ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΜΑΘΗΜΑ: ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΔΙΚΤΥΑ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΕΣ

Στα πλαίσια του μαθήματος καλείστε να υλοποιήσετε ένα πρωτότυπο σύστημα παρακολούθησης κίνησης ενός δικτύου κορμού. Το συνολικό σύστημα θα αποτελείται από 3 επιμέρους υποσυστήματα:

i. Λογισμικό παρακολούθησης κακόβουλης κίνησης σε PC/Laptop.
ii. Λογισμικό διαχείρισης των κόμβων του δικτύου κορμού, διαμόρφωσης κανόνων για κακόβουλη κίνηση και παρουσίασης της πληροφορίας που αφορά τόσο στην κατάσταση του δικτύου όσο και τα στατιστικά της κίνησης.
iii. Λογισμικό παρακολούθησης των στατιστικών της δικτυακής κίνησης μέσα από Smartphone/PDA.

Οι τεχνολογίες που απαιτούνται και θα καλυφθούν στα πλαίσια του μαθήματος είναι οι παρακάτω:

- Παρακολούθηση IP κίνησης (IP traffic monitoring)
- Linux Interface Configuration
- Java SDK
- Web Services
- Android SDK

Το προτεινόμενο εκπαιδευτικό υλικό εμφανίζεται παρακάτω:

- "Understanding IP Addressing: Everything You Ever Wanted To Know", ελεύθερα διαθέσιμο από http://holdenweb.com/static/docs/3comip.pdf
- Java Tutorials: http://docs.oracle.com/javase/tutorial/
- Oracle Java Website: http://www.oracle.com/technetwork/java/javase/overview/index.html
- Eclipse Tutorial for Web Services: https://eclipse.org/webtools/jst/components/ws/tutorials/index.html
- Android Training: http://developer.android.com/training/index.html
- Android API Guides: http://developer.android.com/guide/components/index.html
- Android Emulator: http://developer.android.com/tools/help/emulator.html
- Useful Android Tutorials : http://www.vogella.com/tutorials/android.html

Το λειτουργικό σύστημα αναφοράς θα είναι Linux με Oracle Java EE 8. Η βάση δεδομένων που θα χρειαστεί στο δεύτερο τμήμα θα είναι MySQL Community Server 5.6.21. Το λειτουργικό σύστημα της Android συσκευής θα πρέπει αυστηρά να είναι Android 4.1 ή νεότερο. Το Eclipse θα είναι το εργαλείο ανάπτυξης της εφαρμογής.

Η άσκηση θα παραδοθεί σε τρία τμήματα. Στο πρώτο τμήμα, θα παραδώσετε το λογισμικό που παρακολουθεί την δικτυακή κίνηση μιας συσκευής τύπου PC/Laptop. Το δεύτερο παραδοτέο θα είναι ένα κεντρικό σύστημα-αθροιστής που θα διαμορφώνει τα PC/Laptop με το να στέλνει κανόνες για την παρακολούθηση κακόβουλης κίνησης, καθώς και θα συγκεντρώνει και θα παρουσιάζει την πληροφορία που συλλέγουν οι επιμέρους συσκευές PC/Laptop. Στο τρίτο τμήμα θα παραδώσετε το λογισμικό για Smartphones/Tablets.

ΠΡΩΤΟ ΠΑΡΑΔΟΤΕΟ

ΣΥΣΤΗΜΑ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΔΙΚΤΥΑΚΗΣ ΚΙΝΗΣΗΣ ΣΕ PC/LAPTOP

Το πρώτο παραδοτέο της εργασίας αποτελεί μια πολυνηματική (multithreading) εφαρμογή η οποία θα εκτελείται σε PC/Laptop. Το κύριο νήμα της εφαρμογής θα αναλαμβάνει τα εξής:

