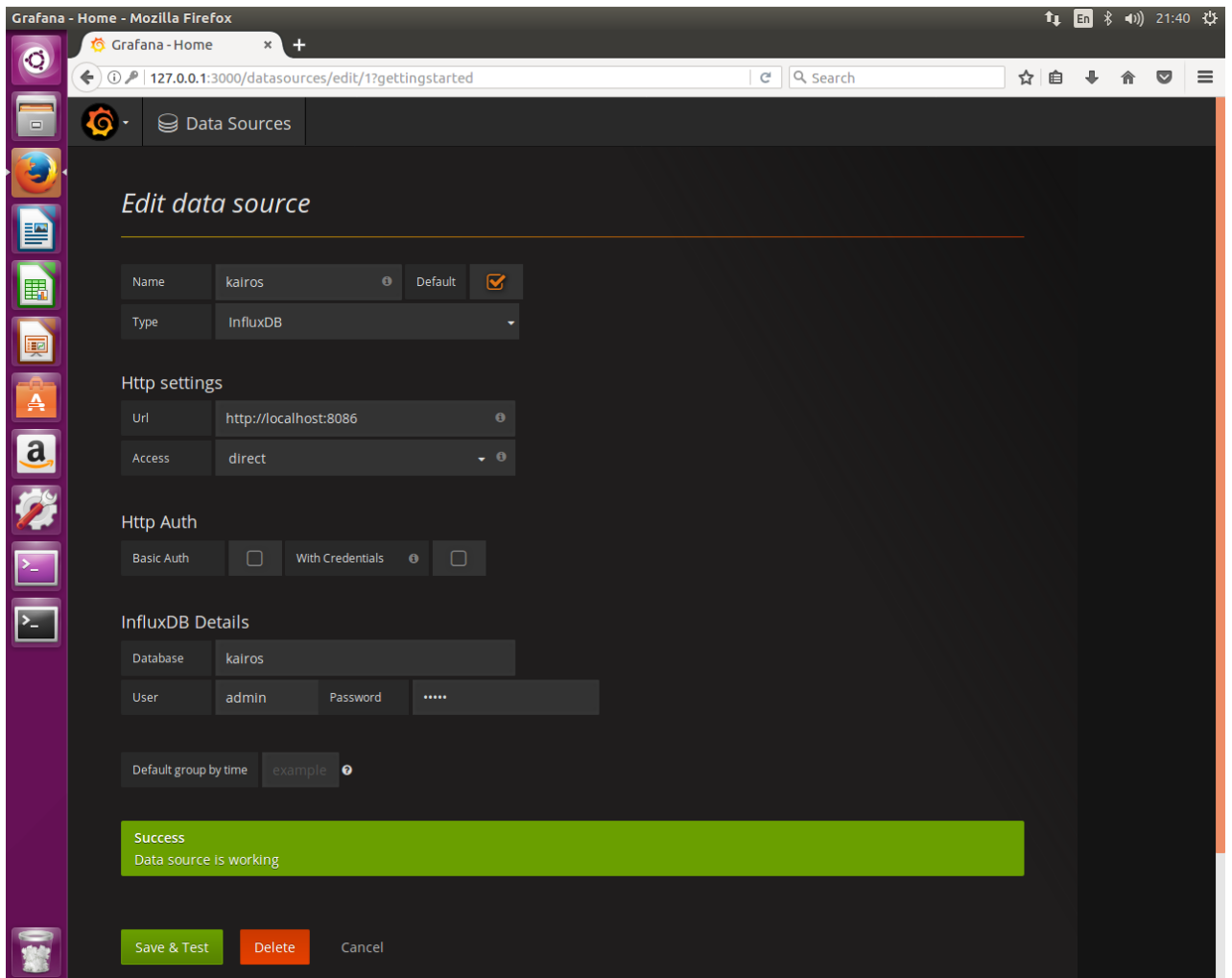
InfluxDB Details

Database: kairos

User:admin

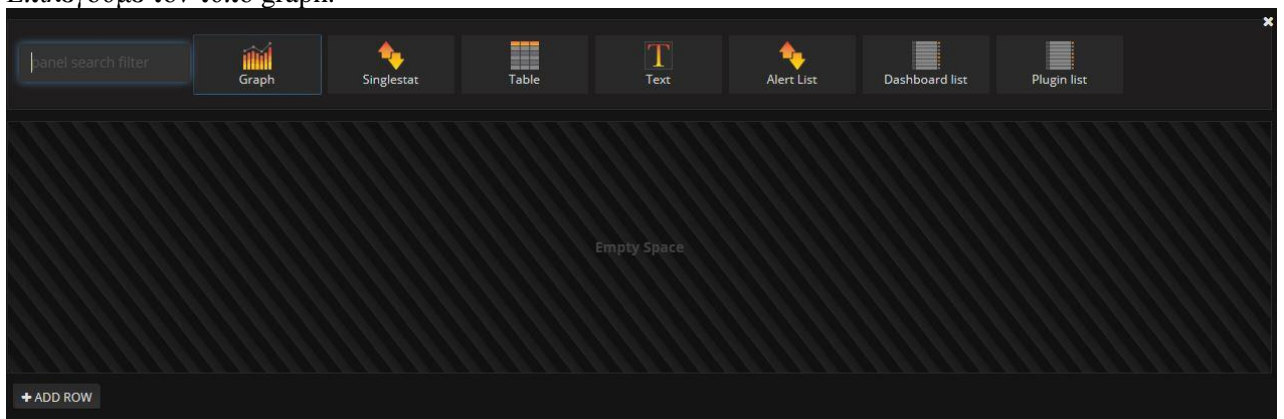
Password:admin

Μετά πατάμε save and test, και μας εμφανίζει μήνυμα success data source is working



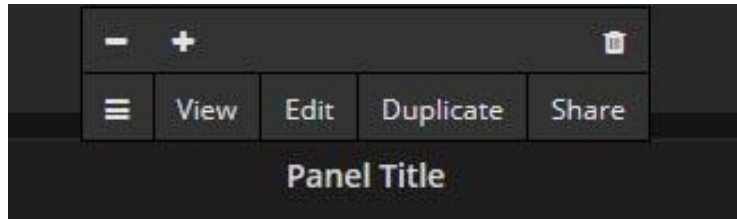
Εικόνα 6-43: Grafana add data source

Μετά πατάμε πάνω αριστερά στο λογότυπο του Grafana και εμφανίζεται ένα νέο μενού, από εκεί πατάμε Dashboards>new για φτιάξουμε ένα νέο. Από το μενού που εμφανίζει, μας δείχνει τους διαθέσιμους τύπους γραφημάτων για να επιλέξουμε για το dashboard. Επιλεγούμε τον τύπο graph.



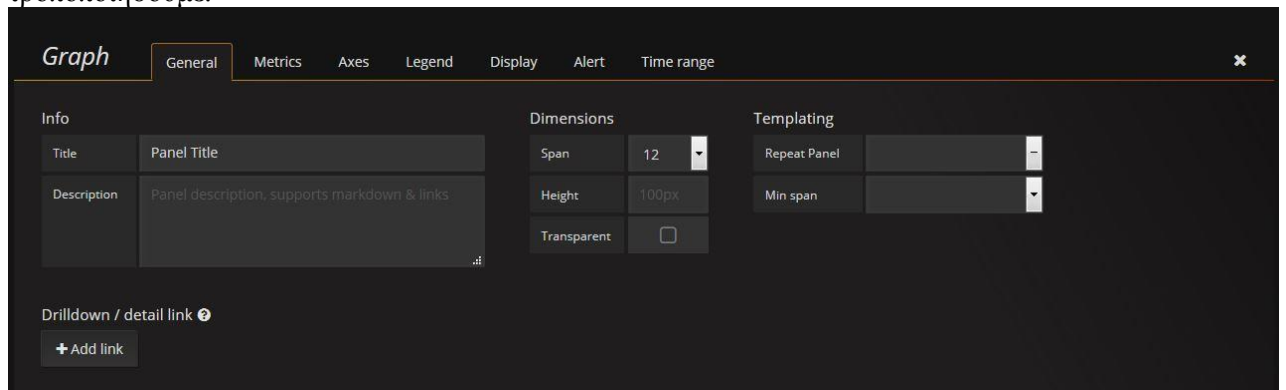
Εικόνα 6-44: Grafana select graph type

Με το που επιλέξουμε τον τύπο που επιθυμούμε, εμφανίζεται το νέο γράφημα και πατάμε πάνω στο όνομα του, όπου εμφανίζει μερικές επιλογές, και από αυτές πατάμε edit. Τώρα ανοίγει ένα νέα καρτέλα κάτω από το γράφημα και από εκεί μπορούμε να το τροποποιήσουμε.



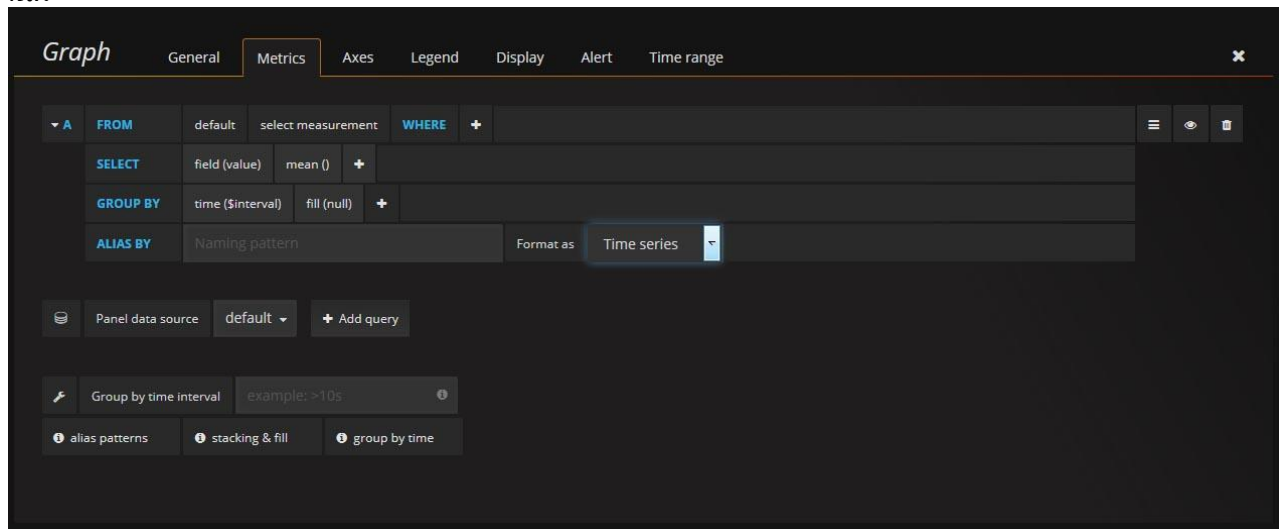
Εικόνα 6-45: Graph available settings

Τώρα ανοίγει ένα νέα καρτέλα κάτω από το γράφημα και από εκεί μπορούμε να το τροποποιήσουμε.



Εικόνα 6-46: Graph general tab

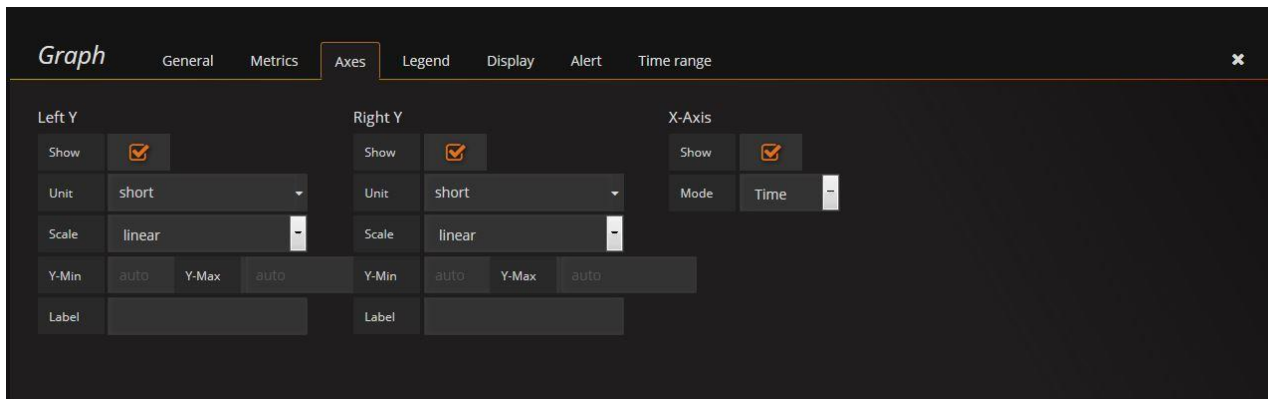
Στην καρτέλα general όπως βλέπουμε μπορούμε να δώσουμε όνομα, περιγραφή, διαστάσεις κτλ



Εικόνα 6-47: Graph metrics tab

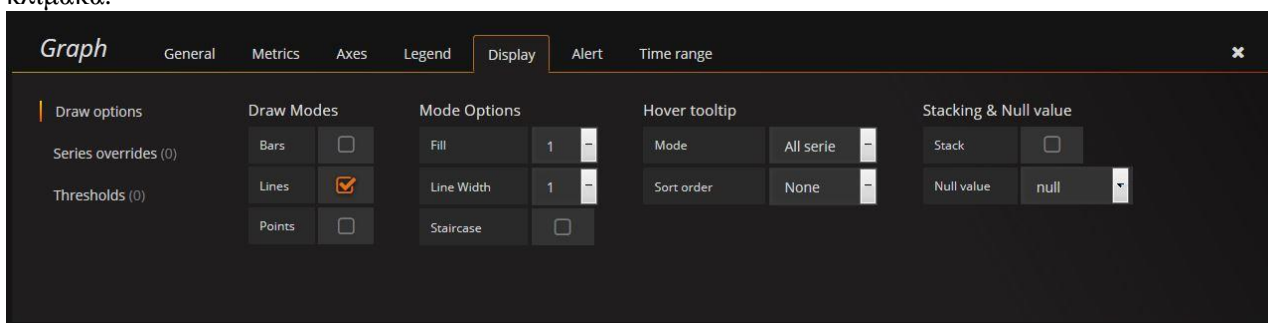
Σε αυτήν την καρτέλα ορίζουμε τι θα παρουσιάζει το γράφημα και κάνουμε την σύνδεση με τη βάση





