

περιλαμβάνονται στα στοιχεία σύνδεσης στον server (socket), το thread στο οποίο γίνεται η επικοινωνία, οι ρυθμίσεις του παιχνιδιού και τα στοιχεία διεπαφών που είναι απαραίτητα για την επικοινωνία χρήστη και εφαρμογής. Οι μέθοδοί της περιλαμβάνουν την διαχείριση των μηνυμάτων που ανταλλάσσει η εφαρμογή με τον server και την αντίστοιχη μεταβολή των διεπαφών εξαιτίας αυτών.

Περιγραφή Λειτουργιών

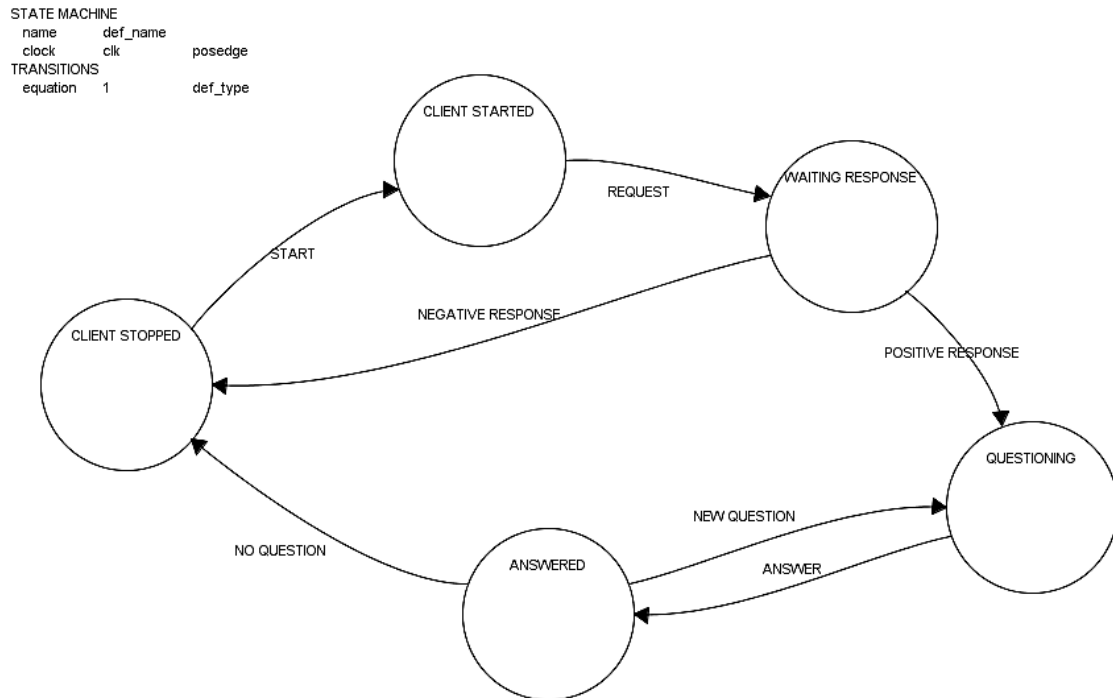
Οι λειτουργίες που εκτελεί ο server είναι η καταχώρηση των στοιχείων σύνδεσης και η εξέλιξη του παιχνιδιού.

Εξέλιξη του παιχνιδιού

Η λειτουργία της εξέλιξης του παιχνιδιού έχει ως εξής:

1. Επιλέγονται τα στοιχεία της σύνδεσης (URL, port) και επιχειρείται σύνδεση στον server.
2. Αν η σύνδεση είναι επιτυχής (δηλαδή ο server αποκρίνεται ότι το παιχνίδι μπορεί να αρχίσει), αναμένει για την υποβολή ερώτησης.
3. Αν ο server αποκριθεί ότι δεν μπορεί να αρχίσει το παιχνίδι τότε ολοκληρώνεται το παιχνίδι και ο client μπορεί να επιχειρήσει μία νέα σύνδεση.
4. Για κάθε ερώτηση που υποβάλλεται στον client και για όσο χρόνο αυτή η ερώτηση παραμένει ενεργή, μπορεί να αποστέλλει την απάντηση του
5. Στην απάντηση του ο server αποκρίνεται. Σε περίπτωση που οι ρυθμίσεις του παιχνιδιού επιτρέπουν, ο χρήστης μπορεί να απαντάει ξανά μετά από εσφαλμένη απάντηση (Εικόνα 4).
6. Όταν ολοκληρώνεται η διαδικασία της ερώτησης τότε, αν δεν έχουν ολοκληρωθεί οι ερωτήσεις, υποβάλλεται η νέα ερώτηση από τον server.
7. Αν ο server ενημερώσει για την ολοκλήρωση του παιχνιδιού τότε ο client μπορεί να επιχειρήσει μία νέα σύνδεση.

Στις παρακάτω εικόνες παρουσιάζεται η λειτουργία του server σε μορφή Finite State Machine διαγράμματα.



Εικόνα 17: Client FSM Διάγραμμα 1

ΚΑΤΑΣΤΑΣΗ	ΜΕΤΑΒΑΣΕΙΣ
CLIENT STOPPED: Ο client είναι εκτός λειτουργίας.	START: Εκκίνηση λειτουργίας του client (επόμενη κατάσταση CLIENT STARTED)
CLIENT STARTED: Ο client έχει ενεργοποιηθεί και αρχικοποιηθεί.	REQUEST: Αίτησης σύνδεσης στον server (επόμενη κατάσταση η WAITING RESPONSE)
WAITING RESPONSE: Ο client αναμένει την απάντηση του server.	POSITIVE RESPONSE: Ο server αποδέχεται την αίτηση του client (επόμενη κατάσταση QUESTIONING) POSITIVE RESPONSE: Ο server δεν αποδέχεται την αίτηση του client (επόμενη κατάσταση CLIENT STOPPED)
QUESTIONING: Έχει υποβληθεί ερώτηση από τον server.	ANSWER: Αποστολή απάντησης για την ενεργή ερώτηση (επόμενη κατάσταση ANSWERED)

ANSWERED: Έχει αποσταλεί απάντηση για την ερώτηση που έχει υποβληθεί.

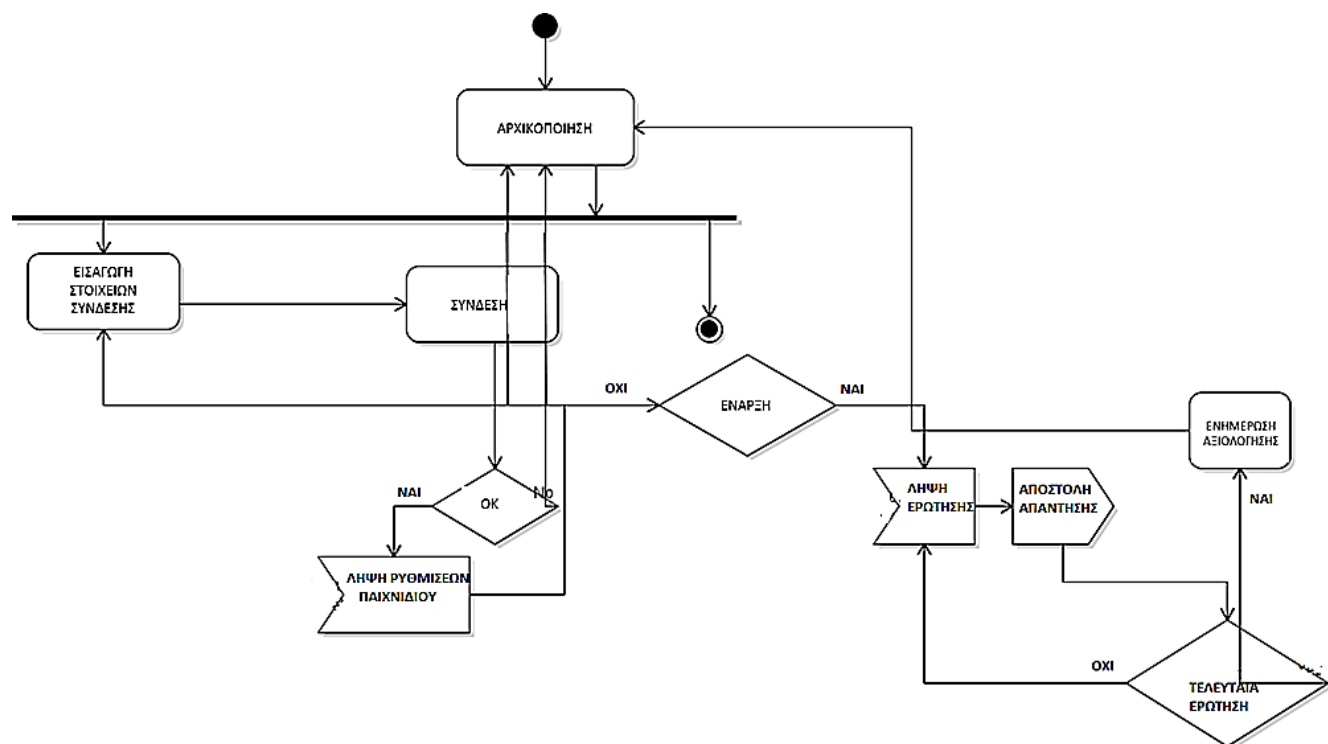
NEXT QUESTION: Υποβάλλεται νέα ερώτηση από τον server (επόμενη κατάσταση QUESTIONING)

NO QUESTION: Δεν υπάρχει άλλη ερώτηση για υποβολή (επόμενη κατάσταση CLIENT STOPPED)

Λειτουργία Καταγραφής Στοιχείων Σύνδεσης

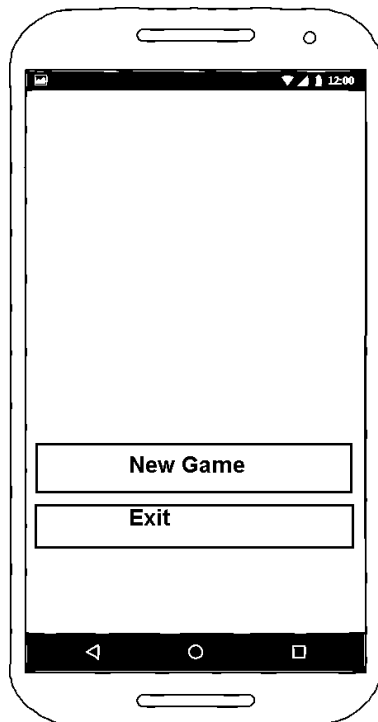
Κατά την λειτουργία αυτή γίνεται προσπάθεια αποκατάστασης σύνδεσης σε δεδομένη IP διεύθυνση, port και με συγκεκριμένο ψευδώνυμο.

Στο παρακάτω διάγραμμα διαδικασιών αποτυπώνεται η συνολική λειτουργία του client.

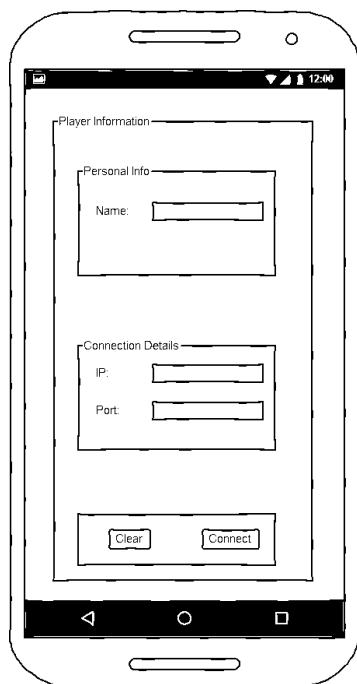


Εικόνα 18: Λειτουργία του Client

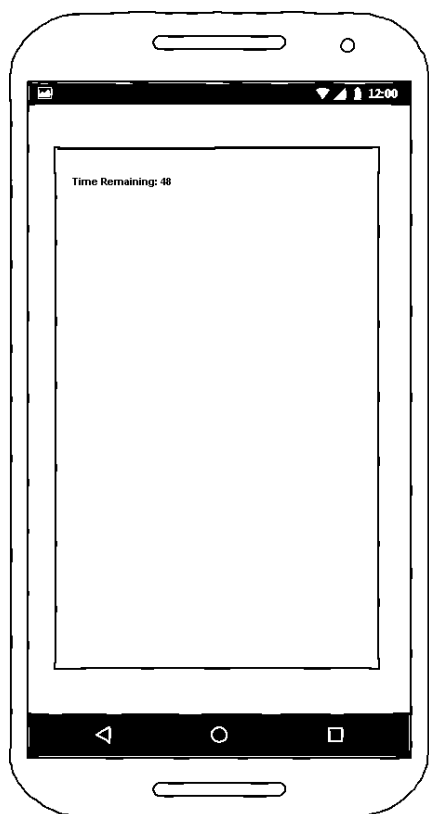
Διεπαφές



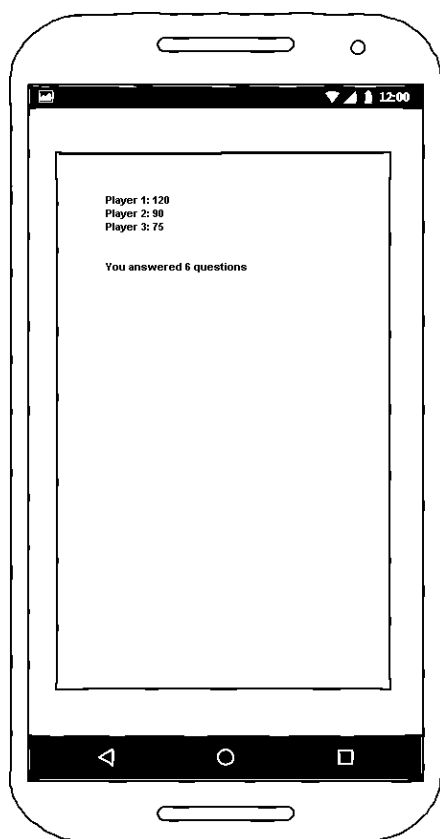
Στην αρχική οθόνη της εφαρμογής ο χρήστης μπορεί να επιλέξει αν θα εκκινήσει νέο παιχνίδι ή αν θα την τερματίσει μέσω την πλήκτρων New Game και Exit αντίστοιχα.



