



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
"Διδακτική της Τεχνολογίας και Ψηφιακά Συστήματα"

ΑΝΑΠΤΥΞΗ ΚΟΙΝΩΝΙΚΗΣ
ΕΦΑΡΜΟΓΗΣ ΦΩΝΗΤΙΚΗΣ
ΣΥΝΘΕΣΗΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΚΑΛΑΪΤΖΙΔΗΣ ΘΕΟΔΩΡΟΣ
ΑΜ :ΜΕ 09052

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΝΙΚΗΤΑΣ-ΜΑΡΙΝΟΣ ΣΓΟΥΡΟΣ

Δεκέμβριος 2013, Αθήνα

Πίνακας Περιεχομένων

Πίνακας Περιεχομένων.....	2
i. Ευρετήριο Εικόνων.....	4
1 Εισαγωγή.....	5
1.1 Δομή Εργασίας.....	5
1.2 Σκοπός.....	6
2 Ανασκόπηση Διαθέσιμων Τεχνολογιών.....	7
2.1 Τα Μέσα Κοινωνικής Δικτύωσης.....	7
2.2 Φωνητική Σύνθεση Text To Speech.....	9
2.3 Ανασκόπηση υπάχουσας κατάστασης αναφορικά με την ανάλυσης και παρουσίαση μηνυμάτων στα κοινωνικά δίκτυα.....	11
2.4 Ενδεικτικές Εφαρμογές.....	13
2.4.1 SocialAudio (http://www.audiotweet.co.nr).....	13
2.4.2 SocialRadio (http://www.thesocialradio.com).....	14
2.4.3 Blindsquare.....	14
3 Περιγραφή Του Συστήματος.....	15
3.1 Αρχιτεκτονική τριών επιπέδων.....	16
3.2 Service-Oriented Αρχιτεκτονική.....	18
3.3 Η αρχιτεκτονική MVC.....	21
3.4 Object Relational Mapping.....	23
3.5 To Play Framework.....	25
3.6 To Twitter Bootstrap.....	26
3.7 Βιβλιοθήκη Text-To-Speech FreeTTS.....	27
3.8 Εργαλεία Ανάπτυξης.....	28
3.8.1 Eclipse Indigo.....	28
3.8.2 MySQL Workbench.....	29
3.8.3 Git.....	29
3.9 Τεχνολογίες Λογισμικού.....	29
3.9.1 Java.....	29
3.9.2 SQL.....	31
3.9.3 HTML5 - WebSocket.....	32
3.10 Ιδιαιτερότητες, προβλήματα και δυσκολίες.....	34
3.10.1 Web Standards.....	34
3.10.2 Rate Limiting.....	35
3.10.3 Api Versioning.....	36
4 Υλοποίηση Συστήματος.....	38
4.1 Δομή Βάσης Δεδομένων.....	38
4.1.1 Account.....	38
4.1.2 Band.....	38
4.1.3 Keyword.....	38
4.1.4 Spoken.....	39
4.1.5 Tweet.....	39
4.1.6 Play Evolutions.....	39
4.1.7 Media Result.....	39
4.1.8 Σχεσιακοί Πίνακες.....	40
4.2 Δομή Πηγαίου Κώδικα.....	42
4.2.1 Controllers και Views.....	43
4.2.2 Models.....	44
4.2.3 Λοιπές Οντότητες.....	45

5	Παρουσίαση Συστήματος.....	47
5.1	Το κεντρικό μενού.....	47
5.2	Η αναπαραγωγή tweets.....	50
5.3	Αναπαραγωγή μουσικής.....	53
6	Συμπεράσματα - Μελλοντικές Προεκτάσεις.....	55
6.1	Βελτίωση Μηχανισμού Text-To-Speech.....	55
6.2	Αναγνώριση Γλώσσας.....	56
6.3	Native Mobile Application.....	56
6.4	Εξειδικευμένο μουσικό API.....	56
I.	Βιβλιογραφία.....	58
a.	Βιβλία.....	58
b.	Ιστοσελίδες.....	58

i. Ευρετήριο Εικόνων

Εικόνα 1: Αρχιτεκτονική τριών επιπέδων	17
Εικόνα 2: Σύνοψη των εξωτερικών APIs με τα οποία επικοινωνεί η εφαρμογή.	20
Εικόνα 3: Το μοτίβο MVC.....	22
Εικόνα 4: Object Relational Mapping.....	24
Εικόνα 5: Σχεσιακό Μοντέλο της βάσης δεδομένων	41
Εικόνα 6: Η δομή του πηγαίνου κώδικα	43
Εικόνα 7: Controllers και views	44
Εικόνα 8: Επισκόπηση των μοντέλων	45
Εικόνα 9: Το κεντρικό user interface	47
Εικόνα 10: Υπομενού Band.....	48
Εικόνα 11: Παράθυρο διαλόγου νέας μπάντας.....	48
Εικόνα 12: Το μενού ρυθμίσεων.....	49
Εικόνα 13: Fader καθορισμού έντασης μουσικής.....	50
Εικόνα 14: Εμφάνιση των keywords	50
Εικόνα 15: Παράθυρο διαλόγου νέας λέξης-κλειδί.....	51
Εικόνα 16: Μήνυμα αναμονής	51
Εικόνα 17: Διάφορα tweets με διαφορετικό sentiment.....	52
Εικόνα 18: Τα κουμπιά ελέγχου αναπαραγωγής	52
Εικόνα 19: Η διαχείριση της λίστας αναπαραγωγής.....	53

1 Εισαγωγή

Η έλευση των κοινωνικών δικτύων τα τελευταία χρόνια έχει επιφέρει τεράστιες αλλαγές στις σχέσεις μεταξύ των ανθρώπων, από τον τρόπο επικοινωνίας με άλλους έως και τον τρόπο με τον οποίο εκφραζόμαστε. Η κοινωνική δικτύωση έχει σε μεγάλο βαθμό ανάγει το διαμοιρασμό περιεχομένου σε κύριο τρόπο έκφρασης της ατομικότητας, παρέχοντας πλατφόρμες ανταλλαγής περιεχομένου μεγάλης εμβέλειας που το καθιστούν προσβάσιμο και άμεσα διαθέσιμο σε πλήθος ανθρώπων. Η ύπαρξη αυτού του τεράστιου όγκου πληροφοριών και δεδομένων, τα οποία είναι συγχρόνως αντιστοιχισμένα σε χρήστες κοινωνικών δικτύων, προωθεί πρωτοβουλίες αξιοποίησης της υψηλής διαθεσιμότητας περιεχομένου με πρωτόγνωρους και δημιουργικούς τρόπους, με σκοπούς που διαφέρουν κάθε φορά ανάλογα με τα κίνητρα της υλοποίησης και τις υπάρχουσες ανάγκες.

1.1 Δομή Εργασίας

Η παρούσα εργασία αποτελείται από επτά κεφάλαια. Στο πρώτο κεφάλαιο γίνεται μια εισαγωγή στο αντικείμενο και παρουσιάζεται ο σκοπός της εργασίας.

Το κεφάλαιο παραθέτει μια συνοπτική περιγραφή της υπάρχουσας κατάστασης και την ανάγκη την οποία έρχεται να ικανοποιήσει η εφαρμογή που αναπτύχθηκε.

Στο δεύτερο κεφάλαιο γίνεται μια ανασκόπηση των διαθέσιμων τεχνολογιών και παρουσιάζεται η εξέλιξη του τεχνολογικού υποβάθρου που μπορεί να υποστηρίξει την προσπάθεια, ενώ παρουσιάζονται και ορισμένες παρόμοιες εφαρμογές.

Στο τρίτο κεφάλαιο παρουσιάζεται η αρχιτεκτονική του συστήματος, γίνεται η τεκμηρίωση της επιλογής της και παρατίθενται οι τεχνολογίες που

χρησιμοποιήθηκαν για την ανάπτυξή του, καθώς και οι ιδιαιτερότητες και δυσκολίες που συναντήθηκαν κατά την υλοποίηση.

Στο τέταρτο κεφάλαιο γίνεται μια αναλυτική παρουσίαση της δομής της βάσης δεδομένων, καθώς και ανάλυση της δομής του πηγαίου κώδικα.

Στο πέμπτο κεφάλαιο παρουσιάζονται ορισμένα σενάρια χρήσης του συστήματος. Γίνεται ανάλυση σε επιμέρους βήματα των τυπικών χρήσεων της εφαρμογής με τη μορφή εγχειριδίου χρήσης.

Τέλος, στο έκτο κεφάλαιο γίνεται μια ανακεφαλαίωση και εξαγωγή συμπερασμάτων αναφορικά με το σύστημα, ενώ παράλληλα προτείνονται μελλοντικές βελτιώσεις και επεκτάσεις.

1.2 Σκοπός

Πρωταρχικός σκοπός της εργασίας είναι η αξιοποίηση της διεπαφής που προσφέρει το Twitter για τη δημιουργία μιας εφαρμογής που να παρέχει στο χρήστη φωνητική αναπαραγωγή των tweets που σχετίζονται με τα ενδιαφέροντά του. Συγχρόνως, είναι επιθυμητή η αναπαραγωγή μουσικής της επιλογής του χρήστη στο παρασκήνιο. Έτσι ο στόχος είναι να μελετηθεί η βιβλιογραφία για να κατανοηθούν οι βασικές ανάγκες και να σχεδιαστεί ένα σωστά δομημένο σύστημα το οποίο να προσφέρει στο χρήστη μια ρευστή ακουστική εμπειρία τροφοδοτούμενη από το δημοφιλές κοινωνικό δίκτυο.

2 Ανασκόπηση Διαθέσιμων Τεχνολογιών

2.1 Τα Μέσα Κοινωνικής Δικτύωσης

Με τον όρο «**κοινωνικό δίκτυο**» (social network) εννοούμε μια κοινωνική δομή αποτελούμενη από ανθρώπους ή οργανισμούς που λέγονται “κόμβοι” (nodes) και οι οποίοι μοιράζονται μεταξύ τους κοινά ενδιαφέροντα, επικοινωνώντας με τη βοήθεια του διαδικτύου. Τα κοινωνικά δίκτυα δεν έχουν γεωγραφικούς περιορισμούς και συνήθως παρουσιάζουν μεγάλη αύξηση των μελών τους. Με τον ορισμό αυτό είναι παραπάνω από σαφές ότι ένα κοινωνικό δίκτυο, όπως υπαινίσσεται άλλωστε και το όνομά του, έχει κοινωνικές προεκτάσεις και θέτει σε αμφισβήτηση την άποψη ότι βασική προϋπόθεση της ύπαρξης κοινωνίας είναι η φυσική παρουσία των μελών της.

Τα μέσα κοινωνικής δικτύωσης διαφοροποιούνται από τα παραδοσιακά και βιομηχανικά μέσα σε πολλές πτυχές, όπως η ποιότητα, η απήχηση, η συχνότητα χρήσης, η χρηστικότητα, και η αμεσότητα. Οι χρήστες του Διαδικτύου εξακολουθούν να περνούν περισσότερο χρόνο με τις ιστοσελίδες κοινωνικής δικτύωσης από κάθε άλλο είδος ιστοσελίδας, ενώ υπολογίζεται ότι ο συνολικός χρόνος που δαπανάται για την κοινωνική μέσων μαζικής ενημέρωσης στις ΗΠΑ σε PC και φορητές συσκευές αυξήθηκαν κατά 37 τοις εκατό σε 121 δισεκατομμύρια λεπτά τον Ιούλιο του 2012 σε σύγκριση με 88.000 εκατομμύρια λεπτά τον Ιούλιο του 2011. Για προωθητές περιεχομένου, τα οφέλη της συμμετοχής σε μέσα κοινωνικής δικτύωσης έχουν προχωρήσει πέρα από την απλή κοινωνικό διαμοιρασμό περιεχομένου στην ανάπτυξη φήμης και να αναζήτηση ευκαιριών σταδιοδρομίας και σταθερό εισόδημα.

Υπάρχουν πολυάριθμα μέσα κοινωνικής δικτύωσης διαθέσιμα στο ευρύ κοινό, τα οποία μπορούν να διαχωριστούν σε διάφορες κατηγορίες:

- Κοινωνικά Δίκτυα Βασισμένα στο Προφίλ Χρήστη: Αφορά τις υπηρεσίες αυτές που βασίζονται στο προφίλ των χρηστών και λειτουργούν έχοντας ως βάση τις σελίδες των μελών τους. Οι σελίδες αυτές περιέχουν πληροφορίες σχετικά με προσωπικά στοιχεία κάθε μέλους

όπως είναι φωτογραφίες, ενδιαφέροντα που μπορεί να έχει και σκέψεις που οι χρήστες δημοσιεύουν όποτε το θελήσουν. Μια σημαντική δυνατότητα των χρηστών είναι ο κοινωνικός διαμοιρασμός περιεχομένου, όπως κείμενα, βίντεο, σχόλια στα προφίλ χρηστών που είναι φίλοι στο δίκτυο, ή να χρησιμοποιούν τα εργαλεία αξιολόγησης για να επικροτήσουν ή όχι τις πληροφορίες που είναι ήδη αναρτημένες. Τέλος, άλλη μια σημαντική δυνατότητα που παρέχεται από τέτοια μέσα είναι η χρήση ζωντανής σύνδεσης με εικόνα και ήχο μεταξύ των χρηστών καθώς και τα σύντομα μηνύματα (chat). Το πιο αντιπροσωπευτικό παράδειγμα τέτοιων κοινωνικών δικτύων είναι το Facebook.

- Κοινωνικά δίκτυα βασισμένα στο περιεχόμενο: Είναι τα δίκτυα που βασίζονται όχι τόσο στο προφίλ του κάθε χρήστη αλλά στο περιεχόμενο που δημοσιεύεται. Υπάρχουν πάρα πολλά παραδείγματα τέτοιων σελίδων, όπως είναι το Flickr και το Tumblr, τα οποία προσφέρουν στους χρήστες τη δυνατότητα κατασκευής προφίλ και προσθήκης φωτογραφιών. Τα μόνα σχόλια που γίνονται αφορούν τις φωτογραφίες και χρήστες που δεν είναι εγγεγραμμένα μέλη αυτών των υπηρεσιών δεν μπορούν να έχουν πρόσβαση σε αυτές. Άλλα παραδείγματα παρόμοιων υπηρεσιών είναι το πασίγνωστο YouTube με το οποίο γίνεται διαμοιρασμός βίντεο, του Shelfari που δίνει τη δυνατότητα στους χρήστες να δημιουργήσουν ράφια με βιβλία που έχουν διαβάσει ή που απλά κατέχουν και να γίνει η ανάλογη αξιολόγηση από τα υπόλοιπα μέλη. Τέλος το last.fm με το οποίο ακούει ο χρήστης μουσική μέσω του διαδικτύου, ή από το σκληρό δίσκο του υπολογιστή του και ταυτόχρονα χτίζεται ένα αναλυτικό προφίλ με τις προτιμήσεις του.
- Εικονικά Περιβάλλοντα Πολλαπλών Χρηστών: Αφορά χρήστες που αλληλεπιδρούν μεταξύ τους χρησιμοποιώντας μια ψεύδη εικονική

ταυτότητα μέσω avatar. Μπορούν μέσω αυτών να κοινωνικοποιούνται, να εξερευνούν τον εικονικό τους κόσμο ,να κάνουν εμπόριο ακόμα και να συμμετάσχουν σε ομαδικές δραστηριότητες. Παραδείγματα αποτελούν παιχνίδια σε φυλλομετρητές, όπως το Shakes & Fidget και το City of Steam.

- Δίκτυα Πρόσφατων Ενημερώσεων: Αφορούν υπηρεσίες που επιτρέπουν στους χρήστες την δημοσίευση μηνυμάτων με περιορισμό χαρακτήρων που φτάνει τους 140. Ιδιαίτερη πρόκληση σε τέτοια δίκτυα αποτελεί η συμπύκνωση των λεγομένων και των νοημάτων που ο χρήστης θέλει να μοιραστεί, εντός του ανώτατου επιτρεπόμενου ορίου χαρακτήρων. Ο πιο γνωστός και πιο διαδεδομένος ιστιότοπος σήμερα είναι το Twitter που έχει 200 εκατομμύρια χρήστες και δέχεται καθημερινά 200 εκατομμύρια περίπου μηνύματα από τους χρήστες του.

2.2 Φωνητική Σύνθεση Text To Speech

Η φωνητική σύνθεση ή σύνθεση ομιλίας είναι η τεχνητή παραγωγή της ανθρώπινης ομιλίας. Ένα σύστημα υπολογιστών που χρησιμοποιείται για το σκοπό αυτό καλείται συνθέτης ομιλίας, και μπορεί να εφαρμοστεί αποκλειστικά σε λογισμικό ή και σε υλικό προϊόντων. Ένα Text-To-Speech (TTS) μετατρέπει συμβολικές φωνητικές παραστάσεις σε ομιλία.

Η βασική μονάδα την οποία λαμβάνει υπ όψιν του ένα σύστημα text-to-speech είναι ο **φώνος** (phone), που ορίζεται ως το βραχύτερο ασυνεχές τεμάχιο ήχου σε μια ροή ομιλίας. Ο φώνος μπορεί να είναι οποιοδήποτε τμήμα ήχου ομιλίας ανεξάρτητα από τη θέση στην φωνολογία της γλώσσας. Μια άλλη σημαντική έννοια γι αυτά τα συστήματα είναι το **φώνημα**, ένα σύνολο, δηλαδή, από φώνους και ηχητικά χαρακτηριστικά τα οποία μπορούν να χαρακτηριστούν ως ίδια στο πλαίσιο της φωνολογίας μιας συγκεκριμένης γλώσσας.