1.Την εκκίνηση ενός νήματος ανίχνευσης διεπαφών δικτύων (Network Interfaces) της συσκευής. Σκοπός του είναι να παρακολουθεί την συσκευή για τυχόν διαγραφή ή εμφάνιση κάποιου καινούργιου Interface. Σε περίπτωση που κάτι τέτοιο συμβεί τότε σταματά/ξεκινά ένα νήμα που υλοποιεί την λειτουργικότητα που περιγράφεται στο (4). Η εφαρμογή θα πρέπει να μπορεί να αντιληφθεί αλλαγές τόσο σε επίπεδο υλισμικού (π.χ. νέο USB Wireless) όσο και σε επίπεδο λογισμικού (π.χ. δημιουργία νέου interface μέσω της εντολής ifconfig) κατά την διάρκεια της εκτέλεσης της.

2. Την εκκίνηση ενός νήματος ταυτοποίησης PC/Laptop το οποίο θα αναλαμβάνει την εγγραφή της συσκευής PC/Laptop στον κόμβο αθροιστή. Κατά την εγγραφή το PC/Laptop θα στέλνει στον αθροιστή έναν μοναδικό σειριακό αριθμό τον οποίο θα διαθέτει η συσκευή (π.χ MAC address).

3. Την εκκίνηση ενός νήματος ενημέρωσης σχετικά με πρότυπα κακόβουλης κίνησης. Το νήμα αυτό θα αναλαμβάνει περιοδικά να στέλνει αιτήσεις στον αθροιστή για την ύπαρξη και λήψη νέων μοτίβων κακόβουλης κίνησης. Η απάντηση περιλαμβάνει 2 είδη κακόβουλης κίνησης: i) κακόβουλες IP διευθύνσεις (malicious IP addresses) που περιέχονται στην επικεφαλίδα ενός πακέτου (είτε source ή destination IP), ii) κακόβουλα μοτίβα συμβολοσειρών (malicious string patterns) που τυχαίνουν στο payload ενός πακέτου. Το νήμα ενημέρωσης αναλαμβάνει να προσθέτει τα νέα malicious IP/patterns σε μια διαμοιραζόμενη μνήμη μοτίβων κακόβουλης κίνησης (Malicious Pattern Shared Memory - MPSM), όπως φαίνεται στον Πίνακα 1, ούτως ώστε να την συμβουλεύονται με τη σειρά τους τα νήματα που παρουσιάζονται στο βήμα (4).

Πίνακας 1: Παράδειγμα μνήμης MPSM

Malicious IPs	Malicious String Patterns
195.134.65.27	http:*
8.8.8.8	[h-k]ello
....	....

4. Για κάθε Interface ξεκινά ένα νήμα που αναλαμβάνει να ελέγχει περιοδικά τα πακέτα που περνάνε από το εκάστοτε Interface. Για κάθε πακέτο που δειγματοληπτεί αναλαμβάνει να ψάξει τόσο για κακόβουλες IP όσο και για κακόβουλο περιεχόμενο στο payload του πακέτου. Το νήμα αυτό θα πρέπει να κρατά στατιστικά και να τα ενημερώνει σε μια διαμοιραζόμενη μνήμη στατιστικών κακόβουλης κίνησης (Statistics Malicious Pattern Shared Memory - S-MPSM), όπως φαίνεται στους Πίνακες 2 & 3.

Πίνακας 2: Παράδειγμα μνήμης S-MPSM

Interface Name	Interface IP	Malicious IP	Frequency
eth0	195.134.65.223	195.134.65.27	5
eth0	195.134.65.223	8.8.8.8	3
lo	127.0.0.1	195.134.65.27	6
lo	127.0.0.1	8.8.8.8	2

Πίνακας 3: Παράδειγμα μνήμης S-MPSM

Interface Name	Interface IP	Malicious String Patterns	Frequency
eth0	195.134.65.223	http:*	3
eth0	195.134.65.223	[h-k]ello	0
lo	127.0.0.1	http:*	2
lo	127.0.0.1	[h-k]ello	1

5. Την εκκίνηση ενός νήματος που θα αναλαμβάνει να δημιουργεί τις μνήμες MPSM & S-MPSM, πριν την δημιουργία των νημάτων που θα κάνουν την ανίχνευση των πακέτων. Ακόμη, το νήμα αυτό θα είναι υπεύθυνο να στέλνει την εικόνα των MPSM & S-MPSM περιοδικά στον αθροιστή.