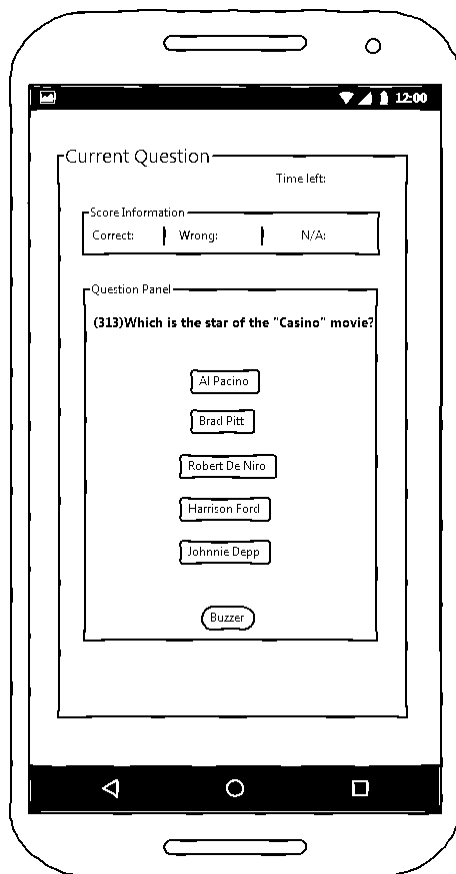
Στην διεπαφή της καταχώρησης των στοιχείων σύνδεσης περιλαμβάνονται textboxes για την καταγραφή της IP διεύθυνσης του server, του port στο οποίο ακούει ο server καθώς και το επιθυμητό ψευδώνυμο που θέλει να χρησιμοποιήσει ο παίκτης. Υπάρχουν επίσης δύο πλήκτρα, ένα για την πραγματοποίηση απόπειρας σύνδεσης (Connect) και ένα για την διαγραφή των καταχωρήσεων που έχει κάνει ο χρήστης.



Στην διεπαφή αυτή προβάλλεται ο χρόνος που απομένει ώστε να ξεκινήσει το παιχνίδι (σε δευτερόλεπτα).



Στην διεπαφή αυτή, προβάλλεται μετά την ολοκλήρωση του παιχνιδιού προβάλλεται η κατάταξη των παικτών που συμμετείχαν καθώς και το σε πόσες ερωτήσεις ο παίκτης απάντησε σωστά.



Η διεπαφή αυτή θα προβάλλεται κατά την εξέλιξη του παιχνιδιού. Θα περιλαμβάνει textboxes για την προβολή:

Το χρόνου που απομένει για την υποβολή απάντησης.

Τον αριθμό των σωστών απαντήσεων που έχει δώσει ο χρήστης.

Τον αριθμό των ερωτήσεων που ο παίκτης απάντησε εσφαλμένα.

Τον αριθμό των ερωτήσεων που ο παίκτης δεν έδωσε απάντηση.

Την τρέχουσα ερώτηση.

Επίσης θα περιλαμβάνεται ένα imageview για την προβολή της αφίσας της εικόνας (αν διατίθεται).

Τέλος θα περιλαμβάνονται πέντε πλήκτρα που θα αντιστοιχούν στις πιθανές απαντήσεις στην ερώτησης καθώς και ένα πλήκτρο το οποίο θα χρησιμοποιείται σε περίπτωση ερωτήσεων τύπου buzzer.

Διαδικτυακή Εφαρμογή

Λειτουργικές Απαιτήσεις

Η web εφαρμογή χρησιμοποιείται για την καταχώρηση, τροποποίηση και διαγραφή εγγραφών από τις οποίες προέρχονται οι ερωτήσεις. Επίσης θα διαμορφώνει ένα κατάλληλο αρχείο το οποίο θα μπορεί ο server να χρησιμοποιήσει για να ενημερώσει την τράπεζα ερωτήσεων του. Οι εγγραφές αυτές είτε προέρχονται από απ' ευθείας καταχωρήσεις του διαχειριστή της είτε από εγγραφές που λαμβάνονται με κατάλληλη διαδικασία από τον δικτυακό τόπο <http://www.omdbapi.com/>.

Μη λειτουργικές Απαιτήσεις

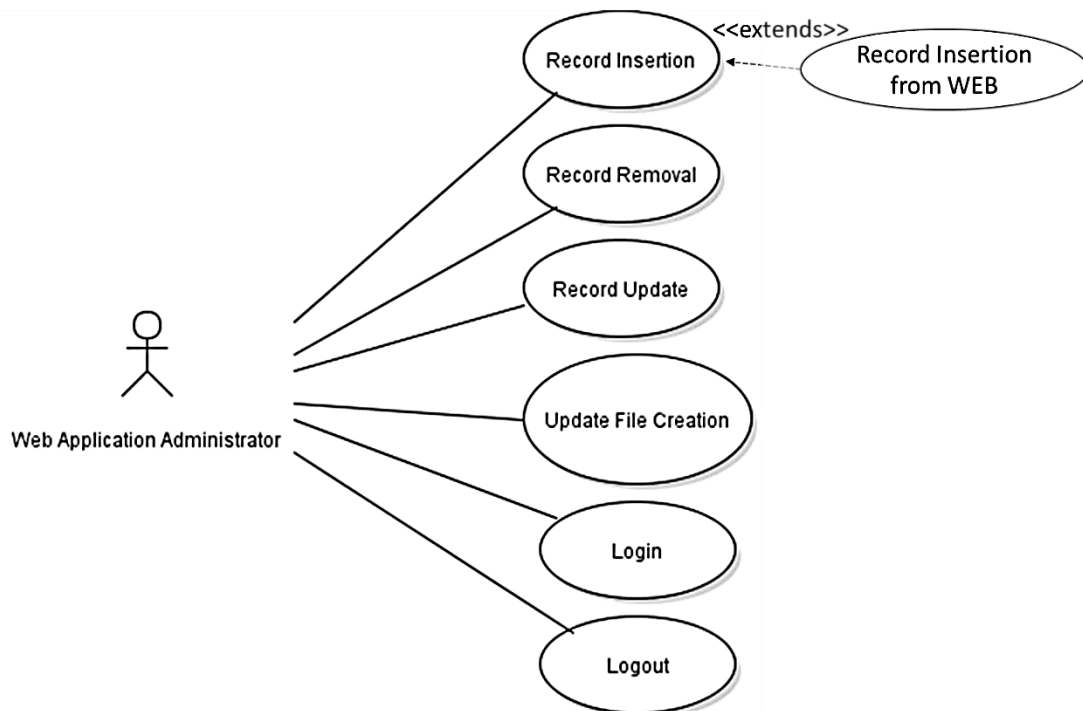
- Η εφαρμογή θα πρέπει να εξασφαλίζει την πρόσβαση σε αυτήν μόνο εξουσιοδοτημένων χειριστών.
- Η εφαρμογή θα πρέπει να είναι διαθέσιμη συνεχώς.

Περιπτώσεις Χρήσης

Οι λειτουργίες που θα πραγματοποιεί ο χρήστης της διαδικτυακής εφαρμογής θα είναι οι εξής:

- Καταχώρηση νέας εγγραφής (Record Insertion)
- Τροποποίηση υπάρχουσας εγγραφής (Record Update)
- Διαγραφή εγγραφής (Record Removal)
- Καταχώρηση εγγραφών από τον δικτυακό τόπο <http://www.omdbapi.com/>.
- Δημιουργία αρχείου ενημέρωσης (Update File Construction)
- Είσοδος χρήστη (Login)
- Έξοδος χρήστη (Logout)

Στο παρακάτω σχήμα φαίνονται οι περιπτώσεις χρήσης σε USE-CASE διάγραμμα.



Εικόνα 19: Web Application Use Case Diagram

Δεδομένα

Η διαδικτυακή εφαρμογή θα υποστηρίζεται από βάση δεδομένων η οποία θα περιλαμβάνει μία οντότητα, την Movie. Η οντότητα αυτή θα έχει την ίδια δομή όπως η αντίστοιχη στην εφαρμογή Server.

Περιγραφή Λειτουργιών

Είσοδος

Η δυνατότητα πραγματοποίησης των υπολοίπων λειτουργιών θα δίνεται μετά από μία διαδικασία ελέγχου συνθηματικών που θα καταχωρεί ο χρήστης. Στην περίπτωση που η διαδικασία αυτή είναι επιτυχής δημιουργείται μία σύνοδος στα πλαίσια της οποίας γίνονται οι δοσοληψίες του χρήστη με την εφαρμογή.

Καταχώρηση

Ο χρήστης καταχωρεί τα στοιχεία της νέας εγγραφής. Διαμορφώνεται κατάλληλο SQL ερώτημα το οποίο και εκτελείται στην βάση δεδομένων.

Καταχώρηση από το WEB

Ο χρήστης καταχωρεί τους τίτλους των ταινιών που θέλει να καταχωρήσει στην βάση δεδομένων. Πραγματοποιείται έλεγχος στον δικτυακό τόπο <http://www.omdbapi.com/> και αν εντοπιστούν οι σχετικές ταινίες καταχωρούνται τα στοιχεία τους

Τροποποίηση

Ο χρήστης επιλέγει την εγγραφή την οποία θέλει να μεταβάλλει. Εκτελείται ένα ερώτημα ανάκτησης των στοιχείων της εγγραφής από την βάση δεδομένων και παρουσιάζονται στον χρήστη. Εκείνος τα μεταβάλλει και στην συνέχεια εκτελεί ένα update SQL ερώτημα στην βάση δεδομένων.

Διαγραφή

Ο χρήστης επιλέγει την εγγραφή την οποία επιθυμεί να διαγράψει και εκτελείται κατάλληλο delete SQL ερώτημα στην βάση δεδομένων.

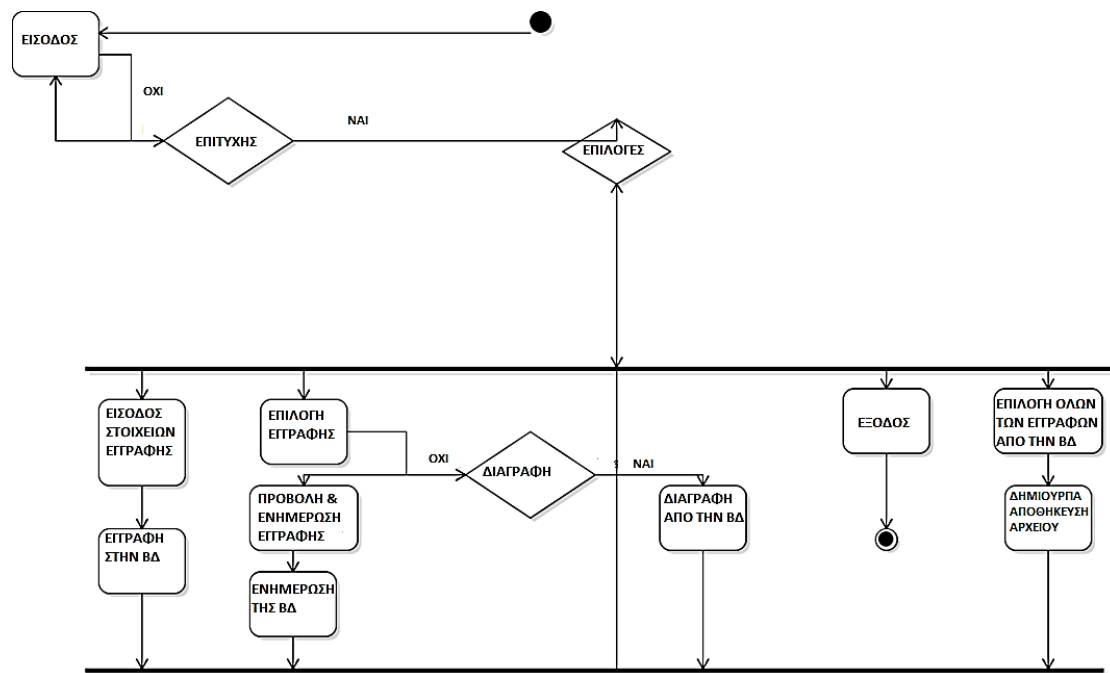
Δημιουργία Αρχείου

Η λειτουργία αυτή ανακτά τα στοιχεία όλων των εγγραφών της βάσης δεδομένων και δημιουργεί ένα αρχείο εγγραφών μεταβλητού μήκους που περιλαμβάνει αυτές.

Έξοδος

Με την λειτουργία αυτή τερματίζεται η σύνοδος του χρήστη.

Στο επόμενο σχήμα φαίνεται η λειτουργία της διαδικτυακής εφαρμογής με την βοήθεια διαγράμματος δραστηριοτήτων.



Εικόνα 20: Διάγραμμα Δραστηριοτήτων της διαδικτυακής εφαρμογής

Διεπαφές

Οι διεπαφές της εφαρμογής θα έχουν ως εξής:

Login

Η διεπαφή για την είσοδο στην εφαρμογή θα περιλαμβάνει δύο texboxes για την καταχώρηση username και password καθώς και ένα πλήκτρο για την ολοκλήρωση της διαδικασίας εισόδου στην εφαρμογή.

Insert Movie

Edit Movie

Delete Movie

Make File

Logout

Στην αρχική οθόνη περιλαμβάνονται σύνδεσμοι προς τις λειτουργίες της διαδικτυακής εφαρμογής.

id	
title	
Year	
Rated	
released	
Genre	
Director	
Writer	
Actors	
plot	
poster	
runtime	
Rating	
Votes	
imdb	
tstamp	

Για την καταχώρηση θα προβάλλεται μία φόρμα στην οποία ο χρήστης θα καταχωρεί τα στοιχεία της εγγραφής σε textboxes και κάνοντας στο πλήκτρο insert θα ολοκληρώνεται η λειτουργία. Το πλήκτρο back επιστρέφει την χρήστη στην αρχική οθόνη.

title

Για την τροποποίηση της εγγραφής θ προβάλλεται μία διεπαφή που θα περιέχει ένα combobox από το οποίο ο χρήστης θα επιλέγει τον τίτλο της ταινίας που θα θέλει να ενημερώσει. Με το πλήκτρο Back θα μπορεί ο χρήστης να επιστρέφει στην αρχική οθόνη. Στην συνέχεια θα προβάλλεται μία διεπαφή παρόμοια με αυτή της καταχώρησης όπου ο στα textboxes θα είναι καταγεγραμμένες οι τρέχουσες τιμές των πεδίων της εγγραφής. Όταν ο χρήστης θα ολοκληρώνει την ενημέρωση τους θα μπορεί να κάνει κλικ ώστε οι μεταβολές να καταγράφονται στην βάση δεδομένων.

title

Για την τροποποίηση της εγγραφής θ προβάλλεται μία διεπαφή που θα περιέχει ένα combobox από το οποίο ο χρήστης θα επιλέγει τον τίτλο της ταινίας που θα θέλει να ενημερώσει. Μετά την επιλογή και με κλικ στο πλήκτρο Delete θα ολοκληρώνεται η διαγραφή από την βάση δεδομένων. Με το πλήκτρο Back θα μπορεί ο χρήστης να επιστρέφει στην αρχική οθόνη.

