



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
**ΑΛΛΑΞΕ ΔΙΑΘΕΣΗ! ΔΗΜΙΟΥΡΓΙΑ VIDEO
MUSIC PLAYLIST ΒΑΣΙΣΜΕΝΗ ΣΤΟ
ΣΥΝΑΙΣΘΗΜΑ**
ΔΕΛΗΠΑΛΤΑ ΙΩΑΝΝΑ
ΜΠΟΥΡΜΠΟΥΛΑΣ ΗΛΙΑΣ

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: ΑΘΑΝΑΣΙΟΣ ΚΟΥΤΡΑΣ

ΠΥΡΓΟΣ, 2018

ΠΙΣΤΟΠΟΙΗΣΗ

Πιστοποιείται ότι η πτυχιακή εργασία με θέμα:

«ΑΛΛΑΞΕ ΔΙΑΘΕΣΗ! ΔΗΜΙΟΥΡΓΙΑ VIDEO MUSIC PLAYLIST ΒΑΣΙΣΜΕΝΗ ΣΤΟ ΣΥΝΑΙΣΘΗΜΑ»

των φοιτητών του Τμήματος ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ

ΙΩΑΝΝΑ ΔΕΛΗΠΑΛΤΑ

ΗΛΙΑΣ ΜΠΟΥΡΜΠΟΥΛΑΣ

παρουσιάστηκε δημόσια και εξετάστηκε στο Τμήμα ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ
στις

15 / 10/ 2018

Ο ΕΠΙΒΛΕΠΩΝ

Ο ΠΡΟΕΔΡΟΣ ΤΟΥ ΤΜΗΜΑΤΟΣ

ΑΘΑΝΑΣΙΟΣ ΚΟΥΤΡΑΣ

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΜΗ ΛΟΓΟΚΛΟΠΗΣ

Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Ακόμα δηλώνω ότι αυτή η γραπτή εργασία προετοιμάστηκε από εμένα προσωπικά και αποκλειστικά και ειδικά για την συγκεκριμένη πτυχιακή εργασία και ότι θα αναλάβω πλήρως τις συνέπειες εάν η εργασία αυτή αποδειχθεί ότι δεν μου ανήκει.

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 1

ΑΜ

ΥΠΟΓΡΑΦΗ

.....
ΙΩΑΝΝΑ ΔΕΛΗΡΑΛΤΑ.....

.....
1150.....

.....


ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 2

ΑΜ

ΥΠΟΓΡΑΦΗ

(σε περίπτωση που είναι απαραίτητο)

.....
ΗΛΙΑΣ ΜΠΟΥΡΜΕΤΣΑΣ 1198.....

.....


ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία εκπονήθηκε από τον φοιτητή Ηλία Μπουρμπούλα και τη φοιτήτρια Ιωάννα Δελήπαλτα του τμήματος Πληροφορικής και ΜΜΕ του ΤΕΙ Δυτικής Ελλάδος, παραρτήματος Πύργου υπό την επίβλεψη του επίκ. καθηγητή του τμήματος Δρ. Αθανάσιου Κούτρα.

Στον κύριο Κούτρα οφείλουμε τις θερμές μας ευχαριστίες για την καθοδήγηση και την υποστήριξη του καθόλη τη διάρκεια εκπόνησης της παρούσας πτυχιακής.

Τέλος ένα μεγάλο και εγκάρδιο ευχαριστώ αξίζουν οι γονείς μας για την υπομονή που υπέδειξαν και την στήριξη τους ηθικά και οικονομικά όλα αυτά τα χρόνια.

ΠΡΟΛΟΓΟΣ

Η παρούσα πτυχιακή εργασία εκπονήθηκε στο τμήμα Πληροφορικής και ΜΜΕ του ΤΕΙ Δυτικής Ελλάδος, παραρτήματος Πύργου και φέρει τον τίτλο "Άλλαξε διάθεση! Δημιουργία Video music playlist βασισμένη στο συναίσθημα". Η έμπνευση του θέματος προήλθε σε συνεργασία με τον υπεύθυνο καθηγητή, Δρ. Αθανάσιο Κούτρα, και αφορούσε την ανάπτυξη διαδικτυακής εφαρμογής που θα βασιζόταν στην υπάρχουσα εκπονηθείσα πτυχιακή με τίτλο "Ανάπτυξη συστήματος αυτόματης αναγνώρισης συναισθημάτων από μουσικά αρχεία" της τότε προπτυχιακής φοιτήτριας Βασιλικής Σασλίδη.

Σκοπός ήταν, έχοντας τα δεδομένα της προαναφερθείσας πτυχιακής εργασίας, η δημιουργία μιας διαδικτυακής εφαρμογής κατά την οποία θα εξελισσόταν και το σύστημα αυτόματης αναγνώρισης που είχε δημιουργηθεί από την κ.Σασλίδη.

Καθώς κατέστη αδύνατον να αντληθούν τα δεδομένα αυτά, η πτυχιακή μας άλλαξε ελαφρώς μορφή. Πλέον, η κατηγοριοποίηση των μουσικών αρχείων έγινε μέσα από έρευνα σε διαδικτυακές πλατφόρμες. Επίσης, ο νέος μας στόχος, δεν είναι η εξέλιξη του συστήματος, αλλά η δημιουργία μίας προσωποποιημένης λίστας τραγουδιών για τον εκάστοτε χρήστη με βάση πάντα το συναίσθημα που επιλέγει.

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία ασχολείται με την δημιουργία μιας διαδικτυακής εφαρμογής σχετιζόμενης με τη μουσική και τα συναισθήματα. Σκοπός της είναι με τη χρήση γλωσσών προγραμματισμού όπως HTML, PHP, Javascript, Angular, Typescript, αλλά και εφαρμογές όπως το NodeJS και το Firebase να παρέχει στο χρήστη τη δυνατότητα να ακούσει μουσική με βάση το συναίσθημα που επιλέγει και σε δεύτερο χρόνο να μπορεί να κατηγοριοποιήσει τα τραγούδια με βάση τη δική του άποψη για το ποιο συναίσθημα του προκαλούν.

Το παρόν βιβλίο χωρίζεται σε τρία μέρη. Ξεκινώντας, βλέπουμε την εξέλιξη της μουσικής, της επαφής του ανθρώπου με αυτήν και όλα αυτά σε συνάρτηση με την εξέλιξη της τεχνολογίας.

Στο δεύτερο μέρος, αναφερόμαστε στα βασικά στοιχεία μερικών από τις πιο σημαντικές γλώσσες προγραμματισμού διαδικτυακών εφαρμογών και τέλος στο τρίτο και τελευταίο μέρος δείχνουμε τον κώδικα που συγγράψαμε και γίνεται η επεξήγηση του.

ABSTRACT

This diploma thesis deals with the creation of an online application related to music and emotions. Its purpose is to use programming languages such as HTML, PHP, Javascript, Angular, Typescript, and applications such as NodeJS and Firebase to give the user the ability to listen to music based on the emotion they choose and to be able to classify songs based on his own view of what emotion they are causing.

This book is divided into three parts. First, we see the evolution of music, how human connects with music, and all these alongside the evolution of technology.

In the second part, we mention the basic elements of some of the most important programming languages for web applications, and finally, in the third and last part we show the code we have written in order to create the application and how it works.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ

Web development, HTML, PHP, Javascript, Angular, Typescript, NodeJS, Firebase

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ	4
ΠΡΟΛΟΓΟΣ	5
ΠΕΡΙΛΗΨΗ	6
ABSTRACT	6
ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ	6
ΠΕΡΙΕΧΟΜΕΝΑ	7
ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ	8
ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ	9
ΕΙΣΑΓΩΓΗ	10
1 ΜΟΥΣΙΚΗ ΚΑΙ ΣΥΝΑΙΣΘΗΜΑ	11
1.1 Η αξία της μουσικής μέχρι σήμερα.....	11
2 WEB DEVELOPMENT	14
2.1 HTML - HyperText Markup Language (Γλώσσα Σήμανσης Υπερκειμένου) 14	
2.2 CSS - Cascading Style Sheets (Διαδοχικά Φύλλα Ύφους).....	15
2.3 PHP (PHP: Hypertext Preprocessor).....	16
2.4 Javascript.....	17
3 ΟΙ ΤΕΧΝΟΛΟΓΙΕΣ ΠΙΣΩ ΑΠΟ ΤΗΝ ΠΑΡΟΥΣΑ ΕΦΑΡΜΟΓΗ	19
3.1 NodeJS	19
3.2 Angular	20
3.3 Typescript	22
3.4 Firebase	22
3.5 JSON	23
4 ΥΛΟΠΟΙΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	24
4.1 ΒΗΜΑΤΑ ΥΛΟΠΟΙΗΣΗΣ	24
4.2 ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ	25
4.2.1 Σύνδεση με Firebase	25
4.2.2 Authentication για τον Firebase.....	25
4.2.3 User registration	26
4.2.4 User log-in	28
4.2.5 Main εφαρμογή.....	29
ΣΥΜΠΕΡΑΣΜΑΤΑ	33
ΒΙΒΛΙΟΓΡΑΦΙΑ	34

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

Εικόνα 1 Δείγμα απλού κώδικα HTML.....	15
Εικόνα 2 Σύνταξη ενός κανόνα στην CSS για μορφοποίηση μιας κεφαλίδας	16
Εικόνα 3 Λογότυπο PHP	16
Εικόνα 4 Δείγμα κώδικα PHP.....	17
Εικόνα 5 Javascript logo	17
Εικόνα 6 Δείγμα κώδικα Javascript.....	18
Εικόνα 7 Λογότυπου NodeJS.....	19
Εικόνα 8 Λογότυπου τηςAngular2	20
Εικόνα 9 Two-way databinding.....	21
Εικόνα 10 Το μοντέλο ιεραρχίας DOM.....	21
Εικόνα 11 Typescriptlogo.....	22
Εικόνα 12 Firebase logo	22
Εικόνα 13 JSON logo	23
Εικόνα 14 Ένα από τα τραγούδια όπως εμφανίζεται στη JSONβάση που χρησιμοποιεί η εφαρμογή μας	24
Εικόνα 15 Front-end registration form	26
Εικόνα 16 Front-end login form	28
Εικόνα 17 Front-end κύριας εφαρμογής.....	29
Εικόνα 18 Το dropdownmenuστο Front-end.....	30
Εικόνα 19 Τα switches στο Front-end	30

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

Πίνακας 1 Blocking & Non-blocking επικοινωνία	20
---	----

ΕΙΣΑΓΩΓΗ

Βλέποντας την εξέλιξη της μουσικής παρατηρούμε ότι ο άνθρωπος από αρχαιοτάτων χρόνων αναζητά την έκφραση συναισθημάτων μέσω αυτής. Έχοντας φτάσει στην εποχή του διαδικτύου , η επαφή του με τη μουσική γίνεται κυρίως μέσω αυτού. Για να υπάρξει μια πλατφόρμα που διαμοιράζεται μουσική θα πρέπει να έχει δημιουργηθεί μέσω της χρήσης γλωσσών προγραμματισμού όπως η HTML, η Javascript, Angular, Typescript, αλλά και εφαρμογές όπως το NodeJS και το Firebase.

