

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



Θέμα Πτυχιακής Εργασίας:

«Εφαρμογή καταγραφής σεισμών με τη χρήση Android™ συσκευών»

«Earthquake detection application using Android™ devices»

Όνομα φοιτητή – ΑΜ	Σαραντίδης Λυκούργος – Π15124
E-mail	ipsarantidis@gmail.com

Επιβλέπων Καθηγητής: Αλέπης Ευθύμιος

Πειραιάς, Σεπτέμβριος 2019



Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω τον Διευθυντή Ερευνών του Γεωδυναμικού Ινστιτούτου του Εθνικού Αστεροσκοπείου Αθηνών Δρ. Γεώργιο Δρακάτο για τον χρόνο και το υλικό που μου διέθεσε ώστε να αποκτήσω μια καλύτερη κατανόηση των σεισμικών φαινομένων.

Ευχαριστώ τους καθηγητές του Πανεπιστημίου Πειραιώς για τις πολύτιμες γνώσεις που μου προσέφεραν κατά την διάρκεια των σπουδών μου.

Επίσης θέλω να απευθύνω θερμές ευχαριστίες στον επίκουρο καθηγητή Αλέπη Ευθύμιο για τη συνεχή και ουσιαστική καθοδήγηση και συνδρομή του πριν και κατά την διάρκεια εκπόνησης της παρούσας εργασίας.

Και τέλος θέλω να ευχαριστήσω την οικογένεια μου για την ηθική τους συμπαράσταση και την ολοπρόθυμη στήριξή τους στην προσπάθειά μου.



Περιεχόμενα

1. Περίληψη	4
2. Εισαγωγή.....	5
3. Ανασκόπηση πεδίου	7
4. Παρουσίαση και χρήση εφαρμογής (user's manual)	11
5. Αρχιτεκτονική συστήματος	19
6. Συμπεράσματα και μελλοντικές επεκτάσεις	24
7. Βιβλιογραφία	25



1. Περίληψη

Το θέμα της παρούσας πτυχιακής είναι μία εφαρμογή, για Android συσκευές, η οποία χρησιμοποιώντας τους αισθητήρες της συσκευής, και συγκεκριμένα το επιταχυνσιόμετρο (accelerometer), και αξιοποιώντας το πλήθος των συσκευών δημιουργεί ένα δίκτυο από συσκευές που καταγράφουν σεισμούς και ειδοποιεί τους χρήστες, σε πραγματικό χρόνο, για τους σεισμούς που λαμβάνουν χώρα.



2. Εισαγωγή

Η εφαρμογή (Earthquake Observer) τρέχει κάθε φορά που η συσκευή συνδέεται στο ρεύμα, δηλαδή φορτίζει. Με αυτόν τον τρόπο πετυχαίνουμε τρία πράγματα:

- δεν υπάρχει θέμα μπαταρίας,
- κατά πάσα πιθανότητα η συσκευή είναι ακίνητη και
- η εφαρμογή θα καταγράφει για αρκετή ώρα, επειδή συνήθως οι περισσότεροι χρήστες αφήνουν για αρκετή ώρα τις συσκευές τους στη φόρτιση, οπότε θα έχουμε αρκετά δεδομένα.

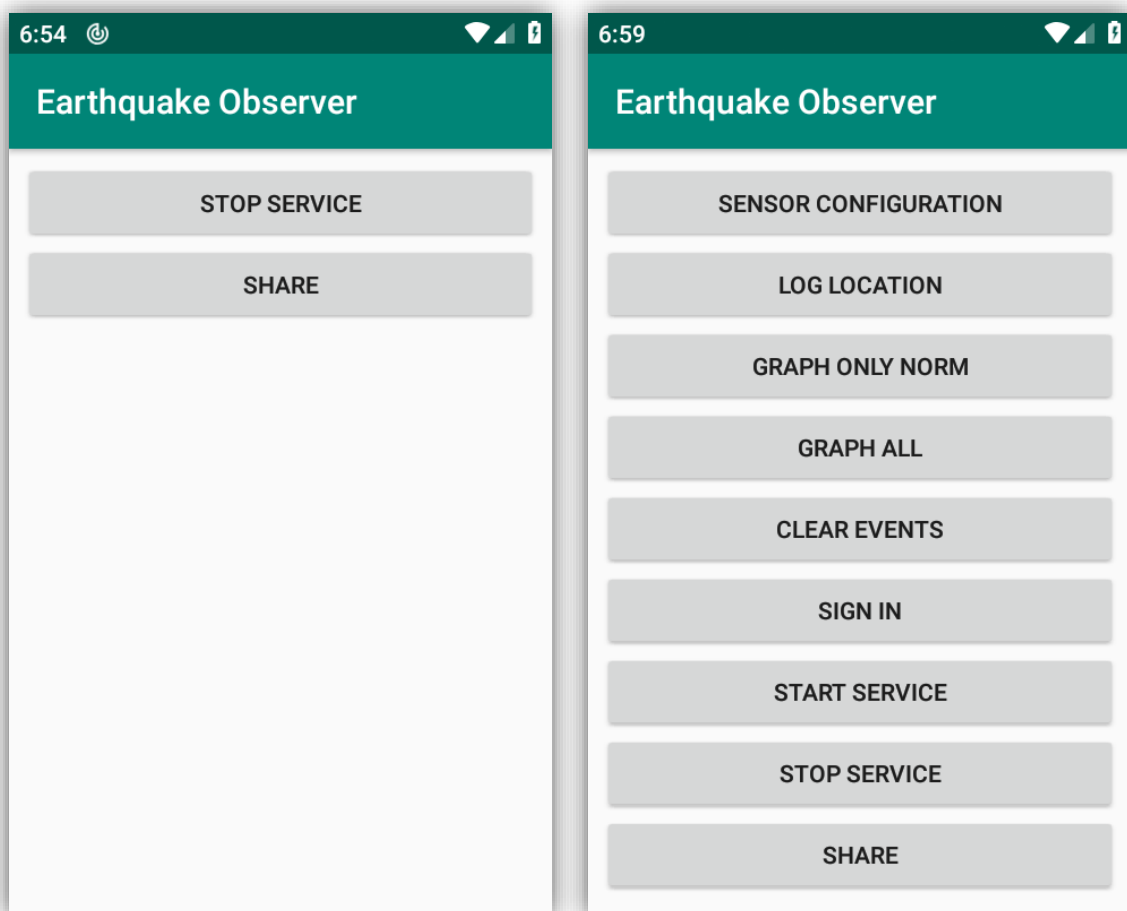
Ο χρήστης δεν αλληλοεπιδρά ιδιαίτερα με την εφαρμογή, εκτός αν είναι διαχειριστής (admin), μιας και το κυρίως μέρος της εφαρμογής δηλαδή η παρακολούθηση των μετρήσεων του επιταχυνσιομέτρου γίνεται στο παρασκήνιο, χωρίς να απαιτείται κάποια ενέργεια από το χρήστη. Ο χρήστης το μόνο που έχει να κάνει είναι να συνδεθεί στην εφαρμογή χρησιμοποιώντας έναν λογαριασμό Google (Google sign-in, SignInActivity) και να ακολουθήσει κάποιες οδηγίες προκειμένου να ληφθούν κάποιες δοκιμαστικές μετρήσεις ώστε να ρυθμιστεί κατάλληλα η συσκευή (ConfigDeviceActivity). Από εκεί και στο εξής η εφαρμογή αρχίζει να καταγράφει μετρήσεις αυτόματα κάθε φορά που η συσκευή συνδέεται στο ρεύμα (δηλαδή φορτίζει). Κάθε φορά που λαμβάνει χώρα ένας σεισμός ο χρήστης ειδοποιείται μέσω της εφαρμογής, σχετικά με την τοποθεσία που έλαβε χώρα ο σεισμός (latitude - longitude), την χρονική στιγμή που άρχισε και ξεκίνησε, και τις σχετικές μετρήσεις που καταγράφηκαν από τις ενεργές συσκευές (δηλαδή τις συσκευές στις οποίες έτρεχε η εφαρμογή επειδή ήταν συνδεδεμένες στο ρεύμα).