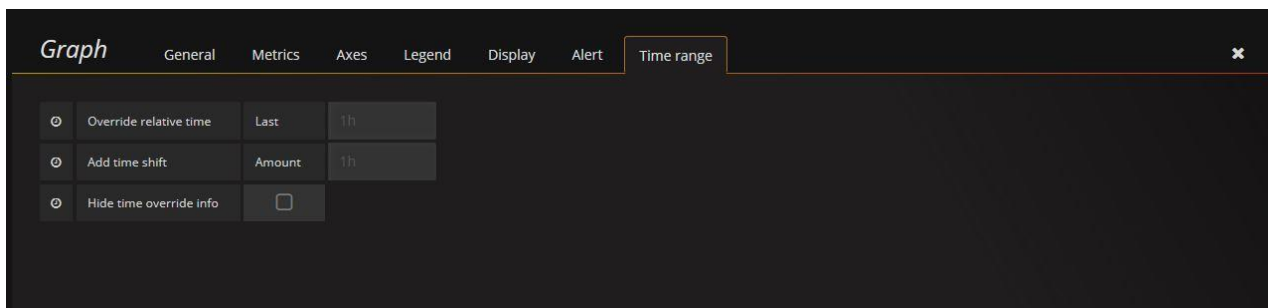
Εικόνα 6-48: Graph axes tab

Εδώ μπορούμε να παραμετροποιήσουμε τους άξονες όπως όνομα, μονάδες μέτρησης, κλίμακα.



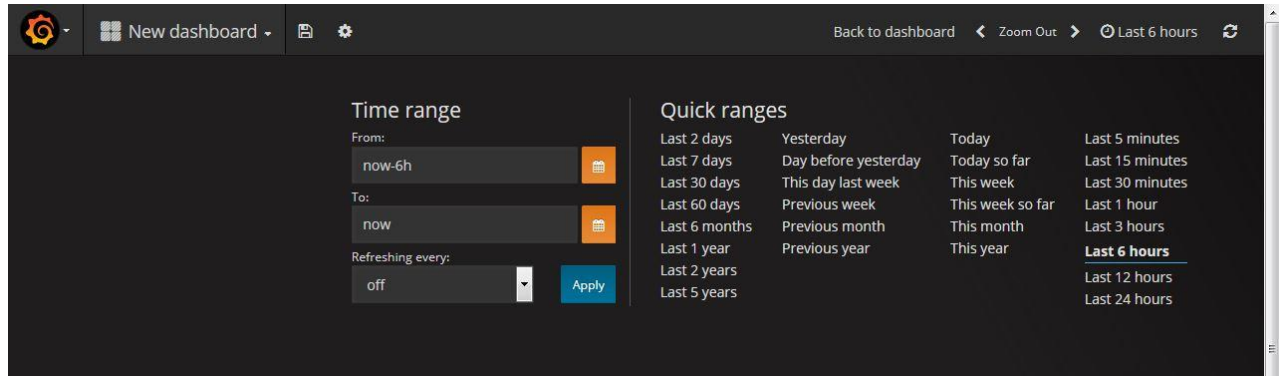
Εικόνα 6-49: Graph display tab

Σε αυτήν την καρτέλα μας δίνεται η δυνατότητα να αλλάξουμε τον τρόπο απεικόνισης του γραφήματος.



Εικόνα 6-50: Graph time range tab

Εδώ έχουμε τη δυνατότητα να αλλάξουμε τον χρόνο ανανέωσης του γραφήματος ανεξάρτητα από το dashboard. Εάν δεν το πειράξουμε τότε το γράφημα θα ακολουθεί το χρόνο που έχουμε ορίσει στο dashboard.



Εικόνα 6-51: Dashboard update time

Αυτή η επιλογή εμφανίζεται μόλις πατήσουμε το με το ρολόι πάνω δεξιά και ουσιαστικά εδώ ρυθμίζουμε το διάστημα που θέλουμε τα γραφήματα μας να δείξουν και το εάν θέλουμε να ανανεώνονται δυναμικά και σε κάθε πόσο χρόνο.

Παρόμοιες είναι οι ρυθμίσεις και για τους υπόλοιπους τύπους γραφημάτων αλλά αυτές είναι οι βασικότερες που υπάρχουν σε όλα τα γραφήματα.

Για την ηλεκτρονική πύλη χρησιμοποιήσαμε τα παρακάτω ερωτήματα στην καρτέλα metrics. Κάθε ερώτημα αποτελεί και ξεχωριστό γράφημα.

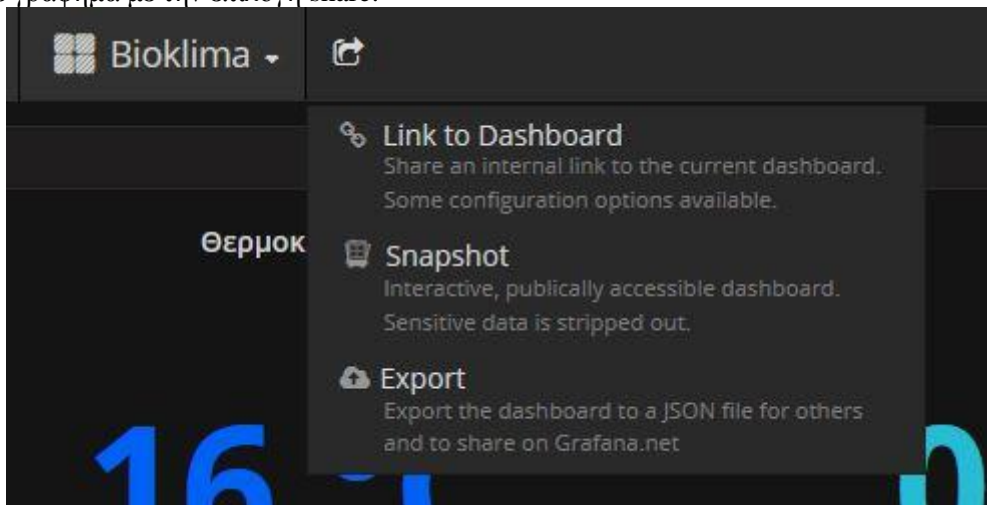
```
SELECT load from metrics where host='bioklima' and type='degc'
```

```
SELECT load from metrics where host='bioklima' and type='rain'
```

```
SELECT load from metrics where host='bioklima' and type='winddir'
```

```
SELECT load from metrics where host='bioklima' and type='humidity'
```

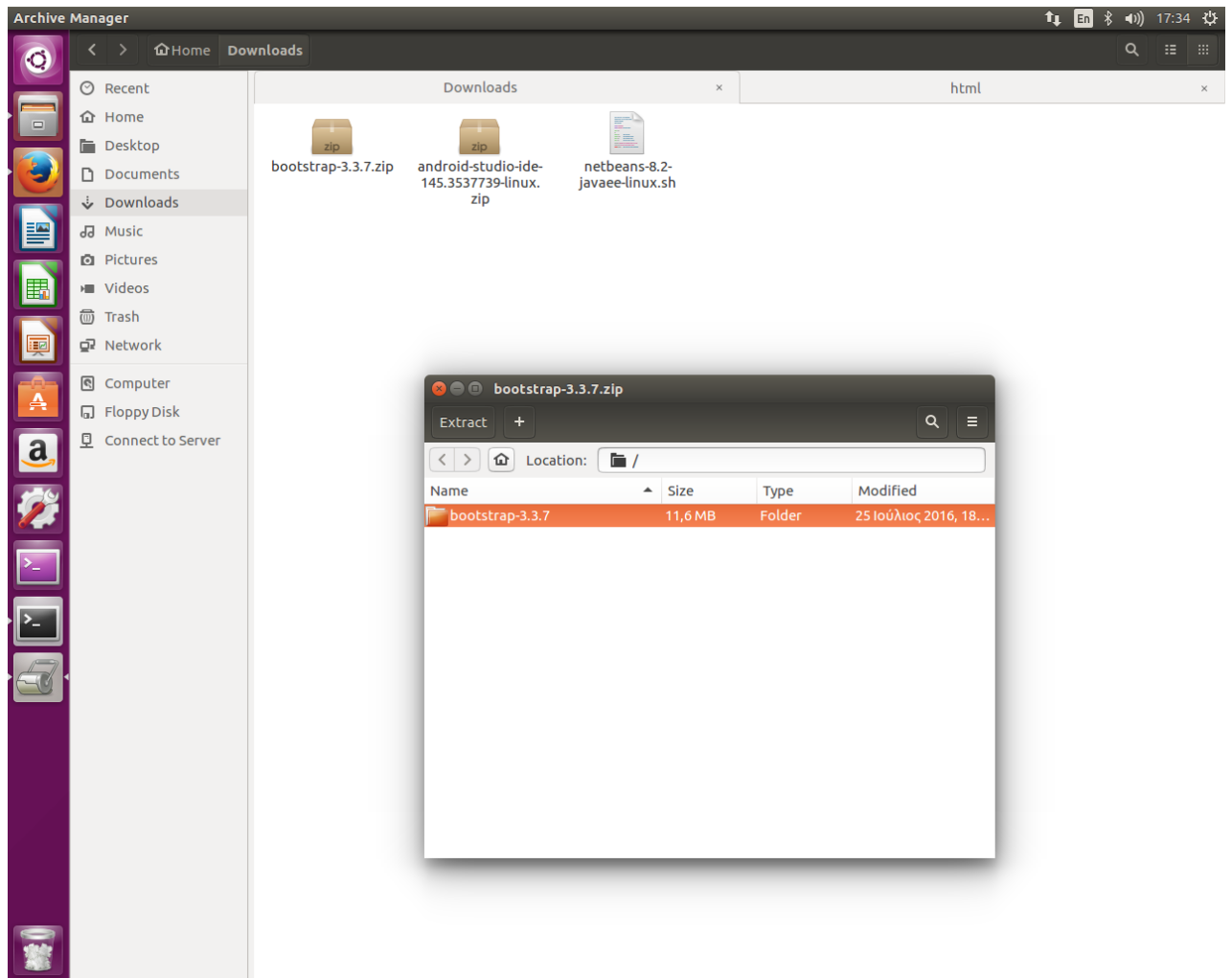
Μια ακόμα μία σημαντική λειτουργία του Grafana είναι ότι μοιραστόμε το Dashboard ή το κάθε γράφημα με την επιλογή share.



Εικόνα 6-52: Grafana share option

## 6.11. Δημιουργία ηλεκτρονικής πύλης με το Bootstrap

Για να φτιάξουμε το front end της ηλεκτρονικής πύλης χρειαζόμαστε το αρχείο που κατεβάσαμε από την ιστοσελίδα του Bootstrap. Το αρχείο λογικά βρίσκεται στην θέση /home/user/Downloads και το αποσυμπιέζουμε.



Εικόνα 6-53: Bootstrap extract

Αφού κάνουμε αποσυμπίεση αντιγράφουμε στην τοποθεσία /var/www/html τα περιεχόμενα των φακέλων ~/bootstrap/dist (css.fonts.js)

~/bootstrap/js (τα προσθέτουμε στο φάκελο js που μόλις αντιγράψαμε)

~/bootstrap/docs/example/carousel (το αρχείο carousel.css στο φάκελο css και carousel.html μετονομάζεται σε index.html)

Επίσης φτιάχνουμε και ένα φάκελο ~/html/src/images για να βάλουμε τις εικόνες, ένα αρχείο στο φάκελο ~/html/js/ που θα το ονομάσουμε grafanaServerIP.js, στο φάκελο ~/html δυο αρχεία με όνομα about.html και bioklima\_dashboard.html και τέλος φτιαχνουμε ένα φακελο με το ονομα ark οπου εκει θα βαλουμε το αρχειο της εφαρμογης android.