The diagram shows a rectangular container. Inside the container, at the top left, is a smaller rectangle labeled "TITLES". Below this rectangle, at the bottom of the container, is a blue button with the text "GET MOVIES" in white capital letters.

Ο χρήστης καταχωρεί σε textbox τους τίτλους των ταινιών που τον ενδιαφέρουν και κάνει κλικ στο πλήκτρο GET MOVIES. Όσες από τις ταινίες εντοπιστούν θα καταγραφούν στην βάση δεδομένων.

Την διαδικασία της σχεδίασης διαδέχεται η υλοποίηση. Στην φάση αυτή της ανάπτυξης του συστήματος έγινε η επιλογή των καταλληλότερων τεχνολογιών και εργαλείων για την πληρέστερη υλοποίηση όσων έχουν προδιαγραφεί στην φάση της σχεδίασης. Στην συνέχεια πραγματοποιήθηκε η ανάπτυξη των επί μέρους εφαρμογών και δοκιμάστηκε στην πράξη η λειτουργικότητα της κάθε μίας ξεχωριστά αλλά και των διαδικασιών στις οποίες είναι απαραίτητη η συνεργασία τους.

Κεφάλαιο 4: Υλοποίηση

Περιγραφή

Κατά την φάση της υλοποίησης εφαρμόστηκαν στην πράξη όσο κατεγράφησαν κατά την φάση της σχεδίασης. Η υλοποίηση έγινε σε δύο κύκλους. Κατά τον πρώτο κύκλο υλοποιήθηκαν οι τρεις διαφορετικές εφαρμογές που συνθέτουν το σύστημα. Στο κύκλο αυτό υλοποιήθηκαν και εξετάστηκαν όσες από τις λειτουργίες μπορούν να εκτελεστούν μεμονωμένα. Στον δεύτερο κύκλο υλοποιήθηκαν οι λειτουργίες που για την εκτέλεση τους απαιτείτο η συνεργασία δύο και η επικοινωνία δύο συσκευών. Ο κύκλος αυτός παρουσίασε το μεγαλύτερο ενδιαφέρον και δυσκολία αφού απαραίτητη προϋπόθεση για την επιτυχή υλοποίηση ήταν ο συντονισμός των συμμετεχόντων μερών.

Η ανάπτυξη των εφαρμογών για κινητά τηλέφωνα έγινε σε γλώσσα προγραμματισμού Java χρησιμοποιώντας όλες τις απαραίτητες βιβλιοθήκες. Η βάση δεδομένων στην εφαρμογή του server αναπτύχθηκε σε SQLite. Ως πλατφόρμα ανάπτυξης χρησιμοποιήθηκε το Android Studio για την ανάπτυξη των λειτουργιών, την ανάπτυξη των διεπαφών αλλά και την δοκιμή τους καθώς έχει την δυνατότητα να προσομοιάσει την λειτουργία των εφαρμογών σε emulators μεγάλης ποικιλίας ρυθμίσεων και προδιαγραφών.

Η ανάπτυξη της διαδικτυακής εφαρμογής έγινε στα πρότυπα της αρχιτεκτονικής των τριών επιπέδων. Το χαμηλότερο επίπεδο της υλοποιείται με βάση δεδομένων MySQL. Το επίπεδο λειτουργιών υλοποιείται σε γλώσσα προγραμματισμού php ενώ οι διεπαφές αναπτύχθηκαν με html. Για την διαχείριση της βάσης δεδομένων χρησιμοποιήθηκε το phpmyadmin. Για την ανάπτυξη των λειτουργιών σε php και των διεπαφών σε html χρησιμοποιήθηκε το περιβάλλον ανάπτυξης netbeans.

Server

Βάση Δεδομένων

Αναπτύχθηκε μία απλή βάση δεδομένων SQLite με δύο πίνακες. Στον πρώτο (movies) καταγράφονται τα στοιχεία των ταινιών και στον δεύτερο οι επιδόσεις των παικτών (scores). Τα πεδία του κάθε πίνακα φαίνονται παρακάτω.

Movie

id	Μοναδικό αναγνωριστικό για την ταινία	Varchar(20)
title	Ο τίτλος της ταινίας	Varchar(100)
year	Το έτος παραγωγής της ταινίας.	Varchar(5)
rated	τιμή πεδίου «βαθμολογήθηκε»	Varchar(20)
released	Η ημερομηνία διάθεσης της ταινίας.	Varchar(20)
genre	Το είδος της ταινίας.	Varchar(20)
director	Ο σκηνοθέτης της ταινίας.	Varchar(500)
writer	Τα ονοματεπώνυμα των συγγραφέων της ταινίας.	Varchar(500)
actors	Τα ονοματεπώνυμα των ηθοποιών που πρωταγωνιστούν στην ταινία.	Varchar(500)
plot	Την πλοκή της ταινίας	Varchar(20)
poster	URL της θέσης που είναι αποθηκευμένη το poster της ταινίας.	Varchar(100)
runtime	Η διάρκεια της ταινίας.	Varchar(20)
rating	Η μέση βαθμολογία που έχει αποσπάσει η ταινία	Varchar(20)
votes	Οι αξιολογήσεις που έχει λάβει η ταινία.	Varchar(20)
imdb	Το imdb id της ταινίας	Varchar(20)
tstamp	Μια χρονική σφραγίδα που θα αναφέρεται στην ημερομηνία τελευταίας επεξεργασίας της εγγραφής (ταινίας)	Varchar(20)

Score

user	Το ψευδώνυμο που χρησιμοποίησε ο χρήστης	Varchar(20)
sdate	Η ημερομηνία που σημειώθηκε η επίδοση	Varchar(20)
score	Το ύψος της επίδοσης	Integer

Λειτουργίες

Κάθε μία από τις λειτουργίες του server αποδόθηκαν σε ένα Activity. Στην αρχική οθόνη της εφαρμογής παρέχονται οι επιλογές για το ποιο από τα Activities θα ενεργοποιηθεί.

Προβολή πληροφοριών περί εφαρμογής

Η λειτουργία αυτή περιλαμβάνεται στην κλάση AboutActivity. Σε αυτήν έχουν καταχωρηθεί συγκεκριμένες συμβολοσειρές στα textboxes που έχουν τοποθετηθεί στην διεπαφή. Επίσης το συμβάν κλικ στα πλήκτρα της διεπαφής (πλην του πλήκτρου back) ενεργοποιεί την μέθοδο openUrl. Η μέθοδος αυτή ελέγχει το tag που έχει καταχωρηθεί στο πλήκτρο στο οποίο έγινε κλικ και ανάλογα με την τιμή ανοίγει συγκεκριμένο url. Το συμβάν κλικ στο πλήκτρο back ενεργοποιεί την FirstScreen Activity.

Ρυθμίσεις Παιχνιδιού

Κατά την λειτουργία αυτή παρέχονται στοιχεία διεπαφών στον χρήστη για να καταχωρήσει τις προτιμήσεις του για το επικείμενο παιχνίδι. Για όσες ρυθμίσεις απαιτείται καταχώρηση της μορφής ναι/όχι έχουν προβλεφθεί ToggleButton που κρατούν τις τιμές αυτές ανάλογα με το αν είναι ενεργοποιημένα ή απενεργοποιημένα. Οι υπόλοιπες τιμές καταχωρούνται σε textboxes. Αυτά ανάλογα με το αν πρέπει να πάρουν μόνο αριθμητικές τιμές ή όχι είναι ρυθμισμένα να αποδέχονται κατάλληλους χαρακτήρες. Το πλήκτρο Single Player, με το συμβάν κλικ, ενεργοποιεί την Activity που διαχειρίζεται την έναρξη και την εξέλιξη του παιχνιδιού για έναν παίκτη, το πλήκτρο

Multiplayer ενεργοποιεί την ίδια Activity που διαχειρίζεται την έναρξη και την εξέλιξη του παιχνιδιού για πολλούς παίκτες ενώ το πλήκτρο Back ενεργοποιεί την αρχική οθόνη της εφαρμογής.

Προβολή Μεγαλύτερων Επιδόσεων

Η λειτουργία αυτή δημιουργεί ένα αντικείμενο MyDBHandler προκειμένου να αποκτήσει πρόσβαση στην SQLite βάση δεδομένων. Στην συνέχεια εκτελείται το ερώτημα για την ανάκτηση των 10 εγγραφών από τον πίνακα scores με την μεγαλύτερη τιμή στο πεδίο score. Τα αποτελέσματα προβάλλονται σε ένα GridView. Κάθε γραμμή του GridView μορφοποιείται με βάση το layout GridLine με αποτέλεσμα να περιλαμβάνει τρία textboxes, ένα για κάθε προβαλλόμενο πεδίο (ψευδώνυμο, ημερομηνία, επίδοση). Στην διεπαφή υπάρχει επίσης ένα πλήκτρο Back στο οποίο όταν γίνεται κλικ ο χρήστης επιστρέφει στην αρχική οθόνη.

Ενημέρωση Βάσης Δεδομένων

Για την ενημέρωση της βάσης δεδομένων ενεργοποιείται το Activity UpdateDBActivity όπου δημιουργείται ένα αντικείμενο MyDBHandler. Η διεπαφή περιμένει από τον χρήστη να καταχωρήσει σε textbox το url στο οποίο βρίσκεται το αρχείο ενημέρωσης της βάσης δεδομένων και σε ToggleButton να ορίσει αν επιθυμεί να διαγραφούν οι τρέχουσες εγγραφές. Στο αντικείμενο αυτό τρέχει η μέθοδος updateFromWeb που λαμβάνει σαν παραμέτρους τις παραπάνω καταχωρήσεις του χρήστη στην διεπαφή. Στην μέθοδο αυτή δημιουργείται ένα αντικείμενο Download το οποίο στον Constructor συνδέεται στο url, που λαμβάνει σαν παράμετρο, με το πρωτόκολλο HTTP και διαβάζει γραμμή-γραμμή το αρχείο που βρίσκεται στην τοποθεσία αυτή. Τις εγγραφές του αρχείου τις καταχωρεί σε ArrayList την οποία και επιστρέφει στην μέθοδο updateFromWeb του αντικειμένου. Η λίστα περιλαμβάνει insert sql εντολές. Αυτές εκτελούνται μία – μία στην βάση δεδομένων προσθέτοντας στον πίνακα movie νέες εγγραφές. Η δεύτερη

παράμετρος της μεθόδου `updateFromWeb` προτιμήθηκε να είναι `false` ώστε της καταχώρησης των νέων εγγραφών να μην προηγείται διαγραφή των τρεχουσών. Με την ολοκλήρωση της διαδικασίας εμφανίζεται κατάλληλο μήνυμα στον χρήστη και εμφανίζεται η αρχική οθόνη της εφαρμογής ενεργοποιώντας το Activity First Screen.

Αρχικοποίηση – Διεξαγωγή – Ολοκλήρωση του παιχνιδιού

Η αρχικοποίηση του παιχνιδιού διαδέχεται τη καταχώρηση των ρυθμίσεων του παιχνιδιού. Οι επιλογές του χρήστη στην `ConfigurationActivity` περνάν και καταγράφονται στην `HomeActivity` που διαχειρίζεται και το παιχνίδι. Σε αυτήν δημιουργείται ένα νέο Thread στο οποίο ανοίγει ένα socket στο port που έχει επιλεγεί. Σε αυτό το port αναμένονται συνδέσεις από τους clients. Η διαδικασία αυτή ξεκινάει με κλικ στο πλήκτρο Connection της διεπαφής.

Παράλληλα γίνεται και η αρχικοποίηση των ερωτήσεων με την μέθοδο `initGame`. Η μέθοδος αυτή δημιουργεί έναν πίνακα αντικειμένων `Question` καθώς και ένα αντικείμενο `MyDBHandler`. Από το τελευταίο χρησιμοποιείται η μέθοδος `makeQuestion` η οποία λαμβάνει σαν παράμετρο το θέμα της ερώτησης (τυχαία επιλογή από πίνακα συμβολοσειρών) καθώς και τα φίλτρα που έχουν οριστεί για το περιεχόμενο των ερωτήσεων. Η `makeQuestion` διαμορφώνει το ερώτημα προς την βάση δεδομένων σύμφωνα με τα φίλτρα που δόθηκαν και λαμβάνει το αποτέλεσμα σε μορφή αντικειμένου `Movie..` Ανάλογα με το θέμα της ερώτησης διαμορφώνει την εκφώνηση της και επιλέγει τυχαία το ποια από τις πέντε επιλογές θα είναι η σωστή. Επίσης καταγράφεται η θέση που είναι αποθηκευμένο το poster της ταινίας. Οι εσφαλμένες επιλογές διαμορφώνονται με ανάκτηση εγγραφών για άλλες ταινίες (με τυχαίο τρόπο) και καταγραφή στα αντίστοιχα πεδία του αντικειμένου `Question` επιστρέφεται.

Αφού δημιουργηθεί ο πίνακας των ερωτήσεων διαμορφώνεται κατάλληλα η διεπαφή ώστε να εμφανίζονται οι χρήστες που έχουν συνδεθεί σε `ListView`. Το `ListView` αυτό χρησιμοποιεί κατάλληλο `Adapter` ώστε να

προσαρμόζεται σε κάθε θέση του ένα αντικείμενο Player. Η μορφή κάθε γραμμής του καθορίζεται από το list_item layout και περιλαμβάνει ένα textbox με το ψευδώνυμο του παίκτη.

Κάθε φορά που εντοπίζεται μία νέα σύνδεση ελέγχεται αν έχει καταγραφεί άλλος χρήστης με το επιθυμητό ψευδώνυμο. Σε θετική περίπτωση προσαρτάται στο επιθυμητό ψευδώνυμο ένας αριθμός από το 1 έως το 32 ώστε να προκύψει ψευδώνυμο που δεν χρησιμοποιείται. Στην συνέχεια δημιουργείται ένα αντικείμενο Player στο οποίο αποδίδεται το έγκυρο πλέον ψευδώνυμο. Το αντικείμενο αυτό καταγράφεται σε λίστα. Σε κατάσταση κατά την οποία πρόκειται να παίξουν πολλοί χρήστες, τότε αν η εισερχόμενη σύνδεση ήταν η πρώτη τότε αναμένονται νέες συνδέσεις για 60'' ενώ στον client αποστέλλεται μήνυμα με τις ρυθμίσεις του παιχνιδιού (μία συμβολοσειρά στην οποία οι τιμές των ρυθμίσεων διαχωρίζονται με τους χαρακτήρες <>). Αν μετά την παρέλευση του χρονικού αυτού διαστήματος δεν γίνει αποδοχή νέας σύνδεσης τότε αποστέλλεται μήνυμα στον συνδεδεμένο client, το socket τερματίζεται και ενεργοποιείται η ConfigurationActivity. Αν ο server δεχθεί επιπλέον σύνδεση, επαναλαμβάνεται η ίδια διαδικασία με αυτήν που ακολουθήθηκε με τον πρώτο χρήστη με την διαφορά ότι το μήνυμα των ρυθμίσεων του παιχνιδιού περιλαμβάνει ως χρόνο αναμονής τον εναπομείναντα. Σε περίπτωση που το παιχνίδι έχει ρυθμιστεί για έναν μόνο παίκτη η υποβολή των ερωτήσεων ξεκινάει με την σύνδεση του.