Όπως φαίνεται και από την παραπάνω περιγραφή, διαχειριστής του συνόλου των νημάτων είναι το αρχικό νήμα που παρακολουθεί τις αλλαγές στον αριθμό των interfaces. Αυτό το νήμα πρέπει

i. Να εκκινήσει νέα νήματα αν εμφανιστούν νέα interfaces και
ii. Να τερματίσει νήματα αν ένα interface δεν υπάρχει πλέον.

Η εφαρμογή θα πρέπει να τερματίζει ομαλά από το command line του υπολογιστή με χρήση κάποιου σήματος (π.χ. Ctrl+C). Κατά τον τερματισμό, θα πρέπει να ενημερώνεται ο αθροιστής πως ο κόμβος πλέον δεν βρίσκεται στο δίκτυο.

ΠΑΡΑΔΟΧΕΣ

Στο πρώτο παραδοτέο πρέπει να αναπτυχθεί μόνο το λογισμικό που αφορά τις συσκευές PC/Laptop. Η επικοινωνία με τον αθροιστή θα είναι "εικονική". Νήματα που αναλαμβάνουν την αποστολή και λήψη δεδομένων από και προς τον αθροιστή, απλά θα εμφανίζουν μηνύματα στην κονσόλα. Η ολοκλήρωση της επικοινωνίας θα υλοποιηθεί στο δεύτερο παραδοτέο. Για την επιτυχή παρακολούθηση της κίνησης και τον έλεγχο της λειτουργικότητα της εφαρμογής στο πρώτο παραδοτέο, θα πρέπει η εφαρμογή να φορτώνει κατά την εκκίνηση ενδεικτικά Malicious IP/Patterns από κάποιο αρχείο τα οποία θα οριστούν από τον προγραμματιστή της εφαρμογής.

ΣΗΜΑΝΤΙΚΑ ΣΗΜΕΙΑ

Η υλοποίηση και εκτέλεση της εφαρμογής θα πρέπει:

i. Να υπακούει στις αρχές του αντικειμενοστρεφούς προγραμματισμού. Για το λόγο αυτό πρέπει να ορισθούν κατάλληλα κλάσεις για την δημιουργία των αντίστοιχων τύπων δεδομένων και νημάτων.
ii. Να είναι οσο το δυνατό παραμετροποιήσημη και δυναμική γίνεται.
iii. Να είναι ευσταθής. Ζητήματα συγχγονισμού και ασυνέπειες στην χρήση δεδομένων πρέπει να αποφεύγονται.

Συνιστάτε η χρήση έτοιμων βιβλιοθηκών που επιτρέπουν την αποδοτική ανίχνευση και ανάγνωση πακέτων από την κάρτα δικτύου. Μία προτεινόμενη είναι και η Jpcap (http://www.techrepublic.com/article/capture-network-packets-in-java-with-jpcap/) με σχετικό documentation και παραδείγματα χρήσης.

ΔΕΥΤΕΡΟ ΠΑΡΑΔΟΤΕΟ

ΣΥΣΤΗΜΑ ΑΘΡΗΣΤΗΣ

Το σύστημα αθροιστής θα υλοποιεί ένα Web Service (WS) το οποίο θα λαμβάνει την πληροφορία περιοδικά από τα PC/Laptops, θα τα αποθηκεύει σε μια βάση δεδομένων και θα τα παρουσιάζει σε αντίστοιχο γραφικό περιβάλλον. Οι διεπαφές που θα υλοποιεί για τον σκοπό αυτό συνοψίζονται στον Πίνακα 4.

