



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Μελέτη, σχεδιασμός και υλοποίηση διαδικτυακής υπηρεσίας καταχώρησης, διαχείρισης και αποστολής πληροφοριών πνευματικής ιδιοκτησίας. Design and implementation of a web service focusing on the storage of intellectual property information for music files and other types of audiovisual material.
Όνοματεπώνυμο Φοιτητή	Δαφνομήλης Νικόλαος-Χρήστος
Πατρώνυμο	Ανδρέας
Αριθμός Μητρώου	ΜΠΣΠ 13026
Επιβλέπων	Ευθύμιος Αλέπης, Επίκουρος καθηγητής



Η παρούσα Μεταπτυχιακή Διατριβή εγκρίθηκε ομόφωνα από την Τριμελή Εξεταστική Επιτροπή που ορίστηκε από τη ΓΣΕΣ του Τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς σύμφωνα με τον Εσωτερικό Κανονισμό Λειτουργίας του Προγράμματος Μεταπτυχιακών Σπουδών “Προηγμένα Συστήματα Πληροφορικής”.

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Ευθύμιος Αλέπης
Επίκουρος καθηγητής

(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

(υπογραφή)

Γεώργιος Τσιχριντζής
Καθηγητής



Μεταπτυχιακή Διατριβή

«Μελέτη, σχεδιασμός και υλοποίηση διαδικτυακής υπηρεσίας καταχώρησης, διαχείρισης και αποστολής πληροφοριών πνευματικής ιδιοκτησίας»

Επιμέλεια:

Δαφνομήλης Νικόλαος – Χρήστος



Ευχαριστίες

Η παρούσα διπλωματική διατριβή υλοποιήθηκε στα πλαίσια του Προγράμματος Μεταπτυχιακών Σπουδών «Προηγμένα Συστήματα Πληροφορικής» του Πανεπιστημίου Πειραιώς.

Θα ήθελα να ευχαριστήσω τον καθηγητή μου Κο Αλέπη Ευθύμιο για τη σημαντική βοήθειά του κατά τη διάρκεια της μελέτης και της υλοποίησης της εργασίας, την οικογένειά μου που με στηρίζει όλα αυτά τα χρόνια καθώς και όλους όσους βρίσκονταν δίπλα μου τους μήνες που χρειάστηκαν για την εκπόνηση αυτής της διατριβής.

Νοέμβριος 2015
Δαφνομήλης Νικόλαος-Χρήστος



Περίληψη

Σκοπός αυτής της διπλωματικής εργασίας είναι η μελέτη, ο σχεδιασμός και η υλοποίηση μίας εφαρμογής καταχώρησης πληροφοριών πνευματικής ιδιοκτησίας για μουσικά και οπτικοακουστικά έργα στο διαδίκτυο, κάνοντας χρήση σύγχρονων τεχνολογιών. Η εργασία βασίζεται στη τεχνολογία των Web Services και πιο συγκεκριμένα στη μέθοδο REST με επιστροφή JSON string. Μέσω της εφαρμογής ο χρήστης έχει τη δυνατότητα να κάνει εγγραφή και σύνδεση στο σύστημα, να καταχωρεί τα δεδομένα του, να τα τροποποιεί και αν δε τα χρειάζεται να τα διαγράφει. Επίσης έχει τη δυνατότητα να κάνει αποστολή της πληροφορίας μέσω email προσθέτοντας τα δικά του σχόλια. Η εφαρμογή έχει πλήρη responsive λειτουργία και υποστηρίζει δίγλωσσο περιεχόμενο (Ελληνικά και Αγγλικά).



Abstract

The purpose of this dissertation is the study, design and implementation of an application focusing on the storage of intellectual property information for music files and other types of audiovisual material available on the internet by using modern technology. It is based on Web Services technology and more specifically on the REST method via JSON string. By using the application, users can sign up and log in, register and edit their data, as well as delete the data if they do not use it. They are also able to share their information via email and add their own comments. The app provides a fully responsive service and supports content in two languages, Greek and English.



Πίνακας Περιεχομένων

1. Πρόλογος.....	9
2. Εισαγωγή	9
3. Ανασκόπηση πεδίου.....	11
3.1 Εταιρείες κατοχύρωσης πνευματικής ιδιοκτησίας	11
3.1.1 Copyright Registration Service (CRS)	12
3.1.2 Artists' Collecting Society (ACS)	13
3.1.3 Design and Artists Copyright Society (DACS)	14
3.1.4 Motion Picture Licensing Corporation (MPLC)	15
3.1.5 The Orchard.....	16
3.2 Υπηρεσίες οργάνωσης περιεχομένου	17
4. Εγχειρίδιο χρήστη (user manual).....	18
4.1 Εγγραφή χρήστη	18
4.2 Έλεγχος πεδίων	19
4.3 Σύνδεση χρήστη	20
4.4 Πίνακας ελέγχου χρήστη	21
4.5 Προβολή και τροποποίηση προφίλ χρήστη	23
4.6 Καταχώρηση μουσικού περιεχομένου	23
4.7 Λίστα τραγουδιών	24
4.8 Λειτουργία αναζήτησης	25
4.9 Τροποποίηση τραγουδιού	26
4.10 Αποστολή τραγουδιού.....	27
4.11 Διαγραφή τραγουδιού.....	28
4.12 Καταχώρηση οπτικοακουστικού περιεχομένου (ταινία)	29
4.13 Λίστα ταινιών	29
4.14 Τροποποίηση ταινίας	30
4.15 Αποστολή ταινίας	31
4.16 Διαγραφή ταινίας	32
4.17 Συμβατότητα με φορητές συσκευές.....	33
5. Εγχειρίδιο προγραμματιστή (developer manual) :.....	37
5.1 Προγράμματα και τεχνολογίες που χρησιμοποιήθηκαν.....	37
5.1.1 Σχετικά με τα Web Services.....	37



5.1.2	Σχετικά με τη PHP	38
5.1.3	Σχετικά με το Slim Framework	40
5.1.4	Σχετικά με το Bootstrap	41
5.1.5	Σχετικά με τη JavaScript	43
5.1.6	Σχετικά με την Angular JS.....	45
5.1.7	Σχετικά με την SQL	46
5.1.8	Σχετικά με τον Atom	47
5.1.9	Σχετικά με το Photoshop	48
5.1.10	Σχετικά με το XAMPP.....	49
5.2	Λειτουργίες της εφαρμογής.....	50
5.3	Αρχιτεκτονική Συστήματος.....	51
5.4	Σχεδιάγραμμα βάσης δεδομένων	52
5.5	Πίνακες βάσης δεδομένων	53
5.5.1	Πίνακας Users	53
5.5.2	Πίνακας Account	54
5.5.3	Πίνακας AuthTokens	55
5.5.4	Πίνακας Release	55
5.5.5	Πίνακας Movie.....	56
5.6	Διάγραμμα περιπτώσεων χρήσης	57
5.7	Επεξήγηση τμημάτων κώδικα	59
6.	Συμπεράσματα και μελλοντικές επεκτάσεις	64
6.1	Συμπεράσματα	64
6.2	Μελλοντικές επεκτάσεις και βελτιώσεις	64
7.	Πίνακας Εικόνων	66
8.	Βιβλιογραφικές Πηγές.....	68



1. Πρόλογος

Η παρούσα διπλωματική εργασία έχει ως σκοπό τη μελέτη, τον σχεδιασμό και την υλοποίηση μίας διαδικτυακής εφαρμογής (web service) καταχώρησης πληροφοριών πνευματικής ιδιοκτησίας για ακουστικά (μουσικά έργα) και οπτικοακουστικά έργα (βίντεο). Οι λειτουργίες της υπηρεσίας που καλούνται να υλοποιηθούν είναι οι ακόλουθες:

Αρχικά θα πρέπει να δημιουργηθεί η landing page για την υποστήριξη της υπηρεσίας. Μέσω αυτής ο επισκέπτης θα μπορεί να ενημερωθεί σχετικά με την υπηρεσία και να επιλέξει εάν θέλει να εγγραφεί στο σύστημα.

- δημιουργία responsive landing page για την υποστήριξη της υπηρεσίας
- εγγραφή του χρήστη στο σύστημα
- σύνδεση στο σύστημα
- καταχώρηση πληροφοριών για μουσικά έργα
- καταχώρηση εικόνας εξωφύλλου ανά μουσικό έργο
- καταχώρηση πληροφοριών για οπτικοακουστικά έργα
- καταχώρηση εικόνας εξωφύλλου ανά οπτικοακουστικό έργο
- τροποποίηση όλων των εγγραφών
- γρήγορη αναζήτηση και επιστροφή αποτελεσμάτων
- αποστολή πληροφοριών μέσω e-mail με authenticated email λογαριασμό
- υποστήριξη πρόσθετου κειμένου – σχολίων κατά την αποστολή
- προσοχή κατά την αποστολή να μη τροποποιείται η πληροφορία και να είναι read only
- responsive λειτουργία για φορητές συσκευές
- ασφάλεια κατά την είσοδο του χρήστη με τη δημιουργία ξεχωριστού κρυπτογραφημένου token που κάνει expire ανά 30 λεπτά
- κρυπτογράφηση κωδικού του κάθε χρήστη βάσει του SHA512 πρωτοκόλλου ασφαλείας
- δίγλωσσο περιεχόμενο, πλήρης υποστήριξη της Αγγλικής και Ελληνικής γλώσσας
- άμεση δυνατότητα επέκτασης της εφαρμογής σε άλλα συστήματα smartphones ή tablets ή άλλα services με την παροχή συγκεκριμένων header types για την υποστήριξη του JSON string επικοινωνίας με τον server

2. Εισαγωγή

Η ραγδαία εξέλιξη της τεχνολογίας έχει βοηθήσει σημαντικά ώστε η ανθρωπότητα να στραφεί σε κάθε μορφή τέχνης και να ενημερώνεται σε παγκόσμιο επίπεδο για τις τάσεις της εποχής. Έτσι κάθε ταλαντούχος άνθρωπος μπορεί να παράγει έργα πολύ πιο εύκολα συγκριτικά με το παρελθόν.

Καθημερινά κατακλυζόμαστε με νέα πληροφορία. Όλο και περισσότερος κόσμος έχει στραφεί στη δημιουργία των δικών του έργων. Πολλοί νέοι ασχολούνται με τη μουσική είτε με την υποκριτική με στόχο να ξεχωρίσουν και να κάνουν το όνειρό τους πραγματικότητα. Δε τα καταφέρνουν όλοι, αλλά κάποιοι με τα δείγματα της δουλειάς τους καταφέρνουν πολλές φορές και κεντρίζουν το ενδιαφέρον των εταιρειών που αναζητούν νέα πρόσωπα με ταλέντο και ένα αξιόλογο δείγμα δουλειάς στο οποίο θα μπορούσαν να εργαστούν πριν το βγάλουν στη κυκλοφορία. Είτε πρόκειται για μουσικό έργο είτε για οπτικοακουστικό δείγμα. Πως προστατεύεται όμως αυτό; Ο οποιοσδήποτε έχει τη δυνατότητα να παράγει ένα έργο και να το κάνει διαθέσιμο στο ευρύ κοινό. Πως όμως προστατεύεται ο δημιουργός του έργου; Πως διασφαλίζει την ακεραιότητα του έργου του; Πως μπορεί να εισπράξει τα κέρδη που του αναλογούν εάν δεν κατοχυρώσει σωστά το πνευματικό του έργο; Στη κατηγορία των έργων πνευματικής ιδιοκτησίας, υπάγονται οι μουσικές δημιουργίες και ό,τι έχει να κάνει με το κομμάτι της σύνθεσης της μουσικής, της συγγραφής των στίχων, της παραγωγής



ακόμη και του ερμηνευτή. Υπάρχουν επίσης όλα τα έργα γραπτού λόγου όπως τα ποιήματα, τα συγγράμματα, οι ιστορίες κλπ. Εκτός αυτών, υπάρχουν στα πνευματικά δικαιώματα τα οπτικοακουστικά έργα και ότι έχει να κάνει με αυτά. Από το κομμάτι της σύλληψης της ιδέας μέχρι το κομμάτι της παραγωγής, τους σεναριογράφους, τους ηθοποιούς και άλλους.

Τα πνευματικά δικαιώματα όμως δεν σταματάνε εκεί. Περιλαμβάνουν και έργα τέχνης, εικόνες, λογότυπα, σλόγκαν, εφαρμογές λογισμικού και άλλα πολλά που πολλές φορές δεν κατοχυρώνονται σωστά και οι δημιουργοί τους χάνουν μελλοντικά τα κέρδη από την εμπορευματοποίηση της δουλειάς τους.

Στη παρούσα εργασία θα μελετήσουμε τα πνευματικά δικαιώματα για μουσικά και οπτικοακουστικά έργα και θα δημιουργηθεί μία διαδικτυακή εφαρμογή καταχώρησης και οργάνωσης της πληροφορίας για κάθε έργο που υπάγεται σε πνευματική ιδιοκτησία. Από χώρα σε χώρα, αλλάζει ο τρόπος προστασίας των πνευματικών δικαιωμάτων για αυτές τις κατηγορίες. Οι κάτοχοι της πνευματικής ιδιοκτησίας κάθε μουσικού έργου στη χώρα μας είναι δικαιούχοι των πνευματικών δικαιωμάτων των έργων τους μέχρι και 70 έτη μετά το θάνατό τους. Μετά από αυτό το χρονικό περιθώριο τα έργα τους υπάγονται στη κατηγορία της ελεύθερης χρήσης από το κοινό, όπως γίνεται για παράδειγμα με όλα τα παραδοσιακά τραγούδια της Ελλάδας, που κατά καιρούς ερμηνεύονται από όλο και περισσότερους νέους καλλιτέχνες σε διάφορες παραλλαγές.

Κατά καιρούς έχουν δημιουργηθεί διάφορα σενάρια για το πώς μπορεί κανείς να προστατέψει το έργο του, αλλά η κυριότερη και πιο σωστή διαδικασία είναι η καταχώρηση του έργου του στο αρμόδιο τμήμα κατοχύρωσης πνευματικής ιδιοκτησίας. Για τα μουσικά έργα στην Ελλάδα υπάρχει η Ανώνυμη Εταιρεία Πνευματικής Ιδιοκτησίας (ΑΕΠΙ) η οποία απαρτίζεται από νομικούς συμβούλους και είναι υπεύθυνη για την ομαλή κατανομή των κερδών από τα πνευματικά δικαιώματα στους νόμιμους δικαιούχους του κάθε μουσικού έργου. Οι περισσότερες εταιρείες παραγωγής μουσικών ή οπτικοακουστικών έργων διατηρούν στο αρχείο τους τις πληροφορίες για τους δημιουργούς και τους πνευματικούς δικαιούχους κάθε έργου και αναγράφουν σε κάθε αρχείο τις περιοχές (territories) στα οποία έχουν δικαιώματα καθώς και τον μοναδικό σειριακό αριθμό που αντιστοιχεί σε κάθε έργο (ISRC). Το αρχείο αυτό αποτελείται από δεκάδες, εκατοντάδες, ενδεχομένως και χιλιάδες spreadsheets, που εμπεριέχουν την πληροφορία και μετά από ένα σημείο δεν είναι εύχρηστο και πολλές φορές οδηγεί σε λάθη.

Οι εταιρείες αυτές όμως βάσει συνεργασιών εμπορεύονται τα έργα και συνεργάζονται με άλλες εταιρείες ανά τον κόσμο όπου κάνουν παραχώρηση των δικαιωμάτων έναντι κάποιου χρηματικού ποσού και κάποιου ποσοστού από τα έσοδα των πωλήσεων ανά χώρα. Η διαδικτυακή εφαρμογή που σχεδιάστηκε και υλοποιήθηκε, αποσκοπεί στο να κάνει τη διαδικασία ανταλλαγής πληροφοριών για μουσικά και οπτικοακουστικά έργα πνευματικής ιδιοκτησίας, πιο γρήγορη και πιο ευέλικτη. Ουσιαστικά απευθύνεται σε εταιρείες παραγωγής, είτε αυτές είναι εταιρείες μουσικής παραγωγής είτε εταιρείες παραγωγής οπτικοακουστικών έργων (βίντεο/ταινίες) με σκοπό να συμβάλει στην οργάνωση του αρχείου κάθε εταιρείας και να βοηθήσει τον διαμοιρασμό της πληροφορίας κατά τη διαδικασία της μεταπώλησης ώστε να γίνει με περισσότερη ταχύτητα σε σύγκριση με τον κλασικό τρόπο που γίνεται σήμερα (ανταλλαγή των spreadsheets).

Η εφαρμογή που δημιουργήθηκε στα πλαίσια της διπλωματικής εργασίας, δίνει τη δυνατότητα σε κάποιον χρήστη να κάνει εγγραφή στο σύστημα και στη συνέχεια αφού πρώτα, συνδεθεί στον διακομιστή, να καταχωρεί πληροφορίες για κάθε «έργο» που έχει στη κατοχή του. Του δίνεται η δυνατότητα, να έχει πρόσβαση σε ένα online σύστημα διαχείρισης, το οποίο περιλαμβάνει όλες τις πληροφορίες σχετικά με τα μουσικά και οπτικοακουστικά έργα των οποίων έχει την άδεια χρήσης και είναι πνευματικός δικαιούχος. Ο κάθε χρήστης μέσω της εφαρμογής, μπορεί να κάνει χρήση εξειδικευμένων υπηρεσιών που σκοπό έχουν, την καλύτερη οργάνωση των μουσικών και οπτικοακουστικών μέσων, των οποίων είναι κάτοχος και μέσω αυτών, συμβάλει στην καλύτερη



οργάνωση και την επιτάχυνση χρονοβόρων διαδικασιών κατά τη διαδικασία της μεταπώλησης της πνευματικής ιδιοκτησίας και προώθησης της πληροφορίας σε τρίτους.

Η εφαρμογή, βασίζεται στη τεχνολογία των web services (REST) και κάνει χρήση νέων τεχνολογιών. Χρησιμοποιήθηκαν σύγχρονες γλώσσες προγραμματισμού και διασύνδεσης του front-end με το back-end και του server side με το web app. Για τη σωστή λειτουργία του server side τμήματος χρησιμοποιήθηκε το Slim Framework που βασίζεται στην πολύ δημοφιλή γλώσσα προγραμματισμού για web based εφαρμογές, PHP. Το web application υλοποιήθηκε με τη χρήση Javascript προγραμματισμού και κυρίως της Angular JS σε συνδυασμό με τη βιβλιοθήκη Bootstrap με το αποτέλεσμα να είναι άκρως λειτουργικό, σύγχρονο και πολύ γρήγορο. Η λειτουργία της εφαρμογής σχεδιάστηκε με σκοπό να είναι όσο το δυνατόν πιο απλή και φιλική στο χρήστη χωρίς να δημιουργούνται προβλήματα κατανόησης ή περιήγησης.

Σκοπός της εφαρμογής είναι να μπορεί να γίνει χρήση της υπηρεσίας από χρήστες που έχουν βασική γνώση στη χρήση ηλεκτρονικού υπολογιστή και διαδικτύου.