Όταν αρχίζει η διαδικασία του παιχνιδιού και οι clients ενημερώνονται για αυτό με μήνυμα. Παράλληλα γίνονται ορατά τα στοιχεία εκείνα των διεπαφών που είναι απαραίτητα για την παρακολούθηση της διαδικασίας του παιχνιδιού και αποτυπώνουν την τρέχουσα ερώτηση, το πόσες ερωτήσεις έχουν απαντηθεί καθώς και το πόσοι παίκτες παραμένουν συνδεδεμένοι κάθε στιγμή. Σε κάθε κύκλο του παιχνιδιού ανακτάται από τον πίνακα των Question αντικειμένων η αντίστοιχη ερώτηση και ένα μήνυμα που περιλαμβάνει την εκφώνηση, τις πιθανές απαντήσεις, το url της αφίσας της αντίστοιχης ταινίας καθώς και την σωστή απάντηση αποστέλλεται στους συνδεδεμένους clients. Το μήνυμα για να μπορεί να είναι διακριτό από τους clients ότι αφορά νέα ερώτηση ξεκινάει με τους χαρακτήρες NEW_QUESTION::XX (XX ο αύξων αριθμός της ερώτησης). Από την στιγμή αυτή και μέχρι το όριο αναμονής για

απαντήσεις ο server αναμένει για απαντήσεις από τους clients. Οι απαντήσεις τους έχουν την μορφή ANSWER (αριθμός που αντιστοιχεί στην επιλογή του χρήστη). Από την απάντηση ξεχωρίζονται το ψευδώνυμο του παίκτη που απέστειλε την απάντηση και η ίδια η απάντηση. Ελέγχεται για την ορθότητα της και ενημερώνονται όλοι οι clients (μέσα από ένα for loop). Αν στις ρυθμίσεις του παιχνιδιού έχει οριστεί ότι θα φαίνεται ποιος δίνει σωστή ή λάθος απάντηση σε όλους τότε το μήνυμα έχει την μορφή RESULT::NICKNAME, με το RESULT να λαμβάνει τιμές CORRECT ή ERROR. Σε διαφορετική περίπτωση το μήνυμα θα έχει μορφή RESULT::N/A. Για κάθε παίκτη που δίνει ορθή απάντηση προστίθενται στο χαρακτηριστικό score του αντίστοιχου αντικειμένου Player η τιμή που αντιστοιχεί σε κάθε σωστή απάντηση. Με την ολοκλήρωση του χρόνου αναμονή αποστέλλονται σε όλους τους clients τα σκορ όλων των παικτών.

Οι buzz ερωτήσεις αντιμετωπίζονται διαφορετικά από τον server. Όταν δεχθεί BUZZ μήνυμα από κάποιον client, αναστέλλει την λειτουργία του thread μέχρι ο χρήστης να επιλέξει στο ListView απάντησε σωστά. Την χρονική στιγμή αυτή επανενεργοποιείται το thread. Όποιος επιλεγεί από το listview λαμβάνει το σκορ για σωστή απάντηση.

Πριν την αποστολή κάθε ερώτησης γίνεται έλεγχος αν κάποιος από τους clients έχει αποσυνδεθεί. Αν αυτό έχει συμβεί τότε διακόπτεται η σύνδεση με αυτόν και στην θέση του στο ListView καταχωρείται η ένδειξη DISCONNECTED. Επίσης το textbox που φαίνεται πόσοι χρήστες είναι συνδεδεμένοι ενημερώνεται με το νέο πλήθος χρηστών.

Όταν ολοκληρωθεί η διαδικασία και τις τελευταίας ερώτησης, δημιουργείται ένα αντικείμενο MyDBHandler ώστε να καταχωρηθούν με κατάλληλα queries τα σκορ των συμμετεχόντων στην βάση δεδομένων. Παράλληλα αποστέλλονται τα αποτελέσματα στους clients. Τέλος διακόπτονται οι συνδέσεις με τους clients, τερματίζεται το socket και το αντίστοιχο thread και ενεργοποιείται το ConfigurationActivity.

Client

Λειτουργίες

Καταχώρηση Στοιχείων Σύνδεσης – Σύνδεση

Κατά την λειτουργία αυτή ο χρήστης καταχωρεί τα στοιχεία σύνδεσης καθώς και το επιθυμητό nickname και προκειμένου να γίνει απόπειρα σύνδεσης στον server κάνει κλικ στο πλήκτρο Connect. Με το συμβάν αυτό δημιουργείται ένα νέο thread μέσα στο οποίο δημιουργείται ένα νέο socket. Αν η δημιουργία του socket είναι επιτυχής τότε ο χρήστης περνάει στην λειτουργία της εξέλιξης του παιχνιδιού. Σε διαφορετική περίπτωση επιστρέφει σε κατάσταση καταχώρησης στοιχείων σύνδεσης στον server.

Εξέλιξη παιχνιδιού

Αν η απόπειρα σύνδεσης στον server είναι επιτυχής τότε λαμβάνει μήνυμα από τον server που τον ενημερώνει για την διατήρηση του nickname που επέλεξε ή για την μεταβολή του. Λαμβάνει επίσης νέο μήνυμα για τις προδιαγραφές του παιχνιδιού. Αν το παιχνίδι οριστεί ότι είναι για έναν παίκτη τότε με την σύνδεση του λαμβάνει και την πρώτη ερώτηση από τον server. Αν έχει οριστεί για την συμμετοχή περισσότερων παικτών, αναμένει για 60” προκειμένου να ενημερωθεί από τον server με νέο μήνυμα αν θα πραγματοποιηθεί η διαδικασία του παιχνιδιού. Με το πέρας του χρονικού αυτού διαστήματος λαμβάνει μήνυμα από τον server. Αν το μήνυμα τον ενημερώνει για την αδυναμία εκτέλεσης του παιχνιδιού τότε το socket κλείνει, τερματίζεται το thread και ο χρήστης επιστρέφει σε κατάσταση καταχώρησης στοιχείων σύνδεσης στον server. Σε περίπτωση που το παιχνίδι πραγματοποιηθεί διαμορφώνεται η οθόνη ανάλογα με τις προδιαγραφές του παιχνιδιού και αρχίζει να λαμβάνει μηνύματα με ερωτήσεις. Τα μηνύματα αποκωδικοποιούνται ώστε οι ερωτήσεις και οι πιθανές απαντήσεις να

προσαρμοστούν τις κατάλληλες θέσεις στις διεπαφές. Παράλληλα εκτελείται μία ασύγχρονη διαδικασία προκειμένου να ανακτηθεί από το διαδίκτυο και να παρουσιαστεί η αφίσα της ταινίας. Με την λήψη κάθε ερώτησης τρέχει ένα ρολόι που δείχνει το χρόνο που έχει ο χρήστης στην διάθεση του να απαντήσει. Στο τέλος του χρόνου ενημερώνει τον server ότι αναμένει την νέα ερώτηση. Αν ο χρήστης μέσα στον χρόνο αυτό απαντήσει επιλέγοντας να κάνει κλικ στη απάντηση που θεωρεί σωστή, αποστέλλεται ένα μήνυμα στον server με την επιλογή του. Είτε η απάντηση είναι σωστή είτε λάθος τα πλήκτρα των πιθανών απαντήσεων απενεργοποιούνται μέχρι να φθάσει νέα ερώτηση. Απενεργοποιημένα παραμένουν στο ίδιο χρονικό διάστημα και τα πλήκτρα για τις βοήθειες. Καθ' όλη την διάρκεια του διατιθέμενου για απαντήσεις χρόνο λαμβάνονται μηνύματα για το αν κάποιοι από τους άλλους clients δίνουν απαντήσεις. Αν στην ρύθμιση του παιχνιδιού καθορίζεται ότι αναμένονται απαντήσεις μέχρι να δοθεί η σωστή τότε μόλις ληφθεί μήνυμα για σωστή απάντηση από τον server απενεργοποιούνται τα πλήκτρα των απαντήσεων. Αν δοθεί ερώτηση Buzz από τον server (ερωτήσεις με πιθανή απάντηση «...»), τότε είναι ενεργοποιημένο το Buzz πλήκτρο μέχρι να ληφθεί μήνυμα από τον server ότι κάποιος από τους παίκτες έκανε κλικ στο δικό του Buzz πλήκτρο. Όταν ο server στείλει μήνυμα για το τέλος του παιχνιδιού, το socket κλείνει, τερματίζεται το thread. Στο μήνυμα για το τέλος του παιχνιδιού ο server στέλνει και τις επιδόσεις όλων των παικτών. Αυτές εμφανίζονται ταξινομημένες σε νέα οθόνη που ανοίγει. Τέλος, με κλικ σε κατάλληλο πλήκτρο ο χρήστης επιστρέφει στην αρχική οθόνη της εφαρμογής.

Διαδικτυακή Εφαρμογή

Καταχώρηση ταινίας

Σε μία html φόρμα καταχωρούνται τα στοιχεία της ταινίας τα οποία καταχωρούνται από τον χρήστη της εφαρμογής. Με την υποβολή της φόρμας στον server διαμορφώνεται ένα Insert query το οποίο αποστέλλεται τον database server και καταχωρείται η εγγραφή στον πίνακα Movie.

Καταχώρηση ταινιών από το WEB (scrapping ³)

Σε ένα textbox καταχωρούνται οι τίτλοι των ταινιών που είναι επιθυμητό να καταχωρηθούν στην βάση δεδομένων της διαδικτυακής εφαρμογής. Εναλλακτικά μπορεί να επιλέγει ο χρήστης αρχείο κειμένου ώστε να συμπληρώνεται το textbox με τα περιεχόμενα του. Ενεργοποιείται στην συνέχεια μία javascript συνάρτηση η οποία προσαρτά σε κατάλληλα διαμορφωμένο url έναν από τους καταγεγραμμένους τίτλους κάθε φορά προκειμένου να ανακτηθούν τα στοιχεία της ταινίας με την μέθοδο web scrapping. Το url χρησιμοποιείται για ajax κλήση σε διαδικασία του <http://www.omdbapi.com/> η οποία επιστρέφει τα στοιχεία της ταινίας σε json μορφή. Τα ληφθέντα στοιχεία χρησιμοποιούνται για να διαμορφωθεί ένα url προκειμένου να τρέξει η διαδικασία καταχώρησης της ταινίας στην βάση δεδομένων (addToDb.php). Με την διαδικασία αυτή διαμορφώνεται ένα Insert query το οποίο αποστέλλεται τον database server και καταχωρείται η εγγραφή στον πίνακα Movie.

```
{ "Title": "Braveheart", "Year": "1995", "Rated": "R", "Released": "24 May 1995", "Runtime": "178 min", "Genre": "Biography, Drama, History", "Director": "Mel Gibson", "Writer": "Randall Wallace", "Actors": "James Robinson, Sean Lawlor, Sandy Nelson, James Cosmo", "Plot": "When his secret bride is executed for assaulting an English soldier who tried to rape her, William Wallace begins a revolt and leads Scottish warriors against the cruel English tyrant who rules Scotland with an iron fist.", "Language": "English, French, Latin, Scottish Gaelic", "Country": "USA", "Awards": "Won 5 Oscars. Another 27 wins & 23 nominations.", "Poster": "http://ia.media-imdb.com/images/M/MV5BNjA4ODYxMDU3Nl5BMl5BanBnXkFtZTcwMzkzMzk3MTk3OA@@._V1_SX300.jpg", "Metascore": "68", "imdbRating": "8.4", "imdbVotes": "663,555", "imdbID": "tt0112573", "Type": "movie", "Response": "True" }
```

Εικόνα 21: Μορφή ανακτηθέντων στοιχείων από το ombdapi

Τροποποίηση ταινίας

³ Διαδικασία κατά την οποία επιδιώκεται η ανάκτηση διαδικτυακού περιεχομένου σε μορφή τέτοια που να καθίσταται εφικτή η επεξεργασία του.

Σε μία html φόρμα παρουσιάζονται σε combobox οι τίτλοι και οι κωδικοί των ταινιών που ανακτώνται με κατάλληλο query στην βάση δεδομένων. Από αυτά επιλέγεται ένα και με την υποβολή της φόρμας δημιουργείται query για την ανάκτηση της εγγραφής με τον συγκεκριμένο κωδικό. Οι τιμές των πεδίων της εγγραφής προβάλλονται σε textboxes σε html φόρμα στα οποία ο χρήστης μπορεί να μεταβάλλει τις τιμές τους. Με την υποβολή της φόρμας αυτής δημιουργείται ένα update ερώτημα με το οποίο αλλάζουν οι αντίστοιχες τιμές στην βάση δεδομένων.

Διαγραφή Ταινίας

Σε μία html φόρμα παρουσιάζονται σε combobox οι τίτλοι και οι κωδικοί των ταινιών που ανακτώνται με κατάλληλο query στην βάση δεδομένων. Από αυτά επιλέγεται ένα και με την υποβολή της φόρμας δημιουργείται query για την διαγραφή της εγγραφής με τον συγκεκριμένο κωδικό.

Είσοδος

Σε html φόρμα ο χρήστης πληκτρολογεί username και password. Αν ο συνδυασμός αυτός είναι ορθός, δημιουργείται μία session μεταβλητή που περιλαμβάνει το username του χρήστη. Η μεταβλητή αυτή χρησιμοποιείται για να ελέγχεται αν έχει δημιουργηθεί ώστε σε θετική περίπτωση να είναι διαθέσιμες οι λειτουργίες της διαδικτυακής εφαρμογής.

Έξοδος

Κατά την λειτουργία της εξόδου καταστρέφεται η session μεταβλητή.

Μετρικές Κώδικα

Για τη διαχείριση των μετρικών του κώδικα, επιλέξαμε τη χρήση του opensource εργαλείου cloc, το οποίο μετράει τις κενές, τις σχολιασθείσες και τις πραγματικές γραμμές κώδικα για ένα δοθέν φάκελο.