Έχοντας επιλέξει το θέμα μας κληθήκαμε να αποφασίσουμε ποιες τεχνολογίες θα χρησιμοποιήσουμε για την υλοποίηση αυτής. Ύστερα από μελέτη των διαθέσιμων τεχνολογιών κάποιες από τις οποίες θα αναφέρουμε μέσα σε αυτό το βιβλίο, και βλέποντας τα μειονεκτήματα και τα πλεονεκτήματα που μας παρείχε η εκάστοτε τεχνολογία και προσέγγιση, καταλήξαμε στην απόφαση να δημιουργήσουμε μια διαδικτυακή εφαρμογή με τη χρήση της γλώσσας Angular Typescript και για τη βάση δεδομένων μας το εργαλείο Firebase της Google.

1 ΜΟΥΣΙΚΗ ΚΑΙ ΣΥΝΑΙΣΘΗΜΑ

1.1 Η αξία της μουσικής μέχρι σήμερα

Η μουσική είναι άρρηκτα συνδεδεμένη με την ανθρώπινη ιστορία, καθώς από την στιγμή που ο άνθρωπος δημιούργησε τις πρώτες κοινωνίες και άρχισε να έχει κοινωνικούς δεσμούς με τους υπόλοιπους ανθρώπους, άρχισε να δημιουργεί μουσική καθώς και όργανα για να την παράγουν.

Η αξία της μουσικής έχει αναγνωριστεί από όλους τους πολιτισμούς των αρχαίων χρόνων. Βλέπουμε ότι είχε εξέχουσα θέση στην ζωή των ανθρώπων ακόμη και μέσα από το γεγονός ότι θεοποιήθηκε και απέκτησε μορφή στο πρόσωπο του θεού Απόλλωνα.

Ένα ακόμη στοιχείο που μας δείχνει το πόσο σημαντική ήταν για τους ανθρώπους η μουσική, είναι ότι στην αρχαία Ελλάδα στην μεγάλη γιορτή των Παναθηναίων, τα οποία γίνονταν προς τιμή της θεάς Αθηνάς πέρα από αθλητικούς αγώνες υπήρχαν και μουσικοί αγώνες με μεγάλα έπαθλα.

Δεν ήταν όμως μόνο οι απλοί καθημερινοί άνθρωποι αυτοί που ασχολούνταν με τη μουσική. Ο φιλόσοφος Πλάτων θεωρούσε ότι «η μουσική είναι η κίνηση του ήχου για να φτάσει την ψυχή και να της διδάξει την αρετή. Η μουσική είναι ένας ηθικός κανόνας. Δίνει ψυχή στο σύμπαν, φτερά στη σκέψη, απογειώνει τη φαντασία, χαρίζει χαρά στη λύπη και ζωή στα πάντα» και ήταν ένας από τους λόγους για τον οποίο πίστευε ότι η μουσική είναι ένα απαραίτητο μέσο διδασκαλίας για την εκπαίδευση των νέων. Η μουσική, σύμφωνα με τον Πλάτωνα, συνέβαλε θετικά στην ψυχική ισορροπία του ανθρώπου και την καλλιέργεια των πνευματικών αρετών του ατόμου.

Ο Αριστείδης Κοϊντιλιανός, αρχαίος Έλληνας μουσικός και συγγραφέας που έζησε τον 3ο αιώνα μ.Χ., στο έργο του «Περί μουσικής» αναφέρει ότι «η μουσική μπορεί να είναι σημαντικότερη ακόμη και από τη διαλεκτική, δεδομένου ότι η τελευταία μπορεί να είναι ευεργετική για την ψυχή, αλλά χωρίς την κατάλληλη μουσική προετοιμασία της ψυχής θα μπορούσε να είναι καταστροφική».

Τον Κοϊντιλιανό απασχόλησε επίσης, το ζήτημα της επίδρασης της μουσικής στην ψυχή και κατ' επέκταση ο ρόλος της μουσικής ως μέσο έκφρασης ή πρόκλησης συναισθημάτων. Ο ίδιος μετά το διαχωρισμό των μερών της μουσικής σε φυσικό, τεχνικό, χρηστικό και εξαγγελτικό αναφέρει και το πρακτικό μέρος στο οποίο εξετάζονται οι τρόποι με τους οποίους η τεχνική γνώση της μουσικής μπορεί να χρησιμοποιηθεί για το καλό της ψυχής.

Σε όλη την ιστορία της ανθρωπότητας η μουσική ως τέχνη αποτελεί μέσο έκφρασης των περισσότερων ανθρώπων και συναντώνται σε όλες τις φάσεις της ζωής ενός ανθρώπου. Θα μπορούσε να χαρακτηριστεί το «όργανο» για την εξωτερική των συναισθημάτων.

Από την πρώτη στιγμή της ζωής μέχρι και το τέλος αυτής, όλα τα κορυφαία στάδια της ύπαρξης συντροφεύονται από μουσική, όπως τα νανουρίσματα που

τραγουδάει η μητέρα στο μικρό της παιδί για να αποκοιμηθεί ή τα διάφορα και αναρίθμητα παιδικά τραγούδια τα οποία υπάρχουν και χαροποιούν τα παιδιά.

Στη συνέχεια έρχονται τα τραγούδια που έχουν σαν στόχο να εκφράσουν τις χάρες και τις λύπες σε όλο το διάβα της ζωής των ανθρώπων, τραγούδια της χαράς (γάμους, γεννήσεις παιδιών), τραγούδια της εργασίας (που είναι αφιερωμένα σε διάφορα επαγγέλματα), τραγούδια της ξενιτιάς, του θανάτου και άλλα πολλά γεγονότα τα οποία στιγματίζουν τον άνθρωπο.

Πέρα όμως από τη μουσική που έχει φτιαχτεί με βάση την έκφραση των συναισθημάτων στην καθημερινή ζωή των ανθρώπων υπάρχει και η θρησκευτική μουσική. Σκοπός ήταν και είναι να δημιουργήσει το κατάλληλο συναισθηματικό φορτίο ώστε να κάνει τον πιστό να νιώσει πιο κοντά με την θεότητα και το ανώτερο ον το οποίο επικαλείται ως θεό. Αυτό επιτυγχάνεται μέσα από τους ψαλμούς, οι οποίοι έχουν μια μουσικότητα και υπάρχουν πολλές και διάφορες τεχνικές για την απαγγελία τους.

1.2 Μουσική και Τεχνολογία

Η πορεία του ανθρώπου με την μουσική είναι αλληλένδετη εδώ και χιλιάδες χρόνια. Με την σταδιακή εξέλιξη της τεχνολογίας κατάφερε να έχει πιο εύκολη πρόσβαση σε αυτή.

Από την στιγμή που κατασκευάστηκε το πρώτο γραμμόφωνο το έτος 1877 από τον Τόμας Άλβα Έντισον άρχισε να πραγματοποιείται η εγγραφή μουσικών τραγουδιών και η αναπαραγωγή τους ανά πάσα ώρα και στιγμή από τους κατόχους των γραμμοφώνων.

Χαρακτηριστικό παραδείγμα είναι ότι μερικές δεκαετίες αργότερα οι εγγραφές δίσκων είναι μαζικές και γίνονται σχεδόν σε όλα τα μήκη και τα πλάτη του τότε σύγχρονου κόσμου. Μιας και η ανάγκη για κατ οίκον μουσική πλέον θεωρείται καθημερινή και αναγκαία.

Όσο όμως η τεχνολογία εξελίσσεται απο το γραμμόφωνο περνάμε στο πικάπ και στη συνέχεια στις ηλεκτρικές συσκευές αναπαραγωγής ήχου, όπως το ραδιοφωνο, το οποίο επί σειρά δεκαετιών σημάδευσε την πορεία της ανθρωπότητας όχι μόνο ως μέσο ψυχαγωγίας αλλά και ενημέρωσης.

Μπορεί να πει κανείς ότι είναι 100 περίπου χρόνια μετά, το 1979, που η μουσική βιώνει την επανάσταση της, καθώς ο χρήστης μπορεί να έχει εύκολη πρόσβαση και σε σχετικά χαμηλό οικονομικό κόστος. Μετά την εφεύρεση του κασετόφωνου (το 1960), η Sony παρουσιάζει το walkman. Η συσκευή αυτή έδωσε την δυνατότητα στο χρήστη να πηγαίνει όπου θέλει, καθώς για να λειτουργήσει χρειαζόταν μόνο μερικές μπαταρίες, και να ακούει ανά πασα ώρα και στιγμή την μουσική που ο ίδιος είχε επιλέξει.