Με τη σύνδεσή του στην εφαρμογή ο χρήστης μπορεί να ξεκινήσει απευθείας να εισάγει τα δεδομένα του. Τα δεδομένα χωρίζονται σε δύο γονικές κατηγορίες. Η πρώτη κατηγορία είναι η μουσική βιβλιοθήκη. Σε αυτή, ο χρήστης εισάγει τις μουσικές κυκλοφορίες που εμπορεύεται η εταιρεία του ή ο ίδιος καταχωρώντας όλες τις βασικές πληροφορίες ανά μουσική κυκλοφορία. Οι πληροφορίες αυτές αποτελούνται από το τίτλο του τραγουδιού, τον ερμηνευτή, τον δημιουργό, τον μουσικοσυνθέτη, τον στιχουργό, τον μοναδικό εμπορικό κωδικό κυκλοφορίας (σε παγκόσμιο επίπεδο) ISRC, το σύνδεσμο στο μουσικό βίντεο του τραγουδιού, τη δισκογραφική εταιρεία, τα σχόλια, καθώς και την ημερομηνία πρώτης κυκλοφορίας. Η δεύτερη κατηγορία είναι η βιβλιοθήκη των βίντεο (οπτικοακουστικά μέσα). Σε αυτή τη κατηγορία, ο χρήστης εισάγει όλα τα δεδομένα που αφορούν τα βίντεο-ταινίες τα οποία εμπορεύεται η εταιρεία του. Οι πληροφορίες αυτές αποτελούνται από το τίτλο της ταινίας, τους πρωταγωνιστές, τον σκηνοθέτη, τον παραγωγό, την εταιρεία παραγωγής, τον μοναδικό κωδικό κυκλοφορίας, τα σχόλια, την ημερομηνία κυκλοφορίας και το εξώφυλλο. Όταν συμπληρώσει τις πληροφορίες, ο χρήστης έχει τη δυνατότητα να καλέσει τις εγγραφές του από τη βάση, να κάνει τροποποίηση της πληροφορίας για κάθε εγγραφή, να τροποποιήσει την εικόνα εξωφύλλου, να κάνει διαγραφή της καταχώρησης και τέλος, μπορεί μέσω της εφαρμογής να αποστείλει την κάθε εγγραφή με email με το πάτημα ενός κουμπιού. Στο email που δημιουργείται την ίδια χρονική στιγμή, ο χρήστης δε μπορεί να τροποποιήσει την πληροφορία που καλείται από τη βάση δεδομένων αλλά μπορεί να προσθέσει τα δικά του σχόλια πριν την αποστολή και να ενημερώσει τον παραλήπτη σχετικά με τη πληροφορία που λαμβάνει.

3. Ανασκόπηση πεδίου

Η καταχώρηση πνευματικής ιδιοκτησίας και η οργάνωση περιεχομένου έχει απασχολήσει κατά καιρούς πολλές εταιρείες που παρουσιάζουν τις δικές τους εκδοχές για την οργάνωση του περιεχομένου. Το πρόβλημα που εστιάζεται σε αυτές τις εφαρμογές είναι ότι δεν είναι τροποποιημένο ώστε να μπορεί να υποστηρίξει όλα τα πεδία που χρειάζονται οι εταιρείες ή οι δικαιούχοι πνευματικής ιδιοκτησίας. Μετά από έρευνα που έγινε στο διαδίκτυο θα μελετήσουμε μερικές πλατφόρμες οργάνωσης περιεχομένου και μία λίστα με εταιρείες κατοχύρωσης πνευματικής ιδιοκτησίας και τις ψηφιακές λύσεις που παρέχουν οι ίδιες στους πελάτες τους.

Πιο συγκεκριμένα κατόπιν έρευνας στο διαδίκτυο βρέθηκαν οι ακόλουθες εταιρίες που παρέχουν υπηρεσίες καταχώρησης πνευματικής ιδιοκτησίας και θα μπορούσε η εφαρμογή που υλοποιήθηκε να συνεργαστεί μαζί τους μέσω του API.

3.1 Εταιρείες κατοχύρωσης πνευματικής ιδιοκτησίας



3.1.1 Copyright Registration Service (CRS)

Η CRS είναι μία νομική εταιρεία που η βασική της υπηρεσία είναι η καταχώρηση πνευματικής ιδιοκτησίας και η προστασία του δημιουργού από αντιγραφή και παράνομη χρήση του έργου του. Ο σκοπός της υπηρεσίας είναι η εγγραφή και καταχώρηση των δημιουργιών των επιχειρήσεων ή των ανεξάρτητων δημιουργών με σκοπό τη προστασία της πνευματικής ιδιοκτησίας από κατάχρηση, εκμετάλλευση κλπ. Μέσω της ιστοσελίδας γίνεται απευθείας η καταχώρηση του έργου και των πληροφοριών που υπάγονται σε αυτό χωρίς να χρειάζεται παρέμβαση δικηγόρων και χωρίς να χρειαστεί ο ιδιοκτήτης να συμπληρώσει ατελείωτα γραφειοκρατικά έγγραφα. Εκτός αυτού το CRS δίνει τη δυνατότητα στους χρήστες να ελέγχουν για κάθε έργο πνευματικής ιδιοκτησίας, να μπορούν να δουν τον δικαιούχο και να επιβεβαιώσουν τη νομική ισχύ του ιδιοκτήτη. Σε περίπτωση αντιγραφής, ο χρήστης μπορεί να κάνει απευθείας online καταγγελία μέσω του CRS.

Εικόνα 1 – Ο ιστότοπος της υπηρεσίας CRS (Copyright Registration Service)



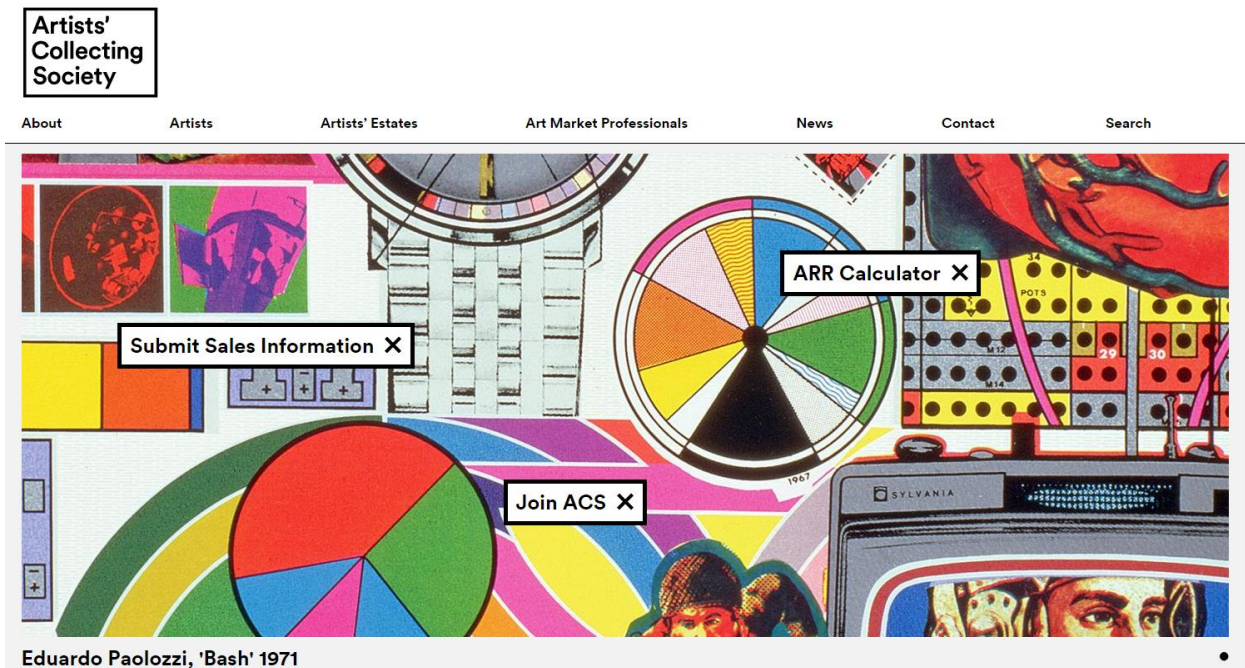
Εικόνα 2 – Σελίδα καταχώρησης περιεχομένου της υπηρεσίας CRS

Παρατηρείται ότι το CRS δίνει τη δυνατότητα στο δημιουργό να ορίσει γεωγραφικές περιοχές στις οποίες είναι νόμιμος δικαιούχος για το κάθε έργο που καταχωρεί.

3.1.2 Artists' Collecting Society (ACS)

Στη Μεγάλη Βρετανία το 2006 ιδρύθηκε μία καινούρια υπηρεσία. Το ACS ή αλλιώς Artists' Collecting Society. Η ACS διαχειρίζεται και εποπτεύει τη συλλογή και διανομή των μουσικών έργων στην Ευρώπη εκπροσωπώντας τους νόμιμους δικαιούχους. Η ACS δίνει τη δυνατότητα στους δικαιούχους να της αναθέτουν την είσπραξη των κερδών από τη χρήση των έργων τους και το κόστος ορίζεται από την ίδια την ACS και παραμένει ανταγωνιστικό με σκοπό να εξασφαλίσει το καλύτερο δυνατό κέρδος στον πελάτη της, δηλαδή τον ιδιοκτήτη της πνευματικής ιδιοκτησίας.

Η ACS εκπροσωπεί ήδη πάνω από 1.000 καλλιτέχνες και μεταξύ αυτών βρίσκονται πολύ γνωστά ονόματα από τη μουσική βιομηχανία.



Εικόνα 3 – Ο ιστότοπος της υπηρεσίας ACS (Artists' Collecting Society)

3.1.3 Design and Artists Copyright Society (DACS)

Ο οργανισμός DACS ιδρύθηκε το 1984 από μία ομάδα καλλιτεχνών και δικηγόρων μετά από την διαπίστωση ότι υπήρχε η ανάγκη για την ύπαρξη ενός οργανισμού που θα διαχειριζόταν τα δικαιώματα των καλλιτεχνών εικόνας στο Ηνωμένο Βασίλειο με σκοπό να διασφαλιστούν τα κέρδη αυτών των πνευματικών δικαιωμάτων. Η βασική υπηρεσία του οργανισμού είναι το copyright licensing όπου μέσω της κατοχύρωσης πνευματικής ιδιοκτησίας οι καλλιτέχνες πληρώνονται σε τέσσερις δόσεις ανά τρέχον έτος. Ο οργανισμός είναι μη κερδοσκοπικός και συντηρείται από την προμήθεια 25% επί των κερδών με σκοπό να καλύπτει τα λειτουργικά του έξοδα.



Εικόνα 4 – Ο ιστότοπος του οργανισμού DACS (Design & Artists Copyright Society)

3.1.4 Motion Picture Licensing Corporation (MPLC)

Ο οργανισμός Motion Picture Licensing Corporation (MPLC) είναι ένας ανεξάρτητος οργανισμός κατοχύρωσης πνευματικής ιδιοκτησίας που σχετίζεται με οπτικοακουστικό περιεχόμενο. Εκπροσωπεί και εξασφαλίζει τα πνευματικά δικαιώματα των παραγωγών ταινιών, των στούντιο παραγωγής, των σεναριογράφων και άλλων. Είναι παγκόσμιο το εύρος της κατοχύρωσης του κάθε έργου και διασφαλίζει τα δικαιώματα μέσω μίας ειδικής άδειας που εκδίδεται για κάθε έργο.



MPLC
Motion Picture Licensing Corporation

The Motion Picture Licensing Corporation (MPLC) is the leader in motion picture copyright compliance, supporting legal access across five continents and more than 30 countries.

Click on your country to be directed to your local MPLC website

[Home](#) | [About Us](#) | [Contact Us](#)

[View a list of MPLC Authorized Film & Television Producers](#)

WORLDWIDE LICENSING DIRECTORY

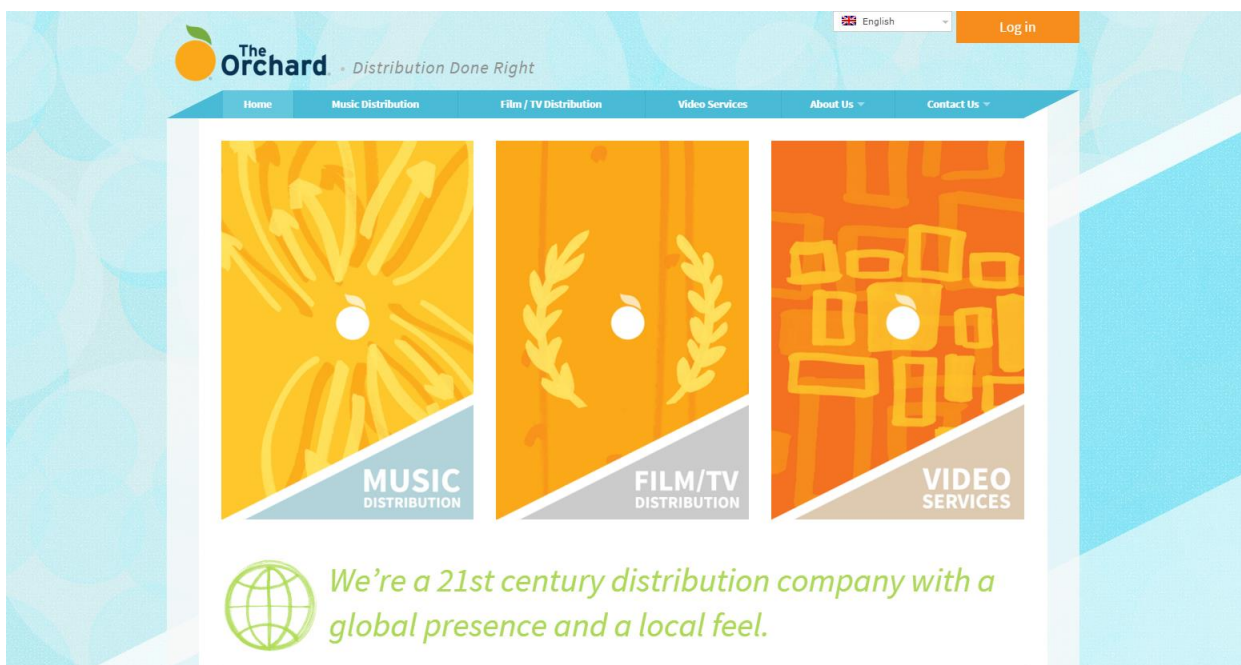
<p>NORTH AMERICA</p> <ul style="list-style-type: none"> • Canada (Churches Only) • USA • Mexico • Other 	<p>EUROPE</p> <ul style="list-style-type: none"> • Austria • Denmark • Finland • Germany • Hungary • Iceland • Ireland • Italy • Norway • Poland • Portugal • Romania • Spain and Andorra • Switzerland • United Kingdom • Other 	<p>ASIA/AFRICA</p> <ul style="list-style-type: none"> • Japan • Malaysia • Singapore • South Africa • Other
<p>SOUTH AMERICA</p> <ul style="list-style-type: none"> • Brazil • Chile • Argentina • Columbia • Ecuador • Guatemala • Nicaragua • Panama • Paraguay • Peru • Uruguay • Other 	<p>OCEANIA</p> <ul style="list-style-type: none"> • Australia (Churches Only) • New Zealand (Churches Only) 	

Home | Worldwide | Terms Of Use | Privacy Policy | My Account
©1995 - 2015 Motion Picture Licensing Corporation. All Rights Reserved.

Εικόνα 5 – Ο ιστότοπος του οργανισμού MPLC (Motion Picture Licensing Corporation)

3.1.5 The Orchard

Η Orchard είναι ίσως η πιο γνωστή εταιρεία διανομής μουσικών και οπτικοακουστικών πληροφοριών και λειτουργεί στη παγκόσμια αγορά και όχι τοπικά ανά χώρα. Είναι ο μεγαλύτερος διανομέας και παρέχει στους πελάτες της ένα αρκετά οργανωμένο πίνακα διαχείρισης των έργων που έχουν καταχωρήσει.



Εικόνα 6 – Ο ιστότοπος της Orchard

Όλες οι παραπάνω εταιρείες κατοχύρωσης πνευματικής ιδιοκτησίας, έχουν και προσφέρουν στους πελάτες τους μοναδικούς κωδικούς για είσοδο στο σύστημά τους μέσω του οποίου απλώς καταγράφουν τους κωδικούς των ακουστικών ή οπτικοακουστικών έργων των οποίων είναι κάτοχοι πνευματικής ιδιοκτησίας και το στέλνουν στην εκάστοτε εταιρεία. Δεν προσφέρεται παράλαυτά η δυνατότητα της προβολής, της οργάνωσης και της διαχείρισης των πληροφοριών του κάθε έργου καθώς και η αποστολή αυτών των πληροφοριών σε τρίτους. Ένα άλλο μεγάλο πρόβλημα είναι το γεγονός ότι οι περισσότερες από αυτές τις εταιρείες δεν είναι συμβατές με φορητές συσκευές και απαιτούν ηλεκτρονικό υπολογιστή για να συνδεθεί κανείς σε αυτές και να μπορεί να κάνει πλήρως χρήση των υπηρεσιών.

3.2 Υπηρεσίες οργάνωσης περιεχομένου

Οι περισσότερες εταιρείες διασφάλισης και διαχείρισης των πνευματικών δικαιωμάτων όπως οι παραπάνω προσφέρουν κάποιες σουίτες οργάνωσης περιεχομένου χωρίς όμως να δίνεται η δυνατότητα διαμοιρασμού της πληροφορίας και η χρήση αυτής τοπικά. Οι δισκογραφικές και οι κινηματογραφικές εταιρείες σήμερα, χρησιμοποιούν διάφορα CMS (Content Management Systems) τα οποία είναι τροποποιημένα στις απαιτήσεις τους για την καταχώρηση πληροφοριών σχετικά με το αντικείμενα της εργασίας τους και γενικά τα περισσότερα από αυτά είναι ελαττωματικά ως προς την χρήση και την υποστήριξη νέων τεχνολογιών. Αυτά τα CMS όμως δεν παρέχουν κάποιες βασικές λειτουργίες. Οι εταιρείες που εμπορεύονται τα δικαιώματα, έχουν έναν διαφορετικό τρόπο οργάνωσης και αρχειοθέτησης ο οποίος είναι αρκετά χρονοβόρος και απλοϊκός. Οι περισσότερες εταιρείες χρησιμοποιούν spreadsheets. Κάθε spreadsheet περιλαμβάνει την πληροφορία για κάθε έργο. Η πληροφορία που αναγράφεται είναι οι κύριοι συντελεστές κάθε δημιουργίας και οι νόμιμοι δικαιούχοι που θα έχουν μερίδιο από τα κέρδη της εμπορευματοποίησης κάθε έργου. Η οργάνωση περιεχομένου όμως μέσω spreadsheets δεν είναι λειτουργική ούτε παραγωγική. Πολλές φορές υπάρχουν απώλειες των αρχείων και σε περίπτωση μεγάλου όγκου από καταχωρήσεις, η



προσπέλαση των πληροφοριών είναι αρκετά δύσκολη. Έγινε έρευνα για υπηρεσίες καταχώρησης πληροφοριών για διάφορα έργα πνευματικής ιδιοκτησίας και παρατηρήθηκε ότι δεν υπάρχουν με αυτή καθ' αυτή την εκδοχή που ζητείται από τις εταιρείες.

Στην παρούσα διπλωματική εργασία θα υλοποιηθεί μία εφαρμογή βασισμένη στην τεχνολογία των web services, που θα καλύπτει το παραπάνω κενό και θα μπορούσε να συνεργαστεί πλήρως με μερικές από τις προαναφερθέντες υπηρεσίες.

4. Εγχειρίδιο χρήστη (user manual)

4.1 Εγγραφή χρήστη

Η λειτουργία της εφαρμογής είναι αρκετά απλή. Απαραίτητη προϋπόθεση για τη χρήση των υπηρεσιών είναι η εγγραφή χρήστη. Ο χρήστης κάνει εγγραφή συμπληρώνοντας την παρακάτω φόρμα. Τα πεδία που καλείται να συμπληρώσει είναι το όνομά του, το επώνυμό του, το email του, ένα επιθυμητό όνομα χρήστη (username), τον αριθμό τηλεφώνου του, την εταιρεία για την οποία εργάζεται και τον κωδικό πρόσβασης.

Συμπληρώστε τη φόρμα εγγραφής!