Διαδικτυακή εφαρμογή

Παρακάτω βλέπουμε τις μετρικές της διαδικτυακής εφαρμογής, η οποία περιέχει 15 διαφορετικά αρχεία κειμένου, από τα οποία τα 14 είναι αυτά που τελικά προσμετρώνται στον κώδικα, καθώς το τελευταίο είναι το αρχείο με τα sql insert queries (.dat format), οπότε και αγνοείται αυτόματα από το εργαλείο. Βλέπουμε, λοιπόν, ότι στα 13 PHP αρχεία έχουμε συνολικά 92 κενές γραμμές, 65 σε σχόλια και 1384 πραγματικού κώδικα. Αντίστοιχα για το sql αρχείο, έχουμε 7 κενές γραμμές, 18 σε σχόλια και 20 πραγματικές.

```
C:\Thodoris\docs\personal\thesis\Android implementation
λ cloc webserver\
    15 text files.
    15 unique files.
    3 files ignored.

http://cloc.sourceforge.net v 1.64 T=0.27 s (51.0 files/s, 5782.8 lines/s)
-----
Language          files      blank      comment      code
-----
PHP                13         92         65         1384
SQL                1          7         18          20
-----
SUM:              14        99         83         1404
-----
```

Client

Αναφορικά με τον client, όπως φαίνεται παρακάτω, αγνοήθηκαν αρκετά αρχεία κι αυτό, γιατί πρόκειται για conf αρχεία παραγόμενα ευθύς εξ αρχής από το Android Studio (π.χ. properties files). Αυτό στο οποίο πρέπει να δώσουμε έμφαση, λοιπόν, είναι οι γραμμές που αφορούν την XML και τη Java, καθώς είναι οι δύο κύριες τεχνολογίες συγγραφής native εφαρμογών Android: οι συνολικά 621 γραμμές XML κώδικα εξηγούνται από το γεγονός υποστήριξης πολλαπλών οθονών, από τη δημιουργία/χρήση custom ui components, από το αρχείο αποθήκευσης των string μεταβλητών κλπ. Αντίστοιχα, για τη Java, έχουμε 1079 γραμμές πραγματικού κώδικα σε 5 μόλις αρχεία, καθώς η πιο βαριά λειτουργικότητα του client είναι η επικοινωνία του με τον server. Τα υπόλοιπα αρχεία από εκεί κι έπειτα είναι σχετικά με το Android Studio και το configuration του Gradle, οπότε δεν θα μας απασχολήσουν.

```

C:\Thodoris\docs\personal\thesis\Android implementation\BuzzGame
λ cloc app --exclude-dir=build
    27 text files.
    27 unique files.
    28 files ignored.

http://cloc.sourceforge.net v 1.64 T=0.12 s (207.5 files/s, 17604.7 lines/s)
-----
Language             files            blank           comment           code
-----
Java                  6                149              100             1079
XML                   17               110               19              621
Groovy                1                 2                 0               24
Prolog                1                 2                 0               15
-----
SUM:                  25              263              119             1739
-----

```

Server

Τέλος, ο server περιέχει 2080 καθαρές γραμμές κώδικα σε 14 διαφορετικά αρχεία, αναφορικά με το κομμάτι της Java, ενώ αρκετές εδώ, παρουσιάζονται και οι γραμμές τις XML, 1216 στον αριθμό, για παρόμοιους με τους προαναφερθέντες λόγους.

```

C:\Thodoris\docs\personal\thesis\Android implementation\iBuzzServer
λ cloc app --exclude-dir=build
    41 text files.
    41 unique files.
    29 files ignored.

http://cloc.sourceforge.net v 1.64 T=0.14 s (288.0 files/s, 30128.5 lines/s)
-----
Language             files            blank           comment           code
-----
Java                  14               292              205             2080
XML                   23               225               19             1216
Groovy                1                 2                 0               24
Prolog                1                 2                 0               15
-----
SUM:                  39              521              224             3335
-----

```

Συνεχίζοντας, λοιπόν, ο καθαρός κώδικας που τελικά αναπτύχθηκε από εμάς, ανέρχεται περίπου στις 6500 γραμμές.

Δοκιμές

Η ανάπτυξη του συστήματος ολοκληρώθηκε με την δοκιμή της ορθής του λειτουργίας. Οι δοκιμές αφορούσαν την λειτουργία μίας εκάστης εφαρμογής αλλά και την συνεργατική λειτουργία τους στα πλαίσια της ένταξης τους στα πλαίσια του αυτού συστήματος. Για τον σκοπό αυτό εγκαταστάθηκε η διαδικτυακή εφαρμογή στην υποδομή ενός παρόχου δωρεάν υπηρεσιών

φιλοξενίας δικτυακών τόπων. Ο πάροχος διαθέτει αποθηκευτικό χώρο επαρκή για την εγκατάσταση και εύρυθμη λειτουργία της εφαρμογής, δυνατότητα ανάπτυξης και απομακρυσμένης διαχείρισης MySql βάσης δεδομένων, δυνατότητα απομακρυσμένης διαχείρισης των αποθηκευμένων αρχείων. Παράλληλα η εφαρμογή του server εγκαταστάθηκε σε έξυπνη κινητή συσκευή ενώ η εφαρμογή του client εγκαταστάθηκε σε ένα πλήθος από emulators. Τόσο η έξυπνη κινητή συσκευή όσο και οι emulators κατά την διάρκεια των δοκιμών ήταν σε κοινό ασύρματο τοπικό δίκτυο. Κατά την διάρκεια της δοκιμής ορθής λειτουργίας του συστήματος εξετάστηκαν τα εξής:

- Η διαχείριση εκ μέρους του server των πολλαπλών ταυτόχρονων συνδέσεων των clients: Ο server ετίθετο σε λειτουργία στην έξυπνη κινητή συσκευή και ένας αριθμός από clients επιχειρούσε την αποκατάσταση σύνδεσης με αυτόν.
- Η ανταλλαγή των μηνυμάτων μεταξύ client και server κατά την διάρκεια διεξαγωγής του παιχνιδιού: Ο server εκκινούσε στην έξυπνη συσκευή μία διαδικασία νέου παιχνιδιού. Ένας αριθμός από clients συμμετείχε στο παιχνίδι και καταγράφονταν η πραγματική συμπεριφορά των εφαρμογών σε σχέση με την προσδοκώμενη.
- Η ομαλή έξοδος του συστήματος από την κατάσταση διεξαγωγής του παιχνιδιού: Με την ολοκλήρωση της κάθε μίας διαδικασίας παιχνιδιού ελέγχεται η ομαλή μετάβαση από την κατάσταση διεξαγωγής παιχνιδιού σε κατάσταση αναμονής.
- Η διαχείριση σφαλμάτων: Δοκιμάστηκε και στις τρεις εφαρμογές η μη ορθή χρήση των λειτουργιών τους από τους πιθανούς χρήστες και ελέγχθηκαν οι συμπεριφορές των εφαρμογών. Με τον τρόπο αυτό επιδιώχθηκε η ομαλή αντιμετώπιση κάθε μη συμβατικής λειτουργίας.
- Η εμφάνιση μηνυμάτων προς τον τελικό χρήστη: Κατά τις παραπάνω αναφερόμενες δοκιμές ελέγχεται ταυτόχρονα και η καταλληλότητα των μηνυμάτων που εμφάνιζαν οι εφαρμογές στις διεπαφές τους.

- Οι λειτουργίες διαχείρισης ερωτήσεων στην διαδικτυακή εφαρμογή: Στην διαδικτυακή εφαρμογή ελέγχθηκαν η εξουσιοδοτημένη πρόσβαση στις λειτουργίες της. Επίσης ελέγχθηκε το κατά πόσο λειτουργούσαν οι λειτουργίες της καταχώρησης, τροποποίησης και διαγραφής ερωτήσεων. Τέλος ελέγχθηκε η συμβατότητα των αρχείων ενημέρωσης της βάσης δεδομένων της εφαρμογής του server που παράγει η διαδικτυακή εφαρμογή.
- Η επικοινωνία της εφαρμογής του server με την διαδικτυακή εφαρμογή: Ελέγχθηκε η ομαλή ενημέρωση της βάσης δεδομένων από αρχείο που έχει δημιουργηθεί με την διαδικτυακή εφαρμογή. Ελέγχθηκε τόσο η καθολική αντικατάσταση της βάσης δεδομένων όσο και η πρόσθεση εγγραφών στην ήδη υπάρχουσα.
- Η δυνατότητα ανάκτησης περιεχομένου από την διαδικτυακή εφαρμογή με την μέθοδο scrapping: Ελέγχθηκε η ορθή ενημέρωση της βάσης δεδομένων από εγγραφές που ανακτήθηκαν με την μέθοδο scrapping.

Ασφάλεια

Εφαρμογές Server – Client

Προκειμένου να λειτουργήσουν αποδοτικά τόσο η εφαρμογή του server όσο και η εφαρμογή του client χρειάζεται να τους παραχωρηθούν από το λειτουργικό σύστημα τα ακόλουθα δικαιώματα:

- android.permission.INTERNET: Επιτρέπει την πρόσβαση της εφαρμογής στο διαδίκτυο κι εξυπηρετεί την ανάκτηση της πιο πρόσφατης έκδοσης της βάσης των ταινιών
- android.permission.WRITE_EXTERNAL_STORAGE: Επιτρέπει στην εφαρμογή αν κάνει εγγραφή στην εξωτερική αποθηκευτική διάταξη κι εξυπηρετεί τη διαχείριση του γενικού πίνακα με τις καλύτερες βαθμολογίες

- android.permission.ACCESS_WIFI_STATE: Επιτρέπει στην εφαρμογή να έχει πρόσβαση σε πληροφορίες σχετικά με τα διαθέσιμα Wi Fi δίκτυα κι εξυπηρετεί τη σύνδεση της συσκευής σ' αυτό που θα επιλέξει ο χρήστης.
- android.permission.CHANGE_WIFI_STATE: Επιτρέπει στην εφαρμογή αν αλλάξει τη κατάσταση σύνδεσης της σε Wi Fi δίκτυο κι εξυπηρετεί την αλλαγή του δικτύου στο οποίο είναι συνδεδεμένη η συσκευή.
- android.permission.CHANGE_NETWORK_STATE: Επιτρέπει στην εφαρμογή να έχει πρόσβαση σε πληροφορίες σχετικά με τα διαθέσιμα δίκτυα κι εξυπηρετεί την ενεργοποίηση του wifi μιας συσκευής, ώστε να συνδεθεί σε κάποιο δίκτυο (πχ ο client στο δίκτυο του server)
- android.permission.ACCESS_NETWORK_STATE: Επιτρέπει στην εφαρμογή να αποκτά τον έλεγχο της κατάστασης σύνδεσης της σε δίκτυο κι εξυπηρετεί τη σύνδεση σε ένα δίκτυο

Για την λήψη αρχείου ενημέρωσης της βάσης δεδομένων ο server συνδέεται στον διακομιστή που ορίζει ο χρήστης και ο οποίος προφανώς τυγχάνει της εμπιστοσύνης του.

Διαδικτυακή Εφαρμογή

Εξουσιοδοτημένη πρόσβαση: Η πρόσβαση στις λειτουργίες γίνεται μόνο μετά την επιτυχή ολοκλήρωση διαδικασίας πιστοποίησης του χρήστη (Login). Με την επιτυχή ολοκλήρωση δημιουργούνται κατάλληλες (session) μεταβλητές συνόδου των οποίων οι τιμές ελέγχονται κάθε φορά που ο χρήστης επιχειρεί να εκτελέσει κάποια από τις λειτουργίες της διαδικτυακής εφαρμογής.

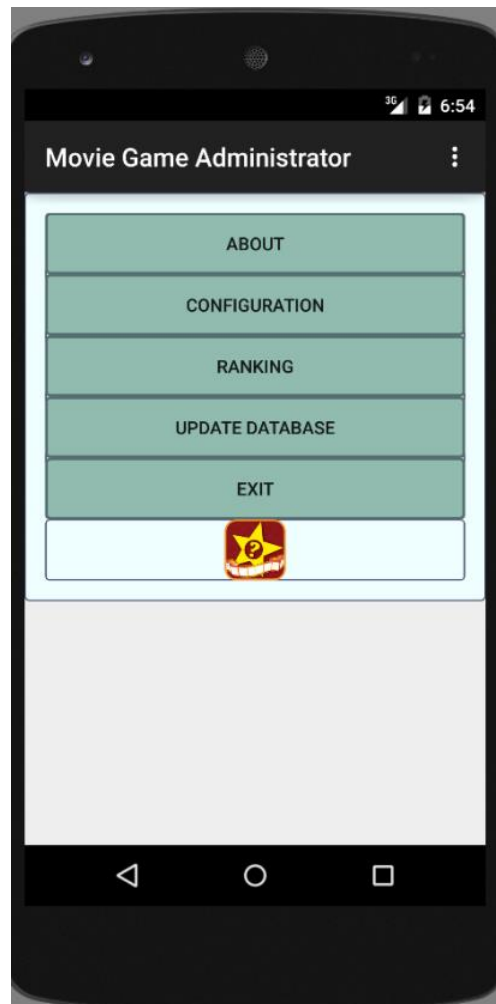
SQL Injection: Κάθε παράμετρος που χρειάζεται να προσαρτηθεί σε εντολές – ερωτήματα στην βάση δεδομένων, φιλτράρεται ώστε να αποκλειστεί το ενδεχόμενο να εκτελέσει κακόβουλος χρήστης επιτυχώς SQL injection.

Χρήση

Server

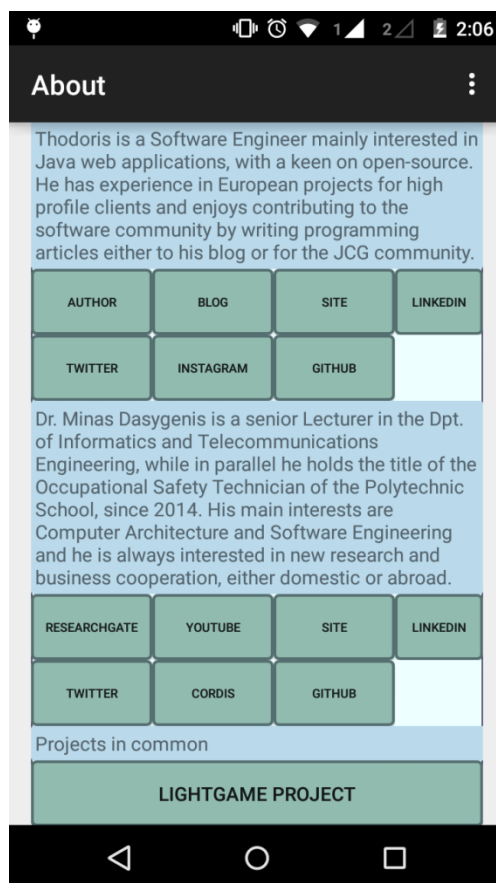
Επιλογή Λειτουργίας

Στην αρχική οθόνη της εφαρμογής server ο χρήστης μπορεί να κάνει κλικ στο πλήκτρο που αντιστοιχεί στην επιθυμητή λειτουργία.