Το walkman σταδιακά παραδίδει την σκυτάλη των πρωτείων που κατείχε στις συσκευές αναπαραγωγής μουσικής, στο φορητό CD player.

Το CD είναι διαθέσιμο στην αγορά από τα τέλη του 1982 και παραμένει ως το επίσημο μέσο για τις εμπορικές μουσικές καταγραφές έως σήμερα.

Το CD είναι ουσιαστικά μία ή περισσότερες στερεοφωνικές διαδρομές οι οποίες μπορούν να αποθηκεύονται χρησιμοποιώντας τη 16bit κωδικοποίηση PCM σε ένα ρυθμό δειγματοληψίας 44,1 kHz ανά κανάλι. Ενώ, η δυνατότητα αποθήκευσης τους σε ασυμπιεστο ψηφιακό ήχο είναι 74 λεπτά (ενώ σήμερα έχει επεκταθεί στα 80 λεπτά). Αυτό έχει ως αποτέλεσμα, καλύτερη ποιότητα ήχου.

Η ψηφιακή εποχή είναι πλέον γεγονός. Από τον Ιούλιο του 1994, κυκλοφόρησε το πρώτο λογισμικό που κωδικοποιούσε MP3 το οποίο ονομάστηκε L3enc. Το πρώτο λογισμικό που μπορούσε να αναπαράγει MP3, ήταν το Winplay 3 το οποίο κυκλοφόρησε στις 9 Σεπτεμβρίου του 1995 και με αυτό πολλοί άνθρωποι είχαν την ευκαιρία να αναπαράγουν αρχεία με ήχο τύπου MP3 στους υπολογιστές τους.

Η επιτυχία του MP3 οφειλόταν κυρίως στην επιτυχία εταιριών λογισμικού και των προγραμμάτων τους όπως το Winamp της Nullsoft (που κυκλοφόρησε το 1997), το mpg123 και το Napster (που κυκλοφόρησε το 1999). Από κοινού αυτά τα προγράμματα έκαναν για τον απλό χρήστη πολύ εύκολη τη διαδικασία της αναπαραγωγής, δημιουργίας, διανομής και συλλογής αρχείων mp3.

Το αποκορύφωμα σε όλη αυτή την διανομή και αναπαραγωγή μουσικών τραγουδιών με την μορφή πλέον του βίντεο όχι μόνο σαν ένα ακουστικό αρχείο, ήταν η δημιουργία μιας παγκόσμιας πλατφόρμας το YouTube.

Το Youtube δημιουργήθηκε το Φεβρουάριο του 2005 και το Νοέμβριο του 2006 ονομάστηκε από το περιοδικό Time ως «Η Εφεύρεση του 2006».

Η πλατφόρμα αυτή, ή καλύτερα η διαδικτυακή αυτή εφαρμογή δίνει την δυνατότητα σε μη εγγεγραμμένοι χρήστες να παρακολουθήσουν τα βίντεο, ενώ οι εγγεγραμμένοι χρήστες επιτρέπεται να ανεβάσουν απεριόριστο αριθμό βίντεο. Έτσι ο καθένας μπορεί να διαμοιραστεί ότι οπτικοακουστικό υλικό έχει και να το κάνει διαθέσιμο σε απεριόριστο αριθμό χρηστών.

Όπως βλέπουμε, πλέον η πρόσβαση στη μουσική είναι πολύ εύκολη. Αφανώς γιατί διαμοιράζεται εύκολα, και αφετέρου καθότι η πρόσβαση σε εφαρμογές όπως το YouTube γίνεται εύκολα με τη εξάπλωση των ασύρματων δικτύων και των smart συσκευών, με τις οποίες μπορεί να συνδεθεί στο διαδίκτυο ανά πάσα στιγμή.

2 WEB DEVELOPMENT

Για να έχει πρόσβαση ο χρήστης στο Youtube και σε κάθε άλλη διαδικτυακή εφαρμογή σημαίνει ότι αυτή η εφαρμογή τρέχει σε ένα προγραμματιστικό περιβάλλον σε έναν απομακρυσμένο σταθμό εξυπηρέτησης και ο χρήστης χρησιμοποιώντας κάποιο πρόγραμμα τύπου web browser, στέλνει μέσω αυτού αιτήματα στον host μέσω ενός πρωτοκόλλου HTTP. Με αυτόν τον τρόπο η εφαρμογή παρέχει τις υπηρεσίες της σε πολλούς σταθμούς εργασίας.

Για να δημιουργηθεί μια εφαρμογή στο σημερινό ιστό σημαίνει ότι έχει συνδυαστεί μία σειρά από τεχνικές και γλώσσες προγραμματισμού και σχεδιασμού . Πιο συγκεκριμένα:

- HTML
- CSS
- PHP
- Javascript

ή όπως στην περίπτωση της δικής μας πτυχιακής και άλλες τεχνικές όπως:

- Angular2

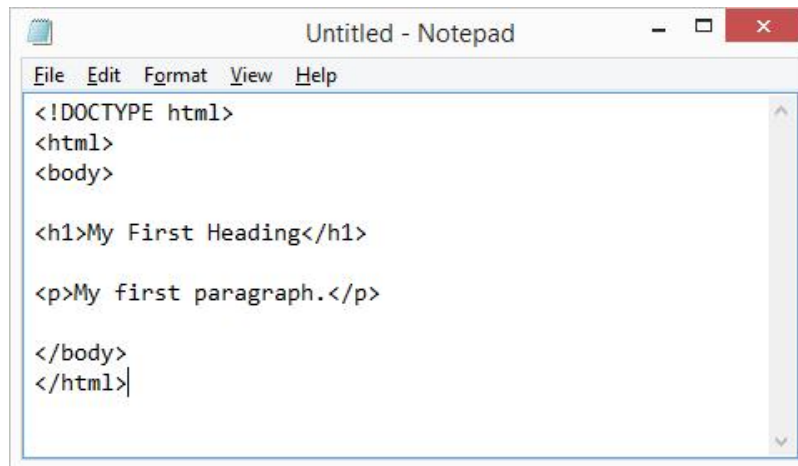
Ας δουμε αναλυτικά την καθε μία.

2.1 HTML - HyperText Markup Language (Γλώσσα Σήμανσης Υπερκειμένου)

Η HTML είναι γλώσσα προγραμματισμού, η χρήση της οποίας μας δίνει τη δυνατότητα να κατασκευάζουμε ιστοσελίδες, δημιουργώντας τη βασική τους δομή. Αυτό επιτυγχάνεται με τη χρήση των στοιχείων HTML, τις ετικέτες (tags). Όταν οι φυλλομετρητές (web browsers) λαμβάνουν ένα έγγραφο HTML επεξεργάζονται τις ετικέτες και εμφανίζουν το αποτέλεσμα του κώδικα στην οθόνη μας, παρουσιάζουν, δηλαδή, το περιεχόμενο της σελίδας χωρίς να εμφανίζει τις ίδιες τις ετικέτες HTML.

Είναι η γλώσσα που μας επιτρέπει την ενσωμάτωση κειμένων, εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και να δώσουμε μια πολύ βασική μορφοποίηση, η οποία ανταποκρίνεται στην κλασσική τυπογραφία. Έτσι δίνει τη δυνατότητα δημιουργίας παραγράφων, κεφαλίδων, λίστων, συνδέσμων, πινάκων.

Επιπλέον, δίνει τη δυνατότητα αποστολής δεδομένων προς τον server με τη χρήση ειδικών αντικειμένων, τις φόρμες δεδομένων, αλλά και την ενσωμάτωση σεναρίων (scripts) εντολών σε γλώσσες όπως η JavaScript, όπου διαδραματίζουν σημαντικό ρόλο και επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML ώστε από στατικές να γίνονται διαδραστικές.



```
File Edit Format View Help
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>
</html>
```

Εικόνα 1 Δείγμα απλού κώδικα HTML

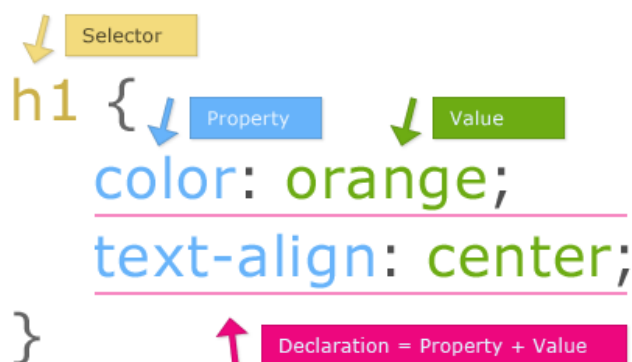
2.2 CSS - Cascading Style Sheets (Διαδοχικά Φύλλα Ύφους)

Τη μορφοποίηση, ή αλλιώς τον έλεγχο της εμφάνισης ενός εγγράφου HTML αναλαμβάνει η CSS (Cascading Style Sheets - Διαδοχικά Φύλλα Ύφους). Μια γλώσσα προγραμματισμού, η οποία εντάσσεται στην κατηγορία των γλωσσών φύλλων ύφους. Χρησιμοποιώντας την CSS έχουμε την δυνατότητα να δημιουργήσουμε οδηγίες μορφοποίησης, χρώματα, μεγέθη και να τις εφαρμόσουμε στα στοιχεία των σελίδων της ιστοσελίδας μας.