Οι φώνοι και τα φωνήματα οργανώνονται σε **diphones** τα οποία ορίζονται ως γειτονικά ζεύγη φώνων. Ένα diphone χρησιμοποιείται για να δηλώσει μια καταγραφή της μετάβασης μεταξύ δυο φώνων. Εάν ο αριθμός των φώνων σε μια γλώσσα είναι P, τότε ο θεωρητικός αριθμός των πιθανών diphones είναι P^2 , αν και δεδομένου ότι όλες οι γλώσσες έχουν περιορισμούς για το τι ήχους μπορεί να συμβεί ένα δίπλα στο άλλο (βλ. phonotactics), ο αριθμός των diphones σε κάθε γλώσσα είναι συνήθως πολύ μικρότερη από P^2 . Η Ισπανική γλώσσα έχει περίπου 800 diphones και τα γερμανικά έχουν περίπου 2.500.

Τα diphones είναι χρήσιμα στη σύνθεση ομιλίας: Όταν τα προηχογραφημένα diphones συνδυάζονται για να δημιουργήσουν συνθετική ομιλία, οι ήχοι που προκύπτουν είναι πολύ πιο φυσικό από το συνδυασμό μόνο απλών φώνων, επειδή οι προφορά κάθε φώνου διαφέρει με βάση τους γύρω του.

Η συνθετική ομιλία μπορεί, λοιπόν, να δημιουργηθεί με συνένωση κομματιών καταγεγραμμένης ομιλίας που είναι αποθηκευμένα σε μια βάση δεδομένων. Συνήθως τα συστήματα διαφέρουν ως προς το μέγεθος των αποθηκευμένων μονάδων ομιλίας. Για παράδειγμα, ένα σύστημα που αποθηκεύει φώνους ή diphones παρέχει το μεγαλύτερο φάσμα της επιθυμητής γλώσσας, αλλά μπορεί να παρουσιάζει έλλειψη σαφήνειας και καθαρότητας ήχου. Για συγκεκριμένους τομείς χρήσης στους οποίους κρίνεται αναγκαίο, η αποθήκευση ολόκληρων λέξεων ή φράσεων επιτρέπει την παραγωγή συνθετικής ομιλίας υψηλής ποιότητας. Εναλλακτικά, ένα σύστημα-συνθέτης μπορεί να ενσωματώσει ένα μοντέλο της φωνητικής οδού και άλλα χαρακτηριστικά της ανθρώπινης φωνής για να δημιουργήσει μια εντελώς "συνθετικές" φωνή.

Η ποιότητα ενός συστήματος - συνθέτη ομιλίας κρίνεται από την ομοιότητα της παραγόμενης ομιλίας του με την ανθρώπινη φωνή και από την ικανότητά της να γίνει κατανοητή. Ένα ευέλικτο text-to-speech πρόγραμμα επιτρέπει σε άτομα με προβλήματα όρασης ή εγκεφαλικές αναπηρίες και προβλήματα στην ανάγνωση να ακούσουν γραπτά έργα σε έναν υπολογιστή στο σπίτι. Πολλά λειτουργικά συστήματα ηλεκτρονικών υπολογιστών έχουν αρχίσει να συμπεριλαμβάνουν συνθέτες ομιλίας από τις αρχές της δεκαετίας του 1990 .

Αναλυτικότερα, κάποιες από τις πιο πρακτικές εφαρμογές συστημάτων φωνητικής σύνθεσης είναι οι ακόλουθες:

1. Ανάγνωση δυναμικού κειμένου. Το Text-to-speech είναι χρήσιμο για φράσεις που διαφέρουν πάρα πολύ για να καταγραφούν και να αποθηκευτούν χρησιμοποιώντας όλες τις πιθανές εναλλακτικές περιπτώσεις. Για παράδειγμα, η φωνητική σύνθεση της ώρας είναι μια καλή χρήση για τη μετατροπή κειμένου σε ομιλία, επειδή η προσπάθεια και η αποθήκευση που εμπλέκονται στην εγγραφή όλων τις πιθανών ωρών, λεπτών και δευτερολέπτων δεν είναι διαχειρίσιμη.
2. Επιμέλεια κειμένων. Η ηχητική επιμέλεια του κειμένου και αριθμών βοηθά το χρήστη να πιάσει λάθη που ξέφυγαν κατά την πληκτρολόγηση και την οπτική διόρθωση.
3. Εξοικονόμηση χώρου αποθήκευσης. Text-to-speech είναι χρήσιμο για φράσεις που θα καταλαμβάνουν πολύ χώρο αποθήκευσης, εάν ήταν ηχογραφημένες σε ψηφιακή μορφή ήχου.
4. Ενημέρωση του χρήστη για διάφορα γεγονότα. Το Text-to-speech λειτουργεί καλά για ενημερωτικά μηνύματα. Για παράδειγμα, για να ενημερώσει τον χρήστη ότι μια εργασία εκτύπωσης είναι πλήρης, η αίτηση θα μπορούσε να πει "Εκτύπωση πλήρης" παρά να εμφανίσει ένα πλαίσιο μηνύματος και να απαιτεί από το χρήστη να κάνει κλικ στο κουμπί OK.
5. Παροχή ακουστικής ανατροφοδότησης. Το text-to-speech μπορεί να παρέχει ακουστική ανατροφοδότηση όταν η οπτική είναι ανεπαρκής ή αδύνατη. Για παράδειγμα, τα μάτια του χρήστη θα μπορούσε να είναι απασχολημένα με άλλη εργασία, όπως η μεταγραφή των δεδομένων από ένα έγγραφο σε χαρτί. Οι χρήστες που έχουν χαμηλή όραση μπορεί να βασίζονται στο text-to-speech ως μοναδικό μέσο για τους ανατροφοδότηση από τον υπολογιστή.

2.3 Ανασκόπηση υπάχουσας κατάστασης αναφορικά με την ανάλυσης και παρουσίαση μηνυμάτων στα κοινωνικά δίκτυα

Τον τελευταίο καιρό παρατηρείται όλο και μεγαλύτερο ενδιαφέρον για υλοποιήσεις text-to-speech εφαρμογών οι οποίες θα παρέχουν ολοκλήρωση με τα πιο δημοφιλή μέσα κοινωνικής δικτύωσης. Οι λόγοι που παρατηρείται μια τέτοια αύξηση είναι κυρίως πρακτικοί. Όπως αναφέρθηκε παραπάνω, παρατηρείται μια αύξηση στην πρόσβαση του παγκοσμίου ιστού από mobile συσκευές, οι οποίες δεν διαθέτουν πληκτρολόγιο και ή μεγάλη οθόνη, καθιστώντας βολική την αντικατάσταση οπτικού περιεχομένου από ακουστικό. Στις περισσότερες περιπτώσεις, οι υλοποιήσεις text-to-speech εφαρμογών στοχεύουν να διεισδύσουν στο πολυάσχολο αστικό περιβάλλον και αποσκοπούν στο να παρέχουν ένα πλαίσιο διαρκούς ενημέρωσης και επικοινωνίας στο χρήστη καθώς αυτός είναι απασχολημένος κάνοντας μια εργασία που να απαιτεί την οπτική του προσοχή. Τέτοια παραδείγματα είναι η ενημέρωση ενώ ο χρήστης οδηγεί, αθλείται ή κάνει μια δουλειά που δεν απαιτεί την πλήρη προσοχή του.

Δεν μπορούμε να αγνοήσουμε, επίσης, τα άτομα με ειδικές ανάγκες, εγκεφαλικά προβλήματα και προβλήματα όρασης ως ομάδες στις οποίες απευθύνονται οι κοινωνικές εφαρμογές φωνητικής σύνθεσης. Τέτοιες εφαρμογές αποσκοπούν στο να κάνουν διαθέσιμα κοινωνικά κανάλια ενημέρωσης και αλληλεπίδρασης σε άτομα των οποίων οι ειδικές ικανότητες δεν τους το επιτρέπουν μέχρι σήμερα, καθώς και επίσης να βελτιώσουν την ήδη παρεχόμενη εμπειρία από κοινωνικά δίκτυα.

Μια άλλη χρήση, για την οποία μάλιστα πρόσφατα η Apple κατέθεσε αίτηση για δίπλωμα ευρεσιτεχνίας, βρίσκεται θορυβώδη περιβάλλοντα, καθώς και στις περιπτώσεις όπου ο χρήστης δεν μπορεί να μιλήσει δυνατά, για παράδειγμα, σε μια συνεδρίαση. Η πρόταση της Apple για το πρώτο πρόβλημα που θα περιλαμβάνουν την ανίχνευση θορύβου στο ακουστικό. Όταν ο εξωτερικός θόρυβος είναι πολύ δυνατός, το τηλέφωνο θα μετατρέψει αυτόματα την ομιλία στην άλλη άκρη σε κείμενο. Και αν ένας χρήστης είναι σε μια συνεδρίαση, αυτός ή αυτή θα είναι σε θέση να πληκτρολογήσετε κείμενο, το οποίο θα μετατρέπεται σε ομιλία του και αποστέλλεται στο άλλο άκρο.

Αρκετές εταιρείες επενδύουν σε ανάπτυξη συστημάτων text-to-speech και γενικά θεωρείται ένας κλάδος με ολοένα και αυξανόμενο ενδιαφέρον. Εκτός από τα εμπορικά πακέτα και APIs, υπάρχουν αρκετές βιβλιοθήκες ανοιχτού κώδικα για την υιοθέτηση την φωνητικής σύνθεσης από διάφορες εφαρμογές. Το γνωστότερο από τα δωρεάν πακέτα είναι το Google Translate API, το οποίο, όχι μόνο παρέχει υψηλής ποιότητας φωνές και συνθέσεις, αλλά έχει διαθέσιμες πάνω από 50 γλώσσες και παρέχει αυτόματη αναγνώριση γλώσσας, κάνοντας δυνατή την ενσωμάτωση της τεχνολογίας από διάφορες εφαρμογές παγκοσμίως. Το Google Translate είναι, όπως είπαμε, διαθέσιμο δωρεάν αλλά για την αποφυγή κατάχρησης από τους χρήστες έχει κάποιους περιορισμούς, όπως το ανώτατο όριο 100 χαρακτήρων και τον περιορισμένο αριθμό επιτρεπόμενων αιτήσεων στο API του ανά λεπτό.

Μια άλλη αρκετά διαδεδομένη text-to-speech βιβλιοθήκη ανοιχτού λογισμικού είναι το FreeTTS, γραμμένο εξ ολοκλήρου σε Java, το οποίο θεωρείται από τα πιο αξιόπιστα και γρήγορα συστήματα φωνητικής σύνθεσης. Το FreeTTS χρησιμοποιείται από την εφαρμογή μας και αναλύεται παρακάτω.

2.4 Ενδεικτικές Εφαρμογές

2.4.1 SocialAudio (<http://www.audiotweet.co.nr>)

Το SocialAudio είναι μια απλή και δωρεάν υπηρεσία ιστού που επιτρέπει τη μετατροπή κειμένου που εισάγει ο χρήστης σε ήχο. Στη συνέχεια, ο χρήστης έχει την επιλογή να δημοσιεύσει τον ήχο στο Twitter και το Facebook. Η εφαρμογή χρησιμοποιεί το Google Translate για να μετατρέψει το κείμενο σε ήχο, ενώ παρέχει τη δυνατότητα δημιουργίας ενός minified url στο bit.ly για το διαμοιρασμό στα κοινωνικά μέσα.

Το SocialAudio υποστηρίζει αυτή τη στιγμή 58 διαφορετικές γλώσσες, όσες δηλαδή παρέχει το Google Translate, ενώ οι δημιουργοί του υποστηρίζουν ότι έχει δημιουργηθεί έτσι ώστε να λειτουργεί με πολλά διαφορετικά κοινωνικά

δίκτυα, αν και η μέχρι σήμερα ενσωμάτωση αφορά τρία από τα μεγαλύτερα κοινωνικά sites, το Facebook, το Twitter και το Tumblr.

2.4.2 SocialRadio (<http://www.thesocialradio.com>)

Το SocialRadio φέρνει τη λειτουργικότητα του text-to-speech στο Twitter, και επιτρέπει στο χρήστη να ακούσει το twitter feed του σαν να ήταν μια ραδιοφωνική εκπομπή. Αυτή τη στιγμή, το SocialRadio είναι διαθέσιμο μόνο ως δωρεάν Android app, ενώ υπό κατασκευή είναι τα iOS και Blackberry.

Η εφαρμογή υποστηρίζει αρκετές γλώσσες και επιτρέπει στο χρήστη να επιλέξει να ακούσει την πλήρη timeline του στο Twitter ή να επικεντρωθεί σε μια συγκεκριμένη λίστα, θέμα, λέξη κλειδί ή hashtag. Το SocialRadio διαβάζει τα tweets δυνατά, με μερικά δευτερόλεπτα καθυστέρηση σε σύγκριση με την πραγματική ώρα άφιξης του tweet στο χρονοδιάγραμμα του χρήστη. Ενδιάμεσα από τα tweets, ο χρήστης μπορεί να ακούει μουσική από τη δική του βιβλιοθήκη ενώ για τους χρήστες που βρίσκονται στις ΗΠΑ, υπάρχει η δυνατότητα να κάνουν streaming μουσικά κομμάτια από το Google Music .

2.4.3 Blindsquare

Το Blindsquare δίνει την ευκαιρία σε άτομα με ειδικές ανάγκες να χρησιμοποιούν συστήματα καθοδήγησης χρησιμοποιώντας user-generated δεδομένα από το Foursquare, text-to-speech, GPS και κάποια OpenStreetMap ολοκλήρωση. Ενώ οι χρήστες βρίσκονται στο δρόμο, η εφαρμογή αφηγείται το ταξίδι τους μέσω ακουστικών με πληροφορίες για κοντινά μέρη, διασταυρώσεις ή καθοδήγησης προς τον προορισμό τους. Ένα κούνημα του iPhone ή iPad δίνει στους χρήστες την τρέχουσα θέση τους και με ένα δεύτερο κούνημα μπορούν να κάνουν check-in σε ένα συγκεκριμένο μέρος στο Foursquare. Η εφαρμογή μπορεί επίσης να ελεγχθεί μέσω μιας απομακρυσμένης Bluetooth συσκευής, ενώ το τηλέφωνο βρίσκεται σε μια τσέπη ή τσάντα. Η εφαρμογή βρίσκεται τώρα σε beta μορφή και είναι διαθέσιμο στο App Store για 14,99 δολάρια.

3 Περιγραφή Του Συστήματος

Στην ενότητα αυτή γίνεται μια συνολική περιγραφή της εφαρμογής. Αναφέρονται οι αρχιτεκτονικές βάσει των οποίων έγινε ο σχεδιασμός και η ανάπτυξη της, καθώς και τα διάφορα εργαλεία που χρησιμοποιήθηκαν για την κατασκευή της.

Το σύστημα αποτελεί μια εφαρμογή web (web application), εφαρμογή, δηλαδή, η οποία είναι προσβάσιμη μόνο μέσω διαδικτύου χρησιμοποιώντας ένα φυλλομετρητή (browser) και, τουλάχιστον στο επίπεδο παρουσίασης, χρησιμοποιεί τεχνολογίες και γλώσσες προσανατολισμένες στον ιστό, όπως Javascript και HTML. Οι web εφαρμογές γίνονται ολοένα και πιο δημοφιλείς κυρίως λόγω της γενικευμένης χρησιμοποίησης και της ευκολίας χρήσης των προγραμμάτων περιήγησης, αλλά και λόγω της ευκολίας διατήρησης αναβαθμισμένου λογισμικού χωρίς τις ανάγκες διανομής και εγκατάστασης στους υπολογιστές που θέλουν να τις χρησιμοποιήσουν.

Η δημιουργία εφαρμογών web συχνά απλοποιείται χρησιμοποιώντας λογισμικό ανοικτού κώδικα, όπως Django, Ruby on Rails και Symfony, τα οποία ονομάζονται web frameworks. Τα frameworks αυτά διευκολύνουν την ταχεία ανάπτυξη εφαρμογών, επιτρέποντας μια ομάδα ανάπτυξης να επικεντρωθεί στα εκάστοτε τμήματα της εφαρμογής για τα οποία είναι αρμόδια, τα οποία είναι μοναδικά για τη δημιουργία της συγκεκριμένης εφαρμογής, χωρίς να χρειάζεται να επιλύσει κοινά αναπτυξιακά θέματα, όπως η διαχείριση των χρηστών, και η εξυγίανση του input. Παρόλο που πολλά από αυτά τα frameworks είναι open source, αυτό δεν είναι κατ' ανάγκη απαίτηση.

Η χρήση των web frameworks μπορεί να μειώσει συχνά τον αριθμό των σφαλμάτων σε ένα πρόγραμμα, καθιστώντας τον κώδικα απλούστερο, και επιτρέποντας μία ομάδα να επικεντρωθεί στο framework, ενώ μια άλλη σε μια συγκεκριμένη περίπτωση χρήσης. Σε εφαρμογές που εκτίθενται σε συνεχείς προσπάθειες εισβολής στο Διαδίκτυο, σχετίζονται με την ασφάλεια προβλήματα μπορούν να προκληθούν από σφάλματα στο πρόγραμμα. Ορισμένα ώριμα

frameworks μπορεί επίσης να προωθήσουν τη χρήση βέλτιστων πρακτικών ιστού.