Εικόνα 22: Server - Αρχική οθόνη

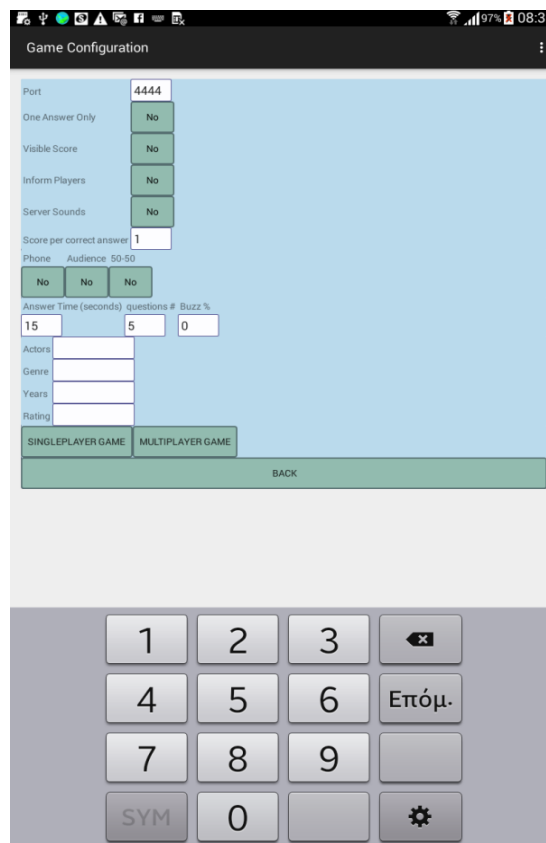
Με κλικ στο πλήκτρο About εμφανίζονται πληροφορίες σχετικές με την ομάδα ανάπτυξης της εφαρμογής καθώς και σύνδεσμοι σε τοποθεσίες που περιέχουν περισσότερες πληροφορίες για αυτή.



Εικόνα 23: Server - Πληροφορίες Ομάδας Ανάπτυξης

Με κλικ στο πλήκτρο Configuration ο χρήστης οδηγείται στην οθόνη καταχώρησης των προδιαγραφών του παιχνιδιού. Εκεί μπορεί αν επιλέξει αν θα αναμένονται απαντήσεις μέχρι να δοθεί η πρώτη σωστή (ToggleButton), το port στο οποίο θα αναμένει για συνδέσεις ο server (Textbox), το αν το σκορ θα είναι ορατό καθ' όλη την διάρκεια του παιχνιδιού (ToggleButton), το αν θα ενημερώνονται οι παίκτες για το πώς απαντάνε οι υπόλοιποι παίκτες (ToggleButton), τους βαθμούς που θα δίνονται για κάθε σωστή απάντηση, το αν θα υπάρχουν διαθέσιμοι ήχοι (ToggleButton), το αν και ποιες βοήθειες θα είναι διαθέσιμες (ToggleButton), το χρόνο που θα είναι διαθέσιμος για απάντηση σε κάθε ερώτηση (Textbox), το πλήθος των ερωτήσεων (Textbox)

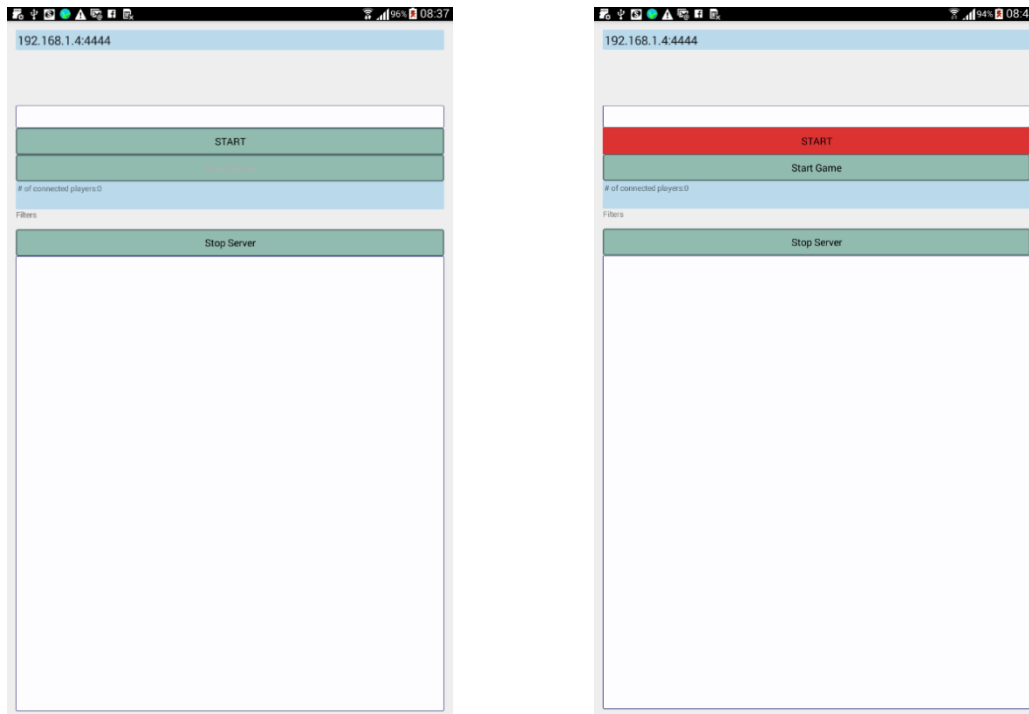
καθώς και το ποσοστό των ερωτήσεων buzz (Textbox). Τα πεδία που γράφονται σε textboxes είναι υποχρεωτικά πλην του ποσοστού buzz ερωτήσεων που λαμβάνει εξ ορισμού την τιμή μηδέν. Με κλικ στο πλήκτρο Next ο χρήστης μεταβαίνει στην οθόνη εκκίνησης του server και με το Back στην αρχική οθόνη της εφαρμογής.



Εικόνα 24: Server - Ρυθμίσεις παιχνιδιού

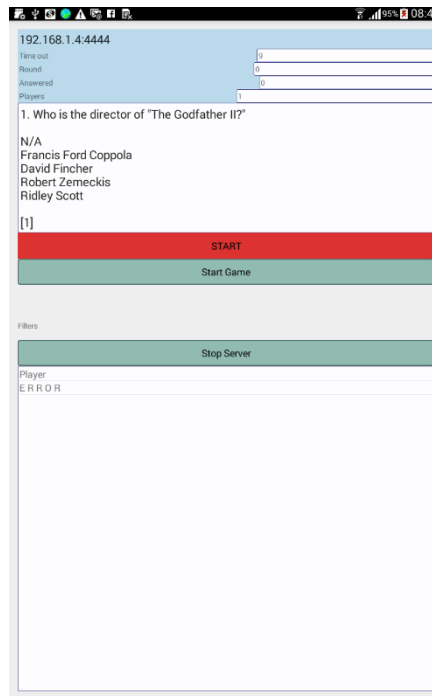
Εκκίνηση Server – Διεξαγωγή Παιχνιδιού

Στην οθόνη της εκκίνησης του server ο χρήστης χρειάζεται να κάνει κλικ στο πλήκτρο Start προκειμένου η εφαρμογή να αναμένει για συνδέσεις από τον client. Με το κλικ του χρήστη το πλήκτρο γίνεται κόκκινο. Κάθε φορά που συνδέεται κάποιος νέος παίκτης εμφανίζεται το ψευδώνυμο του καθώς και ενημερώνεται και το textbox που περιλαμβάνει τον συνολικό αριθμό των συνδεδεμένων clients.



Εικόνα 25: Server – Εκκίνηση

Όταν ο χρόνος αναμονής των συνδέσεων παρέλθει και έχουν συνδεθεί περισσότεροι του ενός παίκτες τότε εμφανίζεται εκ νέου η οθόνη καταχώρησης των στοιχείων του παιχνιδιού. Σε διαφορετική περίπτωση διαμορφώνεται μία διαδραστική λίστα με τα ονόματα των παικτών η οποία θα χρησιμοποιείται για την απόδοση σκορ σε buzz απαντήσεις. Επίσης εμφανίζονται στην οθόνη textboxes – μετρητές για την τρέχουσα ερώτηση, το διατιθέμενο χρόνο για απάντησης, το πόσοι χρήστες είναι συνδεδεμένοι και το πόσες ερωτήσεις έχουν απαντηθεί σωστά. Κάθε ερώτηση φαίνεται στην οθόνη (εκφώνηση, πιθανές απαντήσεις, σωστή απάντηση). Για κάθε παίκτη που για οποιονδήποτε λόγο αποσυνδέεται, στην αντίστοιχη θέση της διαδραστικής λίστας εμφανίζεται η ένδειξη Disconnected και μειώνεται και η ένδειξη του πλήθους των συνδεδεμένων χρηστών. Όταν ολοκληρωθεί το παιχνίδι η αποσυνδεθούν όλοι οι χρήστες τότε εμφανίζεται η οθόνη των ρυθμίσεων του παιχνιδιού.



Εικόνα 26: Server - Διεξαγωγή παιχνιδιού

Προβολή Μεγαλύτερων Επιδόσεων

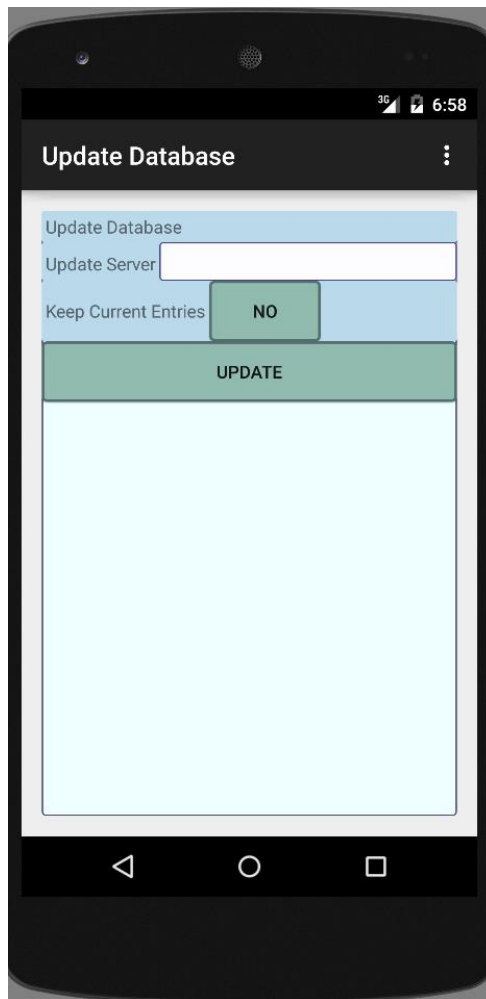
Στην αρχική οθόνη, με κλικ στο πλήκτρο High Scores ο χρήστης οδηγείται στην οθόνη στην οποία προβάλλεται πίνακας με τις 10 καλύτερες επιδόσεις. Με κλικ στο πλήκτρο Back επιστρέφει στην αρχική οθόνη.

User	Date/Time	Score
user7	02/01/2015 00:00:00	559
user8	01/03/2015 00:00:00	209
user8	01/11/2015 00:01:00	190
user8	01/03/2015 00:00:00	167
user6	01/11/2015 00:01:00	156
user8	01/02/2015 00:00:00	119
user8	01/01/2015 00:00:00	109
user8	03/01/2015 00:00:00	106
user7	11/11/2015 00:00:00	103
user8	11/11/2015 00:00:00	102

Εικόνα 27: Server - Καλύτερες επιδόσεις

Ενημέρωση βάσης δεδομένων από το διαδίκτυο

Στην αρχική οθόνη με κλικ στο πλήκτρο Update Database ο χρήστης οδηγείται σε οθόνη όπου πρέπει να καταχωρήσει σε textbox την τοποθεσία από την οποία θα απαντηθεί το αρχείο της ενημέρωσης και να επιλέξει αν επιθυμεί να διαγραφούν οι τρέχουσες εγγραφές. Κάνοντας κλικ στο πλήκτρο Update ολοκληρώνεται η ενημέρωση και εμφανίζεται σχετικό μήνυμα.

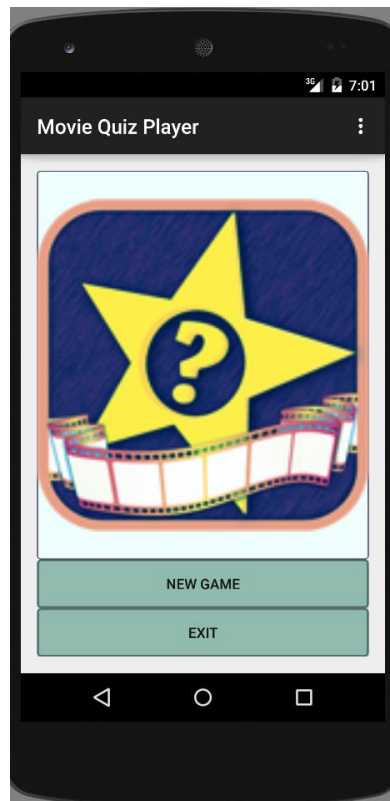


Εικόνα 28: Server - Ενημέρωση Βάσης Δεδομένων

Client

Αρχική Οθόνη Εφαρμογής

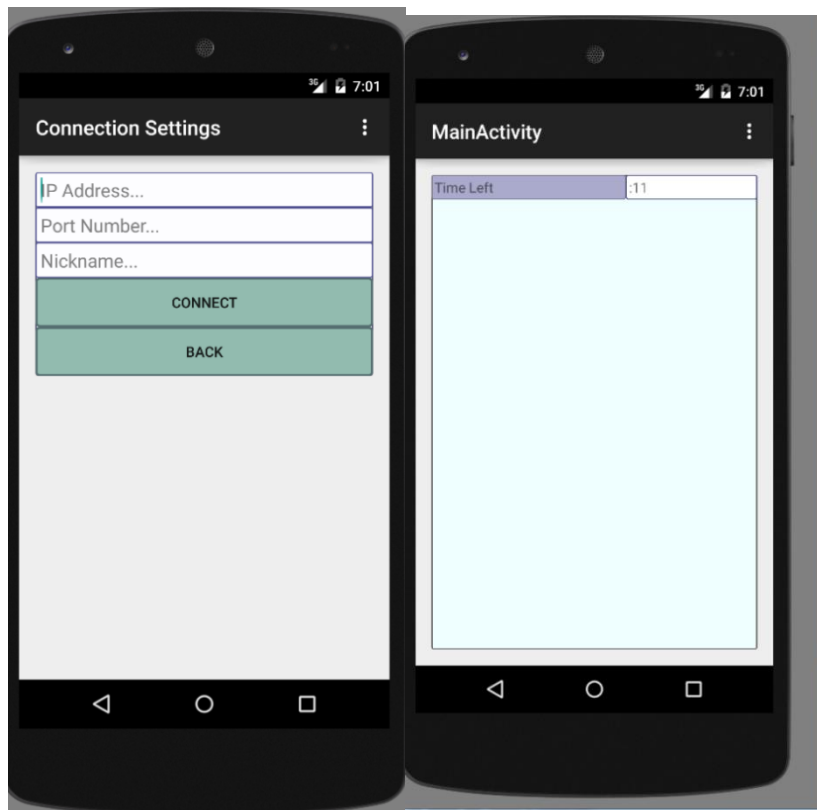
Στην αρχική οθόνη της εφαρμογής ο χρήστης μπορεί να επιλέξει να εκκινήσει ένα νέο παιχνίδι ή να κλείσει την εφαρμογή