Η CSS μπορεί να επέμβει στη μορφοποίηση της ιστοσελίδας είτε με ενσωματωμένη δήλωση στον κώδικα της HTML σε μεμονωμένη ετικέτα, είτε με εσωτερική δήλωση, όπου ο απαραίτητος κώδικας τοποθετείται στην αρχή του εγγράφου HTML ή με εξωτερική δήλωση, όπου τα CSS Styles αποθηκεύονται σε εξωτερικά αρχεία τα οποία δίνουν έναν καθαρότερο κώδικα, χωρίς να εφαρμόζουμε πολλές ιδιότητες και ετικέτες οι οποίοι κάνουν τον κώδικά δυσανάγνωστο. Επίσης σημαντικό είναι και το γεγονός ότι πραγματοποιείται γρηγορότερη πλοήγηση καθώς μέσα από τα στίλ τα οποία έχουν οριστεί διαβάζονται από τον φυλλομετρητή μόνο μια φορά καθώς στη συνέχεια στην cache memory, μειώνοντας έτσι το μέγεθος της πληροφορίας που γίνεται download από τους φυλλομετρητές.

Επιπρόσθετα, η χρήση της CSS έχει το πλεονέκτημα ότι δημιουργείται μικρότερο αρχείο μεγέθους δεδομένου καθώς ο κώδικας μορφοποίησης εγγράφεται μόνο μια φορά και όχι σε κάθε σημείο κατά το οποίο εφαρμόζεται. Επιπλέον υπάρχει και καλύτερο SEO (Search engine optimization) διότι οι μηχανές αναζήτησης δεν «μπερδεύονται» ανάμεσα στο περιεχόμενο και τη μορφοποίηση του, αλλά έχουν πρόσβαση στο μόνο στο περιεχόμενο, οπότε με αυτόν το τρόπο είναι ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν.

Anatomy of a CSS Rule



Εικόνα 2 Σύνταξη ενός κανόνα στην CSS για μορφοποίηση μιας κεφαλίδας

2.3 PHP (PHP: Hypertext Preprocessor)



Εικόνα 3 Λογοτυπο PHP

Η PHP (PHP: Hypertext Preprocessor) είναι μία σεναριακή server-side γλώσσα προγραμματισμού βασισμένη στη C και την Perl και χρησιμοποιείται για την ανάπτυξη σελίδων web οι οποίες διαθέτουν δυναμικό περιεχόμενο.

Για να δώσει αυτή τη δυνατότητα στη σελίδα, ενσωματώνεται στον κώδικα της HTML μέσω της ειδικής ετικέτας `<?php ..?>`.

Σαν γλώσσα προγραμματισμού χρησιμοποιείται για την εκτέλεση σεναρίων (scripts) από την πλευρά του απομακρυσμένου εξυπηρετητή (host). Αυτό το γεγονός έχει σαν αποτέλεσμα ο κώδικας ο οποίος γράφεται για μια ιστοσελίδα σε γλώσσα PHP να μην καθίσταται άμεσα αντιληπτός αλλά στη συνέχεια και έπειτα από την επέμβαση του χρήστη στην ιστοσελίδα.

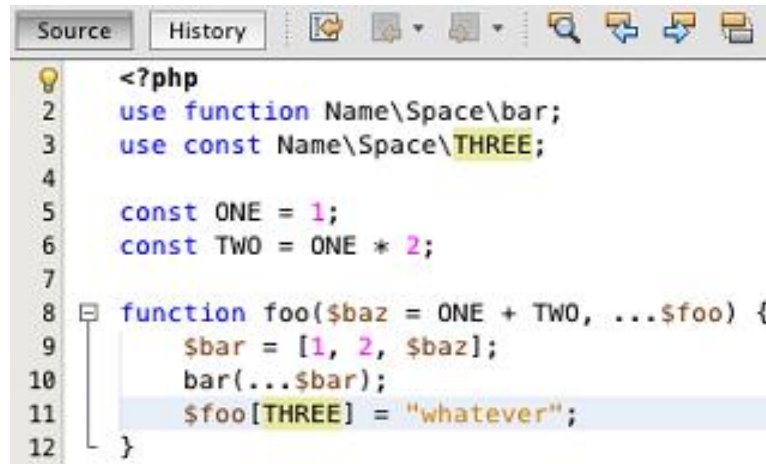
Πιο συγκεκριμένα η λογική διαδικασία που ακολουθεί ένα σενάριο PHP είναι η εξής:

- εκτέλεση του περιεχομένου κώδικα σειριακά, με ροή εκτέλεσης πάνω-κάτω και αριστερά – δεξιά,
- αίτημα για απορρόφηση δεδομένων από μια βάση δεδομένων,
- παραλαβή ή αποστολή δεδομένων,
- τερματισμός της διαδικασίας με την επίδειξη αποτελεσμάτων σε HTML σελίδα.

Η διαδικασία αυτή λέγεται server-side διότι, από τη στιγμή που ο χρήστης επιλέγει μια λειτουργία η οποία παραπέμπει σε εκτέλεση κώδικα PHP, ο κώδικας

αλληλεπιδρά με τη βάση δεδομένων από την «πίσω, αόρατη πλευρά», και επιστρέφει αποτελέσματα σε στατική σελίδα.

Σε συνεργασία με την HTML δημιουργείται το τελικό αρχείο που θα προβληθεί από τον εκάστοτε browser που θα ζητήσει την εμφάνιση της ιστοσελίδα



```
<?php
2 use function Name\Space\bar;
3 use const Name\Space\THREE;
4
5 const ONE = 1;
6 const TWO = ONE * 2;
7
8 function foo($baz = ONE + TWO, ...$foo) {
9     $bar = [1, 2, $baz];
10    bar(...$bar);
11    $foo[THREE] = "whatever";
12 }
```

Εικόνα 4 Δείγμα κώδικα PHP

2.4 Javascript



Εικόνα 5 Javascript logo

Η Javascript είναι μία αντικειμενοστραφής client-side γλώσσα προγραμματισμού που εκτελείται στον web browser του χρήστη. Μαζί με την CSS και την HTML αποτελεί τις βασικές γλώσσες προγραμματισμού του Παγκόσμιου Ιστού. Είναι η γλώσσα που μετατρέπει μια διαδικτυακή εφαρμογή από μία στατική παρουσίαση δεδομένων σε δυναμική. Η εκτέλεση της πραγματοποιείται, μέσω ενός interpreter προγράμματος (διερμηνέα), εφόσον ο πηγαίος κώδικας έχει συνδεθεί σε ένα έγγραφο υπερκειμένου HTML, είτε με εσωτερική ενσωμάτωση ή με σύνδεση ενός εξωτερικού αρχείου .js.

Ένα χαρακτηριστικό της Javascript είναι ότι, λόγω της δημοτικότητας της, μπορεί να εκτελεσθεί σε πολλά περιβάλλοντα / web browsers που διαφέρουν μεταξύ τους και από τον τρόπο που την εκτελούν, γι' αυτό το λόγο η Javascript επιτρέπει στον προγραμματιστή να γράψει κώδικα για αποσφαλμάτωση (debugging).

Ως γλώσσα προγραμματισμού δανείζεται συντακτικά στοιχεία από άλλες γλώσσες προγραμματισμού, όπως η JAVA, η Awk, και η Perl. Αυτός είναι και ο λόγος που και εδώ έχουμε μεταβλητές, σταθερές, βασικούς τύπους δεδομένων (αριθμούς, συμβολοσειρές κ.α.), πίνακες, τελεστές και δομές ελέγχου - βρόγχοι και συναρτήσεις.

Βασική διαφορά αυτής της γλώσσας από άλλες είναι ότι σε αυτήν δεν υπάρχει η δομή class και οι εσωτερικές μεταβλητές κλάσης - μεταβλητές μέλους, εδώ αποτελούν και ονομάζονται ιδιότητες του αντικειμένου (object properties).

Η Javascript παρέχει στον προγραμματιστή ένα πλήθος έτοιμων αντικειμένων, τα αντικείμενα πυρήνα (Core objects) αλλά φυσικά και τη δυνατότητα να δημιουργηθούν νέα.

Η Javascript ως μία ευέλικτη, δυναμική και αρκετά εύκολη στη σύνταξη γλώσσα, έδωσε τη δυνατότητα σε προγραμματιστές να δημιουργήσουν software frameworks, λογισμικά που παρέχουν μία γενική λειτουργία που μπορεί να αλλαχθεί από κάποιον επιπλέον χρήστη για τη διευκόλυνση ανάπτυξης λογισμικών εφαρμογών, προϊόντων και λύσεων.

Τα software frameworks μπορεί να περιλαμβάνουν υποστηρικτικά προγράμματα, μεταγλωττιστές, σετ εργαλείων και διεπαφή προγραμματισμού εφαρμογών (APIs). Έτσι δημιουργήθηκε και το Node.js, ένα εργαλείο που ουσιαστικά μετατρέπει την Javascript σε μία server side γλώσσα προγραμματισμού, πάνω από 25 χρόνια μετά την πρώτη φορά που η Javascript χρησιμοποιήθηκε και ως client-side και server-side. Η πρώτη φορά που χρησιμοποιήθηκε ως server-side ήταν στην υλοποίηση του Netscape Livewire.

```

server.js
1
2
3 var database = require('./database');
4
5 var server = express();
6 server.use(bodyParser.json());
7
8 server.use(function (req, res, next) {
9
10 // allow origins for demo purposes
11 res.setHeader('Access-Control-Allow-Origin', 'http://localhost:8080');
12 res.setHeader('Access-Control-Allow-Methods', 'GET, POST, DELETE');
13 res.setHeader('Access-Control-Allow-Headers', 'X-Requested-With, content-type');
14 next();
15 });
16
17
18 // Bootstrap

```

server.js server = 7 references

- var server = express();
- server.use(bodyParser...
- server.use(function (...
- server.get('/todos', fu...
- server.post('/todos', f...
- server.delete('/todos/...
- server.listen(PORT, fu...