Στο αρχείο grafanaServerIP.js γραφούμε την IP address του server:

```
function ServerIP(){
    var ip = new String("192.168.1.250");
    return ip;
}
```

Στο αρχείο index.html γραφούμε:

```
<!DOCTYPE html>
<html lang="en">

<head>
```

```

<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<!-- The above 3 meta tags *must* come first in the head; any other
head content must come *after* these tags -->
<meta name="description" content="tei weather info">
<meta name="author" content="mangas">
<link rel="icon" href="src/images/favicon.gif">

<title>Μετεωρολογικός σταθμός TEI Κρήτης</title>
<!--Get the Grafana Server ip address-->
<script Language="JavaScript" type="text/javascript"
src="js/grafanaServerIP.js"></script>
<script Language="JavaScript" type="text/javascript"
src="js/frameRedirect.js"></script>

<!-- Bootstrap core CSS -->
<link href="css/bootstrap.min.css" rel="stylesheet">

<!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->
<link href="css/ie10-viewport-bug-workaround.css" rel="stylesheet">

<!-- Just for debugging purposes. Don't actually copy these 2 lines! --
>
<!--[if lt IE 9]><script src="../../assets/js/ie8-responsive-file-
warning.js"></script><![endif]-->
<script src="../../assets/js/ie-emulation-modes-warning.js"></script>

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and
media queries -->
<!--[if lt IE 9]>
<script
src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->

<!-- Custom styles for this template -->
<link href="css/carousel.css" rel="stylesheet">

</head>

<!-- NAVBAR
===== -->
<body>
<div class="navbar-wrapper">
<div class="container">

<nav class="navbar navbar-inverse navbar-static-top">
<div class="container">
<div class="navbar-header">
<button type="button" class="navbar-toggle collapsed" data-
toggle="collapse" data-target="#navbar" aria-expanded="false" aria-
controls="navbar">
<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>

```

```

    <a class="navbar-brand" href="#">Portal</a>
</div>
<div id="navbar" class="navbar-collapse collapse">
  <ul class="nav navbar-nav">
    <li class="active"><a href="#"> Αρχική</a></li>
    <li><a href="about.html"> Σχετικά με τη σελίδα</a></li>
    <li><a
href="http://www.teicrete.gr/el/tei/8528"target="_blank">
Επικοινωνία</a></li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-
toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false">
Μενού ☰ <span class="caret"></span></a>
      <ul class="dropdown-menu">
        <li class="dropdown-header"> Γραφήματα
</li>
        <li><a href="bioklima_dashboard.html">Bioklima</a></li>
        <li role="separator" class="divider"></li>
        <li class="dropdown-header"> Δεδομένα </li>
        <li><a
href="http://bioklima5.prv3.teiher.gr/xls/"target="_blank"> Δεδομένα σε
excel </a></li>
        <li><a
href="http://bioklima5.prv3.teiher.gr/wsn/reports.php"target="_blank">Excel
και CSV</a></li>
        <li role="separator" class="divider"></li>
        <li><a href="./apk/weather.apk"
downlad="Weather.apk">Λήψη εφαρμογής για android</a></li>
      </ul>
    </li>
  </ul>
</div>
</div>
</nav>

</div>
</div>

<!-- Carousel
===== -->
<div id="myCarousel" class="carousel slide" data-ride="carousel">
  <!-- Indicators -->
  <ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0"
class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>
  <div class="carousel-inner" role="listbox">
    <div class="item active">
      
    <div class="container">
      <div class="carousel-caption">
        <h1></h1>
        <p></p>

```

```

        <p><a class="btn btn-lg btn-primary"
href="http://www.epp.teicrete.gr/" role="button" target="_blank">Τμήμα
μηχανικών πληροφορικής ΤΕΙ Κρήτης </a></p>
    </div>
</div>
</div>
<div class="item">
    
    <div class="container">
        <div class="carousel-caption">
            <!--<h1>Another example headline.</h1>
            <p>Cras justo odio, dapibus ac facilisis in, egestas eget
quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id
nibh ultricies vehicula ut id elit.</p>
            <p><a class="btn btn-lg btn-primary" href="#"
role="button">Learn more</a></p>-->
        </div>
    </div>
</div>
<div class="item">
    
    <div class="container">
        <div class="carousel-caption">
            <!--<h1>One more for good measure.</h1>
            <p>Cras justo odio, dapibus ac facilisis in, egestas eget
quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id
nibh ultricies vehicula ut id elit.</p>
            <p><a class="btn btn-lg btn-primary" href="#"
role="button">Browse gallery</a></p>-->
        </div>
    </div>
</div>
</div>
<a class="left carousel-control" href="#myCarousel" role="button"
data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left" aria-
hidden="true"></span>
    <span class="sr-only"> Προηγούμενη </span>
</a>
<a class="right carousel-control" href="#myCarousel" role="button"
data-slide="next">
    <span class="glyphicon glyphicon-chevron-right" aria-
hidden="true"></span>
    <span class="sr-only">Επόμενη</span>
</a>
</div><!-- /.carousel -->

<!-- Marketing messaging and featurettes
===== -->
<!-- Wrap the rest of the page in another container to center all the
content. -->

<div class="container marketing">

    <!-- Three columns of text below the carousel -->
    <div class="row">

```

```

<div class="col-lg-4">
  
  <h2> ΤΕΙ Κρήτης </h2>
  <p> Κεντρική σελίδα του ΤΕΙ Κρήτης ,</p>
  <p><a class="btn btn-default" href="http://www.teicrete.gr"
target="_blank" role="button"> Πατήστε εδώ &raquo;</a></p>
</div><!-- /.col-lg-4 -->
<div class="col-lg-4">
  <a href="http://www.grafana.org" target="_blank"></a>
  <h2> Γραφήματα</h2>
  <p> Δείτε εδώ τα γραφήματα για τον καιρό, αναλυτικότερα!</p>
  <p><a class="btn btn-default" href="bioklima_dashboard.html"
role="button"> Πατήστε εδώ &raquo;</a></p>
</div><!-- /.col-lg-4 -->
<div class="col-lg-4">
  
  <h2> Τμήμα μηχανικών πληροφορικής</h2>
  <p><a class="btn btn-default" href="http://www.epp.teicrete.gr/"
role="button" target="_blank" role="button"> Πατήστε εδώ &raquo;</a></p>
</div><!-- /.col-lg-4 -->
</div><!-- /.row -->

<!-- START THE FEATURETTES -->

<hr class="featurette-divider">

<div class="row featurette">
  <div class="col-md-7">
    <!--img class="featurette-image img-responsive center-block"
data-src="holder.js/500x500/auto" alt="Generic placeholder image"-->
    <iframe id="Celsius" src="" width="100%" height="350"
frameborder="0"></iframe>
    <script>
      var ip = ServerIP();

      document.getElementById("Celsius").setAttribute('src',"http://" + ip
+":3000/dashboard-solo/db/bioklima?panelId=8&fullscreen");
    </script>
  </div>
  <div class="col-md-5">
    <h2 class="featurette-heading"> Τι θερμοκρασία έχουμε; <span
class="text-muted"></span></h2>
    <p class="lead"> Η διακύμανση της θερμοκρασίας σήμερα. </p>
  </div>
</div>

<hr class="featurette-divider">

<div class="row featurette">
  <div class="col-md-7 col-md-push-5" href="http://www.w3schools.com"
target="Rain">
    <iframe id="Rain" src="JavaScript:'content'"
width="100%" height="350" frameborder="0"></iframe>
    <script>

```

```

        var ip = ServerIP();

        document.getElementById("Rain").setAttribute('src',"http://" + ip
+ ":3000/dashboard-solo/db/bioklima?panelId=9&fullscreen");
        </script>

    </div>
    <div class="col-md-5 col-md-pull-7">
        <h2 class="featurette-heading">Πόσο έβρεξε; <span class="text-
muted"></span></h2>
        <p class="lead">Αναλυτικά πόσο έχει βρέξει. </p>
    </div>
</div>

<hr class="featurette-divider">
<div class="col-md-7">
    <iframe name="Wind" id="Wind" src="" width="100%"
height="350" frameborder="0" onload="JavaScript:'content'"></iframe>
    <script>
        var ip = ServerIP();

        document.getElementById("Wind").setAttribute('src',"http://" + ip
+ ":3000/dashboard-solo/db/bioklima?panelId=10&fullscreen");
    </script>
    </div>
    <div class="col-md-5">
        <h2 class="featurette-heading">Πόσο φυσάει σήμερα; <span
class="text-muted"></span></h2>
        <p class="lead">Εδώ απεικονίζονται οι τιμές της ταχύτητας του
άνεμου.</p>
    </div>
</div>

<hr class="featurette-divider">

<!-- /END THE FEATURETTES -->

<!-- FOOTER -->
<footer>
    <p class="pull-right"><a href="#">Επιστροφή στην κορυφή </a></p>
    <p>&copy; 2016 TEI Κρήτης, &middot; <a
href="http://www.teicrete.gr/en/tei/13826"target="_blank">Cookies
policy</a> &middot; <a
href="http://www.teicrete.gr/el/tei/12625"target="_blank">Όροι
Χρήσης</a></p>
</footer>

</div><!-- /.container -->

<!-- Bootstrap core JavaScript
===== -->
<!-- Placed at the end of the document so the pages load faster -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></s
cript>
<script>window.jQuery || document.write('<script
src="js/tests/jquery.min.js"></script>')</script>

```



```

<script src="js/bootstrap.min.js"></script>
<!-- Just to make our placeholder images work. Don't actually copy the
next line! -->
<script src="js/holder.min.js"></script>
<!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->
<script src="js/ie10-viewport-bug-workaround.js"></script>
</body>
</html>

```

### Στο αρχείο `bioklima_dashboard.html` γράφουμε:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other
head content must come *after* these tags -->
    <meta name="description" content="dashboard">
    <meta name="author" content="mangas">
    <link rel="icon" href="src/images/favicon.gif">

    <title> Μετεωρολογικός σταθμός TEI Κρήτης </title>

    <!--Get the Grafana Server ip address-->
    <script Language="JavaScript" type="text/javascript"
src="js/grafanaServerIP.js"></script>
    <!-- Bootstrap core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->
    <link href="../../assets/css/ie10-viewport-bug-workaround.css"
rel="stylesheet">

    <!-- Just for debugging purposes. Don't actually copy these 2 lines! --
>
    <!--[if lt IE 9]><script src="../../assets/js/ie8-responsive-file-
warning.js"></script><![endif]-->
    <script src="../../assets/js/ie-emulation-modes-warning.js"></script>

    <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and
media queries -->
    <!--[if lt IE 9]>
      <script
src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
      <script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->

    <!-- Custom styles for this template -->
    <link href="css/carousel.css" rel="stylesheet">

  </head>
  <!-- NAVBAR
===== -->
  <body>
    <div class="navbar-wrapper">
      <div class="container">

```

```

<nav class="navbar navbar-inverse navbar-static-top">
  <div class="container">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse" data-target="#navbar" aria-expanded="false" aria-
controls="navbar">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="index.html">Portal</a>
    </div>
    <div id="navbar" class="navbar-collapse collapse">
      <ul class="nav navbar-nav">
        <li class=""><a href="index.html">Αρχική</a></li>
        <li><a href="about.html">Σχετικά με τη σελίδα</a></li>
        <li><a
href="http://www.teicrete.gr/el/tei/8528"target="_blank">Επικοινωνία</a></l
i>
          <li class="dropdown">
            <a href="#" class="dropdown-toggle" data-
toggle="dropdown" role="button" aria-haspopup="true" aria-
expanded="false">Μενού <span class="caret"></span></a>
            <ul class="dropdown-menu">
              <li class="dropdown-header">Γραφήματα</li>
              <li class="active"><a href="#">Bioklima</a></li>
              <li><a href="#">Agro Lab</a></li>
              <li role="separator" class="divider"></li>
              <li class="dropdown-header">Δεδομένα</li>
              <li><a
href="http://bioklima5.prv3.teiher.gr/xls/"target="_blank">Δεδομένα σε
excel</a></li>
              <li><a
href="http://bioklima5.prv3.teiher.gr/wsn/reports.php"target="_blank">Excel
και CSV</a></li>
              <li role="separator" class="divider"></li>
              <li><a href="./apk/weather.apk"
download="Weather.apk">Λήψη εφαρμογής για android</a></li>
            </ul>
          </li>
        </ul>
      </div>
    </div>
  </nav>
</div>
</div>

<!-- Carousel
===== -->
<div id="myCarousel" class="carousel slide" data-ride="carousel">
  <!-- Indicators -->
  <ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0"
class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>

```

```

    <li data-target="#myCarousel" data-slide-to="2"></li>
</ol>
<div class="carousel-inner" role="listbox">
  <div class="item active">
    
    <div class="container">
      <div class="carousel-caption">
        <h1></h1>
        <p></p>
        <p><a class="btn btn-lg btn-primary"
href="http://www.epp.teicrete.gr/" role="button" target="_blank">Τμήμα
μηχανικών πληροφορικής του ΤΕΙ</a></p>
      </div>
    </div>
  </div>
  <div class="item">
    
    <div class="container">
      <div class="carousel-caption">
        <!--<h1>Another example headline.</h1>
        <p>Cras justo odio, dapibus ac facilisis in, egestas eget
quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id
nibh ultricies vehicula ut id elit.</p>
        <p><a class="btn btn-lg btn-primary" href="#"
role="button">Learn more</a></p>-->
      </div>
    </div>
  </div>
  <div class="item">
    
    <div class="container">
      <div class="carousel-caption">
        <!--<h1>One more for good measure.</h1>
        <p>Cras justo odio, dapibus ac facilisis in, egestas eget
quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id
nibh ultricies vehicula ut id elit.</p>
        <p><a class="btn btn-lg btn-primary" href="#"
role="button">Browse gallery</a></p>-->
      </div>
    </div>
  </div>
  </div>
  <a class="left carousel-control" href="#myCarousel" role="button"
data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left" aria-
hidden="true"></span>
    <span class="sr-only">Previous</span>
  </a>
  <a class="right carousel-control" href="#myCarousel" role="button"
data-slide="next">
    <span class="glyphicon glyphicon-chevron-right" aria-
hidden="true"></span>
    <span class="sr-only">Next</span>
  </a>
</div><!-- /.carousel -->

```

```

<!-- Dashboard
===== -->
<!-- Wrap the rest of the page in another container to center all the
content. -->
    <div class="container-fluid">
        <div class="row">
            <!-- 16:9 aspect ratio -->
            <div class="col-sl-7" >
                <!--if size od dashboard changes height should also
change-->
                    <iframe id="DashBioklima" class="embed-responsive-
item" width="100%" height="1800"src=""></iframe>
                    <script>
                        var ip = ServerIP();

                        document.getElementById("DashBioklima").setAttribute('src',"http://"
+ ip + ":3000/dashboard/db/bioklima");
                    </script>
                </div>
            </div>