Στην περίπτωση που ο χρήστης έχει δικαιώματα διαχειριστή, δηλαδή ο χρήστης είναι καταχωρημένος ως admin στη βάση, έχει κάποιες περαιτέρω λειτουργίες όπως:

- να επαναλάβει ανά πάσα στιγμή την ρύθμιση της συσκευής (ConfigDeviceActivity)
- να βλέπει πληροφορίες σχετικά με την τοποθεσία της συσκευής (LogLocationActivity)
- να προβάλει την γραφική αναπαράσταση (plot) των μετρήσεων του επιταχυνσιομέτρου (GraphOnlyNormActivity, GraphAllActivity)
- να διαγράψει τις αποθηκευμένες μετρήσεις από τη βάση (saved-events) που έχουν καταχωρηθεί από την συγκεκριμένη συσκευή
- να αποσυνδεθεί από τον υπάρχοντα συνδεδεμένο χρήστη



- να ξεκινήσει ή να σταματήσει την παρακολούθηση των μετρήσεων του επιταχυνσιομέτρου (ObserverService), ανά πάσα στιγμή χωρίς να πληρούνται οι ορισμένες προϋποθέσεις (π.χ. να φορτίζει).

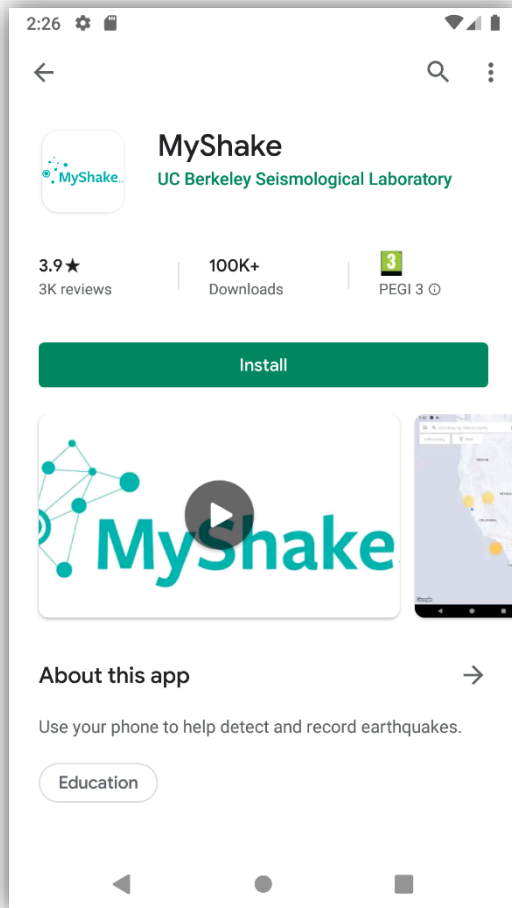




3. Ανασκόπηση πεδίου

- MyShake

(<https://play.google.com/store/apps/details?id=edu.berkeley.bsl.myshake>):



The MyShake app -- from the UC Berkeley Seismology Lab -- is different from all the other earthquake apps. In addition to providing earthquake notifications, maps, and safety tips, your phone helps to detect earthquakes.

Get earthquake ready and help your community. Have earthquake information at your fingertips and help create a global earthquake detection system.

Quickly understand the impact of an earthquake. See damage and shaking reports submitted by other community members along with information from the US Geological Survey.

Stay informed. Get notifications about nearby earthquakes and track earthquakes around

the globe.

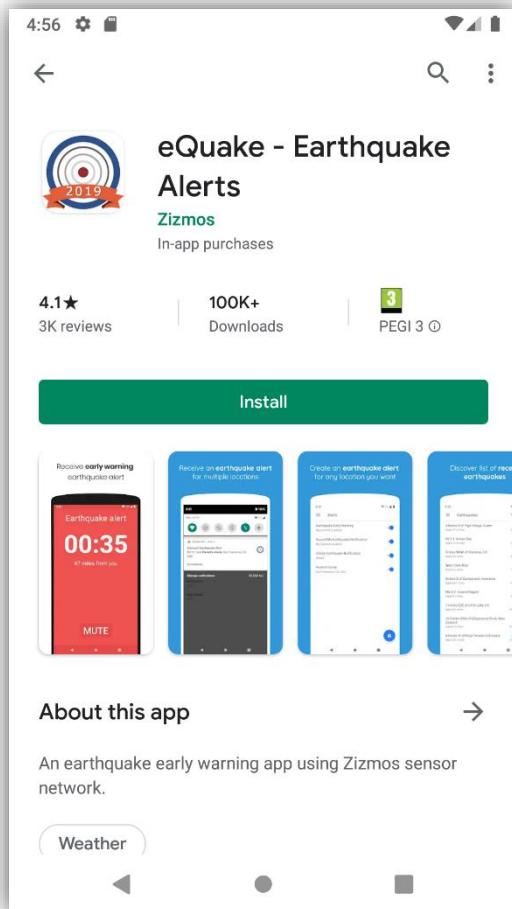
Share your experience. Felt an earthquake? Share your experience with fellow users and MyShake scientists.

Help improve earthquake early warning. Your device becomes an earthquake sensor and joins a smartphone network collecting valuable data. *** Note: MyShake is not issuing earthquake early warning alerts at this time, but does plan to in the future. ***



- eQuake - Earthquake Alerts

(<https://play.google.com/store/apps/details?id=com.zizmos.quake&hl=en>):



eQuake™ is an earthquake early-warning app developed by Zizmos™. It uses a network of seismic sensors to detect earthquakes. When an earthquake occurs, the Zizmos servers send the user an early-warning notification that alerts the user of expected shaking in the area. The ability to detect earthquakes and provide alerts is dependent on the number of sensors in the Zizmos network.

Another feature of eQuake™ is its capability to use your phone as a sensor which contributes to the Zizmos earthquake warning system improving the coverage of seismic detection. The sensor mode only works when the device is charging and connected to a Wi-Fi network.

The app provides other features that give the

user useful earthquake information.

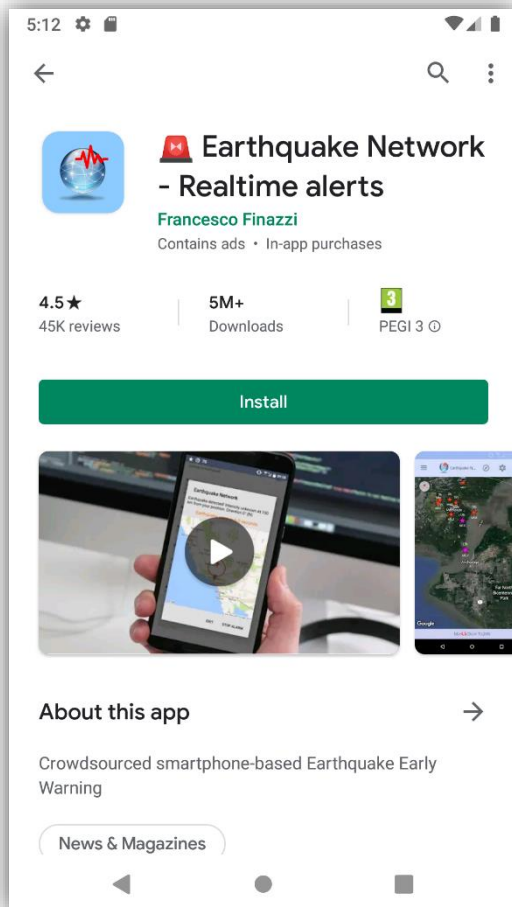
Features include:

- Early detection of earthquakes and early warning of shaking expected in the area. Users in an affected area are notified that an earthquake has occurred, and shaking is expected, granting the user potentially crucial advanced warning.
- A map of recent earthquakes that have occurred worldwide which uses USGS databases for the latest information.
- An earthquake early-warning simulator feature. Allows the user to simulate Zizmos™ earthquake alerts on the phone using historical earthquakes in the vicinity of the current location.