Πίνακας 4: Λίστα με τις διεπαφές του Web Service

public boolean register(String nodeID)	Η διεπαφή αυτή δίνει τη δυνατότητα στα PC/Laptop να εγγραφούν στο σύστημα με ένα μοναδικό κωδικό. Επιστρέφεται η τιμή true/false σε περίπτωση επιτυχούς ή μη εγγραφής.
public MaliciousPatterns maliciousPatternRequest(String nodeID)	Η διεπαφή αυτή δίνει την δυνατότητα στο PC/Laptop να ζητήσει νέα malicious patterns που έχει ορίσει ο διαχειριστής του δικτύου (admin). Στο PC/Laptop στέλνονται μόνο τα patterns για τα οποία δεν είναι ενημερωμένος με βάση την πιο πρόσφατη στο παρελθόν αίτηση.
public void maliciousPatternsStatisticalReport(String nodeID, StatisticalReports m)	Η διεπαφή αυτή επιτρέπει στον PC/Laptop να στέλνει την πιο πρόσφατη εικόνα των interfaces του και στατιστικά κακόβουλης κίνησης που πέρασαν από αυτά στο διάστημα από την πιο πρόσφατη αναφορά μέχρι τώρα.
public boolean unregister(String nodeID)	Καλώντας την διεπαφή αυτή το εκάστοτε PC/Laptop μπορεί να απεγγραφεί από το σύστημα.
public boolean register(String username, String password, AvailableNodes nodes)	Η διεπαφή αυτή δίνει τη δυνατότητα στα Smartphone/Tablets να εγγραφούν στο σύστημα. Οι χρήστες που εγγραφονται στο συστημα πρεπει να ορισουν και τα διαθεσιμα PC/Laptops που έχει στην κατοχή του. Επιστρέφεται η τιμή true/false σε περίπτωση επιτυχούς ή μη εγγραφής.
public List<StatisticalReports> retrieveStatistics(String username, String password)	Η διεπαφή αυτή δίνει τη δυνατότητα στα Smartphone/Tablets να ζητούν στατιστικά για την τρέχουσα κατάσταση του δικτύου.
public String retrieveMaliciousPatterns(String username, String password)	Η διεπαφή αυτή δίνει τη δυνατότητα στον admin μέσω του Smartphone/Tablets του να ζητήσει Malicious IP/Patterns που έχουν ήδη οριστεί.
public void insertMaliciousPatterns(String username,String password, String malliciousIPString stringPatterns)	Η διεπαφή αυτή δίνει τη δυνατότητα στον admin μέσω του Smartphone/Tablets του να ορίσει νέα Malicious Patterns.

Πέρα από τις προαναφερόμενες διεπαφές, η εσωτερική λειτουργικότητα του αθροιστή περιλαμβάνει

1. Την ύπαρξη μιας εσωτερικής μνήμης που θα καταγράφει τους εγγεγραμμένους και ενεργούς κόμβους (PC/Laptop) και για το ποιά Malicious IP/Patters είναι ενήμεροι. Σε περίπτωση εισαγωγής/διαγραφής ενός κόμβου, θα πρέπει η μνήμη να ενημερώνεται κατάλληλα.

2. Την ύπαρξη μιας βάσης δεδομένων που θα κρατά εγγεγραμμένους κόμβους PC/Laptop (ενεργούς και μη), καθώς και τα στατιστικά που αυτοί στέλνουν. Τα στατιστικά θα αποτελούν ιστορικό, κάτι το οποίο σημαίνει ότι νέα ***MaliciousReports*** δεν κάνουν overwrite τα παλιά, αλλά προσθέτονται στο ιστορικό του αθροιστή. Η σχεδίαση της βάσης αυτής πρέπει να ικανοποιεί τα κριτήρια της 2ης Κανονικής Μορφής (2ΚΜ).