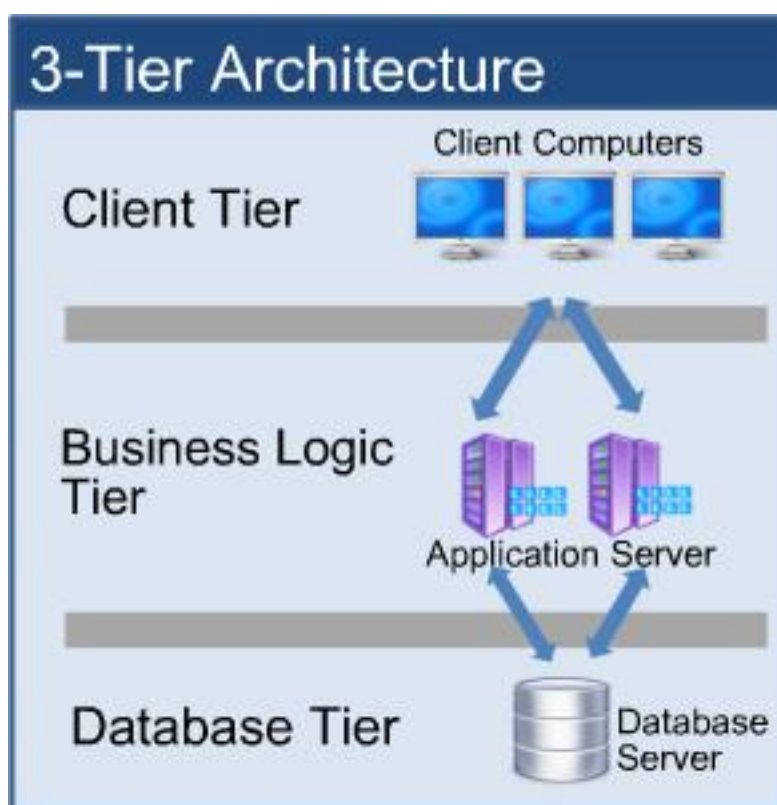
3.1 Αρχιτεκτονική τριών επιπέδων

Η εφαρμογή αναπτύχθηκε χρησιμοποιώντας την αρχιτεκτονική τριών επιπέδων. Η αρχιτεκτονική τριών επιπέδων (3-tier architecture) είναι μια client -server αρχιτεκτονική στην οποία το περιβάλλον εργασίας χρήστη , οι λογικές διαδικασίες (το business logic) και ο υπολογιστής αποθήκευσης και πρόσβασης στα δεδομένα, αναπτύσσονται και διατηρούνται ως ανεξάρτητες μονάδες , τις περισσότερες φορές ακόμη και σε ξεχωριστές πλατφόρμες.

Τα τρία βασικά στοιχεία της αρχιτεκτονικής είναι:

- Επίπεδο παρουσίασης (presentation tier)
Αυτό είναι το κορυφαίο επίπεδο της αρχιτεκτονικής. Η βαθμίδα παρουσίασης εμφανίζει πληροφορίες σχετικά με υπηρεσίες όπως την περιήγηση εμπορευμάτων, την αγορά και τα ψώνια. Επικοινωνεί με άλλες βαθμίδες εξάγοντας αποτελέσματα στον browser / πελάτη βαθμίδας και όλων των άλλων βαθμίδων του δικτύου. Με απλά λόγια είναι ένα στρώμα το οποίο οι χρήστες μπορούν να έχουν άμεση πρόσβαση, όπως μια ιστοσελίδα.
- Επίπεδο εφαρμογής (business logic , τη λογική σειρά, επίπεδο πρόσβασης σε βαθμίδα δεδομένων , ή μέση βαθμίδα)
Το επίπεδο εφαρμογής διαχωρίζεται από την βαθμίδα παρουσίασης και, ως δικό του επίπεδο, ελέγχει τη λειτουργικότητα μιας εφαρμογής διαμέσου λεπτομερούς επεξεργασίας .
- Επίπεδο Δεδομένων (Data tier)
Αυτό το επίπεδο αποτελείται από εξυπηρετητές βάσεων δεδομένων . Εδώ οι πληροφορίες αποθηκεύονται και ανακτώνται. Αυτή η βαθμίδα διατηρεί τα δεδομένα ουδέτερα και ανεξάρτητα από τους application servers ή το business logic. Το να έχουν τα δεδομένα το δικό τους επίπεδο οδηγεί σε βελτίωση της επεκτασιμότητας και την απόδοση.

Η αρχιτεκτονική αποτελεί μια βελτίωση της αρχιτεκτονικής client-server, διαχωρίζοντας την παρουσίαση από τις διαδικασίες επεξεργασίας και στέλνοντας το φόρτο εργασίας στον server, αφήνοντας τον πελάτη να αλληλεπιδρά με έναν “thin client”. Η αρχιτεκτονική αυτή είναι ιδανική για εφαρμογές πολλών χρηστών, ενώ είναι αρκετά αποτελεσματική και στον τομέα της ασφάλειας, καθώς το επίπεδο εφαρμογής προστατεύει τα δεδομένα.



Εικόνα 1: Αρχιτεκτονική τριών επιπέδων

Εκτός από τα γνωστά πλεονεκτήματα της αρχιτεκτονικής με σαφώς καθορισμένες διεπαφές, η αρχιτεκτονική τριών επιπέδων έχει ως στόχο να επιτρέπει οποιαδήποτε από τις τρεις κατηγορίες να αναβαθμιστούν ή να αντικατασταθούν ανεξάρτητα με τις αλλαγές στις απαιτήσεις ή την τεχνολογία. Για παράδειγμα, η αλλαγή του λειτουργικού συστήματος στη βαθμίδα παρουσίασης θα επηρεάσει μόνο τον κώδικα διεπαφή χρήστη (interface).

Τα μειονεκτήματα της αρχιτεκτονική αυτής είναι κυρίως η πολυπλοκότητα που εισάγει στην ανάπτυξη καθώς και στη συντήρηση της εφαρμογής. Επίσης ο φυσικός διαχωρισμός των τριών επιπέδων μέσω της χρήσης διαφορετικών μηχανημάτων για κάθε επίπεδο είναι δυνατό να έχει επιπτώσεις στην ταχύτητα και την απόδοση της εφαρμογής. Τέλος, η αρχιτεκτονική εισάγει επιπλέον κόστος λόγω του διαχωρισμού των επιπέδων και κατά τη διαδικασία scale up.

3.2 Service-Oriented Αρχιτεκτονική

Εάν ανεβούμε ένα επίπεδο αφαίρεσης αντιμετωπίσουμε τα τρία επίπεδα της προηγούμενης αρχιτεκτονικής σαν μια αυτόνομη μονάδα λειτουργικότητας, μπορούμε να πούμε πως η εφαρμογή ακολουθεί επίσης την service-oriented αρχιτεκτονική. Η service-oriented αρχιτεκτονική αποτελεί ένα πρότυπο σχεδιασμού που βασίζεται σε διακριτά κομμάτια λογισμικού που ονομάζονται υπηρεσίες και μπορούν να παρέχουν λειτουργικότητα και σε άλλες εφαρμογές.

Μια υπηρεσία αποτελεί μια αυτόνομη μονάδα λειτουργικότητας και μπορεί να συνδιαστεί με άλλες υπηρεσίες για να παρέχει την πλήρη λειτουργικότητα της εφαρμογής. Ο σκοπός του SOA είναι να επιτρέψει στους χρήστες να συνδιάζουν μεγάλα διακριτά κομμάτια λογισμικού για το σχηματισμό ad-hoc εφαρμογών που στηρίζονται σε υφιστάμενες υπηρεσίες.

Το κύριο όφελος του SOA είναι ότι επιτρέπει την ταυτόχρονη χρήση και εύκολη ανταλλαγή δεδομένων μεταξύ των προγραμμάτων των διαφόρων προμηθευτών χωρίς να απαιτείται επιπλέον προγραμματισμός ή πραγματοποίηση αλλαγών στις υπηρεσίες. Οι υπηρεσίες είναι επίσης επαναχρησιμοποιήσιμες, με αποτέλεσμα να απαιτείται χαμηλότερο κόστος ανάπτυξης και συντήρησης και να παρέχεται μεγαλύτερη αξία όταν η υπηρεσία έχει αναπτυχθεί και δοκιμαστεί. Η ύπαρξη, τέλος, άμεσα διαθέσιμων επαναχρησιμοποιήσιμων υπηρεσιών οδηγεί επίσης σε ταχύτερους χρόνους διάθεσης στην αγορά.

Η επικοινωνία των διακριτών σημείων λογισμικού γίνεται μέσω διεπαφών προγραμματισμού εφαρμογών (Application Programming Interfaces – APIs).

Μια τέτοια διεπαφή καθορίζει τον τρόπο με τον οποίο τα διαφορετικά συστατικά της εφαρμογής θα επικοινωνούν, παρέχοντας προδιαγραφές για ρουτίνες, δομές δεδομένων, κλάσεις αντικειμένων και μεταβλητές που θα χρησιμοποιηθούν κατά τις απομακρυσμένες κλήσεις.

Τα Web APIs, όπως λέγονται συνήθως οι διεπαφές που αναπτύσσονται στα πλαίσια της χρήσης σε ιστοσελίδες και διαδικτυακές εφαρμογές, ορίζονται ως ένα σύνολο μηνυμάτων HTTP τα οποία αποτελούνται από το HTTP ρήμα (GET – POST – PUT – DELETE) και τον ορισμό της δομής απάντησης, που είναι συνήθως eXtensible Markup Language (XML) ή JavaScript Object Notation (JSON). Τα Web APIs, αν και μπορούν να υποστηρίξουν την Service-Oriented αρχιτεκτονική, κινούνται συνήθως μακριά από το πρωτόκολλο SOAP, τείνοντας πιο πολύ στην μεταφορά της κατάστασης αναπαράστασης (Representational State Transfer – REST) των διαδικτυακών πόρων και προωθώντας την προσανατολισμένη σε πόρους αρχιτεκτονική (Resource-Oriented Architecture).

Ο συνδιασμός διαφόρων Web APIs και η ολοκλήρωσή τους σε μια φαινομενικά ενιαία εφαρμογή λέγεται Mashup και αποτελεί νέα τάση στην ανάπτυξη διαδικτυακών εφαρμογών. Συγκεκριμένα η εφαρμογή μας επικοινωνεί με 5 διαφορετικά APIs:

- Twitter API:
Η εφαρμογή αντλεί το κείμενο το οποίο θα μετατραπεί σε speech από το API του Twitter. Συγκεκριμένα, επικοινωνεί με το API για να αντλήσει τις λέξεις-κλειδιά με το μεγαλύτερο ενδιαφέρον τη δεδομένη στιγμή, καθώς και για να φέρει τα tweets για τη λέξη-κλειδί που έχει επιλέξει ο χρήστης.
- Alchemy API:
Αποτελεί μια δωρεάν διεπαφή που παρέχει sentiment analysis για δεδομένο κείμενο.
- TinySong:
Απλή διεπαφή που επιστρέφει τα τραγούδια που ταιριάζουν περισσότερο με τις λέξεις που εισάγει ο χρήστης. Χρησιμοποιείται για να αναγνωρίσει τα κομμάτια που επιθυμεί ο χρήστης να εισάγει στη λίστα

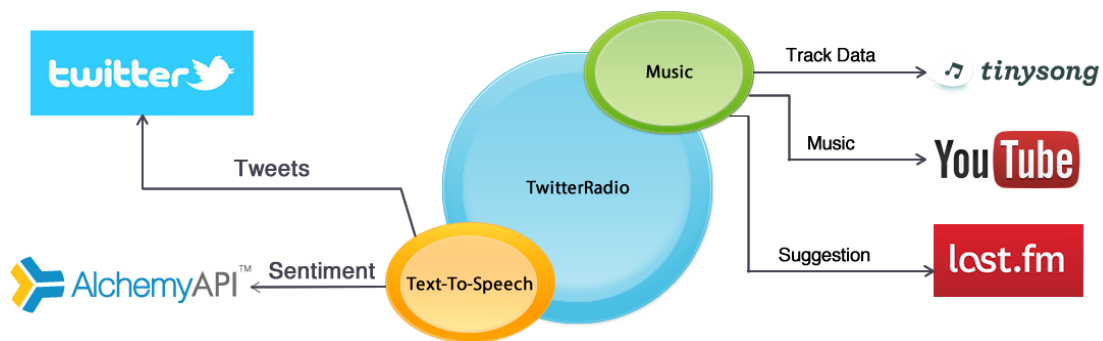
αναπαραγωγής του και να συλλέξει επιπλέον πληροφορίες που θα χρειαστούν για την αναζήτηση του κομματιού στο YouTube.

- LastFM API:

Το API του LastFM χρησιμοποιείται για να βρούμε παρόμοια τραγούδια και δημιουργούς με αυτά που έχει εισάγει ο χρήστης στη λίστα αναπαραγωγής του.

- YouTube API:

Το YouTube χρησιμοποιείται για να αναπαράγει τα τραγούδια που έχει ο χρήστης στη λίστα αναπαραγωγής του. Πιο εξειδικευμένα APIs για αναπαραγωγή μουσικής είναι επίσης το Grooveshark, καθώς και το API του Spotify, τα οποία όμως παρέχονται έναντι πληρωμένης συνδρομής.



Εικόνα 2: Σύνοψη των εξωτερικών APIs με τα οποία επικοινωνεί η εφαρμογή

Εκτός από τα εξωτερικά APIs, η εφαρμογή παρέχει ένα υποτυπώδες API για επιλεγμένες λειτουργίες. Δυνατότητες όπως η αναζήτηση τραγουδιού, η πρόταση (suggestion) τραγουδιού, η αλλαγή radio band και η προβολή των trends αποτελούν ξεχωριστές ενότητες λογισμικού, διαχωρισμένες από το user interface, οι οποίες έχουν δικό τους endpoint και μπορούν να χρησιμοποιηθούν κι από άλλες εφαρμογές. Για παράδειγμα, μπορεί μια εξωτερική εφαρμογή να χτίσει τη δική της λογική εύρεσης και παρουσίασης tweets, αλλά να χρησιμοποιήσει τη δική μας εφαρμογή για να ανακαλύψει παρόμοια μουσική απλώς καλώντας το [http://\[TwitterRadioDomain\]/async/suggestMedia](http://[TwitterRadioDomain]/async/suggestMedia).

3.3 Η αρχιτεκτονική MVC

Το Model-View-Controller (MVC) αποτελεί ένα πρότυπο σχεδιασμού το οποίο θέτει ένα σαφή διαχωρισμό μεταξύ της αναπαράστασης των πληροφοριών και της αλληλεπίδρασης του χρήστη με αυτές, ενώ επίσης καθορίζει και τις αλληλεπιδράσεις μεταξύ των στοιχείων αυτών. Το MVC έχει υιοθετηθεί ευρέως ως αρχιτεκτονική για ανάπτυξη web εφαρμογών σε όλες τις βασικές γλώσσες προγραμματισμού, ενώ έχουν αναπτυχθεί πολυάριθμα frameworks που επιβάλλουν το μοτίβο αυτό.

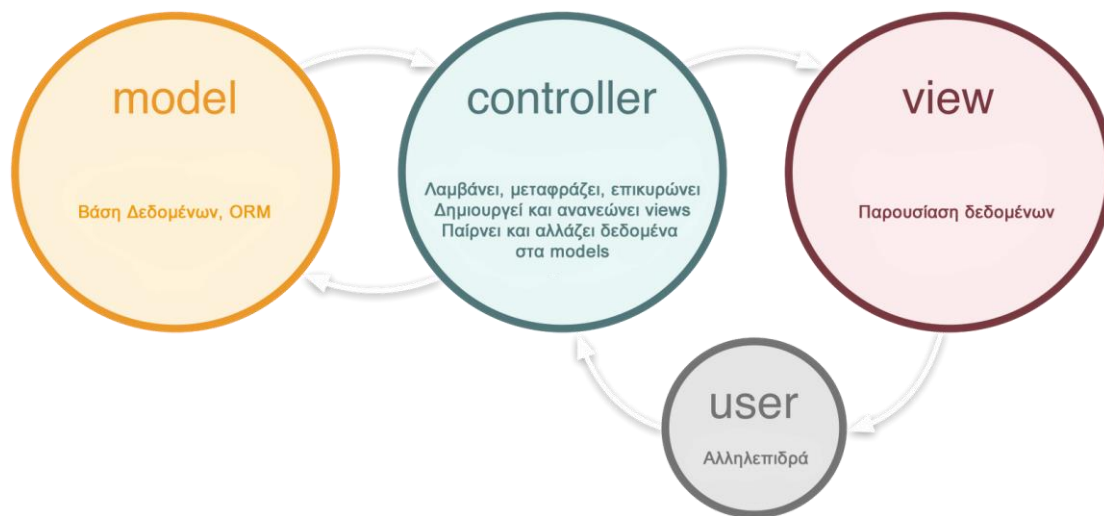
Το πρότυπο MVC ορίζει τρία βασικά είδη στοιχείων στα οποία διαχωρίζει όλη την εφαρμογή. Ο **controller** είναι ο μεσίτης μεταξύ Model και της View. Στέλνει εντολές στο Model για να αλλάξει την κατάστασή του και να πάρνει τα δεδομένα που ζητά και εν συνεχεία και αφού τα επεξεργαστεί τα στέλνει πίσω στο επίπεδο παρουσίασης, κάνοντας αλλαγές στο View.

