



ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

# Κατανεμημένα Συστήματα

## **Τρίτη Εργαστηριακή Άσκηση:**

Επικοινωνία Πελάτη-Εξυπηρετητή με τη χρήση RMI

Διδάσκων: Χρήστος Γκουμόπουλος

Εργαστηριακός Διδάσκοντας: Φακής Αλέξανδρος

## Εισαγωγή

Η απομακρυσμένη επίκληση μεθόδων (RMI), επιτρέπει σε εφαρμογές την κλήση μεθόδων από απομακρυσμένα αντικείμενα, για το διαμοιρασμό πόρων και επεξεργαστικού φόρτου μεταξύ των συστημάτων. Σε αντίθεση με άλλα συστήματα απομακρυσμένης εκτέλεσης που επιτρέπουν μόνο τη μεταφορά απλών τύπων δεδομένων ή συγκεκριμένων δομών, το RMI επιτρέπει σε οποιοδήποτε αντικείμενο της Java να χρησιμοποιηθεί, ακόμα και αν ο εξυπηρετητής δε το έχει ξανασυναντήσει. Το RMI επιτρέπει τόσο στον πελάτη όσο και στον εξυπηρετητή να φορτώσουν δυναμικά νέους τύπους αντικειμένων.

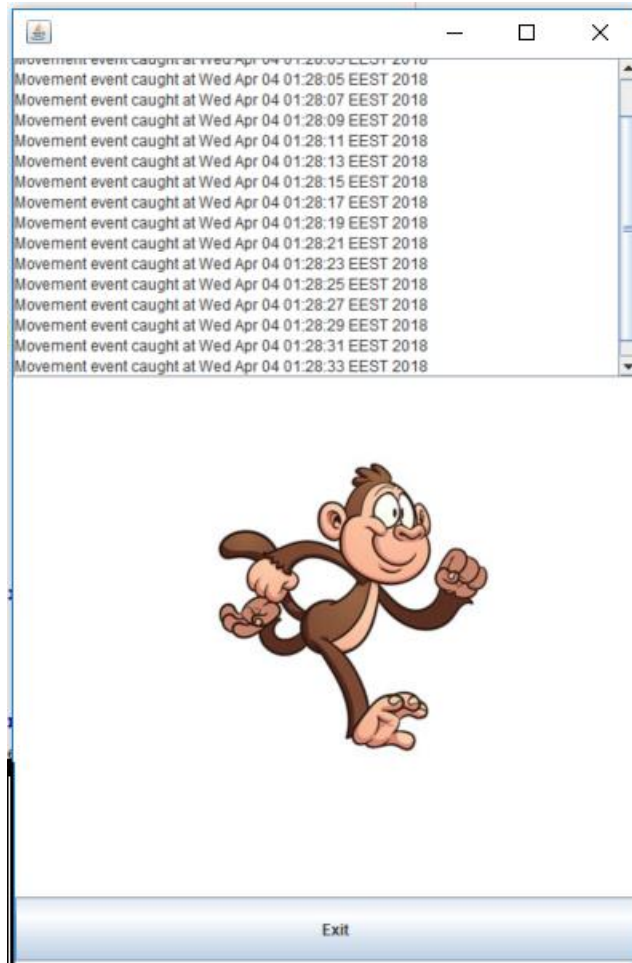
Οι εφαρμογές RMI συνήθως αποτελούνται από δύο ξεχωριστά προγράμματα : τον εξυπηρετητή (server) και τον πελάτη (client). Μια τυπική εφαρμογή εξυπηρετητή δημιουργεί μερικά απομακρυσμένα αντικείμενα, αναφορές σε αυτά ώστε να είναι προσβάσιμα, και περιμένει τους πελάτες να καλέσουν μεθόδους πάνω σε αυτά. Μια τυπική εφαρμογή πελάτη καλεί μεθόδους πάνω σε απομακρυσμένα αντικείμενα μέσω μιας απομακρυσμένης αναφοράς. Το RMI παρέχει το μηχανισμό επικοινωνίας ανάμεσα στον εξυπηρετητή και στον πελάτη. Τέτοιες εφαρμογές αναφέρονται και ως εφαρμογές κατακερμασμένων αντικειμένων (distributed object applications).

Θα πρέπει να χρησιμοποιήσετε τον μηχανισμό επικοινωνίας των RMI για την υλοποίηση μιας εφαρμογής πελάτη-εξυπηρετητή. Πιο συγκεκριμένα, οι διάφοροι πελάτες οι οποίοι είναι συνδεδεμένοι και εγγεγραμμένοι στον κεντρικό εξυπηρετητή θα πρέπει να ειδοποιούνται για διάφορα γεγονότα τα οποία λαμβάνουν χώρα σε αυτόν.