-  Earthquake Network - Realtime alerts

(<https://play.google.com/store/apps/details?id=com.finazzi.distquake>)



Earthquake Network is the most comprehensive app on earthquakes and for most countries of the world it is the only earthquake early warning system

- Real time detection of earthquakes using networks of smartphones
- Earthquake warning seconds in advance
- User's reports on felt earthquakes
- Automatic e-mails and SMS with your coordinates sent when an earthquake is detected in real time
- Earthquake data from national and international seismic networks starting from magnitude 0.0
- Earthquake history 2000-2017
- Chatrooms for exchanging info during earthquakes

- Earthquake notifications through voice synthesizer (only PRO version)
- Earthquake warning in case of vibrations detected by your smartphone (only PRO version)
- Earthquake visualization with augmented reality (only PRO version)
- Chat message priority during emergencies (only PRO version)

The Earthquake Network research project does not receive external funding. By purchasing the PRO version you support the project and its continuous improvement to provide a fast and reliable service. Thank you!

The Earthquake Network research project (<http://www.earthquakenetwork.it>) aims at developing a smartphone-based earthquake early warning system able to detect earthquakes in real time and to alert the population in advance. Smartphones are able to detect earthquakes thanks to the accelerometer on-board each device. When an



earthquake is detected, users with the application installed are immediately alerted. Since earthquake waves travel at a finite speed (from 5 to 10 km/s) it is possible to alert the population not yet reached by the damaging waves of the earthquake. For the scientific details about the project please refer to the Bulletin of the Seismological Society of America journal at <http://goo.gl/IFBLYm>

The Earthquake Network app allows you to receive the real time alerts of the earthquakes detected by the network, to manually report an earthquake you felt and to chat with other people during an earthquake emergency. When an earthquake is detected in real time near your location, the app can send a automatic SMS and e-mails to a list of contacts with your precise position in space. This can be useful for rescuing without the need of user interaction with the app. You can also send a request of help or you can say that you are fine with just one click. The app also shows information on the earthquakes detected by the national and international seismic networks

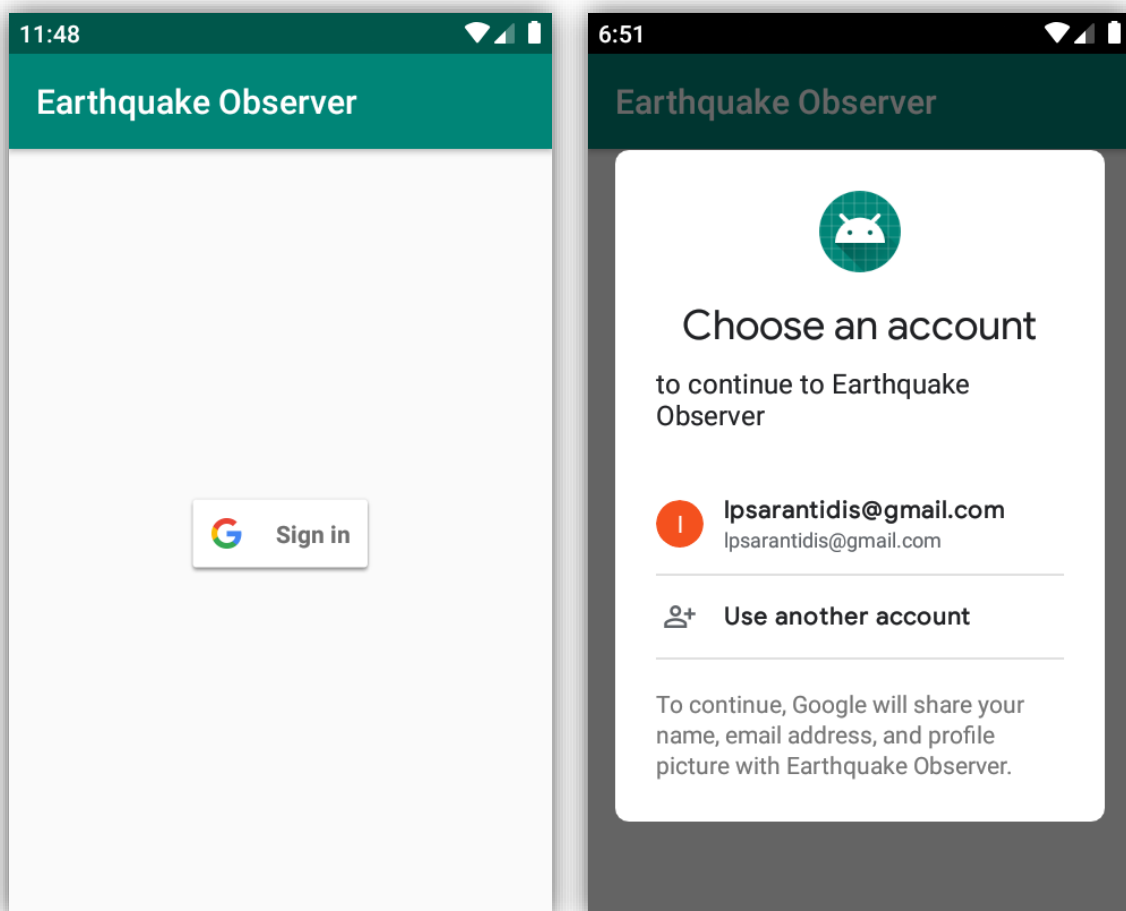
Note that the information on earthquakes detected by national and international seismic networks is usually published with a delay ranging from a few minutes to many hours, depending on the seismic network.



4. Παρουσίαση και χρήση εφαρμογής (user's manual)