Όνομα	Επώνυμο
Ηλεκτρονικό ταχυδρομείο	Όνομα χρήστη
Τηλέφωνο	Εταιρεία
Κωδικός πρόσβασης	Πληκτρολογήστε ξανά τον κωδικό σας

[Εγγραφή](#) [Επιστροφή](#)

[Είστε ήδη μέλος](#)

Labelcopy App | MSc Thesis | Dafnomilis Nicolas - Christos | MPSP13026 | MSc Advanced Information Systems | University of Piraeus | June 2015

Εικόνα 7 – Εγγραφή χρήστη



4.2 Έλεγχος πεδίων

Τα πεδία που καλείται να συμπληρώσει ο χρήστης είναι αρχικά κόκκινα που υποδηλώνει την υποχρεωτική συμπλήρωση του πεδίου. Όταν τα πεδία συμπληρώνονται, αφαιρείται ο έλεγχος και επανέρχονται σε κανονικό χρωματισμό.

Εικόνα 8 - Πριν τη συμπλήρωση των πεδίων

Εικόνα 9 – Μετά τη συμπλήρωση των πεδίων

Όταν συμπληρωθούν τα πεδία της φόρμας, η φόρμα θα έχει την ακόλουθη δομή:

Εικόνα 10 – Συμπληρωμένα πεδία κατά την εγγραφή χρήστη

Λόγω της δίγλωσσης λειτουργίας της εφαρμογής, η φόρμα υποστηρίζεται από την αγγλική γλώσσα σε περίπτωση που το επιθυμεί ο χρήστης. Η εναλλαγή γλώσσας δεν απαιτεί ανανέωση της σελίδας.



Please fill in the form below to sign up!

Name: Nicolas

Dafnomilis: Dafnomilis

Email: ndafnomilis@gmail.com

Nick: nick

Phone: 210-1234567

Record Company: Record Company

Password:

Register Cancel

Already a member?

Labelcopy App | MSc Thesis | Dafnomilis Nicolas - Christos | MPSP13026 | MSc Advanced Information Systems | University of Piraeus | June 2015

Εικόνα 11 – Αγγλική εκδοχή της φόρμας εγγραφής

4.3 Σύνδεση χρήστη

Κατά την ολοκλήρωση της εγγραφής, ο χρήστης μεταφέρεται στη φόρμα σύνδεσης (login).

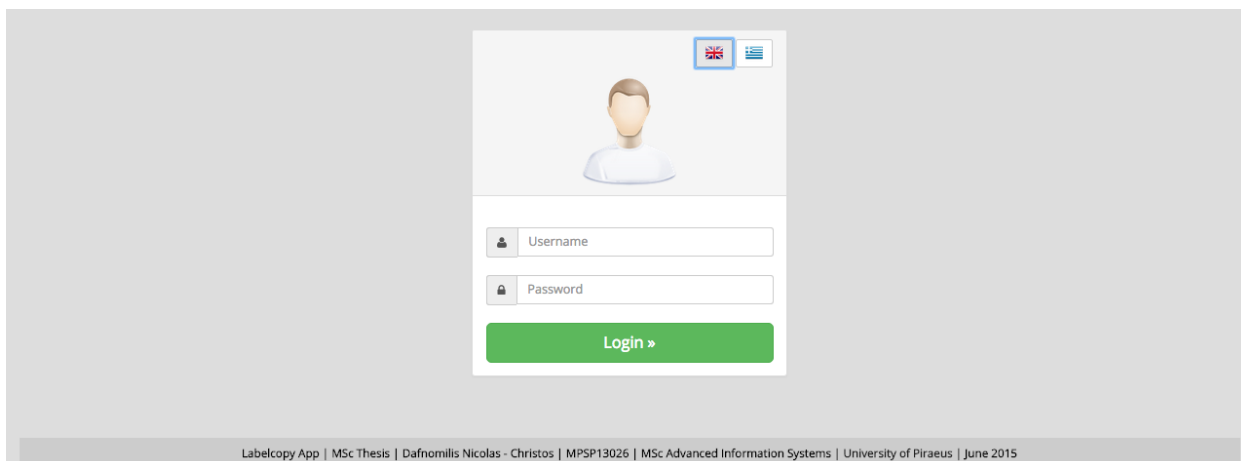
Όνομα χρήστη

Κωδικός πρόσβασης

Είσοδος >

Labelcopy App | MSc Thesis | Dafnomilis Nicolas - Christos | MPSP13026 | MSc Advanced Information Systems | University of Piraeus | June 2015

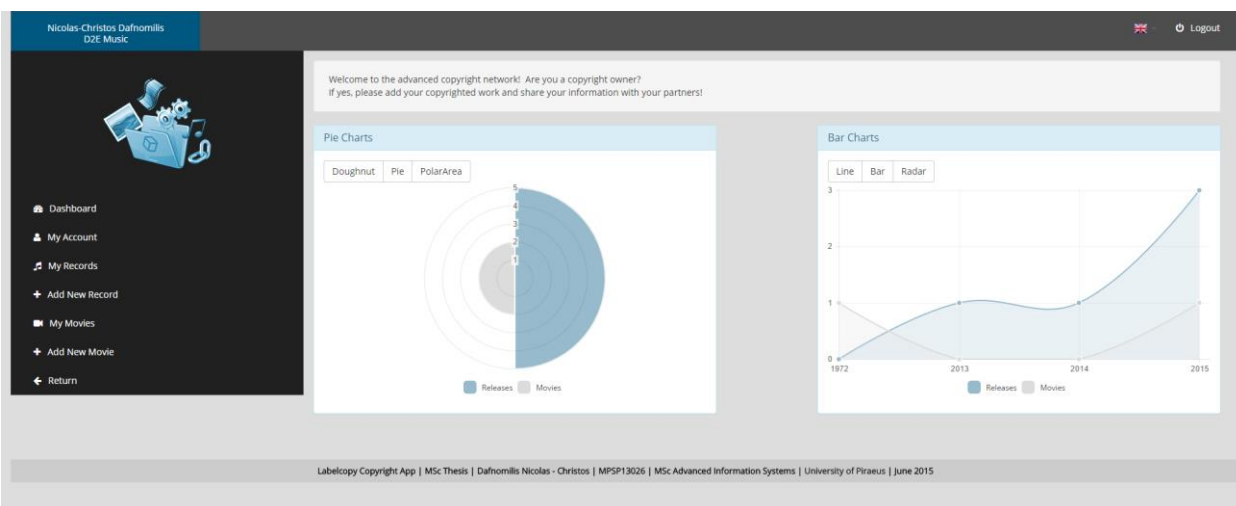
Εικόνα 12 – Η φόρμα σύνδεσης χρήστη στα Ελληνικά



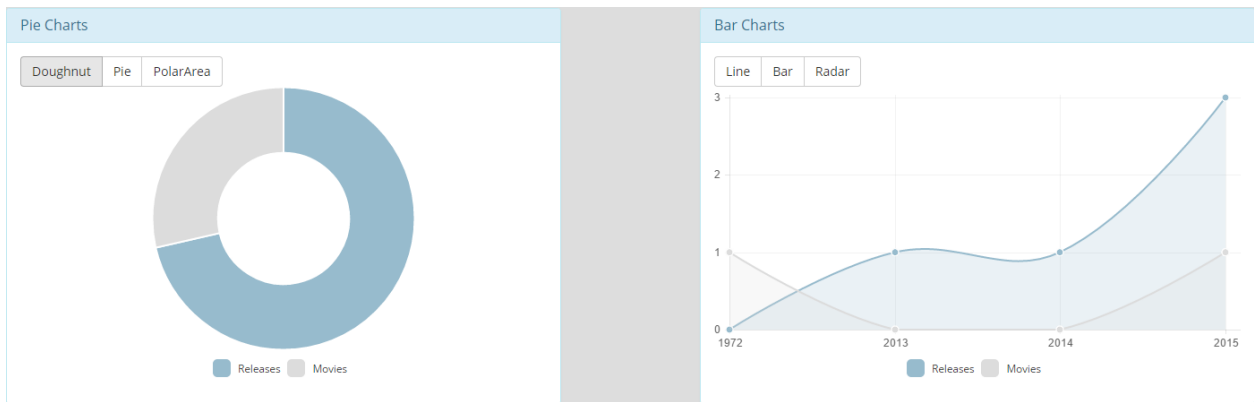
Εικόνα 13 – Η φόρμα σύνδεσης χρήστη στα Αγγλικά

4.4 Πίνακας ελέγχου χρήστη

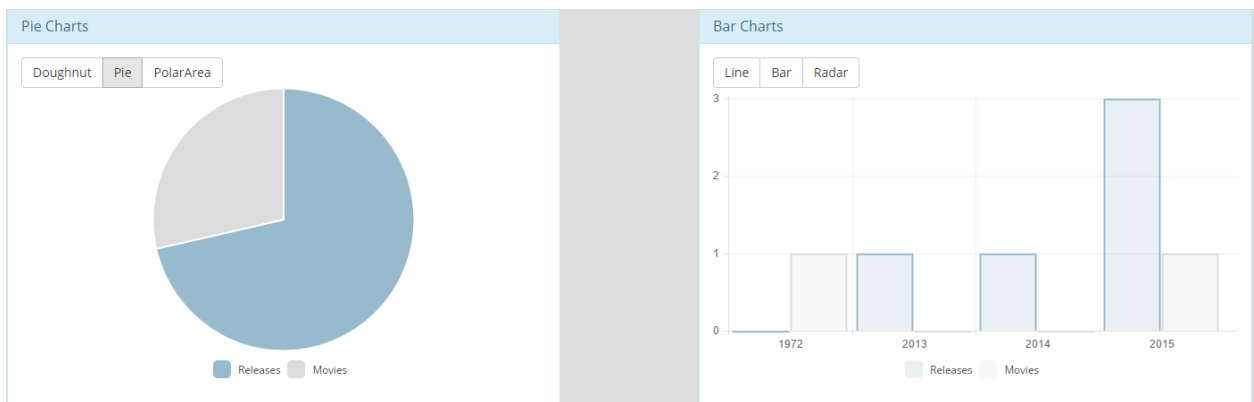
Κατά την είσοδό του στην εφαρμογή ο χρήστης μεταφέρεται στον πίνακα ελέγχου όπου εμφανίζονται τα στατιστικά στοιχεία σχετικά με τις καταχωρήσεις που έχει κάνει μέσω διαγραμμάτων. Στην αριστερή πλευρά διακρίνεται η περιοχή με τα pie chart διαγράμματα όπου ο χρήστης μπορεί να επιλέξει το διάγραμμα που θέλει να μελετήσει και στη δεξιά πλευρά εμφανίζονται τα bar chart διαγράμματα.



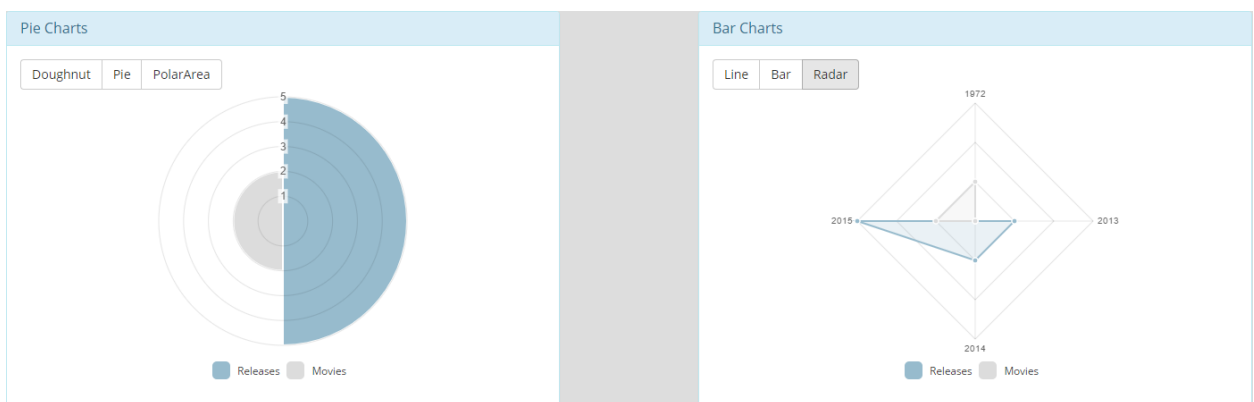
Εικόνα 14 – Πίνακας ελέγχου συνδεδεμένου χρήστη – Στατιστικά στοιχεία



Εικόνα 15 – Στατιστικά στοιχεία καταχωρήσεων



Εικόνα 16 – Στατιστικά στοιχεία καταχωρήσεων



Εικόνα 17 – Στατιστικά στοιχεία καταχωρήσεων



4.5 Προβολή και τροποποίηση προφίλ χρήστη

Επιλέγοντας από το μενού στα αριστερά την επιλογή «Ο λογαριασμός μου» ο χρήστης μεταφέρεται στην οθόνη προβολής και τροποποίησης των στοιχείων του λογαριασμού του. Τα πεδία που εμφανίζονται είναι το όνομα και το επώνυμο, η διεύθυνση ηλεκτρονικού ταχυδρομείου, το τηλέφωνο, το όνομα χρήστη και το πεδίο αλλαγής κωδικού.

Εικόνα 18 – Προβολή και τροποποίηση προφίλ χρήστη

4.6 Καταχώρηση μουσικού περιεχομένου

Επιλέγοντας από το μενού το σύνδεσμο «Προσθήκη Μουσικής» ο χρήστης μεταφέρεται στην οθόνη καταχώρησης μουσικής πληροφορίας. Σε αυτό το σημείο γίνεται η καταχώρηση των πληροφοριών πνευματικής ιδιοκτησίας για κάθε τραγούδι. Τα πεδία που καλείται να συμπληρώσει ο χρήστης είναι συνολικά δώδεκα. Εννέα υποχρεωτικά και τρία προαιρετικά.

- Τα υποχρεωτικά πεδία είναι ο τίτλος του τραγουδιού, ο καλλιτέχνης που το ερμηνεύει, η δισκογραφική εταιρεία που θα έχει τα δικαιώματα, το είδος του τραγουδιού, το όνομα του στιχουργού, το όνομα του παραγωγού, ο μοναδικός κωδικός isrc, η ημερομηνία πρώτης κυκλοφορίας και το όνομα του συνθέτη.
- Τα προαιρετικά πεδία είναι το εξώφυλλο του τραγουδιού, η ηλεκτρονική διεύθυνση του μουσικού βίντεο και τα σχόλια.

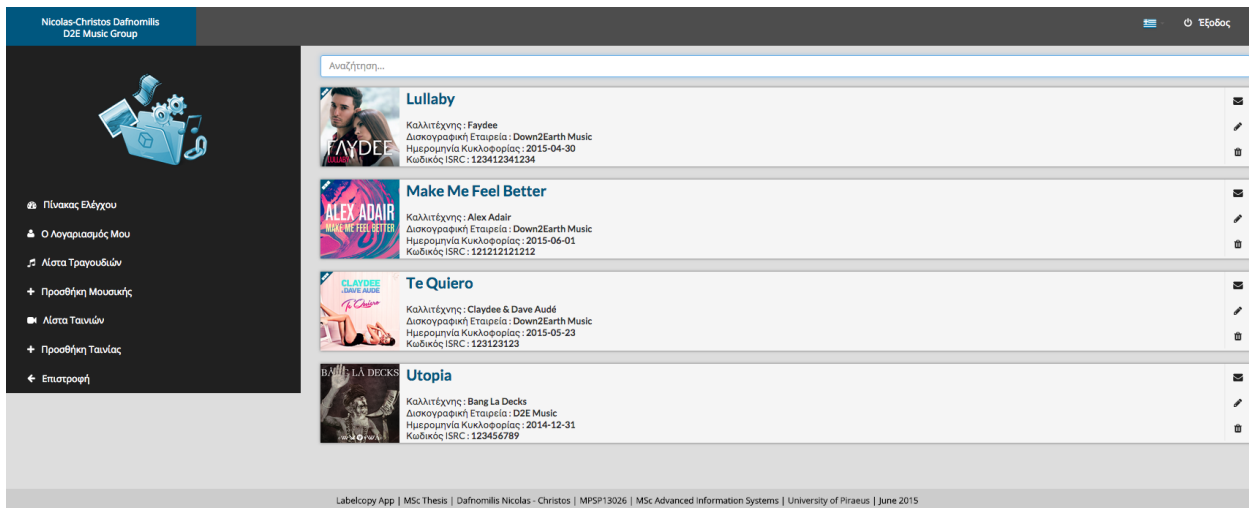
Όπως και στις προηγούμενες φόρμες, υπάρχει έλεγχος συμπλήρωσης των υποχρεωτικών πεδίων. Σε περίπτωση που δεν συμπληρωθεί κάποιο από τα υποχρεωτικά πεδία, ο χρήστης δεν μπορεί να χρησιμοποιήσει το κουμπί «Αποθήκευση» καθώς είναι ανενεργό. Το κουμπί ενεργοποιείται μόλις συμπληρωθούν όλα τα απαραίτητα πεδία. Όταν ο χρήστης συμπληρώσει τα υποχρεωτικά πεδία και κάνει αποθήκευση τότε μεταφέρεται αυτόματα στην οθόνη με τη λίστα των καταχωρήσεων όπου βλέπει τη τελευταία του καταχώρηση.



Εικόνα 19 – Προβολή της οθόνης καταχώρησης μουσικού περιεχομένου

4.7 Λίστα τραγουδιών

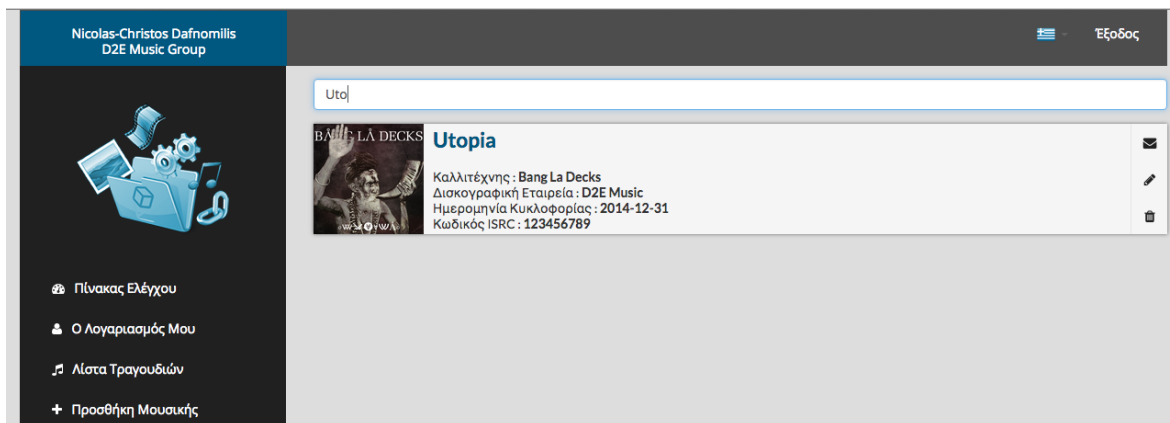
Στη συνέχεια ο χρήστης επιλέγοντας από το μενού την επιλογή «Λίστα Τραγουδιών» μεταφέρεται στην οθόνη που περιέχει όλες τις καταχωρήσεις που έχει κάνει στο σύστημα. Σε περίπτωση που δεν έχει καταχωρήσει κάποια πληροφορία, ο χώρος θα είναι κενός. Στην εικόνα εμφανίζεται η λίστα Τραγουδιών με τέσσερις καταχωρήσεις. Στα αριστερά εμφανίζεται το εξώφυλλο κάθε μουσικής κυκλοφορίας. Στη συνέχεια ο τίτλος είναι ευδιάκριτος συγκριτικά με την υπόλοιπη πληροφορία που εμφανίζεται και αποτελείται από τα πεδία «καλλιτέχνης», «δισκογραφική εταιρεία», «ημερομηνία κυκλοφορίας», «κωδικός ISRC». Για να δει ο χρήστης περισσότερες πληροφορίες πρέπει να κάνει χρήση των τριών επιλογών που εμφανίζονται στα δεξιά κάθε καταχώρησης. Οι επιλογές που έχει είναι «Αποστολή», «Τροποποίηση» και «Διαγραφή». Η εμφάνιση των καταχωρημένων εγγραφών γίνεται με αλφαβητική σειρά.