Εικόνα 6 Δείγμα κώδικα Javascript

3 ΟΙ ΤΕΧΝΟΛΟΓΙΕΣ ΠΙΣΩ ΑΠΟ ΤΗΝ ΠΑΡΟΥΣΑ ΕΦΑΡΜΟΓΗ

Στην παρούσα πτυχιακή μία από τις πρώτες αποφάσεις που έπρεπε να λάβουμε ήταν το ποιές γλώσσες προγραμματισμού θα χρησιμοποιούσαμε για να υλοποιήσουμε την εφαρμογή. Λαμβάνοντας υπόψιν παραμέτρους όπως ευκολία εκμάθησης, προσαρμοστικότητα, και ταχύτητα αποφασίσαμε να βασίσουμε την εφαρμογή στη Javascript και μια σειρά απο frameworks. Σημαντικό ρόλο έπαιξε το γεγονός ότι πλέον με τη χρήση μίας μόνο γλώσσας μπορούσες να γράφεις κώδικα και για την πλευρά του client και του host.

Ακολουθεί αναλυτική περιγραφή των τεχνολογιών:

- a. NodeJS
- b. Angular2
- c. Typescript
- d. Firebase
- e. JSON

3.1 NodeJS



Εικόνα 7 Λογότυπο του NodeJS

Το Node.js ή Node είναι μια πλατφόρμα ανοιχτού κώδικα (open source cross-platform) ανάπτυξης λογισμικού η οποία έχει υλοποιηθεί σε περιβάλλον Javascript, κατασκευασμένη στη V8 JavaScript Engine της Google. Βασικά χαρακτηριστικά του και παράλληλα πλεονεκτήματα έναντι άλλων server-side γλωσσών προγραμματισμού είναι η ταχύτητα, που επιτυγχάνεται γιατί βασίζεται στην C++ και η ασύγχρονη επικοινωνία εισόδου/εξόδου (non-blocking I/O) μεταξύ των υπολογιστικών πόρων, το οποίο επιτυγχάνεται με την χρήση συμβάντων (events) που προσφέρει η Javascript και ονομάζονται callbacks.

Για να γίνει πιο κατανοητή η ασύγχρονη επικοινωνία, παρατίθενται τα βήματα που ακολουθούνται σε ένα blocking μοντέλο κώδικα και σε ένα non-blocking (Πίνακας 1-1).

Πίνακας 1 Blocking & Non-blocking επικοινωνία

Blocking	Non-blocking
1. Διαβάζει το αρχείο	1. Διαβάζει το αρχείο
2. Επεξεργάζεται το αρχείο	1.1. Όταν έχει διαβαστεί, το επεξεργάζεται
3. Εμφανίζει το αποτέλεσμα	1.1.1. Όταν έχει ολοκληρωθεί η επεξεργασία, εμφανίζει το αποτέλεσμα
4. Εκτελεί την επόμενη συνάρτηση	2. Εκτελεί την επόμενη συνάρτηση

Η λειτουργία του Node βασίζεται σε διάφορα εργαλεία (packages) κάποιον διαχειριστή πακέτων, όπως ο NPM που εγκαθιστάται μαζί με το βασικό αρχείο του node.js και τα modules.

Ο διαχειριστής πακέτων επιτρέπει τη δημοσίευση και τον διαμοιρασμό των ανοιχτού κώδικα βιβλιοθηκών Node.js από την κοινότητα, και είναι σχεδιασμένος να απλοποιεί την εγκατάσταση, αναβάθμιση και απεγκατάσταση των βιβλιοθηκών, modules που χειρίζονται διάφορες βασικές λειτουργίες .

Τα modules χειρίζονται το σύστημα αρχείων εισόδου/εξόδου, τη δικτύωση (HTTP, TCP, UDP, DNS, ή TLS/SSL), δυαδικά δεδομένα (ενδιάμεσες μνήμες), λειτουργίες κρυπτογράφησης, ροές δεδομένων και άλλες βασικές λειτουργίες. Τα modules του Node χρησιμοποιούν ένα API που είναι σχεδιασμένο να μειώνει την πολυπλοκότητα της συγγραφής εφαρμογών διακομιστών.

Όπως αναφέραμε, με την εγκατάσταση του βασικού εκτελέσιμου αρχείου, εγκαθιστάται και ο NPM με κάποια βασικά modules, αλλά μπορείς να τον τροφοδοτήσεις με επιπλέον modules, δημιουργημένα από άλλους προγραμματιστές που διατίθενται ελεύθερα στην επίσημη ιστοσελίδα του ή μέσω της γραμμής εντολών (cli) που εγκαθίσταται αυτόματα και αυτή.

Ουσιαστικά η λειτουργία του NPM είναι η εγκατάσταση κώδικα και ο χειρισμός εξαρτήσεων κώδικα από τη γραμμή εντολών.

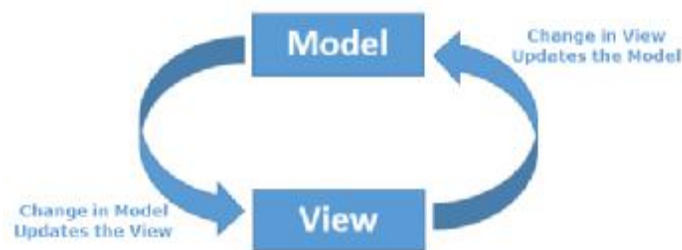
3.2 Angular



Εικόνα 8 Λογότυπο της Angular2

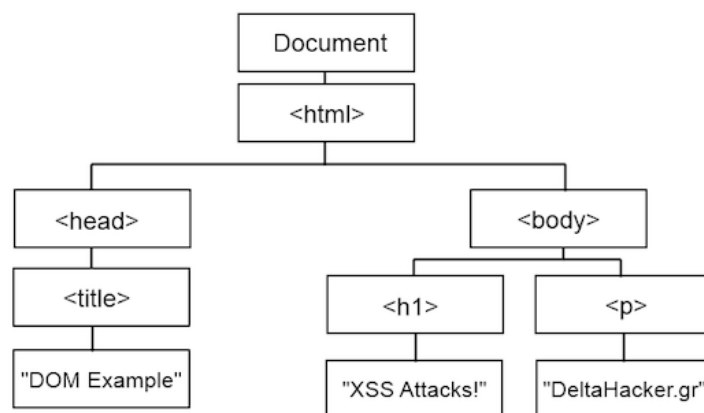
Η Angular ή Angular 2 είναι ένα client-side framework (ανοιχτού κώδικα διαδικτυακό πλαίσιο εφαρμογής) φτιαγμένο σε Typescript, ένα superset της Javascript που δίνει επιπλέον χαρακτηριστικά στις ετικέτες της HTML και έχει ως αποτέλεσμα να μετατρέπει ένα στατικό περιεχόμενο σε δυναμικό. Το βασικό της πλεονέκτημα είναι ότι με τη χρήση της μπορείς να δημιουργήσεις μία εφαρμογή συμβατή με όλες τις πλατφόρμες.

Η Angular φέρνει υπηρεσίες που παραδοσιακά βρίσκονται από την πλευρά του διακομιστή, στην πλευρά του πελάτη, χρησιμοποιώντας dependency injection (πρότυπο σχεδιασμού λογισμικού που υλοποιεί αναστροφή του ελέγχου για την επίλυση εξαρτήσεων(functions)). Κατά συνέπεια, ένα μεγάλο μέρος της επιβάρυνσης στον διακομιστή μειώνεται και έτσι αυξάνεται η λειτουργικότητα.



Εικόνα 9 Two-way databinding

Ένας ακόμη λόγος που η συγκεκριμένη γλώσσα επιφέρει υψηλή λειτουργικότητα και παραγωγικότητα είναι η σύνδεση δεδομένων διπλής κατεύθυνσης (two-way databinding). Η Angular δίνει τη δυνατότητα στα στοιχεία του HTML DOM (To Document Object Model ή σκέτο DOM. Το DOM αποτελεί έναν τρόπο αναπαράστασης του περιεχομένου μιας σελίδας HTML, ο οποίος τηρεί την ιεραρχία των στοιχείων που την απαρτίζουν) να συγχρονίζονται με τα properties του μοντέλου στην κλάση αλλά και με κάποιο event και όταν ο χρήστης αλλάξει τα δεδομένα, οι αλλαγές πραγματοποιούνται αυτόματα. Εν αντιθέσει με εφαρμογές client-side χωρίς αυτή τη λειτουργία, όπου όταν ο χρήστης αλλάξει τα δεδομένα της εφαρμογής, τότε αυτή πρέπει να τα στείλει για αποθήκευση στον server και αυτός να δώσει μία νέα απάντηση.



Εικόνα 10 Το μοντέλο ιεραρχίας DOM

3.3 Typescript

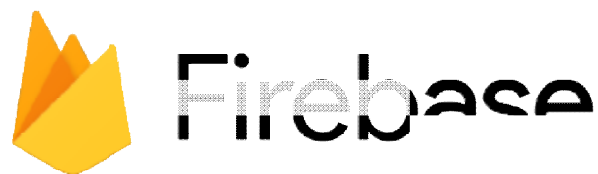


Εικόνα 11 Typescriptlogo

Η TypeScript είναι ένα ανερχόμενο class-based object-oriented πρότυπο που επιτρέπει στους developers να δουλεύουν σε μία κανονική strict γλώσσα, η οποία είναι πολύ πιο εύκολο να αναλυθεί και να απασφαλματοποιηθεί. Δημιουργήθηκε από τη Microsoft θέλοντας να κάνει την Javascript ακόμη περισσότερο διαχειρίσιμη.

Η Typescript δεν ενσωματώνεται στην HTML με κάποια `<script>` ετικέτα αλλά για να τρέξει σε κάποιον browser, θα πρέπει τα αρχεία της να γίνουν compile σε JavaScript. Με απλά λόγια, γράφεις τον κώδικα σε Typescript, δηλαδή απλοποιημένη Javascript και με τη χρήση ενός compiler μεταγλωτίζεται σε Javascript, συμβατή με όλες τις εφαρμογές και browsers.