Στο **model** τοποθετούμε τις λειτουργίες της εφαρμογής που σχετίζονται με την πρόσβαση στη βάση δεδομένων. Οι λειτουργίες αυτές είναι με τη μορφή μεθόδων με τις οποίες εκτελούμε διάφορες λειτουργίες διαχείρισης των δεδομένων που λαμβάνουμε από τη βάση.

.Η **view** αναλαμβάνει την παρουσίαση της εφαρμογής. Τις περισσότερες φορές μία View μιλάει με ένα controller, από τον οποίο δέχεται τα δεδομένα που έχει επεξεργαστεί από το model.

Το πρότυπο MVC αποτελεί μια αφηρημένη υλοποίηση πάνω στην αρχιτεκτονική τριών επιπέδων. Τα πρώιμα MVC frameworks ακολούθησαν τη λογική του thin client, διατηρώντας στο ελάχιστο την επεξεργασία στον πελάτη και τοποθετώντας όλη τη λογική και το βάρος της επεξεργασίας στον server. Σε αυτή την προσέγγιση ο πελάτης στέλνει αιτήσεις στον server και λαμβάνει μια πλήρως αναβαθμισμένη σελίδα κάθε φορά. Καθώς οι client-side τεχνολογίες ωριμάζουν, δημιουργούνται frameworks όπως το JavascriptMVC, και το

Backbone τα οποία επιτρέπουν τμήματα των MVC συστατικών να εκτελούνται εν μέρει στον πελάτη.



Εικόνα 3: Το μοτίβο MVC

Όπως προαναφέρθηκε, το μοτίβο MVC είναι ευρέως διαδεδομένο ως αρχιτεκτονική ανάπτυξης web εφαρμογών καθώς παρουσιάζει σαφή πλεονεκτήματα σε σχέση με παραδοσιακές τεχνικές ανάπτυξης τέτοιων εφαρμογών.

Το πιο βασικό πλεονέκτημα του MVC είναι ο διαχωρισμός των προβλημάτων (separation of concerns). Ουσιαστικά δημιουργείται μία εφαρμογή η οποία έχει τρία επίπεδα, το επίπεδο των models, το επίπεδο των controllers και το επίπεδο των views και το κάθε επίπεδο επιτελεί ξεχωριστό έργο και ταυτόχρονα συνεργάζεται με τα άλλα επίπεδα. Σε μία καλογραμμένη MVC εφαρμογή τα τρία επίπεδα είναι ξεκάθαρα καθορισμένα και δεν συμπλέκονται.

Το δεύτερο πλεονέκτημα της MVC αρχιτεκτονικής είναι η επεκτασιμότητα, η δυνατότητα, δηλαδή, που διαθέτει μία εφαρμογή, κατά την οποία μπορούμε μελλοντικά να προσθέσουμε λειτουργίες σε αυτή ή να αλλάξουμε κάποιες από τις ήδη υπάρχουσες και με άλλα αποτελέσματα. Η συνήθης μορφή με την οποία πραγματώνεται η έννοια της επεκτασιμότητας στο MVC είναι η χρήση plugins, τα οποία φορτώνονται δυναμικά και κατά βούληση και τα οποία είναι δυνατόν

να τροποποιούν τη λειτουργία της εφαρμογής ή να προσθέτουν εντελώς νέα λειτουργικότητα και δυνατότητες.

Ένα άλλο πολύ κρίσιμο πλεονέκτημα είναι η ελεγχιμότητα (testability), καθώς η test-driven ανάπτυξη λογισμικού γίνεται όλο και περισσότερο διαδεδομένη και όλο και περισσότερα frameworks αναπτύσσονται για testing. Οι MVC εφαρμογές έχουν την δυνατότητα να είναι ελέγξιμες και με τον τρόπο αυτό συντηρούνται πιο εύκολα. Στα MVC frameworks μπορούμε με πολλή ευκολία να γράψουμε unit tests με τα οποία ελέγχουμε τους controller και τις λειτουργίες τους, καθώς και τα models.

Τέλος, τα περισσότερα MVC frameworks για web applications δίνουν τη δυνατότητα να έχουμε καθαρές δευθύνσεις url, διευθύνσεις, δηλαδή, στις οποίες οι παράμετροι γίνονται μέρος του url path και οι οποίες είναι πιο φιλικές σε μηχανές αναζήτησης. Για παράδειγμα, σε μια PHP σελίδα, το ακόλουθο url

http://example.com/article/page.php?action=show&art_id=236.

μπορεί να γραφτεί όπως παρακάτω

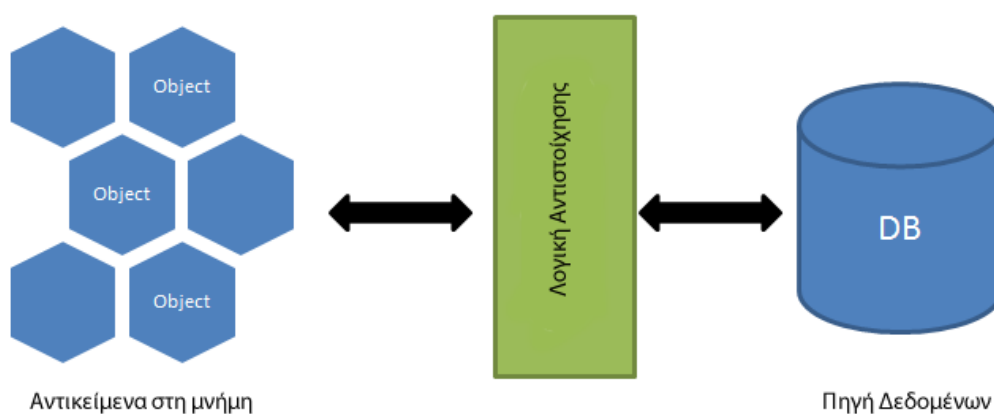
<http://example.com/articles/show/search-engine-friendly-article-name>

3.4 Object Relational Mapping

Το Object-Relational Mapping (ORM) είναι ένας μηχανισμός που καθιστά δυνατή την πρόσβαση και το χειρισμό αντικειμένων μιας εφαρμογής, χωρίς να χρειάζεται να απαιτείται γνώση για τον τρόπο με τον οποίο τα αντικείμενα σχετίζονται με τις πηγές δεδομένων τους. Το ORM επιτρέπει στους προγραμματιστές να θεωρούν δεδομένη τη συνέπεια των business αντικειμένων, ακόμη και οι πηγές των δεδομένων τους και οι εφαρμογές που έχουν πρόσβαση σε αυτά αλλάζουν. Αυτό επιτυγχάνεται με τη μετατροπή των

δεδομένων σε αντικείμενα γλώσσας προγραμματισμού τα οποία προσομοιάζουν μια εικονική αντικειμενοστραφή βάση δεδομένων.

Το ORM διαχειρίζεται τα στοιχεία χαρτογράφησης και αντιστοίχησης μεταξύ ενός συνόλου αντικειμένων και των σχεσιακών βάσεων δεδομένων, αποθετηρίων XML ή άλλων πηγών δεδομένων με τις οποίες τα αντικείμενα σχετίζονται, ενώ ταυτόχρονα χρησιμοποιώντας αφαιρετική λογική (abstraction) κρύβει τις λεπτομέρειες των σχετικών διεπαφών από προγραμματιστές και τον κώδικα υψηλότερου επιπέδου. Έτσι, μέσω της απομόνωσης και ενσωμάτωσης όλων των αλλαγών στην ίδια την πηγή δεδομένων, σε περίπτωση που αλλάξει το API τους, απαιτείται η αλλαγή μόνο του ORM και όχι ολόκληρης της εφαρμογής. Αυτή η ικανότητα επιτρέπει στους προγραμματιστές να επωφεληθούν από τις νέες κλάσεις όταν αυτές γίνονται διαθέσιμες και επίσης καθιστά εύκολη την επέκταση των εφαρμογών. Σε πολλές περιπτώσεις, οι αλλαγές στο ORM μπορούν να ενσωματώσουν νέες τεχνολογίες και λειτουργικότητα χωρίς να απαιτούνται αλλαγές στον κώδικα για τις σχετικές εφαρμογές. Η θέση του ORM σε σχέση με τα υπόλοιπα δομικά στοιχεία μιας εφαρμογής φαίνεται στην ακόλουθη εικόνα



Εικόνα 4: Object Relational Mapping

Η έννοια του ORM υπάρχει ήδη αρκετά χρόνια και απαντώνται πολυάριθμα ώριμα frameworks σε όλες τις ευρέως διαδεδομένες γλώσσες προγραμματισμού,

όπως το Hibernate (και η μεταφορά του σε .NET, NHibernate), το DoctrineORM και το SQLAlchemy, επιτρέποντας στους προγραμματιστές να συνεχίσουν να γράφουν κώδικα στη γλώσσα που γνωρίζουν καλύτερα.

Το μεγαλύτερο μειονέκτημα του ORM είναι η μείωση της απόδοσης μιας εφαρμογής, αφού αποτελεί από μόνο του ένα επιπλέον στρώμα αφαίρεσης, το persistence layer. Η παροχή ενός γενικευμένου API στα ανώτερα λογικά στρώματα απαιτεί μεγαλύτερο χρόνο επεξεργασίας, ενώ είναι εύκολο για προγραμματιστές που δεν γνωρίζουν καλά το framework να εισάγουν επιπλέον καθυστερήσεις που μειώνουν τη συνολική απόδοση και αποτελεσματικότητα. Αρκετές φορές, κρίνεται απαραίτητη η απ' ευθείας επικοινωνία με την πηγή δεδομένων για τη άντληση συγκεκριμένων πληροφοριών με τρόπο που θα ήταν χρονοβόρος χρησιμοποιώντας το ORM. Σε περίπτωση που κάτι τέτοιο γίνεται συχνά σε μια εφαρμογή, είναι επιθυμητή η εξέταση του βαθμού στον οποίο η υιοθέτηση ενός ORM έχει πραγματικό νόημα και ουσιαστικό όφελος.

Επίσης, για την αξιοποίηση των δυνατοτήτων που ένα ORM έχει να προσφέρει, είναι απαραίτητη η εκμάθησή του, καθώς τα περισσότερα τέτοια frameworks έχουν τις δικές τους ιδιαιτερότητες και συμβάσεις. Η διαδικασία εκμάθησης των περισσότερων ORM είναι απαιτητική, αφού τα περισσότερα έχουν μεγάλο learning curve, ενώ υπάρχουν διαφορετικά επίπεδα εκμάθησης, τα υψηλότερα των οποίων μπορούν να κάνουν οδηγίσουν σε μεγαλύτερη απόδοση της εφαρμογής που χρησιμοποιεί το ORM.

3.5 To Play Framework

Το Play είναι ένα web framework ανοιχτού κώδικα, γραμμένο σε Java και Scala, το οποίο ακολουθεί το μοτίβο MVC. Έχει ως στόχο τη βελτίωση της παραγωγικότητας δίνοντας έμφαση στη σύμβαση αντί των ρυθμίσεων (convention over configuration) και την εμφάνιση των σφαλμάτων στο φυλλομετρητή. Επίσης δίνει τη δυνατότητα hot code reloading, την ανανέωση, δηλαδή, μόνο εκείνων των κλάσεων που έχουν τροποποιηθεί από την τελευταία φορά που ένας χρήστης κάλεσε την εφαρμογή στο φυλλομετρητή του, με

αυτόματο τρόπο χωρίς να απαιτείται από τον προγραμματιστή να δίνει κάθε φορά την εντολή για compile.

Το Play Framework είναι έντονα εμπνευσμένο από ώριμα ευρέως υιοθετημένα web frameworks που έχουν αναπτυχθεί για άλλες γλώσσες προγραμματισμού, όπως το Django και το Rails, παρέχοντας έναν built-in JBoss Netty εξυπηρετητή, ενώ εφαρμογές γραμμένες σε Play μπορούν να διανέμονται και σαν WAR αρχεία για τυποποιημένους Java EE application servers.

Τέλος, το framework διαθέτει σαν συστατικά κάποιες ευρέως διαδεδομένες Java βιβλιοθήκες για τη διευκόλυνση ανάπτυξης web εφαρμογών. Κατ' αρχάς, χωρίς να είναι απαραίτητο, παρέχει δυο ξεχωριστά ORM για την επικοινωνία με τη βάση. Επίσης, όπως και πολλά άλλα frameworks που χρησιμοποιούν την αρθρωτή αρχιτεκτονική, παρέχει έναν κεντρικό διαχειριστή προσθέτων πακέτων ο οποίος αναλαμβάνει να κατεβάζει και να διατηρεί ενημερωμένες εκδόσεις των ζητούμενων plugins. Τέλος, εκτός από πληθώρα βοηθητικών όπως JSON και XML parser και προστασία από SQL injection, διαθέτει μηχανισμό διαχείρισης προτύπων templates για το επίπεδο παρουσίασης.

3.6 Το Twitter Bootstrap

Το Twitter Bootstrap είναι ουσιαστικά μια δωρεάν συλλογή εργαλείων για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών. Περιέχει πρότυπα σχεδιασμού σε HTML και CSS για κουμπιά, μενού πλοήγησης, ενότητες διαφήμισης και άλλων στοιχείων του περιβάλλοντος, καθώς και προαιρετικές επεκτάσεις JavaScript. Το Bootstrap είναι λογισμικό ανοικτού κώδικα και είναι διαθέσιμο στο GitHub. Οι προγραμματιστές ενθαρρύνονται να συμμετέχουν στο έργο και να κάνουν τη δική τους συνεισφορά στην πλατφόρμα.

Το Bootstrap έχει σχετικά ελλιπή υποστήριξη για HTML5 και CSS 3, αλλά είναι συμβατό με όλα τα δημοφιλή προγράμματα περιήγησης. Βασικές πληροφορίες

συμβατότητας των ιστοσελίδων ή εφαρμογών είναι διαθέσιμες για όλες τις συσκευές και τα προγράμματα περιήγησης. Υπάρχει μια έννοια της μερικής συμβατότητας που κάνει τα βασικά στοιχεία μιας ιστοσελίδας διαθέσιμα για όλες τις συσκευές και τα προγράμματα περιήγησης. Για παράδειγμα, οι ιδιότητες στρογγυλεμένες γωνίες, κλίσεις και οι σκιές, ιδιότητες που θεσπίστηκαν για το CSS3, χρησιμοποιούνται από Bootstrap παρά την έλλειψη υποστήριξης από παλιότερα προγράμματα περιήγησης στο Web . Αυτά επεκτείνουν τη λειτουργικότητα του πακέτου εργαλείων , αλλά δεν απαιτούνται για τη χρήση του .

Από την έκδοση 2.0 υποστηρίζει επίσης responsive σχεδιασμό. Αυτό σημαίνει ότι η διάταξη των ιστοσελίδων προσαρμόζει δυναμικά , λαμβάνοντας υπόψη τα χαρακτηριστικά της συσκευής που χρησιμοποιείται (PC, tablet, κινητό τηλέφωνο).

3.7 Βιβλιοθήκη Text-To-Speech FreeTTS

Το FreeTTS είναι λογισμικό φωνητικής σύνθεσης που επιτρέπει τη μετατροπή κειμένου σε ήχο. Το FreeTTS αποτελεί λογισμικό ανοιχτού κώδικα και είναι γραμμένο εξ ολοκλήρου στη γλώσσα Java, πάνω στο ενσωματωμένο Sun Speech API, μέσω του οποίου παρέχει ομιλία κειμένου από τρεις διαφορετικές ανδρικές φωνές.

Το FreeTTS μπορεί επίσης να τροποποιηθεί ώστε να ενσωματώνει τη διεπαφή MBROLA, έναν αλγόριθμο φωνητικής σύνθεσης, μέσω του οποίου μπορεί να παρέχει ομιλία από επιπλέον φωνές. Το MBROLA είναι ενεργό πολλά χρόνια αποτελεί προϊόν διεθνούς συνεργασίας διαφόρων εργαστηρίων έρευνας στο πεδίο, παρέχοντας έναν εντυπωσιακά μεγάλο αριθμό βάσεων δεδομένων διφώνων, μέσω των οποίων πραγματοποιείται η ομιλία, που αντιστοιχεί σε μεγάλο αριθμό φωνών σε διαφορετικές γλώσσες.

3.8 Εργαλεία Ανάπτυξης

Τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής είναι τα εξής:

- Eclipse Indigo, εργαλείο συγγραφής κώδικα.
- MySql Workbench, εργαλείο διαχείρισης της βάσης.
- Φυλλομετρητές διαδικτύου για την προβολή, δοκιμή και λήψη feedback από την εφαρμογή
- Git για version control

3.8.1 Eclipse Indigo

Το Eclipse IDE αποτελεί μία ευρέως διαδεδομένη πλατφόρμα ανάπτυξης εφαρμογών της κοινότητας του ανοικτού κώδικα που δημιουργήθηκε από την εταιρεία IBM. Είχε ως σκοπό τη δημιουργία μίας ισχυρής πλατφόρμας ανάπτυξης λογισμικού με βάση τη Java που θα έδινε τη δυνατότητα στο προγραμματιστή να ελέγχει το πρόγραμμα σε ολόκληρο το κύκλο ζωής του. Με το πέρασμα των χρόνων αναπτύχθηκε όλο και περισσότερο λόγω των πρόσθετων λειτουργιών κάνοντας τη δουλειά της ανάπτυξης εφαρμογής όλο και πιο εύκολη καθώς με έτοιμες βιβλιοθήκες κώδικα.