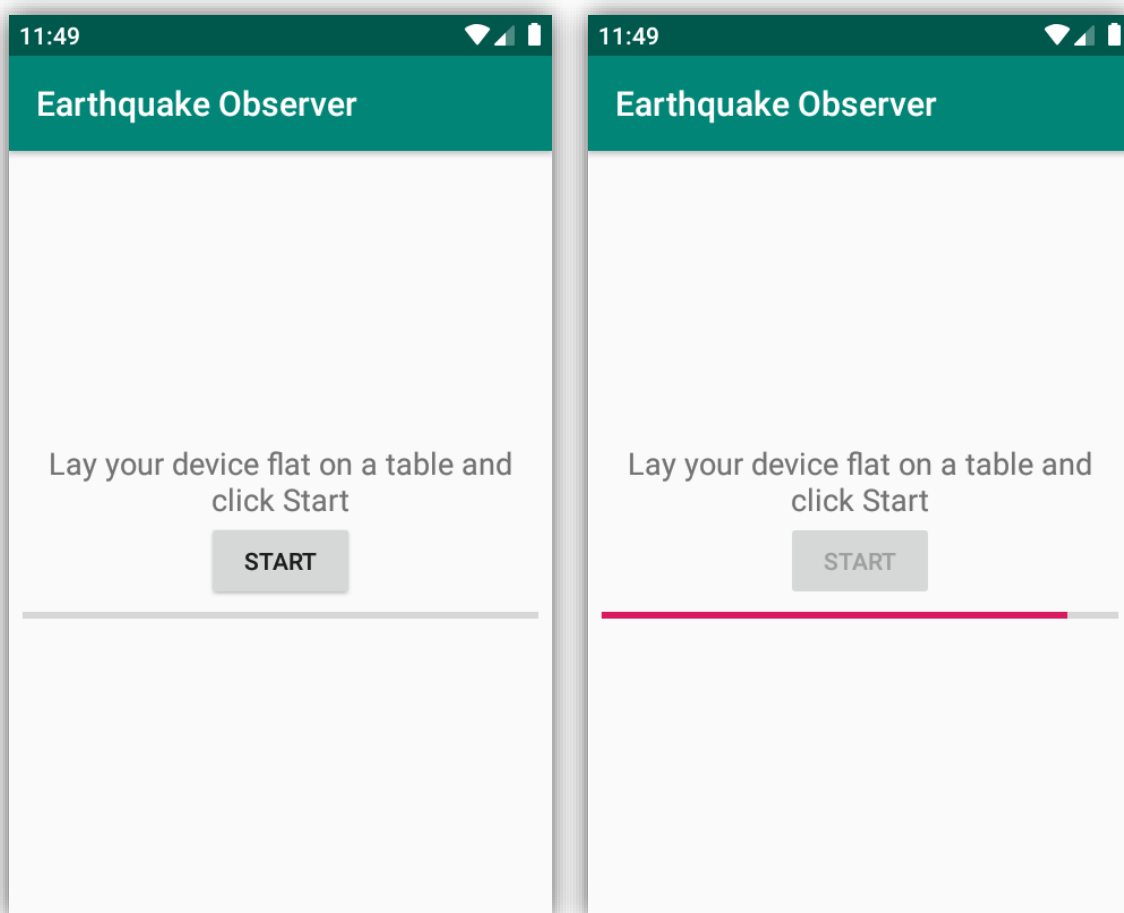
- **Λειτουργίες απλού χρήστη**

Αρχικά ο χρήστης πρέπει να συνδεθεί χρησιμοποιώντας ένα λογαριασμό Google (απαιτείται μόνο κατά την πρώτη είσοδο του χρήστη στην εφαρμογή). Για την ολοκλήρωση αυτού του βήματος απαιτείται σύνδεση στο διαδίκτυο.



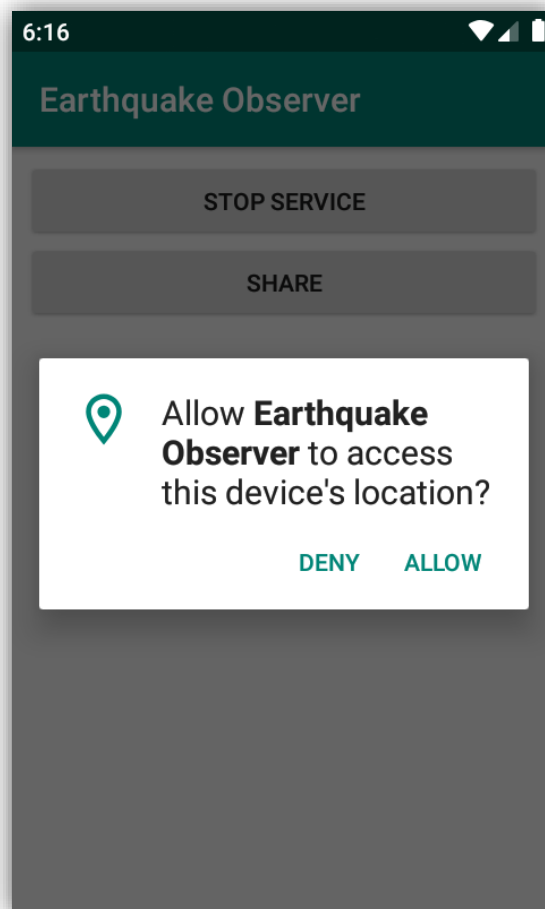


Στη συνέχεια πρέπει να ρυθμιστεί κατάλληλα η συσκευή για να είναι πιο ακριβείς οι μετρήσεις (απαιτείται μόνο κατά την πρώτη είσοδο του χρήστη στην εφαρμογή). Για την ολοκλήρωση αυτού του βήματος απαιτείται σύνδεση στο διαδίκτυο.



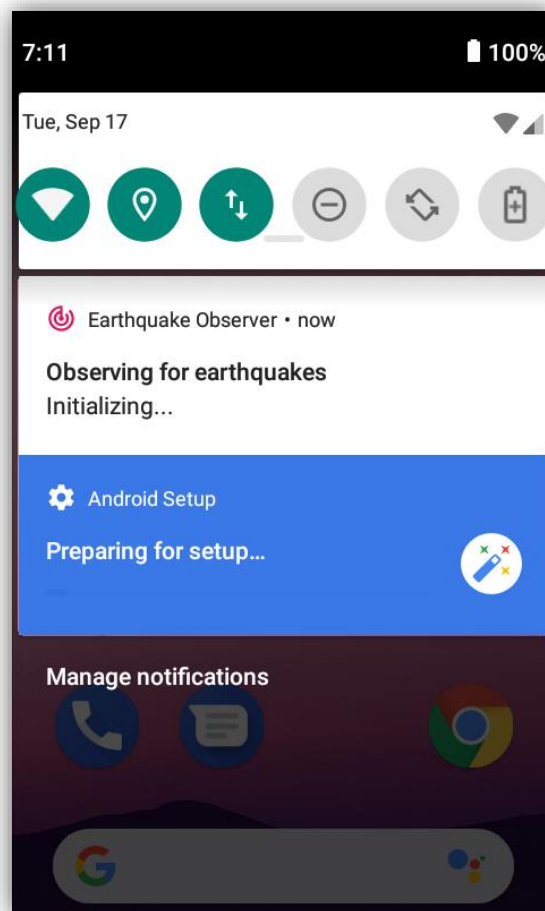


Έπειτα απαιτείται να δοθεί η άδεια στην εφαρμογή να έχει πρόσβαση στην τοποθεσία της συσκευής, ώστε σε περίπτωση σεισμού να είναι δυνατή η εύρεση του τόπου που έλαβε χώρα ο σεισμός. Αν δε δοθεί η άδεια (permission) δε θα μπορεί να λειτουργήσει κανονικά η εφαρμογή, αφού η τοποθεσία είναι απαραίτητη για την “αναφορά” των σεισμών.