            <hr class="featurette-divider">

            <!-- /END THE Dashboard -->

            <!-- FOOTER -->
            <footer>
                <p class="pull-right"><a href="#">Επιστροφή στην κορυφή</a></p>
                <p>&copy; 2017 TEI Κρήτης &middot; <a
href="http://www.teicrete.gr/en/tei/13826"target="_blank">Cookies
policy</a> &middot; <a
href="http://www.teicrete.gr/el/tei/12625"target="_blank">Όροι
χρήσης</a></p>
            </footer>

        </div><!-- /.container -->

        <!-- Bootstrap core JavaScript
===== -->
        <!-- Placed at the end of the document so the pages load faster -->
        <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></s
cript>
        <script>window.jQuery || document.write('<script
src="js/tests/jquery.min.js"></script>')</script>
        <script src="js/bootstrap.min.js"></script>
        <!-- Just to make our placeholder images work. Don't actually copy the
next line! -->
        <script src="js/holder.min.js"></script>
        <!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->
        <script src="js/ie10-viewport-bug-workaround.js"></script>

    </body>
</html>

```

**Και τέλος στο αρχείο about.html γραφούμε:**

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other
head content must come *after* these tags -->
    <meta name="description" content="tei weather info">
    <meta name="author" content="mangas">
    <link rel="icon" href="src/images/favicon.gif">

    <title> Μετεωρολογικός σταθμός TEI Κρήτης </title>

    <!-- Bootstrap core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->
    <link href="../../assets/css/ie10-viewport-bug-workaround.css"
rel="stylesheet">

    <!-- Just for debugging purposes. Don't actually copy these 2 lines! --
>
    <!--[if lt IE 9]><script src="../../assets/js/ie8-responsive-file-
warning.js"></script><![endif]-->
    <script src="../../assets/js/ie-emulation-modes-warning.js"></script>

    <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and
media queries -->
    <!--[if lt IE 9]>
      <script
src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
      <script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->

    <!-- Custom styles for this template -->
    <link href="css/carousel.css" rel="stylesheet">

  </head>
  <!-- NAVBAR
===== -->
  <body>
    <div class="navbar-wrapper">
      <div class="container">

        <nav class="navbar navbar-inverse navbar-static-top">
          <div class="container">
            <div class="navbar-header">
              <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse" data-target="#navbar" aria-expanded="false" aria-
controls="navbar">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
              </button>
              <a class="navbar-brand" href="index.html">Portal</a>
            </div>

```

```

<div id="navbar" class="navbar-collapse collapse">
  <ul class="nav navbar-nav">
    <li class=""><a href="index.html">Αρχική</a></li>
    <li class="active"><a href="#about">Σχετικά με τη
σελίδα</a></li>
    <li><a
href="http://www.teicrete.gr/el/tei/8528"target="_blank">Επικοινωνία</a></l
i>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-
toggle="dropdown" role="button" aria-haspopup="true" aria-
expanded="false">Μενού <span class="caret"></span></a>
      <ul class="dropdown-menu">
        <li class="dropdown-header">Γραφήματα</li>
        <li><a href="bioklima_dashboard.html">Bioklima</a></li>
        <li><a href="#">Agro Lab</a></li>
        <li role="separator" class="divider"></li>
        <li class="dropdown-header">Δεδομένα</li>
        <li><a
href="http://bioklima5.prv3.teiher.gr/xls/"target="_blank">Δεδομένα σε
excel</a></li>
        <li><a
href="http://bioklima5.prv3.teiher.gr/wsn/reports.php"target="_blank">Excel
και CSV</a></li>
        <li role="separator" class="divider"></li>
        <li><a href="./apk/weather.apk"
downlad="Weather.apk">Λήψη εφαρμογής για android</a></li>
      </ul>
    </li>
  </ul>
</div>
</div>
</nav>

</div>
</div>

<!-- Carousel
===== -->
<div id="myCarousel" class="carousel slide" data-ride="carousel">
  <!-- Indicators -->
  <ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0"
class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>
  <div class="carousel-inner" role="listbox">
    <div class="item active">
      
    <div class="container">
      <div class="carousel-caption">
        <h1></h1>
        <p></p>
        <p><a class="btn btn-lg btn-primary"
href="http://www.epp.teicrete.gr/" role="button" target="_blank">Τμήμα
μηχανικών πληροφορικής του ΤΕΙ Κρήτης</a></p>

```

```

        </div>
    </div>
</div>
<div class="item">
    
    <div class="container">
        <div class="carousel-caption">
            <!--<h1>Another example headline.</h1>
            <p>Cras justo odio, dapibus ac facilisis in, egestas eget
quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id
nibh ultricies vehicula ut id elit.</p>
            <p><a class="btn btn-lg btn-primary" href="#"
role="button">Learn more</a></p-->
        </div>
    </div>
</div>
<div class="item">
    
    <div class="container">
        <div class="carousel-caption">
            <!--<h1>One more for good measure.</h1>
            <p>Cras justo odio, dapibus ac facilisis in, egestas eget
quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id
nibh ultricies vehicula ut id elit.</p>
            <p><a class="btn btn-lg btn-primary" href="#"
role="button">Browse gallery</a></p-->
        </div>
    </div>
</div>
</div>
<a class="left carousel-control" href="#myCarousel" role="button"
data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left" aria-
hidden="true"></span>
    <span class="sr-only">Previous</span>
</a>
<a class="right carousel-control" href="#myCarousel" role="button"
data-slide="next">
    <span class="glyphicon glyphicon-chevron-right" aria-
hidden="true"></span>
    <span class="sr-only">Next</span>
</a>
</div><!-- /.carousel -->

<!-- Marketing messaging and featurettes
===== -->
<!-- Wrap the rest of the page in another container to center all the
content. -->

<div class="container marketing">

    <!-- START THE FEATURETTES -->

    <hr class="featurette-divider">

    <div class="row featurette">

```

```

    <div class="col-md-5">
      <!--img class="featurette-image img-responsive center-block"
data-src="holder.js/500x500/auto" alt="Generic placeholder image"-->
      <iframe href="http://www.epp.teicrete.gr" target="_blank"
src="src/images/epp_logo.png" width="300" height="300"
frameborder="0"></iframe>
    </div>
    <div class="col-md-7">
      <h2 class="featurette-heading">Σχετικά με αυτή τη σελίδα<span
class="text-muted"></span></h2>
      <p class="lead">Ο σκοπός αυτής της σελίδας είναι να παρουσιάσει
τα δεδομένα που συλλέγονται από τους αισθητήρες του ΤΕΙ, σε γραφήματα. Η
ιστοσελίδα έχει δημιουργηθεί με <a href="http://getbootstrap.com/"
target="_blank">Bootstrap</a> και <a href="http://www.grafana.org"
target="_blank">Grafana</a>. </p>
    </div>
  </div>

  <hr class="featurette-divider">

  <!-- /END THE FEATURETTES -->

  <!-- FOOTER -->
  <footer>
    <p class="pull-right"><a href="#">Επιστροφή στην κορυφή</a></p>
    <p>&copy; 2017 ΤΕΙ Κρήτης &middot; <a
href="http://www.teicrete.gr/en/tei/13826"target="_blank">Cookies
policy</a> &middot; <a
href="http://www.teicrete.gr/el/tei/12625"target="_blank">Όροι
χρήσης</a></p>
  </footer>

</div><!-- /.container -->

<!-- Bootstrap core JavaScript
===== -->
<!-- Placed at the end of the document so the pages load faster -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></s
cript>
  <script>window.jQuery || document.write('<script
src="js/tests/jquery.min.js"></script>')</script>
  <script src="js/bootstrap.min.js"></script>
  <!-- Just to make our placeholder images work. Don't actually copy the
next line! -->
  <script src="js/holder.min.js"></script>
  <!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->
  <script src="js/ie10-viewport-bug-workaround.js"></script>
</body>
</html>

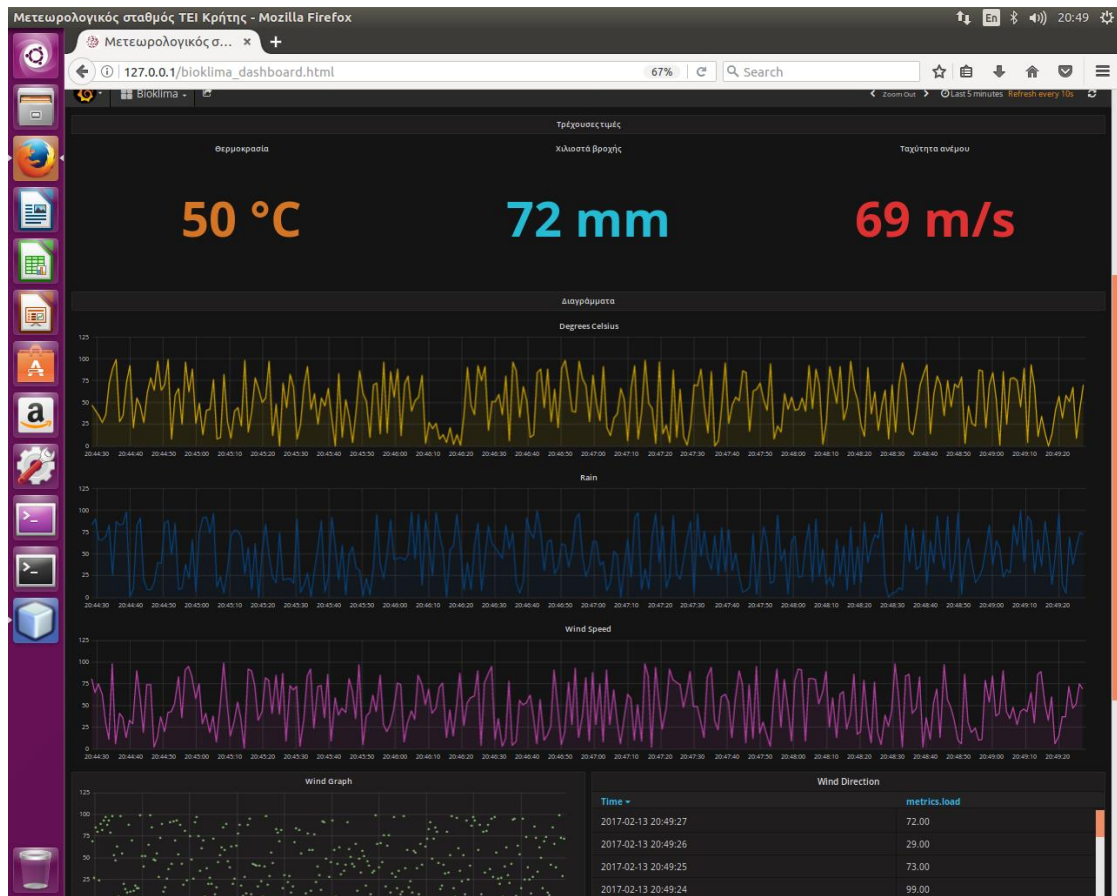
```



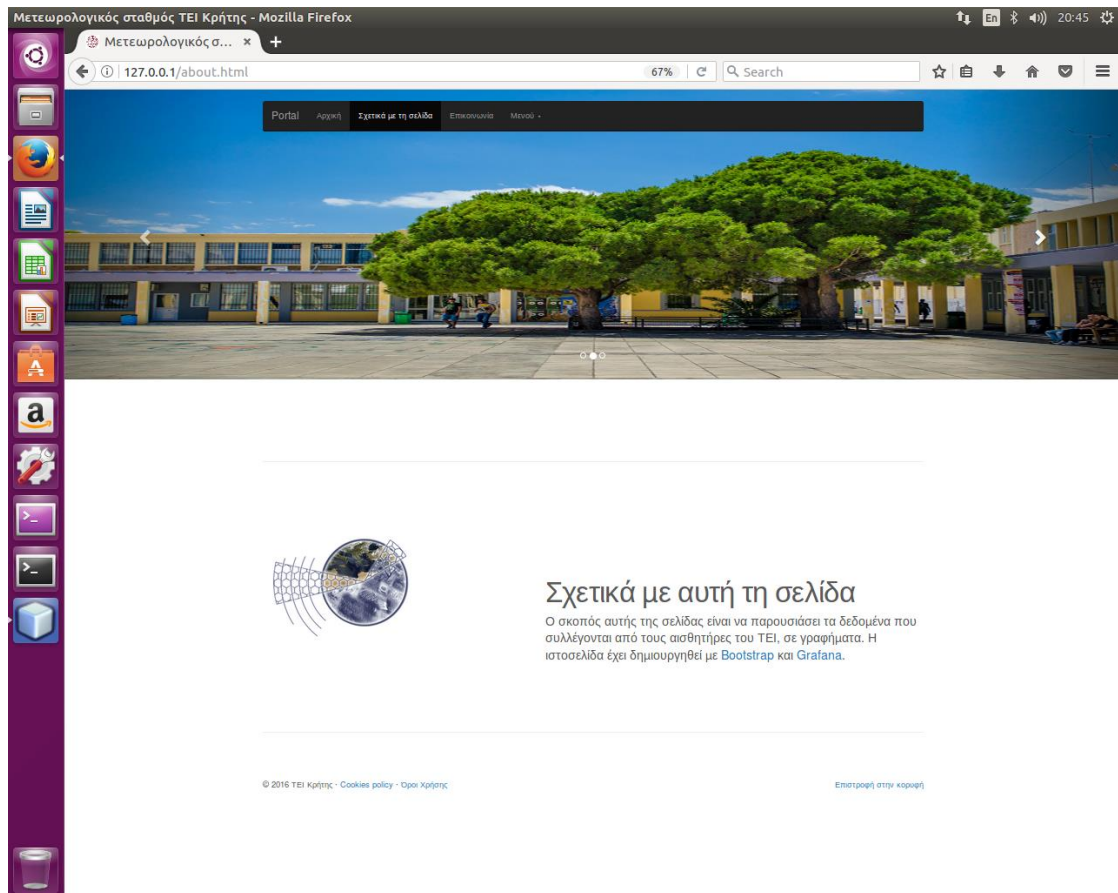
Το αποτέλεσμα θα πρέπει να είναι κάπως έτσι:

The screenshot shows a web browser displaying a weather application. The browser's address bar shows the URL `127.0.0.1/index.html`. The application features a navigation menu with 'Portal', 'Αρχική', 'Σχετικά με τη σελίδα', 'Επικοινωνία', and 'Μενού'. The main content area includes a large banner image of a university building and a navigation bar with three items: 'ΤΕΙ Κρήτης', 'Γραφήματα', and 'Τμήμα μηχανικών πληροφορικής'. Below this, there are three data cards: 1) 'Θερμοκρασία' (Temperature) showing **56 °C** with the text 'Τι θερμοκρασία έχουμε; Η διακύμανση της θερμοκρασίας σήμερα.' 2) 'Χιλιοστά βροχής' (Precipitation) showing **31 mm** with the text 'Πόσο έβρεξε; Αναλυτικά πόσο έχει βρέξει.' 3) 'Χιλιοστά βροχής' (Wind Speed) showing **31 mm** with the text 'Πόσο φυσάει σήμερα; Εδώ απεικονίζονται οι τιμές της ταχύτητας του άνεμου.' The footer contains copyright information for 2017 TUI of Crete and a link to the privacy policy.