3.4 Firebase



Εικόνα 12 Firebase logo

Το Firebase είναι μια Backend-as-a-Service πλατφόρμα ανάπτυξης διαδικτυακών εφαρμογών που ανήκει στην Google και λειτουργεί δηλαδή ως ένας server, προσφέροντας διασύνδεση με μία realtime non-SQL βάση δεδομένων γραμμένη σε JSON. Η χρήση αυτής της πλατφόρμας σε συνδυασμό με την Typescript μας επιτρέπει να δημιουργήσουμε μια εφαρμογή χωρίς να γράψουμε κώδικα για τον

server, καθώς χρειάζεται απλά να της κάνουμε κάποιες ρυθμίσεις και συνδέοντας την με το κώδικα, φέρνει τις απαντήσεις στα queries που του στέλνει ο χρήστης από τη βάση δεδομένων.

3.5 JSON



Εικόνα 13 JSON logo

Το JSON ή JavaScript Object Notation είναι ένα ανοικτό μορφότυπο βασισμένο πάνω σε ένα υποσύνολο της γλώσσας προγραμματισμού JavaScript, Standard ECMA-262 Έκδοση 3η - Δεκέμβριος 1999, το οποίο χρησιμοποιεί κείμενο, που μπορεί να διαβαστεί από τον άνθρωπο, για τη μετάδοση δεδομένων. Είναι τελείως ανεξάρτητο από γλώσσες προγραμματισμού αλλά χρησιμοποιεί πρακτικές (conventions) οι οποίες είναι γνωστές στους προγραμματιστές της οικογένειας προγραμματισμού C.

4 ΥΛΟΠΟΙΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

4.1 ΒΗΜΑΤΑ ΥΛΟΠΟΙΗΣΗΣ

Παρακάτω αναφέρονται τα βήματα που ακολουθήσαμε ώστε να ξεκινήσει η υλοποίηση της διαδικτυακής μας εφαρμογής.

1. Εγκαταστάθηκε ο Node.js δίνοντας μας πρόσβαση στο NPM και με τη σειρά του στα εργαλεία της Typescript ώστε να μπορούμε να κάνουμε το compilation σε javascript.
2. Βρέθηκε ένα πρότυπο θέμα AngularTS το οποίο παρέχει ετοιμα elements (dropdowns, inputs, buttons κλπ), από όπου επιλέξαμε τα εργαλεία που χρειαζόμασταν για να υλοποιήσουμε την εργασία και τα οποία παραμετροποιήσαμε ώστε να κάνουν τις εργασίες που χρειαζόμασταν.
3. Δημιουργήσαμε ένα νέο project στο firebase, το οποίο συνδέσαμε στον κώδικα μας
4. Συντάξαμε τη JSON βάση δεδομένων.
5. Κάναμε deploy στον server και κάναμε την εφαρμογή προσβάσιμη από παντού.



Εικόνα 14 Ένα από τα τραγούδια όπως εμφανίζεται στη JSONβάση που χρησιμοποιεί η εφαρμογή μας

4.2 ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

4.2.1 Σύνδεση με Firebase

Με τη χρήση έτοιμων modules που δίνονται από την επίσημη βιβλιοθήκη της Angular για τον Firebase, AngularFire

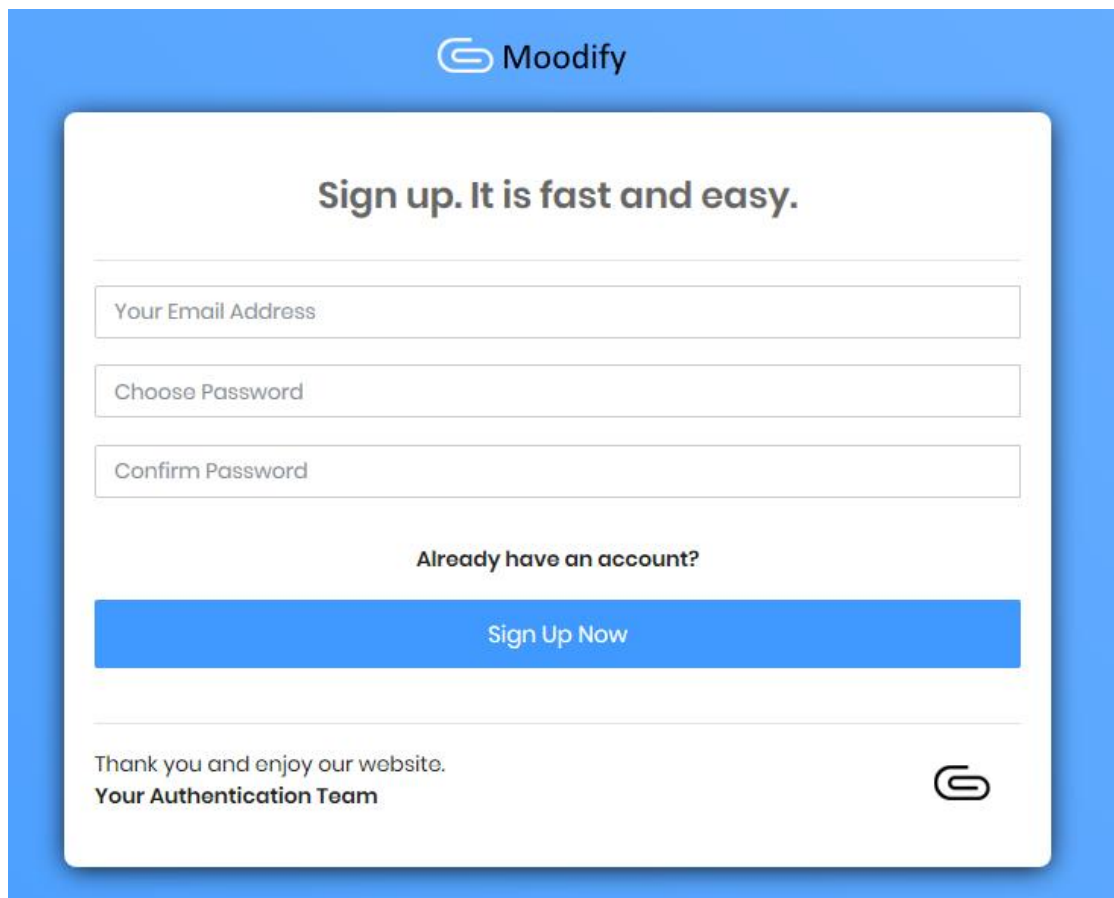
```
import { AngularFireDatabaseModule } from 'angularfire2/database-deprecated';
import { AngularFireAuthModule } from 'angularfire2/auth';
import { AngularFireModule } from 'angularfire2';
import { AngularFireDatabase, FirebaseListObservable,
FirebaseObjectObservable } from 'angularfire2/database-deprecated';
@NgModule({
  imports: [
    CommonModule,
    BasicLoginRoutingModule,
    SharedModule,
    SharedModule,
    FormsModule,
    HttpModule,
    AngularFireModule.initializeApp(firebaseConfig),
    AngularFireDatabaseModule,
    AngularFireAuthModule ],
  declarations: [BasicLoginComponent]
})
export class BasicLoginModule { }
```

4.2.2 Authentication για τον Firebase

Όταν δημιουργήθηκε το project μας στο περιβάλλον του Firebase μας δόθηκαν κάποια αναγνωριστικά τα οποία αντιστοιχούν στο API του Firebase και τα οποία ορίσαμε στη συνέχεια στον κώδικά μας ώστε να μπορεί να συνδέεται το Service του Firebase με τον server όπου θα γινόταν το deploy.

```
export const firebaseConfig = {
  apiKey: "AIzaSyBxOLymhqHuYvRgNLXuLDtOz_sXmMQ8J-A",
  authDomain: "emotionsmusic-7d5bd.firebaseio.com",
  databaseURL: "https://emotionsmusic-7d5bd.firebaseio.com",
  projectId: "emotionsmusic-7d5bd",
  storageBucket: "emotionsmusic-7d5bd.appspot.com",
  messagingSenderId: "644347012872"};
```

4.2.3 User registration



Moodify

Sign up. It is fast and easy.

Your Email Address

Choose Password

Confirm Password

Already have an account?

Sign Up Now

Thank you and enjoy our website.
Your Authentication Team

Εικόνα 15 Front-end registration form

Στο αρχείο `basic-reg.components.ts` μπορούμε να δούμε τον κώδικα ο οποίος επιτρέπει την εγγραφή νέου χρήστη. Ουσιαστικά εφόσον ο χρήστης δώσει το e-mail του και το επιθυμητό password, η Angular ζητάει από το Firebase να φέρει όλα τα τραγούδια που έχει η realtime JSON βάση μας.

```
this.afAuth.auth.createUserWithEmailAndPassword(this.email, this.password)
  .then(function(result)
  {
self.af.object('/Songs').subscribe(
  songs =>
  { var counter = 0;
    if(self.userUpdated)
    {
      self.router.navigateByUrl('/user/profile');
    }
  }
return;
```

Φυσικά υπάρχει και ο κώδικας που κάνει έλεγχο ότι ο κωδικός πρόσβασης που έχει ορίσει ο χρήστης στο πεδίο "password" και η επαλήθευσή του στο πεδίο "confirm password" ταιριάζουν. Αν ο έλεγχος δείξει ότι δεν είναι ίδιοι, βγάζει μήνυμα λάθους.

```
register()
{
  let self = this;
  if(this.password !== this.confirmPassword)
  {
    this.errorMessage = "Password does not mach";
  }
  return;
}
```

Έχοντας πλέον μπροστά της η Angular τα δεδομένα της βάσης και γνωρίζοντας ότι ο χρήστης είναι νέος μέσω του παρακάτω κώδικα, τον τοποθετεί ανά 15 τραγούδια σε μία κατηγορία. Κατά αυτόν τον τρόπο, ο χρήστης βλέπει την εφαρμογή να τρέχει, με 150 τραγούδια τοποθετημένα randomly σε κατηγορίες συναισθημάτων.