Εικόνα 29: Client - Αρχική Οθόνη

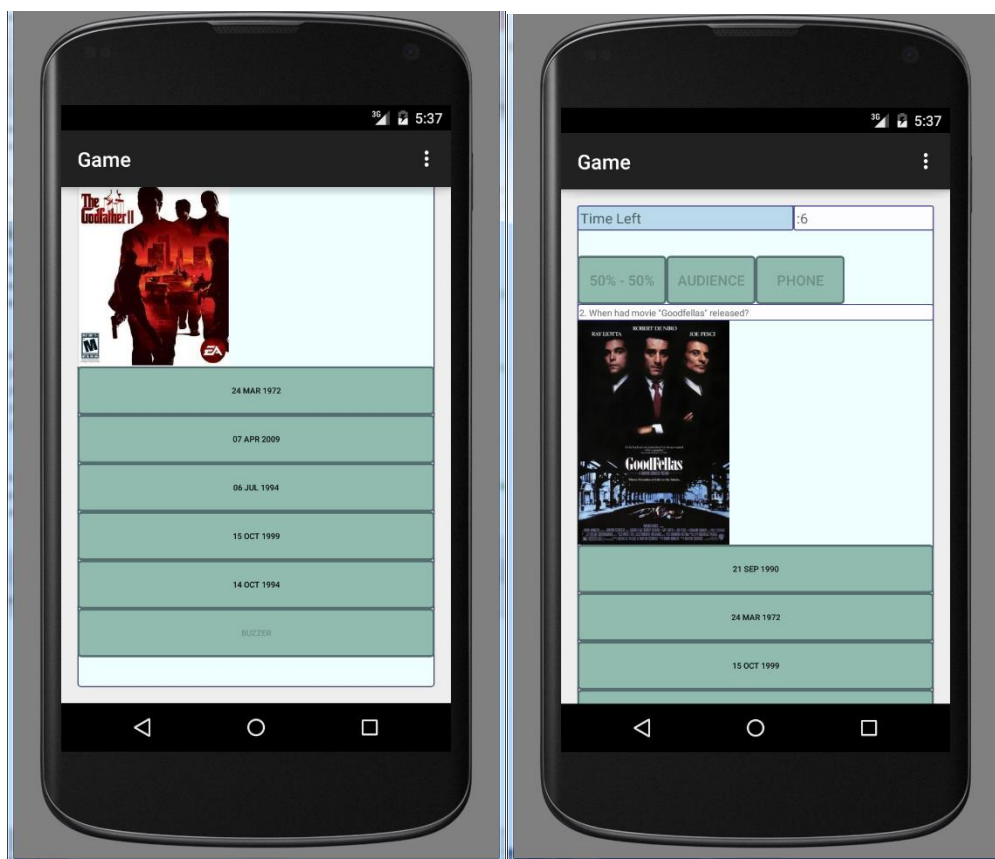
Εκκίνηση Παιχνιδιού

Για την εκκίνηση του παιχνιδιού εμφανίζεται οθόνη όπου ο χρήστης πρέπει να καταχωρήσει την IP διεύθυνση του server, την port που δέχεται συνδέσεις καθώς και το επιθυμητό ψευδώνυμο. Κάνοντας κλικ στο πλήκτρο σύνδεση και αν η σύνδεση στο server είναι επιτυχής, εμφανίζεται χρονόμετρο που ενημερώνει τον χρήστη για τον χρόνο που απομένει για την έναρξη του παιχνιδιού.



Εικόνα 30:Client έναρξη παιχνιδιού

Όταν ο χρόνος αυτό παρέλθει και έχουν συνδεθεί στον server περισσότεροι του ενός clients τότε διαμορφώνεται η οθόνη ανάλογα με τις προδιαγραφές του παιχνιδιού. Σε κάθε ερώτηση εμφανίζεται η εκφώνηση οι πιθανές απαντήσεις σε πλήκτρα τα οποία μπορεί να κάνει κλικ ο χρήστης, ο διατιθέμενος χρόνος για απάντηση, οι βοήθειες (αν διατίθενται), οι επιδόσεις (αν έτσι έχει ρυθμιστεί το παιχνίδι). Με την επιλογή της απάντησης ο χρήστης ενημερώνεται αν έχει απαντήσει σωστά στην ερώτηση. Επίσης ενημερώνεται για τις απαντήσεις των άλλων παικτών. Όταν δεν είναι επιτρεπτό στον παίκτη να απαντήσει τα πλήκτρα των πιθανών απαντήσεων απενεργοποιούνται. Με την ολοκλήρωση του παιχνιδιού εμφανίζεται οθόνη με τις τελικές επιδόσεις των συμμετεχόντων. Στην οθόνη αυτή με κλικ στο πλήκτρο Finish ο χρήστης επιστρέφει στην αρχική οθόνη.

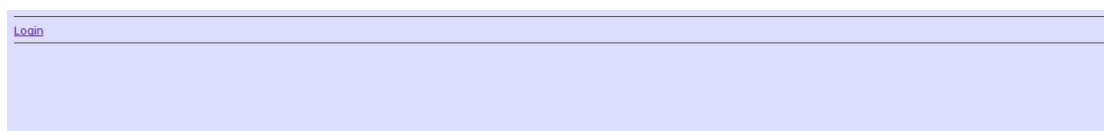


Εικόνα 31: Client-Εξέλιξη παιχνιδιού

Διαδικτυακή Εφαρμογή

Αρχική Οθόνη

Στην αρχική οθόνη ο χρήστης μπορεί να κάνει κλικ στον προβαλλόμενο σύνδεσμο προκειμένου να οδηγηθεί στην διαδικασία εισόδου (login)



Εικόνα 32: Αρχική οθόνη διαδικτυακής εφαρμογή

Είσοδος

Ο χρήστης εισάγει το username και το password και κάνοντας κλικ στο πλήκτρο Login εισέρχεται στην διαδικτυακή εφαρμογή. Σε περίπτωση αποτυχίας ενημερώνεται με κατάλληλο μήνυμα.



The screenshot shows a login interface with a light blue background. At the top, there is a horizontal bar with the text 'Username' followed by a text input field, 'Password' followed by another text input field, and a 'Login' button. Below this bar, there is a navigation menu with links: 'Insert Movie', 'Insert Movies from Web', 'Edit Movie', 'Delete Movie', 'Make SQL file', and 'Logout'.

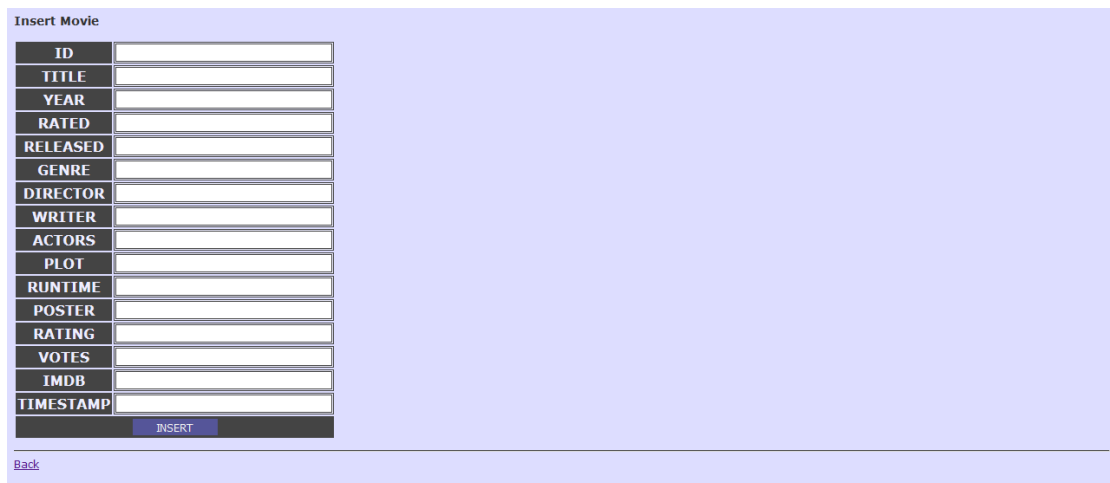
Εικόνα 33: Εισαγωγή στην διαδικτυακή εφαρμογή

Έξοδος

Η έξοδος από την εφαρμογή γίνεται με κλικ στον σύνδεσμο Logout.

Καταχώρηση Εγγραφής

Με κλικ στην επιλογή Insert Movie ο χρήστης οδηγείται στην φόρμα καταχώρησης των στοιχείων ταινίας. Με την ολοκλήρωση της καταχώρησης πρέπει να κάνει κλικ στο πλήκτρο Insert για να καταχωρηθεί η εγγραφή στην βάση δεδομένων.



The screenshot shows the 'Insert Movie' form. It has a title 'Insert Movie' at the top left. Below the title is a table with 16 rows, each representing a field for movie information: ID, TITLE, YEAR, RATED, RELEASED, GENRE, DIRECTOR, WRITER, ACTORS, PLOT, RUNTIME, POSTER, RATING, VOTES, IMDB, and TIMESTAMP. Each field has a corresponding text input field. At the bottom of the table is an 'INSERT' button. Below the button is a 'Back' link.

Εικόνα 34: Καταχώρηση εγγραφής στην διαδικτυακή εφαρμογή

Καταχώρηση Εγγραφών από το Web (scrapping)

Η δημιουργία της βάσης δεδομένων του παιχνιδιού έχει ως εξής: Ο χρήστης καταχωρεί τους τίτλους των επιθυμητών ταινιών σε ένα .txt αρχείο, το οποίο δίνεται σαν είσοδος σ'ένα σενάριο εκτέλεσης ενεργειών των Windows (batch script). Αυτό με τη σειρά του δημιουργεί ερωτήματα στο OMDb Api, από το οποίο και ανακτά τις πληροφορίες για κάθε εισαχθείσα ταινία. Όταν το script τελειώσει με την ανάκτηση πληροφοριών για την κάθε ταινία, δημιουργεί μια βάση δεδομένων, η οποία είναι έτοιμη για χρήση από την android εφαρμογή του server.

```
C:\Workspaces\icte_thesis\scrapper>create_sqlite.bat
Movies' file name: movielist.txt
Database name (provide suffix, too): movies.db
```

Ενημέρωση Εγγραφής

Με κλικ στην επιλογή Edit Movie ο χρήστης οδηγείται στην φόρμα επιλογής ταινίας από combobox. Αφού επιλέξει ταινία και κάνει κλικ στο πλήκτρο Search εμφανίζεται φόρμα παρόμοια με αυτήν της καταχώρησης που όμως στα πεδία της φαίνονται οι τρέχουσες τιμές της εγγραφής. Αφού ο χρήστης αλλάξει τις τιμές που επιθυμεί κάνει κλικ στο πλήκτρο Update για να καταγραφούν οι αλλαγές στην βάση δεδομένων.



Edit Movie

Title: My test movie ▾

SELECT

[Back](#)

Edit Movie	
ID	1001
TITLE	My test movie
YEAR	2012
RATED	6.7
RELEASED	13 Aug 2012
GENRE	Drama
DIRECTOR	Alex Black
WRITER	Alex Black (Screen Play)
ACTORS	Mario Alan, Clint Eastwood, Carl White
PLOT	-
RUNTIME	123
POSTER	http://webneel.com/daily/sites/default/files
RATING	2
VOTES	123
IMDB	
TIMESTAMP	
EDIT	

[Back](#)

Εικόνα 35: Ενημέρωση εγγραφής από την διαδικτυακή εφαρμογή

Διαγραφή Εγγραφής

Με κλικ στην επιλογή Edit Movie ο χρήστης οδηγείται στην φόρμα επιλογής ταινίας από combobox. Αφού επιλέξει ταινία και κάνει κλικ στο πλήκτρο Search εμφανίζεται φόρμα παρόμοια με αυτήν της καταχώρησης που όμως στα πεδία της φαίνονται οι τρέχουσες τιμές της εγγραφής. Αφού ο χρήστης αλλάξει τις τιμές που επιθυμεί κάνει κλικ στο πλήκτρο Update για να καταγραφούν οι αλλαγές στην βάση δεδομένων.

Delete Movie	
Title	My test movie
DELETE	

[Back](#)

Εικόνα 36: Διαγραφή εγγραφής από την διαδικτυακή εφαρμογή

Δημιουργία Αρχείου Ενημέρωσης Βάσης Δεδομένων

Αρκεί ο χρήστης να κάνει κλικ στο πλήκτρο Make SQL file. Με αυτή την ενέργειά του, δημιουργείται ή, σε περίπτωση που υπάρχει ήδη, τροποποιείται καταλλήλως ένα αρχείο το οποίο περιέχει όλα τα sql insert queries, που αντιστοιχούν στις υπάρχουσες στη βάση ταινίες. Αυτό το αρχείο μπορεί έπειτα να χρησιμοποιηθεί ως είσοδος στην υπάρχουσα βάση δεδομένων που συντηρούν οι android εφαρμογές.