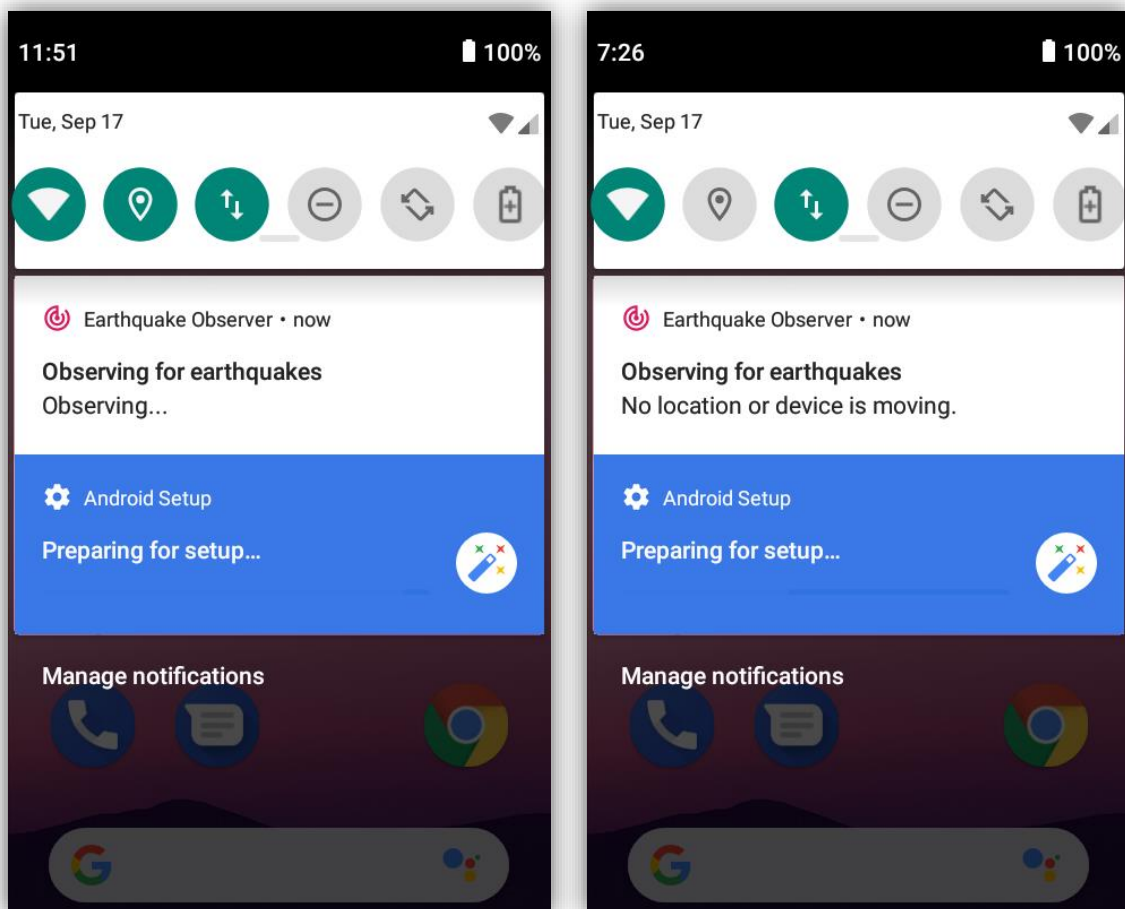


Αφού ολοκληρωθούν τα παραπάνω βήματα, στο εξής κάθε φορά που ο χρήστης συνδέει τη συσκευή στο ρεύμα για να φορτίσει (και υπάρχει σύνδεση στο διαδίκτυο, προκειμένου να επιτευχθεί η καταγραφή των σεισμών σε πραγματικό χρόνο), θα αρχίζει αυτόματα να τρέχει η εφαρμογή μας στο παρασκήνιο (foreground service) χωρίς να απαιτείται κάποια περαιτέρω ενέργεια από το χρήστη. Αρχικά απαιτείται κάποιος χρόνος πρώτου αρχίσει η παρακολούθηση, μέχρι η εφαρμογή να αποκτήσει την ακριβή τοποθεσία της συσκευής ("Initializing...").



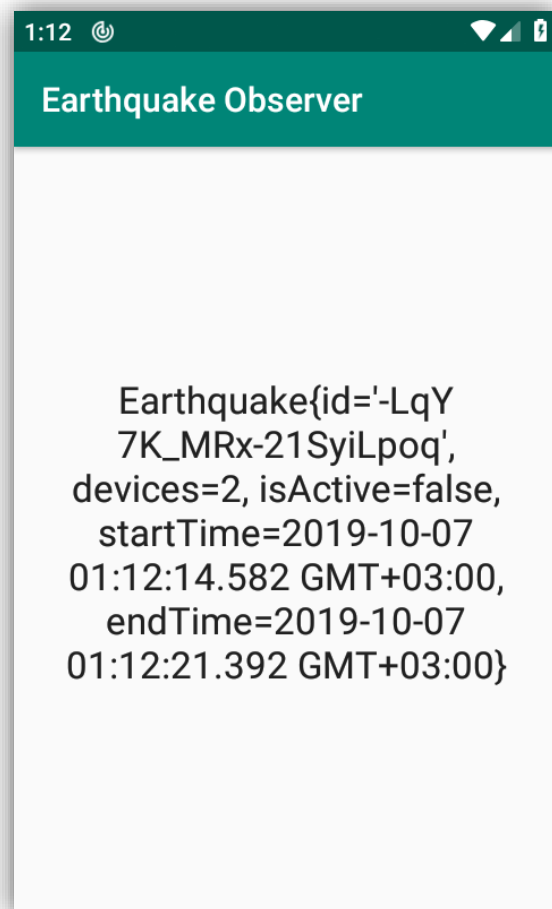
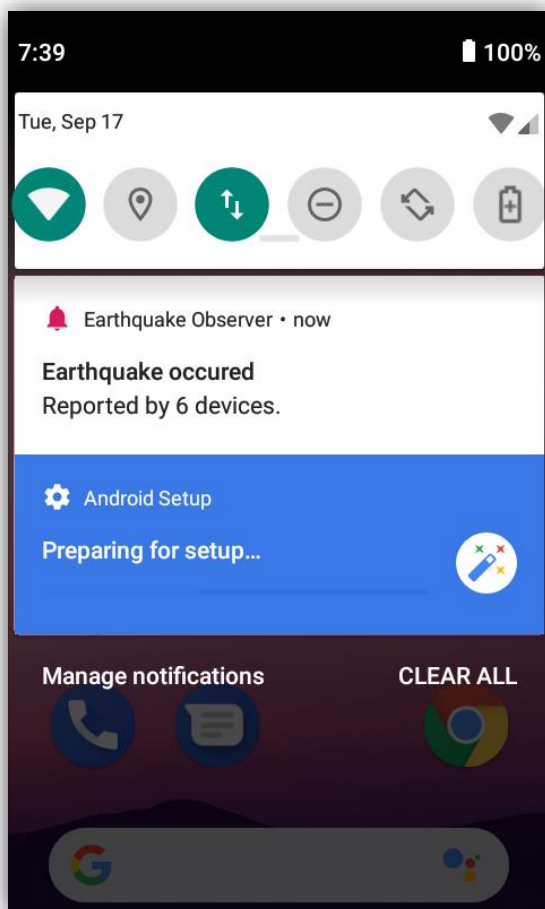


Αφού αποκτηθεί επιτυχώς η τοποθεσία της συσκευής αρχίζει η παρακολούθηση ("Observing..."), ώστε σε περίπτωση που καταγραφούν μετρήσεις αντίστοιχες να καταγραφεί στη βάση στο Cloud (Firebase Realtime Database). Σε περίπτωση όμως που η τοποθεσία της συσκευής είναι απενεργοποιημένη ή συσκευή είναι εν κινήσει, παραδείγματος χάριν φορτίζει από κάποια πηγή ρεύματος σε κάποιο όχημα ή από power bank κλπ., η εφαρμογή παύει να παρακολουθεί ωστόσο ενεργοποιηθεί η τοποθεσία από τις ρυθμίσεις της συσκευής ("No location or device is moving.").