## Σενάριο

Σε ένα ζωολογικό κήπο, εργάζονται τέσσερις φύλακες οι οποίοι έχουν προσληφθεί για την φύλαξη του χώρου. Ο κάθε φύλακας ενημερώνεται ξεχωριστά για οποιαδήποτε κίνηση, από ένα κεντρικό σύστημα παρακολούθησης, στο οποίο είναι συνδεδεμένος, και το οποίο αποτελείται από κάποιες κάμερες που έχουν τοποθετηθεί σε διάφορα σημεία του κήπου και ελέγχεται από έναν RMI εξυπηρετητή. Κάθε κάμερα είναι εξοπλισμένη με έναν ενσωματωμένο αισθητήρα ανίχνευσης κίνησης και έτσι κάθε φορά που ο αισθητήρας κάποιας κάμερας ανιχνεύει κίνηση, ο κεντρικός εξυπηρετητής δημιουργεί ένα νέο συμβάν. Κατά την δημιουργία του, δίνεται σε αυτό μια χρονοσφραγίδα και αποστέλλεται σε κάθε φύλακα ξεχωριστά, μαζί με ένα αρχείο εικόνας που περιέχει ένα στιγμιότυπο της κίνησης που ανιχνεύτηκε. Η εφαρμογή παρακολούθησης του κάθε φύλακα υλοποιεί ένα απλό GUI στο οποίο καταγράφονται τα διάφορα γεγονότα που έλαβε και εμφανίζεται σε αυτό **μόνο** η τελευταία εικόνα που έχει αποσταλεί προς αυτόν. Ο εξυπηρετητής θα πρέπει να ενημερώνει ταυτόχρονα όλους τους φύλακες του κήπου. Επειδή οι φύλακες λειτουργούν σε διαφορετικές βάρδιες, κατά την έναρξη της βάρδιας του, ένας φύλακας εγγράφεται στην υπηρεσία του εξυπηρετητή, ενώ κατά τη λήξη της, κάνει απεγγραφή ώστε να σταματήσει να ενημερώνεται για νέα συμβάντα (subscribe/unsubscribe). Να δημιουργήσετε κατάλληλα μηνύματα διαλόγου για να τις παραπάνω ενέργειες.

Ο εξυπηρετητής εξομοιώνει την λειτουργία του αισθητήρα κίνησης παράγοντας διαδοχικά συμβάντα κίνησης με μια τυχαία χρονική απόσταση μεταξύ τους της τάξης των μερικών δευτερολέπτων (π.χ. από 2 έως 5). Για να υπάρχει μια εναλλαγή της εικόνας μεταξύ διαδοχικών συμβάντων που λαμβάνουν οι φύλακες, σας δίνονται 4 png αρχεία τα οποία θα πρέπει να χρησιμοποιήσετε ως στιγμιότυπα της κάμερας που αποστέλλονται κατά την ανίχνευση κίνησης.



*Εικόνα 1: Ενδεικτικό στιγμιότυπο της εφαρμογής*

Η υλοποίηση της εφαρμογής πελάτη του κάθε φύλακα θα πρέπει να λειτουργεί με τέτοιο τρόπο ώστε να αποφεύγεται το rolling (επαναληπτικά ερωτήματα προς τον εξυπηρετητή) για την εμφάνιση συμβάντων. Αντίθετα, θα πρέπει ο κάθε φύλακας να ειδοποιείται απευθείας από τον εξυπηρετητή μέσω κατάλληλων callback μεθόδων. Αυτό σημαίνει πως ο κάθε πελάτης θα μπορεί να κάνει εξαγωγή κάποιων μεθόδων του, τις οποίες θα μπορεί να καλεί απομακρυσμένα ο εξυπηρετητής.

## Τεχνικές Λεπτομέρειες

- Θα πρέπει να υλοποιήσετε την παραπάνω κατανεμημένη εφαρμογή χρησιμοποιώντας Java RMI. Η εφαρμογή θα πρέπει να υλοποιηθεί σε περιβάλλον NetBeans(ή όποιο άλλο IDE επιθυμείτε) και θα πρέπει να μπορεί να εκτελείται μέσα από το περιβάλλον του εργαλείου χωρίς να χρειάζονται εξωτερικές παρεμβάσεις (π.χ. Εξωτερική εκτέλεση του registry).
- Επιλέξτε ότι δομή επιθυμείτε για την διατήρηση της σύνδεσης του εξυπηρετητή με τον κάθε φύλακα.
- Για να ορίσετε οποιοδήποτε μονοπάτι για την διαχείριση αρχείων, κάντε χρήση ενός σχετικού και όχι απόλυτου μονοπατιού ώστε να είναι εφικτή η διόρθωση της εργασίας.
- Εφόσον δεν προβλέπεται η δυναμική φόρτωση κλάσεων από το ίδιο δίκτυο μπορείτε να μην υλοποιήσετε κάποιον security manager ή security policy για την εφαρμογή αυτή.
- Δημιουργήστε τυχαία γεγονότα από την μεριά του εξυπηρετητή, αποστέλλοντας κάθε φορά και μια τυχαία εικόνα ως στιγμιότυπο από αυτές που σας δίνονται. Ανανεώστε κατάλληλα το γραφικό σας για να εμφανίζονται σωστά οι αλλαγές.

## Οδηγίες Παράδοσης :

Τελική ημερομηνία παράδοσης της εργασίας: **18/04/2018**.

Θα πρέπει να σταλεί ένα αρχείο rar/zip με όνομα ΑριθμόςΜητρώου\_Lab03. Στο αρχείο θα περιέχονται τα εξής:

- Μια αναφορά σε μορφή pdf με οθόνες εκτέλεσης του προγράμματός σας καθώς και ειδικές οδηγίες για την εκτέλεση της εφαρμογής. Στην αρχή του αρχείου θα αναγράφεται ο αριθμός μητρώου και το ονοματεπώνυμο σας.
- Τα αρχεία .java των project σας με σχολιασμό στα πιο βασικά σημεία.
- Η υποβολή κοινών απαντήσεων από διαφορετικούς φοιτητές δεν επιτρέπεται και θεωρείται ως ΑΝΤΙΓΡΑΦΗ. Η αντιγραφή έχει ως αποτέλεσμα το ΜΗΔΕΝΙΣΜΟ ΤΗΣ ΕΡΓΑΣΙΑΣ ΣΥΝΟΛΙΚΑ.