3. Την παροχή μιας γραφικής διεπαφής (Graphical User Interface - GUI) προς τον admin. Το GUI θα δίνει την δυνατότητα:

a. Ο admin να μπορεί να εισάγει νέα Malicious IP/Patterns.
b. Ο admin να μπορεί να επιβλέπει την κατάσταση του δικτύου (πληροφορίες για τους κόμβους και τα στατιστικά κίνησης)

ΠΑΡΑΔΟΧΕΣ

Στο παραδοτέο αυτό πρέπει να ολοκληρωθεί η πλήρης επικοινωνία αθροιστή και PC/Laptops. Η εκτέλεση των δύο αυτών οντοτήτων θα εξεταστεί σε διαφορετικές συσκευές. Για το λόγο αυτό πρέπει οι διευθύνσεις των συσκευών να είναι παραμετροποιήσημες. Η βάση δεδομένων μπορεί να δημιουργείται είτε από το πρόγραμμα του αθροιστή στην εκκίνηση (και πάντα μόνο μια φορά) είτε από SQL scripts. Το μηχάνημα που θα φιλοξενεί την βάση θα πρέπει να παραμετροποιείται. Το γραφικό περιβάλλον θα πρέπει να σχεδιαστεί και υλοποιηθεί δίχως την χρήση βοηθητικών εργαλίων.

Για την λειτουργικότητα που αφορά την επικοινωνία μεταξύ αθροιστή και Smartphone/Tablet, μπορείτε απλά να ορίσετε τις αντίστοιχες διεπαφές του Web Service, χωρίς να προχωρήσετε στην υλοποίηση τους.

ΣΗΜΑΝΤΙΚΑ ΣΗΜΕΙΑ

Η υλοποίηση και εκτέλεση της εφαρμογής θα πρέπει:

i. Να επιτυγχάνει τη σωστή επικοινωνία μεταξύ των δύο οντοτήτων (αθροιστή-PC/Laptop) σε οποιοδήποτε μηχάνημα και αν εκτελούνται.
ii. Να είναι ευσταθής. Ζητήματα συγχρονισμού και ασυνέπειες στην χρήση δεδομένων πρέπει να αποφεύγονται.
iii. Το γραφικό περιβάλλον πρέπει να είναι φιλικό και εύχρηστο προς τον χρήστη admin.

ΤΡΙΤΟ ΠΑΡΑΔΟΤΕΟ

ΛΟΓΙΣΜΙΚΟ ΠΑΡΟΥΣΙΑΣΗΣ ΚΑΙ ΕΙΣΑΓΩΓΗΣ ΔΕΔΟΜΕΝΩΝ ΣΕ Android ΣΥΣΚΕΥΕΣ

Στο τρίτο παραδοτέο καλείστε να σχεδιάσετε μια εφαρμογή για «έξυπνες» συσκευές (Smartphone, Tablet) σε λειτουργικό σύστημα Android που θα πρέπει να ικανοποιεί τις παρακάτω προδιαγραφές.

i. Την ενημέρωση των χρηστών Android σχετικά με την τρέχουσα κατάσταση του δικτύου κορμού.
ii. Την αποθήκευση των στατιστικών του αθροιστή σε μια τοπική βάση δεδομένων στη συσκευή (Smartphone/Tablet).
iii. Την γραφική διεπαφή της Android εφαρμογής για την πλοήγηση του χρήστη, την εισαγωγή και παρουσίαση δεδομένων.