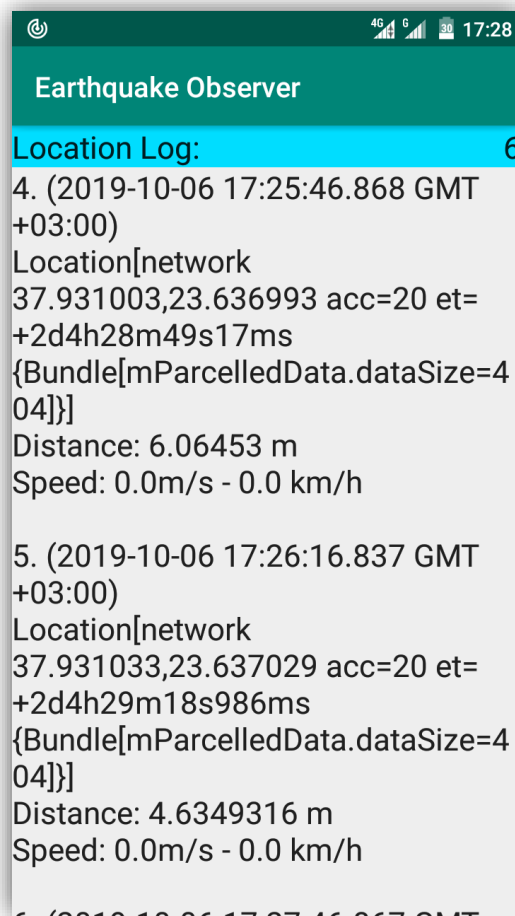
Κάθε φορά που το σύστημα αναγνωρίζει έναν σεισμό όλες οι συσκευές λαμβάνουν μία ειδοποίηση, ανεξαρτήτως από το αν εκείνη τη στιγμή τρέχει η εφαρμογή στο παρασκήνιο (δηλαδή παρακολουθεί για σεισμούς), η οποία έχει πληροφορίες σχετικά με τον σεισμό (την ώρα που άρχισε και τελείωσε ο σεισμός, πόσες συσκευές συνέβαλλαν για την αναγνώριση κα.). Σε περίπτωση που δεν υπάρχει σύνδεση στο διαδίκτυο, ο χρήστης θα λάβει την ειδοποίηση μόλις συνδεθεί στο διαδίκτυο.





- **Επιπλέον λειτουργίες διαχειριστή (admin)**

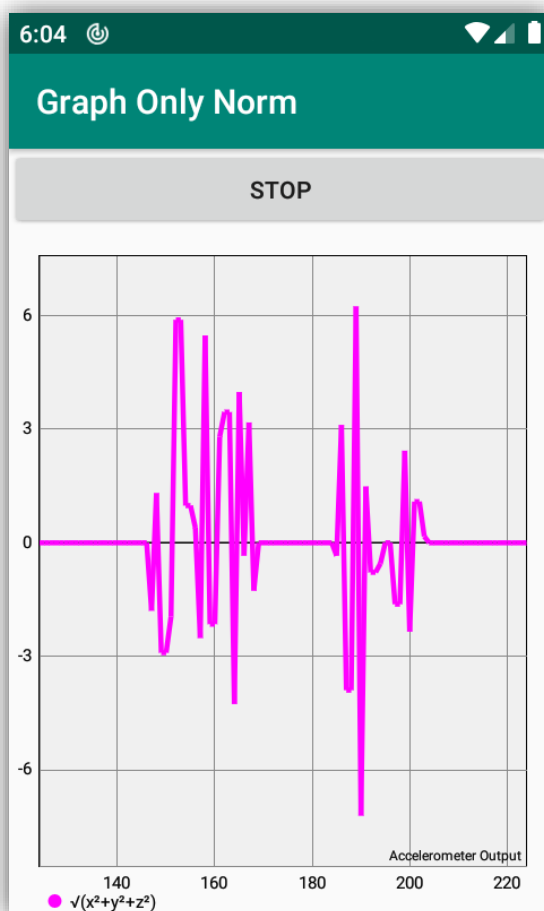
Προβολή πληροφοριών σχετικά με την τοποθεσία της συσκευής, όπως την ακριβή τοποθεσία (Latitude, Longitude), το timestamp της εκάστοτε τοποθεσίας, την απόσταση της από την προηγούμενη τοποθεσία κα.





Προβολή γραφικής αναπαράστασης (plot) των μετρήσεων του επιταχυνσιομέτρου:

- Graph Only Norm: γραφική αναπαράσταση μόνο της κανονικοποιημένης μέτρησης (GraphOnlyNormActivity, βλέπε σελ. 21-22).
- Graph All: γραφική αναπαράσταση των μετρήσεων των αξόνων x, y, z, της κανονικοποιημένης μέτρησης και της μέτρησης εφαρμόζοντας high-pass filter (GraphAllActivity).





5. Αρχιτεκτονική συστήματος

- **Εργαλεία & γλώσσες που χρησιμοποιήθηκαν:**

Για την ανάπτυξη της εφαρμογής χρησιμοποιήθηκε το Android Studio, το οποίο αποτελεί το επίσημο περιβάλλον ανάπτυξης εφαρμογών για Android συσκευές. Κατασκευάζεται από την Google σε συνεργασία με την JetBrains και είναι βασισμένο πάνω στο IntelliJ IDEA της JetBrains. Η εργασία αναπτύχθηκε χρησιμοποιώντας τις γλώσσες Java και XML (Cross Markup Language).

Ενώ για το backend στο οποίο γίνεται χρήση των Cloud Functions της Firebase, χρησιμοποιήθηκε το VS Code (source-code editor που έχει κατασκευαστεί από τη Microsoft) για την ανάπτυξη του κώδικα και το Firebase CLI (βοηθητικό πρόγραμμα που χρησιμοποιείται για τη διαχείριση των Firebase projects και μέσω αυτού αναπτύσσονται και διαχειρίζονται οι λειτουργίες των Cloud Functions). Χρησιμοποιήθηκε η γλώσσα Typescript (υποσύνολο της Javascript, με πιο αυστηρούς κανόνες ώστε να επιβάλλεται σωστότερος τρόπος γραφής του κώδικα και να αποφεύγονται συνήθη λάθη).

Συνολικά, τα εργαλεία της Firebase που χρησιμοποιούνται είναι:

- Authentication (μόνο Google provider)
- Realtime Database
- Cloud Functions (χρησιμοποιούνται Realtime Database triggers)
- Crashlytics & Analytics (χρησιμοποιούνται για debugging, δεν προσθέτουν κάποια επιπλέον λειτουργικότητα στην εφαρμογή)
- Cloud Messaging (χρησιμοποιούνται για να λαμβάνουν ειδοποίηση οι χρήστες σε περίπτωση σεισμού, μέσω των Cloud Functions)

- **Βάση**

Η βάση που χρησιμοποιήθηκε είναι η Realtime Database της Firebase, η οποία βρίσκεται στο cloud (δεν είναι δηλαδή τοπική), ώστε ανά πάσα στιγμή η κάθε συσκευή να “αναφέρει” τις μετρήσεις του επιταχυνσιόμετρου, όταν αυτό είναι απαραίτητο. Στη βάση αποθηκεύονται επιπλέον πληροφορίες σχετικά με:

- τους χρήστες (users) καθώς και ποιοι χρήστες είναι διαχειριστές (admins),



- ο τις συσκευές (devices),
- ο τις μετρήσεις του επιταχυνσιόμετρου, μαζί με σχετικές πληροφορίες όπως: τοποθεσία, ημερομηνία & ώρα κα. (active-events, saved-events) και
- ο τους σεισμούς που έχουν καταγραφεί (earthquakes).

Επιπλέον πληροφορίες που αποθηκεύονται στη βάση έχουν να κάνουν με τις ρυθμίσεις της εφαρμογής, όπως την περίοδο δειγματοληψίας μετρήσεων του επιταχυνσιόμετρου (sampling period), το χρονικό διάστημα που απαιτείται να περάσει ώστε να θεωρηθεί ένα συμβάν (EarthquakeEvent) ότι πρέπει να “αναφερθεί” (να γραφτεί στη βάση) κα. (settings).