Σήμερα υποστηρίζει τις πιο γνωστές γλώσσες προγραμματισμού όπως είναι οι Java, C, C++, Python, Ruby, Fortran, Cobol, PHP κ.α. Επίσης, οι περισσότεροι παροχοί full stacks για ανάπτυξη web εφαρμογών, όπως τα CloudFoundry, Heroku και Amazon AWS παρέχουν δικά τους plugins για τη διευκόλυνση της διαδικασίας deployment στις δικές τους υποδομές. Εκτός από τη βαρύτητα της πλατφόρμας στην αγορά, εξίσου σημαντικό είναι ότι το Eclipse είναι μια εντελώς δωρεάν πλατφόρμα την οποία μπορεί να χρησιμοποιηθεί για την δημιουργία προγραμμάτων προσωπικής χρήσης, εμπορικής ή ανοικτού λογισμικού.

3.8.2 MySQL Workbench

Το MySQL Workbench είναι ένα εργαλείο σχεδιασμού βάσεων δεδομένων, το οποίο ενσωματώνει λειτουργικότητα για τη δημιουργία, διαχείριση και συντήρηση σε ένα ολοκληρωμένο περιβάλλον ανάπτυξης για το σύστημα βάσης δεδομένων MySQL.

Το MySQL Workbench παρέχει στους προγραμματιστές και τους διαχειριστές βάσεων δεδομένων ένα ολοκληρωμένο περιβάλλον με εργαλεία για σχεδιασμό και μοντελοποίηση, SQL ανάπτυξη, δημιουργία αντιγράφων ασφαλείας και database replication.

3.8.3 Git

Το Git είναι ένα κατακευματισμένο σύστημα ελέγχου εκδόσεων και διαχείρισης πηγαίου κώδικα με έμφαση στην ταχύτητα. Κάθε φάκελος εργασίας αποτελεί για το Git ένα πλήρες αποθετήριο (repository) με πληρη καταγραφή ιστορικού και δυνατότητες παρακολούθησης. Η κύρια διαφορά του Git από άλλα συστήματα version control είναι η έλλειψη της έννοιας της κεντρικής δομής που διαχειρίζεται το αποθετήριο. Ένα Git project δεν απαιτεί πρόσβαση στο διαδίκτυο και δεν εξαρτάται από έναν κεντρικό server.

3.9 Τεχνολογίες Λογισμικού

3.9.1 Java

Η Java είναι μία αντικειμενοστρεφής γλώσσα προγραμματισμού που δημιουργήθηκε από την εταιρεία *Sun Microsystems* στις αρχές του 1991. Ο κύριος σκοπός δημιουργίας της ήταν η κάλυψη αναγκών που είχαν προκύψει και που οι ήδη υπάρχουσες γλώσσες προγραμματισμού δεν μπορούσαν να καλύψουν, κυρίως όσον αφορά την ανάγκη μιας γλώσσας που θα είναι ανεξάρτητη της υποκείμενης δομής. Πλέον η Java αποτελεί μια από τις πιο δημοφιλείς γλώσσες

που χρησιμοποιείται για τη δημιουργία εφαρμογών αυτόνομων αλλά και εφαρμογές που εκτελούνται μέσω διαδικτύου.

Υπάρχουν πολλά χαρακτηριστικά που την κάνουν να ξεχωρίζει με τα κυριότερα να είναι τα εξής:

- Είναι πολύ απλή στη χρήση της και αυτό οφείλεται στο ότι οι δημιουργοί της προσπάθησαν να αφήσουν έξω χαρακτηριστικά άλλων γλωσσών προγραμματισμού που δυσκόλευαν το χρήστη.
- Είναι αντικειμενοστρεφής δηλαδή κατά τον προγραμματισμό οργανώνει τον κώδικα σε αντικείμενα. Κάθε αντικείμενο έχει τις δικές του λειτουργίες και τεχνικές και έχει την ιδιότητα να κάνει απόκρυψη ορισμένων λειτουργιών του και να κληρονομεί ιδιότητες από άλλες μονάδες.
- Είναι μεταγλωττιζόμενη δηλαδή παράγει ένα ειδικό κώδικά byte ο οποίος μπορεί να εκτελεστεί από οποιοδήποτε λειτουργικό σύστημα που απλά διαθέτει ένα διερμηνέα java. Επίσης επειδή ο μεταγλωττιστής της java δεν παράγει γηγενή κώδικα προσανατολισμένο σε ένα τύπο υπολογιστή οι εφαρμογές έχουν ανεξαρτησία και στην πλατφόρμα εκτέλεσης.
- Από την αρχή σχεδιάστηκε ώστε να προσφέρει ασφάλεια στην εκτέλεση του κώδικα σε δίκτυο επιτρέποντας την κατασκευή προγραμμάτων ελεύθερων από ιούς και με αδύνατη τροποποίηση τους.
- Υποστηρίζει την πολυνημάτωση, δηλαδή δίνει τη δυνατότητα σε ένα πρόγραμμα να έχει πολλές ξεχωριστές διαδικασίες, οι οποίες να εκτελούνται ταυτόχρονα και ανεξάρτητα η μία από την άλλη.
- Είναι μια δυναμική γλώσσα προγραμματισμού αφού έχει δημιουργηθεί για να μπορεί να προσαρμόζεται σε ένα συνεχώς εξελισσόμενο περιβάλλον. Παρέχει βιβλιοθήκες οι οποίες αναπτύσσονται και προσθέτονται στις ήδη υπάρχουσες δίνοντας τη δυνατότητα για δημιουργία νέων εφαρμογών.
- Τέλος αποτελεί μια γλώσσα προγραμματισμού υψηλής απόδοσης αφού η παραγωγή εντολών μηχανής είναι απλή και γρήγορη, έχοντας ως

συνέπεια η τελική μορφή του κώδικα να είναι μικρή σε μέγεθος και ταχύτατη στην εκτέλεση του.

3.9.2 SQL

Η SQL (Structured Query Language), αποτελεί σήμερα την πιο διαδεδομένη γλώσσα που χρησιμοποιείται για την ανάπτυξη και τη διαχείριση σχεσιακών βάσεων δεδομένων. Η SQL αποτελείται από τη γλώσσα ορισμού δεδομένων (DDL, Data Definition Language) και τη γλώσσα χειρισμού δεδομένων (DML, Data Manipulation Language).

Η γλώσσα ορισμού δεδομένων περιέχει τις εντολές αυτές που επιτρέπουν στο χρήστη να ορίζει και να τροποποιήσει το σχεσιακό του σχήμα αλλά και να δημιουργήσει, να αλλάξει ή να διαγράψει σχέσεις. Συμπεριλαμβάνει επίσης και τις κατάλληλες εντολές για τη δημιουργία και την επεξεργασία όψεων (views). Από την άλλη η γλώσσα χειρισμού δεδομένων περιέχει όλες τις εντολές αυτές που χρειάζεται ο χρήστης για να επεξεργαστεί τα στοιχεία της βάσης του. Σε αυτές περιέχονται εντολές για την εμφάνιση καταχώρηση, τροποποίηση και διαγραφή δεδομένων και σχέσεων.

Τα κυριότερα πλεονεκτήματα της είναι τα εξής:

- Υποστηρίζεται από οποιοδήποτε σχεσιακό σύστημα με συνέπεια η διατύπωση των ερωτημάτων προς τη βάση να είναι ανεξάρτητη από το σύστημα διαχείρισης βάσεων δεδομένων.
- Πλέον πολλές γλώσσες προγραμματισμού έχουν αναπτυχθεί ώστε να μπορούν να υποστηρίζουν τη διατύπωση ερωτημάτων σε SQL.
- Χρησιμοποιείται για όλες τις λειτουργίες του συστήματος διαχείρισης βάσεων δεδομένων.
- Είναι πολύ απλή στη σύνταξη της διευκολύνοντας το χρήστη από τη χρήση της και αποδεσμεύοντας τον από τις τεχνικές λεπτομέρειες υλοποίησης.

Η εφαρμογή δίνει τη δυνατότητα χρήσης δυο συστημάτων ORM, καθιστώντας την ανάπτυξη ανεξάρτητη από τον τύπο της βάσης δεδομένων που χρησιμοποιείται. Είναι δυνατόν, δηλαδή, η εφαρμογή να δουλέψει απρόσκοπτα με διαφορετικούς τύπους βάσεων δεδομένων, αν το σχεσιακό σχήμα κατά την αλλαγή παραμένει το ίδιο. Εντούτοις πρέπει να αναφερθεί ότι η εφαρμογή αναπτύχθηκε χρησιμοποιώντας τη βάση δεδομένων MySQL Community Edition.

3.9.3 HTML5 – WebSocket

Η HTML5 αποτελεί την πέμπτη αναθεώρηση του προτύπου HTML που δημιουργήθηκε το 1990 και έχει τυποποιηθεί ως HTML 4 από το 1997. Η HTML είναι μια γλώσσα σήμανσης που χρησιμοποιείται για την δόμηση και παρουσίαση περιεχομένου για τον παγκόσμιο ιστό και αποτελεί μια βασική τεχνολογία του Διαδικτύου. Στην πέμπτη της έκδοση, ως βασικοί στόχοι τεθήκαν η βελτίωση της γλώσσας με την υποστήριξη για τις τελευταίες τεχνολογίες πολυμέσων και παράλληλα η διατήρησή της σε ένα επίπεδο εύκολα αναγνώσιμο από τον άνθρωπο και κατανοητό από υπολογιστές και συσκευές (προγράμματα περιήγησης στο Web, προγράμματα ανάλυσης, κινητά και tablets κ.λπ.).

Η HTML5 είναι μια απάντηση στο γεγονός ότι η HTML και XHTML σε κοινή χρήση για το World Wide Web είναι ένα μείγμα από χαρακτηριστικά που έχουν εισαχθεί από διάφορες προδιαγραφές, μαζί με εκείνες που εισήγαγαν τα προϊόντα λογισμικού τρίτων εταιριών, όπως οι φυλλομετρητές, αυτά που καθορίζονται από κοινές πρακτικές, και τα διάφορα συντακτικά λάθη στις υπάρχουσες σελίδες του Παγκόσμιου Ιστού. Αποτελεί επίσης μια προσπάθεια να καθοριστεί μια ενιαία γλώσσα σήμανσης που μπορεί να γραφτεί σε μορφή HTML ή XHTML. Περιλαμβάνει αναλυτικά μοντέλα επεξεργασίας για την ενθάρρυνση ανάπτυξης ολοένα και περισσότερο διαλειτουργικών εφαρμογών. Επεκτείνει, βελτιώνει και εξορθολογίζει τη σήμανση για τις σελίδες, και εισάγει ετικέτες σήμανσης για διεπαφές προγραμματισμού εφαρμογών (APIs) για σύνθετες εφαρμογές web. Η HTML5 αποτελεί επίσης υποψήφια τεχνολογία για την ανάπτυξη cross-platform mobile εφαρμογών, καθώς πολλά χαρακτηριστικά

της έχουν κατασκευαστεί έτσι ώστε είναι σε θέση να τρέξει σε χαμηλής ισχύος συσκευές, όπως smartphones και tablets .

Η HTML5 προσθέτει πολλά νέα συντακτικά χαρακτηριστικά αναφορικά με τεχνολογίες πολυμέσων που προωθούν την αντικατάσταση της γενικής ετικέτας <object>. Αυτές περιλαμβάνουν τις νέες ετικέτες <video>, <audio> και τα στοιχεία <canvas>, καθώς και την ενσωμάτωση των κλιμακούμενων διανυσματικών γραφικών (SVG) και MathML για μαθηματικούς τύπους . Τα χαρακτηριστικά αυτά είναι σχεδιασμένα ώστε να καθίσταται εύκολη η ενσωμάτωση και η διαχείριση πολυμεσικού και γραφικού περιεχομένου στο διαδίκτυο, χωρίς οι προγραμματιστές να χρειάζεται να καταφύγουν σε τρίτα ιδιότητα plugins και APIs.

Το WebSocket είναι μια τεχνολογία web που παρέχει ένα πλήρως αμφίδρομο κανάλι επικοινωνίας μέσω μίας μόνο σύνδεσης TCP. Το πρωτόκολλο WebSocket τυποποιήθηκε από το IETF το 2011 και το WebSocket API αναμένεται να τυποποιηθεί από το W3C.

WebSocket έχει σχεδιαστεί για να εφαρμοστεί σε φυλλομετρητές και web servers, αλλά μπορεί να χρησιμοποιηθεί από οποιονδήποτε πελάτη ή εφαρμογή διακομιστή. Η WebSocket πρωτόκολλο αποτελεί ένα ανεξάρτητο TCP πρωτόκολλο του οποίου η σχέση με το HTTP είναι ότι η χειραψία του ερμηνεύεται από τους διακομιστές HTTP ως αίτημα Upgrade. Το WebSocket πρωτόκολλο καθιστά δυνατή μεγαλύτερη αλληλεπίδραση μεταξύ ενός προγράμματος περιήγησης και μια ιστοσελίδα, διευκολύνοντας τη μεταφορά και παρουσίαση περιεχομένου σε πραγματικό χρόνο. Αυτό γίνεται εφικτό μέσω της παροχής ενός τυποποιημένου τρόπου αποστολής περιεχομένου από το διακομιστή στο πρόγραμμα περιήγησης χωρίς να απαιτείται κάποια ενέργεια από τον πελάτη, και με τη διατήρηση μιας ανοικτής σύνδεσης επιτρέποντας επικοινωνία μέσω μηνυμάτων που ταξιδεύουν μπρος και πίσω. Με τον τρόπο αυτό μια αμφίδρομη (bi-directional) συνεχιζόμενη συζήτηση μπορεί να λάβει χώρα μεταξύ ενός προγράμματος περιήγησης και ενός διακομιστή. Ένα

παρόμοιο αποτέλεσμα έχει επιτευχθεί σε μη τυποποιημένο τρόπο, με stop-gap τεχνολογίες όπως το Comet, ή με την τεχνική long-polling.

Επιπλέον, τα μηνύματα μεταφέρονται από την TCP θύρα 80, η οποία είναι ιδανική για αυτά περιβάλλοντα στα οποία συνήθως μπλοκάρονται μη τυποποιημένες συνδέσεις με το Διαδίκτυο χρησιμοποιώντας firewall. Το WebSocket πρωτόκολλο υποστηρίζεται πλέον από πολλούς περιηγητές, όπως το Google Chrome, τον Internet Explorer, το Firefox, το Safari και το Opera.

3.10 Ιδιαιτερότητες, προβλήματα και δυσκολίες

Κατά τη διάρκεια ανάπτυξης της εφαρμογής, παρουσιάστηκαν κάποια προβλήματα και δυσκολίες που έπρεπε να επιλυθούν για την ολοκλήρωσή της. Κάποια από τα προβλήματα αποτελούν ιδιαιτερότητες της επιλεγμένης αρχιτεκτονικής υλοποίησης, ενώ άλλα προέρχονται από τους περιορισμούς που θέτουν οι τεχνολογίες που χρησιμοποιούνται και η πλατφόρμα για την οποία αναπτύσσεται η εφαρμογή.

3.10.1 Web Standards

Όπως αναφέρθηκε προηγουμένως, η διεπαφή του πελάτη με την εφαρμογή έχει υλοποιηθεί με client-side τεχνολογίες και γλώσσες, είναι δηλαδή γραμμένη σε HTML5, CSS και JavaScript. Η HTML5 αποτελεί μια αναθεώρηση της γλώσσας HTML η οποία όμως δεν έχει τυποποιηθεί. Ως εκ τούτου, η υλοποίηση των διαφορετικών στοιχείων της δεν είναι ομοιόμορφη, με τον κάθε φυλλομετρητή να παρέχει τη δική του υλοποίηση για κάθε τεχνολογία ή API της HTML5. Έτσι, αρκετές φορές είναι απαραίτητη η εύρεση workarounds για χαρακτηριστικά και τεχνολογίες της γλώσσας τα οποία δεν έχουν υιοθετηθεί καθολικά

Παράδειγμα ενός τέτοιου χαρακτηριστικού αποτελεί το Audio API της HTML, η διεπαφή, δηλαδή, αναπαραγωγής αρχείων ήχου που δημιουργεί ο text-to-speech μηχανισμός. Η υλοποίηση του Audio API διαφέρει σε κάθε φυλλομετρητή, με ορισμένους να είναι σε θέση να αναπαράγουν λιγότερους τύπους αρχείων ήχου,

ενώ σε μερικές περιπτώσεις παρατηρούνται μέθοδοι που δεν έχουν υλοποιηθεί καθόλου. Ένα άλλο παράδειγμα αποτελεί η υλοποίηση του WebSocket, το οποίο χρησιμοποιείται για να μεταφέρει τα δεδομένα ενός tweet από το server στη σελίδα σε πραγματικό χρόνο. Η υιοθέτηση του WebSocket έχει γίνει σε αρκετά πρόσφατες εκδόσεις των περισσότερων φυλλομετρητών, ενώ σε μερικούς είναι εξ ορισμού απενεργοποιημένο.