Σκοπός της Android εφαρμογής είναι να παρουσιάζει στον χρήστη της, εγγραμμένο ή administrator (θα οριστεί διαφορετική λειτουργικότητα για την κάθε περίπτωση) τα στατιστικά στοιχεία τα οποία έχουν συλλεχθεί και αποθηκευτεί από τα παραδοτέα 1 & 2. Η Android εφαρμογή θα επικοινωνεί περιοδικά με τον αθροιστή και θα λαμβάνει την τρέχουσα κατάσταση του δικτύου. Σε περίπτωση που δεν υπάρχει σύνδεση με το δίκτυο, θα εμφανίζονται στον χρήστη τα τελευταία στοιχεία που έχουν αποθηκευτεί στην τοπική βάση της Android συσκευή κατά την τελευταία ενημέρωση. Τα στοιχεία αυτά θα πρέπει να είναι τα πιο πρόσφατα ενημερωμένα, δηλαδή αυτά που υπήρχαν στον αθροιστή πριν την λειτουργία εκτός σύνδεσης. Για τον λόγο αυτό θα πρέπει περιοδικά να γίνεται έλεγχος αν υπάρχει σύνδεση στο δίκτυο και όταν αυτή επιτευχθεί να αποστέλλεται το τελευταίο αίτημα που δεν έχουν αποσταλεί στον εξυπηρετητή. Θα πρέπει να σημειωθεί ότι η λειτουργικότητα αυτή θα να είναι ανεξάρτητη από το αν ο χρήστης έχει βγει από την εφαρμογή (οπότε αυτή έχει τερματίσει).

Η γραφική διεπαφή της Android εφαρμογής θα πρέπει να πληρεί τον παρακάτω σχεδιασμό/λειτουργικότητα.

1. Κατά την εκκίνηση της εφαρμογής θα υπάρχει μία αρχική οθόνη (activity) που θα επιτρέπει στο χρήστη την εισαγωγή των στοιχείων του στην εφαρμογή (login). Σε περίπτωση που δεν έχει ήδη λογαριασμό (register) θα τον παραπέμπει σε άλλη οθόνη (activity). Εφόσον υπάρχει ήδη χρήστης με αυτά τα στοιχεία θα εμφανίζεται στο χρήστη αντίστοιχο μήνυμα ώστε να εισάγει άλλα στοιχεία.

2. Στη συνέχεια εμφανίζεται στην οθόνη ένα menu που αποτελείται από εξής tabs:

a. Το tab παρουσίασης όπου εμφανίζονται στατιστικά των PC/Laptop που έχουν συλλεχθεί από τα πρώτα 2 παραδοτέα. Σε αυτό το tab ο χρήστης θα μπορεί να δει για κάθε PC/Laptop i) τα interface που διαθέτει, ii) στατιστικά που αφορούν σε Malicious IPs/Patterns. Επιπλέον, θα πρέπει ο χρήστης να μπορεί να φιλτράρει τα αποτελέσματα και να εμφανίζει τα δικά του PC/laptop όπως τα έχει ορίστει κατά την εγγραγή στον αθροιστή

b. Το tab με τα στοιχεία των τερματικών και των interfaces του. Σε αυτό το tab παρουσιάζονται με μορφή λίστας τα interfaces ενός συγκεκριμένου τερματικού που έχει επιλέξει ο χρήστης. Επιλέγοντας ένα στοιχείο της λίστας θα εμφανίζονται σε νέα οθόνη τα λεπτομερή χαρακτηριστικά του συγκεκριμένου interface.

3. Στην πετρίπτωση πού ο χρήστης της Android εφαρμογής είναι και admin του δικτύου, το menu επεκτείνεται κατά δυο ακόμα tabs, όπου:

a. Το tab για malicious patterns θα επιτρέπει στον admin να ορίζει νέα malicious ip & string patterns. Οι προσθήκες αυτές θα πρέπει να αποστέλονται στον αθροιστή.

b. Το tab διαχείρισης PC/Laptop που θα δίνει την δυνατότητα στον admin να διαγράψει κάποιο τερματικό από το δίκτυο.

Σε περίπτωση μη ύπαρξης σύνδεσης, οι ενέργεια του admin θα κρατούνται προσωρινά στην τοπική βάση και θα αποστέλονται στον αθροιστή μόλις αποκατασταθεί η σύνδεση.