Εικόνα 6-54: Portal



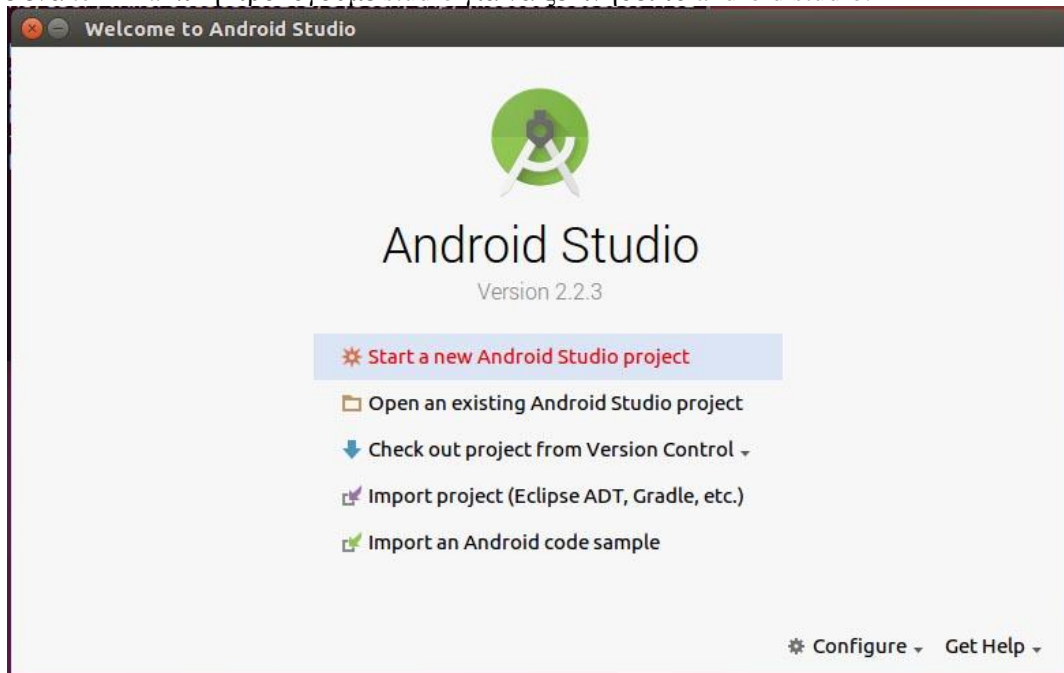
Εικόνα 6-55: Portal bioklima page



Εικόνα 6-56: Portal about page

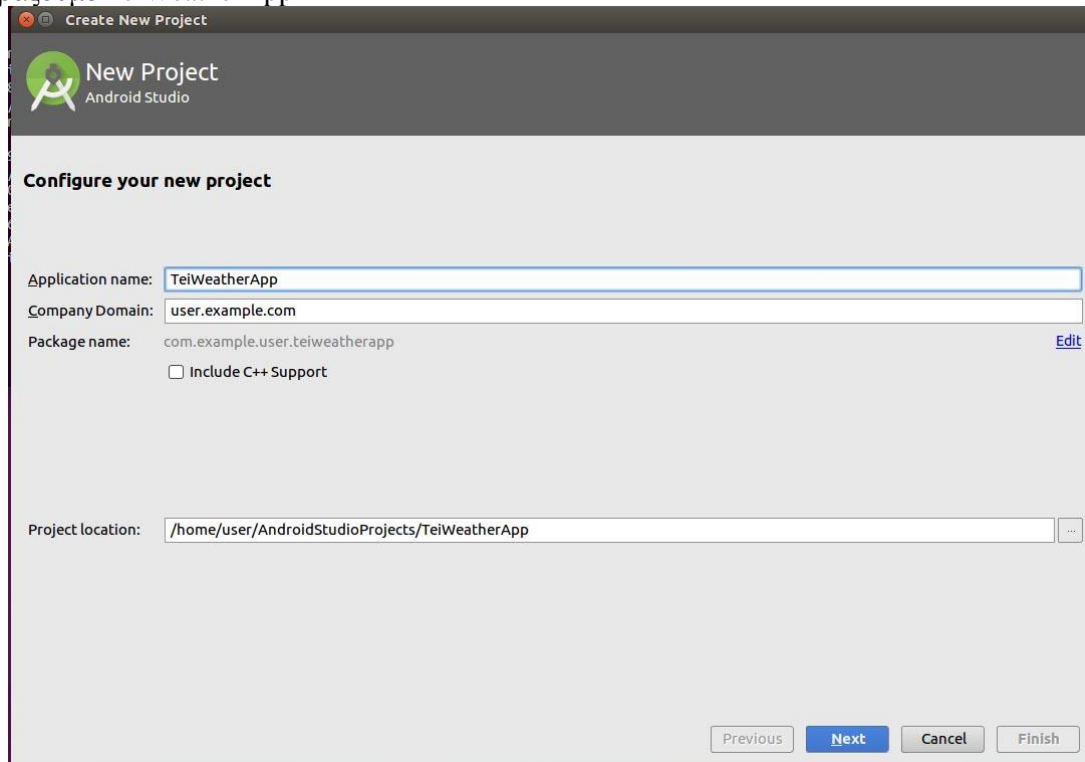
## 6.12. Δημιουργία εφαρμογής για κινητούς χρήστες με το Android Studio

Σε ένα terminal πληκτρολογούμε studio για να ξεκινήσει το android studio.



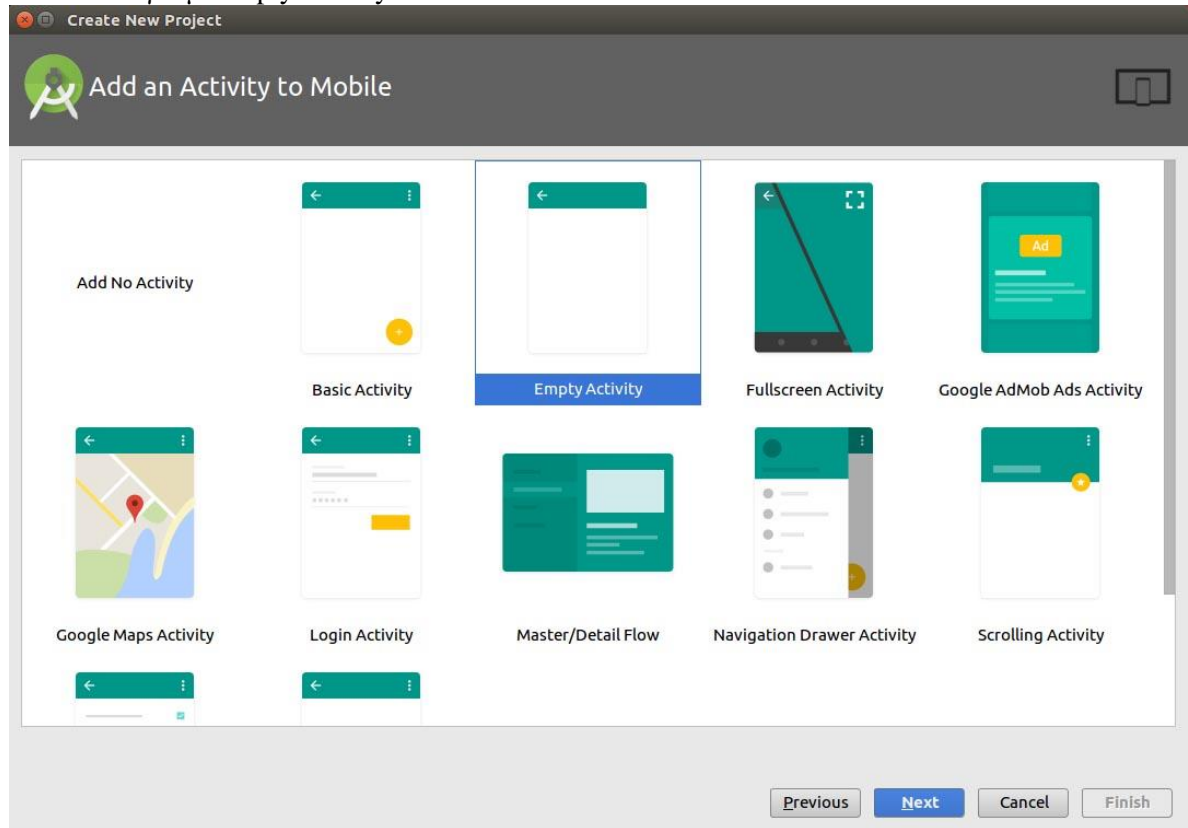
Εικόνα 6-57: Android Studio App

Επιλέγουμε να φτιάξουμε ένα νέο android project και ξεκινάει ένας νέο οδηγός. Στο όνομα βάζουμε TeiWeatherApp