Για την αντιμετώπιση προβλημάτων που σχετίζονται με τις διαφορετικές υλοποιήσεις των web standards από τους browsers είναι απαραίτητη η επιπλέον συγγραφή client-side κώδικα που λειτουργεί σαν middleware, παρέχοντας ένα επιπλέον επίπεδο αφαίρεσης στις μεθόδους που χρησιμοποιούν τα APIs αυτά. Στο επίπεδο αυτό γίνεται έλεγχος για τη διαθεσιμότητα των επιθυμητών μεθόδων και χαρακτηριστικών από τον τρέχοντα φυλλομετρητή και επιστρέφονται δεδομένα προσαρμοσμένα στη συμπεριφορά του. Το γεγονός, επίσης, ότι κάποιοι φυλλομετρητές δεν είναι σε θέση να αναπαραγάγουν αρχεία mp3 μέσω της υλοποίησης του Audio API, οδήγησε στη χρήση αρχείων wav για την αναπαραγωγή των tweets.

3.10.2 Rate Limiting

Σχεδόν όλα τα APIs επιβάλλουν κάποια όρια ταχύτητας κλήσεων από εξωτερικές εφαρμογές. Ιδίως τα πολύ δημοφιλή APIs όπως αυτά του Twitter ή του Facebook, τα οποία δέχονται εκατομμύρια αιτήσεις ημερησίως, υιοθετούν τις πρακτικές του rate limiting για να αποφύγουν περιπτώσεις κατάχρησης της διεπαφής από τρίτους. Αν κάποια εφαρμογή ξεπεράσει τα επιτρεπτά όρια κλήσεων στη μονάδα του χρόνου τιμωρείται με προσωρινή απαγόρευση πρόσβασης στο API, ανάλογα με τις πολιτικές της κάθε εταιρείας. Τέτοια APIs παρέχουν συνήθως τη δυνατότητα αύξησης των ορίων με πληρωμένες συνδρομές ή κατόπιν ειδικής συνεννόησης με τους υπεύθυνους των τρίτων εφαρμογών.

Το rate limiting εισάγει ένα επίπεδο δυσκολίας στη ανάπτυξη καθώς είναι εύκολο για μια εφαρμογή να ξεπεράσει το όριο κλήσεων κατά τη διάρκεια των δοκιμών. Η προσωρινή απαγόρευση πρόσβασης που τίθεται σε ισχύ σε αυτές τις περιπτώσεις προσθέτει εμπόδια στην ανάπτυξη, βάζοντας προσωρινό φρένο στη διαδικασία μέχρι την άρση της απαγόρευσης.

3.10.3 Api Versioning

Όπως αναφέρθηκε προηγουμένως, η εφαρμογή ακολουθεί την Service-Oriented αρχιτεκτονική, παρέχοντας μεγάλο μέρος της σαν διακριτά τμήματα λογισμικού τα οποία μπορούν να κληθούν από εξωτερικές εφαρμογές, καθώς και καλώντας η ίδια εξωτερικά APIs για να πάρει δεδομένα. Όταν η υλοποίηση μιας υπηρεσίας, η μορφή των δεδομένων που απαιτεί σαν είσοδο ή η μορφή των δεδομένων που επιστρέφει αλλάξει, η κοινή πρακτική είναι η διατήρηση της παλιάς υλοποίησης και η δημιουργία ενός καινούριου endpoint στο οποίο φιλοξενείται η νέα.

Στα Web APIs, όπου το endpoint είναι το URL στο οποίο βρίσκεται η υπηρεσία, η νέα έκδοση της περιλαμβάνεται στη διεύθυνσή της. Έτσι για παράδειγμα, μια REST υπηρεσία υπό αναβάθμιση, η οποία μπορεί να κληθεί από τη διεύθυνση [http://\[domain\]/api/v1/service](http://[domain]/api/v1/service), μπορεί να συνεχίσει να υπάρχει παράλληλα με τη 2^η έκδοση που θα βρίσκεται στο [http://\[domain\]/api/v2/service](http://[domain]/api/v2/service). Με αυτόν τον τρόπο είναι δυνατή η παροχή της αναβαθμισμένης υπηρεσίας, διατηρώντας παράλληλα και την παλιά για συμβατότητα προς τα πίσω με εφαρμογές που ήδη την χρησιμοποιούν.

Το πρόβλημα συναντάται όταν για διάφορους λόγους μια συγκεκριμένη υλοποίηση εγκαταλείπεται τελείως, παρέχοντας μόνο την καινούρια έκδοση της υπηρεσίας. Σε αυτή την περίπτωση, οι εφαρμογές που επικοινωνούν με τη συγκεκριμένη υλοποίηση δεν μπορούν πια να λειτουργήσουν, κάνοντας απαραίτητη την εκ νέου ανάπτυξη λογισμικού για να προσαρμοστούν στη νέα έκδοση. Μια τέτοια περίπτωση παρουσιάστηκε με το API του Twitter, το οποίο σταμάτησε τελείως τη λειτουργία της έκδοσης 1.0, κάνοντας υποχρεωτική τη

χρήση της έκδοσης 1.1 και τη διόρθωση της εφαρμογής ώστε να είναι σε θέση να λαμβάνει tweets με τις νέες παραμέτρους.

4 Υλοποίηση Συστήματος

4.1 Δομή Βάσης Δεδομένων

Η βάση δεδομένων έχει κατασκευαστεί εξ ολοκλήρου από το ORM του Play Framework και αντιστοιχεί με το Model επίπεδο του MVC. Αναλυτικότερα, οι βασικές οντότητες- πίνακες είναι:

4.1.1 Account

Ο πίνακας αυτός διατηρεί τις πληροφορίες εισόδου του χρήστη στο σύστημα. Ο χρήστης αυθεντικοποιείται χρησιμοποιώντας το λογαριασμό του στο Twitter και τα authentication token και authentication secret που επιστρέφονται από την αλληλεπίδραση με το OAuth αποθηκεύονται στη βάση για να μην είναι απαραίτητη η εκ νέου αυθεντικοποίηση του χρήστη στο μέλλον.

4.1.2 Band

Ο πίνακας αυτός διατηρεί τα radio bands τα οποία δημιουργεί ο χρήστης για να αποθηκεύσει τα keywords που θέλει να ψάξει. Κάθε band έχει ένα μοναδικό αναγνωριστικό, ένα όνομα με το οποίο τη δημιούργησε ο χρήστης, και μια αναφορά στο αναγνωριστικό του χρήστη στον οποίο ανήκει. Επίσης διατηρεί ένα flag που καθορίζει αν το band αυτό χρησιμοποιείται για να φέρνει localized trends από το twitter.

4.1.3 Keyword

Στον πίνακα αυτό αποθηκεύονται πληροφορίες για τις λέξεις κλειδιά τις οποίες αναζητά ο χρήστης. Αποθηκεύεται το όνομα / κείμενο της λέξης, η ημερομηνία δημιουργίας και το αν η λέξη είναι hashtag, αν δηλαδή αρχίζει με το σύμβολο της δίεσης (#) και αναφέρεται σε ένα δημοφιλές topic στο twitter το οποίο οι χρήστες εισάγουν στα tweets τους για να δώσουν επιπλέον σημασία.

4.1.4 Spoken

Ο πίνακας Spoken είναι σχεσιακός. Διατηρεί πληροφορίες για το ποια tweets έχουν ακουστεί ανά keyword και ανα χρήστη, καθώς και την ημερομηνία κατά την οποία ένα tweet «ειπώθηκε»

4.1.5 Tweet

Στο κέντρο του επιπέδου δεδομένων βρίσκεται ο πίνακας Tweet, ο οποίος διατηρεί πληροφορίες για το κείμενο και τον χρήστη που το δημοσίευσε, καθώς και τη διεύθυνση στην οποία είναι αποθηκευμένο το αρχείο ήχου με το οποίο έχει συσχετιστεί. Τέλος, εκτός από την ημερομηνία δημιουργίας, στον πίνακα tweet αποθηκεύεται και η πληροφορία για το αντιλαμβανόμενο, θετικό ή αρνητικό, συναίσθημα που απορρέει από το κείμενο του tweet.

4.1.6 Play Evolutions

Ο πίνακας Play Evolutions δημιουργείται αυτόματα από το ORM του Play Framework. Διατηρεί πληροφορίες σχετικές με το τρέχουν σχήμα της βάσης και αν αυτό έχει αλλάξει, ενώ διατηρεί και αναφορές στα sql scripts που το Play εκτελεί αυτόματα για να αναβαθμίσει τη βάση δεδομένων ώστε να αναποκρίνεται στο επίπεδο των models. Ο πίνακας play_evolutions ελέγχεται κατά την εκκίνηση της εφαρμογής για να διαπιστωθεί κατά πόσο η βάση δεδομένων είναι ενημερωμένη.

4.1.7 Media Result

Στον πίνακα αυτόν αποθηκεύονται τα μουσικά κομμάτια που προσθέτει ο χρήστης στη λίστα αναπαραγωγής. Η αποθήκευση των κομματιών γίνεται ανά χρήση και στη βάση διατηρείται το link από το οποίο γίνεται η αναπαραγωγή.

4.1.8 Σχισιακοί Πίνακες

Επιπλέον υπάρχουν στη βάση πίνακες οι οποίοι εξυπηρετούν το σκοπό δημιουργίας συσχετίσεων μεταξύ των αντικείμενων στο επίπεδο model. Για παράδειγμα, οι πίνακες `tweet_keyword` και `keyword_tweet` συνθέτουν μια πολλά προς πολλά συσχέτιση μεταξύ των μοντέλων `Tweet` και `Keyword`, αφού ένα tweet είναι δυνατό να επιστραφεί από διαφορετικές αναζητήσεις λέξεων-κλειδιών, ενώ μια λέξη κλειδί επιστρέφει, προφανώς, περισσότερα από ένα tweets. Παρομοίως, ο πίνακας `keyword_band` αναφέρεται σε μια ένα προς πολλά σχέση μεταξύ των μοντέλων `band` και `keyword`, αφού σε ένα radio band ο χρήστης μπορεί να προσθέσει έως και 10 λέξεις κλειδιά.

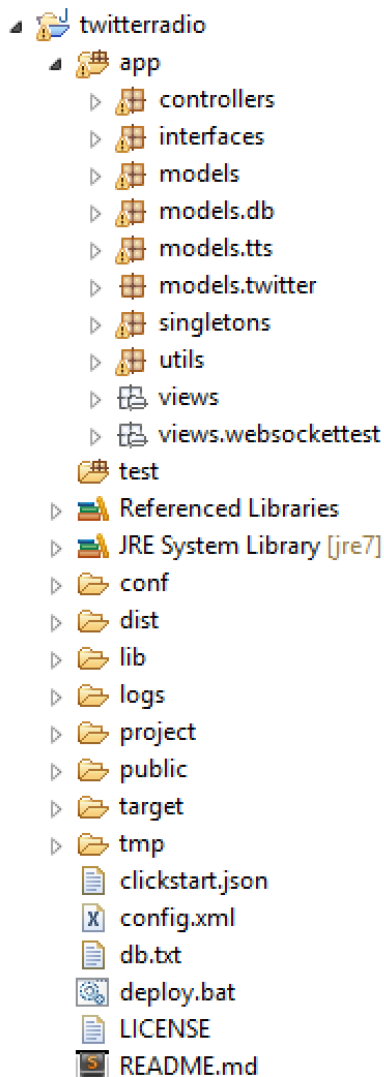
4.2 Δομή Πηγαίου Κώδικα

Η βασική δομή σε ένα Play Framework project περιέχει τους φακέλους /app, μέσα στους οποίους βρίσκεται ο πηγαίος κώδικας όλης της εφαρμογής, δηλαδή τα models, controllers και τα views. Στο φάκελο /conf βρίσκονται τα αρχεία ρυθμίσεων της εφαρμογής, που περιέχουν πληροφορίες για διάφορες λειτουργίες της, όπως τη σύνδεση με τη βάση δεδομένων και τη δρομολόγηση των αιτήσεων στο server (routing) στον αντίστοιχο controller, καθώς και τιμές μεταβλητών περιβάλλοντος. Επίσης στο φάκελο αυτό περιλαμβάνονται και τα αρχεία ρυθμίσεων τυχόν επιπροσθέτων (plugins).

Στον φάκελο /project βρίσκονται οι ρυθμίσεις του διαχειριστή πακέτων. Στο αρχείο build.scala καταγράφονται τα plugins που χρησιμοποιεί η εφαρμογή, καθώς και η τυχόν επιθυμητή έκδοση αυτών, ώστε ο μηχανισμός του play framework να τα κατεβάζει αυτόματα και να τα κάνει διαθέσιμα στην εφαρμογή, ακόμη κι αν τρέχει σε διαφορετικό περιβάλλον από αυτό που αναπτύχθηκε. Τέλος, στο φάκελο αυτό αποθηκεύονται οι compiled κλάσεις που παράγει το play framework και από τις οποίες γίνεται η εκτέλεση.

Τέλος, στο φάκελο public βρίσκονται οποιαδήποτε assets είναι απαραίτητα για την εμφάνιση της εφαρμογής αλλά δεν απαιτούνται για το compilation. Στο φάκελο αυτό τοποθετούνται ουσιαστικά τα public resources που έχουν να κάνουν με τις τεχνολογίες πελάτη, όπως css και javascript αρχεία. Η εφαρμογή TwitterRadio, όταν δε λειτουργεί συνδεδεμένη με κάποια υπηρεσία cloud storage, αποθηκεύει τα παραγόμενα αρχεία ήχου στο φάκελο αυτό.

Κατά την υλοποίηση της εφαρμογής δημιουργήθηκαν κάποιο επιπλέον φάκελοι οι οποίοι δεν δημιουργούνται αυτόματα με ένα play framework project. Στο φάκελο lib βρίσκονται όσες εξωτερικές βιβλιοθήκες δεν ήταν δυνατό να βρει αυτόματα ο διαχειριστής πακέτων του Play, όπως οι βιβλιοθήκες του FreeTTS και η βιβλιοθήκη που χρησιμοποιήθηκε για την επικοινωνία με το twitter, το Twitter4j. Επίσης στο φάκελο dist βρίσκονται τα τελικά αρχεία της εφαρμογής, έτοιμα να μεταφερθούν και να εκτελεστούν σε περιβάλλον παραγωγής.

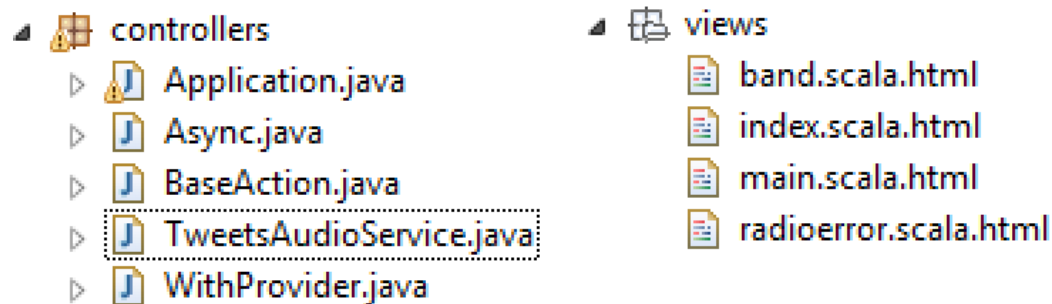


Εικόνα 6: Η δομή του πηγαίνου κώδικα

4.2.1 Controllers και Views

Ακολουθώντας το μοτίβο MVC, τα αρχεία μέσα στο φάκελο app είναι οργανωμένα έτσι ώστε να προσομοιάζουν την αρχιτεκτονική. Στο φάκελο controllers βρίσκονται οι κλάσεις οι οποίες δέχονται τις αιτήσεις των χρηστών στο server, αλληλεπιδρούν με τα δεδομένα των models και επιστρέφουν τα δεδομένα στο view για οπτική διαμόρφωση. Κάθε View αντιστοιχεί σε ένα action (μια μέθοδο) του Application controller, που είναι ουσιαστικά το μόνο controller που παρέχει user interface. Το Async controller παρέχει δεδομένα σε ασύγχρονες κλήσεις από το interface ή σε εξωτερικές εφαρμογές, επιστρέφοντάς τα κωδικοποιημένα σε JSON. Το BaseAction καλείται από τα

υπόλοιπα controllers για να επιβεβαιώσει ότι ο χρήστης είναι συνδεδεμένος ενώ τα άλλα δυο controllers είναι βοηθητικά.