- **Τρόπος λειτουργίας της εφαρμογής Earthquake Observer**

1. *Δημιουργία λογαριασμού και σύνδεση (SignInActivity.java)*

Για να χρησιμοποιηθεί η εφαρμογή αρχικά ο χρήστης πρέπει να δημιουργήσει έναν λογαριασμό, απλά επιλέγοντας έναν λογαριασμό Google της συσκευής ή δημιουργώντας έναν καινούριο (απαιτείται όμως υποχρεωτικά λογαριασμός Google) ή σε περίπτωση προηγούμενης εγκατάστασης της εφαρμογής απλά να συνδεθεί επιλέγοντας τον προηγούμενο λογαριασμό που είχε δημιουργήσει για να συνδεθεί. Και αφού ολοκληρωθεί επιτυχώς η δημιουργία του λογαριασμού (προστεθεί δηλαδή στους χρήστες του Firebase Authentication), αποθηκεύονται σχετικές πληροφορίες στη βάση (χρησιμοποιώντας την κλάση User). Μετά την ολοκλήρωση δημιουργίας λογαριασμού και σύνδεσης του στην εφαρμογή, δε θα χρειαστεί ο χρήστης να επαναλάβει τη διαδικασία δημιουργίας λογαριασμού και σύνδεσης στην εφαρμογή (εκτός αν την απεγκαταστήσει).

2. *Ρύθμιση της συσκευής (ConfigDeviceActivity.java)*

Στη συνέχεια αφού ολοκληρωθεί επιτυχώς η δημιουργία του λογαριασμού, πρέπει να διαμορφωθεί κατάλληλα η εφαρμογή, δηλαδή να υπολογιστεί η τιμή ισορροπίας του επιταχυνσιόμετρου σε κατάσταση ηρεμίας. Πρέπει η συσκευή να είναι τοποθετημένη σε ένα επίπεδο και να μην μετακινείται, μέχρι να ολοκληρωθεί η διαδικασία. Σε περίπτωση που κουνηθεί η συσκευή, η διαδικασία θα αρχίσει από την αρχή. Αφού ολοκληρωθεί επιτυχώς η διαμόρφωση της εφαρμογής, αποθηκεύονται πληροφορίες σχετικές με:

- ο την συσκευή και τον αισθητήρα (επιταχυνσιόμετρο) που διαθέτει,



- ο σε ποιο χρήστη ανήκει η συσκευή (ένας χρήστης (user) μπορεί να έχει παραπάνω από μία συσκευές),
- την ημερομηνία & ώρα δημιουργίας της συσκευής κα. (χρησιμοποιώντας την κλάση Device).

Μετά την ολοκλήρωση ρύθμισης της συσκευής, δε θα χρειαστεί ο χρήστης να επαναλάβει τη διαδικασία ρύθμισης εφαρμογή (εκτός αν την απεγκαταστήσει).

3. Καταγραφή σεισμών (*ObserverService.java*)

Για την παρακολούθηση των μετρήσεων του επιταχυνσιόμετρου χρησιμοποιείται μία Foreground Service (*ObserverService*, η οποία κληρονομεί από την κλάση *Service*), τρέχει δηλαδή στο παρασκήνιο (background) χωρίς να διακόπτει ή ενοχλεί το χρήστη σε οτιδήποτε άλλο κάνει, παρά μόνο παρουσιάζει μία ειδοποίηση (ongoing notification), η οποία φεύγει όταν για κάποιο λόγο σταματήσει η παρακολούθηση για σεισμούς. Για να καταφέρουμε να προγραμματίσουμε την παρακολούθηση, μέσω της *ObserverService*, μόνο όταν η συσκευή φορτίζει και έχει σύνδεση στο διαδίκτυο, γίνεται χρήση του *JobScheduler* χρησιμοποιώντας τους αντίστοιχους περιορισμούς. Για να διακοπεί η παρακολούθηση πρέπει να συμβεί τουλάχιστον ένα από τα παρακάτω:

- να αποσυνδεθεί η συσκευή από το ρεύμα (*ACTION_POWER_DISCONNECTED* της κλάσης *Intent*) ή
- να διακοπεί η σύνδεση στο διαδίκτυο (*CONNECTIVITY_ACTION* της κλάσης *ConnectivityManager*)

Για την “ανίχνευση” των παραπάνω χρησιμοποιείται ένας *Receiver* (*StopObserverReceiver*, ο οποίος κληρονομεί από την κλάση *Receiver*), ο οποίος “ενεργοποιείται” κατά την δημιουργία (*onCreate*) της *ObserverService* και “απενεργοποιείται” κατά τον τερματισμό (*onDestroy*) της και παρακολουθεί για τις παραπάνω “εκπομπές” (broadcasts).

Όσον αφορά τον τρόπο παρακολούθησης των μετρήσεων και την αποθήκευσή τους στη βάση στο cloud, όταν αυτό είναι απαραίτητο χρησιμοποιείται η κλάση *EarthquakeManager*. Επειδή οι μετρήσεις που μας δίνει το επιταχυνσιόμετρο είναι συνολικά 3 (x, y, z, μία για κάθε άξονα αντίστοιχα) γίνεται μία κανονικοποίηση (normalization) αυτών, ώστε να μην



χρειάζεται να παρατηρούμε ταυτόχρονα και τις τρεις τιμές και συνεπώς να μην αποθηκεύονται και οι τρεις, χρησιμοποιώντας τον τύπο:

$$\sqrt{x^2 + y^2 + z^2}$$

Με αυτόν τον τρόπο, έχοντας ρυθμίσει το επιταχυνσιόμετρο να μας δίνει μετρήσεις ανά συγκεκριμένη περίοδο (sampling period, 100 milliseconds) γεμίζουμε μία λίστα συγκεκριμένου μεγέθους (χωρητικότητας 10 μετρήσεων) με τις μετρήσεις, αφού πρώτα τις κανονικοποιήσουμε, χρησιμοποιώντας τον παραπάνω τύπο. Τα βήματα που ακολουθούνται είναι τα εξής:

- i. Κάθε φορά που γεμίζει η λίστα ελέγχουμε αν ο μέσος όρος των μετρήσεων ξεπερνάει ένα ορισμένο κατώφλι (sensorValueThreshold). Αν ναι δημιουργούμε τοπικά ένα EarthquakeEvent, το οποίο χαρακτηρίζεται ως minor event, και στη συνέχεια αδειάζουμε τη λίστα για να γεμίσει εκ νέου με τις καινούριες μετρήσεις.
- ii. Αν στην επόμενη επανάληψη συνεχίσει ο μέσος όρος να ξεπερνάει την τιμή του κατωφλίου, τότε ενημερώνουμε ορισμένα πεδία του υπάρχοντος τοπικού EarthquakeEvent, το οποίο ακόμα χαρακτηρίζεται ως minor. Αν όχι τότε το διαγράφουμε.
- iii. Αν το τοπικό EarthquakeEvent συνεχίσει να υπάρχει και η χρονική του διάρκεια (duration) ξεπεράσει ένα ορισμένο όριο (minEventDuration) το EarthquakeEvent προάγεται σε major και αποθηκεύεται στη βάση στα active-events (αποθηκεύονται ανά συσκευή, ανά πάσα στιγμή υπάρχει το πολύ ένα ενεργό EarthquakeEvent), όπου και θα ενημερώνεται σε κάθε επόμενη επανάληψη, παράλληλα με το τοπικό, όσο συνεχίζει να υφίσταται (όπως ακριβώς στο ii).
- iv. Όταν δεν ισχύει ο έλεγχος που κάνουμε στο ii, το τοπικό EarthquakeEvent, major πλέον, διαγράφεται και στη βάση από τα active-events μεταφέρεται στα saved-events (αποθηκεύονται ανά συσκευή, μία συσκευή μπορεί να έχει πολλά αποθηκευμένα EarthquakeEvent).