Εικόνα 20 – Προβολή της λίστας τραγουδιών

4.8 Λειτουργία αναζήτησης

Στο επάνω μέρος της οθόνης παρατηρείται η μπάρα αναζήτησης. Κατά την αναζήτηση, γίνεται κλήση του συστήματος στον διακομιστή, όπου καλεί όλη τη πληροφορία και βάσει του τι πληκτρολογεί ο χρήστης, του επιστρέφει όλα τα αποτελέσματα που περιέχουν τους χαρακτήρες που καλεί ο χρήστης χωρίς να χρειάζεται επαναφόρτωση της σελίδας. Με αυτό το τρόπο αποφεύγουμε τη διαδικασία του pagination καθώς η αναζήτηση εμφανίζει μόνο τα επιθυμητά αποτελέσματα και όχι όλες τις καταχωρήσεις. Αυτό κάνει την υπηρεσία αναζήτησης πολύ ελαστική και γρήγορη σε σχέση με τον κλασικό τρόπο κλήσης πληροφοριών από τη βάση και εμφάνιση αυτών σε νέα σελίδα με επαναφόρτωση ολόκληρης της σελίδας.

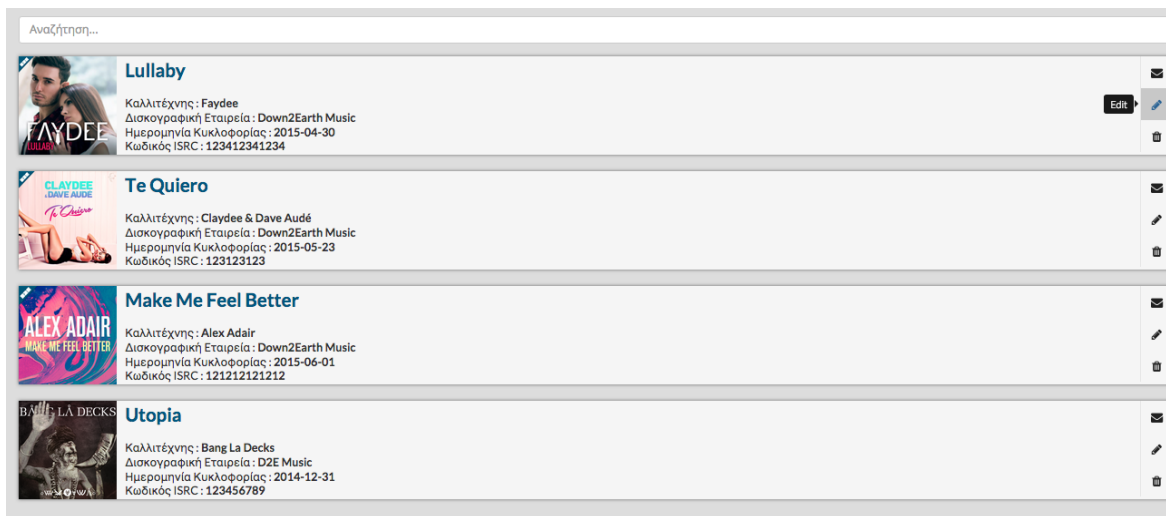


Εικόνα 21 – Εμφάνιση της λειτουργίας αναζήτησης



4.9 Τροποποίηση τραγουδιού

Στη λίστα που εμφανίζονται οι εγγραφές παρατηρείται ότι η κάθε εγγραφή στα δεξιά της φέρει τρεις επιλογές. Η πρώτη είναι η λειτουργία αποστολής, η δεύτερη είναι η λειτουργία επεξεργασίας και η τρίτη είναι η λειτουργία διαγραφής της εγγραφής από τη βάση. Πηγαίνοντας τον κέρσορα στο κουμπί της επεξεργασίας (εικονίδιο «μολύβι») εμφανίζεται το μήνυμα “Edit” μέσω βοηθητικής σήμανσης τύπου “tooltip”.



Εικόνα 22 – Εμφάνιση της επιλογής τροποποίησης της εγγραφής και της σήμανσης tooltip

Κάνοντας κλικ στο edit ο χρήστης μεταφέρεται στην οθόνη επεξεργασίας της εγγραφής. Η οθόνη επεξεργασίας είναι παρόμοια με την αρχική καταχώρηση απλώς καλεί τα στοιχεία από τη βάση δεδομένων και ο χρήστης μπορεί να τα τροποποιήσει και να ανανεώσει την εγγραφή στη βάση κάνοντας κλικ στο πλήκτρο «Αποθήκευση». Εναλλακτικά με κλικ στο πλήκτρο «Ακύρωση» ο χρήστης μεταφέρεται αυτόματα στην οθόνη με τη λίστα εγγραφών.



Εικόνα 23 – Εμφάνιση της φόρμας τροποποίησης της εγγραφής

4.10 Αποστολή τραγουδιού

Μία βασική λειτουργία της εργασίας είναι η δυνατότητα που έχει ο χρήστης να κάνει με ένα κλικ, αποστολή της κάθε εγγραφής, μέσω ηλεκτρονικού ταχυδρομείου (email). Επιλέγοντας το πρώτο πλήκτρο με το εικονίδιο «φάκελος» από τη δεξιά πλευρά κάθε εγγραφής εμφανίζεται ένα pop up παράθυρο που περιέχει μία φόρμα με τρία πεδία. Το πρώτο πεδίο είναι ένα textarea που εισάγεται η ηλεκτρονική διεύθυνση του παραλήπτη. Το δεύτερο πεδίο περιέχει τις πληροφορίες του αρχείου που θα αποσταλούν. Οι πληροφορίες αυτές αποτελούνται από τα πεδία:

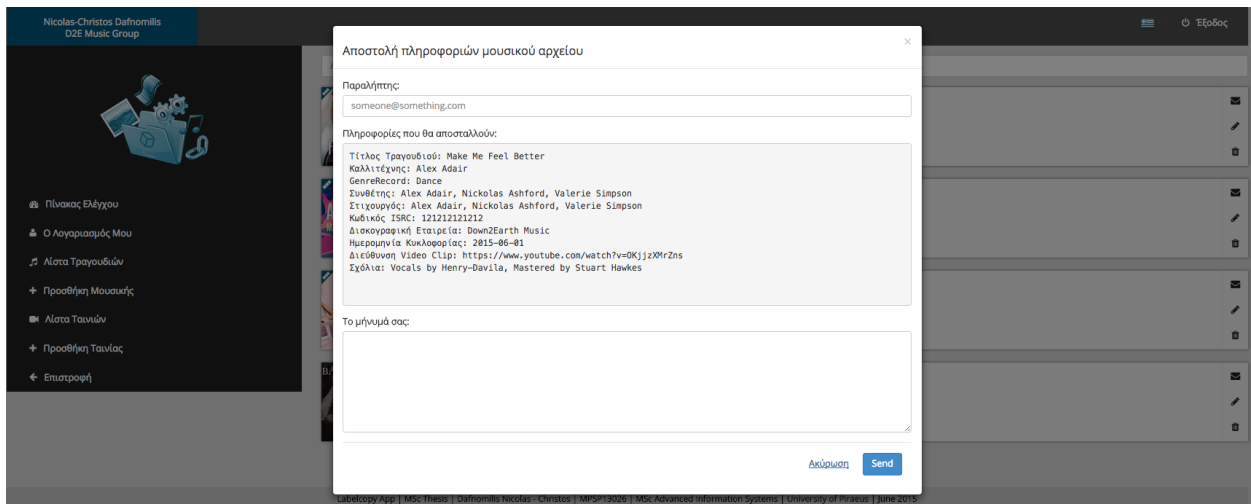
1. «Τίτλος»
2. «Καλλιτέχνης»
3. «Μουσικό Είδος»
4. «Συνθέτης»,
5. «Στιχουργός»,
6. «Κωδικός ISRC»,
7. «Δισκογραφική Εταιρεία»,
8. «Ημερομηνία Κυκλοφορίας»,
9. «Διεύθυνση Βίντεο Κλιπ»,
10. «Σχόλια»

Οι παραπάνω πληροφορίες καλούνται απευθείας από τη βάση δεδομένων και ο χρήστης δεν έχει δικαίωμα να τα τροποποιήσει κατά τη διαδικασία της αποστολής. Αυτό γίνεται κυρίως για να προστατευθεί η πληροφορία που αποστέλλεται.

Για να μπορεί όμως να στείλει κι ένα κείμενο της επιλογής του, ένα μήνυμα στον παραλήπτη έχει προστεθεί το τρίτο πεδίο. Το τρίτο πεδίο είναι ένα textarea στο οποίο ο χρήστης έχει τη δυνατότητα



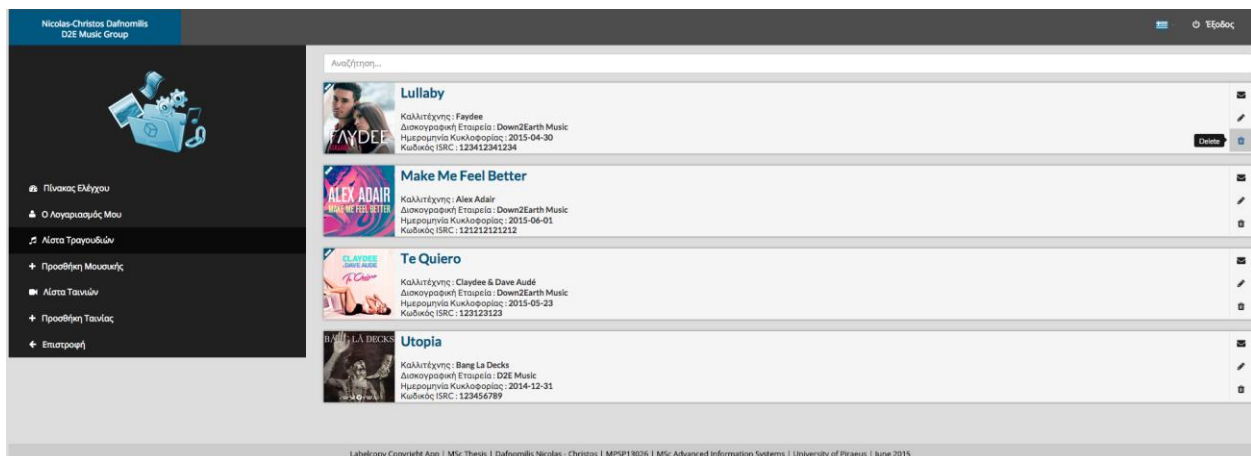
να εισάγει το δικό του κείμενο κατά την αποστολή.



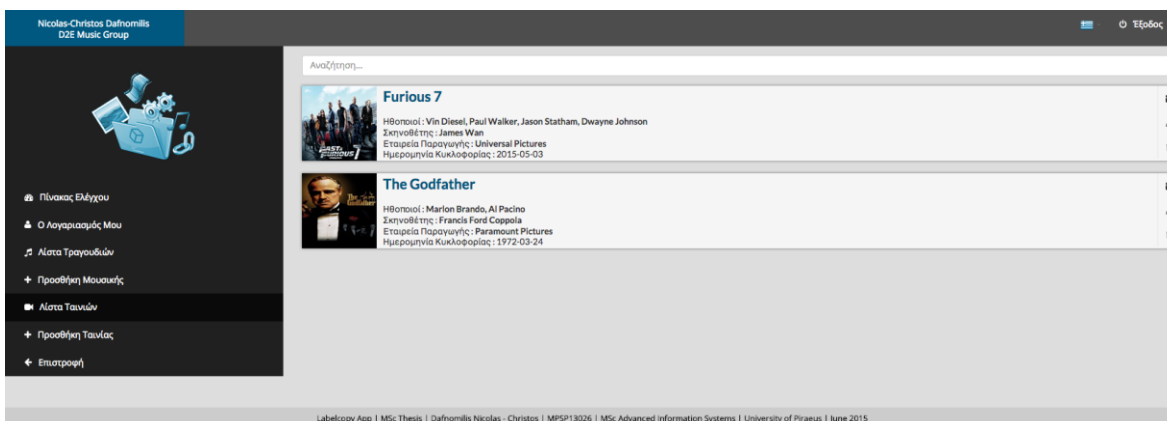
Εικόνα 24 – Εμφάνιση παραθύρου με τη πληροφορία του τραγουδιού που θα αποσταλεί

4.11 Διαγραφή τραγουδιού

Η αφαίρεση της κάθε καταχώρησης από τη βάση γίνεται κάνοντας κλικ στο πλήκτρο με το εικονίδιο του κάδου.



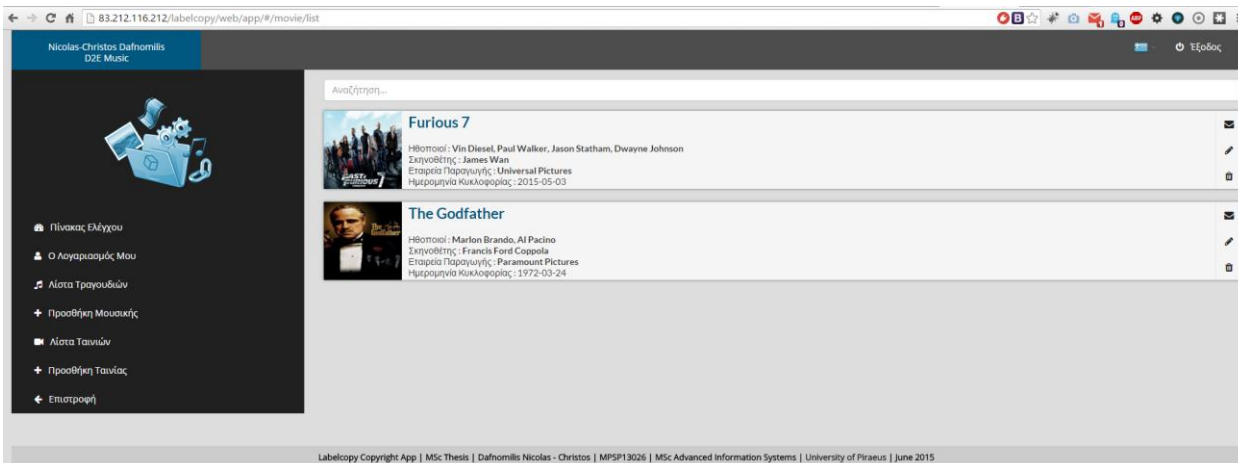
Εικόνα 25 – Διαγραφή καταχώρησης



Εικόνα 27 – Προβολή της λίστας ταινιών

4.14 Τροποποίηση ταινίας

Στη λίστα που εμφανίζονται οι ταινίες παρατηρείται ότι η κάθε εγγραφή στα δεξιά της φέρει τρεις επιλογές (ίδιο layout με τη λίστα των τραγουδιών). Η πρώτη είναι η λειτουργία αποστολής, η δεύτερη είναι η λειτουργία επεξεργασίας και η τρίτη είναι η λειτουργία διαγραφής της εγγραφής από τη βάση. Πηγαίνοντας τον κέρσορα στο κουμπί της επεξεργασίας (εικονίδιο «μολύβι») εμφανίζεται το μήνυμα “Edit” μέσω βοηθητικής σήμανσης τύπου “tooltip”.



Εικόνα 28 – Εμφάνιση της επιλογής τροποποίησης της εγγραφής και της σήμανσης tooltip

Κάνοντας κλικ στο edit ο χρήστης μεταφέρεται στην οθόνη επεξεργασίας της εγγραφής. Η οθόνη επεξεργασίας είναι παρόμοια με την αρχική καταχώρηση απλώς καλεί τα στοιχεία από τη βάση δεδομένων και ο χρήστης μπορεί να τα τροποποιήσει και να ανανεώσει την εγγραφή στη βάση κάνοντας κλικ στο πλήκτρο «Αποθήκευση». Εναλλακτικά με κλικ στο πλήκτρο «Ακύρωση» ο χρήστης μεταφέρεται αυτόματα στην οθόνη με τη λίστα εγγραφών.



Εικόνα 29 – Οθόνη τροποποίησης της εγγραφής ταινίας

4.15 Αποστολή ταινίας

Όπως και στην λειτουργία αποστολής των πληροφοριών της κάθε μουσικής εγγραφής έτσι και στη κατηγορία των ταινιών μία βασική λειτουργία, της εφαρμογής, που έχει ο χρήστης είναι να κάνει με ένα κλικ, αποστολή της κάθε εγγραφής, μέσω ηλεκτρονικού ταχυδρομείου (email). Επιλέγοντας το πρώτο πλήκτρο με το εικονίδιο «φάκελος» από τη δεξιά πλευρά κάθε εγγραφής εμφανίζεται ένα pop up παράθυρο που περιέχει μία φόρμα με τρία πεδία. Το πρώτο πεδίο είναι ένα textarea που εισάγεται η ηλεκτρονική διεύθυνση του παραλήπτη. Το δεύτερο πεδίο περιέχει τις πληροφορίες του αρχείου που θα αποσταλούν. Οι πληροφορίες αυτές αποτελούνται από τα πεδία:

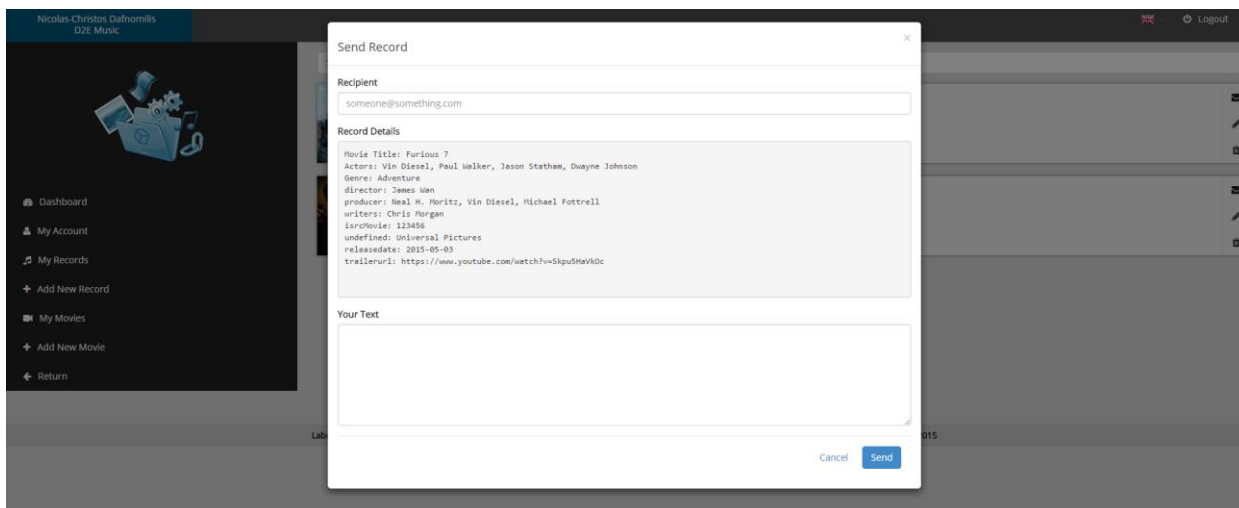
1. «Τίτλος»
2. «Ηθοποιοί»
3. «Είδος»
4. «Σκηνοθέτης»
5. «Παραγωγός»
6. «Σεναριογράφος»
7. «Κωδικός ISRC»
8. «Ημερομηνία Κυκλοφορίας»
9. «Διεύθυνση Trailer»
10. «Σχόλια»

Οι παραπάνω πληροφορίες καλούνται απευθείας από τη βάση δεδομένων και ο χρήστης δεν έχει δικαίωμα να τα τροποποιήσει κατά τη διαδικασία της αποστολής. Αυτό γίνεται κυρίως για να προστατευθεί η πληροφορία που αποστέλλεται.