Συμπεράσματα – Επεκτάσεις

Το λειτουργικό σύστημα Android βάσισε την επιτυχία του ως προς την διεισδυτικότητα του σε αρκετά μεγάλο βαθμό στο γεγονός ότι είναι opensource. Αυτό έκανε πολλούς προγραμματιστές να ασχοληθούν τόσο με την επέκταση των λειτουργιών του ίδιου του λειτουργικού συστήματος όσο και με την ανάπτυξη εφαρμογών που να τρέχουν σε αυτό καθώς είχαν την δυνατότητα να επεμβαίνουν στις λειτουργίες του αλλά μπορούσαν να απευθυνθούν και σε ένα ευρύτατο κοινό. Η κατάσταση αυτή συνέβαλε στην δημιουργία μίας μεγάλης κοινότητας που έχουν ως προσανατολισμό την ανάπτυξη Android εφαρμογών. Η ευρεία αυτή κοινότητα συνέβαλε με τον τρόπο της στην ταχύτερη και αρτιότερη ανάπτυξη του συστήματος. Για κάθε λειτουργία που αναπτύχθηκε στα πλαίσια του συστήματος ήταν εφικτό να εντοπιστούν ένα αρκετά μεγάλο πλήθος πηγών όπου χρειάστηκε υποστήριξη. Η υποστήριξη αυτή υπήρξε με την μορφή tutorials, συμβουλών σε διάφορα forum ή παραδειγμάτων όπου σε πολλές περιπτώσεις η περιγραφή των λύσεων ήταν ακριβής και λεπτομερής τόσο όσο να μπορεί να γίνει κατανοητή από developers οποιασδήποτε εμπειρίας. Το στοιχείο αυτό ενίσχυσε και το γεγονός ότι η τεχνολογία Android βασίζεται στην γλώσσα προγραμματισμού Java. Η Java είναι μια ισχυρή, αναγνωρισμένη και ώριμη γλώσσα προγραμματισμού πλήρως αποδεκτή από τις κοινότητες των προγραμματιστών και υιοθετημένη από αρκετές τεχνολογίες. Έτσι ήταν σχετικά εύκολο να εντοπιστούν λύσεις σε προβλήματα που παρουσιάστηκαν κατά την ανάπτυξη αλλά και κατά την διάρκεια των δοκιμών των εφαρμογών του client και του server.

Ανάλογη τεχνολογία χρησιμοποιήθηκε και για την ανάπτυξη του web server. Η php που χρησιμοποιήθηκε για την ανάπτυξη των λειτουργιών του web server επίσης βάσισε σε μεγάλο βαθμό την ανάπτυξη της στο γεγονός ότι ήταν opensource γλώσσα. Η μεγάλη κοινότητα προγραμματιστών που έχει διαμορφωθεί παγκοσμίως κατέστησε εύκολη την πρόσβαση σε χρήσιμες πληροφορίες για την ταχεία και ορθή ανάπτυξη των λειτουργιών του web server. Η βάση δεδομένων που χρησιμοποιήθηκε στην διαδικτυακή εφαρμογή, η MySql, είναι επίσης opensource και αρκετά ώριμη αφού αποτελεί

για αρκετά χρόνια βασική προτίμηση των προγραμματιστών για την οργάνωσή των δεδομένων των εφαρμογών που αναπτύσσουν. Έτσι και για την ανάπτυξη της βάσης δεδομένων της διαδικτυακής εφαρμογής ήταν εύκολο να βρεθούν λύσεις με απλές αναζητήσεις στο διαδίκτυο.

Η χρήση δημοφιλών opensource προσεγγίσεων για την ανάπτυξη των τριών εφαρμογών έδωσε την ευκαιρία στην ομάδα εργασίας να προμηθευτεί δωρεάν και να χρησιμοποιήσει ισχυρά περιβάλλοντα ανάπτυξης. Το Android Studio αποδείχθηκε ένα πλήρες πακέτο λογισμικού ικανό να ανταποκριθεί στις απαιτήσεις δύο απαιτητικών εφαρμογών, όπως ήταν αυτές που σχεδιάστηκαν και αναπτύχθηκαν στα πλαίσια της παρούσας εργασίας. Ο προγραμματιστής μέσα από τις λειτουργίες του έχει την δυνατότητα να διαμορφώσει τις διεπαφές των εφαρμογών με γραφικό τρόπο και παράλληλα να ελέγχει και τον XML κώδικα τους. Ο μηχανισμός ελέγχου συντακτικών σφαλμάτων αλλά και πρότασης ορθής σύνταξη του προγράμματος υπήρξε εξαιρετικά εξυπηρετικός και συνετέλεσε στην ταχύτερη συγγραφή του κώδικα αφού τα περισσότερα από τα σχετικά σφάλματα επιλύθηκαν πριν από την διαδικασία του debugging. Αρκετά χρήσιμος ήταν και ο μηχανισμός παραγωγής εικονικών μηχανών που ενσωματώνεται στο περιβάλλον ανάπτυξης που έδωσε την δυνατότητα δημιουργίας πολλαπλών εικονικών μηχανών προκειμένου να προσομοιωθεί η διαδικασία συμμετοχής πολλαπλών παικτών στο παιχνίδι. Εξίσου χρήσιμη υπήρξε και η δυνατότητα debugging χρησιμοποιώντας πραγματική κινητή συσκευή ώστε να χρησιμοποιηθεί για την δοκιμή της εφαρμογής του server. Έτσι οι δοκιμές των εφαρμογών client – server έγιναν σε πραγματικές συνθήκες. Τα μηνύματα σφαλμάτων υπήρξαν αρκετά κατατοπιστικά ως προς τις δυσλειτουργίες που παρατηρήθηκαν αφού ακόμα και αν τα ίδια δεν φανέρωναν την αιτία του σφάλματος ή/και τον τρόπο αποκατάστασης του, προσέφεραν αποδοτικές λέξεις κλειδιά για αναζήτηση λύσης στο διαδίκτυο. Το περιβάλλον ανάπτυξης netbeans αποτελεί ένα αρκετά πιο ώριμο από το Android Studio περιβάλλον ανάπτυξης για ένα μεγάλο εύρος γλωσσών προγραμματισμού (και για php). Προσφέρει παρόμοιους μηχανισμούς διαχείρισης σφαλμάτων με αυτούς του Android Studio καθώς επίσης και ολοκληρωμένη υποστήριξη στην συγγραφή php κώδικα. Το phpmyadmin είναι ένα απλό στην χρήση σύστημα διαχείρισης

βάσης δεδομένων το οποίο μπορεί να ανταποκριθεί ικανοποιητικά σε απαιτήσεις πολύ μεγαλύτερες από αυτές της παρούσας διαδικτυακής εφαρμογής.

Η λειτουργία των εφαρμογών android φανέρωσε ότι είναι εφικτό να λειτουργούν οι συσκευές Android ως απλοί server αν γίνει αποδοτική διαχείριση των διατιθεμένων πόρων. Η λειτουργία και η επικοινωνία των εφαρμογών έγινε χωρίς προβλήματα και η διαδικασία των παιχνιδιών ολοκληρώθηκε επίσης χωρίς προβλήματα. Καταδείχθηκε επίσης ότι και η επικοινωνία των εφαρμογών Android με άλλες διαδικτυακές εφαρμογές μπορεί επίσης να γίνει αποδοτικά. Για να επιτευχθεί η αποδοτικότητα αυτή γίνεται χρήση των χαρακτηριστικών του πολυνηματικού προγραμματισμού. Κάθε απαιτητική λειτουργία επιδιώκεται να λειτουργεί σε διαφορετικό νήμα. Συνεπώς, ο συγγραφέας προτείνει ανεπιφύλακτα παρόμοιες υλοποιήσεις, καθώς υπερταρούν των κλασικών server εφαρμογών, οι οποίες φιλοξενούνται συνήθως σε κάποιο απομακρυσμένο μηχάνημα. Προφανώς, τέτοια μηχανήματα έχουν σοβαρά κόστη συντήρησης και με την τροπή της τεχνολογίας, πρέπει εμείς, ως developers, να ελαχιστοποιούμε το οποιοδήποτε κόστος. Φυσικά, η επιλογή μιας λύσης όπως αυτή που υλοποιήθηκε στην παρούσα διπλωματική εργασία, απαιτεί πολύ προσεκτική διαχείριση των threads, ώστε να μην δημιουργθούν απρόσμενες καταστάσεις.

Η λειτουργία της προβολής των πληροφοριών που σχετίζονται με την ομάδα ανάπτυξης του συστήματος κατέδειξε την αρμονική συνεργασία των εφαρμογών που αναπτύσσονται για λειτουργία σε Android με τις εφαρμογές που παρέχονται με τον πυρήνα του συστήματος όπως ο Browser.

Η αρχιτεκτονική του συστήματος που αναπτύχθηκε είναι αρθρωτή. Αυτό σημαίνει ότι οι λειτουργίες της εφαρμογής έχουν κατανεμηθεί σε διαφορετικές μονάδες (είτε σε διαφορετικές εφαρμογές είτε σε διαφορετικές διεργασίες και δομικές μονάδες). Κατά συνέπεια είναι εφικτή η μεταβολή μέρους των λειτουργιών του συστήματος χωρίς να είναι αναγκαία η καθολική διαφοροποίησή του. Είναι επίσης δυνατή η επέκταση του σχετικά εύκολα. Οι μελλοντικές επεκτάσεις που μπορεί να εφαρμοστούν στο σύστημα είναι:

- Υποστήριξη πολλαπλών θεματολογιών για την διαμόρφωση των ερωτήσεων. Αυτό μπορεί να γίνει με την επέκταση των βάσεων δεδομένων με άλλους πίνακες οι οποίοι να είναι διαμορφωμένοι κατάλληλα ώστε να καταχωρούνται σε αυτούς πληροφορίες και από άλλα πεδία ενδιαφέροντος. Στην συνέχεια χρειάζεται να αναπτυχθούν κατάλληλες μέθοδοι στην κλάση MyDBHandler, στα πρότυπα της `makeQuestion` που να μπορούν να δημιουργούν ερωτήσεις πολλαπλών πιθανών απαντήσεων από τις πληροφορίες αυτές.
- Συγχώνευση των εφαρμογών client και server σε μία. Αυτό μπορεί να επιτευχθεί με την συγχώνευση των Activities των δύο εφαρμογών και την ανάπτυξη της δυνατότητας της να λειτουργεί τότε ως server και τότε ως client. Η διαδικασία αυτή δεν απαιτεί κάτι περισσότερο από την προσθήκη στην διεπαφή της αρχικής οθόνη ενός διακόπτη ο οποίος θα μεταβάλλει την διαδοχή στην πλοήγηση μεταξύ των διαθεσίμων λειτουργιών της.
- Μόνιμη συσχέτιση του παίκτη με τις συμμετοχές στα παιχνίδια. Αυτό θα μπορούσε να συμβεί με την προσθήκη μίας διαδικασίας εγγραφής των χρηστών σε κατάλληλα γραμμογραφημένο πίνακα της βάσης δεδομένων του server. Η σύνδεση στο παιχνίδι θα απαιτούσε συνδυασμό username/password και ο παίκτης θα συμμετείχε στο παιχνίδι με το username του.
- Μετατροπή της διαδικτυακής εφαρμογής με τρόπο τέτοιο που να μπορεί να λειτουργήσει ως server. Θα μπορούσε να γίνει η ανάπτυξη Java εφαρμογής στην οποία θα συμπεριλαμβάνονταν όλες οι κλάσεις της server εφαρμογής. Για την διαχείριση της θα χρειαζόταν να διαμορφωθούν κατάλληλες διεπαφές σε jsp ή ακόμα και σε php, από τις οποίες οι χρήστες θα καταχωρούσαν τις παραμέτρους των παιχνιδιών και θα έθεταν σε λειτουργία το socket λήψης αιτήσεων από τις Android εφαρμογές client. Θα απαιτούνταν πολύ μικρές επεμβάσεις στις λειτουργίες εξέλιξης του παιχνιδιού.

- Πολύγλωσση υποστήριξη. Όλα τα μηνύματα που εμφανίζονται στις εφαρμογές είναι καταχωρημένα σε xml αρχείο και κάθε φορά που χρησιμοποιούνται ανακτώνται από αυτό. Έτσι με αλλαγή των αρχείων αυτών με άλλα που θα περιέχουν τα αντίστοιχα μηνύματα σε διαφορετική γλώσσα θα κάνουν την εφαρμογής να λειτουργεί στην μητρική γλώσσα κάθε χρήστη.

Γενικότερα διαφαίνεται ότι η εφαρμογή είναι εύκολα επεκτάσιμη αφού όλες οι παραπάνω αναβαθμίσεις μπορούν να πραγματοποιηθούν με μικρές μεταβολές σε μικρό σχετικά μέρος του συστήματος χωρίς να επηρεάζονται τα περισσότερα από αυτά.

Η σχεδίαση και ανάπτυξη του όλου συστήματος, αποτέλεσε ευκαιρία ενασχόλησης του δημιουργού με ένα ευρύ σύνολο των χαρακτηριστικών των εφαρμογών που τρέχουν σε λειτουργικό σύστημα Android. Εξετάστηκε η οργάνωση των δεδομένων εφαρμογής σε τοπική βάση δεδομένων, η ανταλλαγή δεδομένων με διαδικτυακή εφαρμογή, η ανταλλαγή δεδομένων και ο συγχρονισμός μεταξύ διαφορετικών εφαρμογών Android, η πλοήγηση ανάμεσα σε διαφορετικές λειτουργίες και η ανταλλαγή δεδομένων μεταξύ αυτών, η προβολή πολυμέσων. Οι απαιτήσεις του συστήματος κάλυψαν ένα μεγάλο μέρος των ζητούμενων από αντίστοιχες εφαρμογές.

Αναφορές

- Burns, J. (2009). *Mobile Application Security on Android*. Ανάκτηση από Black Hat: <http://www.blackhat.com/presentations/bh-usa-09/BURNS/BHUSA09-Burns-AndroidSurgery-PAPER.pdf>
- Frenkiel, R. (2001). *A Brief History of Mobile Communications*. Ανάκτηση από WINLAB: http://www.winlab.rutgers.edu/~narayan/Course/Wireless_Revolution/vts%20article.pdf
- Harada, H., & Prasad, R. (2002). *Simulation and Software Radio for Mobile Communications*. Artech House.
- Jo Foley, M. (2012). *Microsoft's Windows Phone 8 finally gets a 'real' Windows core*. Ανάκτηση 2015, από ZDNET: <http://www.zdnet.com/blog/microsoft/microsofts-windows-phone-8-finally-gets-a-real-windows-core/12975>
- Pashtan, A. (2006). *Wireless Terrestrial Communications: Cellular Telephony*. Illinois: Eolls Publishers.
- Smith, G. (2013). *Forty Year Mobile Phone Anniversary*. Ανάκτηση 2015, από A New Domain: <http://anewdomain.net/2013/04/03/history-of-the-mobile-phone-infographic-motorola-dynatac-1973-plus-mobile-1946-innovations/>
- Tutorials Point. (2010). *History of Mobile Communication*. Ανάκτηση 2015, από Tutorials Point: http://www.tutorialspoint.com/umts/umts_history_of_mobile_communication.htm
- Windows. (2015). *Τι νέο υπάρχει στο Windows Phone 8.1*. Ανάκτηση 2015, από Windows Phone: <http://www.windowsphone.com/el-gr/how-to/wp8/basics/whats-new-in-windows-phone>