- **Τρόπος λειτουργίας της εφαρμογής Earthquake Observer Server**

Παράλληλα τρέχει και μία άλλη εφαρμογή (Earthquake Observer Server) σε μία μόνο συσκευή η οποία συνεχώς φορτίζει και έχει σύνδεση στο διαδίκτυο, στην οποία τρέχει μία Foreground Service (ServerService, η οποία κληρονομεί από την κλάση Service). Σκοπός της



είναι να παρακολουθεί τις εγγραφές στα active-events στη βάση. Αν το πλήθος των EarthquakeEvent ξεπεράσει κάποιο ορισμένο όριο (minDeviceCount), τότε δημιουργείται ένα αντικείμενο Earthquake το οποίο περιλαμβάνει πληροφορίες όπως:

- τη λίστα των συσκευών (MinimalDevice) που ανέφεραν το συγκεκριμένο σεισμό (για κάθε συσκευή αποθηκεύονται πληροφορίες όπως deviceId, eventId, timestamp, latitude, longitude)
- τη χρονική στιγμή που άρχισε ο σεισμός (startTime)
- τη χρονική στιγμή που τελείωσε ο σεισμός (endTime)
- το αν είναι ενεργός ο σεισμός ή όχι (isActive)

και αποθηκεύεται στη βάση στους earthquakes.

Η παραπάνω ενέργεια με τη σειρά της ενεργοποιεί έναν database trigger που έχει δημιουργηθεί με τη βοήθεια των Cloud Functions και όλοι οι χρήστες λαμβάνουν μία ειδοποίηση (μέσω του Cloud Messaging της Firebase) με πληροφορίες σχετικά με το σεισμό που έλαβε χώρα.

Όσο το πλήθος των EarthquakeEvent στα active-events είναι μεγαλύτερο του ορίου, σε περίπτωση που προστεθεί και άλλη συσκευή με ενεργό EarthquakeEvent στα active-events, το υπάρχον αντικείμενο Earthquake ενημερώνεται, δηλαδή προστίθεται στη λίστα των συσκευών οι αντίστοιχες πληροφορίες.

Όταν το πλήθος των EarthquakeEvent στα active-events γίνει μικρότερο του ορίου ενημερώνονται η χρονική στιγμή που τελείωσε ο σεισμός και χαρακτηρίζεται ανενεργός (isActive: false).



6. Συμπεράσματα και μελλοντικές επεκτάσεις

Στην παρούσα εργασία αναπτύχθηκε μία εφαρμογή που έχει ως σκοπό να καταγράφει σεισμούς χρησιμοποιώντας ένα δίκτυο Android™ συσκευών, ως σεισμολογικούς σταθμούς και να ενημερώνει τους χρήστες, σε πραγματικό χρόνο, για τους σεισμούς που λαμβάνουν χώρα. Κάποιες περαιτέρω επεκτάσεις που μπορούν να γίνουν είναι οι ακόλουθες:

- Για κάθε σεισμό αφού αποθηκευτούν οι πληροφορίες σχετικά με την τοποθεσία της κάθε συσκευής, την χρονική στιγμή που έφτασε σε αυτήν κλπ., στη συνέχεια να γίνεται και υπολογισμός του επικέντρου του σεισμού που έλαβε χώρα.
- Το μεγάλο πλήθος των δεδομένων που θα συλλεχθούν κατά τη διάρκεια της λειτουργίας της εφαρμογής, θα μπορούσαν να αξιοποιηθούν για ερευνητικούς σκοπούς και περαιτέρω μελέτη των σεισμικών φαινομένων, εκτός των πλαισίων της εφαρμογής. Αλλά και στα πλαίσια της εφαρμογής, να βελτιωθεί η αναγνώριση των σεισμών με την αναζήτηση κάποιου μοτίβου στους σεισμούς, που θα έχει ως καλύτερη αναγνώριση των σεισμών και αποφυγή λανθασμένης αναγνώρισης συμβάντων ως σεισμών.
- Κάποιες ρυθμίσεις όπως είναι η συχνότητα δειγματοληψίας των μετρήσεων του επιταχυνσιομέτρου (samplingPeriod), η ελάχιστη διάρκεια ενός συμβάντος για να “αναφερθεί” από τη συσκευή (minEventDuration) κλπ., να είναι δυνατόν να τροποποιηθούν, από ένα χρήστη με δικαιώματα διαχειριστή και στη συνέχεια να ενημερωθούν αντίστοιχα οι εφαρμογές που ήδη είναι σε λειτουργία. Έτσι θα γίνεται σωστότερα η αναγνώριση των σεισμών και αποφευχθούν λάθος αναγνωρίσεις.



7. Βιβλιογραφία

Broadcast Receivers:

- <https://developer.android.com/guide/components/broadcasts>
- <https://android.jlelse.eu/broadcast-receivers-for-beginners-a9d7aa03fb76>
- <https://www.vogella.com/tutorials/AndroidBroadcastReceiver/article.html>

Services & Background processing:

- <https://android-developers.googleblog.com/2018/10/modern-background-execution-in-android.html>
- <https://developer.android.com/training/monitoring-device-state/doze-standby.html>
- <https://android-developers.googleblog.com/2018/12/effective-foreground-services-on-android-11.html>
- <https://www.vogella.com/tutorials/AndroidTaskScheduling/article.html>
- <https://medium.com/@varun93342/difference-between-thread-service-and-async-task-in-android-d6e37960e56c>
- <https://www.vogella.com/tutorials/AndroidServices/article.html#platform-service-and-custom-services>

Sensors:

- https://developer.android.com/guide/topics/sensors/sensors_overview
- https://developer.android.com/guide/topics/sensors/sensors_motion

Location:

- <https://developer.android.com/training/location/receive-location-updates>
- <https://developer.android.com/guide/topics/location/strategies>
- https://www.tutorialspoint.com/android/android_location_based_services

Realtime Database:

- <https://firebase.google.com/docs/database/android/start>

Authentication:

- https://firebase.google.com/docs/auth/android/google-signin?utm_source=studio
- <https://developers.google.com/identity/sign-in/android/sign-in>



- <https://github.com/firebase/quickstart-android/blob/3377e7b99c1e4603e93fa3e96f8f6f77ab882427/auth/app/src/main/java/com/google/firebase/quickstart/auth/java/GoogleSignInActivity.java#L74-L78>

Cloud Functions:

- <https://www.youtube.com/playlist?list=PLI-K7zZEsYLkPZHe41m4jfAxUi0JLgSM>
- <https://medium.com/codingthesmartway-com-blog/introduction-to-firebase-cloud-functions-c220613f0ef>
- <https://medium.com/@jerinamathews/send-firebase-cloud-messaging-fcm-to-a-topic-using-cloud-function-when-realtime-database-value-fa78fa758549>
- <https://cloud.google.com/functions/docs/calling/realtime-database>
- <https://firebase.googleblog.com/2018/01/why-you-should-use-typescript-for.html>

Cloud Messaging (FCM):

- https://firebase.google.com/docs/cloud-messaging/concept-options#notifications_and_data_messages
- <https://firebase.google.com/docs/cloud-messaging/android/receive>
- <https://firebase.googleblog.com/2018/09/handle-fcm-messages-on-android.html>
- <https://firebase.googleblog.com/2019/02/life-of-a-message.html>
- <https://android-developers.googleblog.com/2018/09/notifying-your-users-with-fcm.html>

Παρόμοιες εφαρμογές στο Google Play™:

- <https://play.google.com/store/apps/details?id=edu.berkeley.bsl.myshake>
- <https://play.google.com/store/apps/details?id=com.zizmos.earthquake>
- <https://play.google.com/store/apps/details?id=com.finazzi.distquake>
- <https://play.google.com/store/apps/details?id=de.gempa.android.eqinfo>