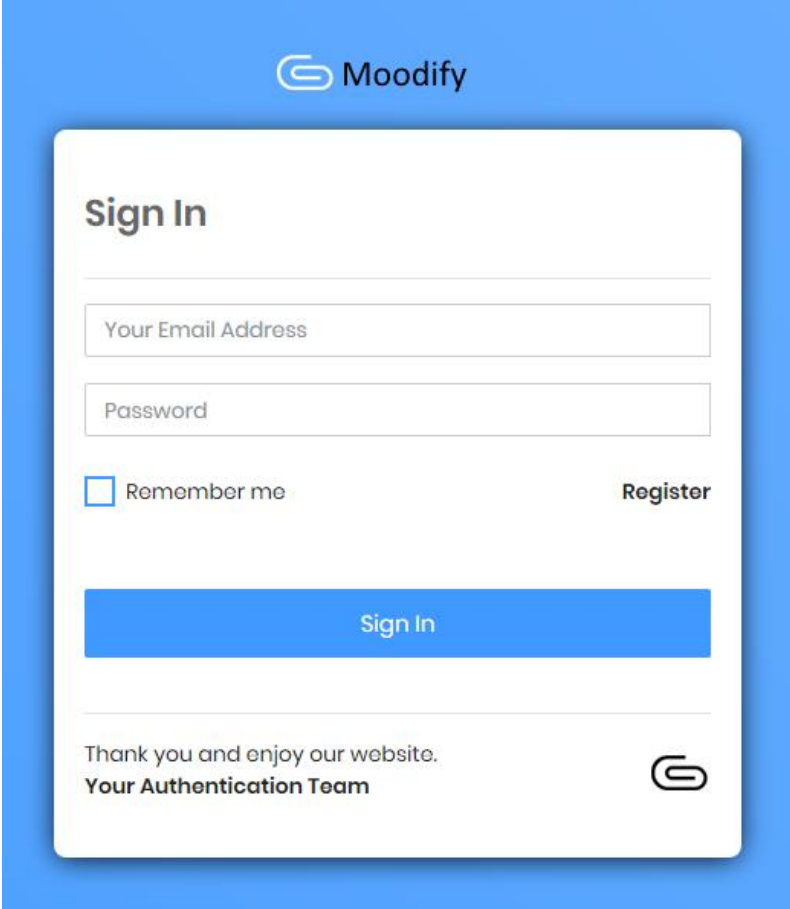
```
for(let song of songs)
{
  if(counter <= 15)
  {
    if(song["Emotions"].Humorous)
    {
      song["Emotions"].Humorous.push(self.email);
    }
    else
    {
      song["Emotions"].Humorous = [self.email];
    }
  }
  else if(counter > 15 && counter <= 30)
  {
    if(song["Emotions"].Sadness)
    {
      song["Emotions"].Sadness.push(self.email);
    }
    else
    {
      song["Emotions"].Sadness = [self.email];
    }
  }
}

[...]
```

```
counter = counter + 1;
self.af.object('/Songs/' + (counter - 1))
```

```
.update(song);
  if(counter == 150)
  {
    self.router.navigateByUrl('/user/profile');
self.userUpdated = true;
    break;
  }
}
```

4.2.4 User log-in



The image shows a 'Sign In' form for the 'Moodify' application. The form is centered on a blue background. It features a white card with rounded corners. At the top of the card, the 'Moodify' logo is displayed. Below the logo, the text 'Sign In' is prominently shown. The form includes two input fields: 'Your Email Address' and 'Password'. A checkbox labeled 'Remember me' is positioned below the password field, and a 'Register' link is located to its right. A large blue button with the text 'Sign In' is centered below the input fields. At the bottom of the card, a message reads 'Thank you and enjoy our website. Your Authentication Team' followed by the Moodify logo.

Εικόνα 16 Front-end login form

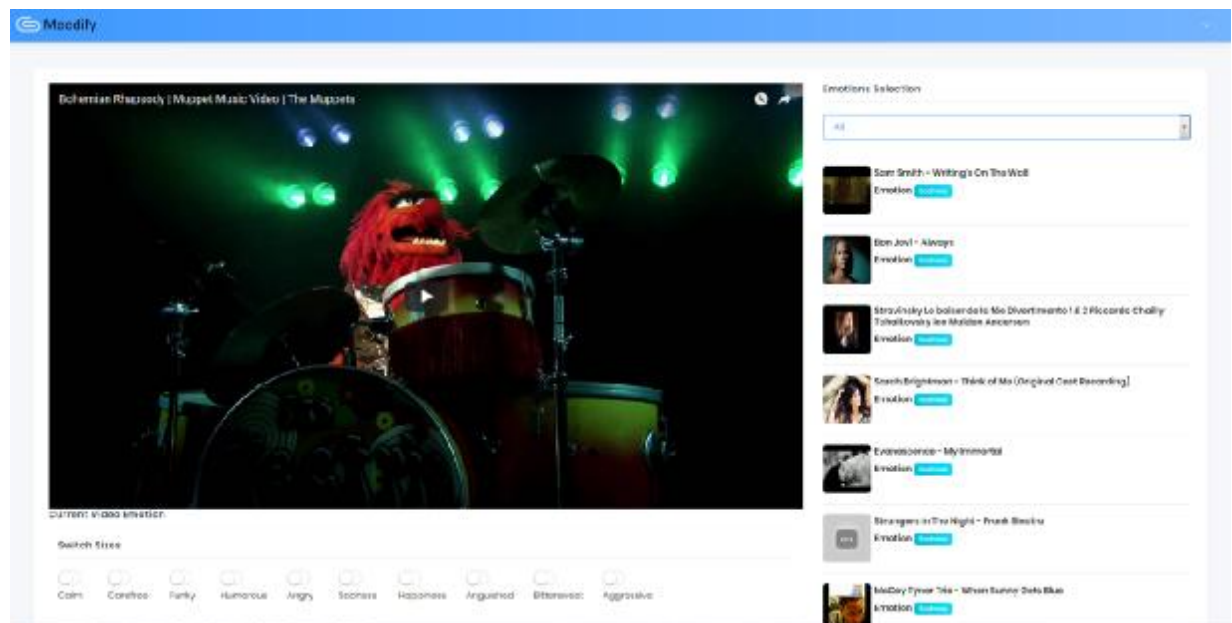
Στο αρχείο `basic-login.components.ts` μπορούμε να δούμε τον κώδικα ο οποίος επιτρέπει την είσοδο υπάρχοντος χρήστη. Με τον παρακάτω κώδικα, η Typescript εφόσον ο χρήστης δώσει τα στοιχεία του ζητάει από το Firebase να επικοινωνήσει με την JSON και να ελέγξει αν υπάρχει σαν χρήστης. Αν δεν υπάρχει, του εμφανίζει μήνυμα λάθους.

```

let self = this;
  this.afAuth.auth.signInWithEmailAndPassword(this.email, this.password)
  .then(function(result)
  {
    console.log("result",result);
    self.router.navigateByUrl('/user/profile');
  })
  .catch(function(error)
  {
    self.errorMessage = error["message"];
    console.log("error",error);
  });
}

```

4.2.5 Main εφαρμογή



Εικόνα 17 Front-end κύριας εφαρμογής

Εφόσον ο χρήστης έχει κάνει τη σύνδεση του και έχει μπει στην εφαρμογή βλέπει το κύριο γραφικό της. Πίσω από αυτό που βλέπει ο χρήστης τρέχει ο κώδικας του αρχείου `user-profile.component.ts`.

Ο χρήστης έχει την επιλογή να επιλέξει μέσω ενός dropdown menu ποια κατηγορία τραγουδιών θέλει να ακούσει.

Emotions Selection



Εικόνα 18 Το dropdown menu στο Front-end

Ο κώδικας ελέγχει για το εκάστοτε τραγούδι αν ο χρήστης βρίσκεται στον πίνακα της επιλεγμένης κατηγορίας στην βάση και εφόσον υπάρχει εμφανίζει στην playlist το τραγούδι.

```
public isVisible = function(song)
{
  if(this.selectedEmotion == "Happiness" && song.Emotions.Happiness)
  {
    this.labelEmotion = "Happiness";
    return song.Emotions.Happiness.includes(this.currentUserEmail);
  }
}
```

Ο χρήστης στην συνέχεια επιλέγοντας ένα τραγούδι δίνει στον κώδικα εντολή να εμφανίσει στον player το τραγούδι που έχει επιλεγθεί.

```
public addVideoToPlayer = function(song, i)
{
  console.log("index:", i);
  this.currentSongKey = i;
  this.currentSong = song;
  window.scrollTo(0, 0);
  var currentLink =
  song.SongInfo.Link.replace("https://www.youtube.com/watch?v=", "https://www.youtu
  be.com/embed/");
  this.sanitizedVideo = this.sanitizeUrl(currentLink);
}
```

Κάτω από τον player βρίσκονται τα switches, ένα από τα οποία είναι ενεργοποιημένο ανάλογα με το ποια κατηγορία συναισθήματος είναι επιλεγμένη.