ΠΑΡΑΔΟΧΕΣ

Στο παραδοτέο αυτό πρέπει να ολοκληρωθεί η επικοινωνία μεταξύ αθροιστή και Smartphone/Tablet.

i. public Boolean login (String username, String password) : Στην μέθοδο αυτή ο χρήστης εισάγει τα προσωπικά του στοιχεία και εάν η σύνδεση είναι επιτυχής τοτε η εφαρμογή εμφανίζει στον χρήστη κατευθείαν το tab παρουσιάσης με τα στατιστικά όλων των τερματικών.
ii. public Boolean logout () : Με την μέθοδο αυτή ο χρήστης θα μπορεί να κάνει αποσύνδεση από την εφαρμογή. Εάν η αποσύνδεση είναι επιτυχής τότε η εφαρμογή θα μεταβαίνει τον χρήστη απευθείας στην αρχική οθόνη (activity) που έχει περιγραφεί παραπάνω.
iii. Κατά την περίπτωση όπου ο χρήστης είναι συνδεδεμένος στην εφαρμογή με τα προσωπικά του στοιχεία και βγει από την εφαρμογή, χωρίς όμως να κάνει αποσύνδεση, η εφαρμογή να το θυμάται την επόμενη φορά που θα ξαναμπεί χωρίς να του ζητήσει ξανά τα προσωπικά του στοιχεία. Αυτό θα πρέπει να γίνεται με έναν έξυπνο και γρήγορο τρόπο ώστε να καθυστερεί η εφαρμογή.

ΣΗΜΑΝΤΙΚΑ ΣΗΜΕΙΑ

Η υλοποίηση και εκτέλεση της εφαρμογής θα πρέπει:

i. Η γραφική διεπαφή θα πρέπει να είναι φιλική ως προς τον χρήστη και να μπορεί να πλοηγηθεί στην εφαρμογή με ευκολία.
ii. Να είναι οσο το δυνατό παραμετροποιήσημη και δυναμική γίνεται.

Συμβουλευτείτε της οδηγίες που δίνονται στο Android API ώστε να βελτιώσετε την απόδοση της εφαρμογής σας (http://developer.android.com/training/best-performance.html). Λύσεις οι οποίες έχουν σαν αποτέλεσμα μηνύματα ANR (Android Not Responsive) θα έχουν σημαντική βαθμολογική επίπτωση.

ΟΔΗΓΙΕΣ

Η άσκηση θα παραδοθεί σε 3 επιμέρους παραδοτέα. Ιδανικά, με την παράδοση του 3ου παραδοτέου θα πρέπει να υπάρχει ένα πλήρες λειτουργικό σύστημα παρακολούθησης κακόβουλης κίνησης σε μια τοπολογία ενός δικτύου κορμού και διοχέτευσης της πληροφορίας σε αιτούντες χρήστες. Το σύστημα θα πρέπει να είναι σε θέση να διασυνδέει N κινητά και M PC/Laptops με έναν αθροιστή.

Παραδοτέο 1 (11-11-2014, μεσάνυχτα μέσω eclass):

Το πρώτο παραδοτέο θα περιέχει το σύστημα παρακολούθησης κίνησης σε PC/Laptop. Θα παραδώσετε μέσω e-class τον κώδικα, αναλυτικές σημειώσεις για την δομή του καθώς και τους ελέγχους που κάνατε για να επιβεβαιώσετε την ορθότητά του.

Παραδοτέο 2 (09-12-2014, μεσάνυχτα μέσω eclass):

Το δεύτερο παραδοτέο θα περιέχει το σύστημα του αθροιστή. Θα παραδώσετε μέσω e-class τον κώδικα, αναλυτικές σημειώσεις για την δομή του καθώς και τους ελέγχους που κάνατε για να επιβεβαιώσετε την ορθότητά του.

Παραδοτέο 3 (13-01-2015, μεσάνυχτα μέσω eclass):

Το τρίτο παραδοτέο θα περιέχει το πλήρες λειτουργικό σύστημα. Χρησιμοποιώντας τα δύο πρώτα παραδοτέα θα δομήσετε την Android εφαρμογή και το πλήρες σύστημα. Θα παραδώσετε μέσω e-class τον κώδικα, αναλυτικές σημειώσεις για την δομή του καθώς και τους ελέγχους που κάνατε για να επιβεβαιώσετε την ορθότητά του.