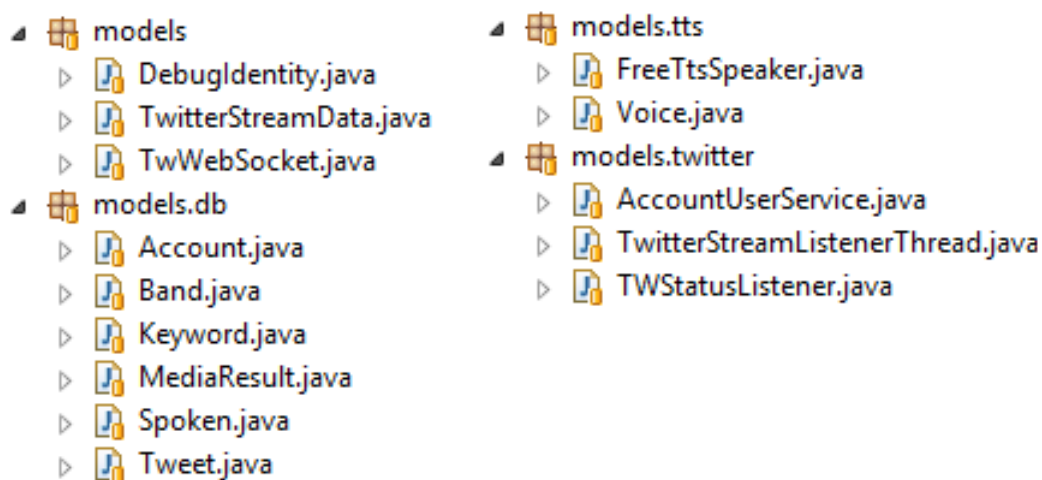


Εικόνα 7: Controllers και views

4.2.2 Models

Όσον αφορά τα models, είναι χωρισμένα σε φακέλους/namespaces, οργανωμένα ανάλογα με τη λειτουργία που επιτελούν και το πλαίσιο στο οποίο ενεργούν. Τα κύρια models, με την κλασική έννοια που απαντάται στο μοτίβο MVC, βρίσκονται στο φάκελο models.db και αποτελούν τις κύριες οντότητες που αποθηκεύονται στη βάση δεδομένων. Τα μοντέλα που βρίσκονται στο φάκελο models χρησιμοποιούνται από controllers αλλά αποτελούν οντότητες δεν έχουν παρουσία στη βάση. Παρόλα αυτά κληρονομούν από την κλάση Model καθώς χρειάζονται απαραίτητες βοηθητικές μεθόδους και interfaces του Play Framework για λειτουργίες όπως το JSON Serialization.

Στο φάκελο models.tts βρίσκονται οι κλάσεις που συμμετέχουν στη μετατροπή των tweets σε ήχο. Η κλάση Voice αποτελεί ένα wrapper για τις φωνές που παρέχει το FreeTTS ενώ η FreeTtsSpeaker πραγματοποιεί την text-to-speech μετατροπή. Τέλος, οι κλάσεις που βρίσκονται στο φάκελο models.twitter χειρίζονται την αλληλεπίδραση με το API του Twitter. Το TwitterStreamListenerThread αποτελεί το μοναδικό ανα χρήση thread που αρχικοποιεί έναν TWStatusListener για να φέρει τα tweets. Τέλος, η κλάση AccountUserService διαχειρίζεται τα tokens του χρήστη που χρειάζονται για την αυθεντικοποίηση στο Twitter και έχουν αποθηκευτεί τοπικά.



Εικόνα 8: Επισκόπηση των μοντέλων

















4.2.3 Λοιπές Οντότητες

Για την ανάπτυξη της εφαρμογής χρησιμοποιήθηκαν, επίσης, κλάσεις που αρχικοποιούνται μια φορά στον κύκλο ζωής της εφαρμογής και οποιαδήποτε μετέπειτα αλληλεπίδραση με αυτές γίνεται με τη στατική αναφορά του μοναδικού object που υπάρχει στη μνήμη, σύμφωνα με το προγραμματιστικό μοτίβο του singleton. Τέτοιες κλάσεις αποτελούν συνήθως managers που διατηρούν στην προσωρινή μνήμη αντικείμενα τα οποία είναι θεμιτό να καταστρέφονται με το κλείσιμο της εφαρμογής.

Για παράδειγμα, ο AccountManager διατηρεί στη μνήμη τους χρήστες που έχουν αυθεντικοποιηθεί στην εφαρμογή. Ο BandManager διατηρεί τις μπάντες κάθε χρήστη και είναι υπεύθυνος για τη διατήρηση και την ανανέωση της μπάντας με τα trends, ενώ ο WebSocketManager διατηρεί στη μνήμη ένα Web Socket για κάθε χρήση, μέσω του οποίου γίνεται η σε πραγματικό χρόνο επικοινωνία με το φυλλομετρητή του. Τέλος, ο TtsBroker ανιχνεύει και αρχικοποιεί τις διαθέσιμες μεθόδους μετατροπής text-to-speech, ενώ αποτελεί και ένα επίπεδο αφαίρεσης μέσω του οποίου γίνονται όλες οι αιτήσεις για μετατροπή.

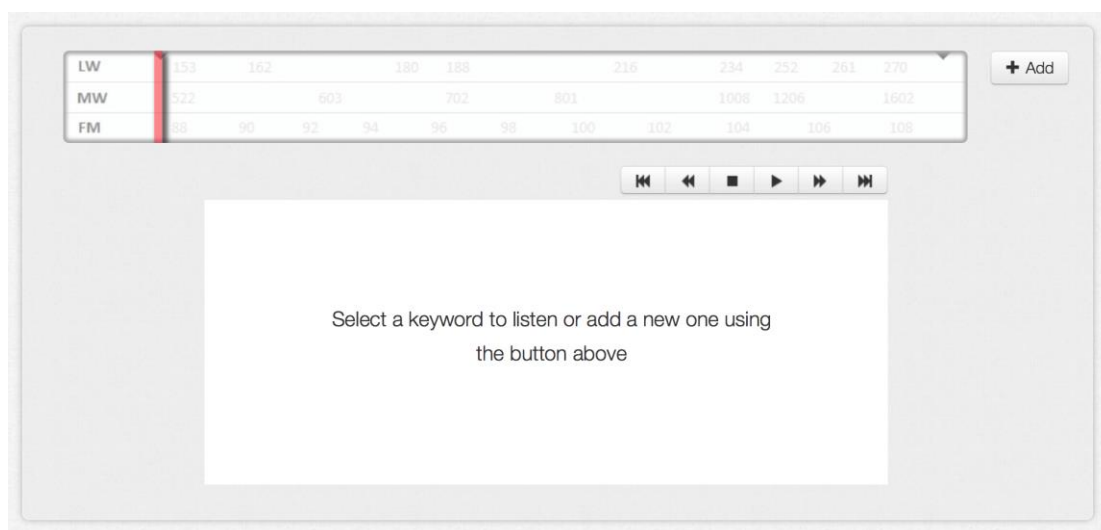
Τέλος, αναπτύχθηκαν και κάποιες βοηθητικές κλάσεις οι οποίες περιέχουν αποκλειστικά στατικές μεθόδους που αποτελούν επαναχρησιμοποιήσιμα blocks

κώδικα, καθώς και σταθερά readonly μέλη. Οι κλάσεις αυτές επιτελούν λειτουργίες όπως η εκτύπωση στη γραμμή εντολών (command line) και η δήλωση σταθερών μεταβλητών που χρησιμοποιούνται στο user interface.

- ▲  singletons
 - ▶  AccountManager.java
 - ▶  BandManager.java
 - ▶  MediaManager.java
 - ▶  SentimentManager.java
 - ▶  SessionBroker.java
 - ▶  TtsBroker.java
 - ▶  TwitterBroker.java
 - ▶  WebSocketManager.java
- ▲  utils
 - ▶  Global.java
 - ▶  InMemoryUserService.java
 - ▶  Reflector.java
 - ▶  TweetRequest.java
 - ▶  U.java
 - ▶  UI.java

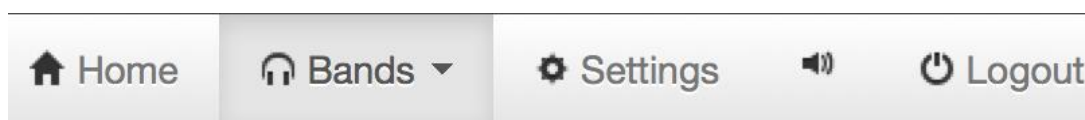
5 Παρουσίαση Συστήματος

Η βασική λειτουργία της εφαρμογής είναι η φωνητική μετατροπή tweets μέσω μηχανισμού text-to-speech σε ήχο και η αναπαραγωγή του, παίζοντας παράλληλα μουσική στο παρασκήνιο. Η λειτουργικότητα αυτή προσομοιάζει τη λειτουργία του ραδιοφώνου, γι αυτό και το user interface έχει κατασκευαστεί ώστε να θυμίζει παλιό αναλογικό ραδιόφωνο με βελόνα επιλογής συχνότητας.

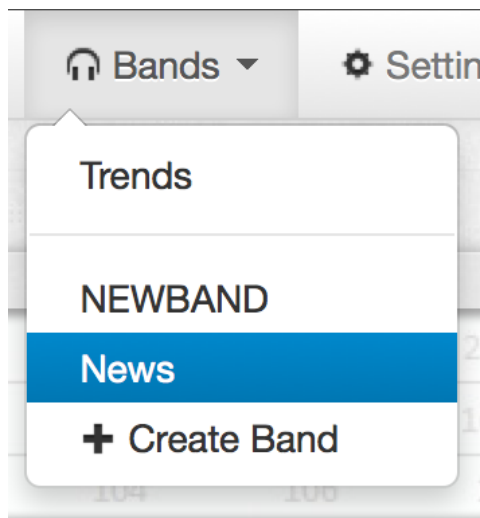


Εικόνα 9: Το κεντρικό user interface

5.1 Το κεντρικό μενού

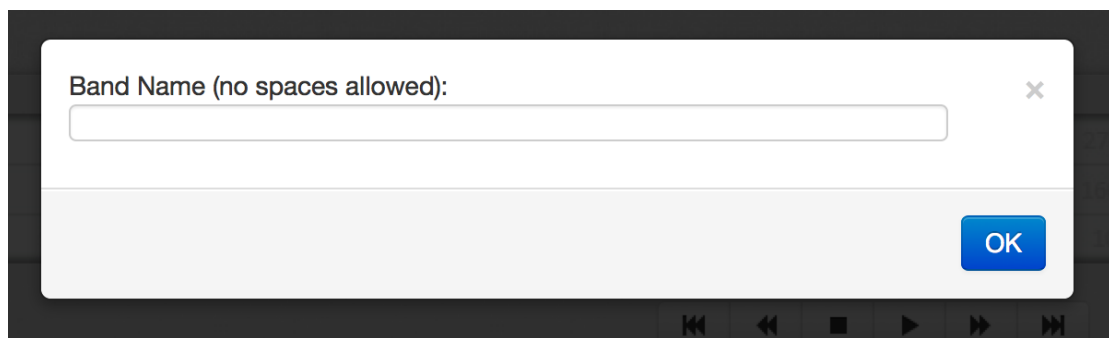


Το κεντρικό μενού αποτελείται από πέντε κύρια κουμπιά. Το κουμπί Home μεταφέρει το χρήστη στην αρχική σελίδα της εφαρμογής. Το κουμπί Bands ανοίγει ένα υπομενού με τις ραδιοφωνικές μπάντες που έχει δημιουργήσει ο χρήστης. Οι μπάντες του χρήστη είναι οργανωμένες συλλογές από λέξεις-κλειδιά και αποτελούν την κύρια οντότητα που χρησιμοποιεί ο χρήστης κατά την αλληλεπίδρασή του με την εφαρμογή. Η πρώτη μπάντα στο μενού (Trends) δημιουργείται αυτόματα από την εφαρμογή και περιέχει τις λέξεις κλειδιά που παρουσιάζουν μεγάλο ενδιαφέρον τη συγκεκριμένη στιγμή.



Εικόνα 10: Υπομενού Band

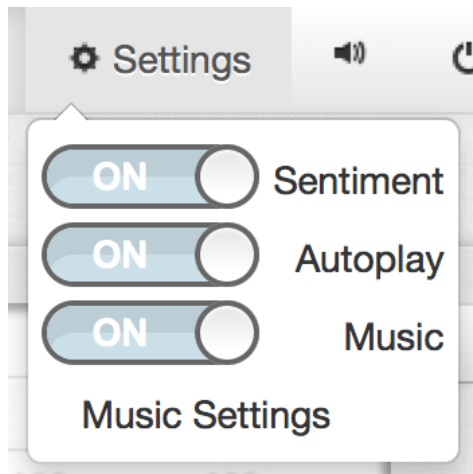
Όταν ο χρήστης επιλέξει μια μπάντα από αυτές που έχει ήδη φτιάξει, μεταφέρεται στην οθόνη αναπαραγωγής tweets και οι λέξεις κλειδιά που περιέχονται στη μπάντα τοποθετούνται πάνω στο αναλογικό ραδιόφωνο σαν συχνότητες. Στην περίπτωση που ο χρήστης επιλέξει να δημιουργήσει μια νέα μπάντα, το σύστημα θα του παρουσιάσει ένα παράθυρο διαλόγου για να συμπληρώσει το όνομα της μπάντας και κατόπιν θα τον μεταφέρει στην οθόνη αναπαραγωγής tweets.



Εικόνα 11: Παράθυρο διαλόγου νέας μπάντας

Το μενού settings περιέχει κάποιες στοιχειώδεις επιλογές παραμετροποίησης της εφαρμογής. Ο διακόπτης Sentiment ενεργοποιεί ή απενεργοποιεί τη χρήση χρωματισμών για την παρουσίαση ενός tweet σαν θετικό ή αρνητικό. Κατά την απενεργοποίηση όλα τα tweets χρωματίζονται γκρι ενώ με την ενεργοποίηση

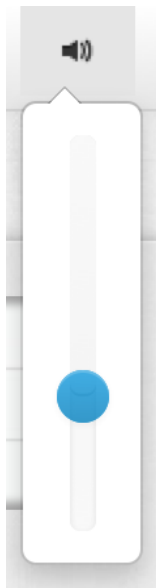
επανέρχονται στο χρώμα που τους αποδόθηκε από το μηχανισμό sentiment analysis.



Εικόνα 12: Το μενού ρυθμίσεων

Ο διακόπτης Autoplay ορίζει αν η αναπαραγωγή των tweets αρχίζει αυτόματα με την άφιξη του πρώτου tweet στον πελάτη. Με την απενεργοποίηση το σύστημα απαιτεί από το χρήστη να πατήσει το κουμπί play για να αρχίσει την αναπαραγωγή. Ο διακόπτης music ενεργοποιεί ή απενεργοποιεί την αναπαραγωγή μουσικής στο παρασκήνιο. Η τελευταία επιλογή “Music Settings” αφορά τη δημιουργία λίστας αναπαραγωγής μουσικής και αναλύεται στην ενότητα 5.3

Το τέταρτο κουμπί εμφανίζει σαν υπομενού ένα fader ο οποίος ελέγχει την ένταση της αναπαραγωγής μουσικής.



Εικόνα 13: Fader καθορισμού έντασης μουσικής

Τέλος, το πέμπτο κουμπί με την ένδειξη “Logout” πραγματοποιεί έξοδο του χρήστη από το σύστημα, στο οποίο έχει κάνει login με το Twitter account του.

5.2 Η αναπαραγωγή tweets

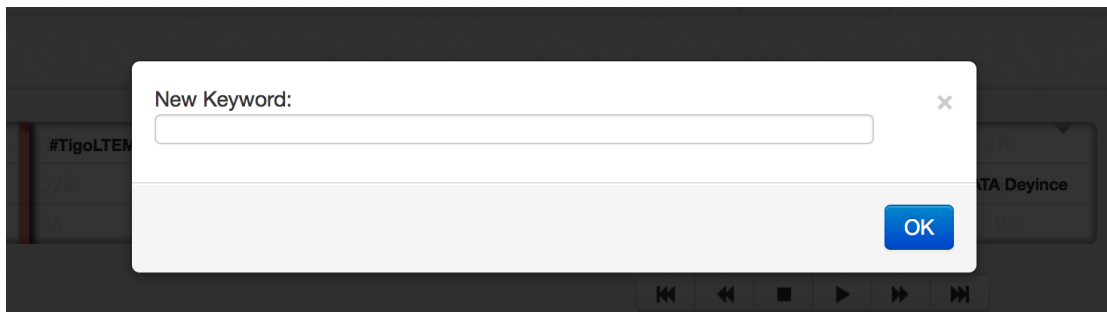
Για την εκκίνηση της αναπαραγωγής των tweets απαιτείται η εισαγωγή από το χρήστη μιας λέξης-κλειδί βάσει της οποίας θα γίνει η αναζήτηση στο Twitter. Οι λέξεις κλειδιά παρουσιάζονται στο user interface σαν συχνότητες πάνω σε ένα αναλογικό ραδιόφωνο.

A screenshot of a frequency tuner interface. It features a horizontal axis with a red vertical line indicating the current frequency. The axis is labeled with various keywords and their corresponding frequencies. A '+ Add' button is visible on the right side.