Για να μπορεί όμως να στείλει κι ένα κείμενο της επιλογής του, ένα μήνυμα στον παραλήπτη έχει



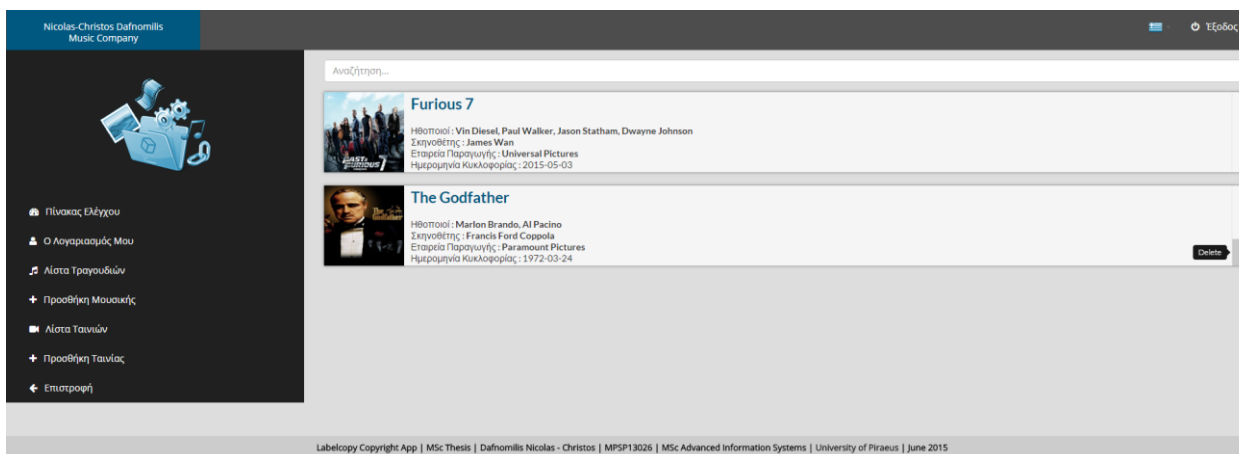
προσθεθεί το τρίτο πεδίο. Το τρίτο πεδίο είναι ένα textarea στο οποίο ο χρήστης έχει τη δυνατότητα να εισάγει το δικό του κείμενο κατά την αποστολή.



Εικόνα 30 – Οθόνη αποστολής πληροφοριών ταινίας

4.16 Διαγραφή ταινίας

Η αφαίρεση της κάθε καταχώρησης από τη βάση γίνεται κάνοντας κλικ στο πλήκτρο με το εικονίδιο του κάδου στη δεξιά πλευρά της καταχώρησης.

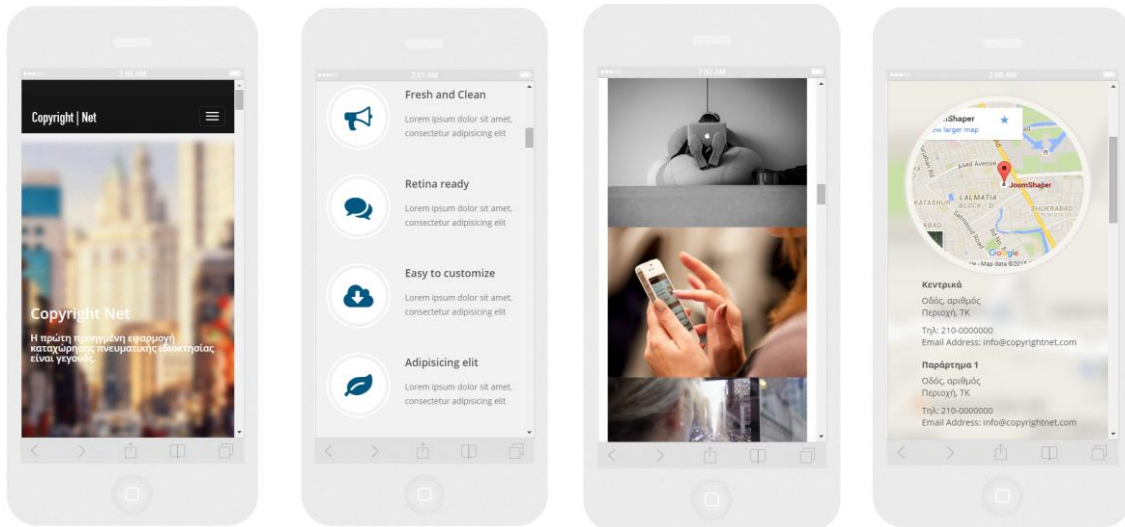


Εικόνα 31 – Οθόνη αποστολής πληροφοριών ταινίας

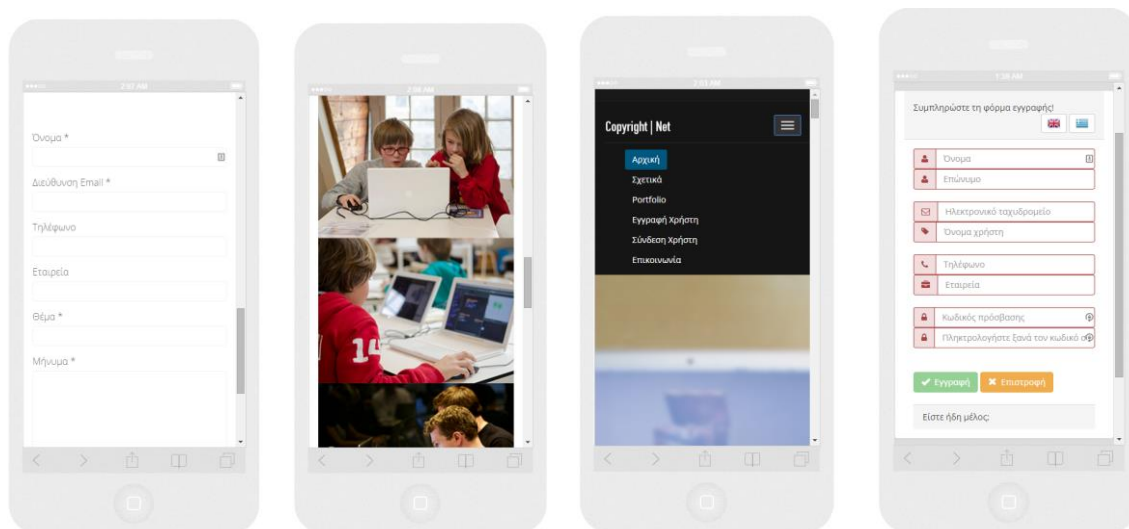


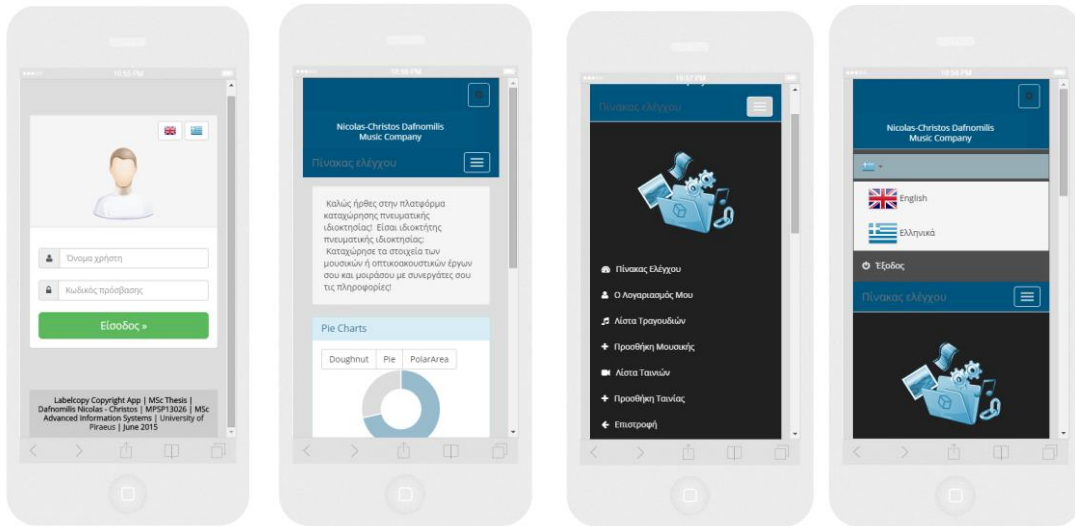
4.17 Συμβατότητα με φορητές συσκευές

Η εφαρμογή είναι πλήρως συμβατή με τις περισσότερες συσκευές και βασίζεται στη βιβλιοθήκη Bootstrap. Παρακάτω παρουσιάζονται οι οθόνες πλοήγησης στην εφαρμογή σε λειτουργία responsive για φορητές συσκευές τύπου smartphone με ανάλυση οθόνης 1334 x 740 pixels και οθόνη μεγέθους 4,7 ιντσών.



Εικόνα 32 - Οθόνες της βασικής ιστοσελίδας της εφαρμογής

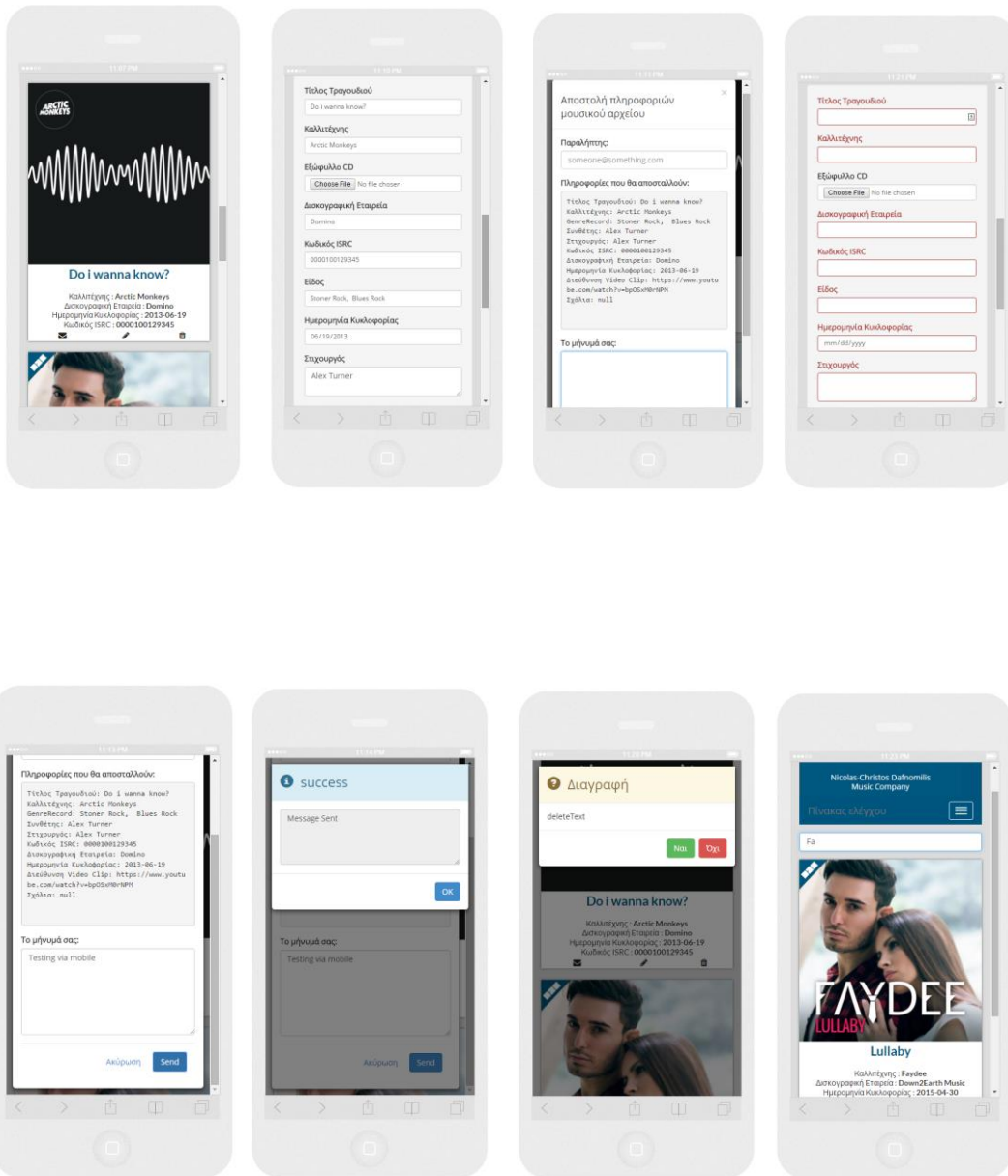


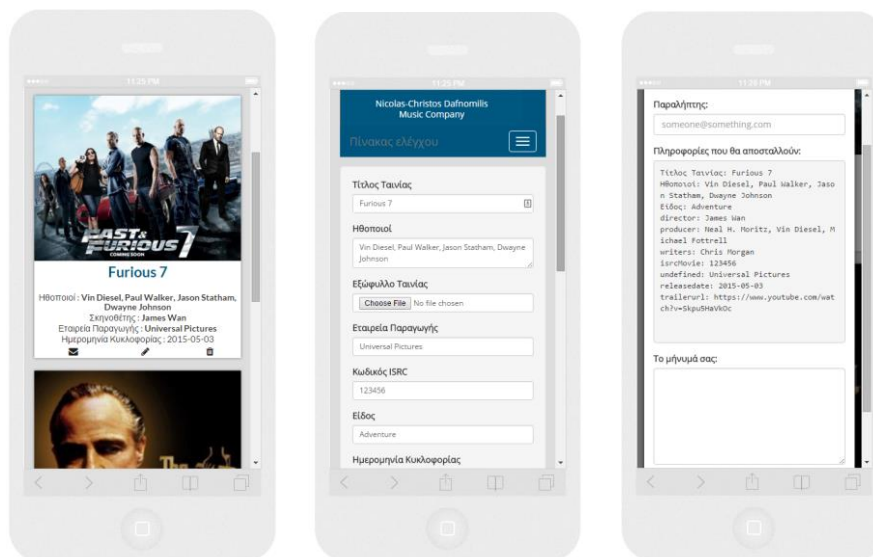


Οι δοκιμές για πλήρη υποστήριξη των φορητών συσκευών έγινε με τη βοήθεια του online εργαλείου responsinator.



Εικόνα 33 – Οθόνες εμφάνισης στατιστικών στοιχείων για τις εγγραφές της εφαρμογής





Επίσης η εφαρμογή είναι πλήρως συμβατή με όλες τις αναλύσεις που υποστηρίζουν οι οθόνες των περισσότερων tablets που υπάρχουν στην αγορά.



5. Εγχειρίδιο προγραμματιστή (developer manual) :

5.1 Προγράμματα και τεχνολογίες που χρησιμοποιήθηκαν

- Web Services
- PHP
- Slim Framework
- Bootstrap v3
- JavaScript (Angular JS)
- SQL
- Atom editor
- Photoshop
- XAMPP

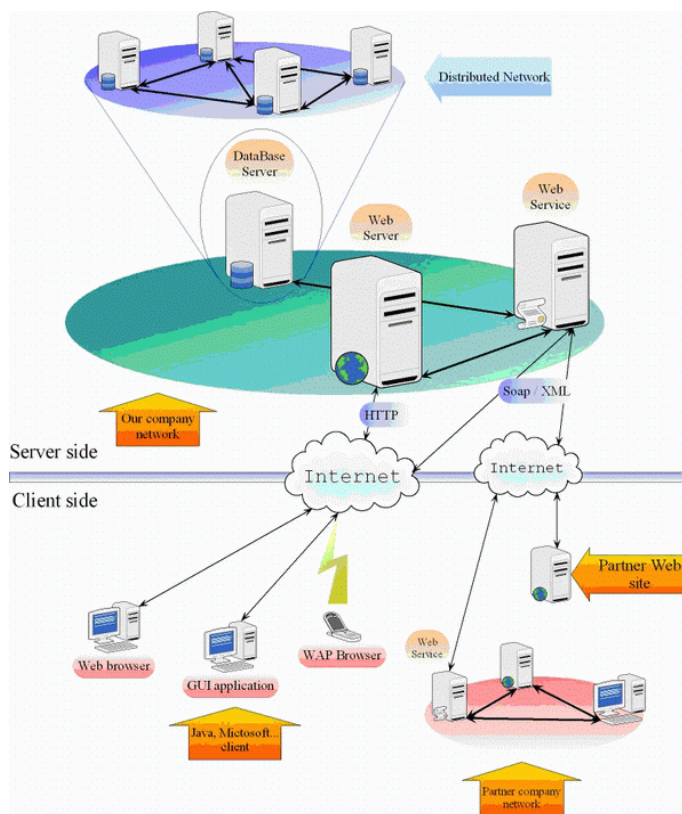
5.1.1 Σχετικά με τα Web Services

Η έννοια των web services διαφέρει τα τελευταία χρόνια και δεν υπάρχει μια κοινώς αποδεκτή έννοια ακόμα. Οι ορισμοί που δίνονται στη τεχνολογία των web services, διαφέρει κατά πολύ από εταιρεία σε εταιρεία. Κάθε εταιρεία που δημιουργεί, αναπτύσσει και παράγει διάφορα εργαλεία για τα web services, δίνει και διαφορετικό ορισμό ως προς αυτά, τη χρήση τους και τις λειτουργίες τους. Ο επικρατέστερος ορισμός μέχρι και σήμερα έρχεται από την IBM. Αυτός ο ορισμός αναφέρει ότι τα web services, είναι η τεχνολογία που επιτρέπει σε διάφορες εφαρμογές την μεταξύ τους επικοινωνία, ανεξάρτητα από τη πλατφόρμα ανάπτυξής του και ανεξάρτητα από τη γλώσσα προγραμματισμού στην οποία έχουν δημιουργηθεί. Ένα web service, είναι το σύνολο των λειτουργιών οι οποίες μπορούν να προσπελαστούν μέσω διαδικτύου, είτε μέσω πρότυπων μηνυμάτων XML. Όλα τα web services, αναφέρει η Microsoft, έχουν τρία κοινά χαρακτηριστικά.

Σαν τεχνολογία είναι ιδιαίτερα χρήσιμα ως προς τη λειτουργικότητα για τους χρήστες του διαδικτύου.

Τα web services δίνουν τη δυνατότητα στο χρήστη να μπορεί να αναπτύξει μία εφαρμογή η οποία να μπορεί να επικοινωνεί μαζί τους.

Τα web services καταχωρούνται και είναι εύκολα προσπελάσιμα από τους χρήστες. Στην ουσία τα web services είναι ένα σύνολο συστημάτων που αποτελείται από πολλά υπολογιστικά συστήματα τα οποία επικοινωνούν μεταξύ τους μέσω του διαδικτύου και όλα μαζί δημιουργούν ένα μεγαλύτερο σύστημα. Αυτή η λειτουργία των web services, τα κάνει να υπερτερούν σε σύγκριση με τις προηγούμενες κατακευματισμένες τεχνολογίες. Τα πλεονεκτήματα των web services είναι πάρα πολλά. Το μεγαλύτερο πλεονέκτημα είναι ότι δίνουν τη δυνατότητα για ευκολότερο χειρισμό των δεδομένων. Παλιότερα η επικοινωνία μεταξύ των κατακευματισμένων τεχνολογιών ήταν η μικρή συνδεσιμότητα μεταξύ των συστημάτων. Τα web services χρησιμοποιούν τη γλώσσα XML η οποία μπορεί να περιέχει πληροφορία για οποιαδήποτε δεδομένα που θα ανταλλάσσονται μεταξύ των συστημάτων. Έτσι με αυτό το τρόπο αποφεύγεται η ισχυρή συνδεσιμότητα του παρελθόντος και γίνεται χρήση της χαλαρής συνδεσιμότητας που καθιστούν τα web services πιο ευέλικτα.