Εικόνα 19 Τα switches στο Front-end

Παρακάτω βλέπουμε απόσπασμα από τον κώδικα που δημιουργεί 2 από τα 10 switches.

```
if(song.Emotions.Happiness &&
song.Emotions.Happiness.includes(this.currentUserEmail))
  { this.isHappiness= true;
    this.isSadness= false;
    this.isCalm= false;
    this.isCarefree= false;
    this.isFunky= false;
    this.isHumorous= false;
    this.isAngry= false;
    this.isAnguished= false;
    this.isBittersweet= false;
    this.isAggressive= false;}

if(song.Emotions.Sadness &&
song.Emotions.Sadness.includes(this.currentUserEmail))
  { this.isHappiness= false;
    this.isSadness= true;
    this.isCalm= false;
    this.isCarefree= false;
    this.isFunky= false;
    this.isHumorous= false;
    this.isAngry= false;
    this.isAnguished= false;
this.isBittersweet= false;
    this.isAggressive= false;}
```

Αν ακούγοντας κάποιο τραγούδι, ο χρήστης θέλει να αλλάξει την κατηγορία στην οποία του το έχει εμφανίσει, μπορεί να ενεργοποιήσει το switch που αντιστοιχεί στο νέο συναίσθημα. Αυτή η ενέργεια, το click, ενεργοποιεί τον κώδικα ο οποίος εφόσον πρώτα ελέγξει ότι δεν έκανες click στο ήδη ενεργοποιημένο switch / συναίσθημα, με μία σειρά από IFs όπως η παρακάτω:

```
if(emotion == 'Calm' && this.isCalm === true)
  {   event.preventDefault();
      event.stopPropagation();
      return false;}
```

Πάει στο αρχείο της JSON και βγάξει το χρήστη από την κατηγορία που ήταν. Στον κώδικα που ακολουθεί βλέπουμε την διαδικασία αυτή για μία εκ των κατηγοριών.

```
if(this.currentSong.Emotions.Happiness)
{
    this.currentSong.Emotions.Happiness =
    this.currentSong.Emotions.Happiness.filter(item => item !== this.currentUserEmail)
}
```

Στη συνέχεια, μπαίνει στον πίνακα Emotion του JSON και τοποθετεί τον χρήστη στο συναίσθημα που επέλεξε και ενεργοποιεί το switch που αντιστοιχεί στο επιλεγμένο συναίσθημα.

```
if(emotion == 'Aggressive' && !this.currentSong.Emotions.Aggressive)
{
    this.currentSong.Emotions.Aggressive = [ this.currentUserEmail ];
    this.isHappiness= false;
    this.isSadness= false;
    this.isCalm= false;
    this.isCarefree= false;
    this.isFunky= false;
    this.isHumorous= false;
    this.isAngry= false;
    this.isAnguished= false;
    this.isBittersweet= false;
    this.isAggressive= true;
}else if(emotion == 'Aggressive' && this.currentSong.Emotions.Aggressive)
}
```


ΣΥΜΠΕΡΑΣΜΑΤΑ

Με την βοήθεια των γλωσσών και τεχνικών προγραμματισμού τα οποία αναλύθηκαν στα προηγούμενα κεφάλαια δημιουργήθηκε η διαδικτυακή εφαρμογή μας. Επίσης με την βοήθεια του Backend-as-a-Service Firebase η εφαρμογή είναι προσπελάσιμη από οποιαδήποτε πλατφόρμα λόγω του γεγονότος ότι το Firebase μας δίνει την δυνατότητα σε πραγματικό χρόνο να γίνεται η σύνδεση με την βάση δεδομένων της JSON. Με αυτό τον τρόπο ο χρήστης πληκτρολογώντας το URL της εφαρμογής σε οποιονδήποτε φυλλομετρητή (browser) έχει πρόσβαση σε αυτήν για άμεση χρήση, όπως αυτή έχει οριστεί από τις προδιαγραφές τις οποίες δόθηκαν κατά την δημιουργία της.

Όπως κάθε εφαρμογή, έτσι και η δική μας έχει περιθώρια εξέλιξης και επέκτασης των λειτουργιών της. Μία από τις λειτουργίες που θα μπορούσαν να προστεθούν είναι η προσθήκη νέων τραγουδιών στην εφαρμογή, από τον ίδιο το χρήστη. Να δίνεται ουσιαστικά, μία φόρμα η οποία να επικοινωνεί με τη βάση δεδομένων και να δημιουργεί νέες οντότητες για τα τραγούδια, οι οποίες να γίνονται άμεσα διαθέσιμες και στους υπόλοιπους χρήστες ή με την χρήση κάποιου checkbox να έχει τη δυνατότητα να προσθέτει αυτό το νέο τραγούδι αποκλειστικά στη δική του λίστα, χωρίς να ενημερώνονται οι λίστες των υπόλοιπων χρηστών.

Επιπρόσθετα, μια ακόμη επιλογή που θα ήταν δυνατόν να προστεθεί για τους χρήστες, είναι να μπορούν να διαγράψουν ένα τραγούδι από όλες τις λίστες τους. Έτσι, μόνο για τους ίδιους, το τραγούδι να μην εμφανίζεται σε καμία κατηγορία - συναισθημα. Αυτή η λειτουργία δημιουργεί βέβαια την ανάγκη να υπάρχει ένα section στην εφαρμογή όπου οι χρήστες να έχουν πρόσβαση στα τραγούδια που έχουν "διαγραφεί" ώστε να μπορούν να τα επαναφέρουν.

Ακόμη μια δυναμική επέκταση της εφαρμογής είναι να υπάρχουν ανα τραγούδι στατιστικά στοιχεία τα οποία να εμφανίζονται στο χρήστη όταν επιλέγει να κατηγοριοποιήσει το τραγούδι. Με αυτόν τον τρόπο, θα γνωρίζει πόσο τις εκατό τον χρηστών έχει την ίδια άποψη μαζί του, και σε ποιες άλλες κατηγορίες - συναισθήματα έχει τοποθετηθεί το τραγούδι.

Τέλος, θα μπορούσε να δημιουργηθεί και το αντίστοιχο mobile application, ώστε ο χρήστης να έχει τη δυνατότητα να χρησιμοποιεί την διαδικτυακή εφαρμογή στο κινητό του μέσω εφαρμογής κινητού και όχι μέσω mobile browser όπως συμβαίνει τώρα.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- ΠαναγιώτηςΔ. Κεντερλής (2009) Ανάπτυξη διαδικτυακών εφαρμογών - Θεωρία και πράξη, Αθήνα: Κεντερλής
- Seabrook, John (2012). «StreamingDreams». TheNewYorker. 6 Ιανουαρίου 2012
- Du Gay, Paul; et al. (1997). Doing Cultural Studies: The Story of the Sony Walkman. London: SAGE Publications, in association with The Open University
- <https://el.wikipedia.org/wiki/YouTube> (Σεπτέμβριος 2018)
- <https://el.wikipedia.org/wiki/MP3>(Σεπτέμβριος 2018)
- Καϊμάκης Π. (2004) Φιλοσοφία και Μουσική. Η μουσική στους Πυθαγορείους, τον Πλάτωνα, τον Αριστοτέλη και τον Πλωτίνο, Αθήνα:Μεταίχμιο
- Καϊμάκης Π.(2008) «Αριστείδης Κοϊντιλιανός: Ένας αισθητικός της μουσικής τηςΥστερης Αρχαιότητας», στο: Αισθητική και Τέχνη, Διεπιστημονικές προσεγγίσεις στην μνήμη Παναγιώτη και Έφης Μιχελή, Ηράκλειο: Πανεπιστημιακές εκδόσεις Κρήτης
- Μαργαρίτα Ισμήνη Λοΐζου (2009), Μουσική και Συναίσθημα, Μεταπτυχιακή εργασία, Α.Π.Θ. Φιλοσοφική Σχολή, Τμήμα Φιλοσοφίας & Παιδαγωγικής, επιβλ. καθηγητής Π.Καϊμάκης, Θεσσαλονίκη
- Εμμανουήλ Κυριαζάκος (2017), Η μουσική και τα τραγούδια ως μέσο περιβαλλοντικής ευαισθητοποίησης: απόψεις Ελλήνων καλλιτεχνών, Μεταπτυχιακή εργασία, Πανεπιστήμιο Αιγαίου, Σχολή Ανθρωπιστικών Επιστημών, Τμήμα επιστημών της προσχολικής αγωγής και του εκπαιδευτικού σχεδιασμού, επιβλ. καθηγητής: Ν. Στάυρου, Ρόδος
- https://el.wikiversity.org/wiki/Εκμάθηση_PHP (Οκτώβριος 2018)
- <http://www.tutor.edu.gr/index.php/javascript/javascript-i> (Οκτώβριος 2018)
- https://developer.mozilla.org/en-US/docs/Archive/Web/Server-Side_JavaScript/Walkthrough (Οκτώβριος 2018)
- <https://en.wikipedia.org/wiki/JavaScript#Functional> (Οκτώβριος 2018)
- https://en.wikipedia.org/wiki/Software_framework (Οκτώβριος 2018)
- <https://opensourceforu.com/2016/01/an-introduction-to-node-js-the-server-side-javascript/> (Οκτώβριος 2018)
- <https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js> (Οκτώβριος 2018)
- <https://www.w3schools.com/nodejs/> (Οκτώβριος 2018)
- [https://en.wikipedia.org/wiki/Npm_\(software\)](https://en.wikipedia.org/wiki/Npm_(software)) (Οκτώβριος 2018)
- Βασίλειος Καραβασίλης (2014), AngularJS, Μονάδα Αριστείας ΕΛΛΑΚ, Διαθέσιμο: https://ma.ellak.gr/edu/pluginfile.php/1174/mod_resource/content/1/ma%40ioa-S04E03-L2.pdf
- <https://web-mate.gr/ti-einai-i-angular/> (Οκτώβριος 2018)
- https://www.w3schools.com/angular/angular_scopes.asp (Οκτώβριος 2018)
- <https://developers.sap.com/tutorials/angular-add-javascript.html> (Οκτώβριος 2018)
- <https://www.typescriptlang.org/docs/handbook/typescript-in-5-minutes.html>(Οκτώβριος 2018)
- <https://www.firebit.gr/2017/03/02/γιατί-να-μάθω-angular-2/> (Οκτώβριος 2018)
- <https://firebase.google.com/docs/database/> (Οκτώβριος 2018)
- <https://howtofirebase.com/what-is-firebase-fcb8614ba442> (Οκτώβριος 2018)
- <https://www.copterlabs.com/json-what-it-is-how-it-works-how-to-use-it/> (Οκτώβριος 2018)
- <https://www.json.org/json-el.html> (Οκτώβριος 2018)