LW	#MileyCyrus 62	Beyonce 88	216	234	252	261	270	+ Add				
MW	522	MichaelJackson	702	801	1008	1206	1602					
FM	88	90	92	Pharell	96	98	100	102	104	106	108	

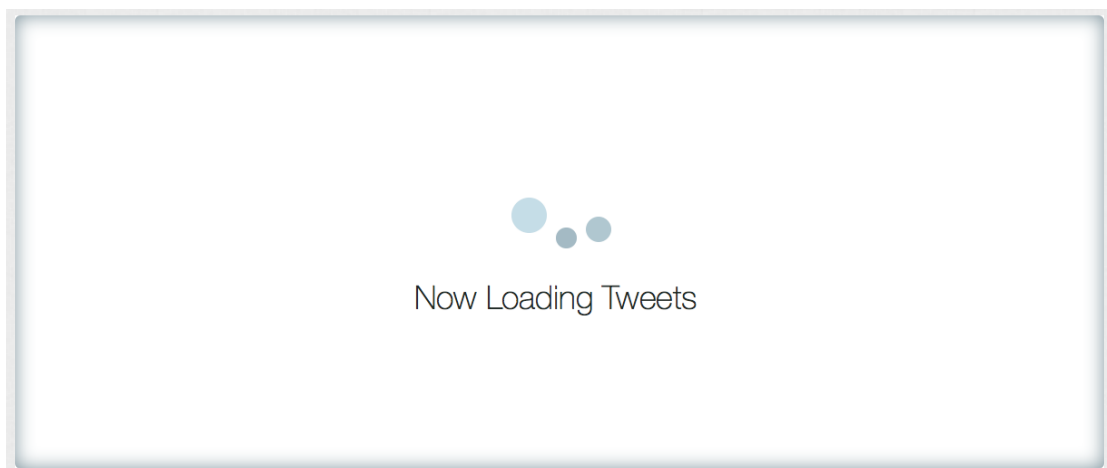
Εικόνα 14: Εμφάνιση των keywords

Όταν ο χρήστης πραγματοποιήσει αναζήτηση για μια λέξη-κλειδί, η εφαρμογή επικοινωνεί με το Twitter και δέχεται tweets τα οποία περνάει από το μηχανισμό text-to-speech και τα μετατρέπει σε αρχεία ήχου που αποθηκεύονται στο server. Ο χρήστης μπορεί να πραγματοποιήσει αναζήτηση πατώντας πάνω σε μια λέξη κλειδί από τις ήδη υπάρχουσες ή να πατήσει το κουμπί Add για να προσθέσει μια καινούρια.



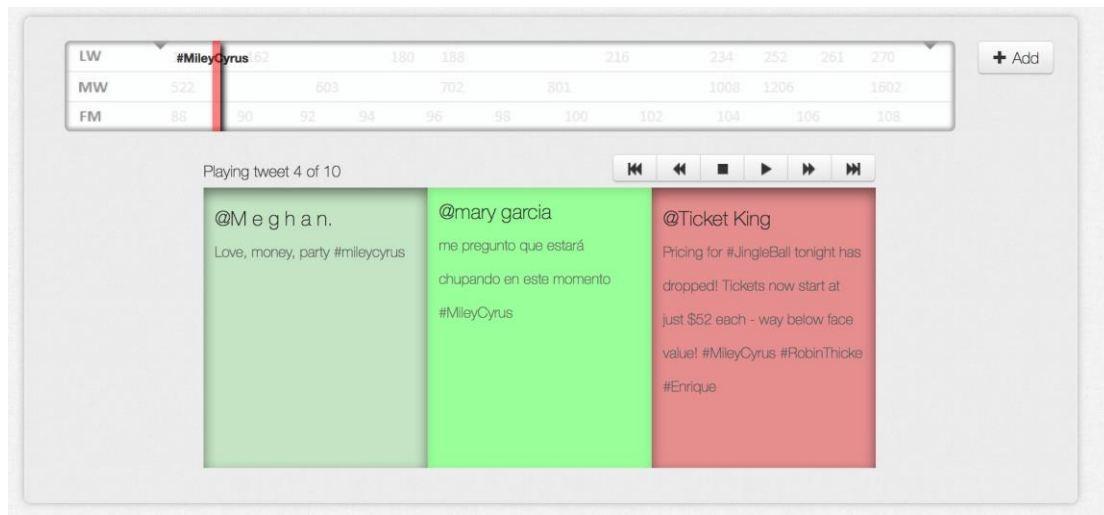
Εικόνα 15: Παράθυρο διαλόγου νέας λέξης-κλειδί

Όταν ο χρήστης επιλέξει τη νέα λέξη-κλειδί, η λέξη προστίθεται στο interface και η κόκκινη βελόνα μετακινείται πάνω της, εμφανίζοντας παράλληλα μήνυμα αναμονής.



Εικόνα 16: Μήνυμα αναμονής

Κατόπιν τα tweets περνάνε από το μηχανισμό ανάλυσης συναισθήματος (sentiment analysis) και χρωματίζονται κατάλληλα. Ένα tweet μπορεί να χαρακτηριστεί «θετικό» ή «αρνητικό» και συνοδεύεται από μια αριθμητική τιμή που συμβολίζει το βαθμό confidence της επιλογής αυτής. Έτσι, τα θετικά tweets χρωματίζονται πράσινα ενώ τα αρνητικά κόκκινα, λαμβάνοντας υπόψιν και τον βαθμό confidence για την επιλογή των ενδιάμεσων αποχρώσεων. Για παράδειγμα, ένα tweet που έχει χαρακτηριστεί θετικό με confidence 0.8 θα χρωματιστεί με πιο «ζωντανό» πράσινο από ένα άλλο που έχει χαρακτηριστεί θετικό με confidence 0.2.



Εικόνα 17: Διάφορα tweets με διαφορετικό sentiment

Η αναπαραγωγή των tweets αρχίζει αυτόματα, αν είναι ενεργοποιημένη από το μενού η σχετική ρύθμιση. Όταν η αναπαραγωγή ενός tweet τελειώσει αρχίζει αυτόματα η αναπαραγωγή του επομένου, εκτός αν πρόκειται για το τελευταίο tweet, οπότε η αναπαραγωγή σταματάει. Τα tweets που εμφανίζονται ταυτόχρονα κάθε φορά είναι τρία, ενώ κάθε φορά που τελιώνει η αναπαραγωγή ενός tweet ή πατιέται το κουμπί «επόμενο» πραγματοποιείται κύλιση όλων των tweets προς τα αριστερά ώστε να φανούν στην εικόνα και τα υπόλοιπα. Η ροή λήψης νέων tweets τροφοδοτείται από το web socket και είναι συνεχής.

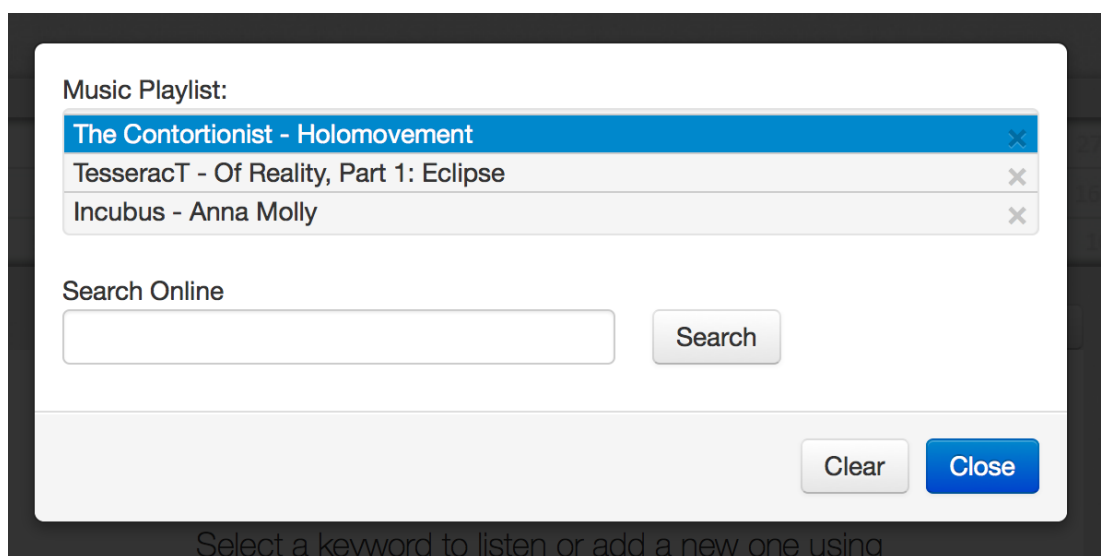
Ο χρήστης έχει συγχρόνως έλεγχο στην αναπαραγωγή μέσω των καθιερωμένων κουμπιών αναπαραγωγής. Στο χρήστη δίνονται οι επιλογές αναπαραγωγής, διακοπής, μετάβασης ένα tweet μπροστά ή πίσω και μετάβαση στο πρώτο ή το τελευταίο tweet.



Εικόνα 18: Τα κουμπιά ελέγχου αναπαραγωγής

5.3 Αναπαραγωγή μουσικής

Κάνοντας κλικ στην επιλογή “Music Settings” από το μενού Settings ο χρήστης μπορεί να δει τη λίστα αναπαραγωγής μουσικής. Το παράθυρο διαλόγου που εμφανίζεται δείχνει τα κομμάτια που έχει προσθέσει ο χρήστης στη λίστα του και τα οποία αναπαράγονται παράλληλα με τα tweets.



Εικόνα 19: Η διαχείριση της λίστας αναπαραγωγής

Το παράθυρο ρυθμίσεων παρέχει κάποιες βασικές λειτουργίες πάνω στη λίστα αναπαραγωγής. Ο χρήστης μπορεί να κάνει αναζήτηση μουσικών κομματιών πληκτρολογώντας τις επιθυμητές λέξεις κλειδιά στο πεδίο Search Online. Τα δεδομένα εξετάζονται από το TinySong API και η απάντηση τροφοδοτείται στο API του YouTube για να αποκτήσουμε τη διεύθυνση στην οποία βρίσκεται το τραγούδι.

Ο χρήστης μπορεί επίσης να διαγράψει επιλεγμένα τραγούδια, κάνοντας κλικ στο X που βρίσκεται δεξιά στο τραγούδι που επιθυμεί να διαγράψει. Μπορεί επίσης να διαγράψει όλα τα κομμάτια που έχει προσθέσει, κάνοντας κλικ στο κουμπί Clear και καθαρίζοντας εντελώς τη λίστα αναπαραγωγής. Τέλος, μπορεί να μεταβεί απευθείας στο κομμάτι της επιλογής του, παρακάμπτοντας όλα τα ενδιάμεσα, κάνοντας κλικ πάνω του.

Η εφαρμογή χρησιμοποιεί ένα μηχανισμό που τρέχει ανα σύντομα διαστήματα μερικών δευτερολέπτων και ελέγχει τον αριθμό των κομματιών της λίστας που απομένουν. Εάν ο αριθμός αυτός είναι μικρός, επιχειρεί να προσθέσει αυτόματα ένα κομμάτι στη λίστα, ώστε να συνεχίσει απρόσκοπτα η αναπαραγωγή. Για την αυτόματη επιλογή κομματιού λαμβάνονται υπ όψιν τα τελευταία τραγούδια που άκουσε ο χρήστης και με τη βοήθεια του API του LastFM βρίσκει παρόμοια τραγούδια και προσθέτει στη λίστα ένα από αυτά.

6 Συμπεράσματα – Μελλοντικές Προεκτάσεις

Η παρούσα εργασία καλύπτει τις πιο βασικές λειτουργίες μιας εφαρμογής που προσομοιώνει την αδιάκοπη ροή ενός ραδιοφωνικού σταθμού, αντλώντας περιεχόμενο από ένα δημοφιλές κοινωνικό δίκτυο. Σίγουρα όμως θα μπορούσαν να υπάρξουν βελτιώσεις που θα αύξαναν την αποδοτικότητα και τη χρησιμότητά της.

6.1 Βελτίωση Μηχανισμού Text-To-Speech

Η βιβλιοθήκη FreeTTS είναι αναμφίβολα μια από τις καλύτερες open source text-to-speech βιβλιοθήκες. Είναι εύκολη στη χρήση, παράγει ήχο σχετικά υψηλής ποιότητας, είναι επεκτάσιμη επιτρέποντας και την εισαγωγή mp3 φωνών και το αντιλαμβανόμενο αποτέλεσμα της μετατροπής είναι σίγουρα καλύτερο από πιο ώριμες open source.

Αρκετές φορές όμως φαίνεται να παράγει μη αποδεκτά και μη αντιληπτά αποτελέσματα. Επίσης, η έρευνα στον τομέα της φωνητικής σύνθεσης έχει προχωρήσει αρκετά, με εμπορικές λύσεις και βιβλιοθήκες να παράγουν εντυπωσιακά αποτελέσματα κατά τη μετατροπή κειμένου σε ήχο. Πολύ καλή υλοποίηση, επίσης, αποτελεί το Web Speech API της Google, το οποίο αν και παρέχεται δωρεάν επιβάλλει ένα μέγιστο όριο 100 χαρακτήρων. Για την υιοθέτηση κάποιων από τις παραπάνω λύσεις απαιτείται η αγορά λογισμικού ή η πληρωμένη συνδρομή.

Δεδομένης της υψηλής ποιότητας των αποτελεσμάτων, ίσως θα έπρεπε να εξεταστούν και εμπορικές επιλογές για την οριστική υιοθέτηση κάποιου πιο αξιόπιστου μηχανισμού text-to-speech.

6.2 Αναγνώριση Γλώσσας

Για τους σκοπούς της υλοποίησης, κυρίως επειδή οι περισσότερες open source text-to-speech βιβλιοθήκες υποστηρίζουν μόνο την αγγλική γλώσσα, έγινε η παραδοχή ότι τα tweets που θα μετατρέπονται είναι αποκλειτικά τα αγγλικά. Μια σημαντική εξέλιξη, λοιπόν, της εφαρμογής, θα ήταν η δυνατότητα για αρχή να αναγνωρίζει και κατόπιν να μετατρέπει και άλλες γλώσσες, πέρα από τα αγγλικά.

Ήδη υπάρχουν αρκετά ανοιχτά APIs που παρέχουν δωρεάν αναγνώριση γλώσσας για δοθέν κείμενο, και η ιδέα της παραχώρησης αυτής της λειτουργικότητας σε εξωτερικό API ταιριάζει με τη Mashup φιλοσοφία που ακολουθεί η εφαρμογή. Βέβαια, απαιτείται, εκτός από την αναγνώριση, και η υποστήριξη της γλώσσας από το μηχανισμό text-to-speech, οπότε αυτή η πρόταση σχετίζεται άμεσα με την προηγούμενη.

6.3 Native Mobile Application

Ο αριθμός των mobile συσκευών τείνει πλέον να ξεπεράσει τον αριθμό επιτραπέζιων και φορητών υπολογιστών, γι αυτό και πρώτη μέριμνα των σύγχρονων εφαρμογών είναι η δημιουργία application για κάθε διαθέσιμη πλατφόρμα. Όλα τα κινητά πλέον έχουν HTML5 φυλλομετρητές με τους οποίους είναι συμβατή η εφαρμογή, αλλά μια native εφαρμογή παρέχει επιπλέον δυνατότητες και αφαιρεί της ιδιαιτερότητες του web, ενώ παρέχει πιο αποτελεσματικές τεχνικές ανάπτυξης για εφαρμογές που κάνουν χρήση media streaming.

6.4 Εξειδικευμένο μουσικό API

Ένα άλλο χαρακτηριστικό που θα μπορούσε να βελτιωθεί είναι ο τρόπος με τον οποίο γίνεται η αναπαραγωγή μουσικής και με τον οποίο προτείνονται καινούρια κομμάτια στο χρήστη.

Η αναπαραγωγή μουσικής από το YouTube έχει πάντα τον κίνδυνο να παρεμβληθεί ενδιάμεσα κάποια διαφήμιση, πράγμα που θα έχει αρνητικές επιπτώσεις στη εμπειρία του χρήστη με την εφαρμογή. Επίσης, το YouTube δεν είναι μια μουσική υπηρεσία, πράγμα που σημαίνει ότι στα αποτελέσματα της αναζήτησης για ένα κομμάτι μπορεί να συμπεριλάβει ζωντανές και μη αυθεντικές εκτελέσεις, καθώς και μη σχετικά βίντεο.

Υπάρχουν διάφορες μουσικές υπηρεσίες, όπως το GrooveShark και το Spotify API τα οποία παρέχουν, εκτός από αφοσιωμένες υποδομές παροχής τραγουδιών για streaming, αναλυτικά metadata και προχωρημένες υπηρεσίες εύρεσης σχετικής μουσικής βασισμένη σε τραγούδια, δημιουργούς ή άλλες παραμέτρους. Έτσι θα ήταν δυνατή η δημιουργία μιας λίστας αναπαραγωγής για την εφαρμογή η οποία θα λάμβανε υπο όψιν, όχι μόνο τα τελευταία κομμάτια που άκουσε ο χρήστης, αλλά και τα τελευταία tweets καθώς και το αν ήταν θετικά ή αρνητικά.

I. Βιβλιογραφία

a. Βιβλία

1. Matthew A. Russell (2011) - *Mining the Social Web*, O'Reilly Media
2. Ian F. Darwin (2001) - *Java Cookbook*, O'Reilly Media
3. Andy Petrella (2013) - *Learning Play! Framework 2*, Packt Publishing
4. Peter Lubbers, Brian Albers, Frank Salim (2010) - *Pro HTML5 Programming*, Apress

b. Ιστοσελίδες

1. <http://www.playframework.com/documentation/2.1.x/Home>
2. <http://freetts.sourceforge.net/docs/index.php>
3. <http://tcts.fpms.ac.be/synthesis/>
4. http://en.wikipedia.org/wiki/Service-oriented_architecture
5. http://en.wikipedia.org/wiki/Representational_state_transfer
6. <http://www.websocket.org>
7. <http://blog.simcrest.com/what-is-3-tier-architecture-and-why-do-you-need-it/>
8. <http://www.stackoverflow.com>
9. http://en.wikipedia.org/wiki/Speech_synthesis