Εικόνα 34 – Διάγραμμα παρουσίασης της δομής των web services

Τα web services (Application Services) αποτελούν ένα από τα σημαντικότερα τμήματα του σύγχρονου προγραμματισμού. Είναι εύκολα στη κατανόηση, στην υλοποίηση και επιτρέπουν στο μέσο χρήστη – προγραμματιστή να δημιουργήσει το δικό του web service ακόμη και για κάποια προσωπική ιστοσελίδα. Υπάρχουν δύο διαφορετικές μέθοδοι στη δημιουργία των web services. Θα εξετάσουμε και τις δύο και θα αναλύσουμε κάποια σημεία-κλειδιά ως προς τη χρησιμότητα της κάθε μεθόδου. Οι μέθοδοι αυτές είναι η REST και η SOAP μέθοδος. Η SOAP ή αλλιώς Simple Object Access Protocol βασίζεται σε XML και είναι πολύ εύκολο να αναγνωσθεί. Είναι ουσιαστικά βασισμένη στο απλό πρωτόκολλο XML για να ανταλλάξει πληροφορία μεταξύ δύο διαφορετικών γλωσσών προγραμματισμού. Η δεύτερη μέθοδος είναι η μέθοδος REST (Representational State Transfer) η οποία είναι μία αρχιτεκτονική που τρέχει μέσω του πρωτοκόλλου HTTP. Με τη μέθοδο REST τα web services μπορούν να καλούνται είτε μέσω XML είτε μέσω JSON αρχείων. Η διαδικασία δημιουργίας ενός web service με τη μέθοδο REST είναι αρκετά γρήγορη και εύκολη σε σύγκριση με άλλες μεθόδους. Η μέθοδος REST υποστηρίζει σχεδόν όλες τις κλήσεις συστήματος της HTTP. Αυτές είναι οι GET, POST, PUT και DELETE οι οποίες χρησιμοποιούνται ανάλογα τις ανάγκες του web service.

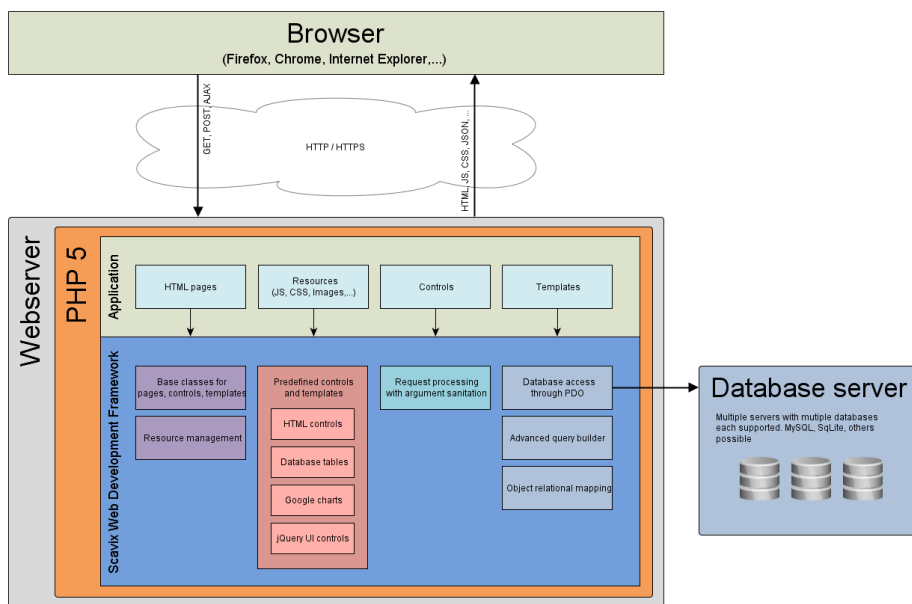
5.1.2 Σχετικά με τη PHP

Η PHP είναι μία γλώσσα προγραμματισμού από την πλευρά του διακομιστή (server-side), σχεδιασμένη για χρήση στο web. Η ονομασία της άλλαξε σύμφωνα με τη σύμβαση GNU από “Personal Home Page” σε “PHP Hypertext Preprocessor” (προεπεξεργαστής κειμένου PHP). Η PHP συγκριτικά με τους ανταγωνιστές της Perl, JSP, ASP προσφέρει υψηλή απόδοση, μεταφερσιμότητα, χαμηλό κόστος και διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων



δεδομένων. Μέσα σε μία HTML σελίδα υπάρχει η δυνατότητα ενσωμάτωσης PHP κώδικα, που θα εκτελείται κάθε φορά που η σελίδα δέχεται επίσκεψη. Ο PHP κώδικας μεταφράζεται στον web server και δημιουργεί HTML ή άλλη έξοδο που θα εμφανιστεί στο χρήστη.

Η ιστορία της PHP ξεκινάει το 1994, όταν ο τότε φοιτητής Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού Perl ένα απλό script με όνομα rhp.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα με στατιστικά στοιχεία για τα άτομα που έβλεπαν το βιογραφικό του σημείωμα σε κάποια ιστοσελίδα. Αργότερα αυτό το script το μοίρασε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να το εξελίξει και προσθέσει μερικές ακόμη λειτουργίες. Η γλώσσα εκείνη την εποχή στη πρώιμη μορφή της είχε την ονομασία PHP/FI. Το 1997 η PHP/FI πέρασε στην έκδοση 2.0, βασιζόμενη στη γλώσσα προγραμματισμού C έχοντας καταφέρει να χρησιμοποιείται σε περισσότερους από 50.000 ιστότοπους. Στη συνέχεια του 1997 οι Andi Gutmans και Zeev Suraski αποφάσισαν να ξαναγράψουν από την αρχή την γλώσσα βασιζόμενοι όμως κατά πολύ στην PHP/FI 2.0. Το αποτέλεσμα ήταν να δημιουργηθεί η έκδοση 3.0 της PHP η οποία θύμιζε περισσότερο τη μορφή που έχει σήμερα. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Το 1994 κυκλοφόρησε η έκδοση 4 και δέκα χρόνια αργότερα, το 2004 κυκλοφόρησε η έκδοση 5 που ισχύει μέχρι και σήμερα. Η ανάπτυξη της PHP δε σταματάει εδώ όμως καθώς έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για τους προγραμματιστές που θέλουν να τη χρησιμοποιήσουν.



Εικόνα 35 – Παρουσίαση χρήσης της Php

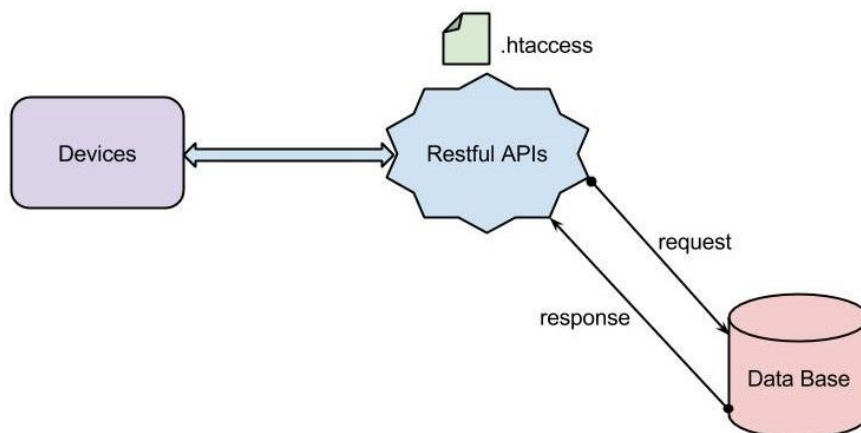
Η PHP καθιερώθηκε ως η καλύτερη γλώσσα προγραμματισμού για τη δημιουργία σελίδων στο web με δυναμικό περιεχόμενο και η χρήση της πολλαπλασιάστηκε τα τελευταία χρόνια. Η λειτουργία μίας δυναμικής ιστοσελίδας στο web η οποία έχει υλοποιηθεί με PHP είναι σχετικά απλή. Η σελίδα περνάει από επεξεργασία από κάποιον διακομιστή (συνήθως από τον Apache) ώστε να δημιουργηθεί το τελικό περιεχόμενο που θα σταλεί στον περιηγητή και θα εμφανίσει κώδικα HTML.



5.1.3 Σχετικά με το Slim Framework

Το Slim Framework είναι ένα micro framework που βασίζεται στη γλώσσα προγραμματισμού PHP. Δημιουργήθηκε το 2010 από τον Josh Lockhart και θεωρείται ως ένα από τα καλύτερα micro frameworks για εφαρμογές τύπου web services. Η αρχή έγινε το Σεπτέμβριο του 2010 όταν ο Lockhart δημιούργησε το πρώτο repository για το Slim στο GitHub ανεβάζοντας όλη τη δουλειά που είχε κάνει τους προηγούμενους μήνες για να βγει η πρώτη σταθερή έκδοση και να γίνει διαθέσιμη στο ευρύ κοινό. Η αρχική ιδέα ήταν να δημιουργηθεί ένα framework που θα επέτρεπε πολύ γρήγορα στον προγραμματιστή να δημιουργεί τα δικά του applications χωρίς να ασχολείται εκτενώς με το server side κομμάτι που μέχρι τότε γινόταν από πανίσχυρα frameworks όπως τα Symfony, Cake και Codelgnitor. Αυτά τα frameworks απαιτούν από τον προγραμματιστή χρόνο για να τα μελετήσει, είναι μεγάλα σε εύρος και πολλές φορές αποτελούν κατασταλτικό παράγοντα κατά τη δημιουργία ενός νέου προγράμματος λόγω της πολυπλοκότητας που φέρουν. Το σημαντικό για τον Lockhart ήταν να δημιουργηθεί ένα framework που θα ήταν προσεγμένο, απλό στη χρήση, με σκοπό να είναι παραγωγικό και να κάνει τα απολύτως απαραίτητα. Έτσι το Slim θα έπρεπε να κάνει λήψη του HTTP Request, να επικαλείται τον κατάλληλο κώδικα για τον controller της εκάστοτε εφαρμογής και στη συνέχεια να επιστρέφει ένα HTTP Response.

Το όνομά Slim προήλθε από μία σύντομη έρευνα που έκανε ο Lockhart σχετικά με τις ονομασίες που θα μπορούσαν να μεταφέρουν το νόημα του framework που έγραψε. Ήθελε μία λέξη που θα δήλωνε το πόσο απλό είναι να δημιουργείς web services με το framework του με αποτέλεσμα να καταλήξει στην ονομασία «Slim». Το Slim δίνει τη δυνατότητα στον προγραμματιστή να δημιουργεί τις δικές του web based ισχυρές εφαρμογές χωρίς να τον επηρεάζουν τα χαμηλά στρώματα του πρωτοκόλλου HTTP. Πολύ σημαντικά θεωρούνται τα πρόσθετα χαρακτηριστικά που παρέχει το Slim στον προγραμματιστή. Αυτά είναι το middleware το οποίο είναι ένα εργαλείο του Slim που μεσολαβεί πριν φτάσει ο κώδικας στον διακομιστή, οι βοηθητικές μέθοδοι για HTTP caching και η ικανότητα να χρησιμοποιήσει κανείς custom views για διάφορα πρότυπα. Με τον όρο micro framework εννοούμε μία συλλογή από απλά εργαλεία που είναι απαραίτητα στον προγραμματιστή και τον διευκολύνουν κατά τη διαδικασία της ανάπτυξης της εφαρμογής. Το Slim κερδίζει έδαφος στις προτιμήσεις των προγραμματιστών λόγω του ότι έχει ιδιαίτερα λίγες κλάσεις συγκριτικά με άλλα frameworks όπως το Silex που βασίζεται στο Symfony. Εκτός αυτού ο κώδικας του Slim είναι ιδιαίτερα απλός και στην ανάγνωση αλλά και στην κατανόηση, αποφεύγοντας συντακτικές συντομεύσεις που δυσκολεύουν τους προγραμματιστές καθώς και δύστροπες αρχιτεκτονικές έτσι ώστε κάθε προγραμματιστής PHP να μπορεί να καταλάβει τον πηγαίο κώδικα του Slim. Σήμερα το Slim μετράει πάνω από 3000 αστέρια στο GitHub και περιλαμβάνεται στη λίστα με τα πιο προσεγμένα PHP repositories κατέχοντας την 13^η θέση.



Εικόνα 36 – Παρουσίαση της δομής του Slim Framework

5.1.4 Σχετικά με το Bootstrap

Το Bootstrap αποτελεί μία μεγάλη βιβλιοθήκη εργαλείων ανοιχτού κώδικα. Μπορεί να χρησιμοποιηθεί από οποιονδήποτε και είναι ίσως ένα από τα πιο δημοφιλή έργα ανάπτυξης των τελευταίων ετών στη κατηγορία του ελεύθερου λογισμικού. Περιέχει την HTML και CSS για τις μορφοποιήσεις τυπογραφίας, ενσωματώνει διάφορα κουμπιά πλοήγησης και άλλα στοιχεία του web περιβάλλοντος καθώς επίσης και πολλές επεκτάσεις JavaScript.

Πίσω από την έλευση του Bootstrap βρίσκονται οι προγραμματιστές Mark Otto και Jacob Thornton της εταιρείας Twitter. Το Bootstrap αναπτύχθηκε ως ένα εσωτερικό εργαλείο της εταιρείας με σκοπό να κάνει τον ιστότοπο του Twitter όσο πιο ευέλικτο και γρήγορο μπορούσε να γίνει. Η πρώτη εγκατάσταση της βιβλιοθήκης του Bootstrap έγινε κατά τη διάρκεια της πρώτης Hackweek του Twitter. Μετά την ολοκλήρωση της παρουσίασης όλο και περισσότεροι συνάδελφοί του Mark Otto ξεκίνησαν να χρησιμοποιούν τη βιβλιοθήκη. Μετά από αρκετές σκέψεις, το 2011 αποφασίστηκε εν τέλει να κυκλοφορήσει το Bootstrap ως λογισμικό ανοιχτού κώδικα από το Twitter. Το 2012 ήταν το πιο περιζήτητο και δημοφιλές έργο ανάπτυξης μεταξύ των προγραμματιστών στο GitHub.

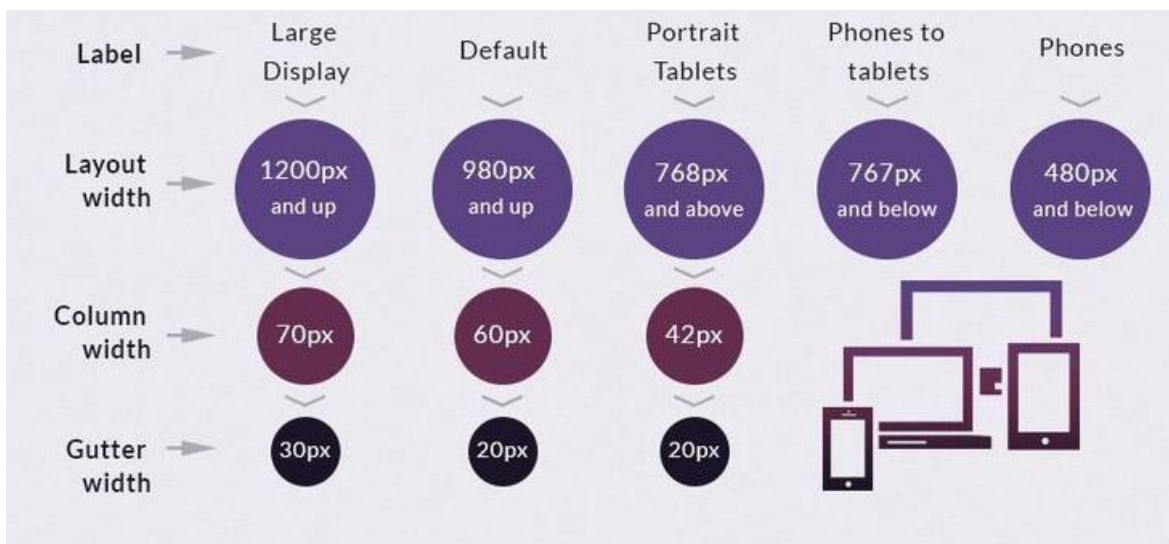
Ένα πολύ σημαντικό πλεονέκτημα του Bootstrap είναι η συμβατότητα του από όλους τους browsers και όλες τις συσκευές και περιλαμβάνει αρκετά χαρακτηριστικά της html και του Css3 με αποτέλεσμα να είναι άκρως λειτουργικά τα πακέτα εργαλείων που προσφέρει. Η έκδοση 2.0 του Bootstrap έφερε τα πάνω κάτω στη κοινότητα των προγραμματιστών για φορητές συσκευές καθώς η νέα αναβαθμισμένη βιβλιοθήκη του Bootstrap υποστήριζε τη λειτουργία responsive για όλους τους τύπους οθόνης. Αυτό σήμαινε την εδραίωση του Bootstrap ως πρώτη επιλογή για τον προγραμματισμό σε φορητές συσκευές (smartphones και tablets).



Εικόνα 37 – Διάφοροι τύποι οθόνης που υποστηρίζονται από το Bootstrap

Η δυνατότητα που δίνει το Bootstrap στον προγραμματιστή να μετατρέπει απευθείας το δημιούργημά του συμβατό με όλες τις οθόνες είναι ένας παράγοντας που έκανε το Bootstrap τόσο δημοφιλές. Είναι πολύ σημαντικό η ιστοσελίδα να μπορεί να προσαρμόζεται στην οθόνη της συσκευής που χρησιμοποιούμε. Με τη χρήση του responsive web design οι ιστοσελίδες μπορούν να λειτουργούν και ως εφαρμογές όταν ανοίγονται από υψηλής τεχνολογίας φορητές συσκευές. Αυτό είναι ιδιαίτερα σημαντικό για το λόγο του ότι τα τελευταία χρόνια έχει παρατηρηθεί ραγδαία αύξηση χρήσης του διαδικτύου από φορητές συσκευές και πτώση της περιήγησης από υπολογιστές.

Η ανάπτυξη δε σταμάτησε εκεί όμως, καθώς η αρμόδια ομάδα πίσω από το bootstrap ενθαρρύνει την κοινότητα των προγραμματιστών να συμμετέχουν ενεργά στην ανάπτυξη νέων εργαλείων με σκοπό την βελτιστοποίηση του Bootstrap.



Εικόνα 38 – Παρουσίαση responsive στοιχείων του Bootstrap

Το Bootstrap αποτελείται από μία σειρά αρχείων stylesheet που περιέχουν τη πληροφορία του κάθε εργαλείου που υπάρχει στη βιβλιοθήκη. Το Bootstrap μπορεί να παραμετροποιηθεί από τους προγραμματιστές ώστε να επιλέγεται το ποια στοιχεία της βιβλιοθήκης θα κρατηθούν και θα χρησιμοποιηθούν στο έργο τους και ποια όχι.