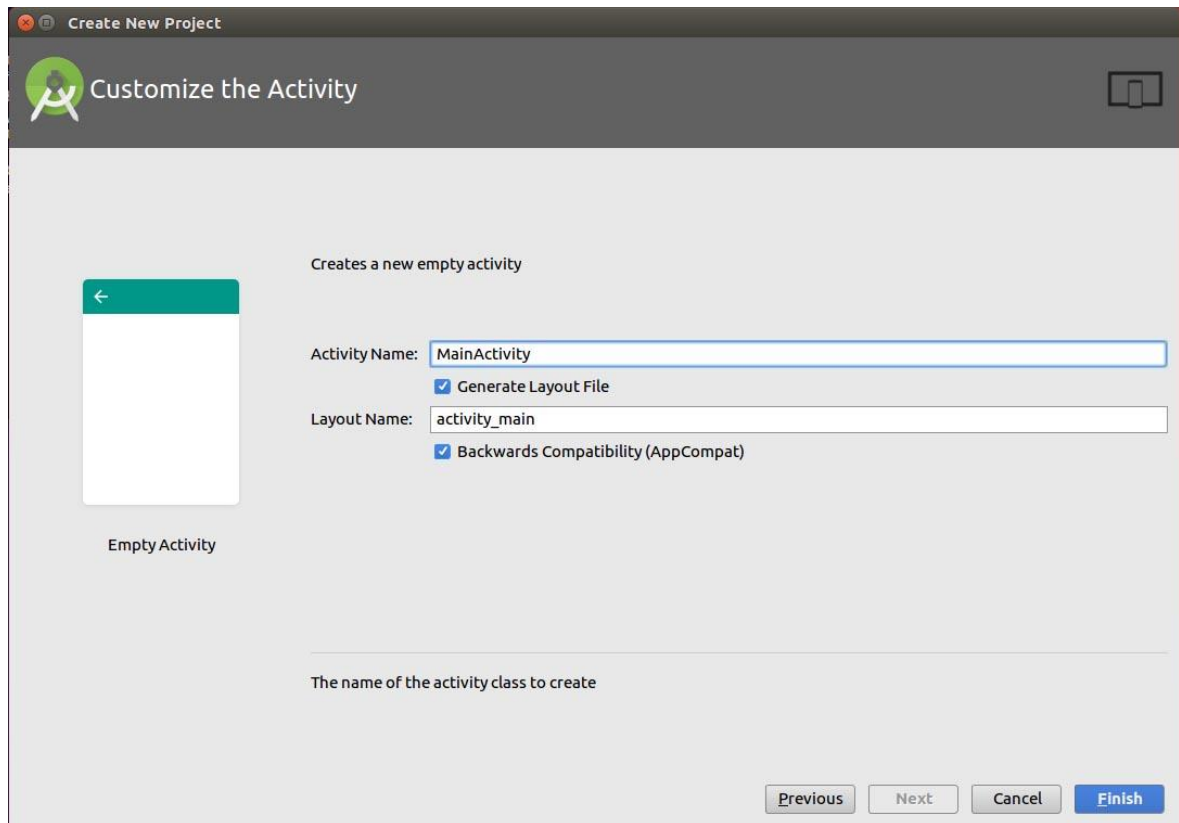
Εικόνα 6-58: Android project name

Μετά επιλέγουμε empty activity



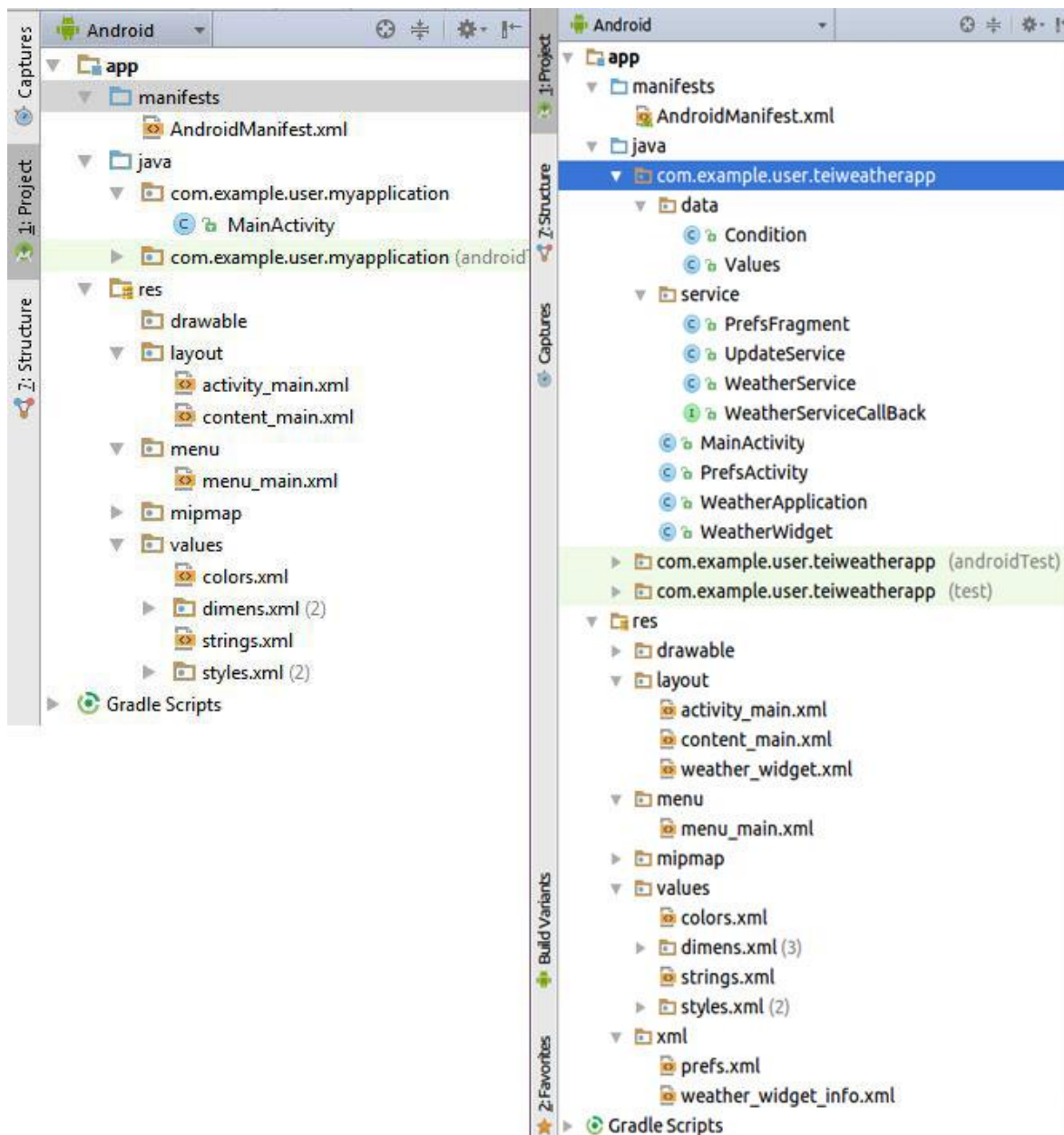
Εικόνα 6-59: Android activity choose

Και τελειώνοντας αφήνουμε το όνομα ως έχει και πατάμε finish



Εικόνα 6-60: Android activity name

Μόλις ανοίξει το νέο μας project θα πρέπει να τροποποιήσουμε τη δομή του όπως παρακάτω:



Εικόνα 6-61: Android project refactor

Στο αρχείο app/manifests/AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.user.teiweatherapp" >

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"/>

    <application
        android:allowBackup="true"
        android:icon="@drawable/sun"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme" >
        <activity
```

```

android:name="com.example.user.teiweatherapp.MainActivity"
    android:label="@string/app_name"
    android:theme="@style/AppTheme.NoActionBar" >
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category
android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</activity>

android:name="com.example.user.teiweatherapp.PrefsActivity"
    android:launchMode="singleTask" />

    <receiver
android:name="com.example.user.teiweatherapp.WeatherWidget"
    android:label="@string/widget_name">
    <intent-filter>
        <action
android:name="android.appwidget.action.APPWIDGET_UPDATE" />
        </intent-filter>

        <meta-data
            android:name="android.appwidget.provider"
            android:resource="@xml/weather_widget_info" />
    </receiver>
</application>

</manifest>

```

Στο αρχείο `app/java/com.example.user.teiweatherapp/data/Condition:`

```

package com.example.user.teiweatherapp.data;

import android.util.Log;

import java.util.Calendar;

/**
 * Created by PanagiotisGiannikakis on 10/2/2017.
 */
public class Condition {

    private int rain;
    private int wind;
    private int temperature;
    private String condition;
    private int id;
    private Calendar calendar;
    private int currentTime;

    public Condition(Values values) {
        this.rain = values.getLoaded_rain();
        this.wind = values.getLoaded_speed();
        this.temperature = values.getLoaded_celsius();
        this.condition = "unknown";
        id = 45;
        calendar = Calendar.getInstance();
        currentTime = calendar.get(Calendar.HOUR_OF_DAY);
        Log.d("Condition", "Hour of day " + currentTime);
    }

```



```

    if ((currentTime >= 8) && (currentTime <= 19)) {
        //if it is daytime
        if ((temperature >= 30) && (wind <= 3) && (rain <= 1)) {
            condition = "Hot";
            id = 37;
        } else if (((temperature >= 15) && (temperature <= 30) &&
(wind <= 3) && (rain <= 2))) {
            condition = "Nice Weather";
            id = 33;
        } else if (((temperature >= 15) && (temperature <= 30) &&
(wind >= 3) && (rain <= 2))) {
            condition = "Partly Windy";
            id = 25;
        } else if (((temperature >= 1) && (temperature <= 15) &&
(wind >= 3) && (rain >= 3))) {
            condition = "Cold,Partly Windy,Possible Rain";
            id = 1;
        } else if (((temperature <= 1) && (wind >= 3) && (rain >=
3))) {
            condition = "Cold,Partly Windy,Possible Snow";
            id = 14;
        }
    } else { // if it is night
        if ((temperature >= 30) && (wind <= 3) && (rain <= 1)) {
            condition = "Hot";
            id = 32;
        } else if (((temperature >= 15) && (temperature <= 30) &&
(wind <= 3) && (rain <= 2))) {
            condition = "Nice Weather";
            id = 34;
        } else if (((temperature >= 15) && (temperature <= 30) &&
(wind >= 3) && (rain <= 2))) {
            condition = "Partly Windy";
            id = 30;
        } else if (((temperature >= 1) && (temperature <= 15) &&
(wind >= 3) && (rain >= 3))) {
            condition = "Cold,Partly Windy,Possible Rain";
            id = 46;
        } else if (((temperature <= 1) && (wind >= 3) && (rain >=
3))) {
            condition = "Cold,Partly Windy,Possible Snow";
            id = 47;
        }
    }
}

public String getCondition() {
    return condition;
}

public int getId() {
    return id;
}
}

```

Στο αρχείο app/java/com.example.user.teiweatherapp/data/Values:  
**package** com.example.user.teiweatherapp.data;

```
/**
```

```

 * Created by PanagiotisGiannikakis on 5/1/2017.
  */
public class Values {
    private int loaded_celsius;
    private int loaded_rain;
    private int loaded_speed;

    public int getLoaded_celsius() {
        return loaded_celsius;
    }

    public void setLoaded_celsius(int loaded_celsius) {
        this.loaded_celsius = loaded_celsius;
    }

    public int getLoaded_rain() {
        return loaded_rain;
    }

    public void setLoaded_rain(int loaded_rain) {
        this.loaded_rain = loaded_rain;
    }

    public int getLoaded_speed() {
        return loaded_speed;
    }

    public void setLoaded_speed(int loaded_speed) {
        this.loaded_speed = loaded_speed;
    }
}
Στο αρχείο app/java/com.example.user.teiweatherapp/service/PrefsFragment:
package com.example.user.teiweatherapp.service;

import android.os.Bundle;
import android.preference.PreferenceFragment;

import com.example.user.teiweatherapp.R;

/**
 * Created by PanagiotisGiannikakis on 24/2/2016.
 */
public class PrefsFragment extends PreferenceFragment {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        addPreferencesFromResource(R.xml.prefs);
    }
}
Στο αρχείο app/java/com.example.user.teiweatherapp/service/UpdateService:
package com.example.user.teiweatherapp.service;

import android.app.Service;
import android.content.Intent;
import android.os.IBinder;
import android.support.annotation.Nullable;
import android.util.Log;

import com.example.user.teiweatherapp.WeatherApplication;

/**

```

```

* Created by PanagiotisGiannikakis on 28/3/2017. Need manifest
declaration
*/
public class UpdateService extends Service {
    private static final String TAG =
UpdateService.class.getSimpleName();
    private Updater updater;
    private WeatherApplication weatherApplication;
    private WeatherService updateWeatherService;

    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }

    @Override
    public void onCreate() {
        super.onCreate();
        updater = new Updater();
        weatherApplication = (WeatherApplication) getApplication();
        Log.d(TAG, "ServiceOnCreated");
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId)
{
        if (!weatherApplication.isServiceRunning()) {
            updater.start();
            weatherApplication.setServiceRunning(true);
        }
        Log.d(TAG, "ServiceOnStarted");
        //return super.onStartCommand(intent, flags, startId);
        return START_STICKY;
    }

    @Override
    public void onDestroy() {
        super.onDestroy();
        // check if thread is running,if so stop it!
        if (weatherApplication.isServiceRunning()) {
            updater.interrupt();
            updater = null;
        }
        Log.d(TAG, "ServiceOnDestroyed");
    }

    protected class Updater extends Thread {
        private static final long TIME = 60000;

        public Updater() {
            super("Updater");
        }

        @Override
        public void run() {
            while (weatherApplication.isServiceRunning()) {

```

```
        weatherApplication.setServiceRunning(true);
        Log.d(TAG, "running");
        updateWeatherService = new
WeatherService(weatherApplication);

        if (weatherApplication.checkNetworkConnection()) {
updateWeatherService.refreshWeather(weatherApplication.getIp(),
weatherApplication.getType());
        }
        // Sleep
        try {
            Updater.sleep(Updater.TIME);
        } catch (InterruptedException e) {
            weatherApplication.setServiceRunning(false);
        }
    }
}
}
```

Στο αρχείο `app/java/com.example.user.teiweatherapp/service/WeatherService:`

```
package com.example.user.teiweatherapp.service;

import android.net.Uri;
import android.os.AsyncTask;
import android.util.Log;

import com.example.user.teiweatherapp.data.Values;

import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.URL;
import java.net.URLConnection;

/**
 * Created by PanagiotisGiannikakis on 22/2/2016.
 */
public class WeatherService {
    private WeatherServiceCallback callback;
    private String error;
    private Values values;

    public WeatherService(WeatherServiceCallback callback) {
        this.callback = callback;
    }

    public void refreshWeather(final String ip, final String type) {
        this.values = new Values();

        new AsyncTask<String, Void, String>() {
            @Override
            protected String doInBackground(String... params) {
                //create query for database
                String influxQuery = String.format("SELECT
LAST(load)FROM metrics WHERE host = 'bioklima' AND type = '%s'",
type);
            }
        }
    }
}
```

```

        String endpoint =
String.format("http://%s:8086/query?q=%s&db=kairos", ip,
Uri.encode(influxQuery));

        try {
            //create connection
            URL url = new URL(endpoint);
            URLConnection connection = url.openConnection();
            InputStream inputStream =
connection.getInputStream();
            //buffer response
            BufferedReader bufferedReader = new
BufferedReader(new InputStreamReader(inputStream));
            //read response
            StringBuilder databaseResponse = new
StringBuilder();

            String line;
            while ((line = bufferedReader.readLine()) !=
null) {

                databaseResponse.append(line);
            }
            return databaseResponse.toString();
        } catch (IOException e) {
            error = ("Network is unreachable. Please check
your network connection");
            Log.d("doInBackground", e.toString());
        }
        return null;
    }

    @Override
    protected void onPostExecute(String queryResponse) {
        // check for response and errors
        if (queryResponse == null && error != null) {
            callBack.serviceFailure(error);
            return;
        }
        try {
            //extract results
            JSONObject as = new JSONObject(queryResponse);

            int getValue =
as.getJSONArray("results").getJSONObject(0).getJSONArray("series").ge
tJSONObject(0).getJSONArray("values").getJSONArray(0).getInt(1);
            Log.d("onPostExecute", String.valueOf(getValue));
            //adjust the values
            if (type == "degc") {
                values.setLoaded_celsius(getValue);
            } else if (type == "rain") {
                values.setLoaded_rain(getValue);
            } else if (type == "speed") {
                values.setLoaded_speed(getValue);
            }
            //call back service
            callBack.serviceSuccess(values);
        } catch (Exception e) {
            Log.e("onPostExecute", "exception", e);
            callBack.serviceFailure(e.toString());
        }
    }
}.execute();

```