ΔΙΕΥΚΡΥΝΙΣΕΙΣ:

- Κάθε παραδοτέο μπορεί να υλοποιηθεί από ομάδα των 3 ατόμων (μέγιστο).
- Οι ημερομηνίες παράδοσης είναι **ανελαστικές.**
- Κατόπιν της παράδοσης, θα ακολουθεί εξέταση στα εργαστήρια όπου θα ελέγχεται η ορθότητα της υλοποίησης και θα ζητείται η συγγραφή κώδικα από όλα τα μέλη της ομάδας. **Θα πρέπει να είστε σε θέση να τεκμηριώσετε πλήρως τις επιλογές σας και την υλοποίησή σας τόσο θεωρητικά όσο και πρακτικά.**
- **Η βαθμολογία σας προκύπτει τόσο από την λειτουργικότητα της εφαρμογής όσο και από την σχεδίασή της.** Συνεπώς, δύο υλοποιήσεις που παρέχουν την ίδια λειτουργικότητα βαθμολογούνται διαφορετικά, ανάλογα πάντα με την ακολουθούμενη σχεδιαστική προσέγγιση.
- **Τα παραδοτέα κρίνονται όχι μόνο με βάση την ορθότητα αλλά και με βάση την απόδοση τους και την παρουσίασή τους (π.χ. η γραφική διεπαφή του 3ου παραδοτέου θα πρέπει να είναι ευκρινής και φιλική προς το χρήστη).**

- Δύο εβδομάδες πριν την παράδοση της εργασίας θα ανακοινώνονται αναλυτικές οδηγίες για την δομή του κειμένου που θα συνοδεύει το παραδοτέο. **Κατ' ελάχιστο το κείμενο πρέπει να περιέχει τα παρακάτω**:
    - Περιγραφή των κλάσεων (Classes) που υλοποιήσατε και ένα σχεδιάγραμμα που δείχνει πως αυτές αλληλεπιδρούν (Class Diagram).
    - Για κάθε κλάση να φτιάξετε έναν πίνακα που αναφέρει τις μεθόδους και να έχετε σύντομη περιγραφή (μία ή δύο προτάσεις) του ρόλου(/λειτουργία) της κάθε μεθόδου.
    - Σύντομη περιγραφή των δοκιμών/tests που κάνατε.
    - Ενσωμάτωση του παραδοτέου με το αποτέλεσμα της προηγούμενης εργασίας και εκ νέου διεξαγωγή ελέγχων.
- Θα διενεργηθούν δύο γραπτές εξετάσεις, Φεβρουάριο και Σεπτέμβριο. **Συμμετοχή σε αυτές θα έχουν μόνο όσοι επιτύχουν στην άσκηση.** Η γραπτή εξέταση θα είναι pass/fail. Οι επιτυχόντες στην εξέταση θα λάβουν τον συνολικό βαθμό της άσκησης σαν βαθμό του μαθήματος.
- Η βαθμολογία ορίζεται ως εξής: 0.35*1^{ο} παραδοτέο + 0.25*2^{ο} παραδοτέο + 0.25*3^{ο} παραδοτέο + 0.15*Συνολική λειτουργική έκδοση

Αποτυχία σε κάποιο τμήμα της άσκησης (ανεπάρκεια, <u>αντιγραφή</u> κλπ) σημαίνει αυτόματα και αποτυχία στο μάθημα. Η μεταξύ σας συνεργασία σε επίπεδο ανταλλαγής ιδεών και προτάσεων όχι μόνο δεν απαγορεύεται, το αντίθετο, ενθαρρύνεται. **Απαγορεύεται αυστηρά η χρήση τμημάτων λογισμικού από συναδέλφους σας ή τρίτες πηγές.**

Για την επίλυση απορριών θα δημιουργηθεί λίστα στην οποία καλείστε όλοι να εγγραφείτε. Επίλυση απορριών θα γίνεται μόνο μέσω της λίστας ή στα πλαίσια των διαλέξεων.