Η δομή του αρχικού stylesheet του Bootstrap αποτελείται από σταθερό πλάτος μήκους 940 pixels. Εάν επιθυμεί ο κάθε προγραμματιστής μπορεί να τροποποιήσει την διάσταση αυτή. Παρόλο που το default στυλ είναι στα 940 pixels το Bootstrap περιλαμβάνει διάφορες παραλλαγές για οθόνες φορητών συσκευών είτε σε portrait λειτουργία είτε σε landscape λειτουργία με υποστήριξη όλων των οθονών είτε είναι υψηλής είτε χαμηλής ανάλυσης οθόνες. Κάθε παραλλαγή ρυθμίζει το πλάτος των στηλών. Εκτός των παραπάνω το Bootstrap περιέχει πολλά συστατικά της JavaScript με τη μορφή των jQuery plugins. Αυτά μπορούν να επεκτείνουν κατά πολύ τη λειτουργικότητα διαφόρων στοιχείων της διασύνδεσης. Μερικά από αυτά είναι η υποστήριξη των modals, extended menus, scrollspy, tabs, tooltips, popover, alert, button, collapse και carousel. Η ενσωμάτωση και κλήση του Bootstrap είναι πολύ απλή διαδικασία. Ο προγραμματιστής απλώς κατεβάζει τα αρχεία του Bootstrap και τα καλεί από το header της HTML.

5.1.5 Σχετικά με τη JavaScript

Η γλώσσα προγραμματισμού JavaScript δημιουργήθηκε αρχικά από τον Brendan Eich, έναν μηχανικό της εταιρείας Netscape. Η αρχική της ονομασία ήταν Mocha. Στη πορεία, από Mocha μετονομάστηκε σε LiveScript για να καταλήξει στην τελική μέχρι σήμερα ονομασία, JavaScript. Πήρε το όνομα αυτό κυρίως λόγω του ότι η ανάπτυξή της επηρεάστηκε περισσότερο από τη γλώσσα προγραμματισμού Java, ενώ στη πραγματικότητα οι δύο γλώσσες έχουν ελάχιστα κοινά στοιχεία μεταξύ τους. Η πρώτη κυκλοφορία στην αγορά, έγινε το 1995 με την ονομασία LiveScript και ήταν η έκδοση beta της LiveScript που συνόδευε το πολύ γνωστό για την εποχή πρόγραμμα περιήγησης στο Web, το Netscape Navigator. Λίγους μήνες αργότερα, η Sun Microsystems, ανακοίνωσε την ενσωμάτωση της γλώσσας προγραμματισμού LiveScript στο Netscape Navigator 2 και τη



μετονομασία της σε JavaScript. Η JavaScript είναι πολύ δημοφιλής client-side γλώσσα προγραμματισμού και χρησιμοποιείται κυρίως για εκτέλεση κώδικα σε ιστοσελίδες.

Η JavaScript έχει γίνει μία από τις πιο δημοφιλείς γλώσσες προγραμματισμού ηλεκτρονικών υπολογιστών στον Παγκόσμιο Ιστό (Web). Αρχικά, όμως, πολλοί επαγγελματίες προγραμματιστές υποτίμησαν τη γλώσσα διότι το κοινό της ήταν ερασιτέχνες συγγραφείς ιστοσελίδων και όχι επαγγελματίες προγραμματιστές (και μεταξύ άλλων λόγων). Με τη χρήση της τεχνολογίας Ajax, η JavaScript γλώσσα επέστρεψε στο προσκήνιο και έφερε πιο επαγγελματική προσοχή προγραμματισμού. Το αποτέλεσμα ήταν ένα καινοτόμο αντίκτυπο στην εξάπλωση των πλαισίων και των βιβλιοθηκών, τη βελτίωση προγραμματισμού με JavaScript, καθώς και αυξημένη χρήση της JavaScript έξω από τα προγράμματα περιήγησης στο Web. Τον Ιανουάριο του 2009, το έργο Common JS ιδρύθηκε με στόχο τον καθορισμό ενός κοινού προτύπου βιβλιοθήκης κυρίως για την ανάπτυξη της JavaScript έξω από το πρόγραμμα περιήγησης και μέσα σε άλλες τεχνολογίες (π.χ. Server-side).

Η JavaScript δεν θα πρέπει να συγχέεται με την γλώσσα προγραμματισμού Java. Αυτές οι δύο γλώσσες είναι τελείως διαφορετικές μεταξύ τους και γίνεται χρήση αυτών σε διαφορετικές εφαρμογές. Η Java και η JavaScript επηρεάστηκαν σε πολύ μεγάλο βαθμό από τη γλώσσα προγραμματισμού C. Η JavaScript είναι script γλώσσα που βασίζεται σε αντικείμενα. Είναι λοιπόν αντικειμενοστραφής και χρησιμοποιείται κυρίως από τη πλευρά του πελάτη.

Σήμερα χρησιμοποιείται ευρέως η JavaScript και οι διάφορες βιβλιοθήκες τις κυρίως στο web όπου προσθέτει δυναμικότητα στον ιστότοπο που εφαρμόζεται. Η JavaScript τρέχει στον περιηγητή και προστίθεται σε ένα υπάρχον έγγραφο HTML. Προσθέτει ειδικές ετικέτες στην HTML και μέσω αυτών των ετικετών ο περιηγητής παίρνει εντολή να «τρέξει» διάφορα scripts.

Ως γλώσσα προγραμματισμού η JavaScript γίνεται όλο και πιο δημοφιλής. Μάλιστα αναφέρεται ότι στο μέλλον θα είναι η αποκλειστική γλώσσα για εφαρμογές φορητών συσκευών και γενικότερα για διαδικτυακές εφαρμογές. Μέχρι στιγμής έχουν υλοποιηθεί πανίσχυρες βιβλιοθήκες, που διευκολύνουν την ανάπτυξη των web based εφαρμογών και των φορητών συσκευών (smartphones & tablets). Συγκριτικά με τις άλλες γλώσσες η JavaScript είναι πολύ γρήγορη. Επιτρέπει άμεση επικοινωνία μεταξύ του server και του client. Εκτός αυτού είναι συμβατή με όλες τις άλλες γλώσσες προγραμματισμού στο web (PHP, python και άλλες). Η JavaScript, υποστηρίζει και offline λειτουργία, δεν χρειάζεται υψηλή κατανάλωση πόρων του συστήματος για να λειτουργήσει με αποτέλεσμα οι περισσότερες συσκευές να είναι συμβατές με αυτή.

Τα τελευταία χρόνια όλο και μεγαλύτερες εταιρίες πληροφορικής στηρίζουν το λογισμικό τους στην JavaScript καθώς έχουμε περάσει σε μία εποχή που τα περισσότερα services είναι μέσω του web και το cloud έχει ξεκινήσει να αποτελεί κομμάτι της καθημερινότητάς μας.

Η JavaScript αποτελείται από μερικές πολύ ενδιαφέρουσες και ισχυρές βιβλιοθήκες που αναδεικνύουν την ευχρηστία της γλώσσας και τη συνεισφορά της στον ιντερνετικό προγραμματισμό. Μερικές από τις βιβλιοθήκες αυτές είναι οι:

- 1) Angular JS
- 2) Node.JS
- 3) Backbone.JS
- 4) PhoneGap

Στην παρούσα εργασία χρησιμοποιήθηκε η Angular JS για την υλοποίηση της εφαρμογής σε συνεργασία με την γλώσσα προγραμματισμού PHP.



5.1.6 Σχετικά με την Angular JS

Η Angular JS είναι ένα framework ανοιχτού κώδικα της εταιρείας Google. Μέσω της χρήσης της Angular απλοποιείται σε μεγάλο βαθμό η δυσκολία του προγραμματισμού για το web.

Η Angular JS είναι ένα framework ανοιχτού λογισμικού της Google. Σχεδιάστηκε με στόχο να:

1. Συμβάλει στην αποδέσμευση του DOM από την λογική της εφαρμογής, με σκοπό να βελτιωθεί ο τρόπος που ελέγχεται ο κώδικας.
2. Έχει ο έλεγχος της εφαρμογής την ίδια σημασία με τη διαδικασία της υλοποίησης. Αυτό είναι σημαντικό καθώς οι δοκιμές επηρεάζονται σε μεγάλο βαθμό από τη δομή του κώδικα.
3. Αποσυνδέει τη πλευρά του client, δηλαδή τον browser από τη πλευρά του server. Με αυτό τον τρόπο επιτυγχάνεται ταχύτερη ανάπτυξη των εφαρμογών καθώς δουλεύονται σε διαφορετικούς ή και ίδιους χρόνους.
4. Ακολουθεί το μοντέλο MVC της ανάπτυξης λογισμικού και συμβάλει στη χαλαρή διασύνδεση μεταξύ της παρουσίασης των δεδομένων και της λογικής.
5. Μειώνει δραματικά τον κώδικα που χρειάζεται να εγγραφεί για το back end τμήμα του λογισμικού και να βοηθήσει οι web εφαρμογές να είναι μικρότερες σε όγκο με στόχο την ταχύτερη λειτουργία τους.

Η Angular JS συνδέει την HTML σε αντικείμενα JavaScript. Δηλαδή views με models. Όταν λοιπόν τροποποιείται το μοντέλο, τότε η σελίδα ανανεώνεται αυτόματα. Το ίδιο ισχύει και για την αντίστροφη διαδικασία. Αυτό εξοικονομεί χρόνο στον προγραμματιστή που δεν χρειάζεται να κάνει και στις δύο μεριές τις τροποποιήσεις.

Είναι μία γλώσσα προγραμματισμού front-end κατάλληλη για γραφικά περιβάλλοντα κυρίως για εφαρμογές για το web. Το μοντέλο MVC που ακολουθεί την κάνει ευέλικτη και διαχωρίζει την ευθύνη για κάθε κομμάτι της εφαρμογής σε τρία διαφορετικά μέρη. Αυτά είναι:

1. Οι χειριστές (Controllers)
2. Τα μοντέλα (Models)
3. Οι απεικονίσεις (Views)

Η Angular κατά την ενσωμάτωσή της σε μία εφαρμογή ξεκινάει διαβάζοντας το DOM της εκάστοτε σελίδας και αναγνωρίζοντας ένα σύνολο από τις προκαθορισμένες ιδιότητες που ενσωματώνονται στα τμήματα του DOM.

Όπως αναφέρθηκε παραπάνω ένα πολύ βασικό χαρακτηριστικό της Angular JS είναι η δέσμευση των δεδομένων διπλής κατεύθυνσης (two-way data binding) και η εξοικονόμηση χρόνου στον προγραμματιστή καθώς όταν αλλάζει η μία πλευρά, αλλάζει αυτόματα και η άλλη.

Για να ξεκινήσει κανείς με την Angular, θα πρέπει να φορτώσει την βιβλιοθήκη της Angular στη σελίδα και στη συνέχεια να εισάγει την ιδιότητα ng-app μέσα σε ένα div.

```
<div class="view-container" ng-app="labelcopy">
...
</div>
```

Για την ανάπτυξη μίας εφαρμογής, το Angular ορίζει τέσσερα είδη ενότητων που το καθένα εξυπηρετεί και διαφορετικό σκοπό.



Αυτά είναι οι χειριστές (controllers), οι υπηρεσίες (services), οι οδηγίες ή αλλιώς ντιρεκτίβες (directives) και οι σταθερές (constants).

Πιο αναλυτικά, οι χειριστές (controllers) είναι τα τμήματα κώδικα που ενώνουν τα δεδομένα με τη λογική της εφαρμογής μέσω του model. Το model παρέχεται στους controllers μέσω της \$scope. Ό,τι προστίθεται στη συνέχεια στο αντικείμενο \$scope του controller, μπορεί να χρησιμοποιηθεί στα HTML αρχεία της εφαρμογής.

Οι υπηρεσίες (services) είναι τα τμήματα του κώδικα που στέλνουν request για λήψη των δεδομένων από τον εξυπηρετητή (server) αλλά εκτός από αυτό, ορίζουν και διάφορα πολύπλοκα και σύνθετα κομμάτια αλγορίθμων ως μέσο μεταφοράς των δεδομένων τους μέσω των διάφορων controllers.

Οι ντιρεκτίβες (directives) ορίζουν τις οντότητες που εκθέτουν μία διεπαφή, παράγουν δεδομένα και ορίζουν το δικό τους view. Αποτελούν μία από τις πιο σημαντικές λειτουργίες της Angular JS.

Οι σταθερές είναι υποπεριπτώσεις υπηρεσιών που ορίζουν ένα σύνολο τιμών που ανταποκρίνονται σε συγκεκριμένα keys και μπορούν να διαμοιράζονται σε διάφορα επιμέρους τμήματα της εφαρμογής στα οποία είναι απαραίτητα.

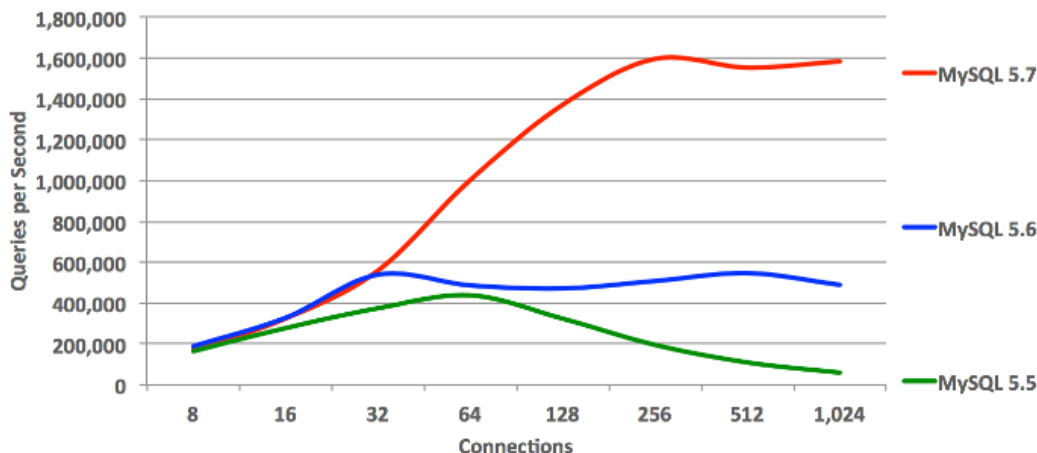
5.1.7 Σχετικά με την SQL

Η γλώσσα SQL (Structured Query Language) είναι ίσως η πιο διαδομένη γλώσσα που χρησιμοποιείται στην επικοινωνία μεταξύ χρηστή και σχεσιακών Βάσεων Δεδομένων. Πρόκειται για μια γλώσσα στην οποία ο χρήστης διατυπώνει διάφορα αιτήματα και το DBMS αναλαμβάνει να τα ικανοποιήσει επιστρέφοντας αποτελέσματα στο σύστημα. Η SQL δίνει την δυνατότητα στο χρηστή να δημιουργεί, να τροποποιεί τους πίνακες της βάσης καθώς και να αναζητά πληροφορίες από τη βάση εφαρμόζοντας διάφορα κριτήρια. Ένα υποσύνολο των Structured Query Languages είναι η MySQL είναι η πιο γνωστή και διαδεδομένη μορφή βάσεων δεδομένων. Ανήκει στη κατηγορία των προγραμμάτων ανοικτού λογισμικού, δηλαδή διανέμεται δωρεάν, και είναι διαθέσιμη σε όλους όσους θέλουν να υλοποιήσουν εφαρμογές. Είναι ταχύτερη και προτιμάται σε σχέση με άλλους τύπους βάσεων δεδομένων λόγω της υψηλής συμβατότητάς της. Διανέμεται δωρεάν μέσω του επίσημου ιστότοπου: www.mysql.com και η εγκατάστασή της είναι πολύ απλή διαδικασία. Οι βάσεις δεδομένων που τρέχουν σε MySQL είναι πολύ σταθερές και δοκιμασμένες όλα αυτά τα χρόνια χωρίς να παρουσιάζουν προβλήματα. Αποτελούν την πιο «ελαφριά» μορφή βάσεων δεδομένων και οι απαιτήσεις που έχει η MySQL σε υπολογιστική ισχύ είναι ελάχιστες. Εκτός από τα παραπάνω η MySQL έχει εδραιωθεί κυρίως για το λόγο του ότι υποστηρίζει μηδενικούς χρόνους απόκρισης σε σύγκριση με άλλα πακέτα βάσεων δεδομένων.



Benchmarks

MySQL 5.7 Sysbench Benchmark: SQL Point Selects



MySQL Performance: Yes, we can do more than 1.6M QPS (SQL) on MySQL 5.7 GA »

Εικόνα 39 - Διάγραμμα απόκρισης μεταξύ των εκδόσεων της MySQL

Εκτός αυτού η MySQL συνεχώς βελτιώνεται και συνεχώς βελτιώνει τη ταχύτητά απόκρισης στα queries που δέχεται ανά δευτερόλεπτο με την έκδοση 5.7 να έχει απίστευτα υψηλά νούμερα απόκρισης σε μεγάλο αριθμό ταυτόχρονων συνδέσεων.

5.1.8 Σχετικά με τον Atom

Ο Atom, είναι ένας text και source editor, ανοικτού κώδικα, ο οποίος διανέμεται δωρεάν. Είναι κατάλληλος και για Windows λειτουργικά συστήματα αλλά και για Linux και OS X. Υποστηρίζει τις περισσότερες τεχνολογίες και έχει πλήρη υποστήριξη καθώς τα πρόσθετά του είναι γραμμένα σε Node JS. Επίσης έχει ενσωματωμένο Git Control και έχει υλοποιηθεί από το GitHub. Σαν editor αποτελεί έναν από τους ταχύτερα αναπτυσσόμενους και προσφέρει desktop εφαρμογή με υποστήριξη τεχνικών με τις οποίες γράφονται οι web editors. Είναι πολύ γρήγορος και έχει μεγάλη υποστήριξη από τη κοινότητα προγραμματιστών. Ο Atom βασίζεται στο Electron ή αλλιώς στο Atom Shell. Αυτό είναι ένα framework που επιτρέπει τα cross-platform προγράμματα που χρησιμοποιούν Chromium και IO.js που επίσης έχουν αναπτυχθεί από το GitHub. Κυκλοφόρησε σε beta έκδοση τον Ιούνιο του 2015 και οι προγραμματιστές πιστεύουν ότι είναι ο editor του 21^{ου} αιώνα.



```
atom.coffee
18
19 # Essential: Atom global for dealing with packages, themes, menus, and the win
20 #
21 # An instance of this class is always available as the `atom` global.
22 module.exports =
23 class Atom extends Model
24   @version: 1 # Increment this when the serialization format changes
25
26   # Load or create the Atom environment in the given mode.
27   #
28   # Returns an Atom instance, fully initialized.
29   @loadOrCreate: (mode) ->
30     startTime = Date.now()
31     atom = @deserialize(@loadState(mode)) ? new this({mode, @version})
32     atom.deserializeTimings.atom = Date.now() - startTime
33
```

Εικόνα 40 – Το user interface του ATOM

5.1.9 Σχετικά με το Photoshop

Το Adobe Photoshop είναι το κορυφαίο πρόγραμμα ψηφιακής επεξεργασίας γραφικών που αναπτύχθηκε και κυκλοφόρησε από την Adobe Systems. Χρησιμοποιείται από εκατομμύρια επαγγελματίες γραφίστες, φωτογράφους, σχεδιαστές όσο και από αρχάριους χρήστες. Η κοινότητα του Photoshop απαρτίζεται από φανατικούς της επεξεργασίας εικόνας, της γραφιστικής και υπάρχει πολύς όγκος από βιβλιογραφία σχετικά με τη χρήση του Adobe Photoshop, τόσο από αρχάριους χρήστες όσο και από πιο γνώστες. Η ιστορία του Photoshop ξεκινάει το 1987 ο Τόμας Κνολ, ανέπτυξε ένα πρόγραμμα που εμφάνιζε εικόνες σε αποχρώσεις του γκριζου (grayscale) σε μονοχρωματικό περιβάλλον. Αυτό το πρόγραμμα πήρε στη πρώτη του έκδοση την ονομασία «Display». Τότε ο αδερφός του Τόμας Κνολ, ο Τζον Κνολ, πρότεινε να αναπτύξουν το πρότυπο και να κυκλοφορήσουν ένα πρόγραμμα για επεξεργασία εικόνας. Τότε ο Τόμας Κνολ αποφασισμένος να το επιτύχει κάνει ένα διάλειμμα από τις σπουδές του και εργάστηκε εντατικά επάνω στο πρότζεκτ ImagePro που ήταν η νέα ονομασία του Display. Στη συνέχεια πίστεψε ότι το πρόγραμμα χρειάζεται ένα άλλο όνομα και μέσα στο ίδιο έτος μετονόμασε το πρόγραμμα σε Photoshop. Προωθώντας το πρόγραμμα, συνεργάστηκε για πρώτη φορά με την εταιρεία σαρωτών «BarneyScan» η οποία συμπεριέλαβε το πρόγραμμα ως το λογισμικό που θα συνόδευε τους σαρωτές της. Έχοντας μεγάλη πεποίθηση για το προϊόν τους ο Τζον ταξίδεψε στη Silicon Valley όπου παρουσίασε το πρόγραμμα στις εταιρείες Apple και Adobe. Εν τέλει η Adobe αποφάσισε να προβεί σε αγορά των δικαιωμάτων του Photoshop το Σεπτέμβριο του 1988. Σήμερα αποτελεί μέρος της Adobe Creative Suite και του Creative Cloud και γνωρίζει μεγάλη αναγνώριση από όσους ασχολούνται με την ψηφιακή επεξεργασία εικόνας.