```

    }
}

```

**Στο αρχείο app/java/com.example.user.teiweatherapp/service/WeatherServiceCallBack:**

```

package com.example.user.teiweatherapp.service;

import com.example.user.teiweatherapp.data.Values;

/**
 * Created by PanagiotisGiannikakis on 22/2/2016.
 */
public interface WeatherServiceCallBack {
    void serviceSuccess(Values values);
    void serviceFailure(String string);
}

```

**Στο αρχείο app/java/com.example.user.teiweatherapp/MainActivity:**

```

package com.example.user.teiweatherapp;

import android.app.ProgressDialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.preference.PreferenceManager;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.user.teiweatherapp.service.UpdateService;

/**
 * Created by PanagiotisGiannikakis on 24/2/2016.
 */

public class MainActivity extends AppCompatActivity {
    private static final String TAG =
MainActivity.class.getSimpleName();
    protected static ProgressDialog dialog; // the loading dialog
    private String[] type = {"degc", "rain", "speed"}; //for the
influxdb query; what we want
    private SharedPreferences prefs; // get preferences values; here
new ip for server
    private String ip; // to set a new ip for the server
    //private Condition condition;
    protected WeatherApplication weatherApplication;
    protected static boolean isVisible = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        isVisible(true);
        Log.i(TAG, (String.valueOf(isVisible)));
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
}

```

```

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);

weatherApplication = (WeatherApplication) getApplication();
weatherApplication.setWeatherIconImageView((ImageView)
findViewById(R.id.weatherIconImageView));
weatherApplication.setTemperatureTextView((TextView)
findViewById(R.id.temperatureTextView));
weatherApplication.setConditionTextView((TextView)
findViewById(R.id.ConditionTextView));
weatherApplication.setLocationTextView((TextView)
findViewById(R.id.LocationTextView));
dialog = new ProgressDialog(this);
updateStatusManually();
FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        updateStatusManually();
    }
});

}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it
    is present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    if (weatherApplication.isServiceRunning()) {
menu.findItem(R.id.itemToggleService).setTitle(R.string.titleServices
top);
    }
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    switch (item.getItemId()) {
        case R.id.action_settings_update:
            updateStatusManually();
            return true;
        case R.id.preferences:
            //show preference menu
            Intent intent = new Intent(this,
PrefsActivity.class);
            startActivity(intent);
            Toast.makeText(MainActivity.this, "Set Server IP",
Toast.LENGTH_LONG).show();
            return true;
        case R.id.itemToggleService:
            if (weatherApplication.isServiceRunning()) {
                stopService(new Intent(this,
UpdateService.class));
                //weatherApplication.file.delete();
            }
    }
}

```

```

        Log.d(TAG, "stopService");
        item.setTitle(R.string.titleServiceStart);
    } else {
        startService(new Intent(this,
UpdateService.class));
        Log.d(TAG, "startService");
        item.setTitle(R.string.titleServiceStop);
    }
    return true;
default:
    return super.onOptionsItemSelected(item);
}
}

public void updateStatusManually() {
    if (weatherApplication.checkNetworkConnection()) {
        prefs =
PreferenceManager.getDefaultSharedPreferences(this);
        dialog.setMessage("Loading...");
        dialog.show();
        ip = prefs.getString("ServerAddress",
"teiwetherstats.ddnsking.com");

        weatherApplication.weatherService.refreshWeather(ip,
type);
    } else {
        Toast.makeText(this, R.string.networkConnectionError,
Toast.LENGTH_SHORT).show();
    }
}

private void isVisible(boolean b) {
    isVisible = b;
}

@Override
protected void onStop() {
    super.onStop();
    isVisible(false);
}

@Override
protected void onPause() {
    super.onPause();
    isVisible(false);
}

@Override
protected void onResume() {
    super.onResume();
    isVisible(true);
}

@Override
protected void onDestroy() {
    super.onDestroy();
    isVisible(false);
}

```



```

    }
}

```

**Στο αρχείο app/java/com.example.user.teiweatherapp/service/PrefsActivity:**

```

package com.example.user.teiweatherapp;

import android.app.Activity;
import android.os.Bundle;

import com.example.user.teiweatherapp.service.PrefsFragment;

/**
 * Created by PanagiotisGiannikakis on 24/2/2016.
 */
public class PrefsActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Display the fragment as the main content.
        getSupportFragmentManager().beginTransaction()
            .replace(android.R.id.content, new PrefsFragment())
            .commit();
    }
}

```

**Στο αρχείο app/java/com.example.user.teiweatherapp/service/WeatherApplication:**

```

package com.example.user.teiweatherapp;

import android.app.ActivityManager;
import android.app.Application;
import android.appwidget.AppWidgetManager;
import android.content.ComponentName;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.graphics.drawable.Drawable;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.preference.PreferenceManager;
import android.util.Log;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.user.teiweatherapp.data.Condition;
import com.example.user.teiweatherapp.data.Values;
import com.example.user.teiweatherapp.service.UpdateService;
import com.example.user.teiweatherapp.service.WeatherService;
import com.example.user.teiweatherapp.service.WeatherServiceCallBack;

import java.util.List;

/**
 * Created by PanagiotisGiannikakis on 28/3/2017.
 */
public class WeatherApplication extends Application implements
WeatherServiceCallBack {
    //Tag for debugging

```

```

    private static final String TAG =
WeatherApplication.class.getSimpleName();
    private boolean isServiceRunning;
    private ImageView weatherIconImageView;
    private TextView temperatureTextView;
    private TextView conditionTextView;
    private TextView locationTextView;
    protected WeatherService weatherService;
    protected String[] type = {"degc", "rain", "speed"}; //for the
influxdb query; what we want
    protected SharedPreferences prefs; // get preferences values; here
new ip for server
    protected String ip; // to set a new ip for the server
    protected Condition condition;
    protected int resourceID;
    private Drawable weatherIconDrawable;
    private WeatherWidget weatherWidget;
    private Context context;
    private AppWidgetManager appWidgetManager;

    @Override
    public void onCreate() {
        super.onCreate();
        weatherService = new WeatherService(this);
        weatherWidget = new WeatherWidget();
        prefs = PreferenceManager.getDefaultSharedPreferences(this);
        ip = prefs.getString("Server IP",
"teiiweatherstats.ddnsking.com");
        Log.d(TAG, "onCreate");
    }

    @Override
    public void onTerminate() {
        super.onTerminate();
        stopService(new Intent(this, UpdateService.class));
    }

    @Override
    public void serviceSuccess(Values values) {
        if (MainActivity.dialog != null) {
            MainActivity.dialog.hide();
        }

        // get the info from the currently running task
        ActivityManager am = (ActivityManager)
this.getSystemService(ACTIVITY_SERVICE);
        List<ActivityManager.RunningTaskInfo> taskInfo =
am.getRunningTasks(1);
        Log.d("topActivity", "CURRENT Activity :" +
taskInfo.get(0).topActivity.getClassName());
        Log.d(TAG, this.getPackageName());
        ComponentName componentInfo = taskInfo.get(0).topActivity;
        componentInfo.getPackageName();
        componentInfo.getClass().getMethods();

        //Update Application Content
        condition = new Condition(values);
        resourceID = getResources().getIdentifier("drawable/icon_" +
condition.getId(), null, getPackageName());

```

```

        weatherIconDrawable = getResources().getDrawable(resourceID);
        //if mainActivity is running then update it
        if (MainActivity.isVisible) {
            locationTextView.setText(R.string.location_Heraklion);

        temperatureTextView.setText(String.valueOf(values.getLoaded_celsius()
) + "\u00B0" + "C");

        weatherIconImageView.setImageDrawable(weatherIconDrawable);

        conditionTextView.setText(condition.getCondition().toString());
    }
    //Update Widget Content
    context = this;
    appWidgetManager = AppWidgetManager.getInstance(context);
    ComponentName name = new ComponentName(context,
WeatherWidget.class);
    int[] ids = appWidgetManager.getAppWidgetIds(name);
    weatherWidget.onUpdate(context, appWidgetManager, ids,
values, condition, resourceID);
    }

    //call this to check if there is connection to wifi or mobile
data
    public boolean checkNetworkConnection() {
        ConnectivityManager cm = (ConnectivityManager)
this.getSystemService(Context.CONNECTIVITY_SERVICE);
        NetworkInfo activeNetwork = cm.getActiveNetworkInfo();
        boolean isConnected = activeNetwork != null &&
activeNetwork.isConnectedOrConnecting();
        return isConnected;
    }

    @Override
    public void serviceFailure(String exception) {
        if (MainActivity.dialog != null) {
            MainActivity.dialog.hide();
            Toast.makeText(this, exception,
Toast.LENGTH_SHORT).show();
        }
    }

    public boolean isServiceRunning() {
        return isServiceRunning;
    }

    public void setServiceRunning(boolean isServiceRunning) {
        this.isServiceRunning = isServiceRunning;
    }

    public ImageView getWeatherIconImageView() {
        return weatherIconImageView;
    }

    public void setWeatherIconImageView(ImageView
weatherIconImageView) {
        this.weatherIconImageView = weatherIconImageView;
    }

    public TextView getTemperatureTextView() {
        return temperatureTextView;
    }

```

```

    }

    public void setTemperatureTextView(TextView temperatureTextView)
    {
        this.temperatureTextView = temperatureTextView;
    }

    public TextView getConditionTextView() {
        return conditionTextView;
    }

    public void setConditionTextView(TextView conditionTextView) {
        this.conditionTextView = conditionTextView;
    }

    public TextView getLocationTextView() {
        return locationTextView;
    }

    public void setLocationTextView(TextView locationTextView) {
        this.locationTextView = locationTextView;
    }

    public String getIp() {
        return ip;
    }

    public void setIp(String ip) {
        this.ip = ip;
    }

    public String[] getType() {
        return type;
    }

    public void setType(String[] type) {
        this.type = type;
    }
}

```

**Στο αρχείο app/java/com.example.user.teiweatherapp/service/WeatherWidget:**

```

package com.example.user.teiweatherapp;

import android.app.PendingIntent;
import android.appwidget.AppWidgetManager;
import android.appwidget.AppWidgetProvider;
import android.content.ComponentName;
import android.content.Context;
import android.content.Intent;
import android.widget.RemoteViews;

import com.example.user.teiweatherapp.data.Condition;
import com.example.user.teiweatherapp.data.Values;

import static android.content.Intent.FILL_IN_ACTION;

/**
 * Implementation of App Widget functionality.
 */
public class WeatherWidget extends AppWidgetProvider {

```

```

static PendingIntent pendingIntent;
static Intent intent;

public void onUpdate(Context context, AppWidgetManager
appWidgetManager, int[] appWidgetIds, Values values, Condition
condition, int resourceID) {
    // There may be multiple widgets active, so update all of
them
    final int N = appWidgetIds.length;
    for (int i = 0; i < N; i++) {

        updateAppWidget(context, appWidgetManager,
appWidgetIds[i], values, condition, resourceID);
    }

    @Override
public void onEnabled(Context context) {
        // Enter relevant functionality for when the first widget is
created
    }

    @Override
public void onDisabled(Context context) {
        // Enter relevant functionality for when the last widget is
disabled
    }

    static void updateAppWidget(Context context, AppWidgetManager
appWidgetManager, int appWidgetId, Values values, Condition
condition, int resourceID) {

        appWidgetManager = AppWidgetManager.getInstance(context);
        // Construct the RemoteViews object
        RemoteViews remoteViews = new
RemoteViews(context.getPackageName(), R.layout.weather_widget);
        ComponentName thisWidget = new ComponentName(context,
WeatherWidget.class);
        remoteViews.setTextViewText(R.id.WidgetTextTemperature,
String.valueOf(values.getLoaded_celsius()) + "\u00B0" + "C");
        remoteViews.setTextViewText(R.id.WidgetTextCondition,
String.valueOf(condition.getCondition().toString()));
        remoteViews.setTextViewText(R.id.WidgetTextLocation,
context.getString(R.string.location_Heraklion));
        remoteViews.setImageViewResource(R.id.WidgetImageView,
resourceID);
        //add onClickListener
        intent = new Intent(context, MainActivity.class);
        pendingIntent = PendingIntent.getActivity(context, 0, intent,
FILL_IN_ACTION);
        remoteViews.setOnClickPendingIntent(R.id.WidgetImageView,
pendingIntent);
        // Instruct the widget manager to update the widget
        appWidgetManager.updateAppWidget(thisWidget, remoteViews);
    }
}

```

Στο αρχείο app/res/layout/activity\_main.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
xmlns:android="http://schemas.android.com/apk/res/android"

```

```

xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:fitsSystemWindows="true"
android:background="#9ee1fd"
tools:context=".MainActivity">

<android.support.design.widget.AppBarLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/AppTheme.AppBarOverlay">

    <android.support.v7.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        android:background="?attr/colorPrimary"
        app:popupTheme="@style/AppTheme.PopupOverlay" />

</android.support.design.widget.AppBarLayout>

<include layout="@layout/content_main" />

<android.support.design.widget.FloatingActionButton
    android:id="@+id/fab"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|end"
    android:layout_margin="@dimen/fab_margin"
    android:src="@android:drawable/ic_menu_rotate" />

</android.support.design.widget.CoordinatorLayout>

```

Στο αρχείο app/res/layout/content\_main.xml:

```

<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:showIn="@layout/activity_main"
    tools:context=".MainActivity">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/weatherIconImageView"
        android:src="@drawable/icon_50"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```

        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="@string/temperature"
        android:textStyle="bold"
        android:typeface="serif"
        android:id="@+id/temperatureTextView"
        android:layout_below="@+id/weatherIconImageView"
        android:layout_centerHorizontal="true"
        android:textSize="24dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="@string/condition"
    android:id="@+id/ConditionTextView"
    android:layout_below="@+id/temperatureTextView"
    android:layout_centerHorizontal="true"
    android:textStyle="bold"
    android:textSize="20dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:text="@string/location"
    android:id="@+id/locationTextView"
    android:layout_below="@+id/ConditionTextView"
    android:layout_centerHorizontal="true"
    android:textStyle="bold"
    android:textSize="18dp" />

<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/LogoImageView"
    android:src="@drawable/icon_49"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="40dp"
    android:layout_marginLeft="100dp"
    android:layout_marginRight="100dp"
    android:layout_marginTop="0dp" />
</RelativeLayout>

```

Στο αρχείο `app/res/layout/weather_widget.xml`:

```

<GridLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="#6ba4ff"
    android:columnCount="2"
    android:padding="0dp"
    android:rowCount="3"
    android:useDefaultMargins="false">

<ImageView
    android:id="@+id/WidgetImageView"
    android:layout_width="80dp"
    android:layout_height="60dp"
    android:layout_column="0"

```

```

        android:layout_row="0"
        android:layout_rowSpan="3"
        android:clickable="true"
        android:contentDescription="@string/temperatureIcon"
        android:src="@drawable/icon_50" />

<TextView
    android:id="@+id/WidgetTextTemperature"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_column="1"
    android:layout_columnSpan="1"
    android:layout_row="0"
    android:text="@string/temperature"
    android:textAppearance="?android:attr/textAppearanceLarge" />

<TextView
    android:id="@+id/WidgetTextCondition"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_column="1"
    android:layout_columnSpan="1"
    android:layout_row="1"
    android:text="@string/condition"
    android:textAppearance="?android:attr/textAppearanceMedium"
/>

<TextView
    android:id="@+id/WidgetTextLocation"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_column="1"
    android:layout_columnSpan="1"
    android:layout_row="2"
    android:text="@string/location"
    android:textAppearance="?android:attr/textAppearanceSmall" />
</GridLayout>

```

Στο αρχείο `app/res/menu/menu_main.xml`:

```

<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context=".MainActivity">
    <!-- "settings" submenu -->
    <item
        android:id="@+id/action_settings_update"
        android:orderInCategory="100"
        android:title="@string/action_settings"
        app:showAsAction="never" />
    <item
        android:id="@+id/preferences"
        android:orderInCategory="100"
        android:title="@string/setEndpointIP"
        app:showAsAction="never" />
</menu>

```

Στο αρχείο `app/res/values/strings.xml`:

```

<resources>
    <string name="app_name">TeiWeatherApp</string>
    <string name="widget_name">TeiWeatherWidget</string>
    <string name="action_settings">Update</string>

```



```

<string name="temperature">Temperature</string>
<string name="condition">Condition</string>
<string name="location">Location</string>
<string name="location_Heraklion">Heraklion</string>
<string name="setEndpointIP">Set Server IP</string>
<string name="prefsTitle">Preferences</string>
<string name="prefsSummaryServerAddress">Please enter the Server
Address</string>
<string name="temperatureIcon">TemperatureIcon</string>

</resources>

```

**Στο αρχείο app/res/xml/prefs.xml:**

```

<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen
xmlns:android="http://schemas.android.com/apk/res/android">

    <EditTextPreference
        android:key="ServerAddress"
        android:password="false"
        android:summary="@string/prefsSummaryServerAddress"
        android:title="http:// address" />

</PreferenceScreen>

```

**Στο αρχείο app/res/xml/weather\_widget\_info.xml:**

```

<?xml version="1.0" encoding="utf-8"?>
<appwidget-provider
xmlns:android="http://schemas.android.com/apk/res/android"
    android:initialKeyguardLayout="@layout/weather_widget"
    android:initialLayout="@layout/weather_widget"
    android:minHeight="70dp"
    android:minWidth="200dp"
    android:previewImage="@drawable/sun"
    android:resizeMode="horizontal|vertical"
    android:updatePeriodMillis="86400000"
    android:widgetCategory="home_screen"></appwidget-provider>

```

Τέλος στον φάκελο app/res/drawable θα χρειαστούμε κάποιες εικόνες που θα παρουσιάζουν τις ανάλογες καιρικές συνθήκες. Μπορούμε να κατεβάσουμε κάποια πακέτα εικόνων από τις παρακάτω διευθύνσεις αλλά θα χρειαστούμε τις άδειες των δημιουργών τους πρώτα.

<http://vclouds.deviantart.com/art/VClouds-Weather-Icons-179152045>

<http://aipotudeng.deviantart.com/art/Weather-Icon-174370042>

Προαιρετικά εάν θέλουμε η υπηρεσία ενημέρωσης να ξεκινάει κάθε φορά που ξεκινάει το κινητό του χρήστη προσθέτουμε τον παρακάτω κώδικα στο manifest:

```

<receiver
android:name="com.example.user.teiweatherapp.service.StartServiceAtBoot">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED" />
    </intent-filter>
</receiver>

```

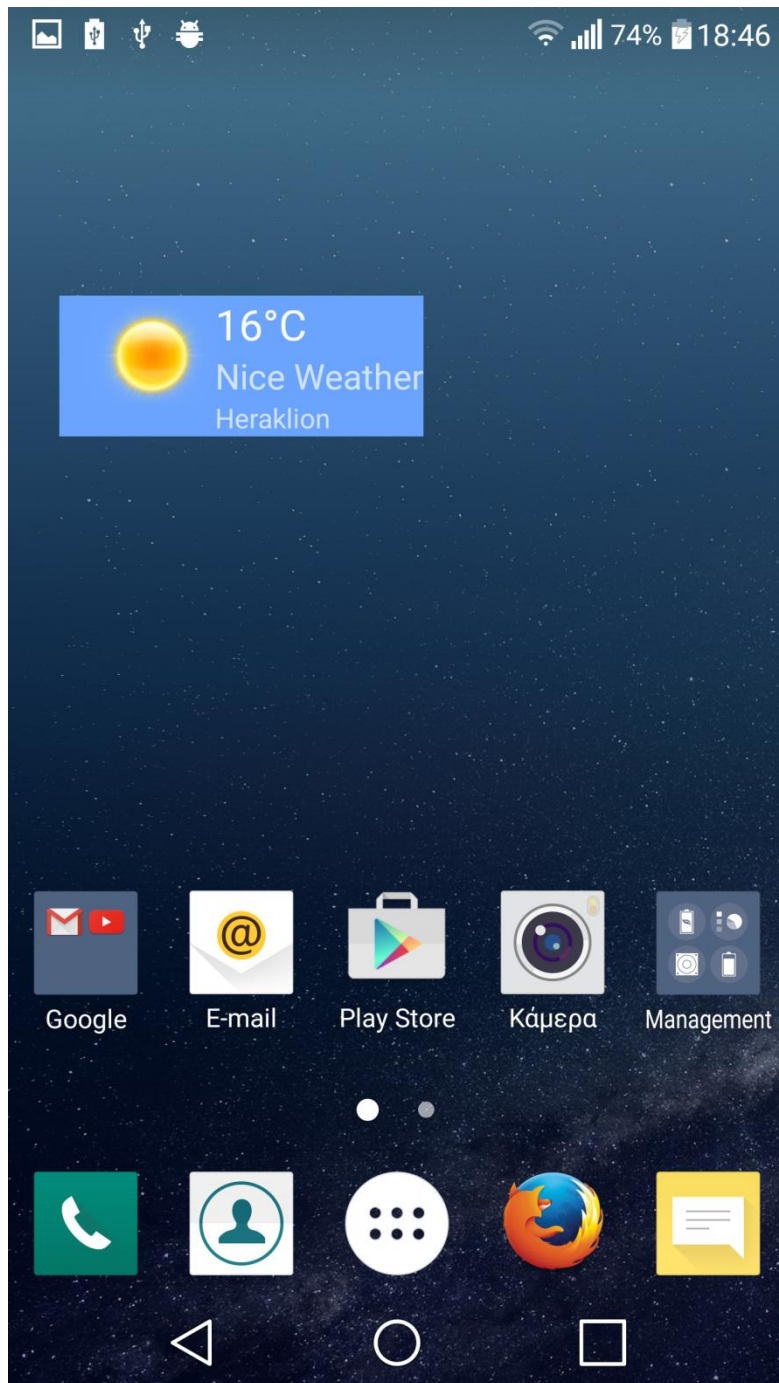
Και επίσης δημιουργούμε μια νέα κλάση με όνομα StartServiceAtBoot και γραφούμε:

```
public class StartServiceAtBoot extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        if
        ("android.intent.action.BOOT_COMPLETED".equals(intent.getAction())) {
            Intent pushIntent = new Intent(context,
            UpdateService.class);
            context.startService(pushIntent);
        }
    }
}
```

Το τελικό αποτέλεσμα θα πρέπει να είναι όπως παρακάτω:



Εικόνα 6-62: Android app main activity



Εικόνα 6-63: Android app widget

## 7. Συμπεράσματα

Με το πέρας της πτυχιακής εργασίας ολοκληρώθηκε ένα portal και μία εφαρμογή για κινητές συσκευές οι οποίες απεικονίζουν μετεωρολογικά δεδομένα από τους μετεωρολογικούς σταθμούς του ΤΕΙ.

Τα δεδομένα παρουσιάζονται σε φιλική προς το χρήστη μορφή με τη βοήθεια τεχνολογιών αιχμής όπως το πλαίσιο εφαρμογής Grafana, το Bootstrap, το CSS3, η γλώσσα προγραμματισμού HTML5 και η βάση δεδομένων Influxdb. Για την εφαρμογή σε κινητές συσκευές χρησιμοποιήθηκε το περιβάλλον ανάπτυξης Android Studio.

Επιπλέον, με την ολοκλήρωση της πτυχιακής μου εργασίας εξοικειώθηκα και απέκτησα δεξιότητες με τεχνολογίες αιχμής. Είμαι σε θέση να σχεδιάσω μία ηλεκτρονική πύλη, να αναπτύξω μία εφαρμογή για κινητή συσκευή και γενικότερα να προγραμματίσω σε πλαίσιο εφαρμογής ανοιχτού ή μη κώδικα.

## Βιβλιογραφία

- [1] <https://www.w3.org/TR/html5/>
- [2] <https://www.javascript.com/>
- [3] <https://en.wikipedia.org/wiki/HTML>
- [4] <https://www.w3.org/TR/2001/WD-css3-roadmap-20010523/>
- [5] <https://jquery.com/>
- [6] <http://getbootstrap.com/>
- [7] <http://grafana.org/>
- [8] <https://influxdata.com/>
- [9] <http://www.businessinsider.in/We-Asked-Executives-About-The-Internet-Of-Things-And-Their-Answers-Reveal-That-Security-Remains-A-Huge-Concern/articleshow/45959921.cms>
- [10] <https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/>
- [11] <https://www.wired.com/2012/03/petraeus-tv-remote/>
- [12] <http://www.bbc.com/news/technology-34324247>
- [13] [https://www.cs.cmu.edu/~coke/history\\_long.txt](https://www.cs.cmu.edu/~coke/history_long.txt)
- [14] <http://www.pewinternet.org/2014/05/14/main-report-an-in-depth-look-at-expert-responses/>
- [15] [http://www.cisco.com/c/dam/en\\_us/about/ac79/docs/innov/IoT\\_IBSG\\_0411FINAL.pdf](http://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf)
- [16] [https://en.wikipedia.org/wiki/Internet\\_of\\_things](https://en.wikipedia.org/wiki/Internet_of_things)
- [17] <http://www.economist.com/node/15557443>
- [18] <http://www.nature.com/nature/journal/v455/n7209/full/455001a.html>
- [19] <http://www.martinhilbert.net/WorldInfoCapacity.html/>
- [20] <https://www.villanovau.com/resources/bi/what-is-big-data/#.W1e-M2WXdcs>
- [21] <http://www.informationweek.com/big-data/big-data-analytics/big-data-avoid-wanna-v-confusion/d/d-id/1111077?>
- [22] [http://www.martinhilbert.net/wp-content/uploads/2015/01/BigData4Dev\\_Hilbert2014.pdf](http://www.martinhilbert.net/wp-content/uploads/2015/01/BigData4Dev_Hilbert2014.pdf)
- [23] [http://mike2.openmethodology.org/wiki/Big\\_Data\\_Solution\\_Offering](http://mike2.openmethodology.org/wiki/Big_Data_Solution_Offering)
- [24] <http://www.howtogeek.com/185876/what-is-fog-computing/>
- [25] <http://www.marklogic.com/blog/schema-on-read-vs-schema-on-write/>
- [26] [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=1926431](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1926431)
- [27] [http://download.intel.com/newsroom/kits/idf/2015\\_fall/pdfs/Intel-5G\\_MegaSession\\_Factsheet.pdf](http://download.intel.com/newsroom/kits/idf/2015_fall/pdfs/Intel-5G_MegaSession_Factsheet.pdf)
- [28] [https://www.wired.com/2012/03/ff\\_nsadatacenter/all/1](https://www.wired.com/2012/03/ff_nsadatacenter/all/1)
- [29] <https://www.ijedr.org/papers/IJEDR1504022.pdf>
- [30] [http://www.cisco.com/c/dam/en\\_us/about/ac79/docs/ps/motm/IoE-Smart-City\\_PoV.pdf](http://www.cisco.com/c/dam/en_us/about/ac79/docs/ps/motm/IoE-Smart-City_PoV.pdf)
- [31] <http://cordis.europa.eu/fp7/ict/enet/documents/publications/iot-between-the-internet-revolution.pdf>
- [32] <https://httpd.apache.org/>
- [33] <https://netbeans.org/>