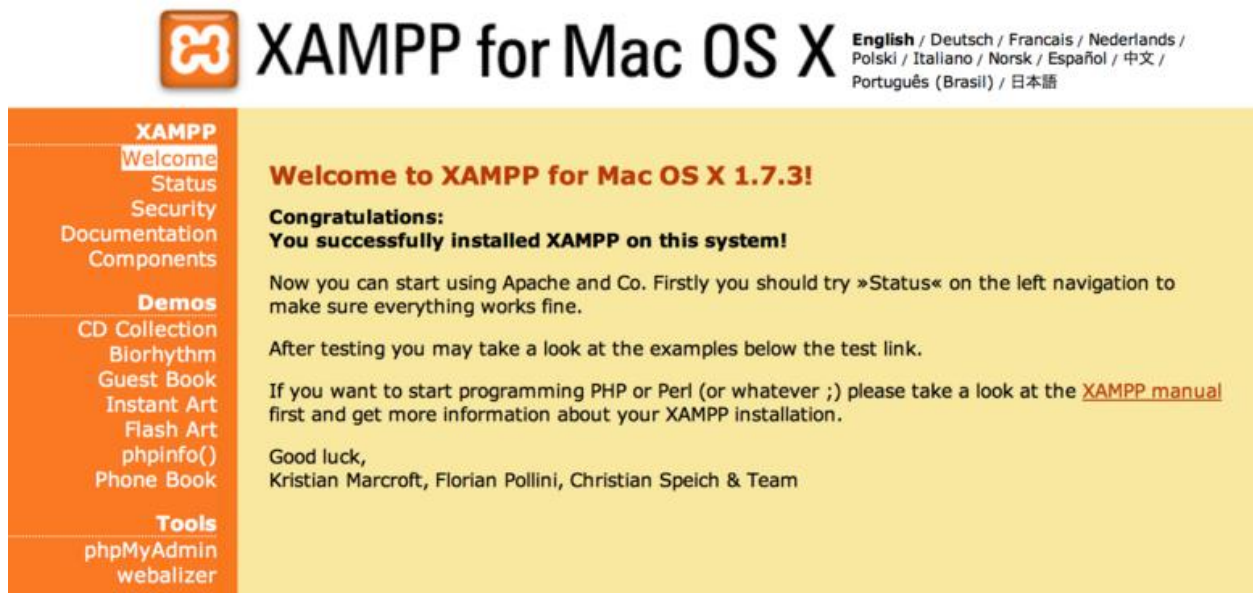
Εικόνα 41 - Η οθόνη έναρξης της τελευταίας έκδοσης του Photoshop με αναφορά στα 25 χρόνια από τη δημιουργία του δημοφιλιούς λογισμικού

5.1.10 Σχετικά με το XAMPP

Η ονομασία XAMPP αποτελεί το ακρωνύμιο από τα αρχικά των X Platforms (cross-platform software, δηλαδή του λογισμικού ανεξάρτητης πλατφόρμας), του Apache HTTP εξυπηρετητή, της MySQL, της PHP και της Perl. Είναι ένα ελεύθερο λογισμικό το οποίο είναι πολύ χρήσιμο σε προγραμματιστές διαδικτυακών εφαρμογών. Το XAMPP περιέχει ένα εξυπηρετητή ιστοσελίδων μέσω του οποίου μπορεί να εξυπηρετεί δυναμικές και σύγχρονες ιστοσελίδες βασισμένες στη τεχνολογία της PHP και της MySQL. Είναι συμβατό με όλα τα λειτουργικά συστήματα (Windows, MacOS X, Linux, Solaris, κλπ) και αποτελεί την πρώτη επιλογή όσων σχεδιάζουν και αναπτύσσουν ιστοσελίδες βασισμένες στις τεχνολογίες PHP, JSP, Servlets.

Η κοινότητα υποστήριξης του XAMPP είναι πολύ μεγάλη και συνεχώς αναπτύσσει το XAMPP ώστε να είναι συμβατό με νέες εκδόσεις του εξυπηρετητή ιστοσελίδων (Apache) αλλά και με νέες εκδόσεις των MySQL, PHP και Perl. Δύο σημαντικές προσθήκες του XAMPP είναι τα πακέτα OpenSSL αλλά και το PhpMyAdmin.

Για τη σωστή λειτουργία του XAMPP και για να είναι γρήγορο και αποδοτικό, έχουν απενεργοποιηθεί πολλές σημαντικές ρυθμίσεις ασφαλείας που χρησιμοποιούνται στο διαδίκτυο. Αυτό συμβαίνει γιατί το XAMPP τρέχει τα αρχεία τοπικά χωρίς να απαιτεί σύνδεση στο διαδίκτυο. Όταν εγκατασταθεί το XAMPP σε έναν ηλεκτρονικό υπολογιστή, τότε διαχειρίζεται την localhost ως έναν απομακρυσμένο κόμβο. Αυτός ο κόμβος συνδέεται με το πρωτόκολλο μεταφοράς αρχείων, δηλαδή το FTP. Τα στοιχεία σύνδεσης στον FTP είναι προκαθορισμένα και είναι για το όνομα χρήστη το «newuser» και για τον κωδικό το «wamprr», ενώ για τη βάση δεδομένων υπάρχει προκαθορισμένα ο χρήστης «root» χωρίς να έχει κάποιο κωδικό πρόσβασης.



Εικόνα 42 – Η αρχική οθόνη του XAMPP

5.2 Λειτουργίες της εφαρμογής

Οι λειτουργίες της εφαρμογής είναι:

1. Εγγραφή χρήστη
2. Σύνδεση χρήστη
3. Καταχώρηση μουσικής εγγραφής
4. Τροποποίηση μουσικής εγγραφής
5. Διαγραφή μουσικής εγγραφής
6. Αποστολή μουσικής εγγραφής
7. Τροποποίηση προφίλ χρήστη
8. Προβολή λίστας μουσικής
9. Προβολή εγγραφής μουσικής
10. Καταχώρηση οπτικοακουστικής εγγραφής
11. Τροποποίηση οπτικοακουστικής εγγραφής
12. Διαγραφή οπτικοακουστικής εγγραφής
13. Αποστολή οπτικοακουστικής εγγραφής
14. Αποσύνδεση χρήστη



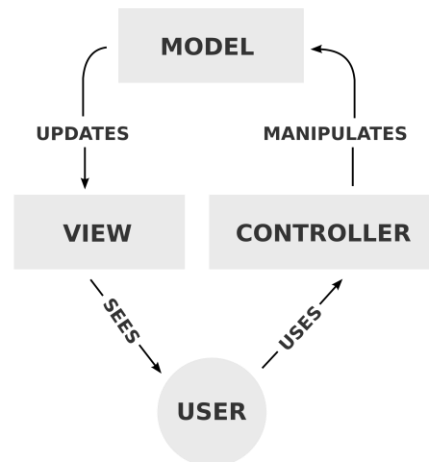
5.3 Αρχιτεκτονική Συστήματος

Η εφαρμογή που αναπτύχθηκε αποτελείται από τρία μέρη.

Το client μέρος που αποτελεί το front – end τμήμα και έχει υλοποιηθεί με χρήση της HTML & Angular JS και βασίζεται στην MVC Framework αρχιτεκτονική

Το κομμάτι κώδικα που τρέχει στον διακομιστή (server-side) και βασίζεται στο Slim micro-framework και την PHP. Οι καταχωρήσεις στη βάση έχουν γίνει με PDOstatements με σκοπό την καλύτερη οργάνωση και τη βελτιστοποίηση ταχύτητας του συστήματος. Ο server επιστρέφει σε κάθε κλήση του συστήματος ένα JSON με την πληροφορία που ζητείται. (REST API web service)

Και τέλος το κομμάτι της Βάσης Δεδομένων και του DB Server που έχει υλοποιηθεί σε MySQL λόγω της υψηλής ταχύτητας που παρέχει και σε αυτή καταχωρείται όλη η πληροφορία της εφαρμογής.

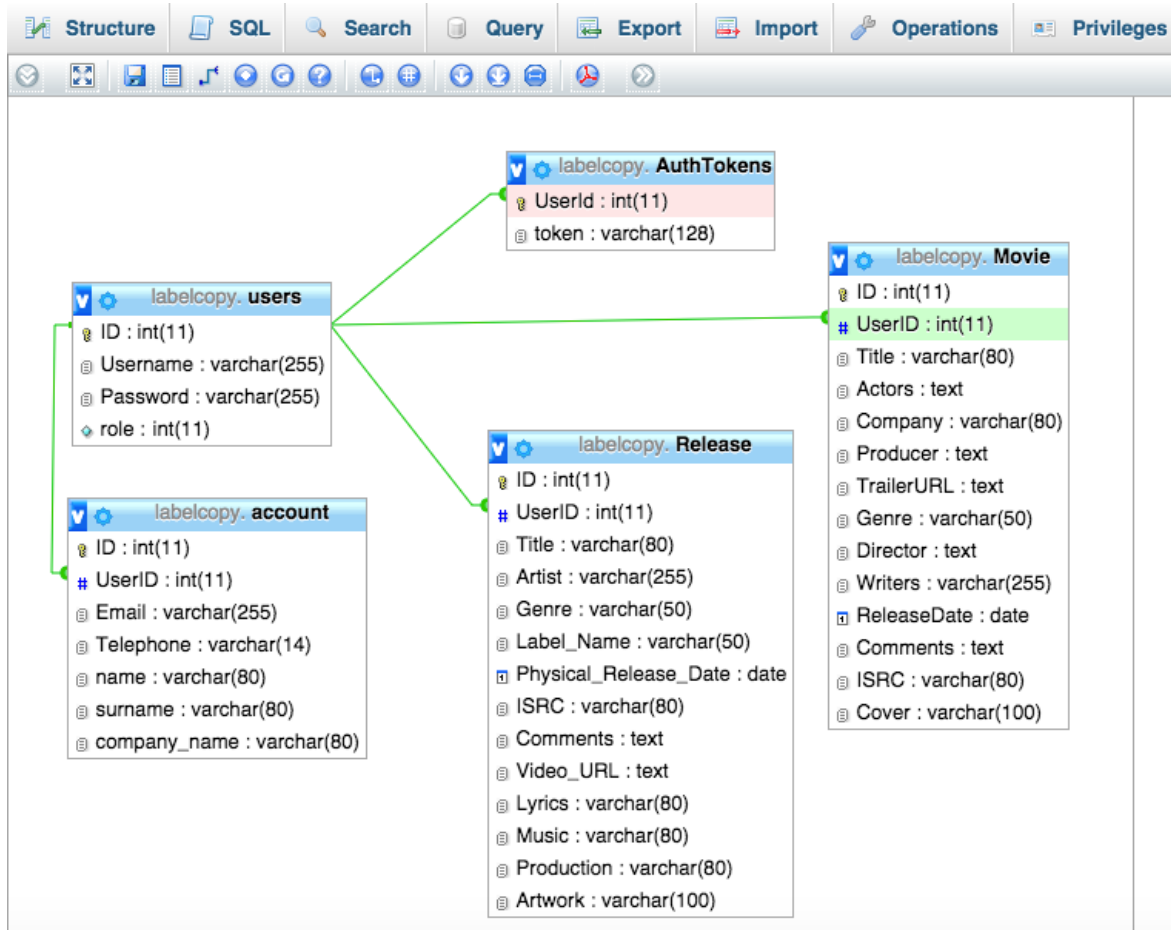


Εικόνα 43 – Δομή αρχιτεκτονικής MVC



5.4 Σχεδιάγραμμα βάσης δεδομένων

Η εφαρμογή συνδέεται με την Βάση Δεδομένων με όνομα labelcopy. Η βάση αποτελείται από πέντε πίνακες των οποίων η δομή και η διασύνδεση εμφανίζεται στη παρακάτω εικόνα. Κάθε πίνακας έχει διαφορετικά πεδία που επικοινωνούν μεταξύ τους. Τα περισσότερα πεδία είναι υποχρεωτικού χαρακτήρα.



Εικόνα 44 – Το σχεδιάγραμμα της βάσης δεδομένων της εφαρμογής – Απεικόνιση πινάκων



5.5 Πίνακες βάσης δεδομένων

Η βάση της εφαρμογής ονομάζεται labelcopy και αποτελείται από τους πίνακες: Users, Account, AuthTokens, Release, Movie.

Table	Action	Rows	Type	Collation	Size	Overhead
account	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8_bin	32 Kib	-
authtokens	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_bin	32 Kib	-
movie	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8_bin	32 Kib	-
release	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8_bin	32 Kib	-
users	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8_bin	16 Kib	-
5 tables	Sum	15	InnoDB	utf8_bin	144 Kib	0 B

Εικόνα 45 – Πίνακες βάσης δεδομένων

5.5.1 Πίνακας Users

Ο πίνακας Users αποτελείται από τέσσερα πεδία. Το ID, το username, το password και το role. Το πεδίο ID παίρνει το μοναδικό ID του κάθε εγγεγραμμένου χρήστη. Οι τιμές που δέχεται είναι τύπου integer (int). Στο πεδίο username καταχωρείται το όνομα χρήστη και οι εγγραφές που δέχεται είναι τύπου varchar. Το ίδιο ισχύει και στο πεδίο του password όπου καταχωρείται ο κωδικός χρήστη. Το τέταρτο πεδίο role έχει οριστεί για μελλοντική χρήση καθώς διαχωρίζει τους χρήστες από τους διαχειριστές του συστήματος. Σε μελλοντική αναβάθμιση της εφαρμογής θα μπορούσαν οι χρήστες να έχουν bit 0 και οι διαχειριστές bit 1.



#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	ID	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index Spatial Fulltext Distinct values
2	Username	varchar(255)	utf8_bin		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
3	Password	varchar(255)	utf8_bin		No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values
4	role	bit(1)			No	None		Change Drop Primary Unique Index Spatial Fulltext Distinct values

Space usage		Row statistics	
Data	16 KiB	Format	compact
Index	0 B	Collation	utf8_bin
Total	16 KiB	Next autoindex	15
		Creation	Oct 18, 2015 at 07:48 PM

Εικόνα 46 – Δομή του πίνακα users

5.5.2 Πίνακας Account

Ο πίνακας account, περιέχει τους λογαριασμούς των εγγεγραμμένων χρηστών στο σύστημα. Εμφανίζει το ID, το email εγγραφής, το τηλέφωνο, το όνομα, το επώνυμο και το όνομα της εταιρείας για την οποία εργάζεται.

ID	UserID	Email	Telephone	name	surname	company_name
7	9	ndafnomilis@gmail.com	697XXXXXXX1	Nicolas-Christos	Dafnomilis	Music Company
10	12	demo@demo.fr	123123123	Demo	User	Company
11	13	thomas@thomas.gr	123456789	Thomas	Thomas	Thom Records
12	14	thomas@thomas.gr	123456789	thomas	thomas	Thom Records

Εικόνα 47 – Δομή του πίνακα account



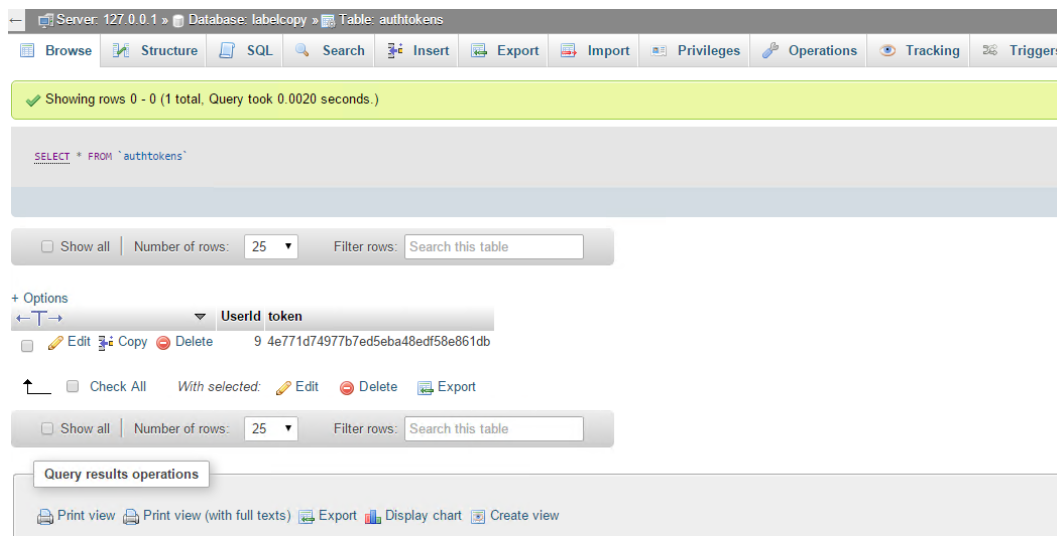
Η δομή του πίνακα account περιλαμβάνει όλα τα πεδία που περιέχουν τα προσωπικά στοιχεία του κάθε χρήστη και είναι άμεσα συνδεδεμένος ο πίνακας account με τον πίνακα users. Τα πεδία που περιέχει ο πίνακας account είναι τα:

- 1) ID
- 2) UserID
- 3) Email
- 4) Telephone
- 5) Name
- 6) Surname
- 7) Company_Name

Η διασύνδεση του πίνακα users με τον πίνακα account γίνεται μέσω του UserID.

5.5.3 Πίνακας AuthTokens

Ο πίνακας AuthTokens, περιέχει τα generated tokens για κάθε χρήστη που έχει κάνει login στο σύστημα. Κατά τη σύνδεση στο σύστημα ο κάθε χρήστης λαμβάνει ένα μοναδικό token το οποίο αποτρέπει άλλους χρήστες να συνδεθούν με τα ίδια στοιχεία σύνδεσης όσο αυτός είναι συνδεδεμένος στο σύστημα. Κατά την αποσύνδεση το token διαγράφεται από τη βάση.



Εικόνα 48 – Δομή του πίνακα AuthTokens

5.5.4 Πίνακας Release

Ο πίνακας release περιέχει τα πεδία με τη πληροφορία για κάθε μουσικό έργο που καταχωρείται στη βάση. Αυτά τα πεδία είναι το ID κάθε έργου, το UserID για να ξέρουμε ποιος χρήστης το έχει καταχωρήσει, ο τίτλος του τραγουδιού, το όνομα του ερμηνευτή, το μουσικό είδος, το όνομα της εταιρείας που έκανε τη παραγωγή, την ημερομηνία κυκλοφορίας, τον μοναδικό κωδικό αριθμό isrc, τα σχόλια, τον σύνδεσμο για το μουσικό βίντεο, τους στιχουργούς, τους συνθέτες, τους



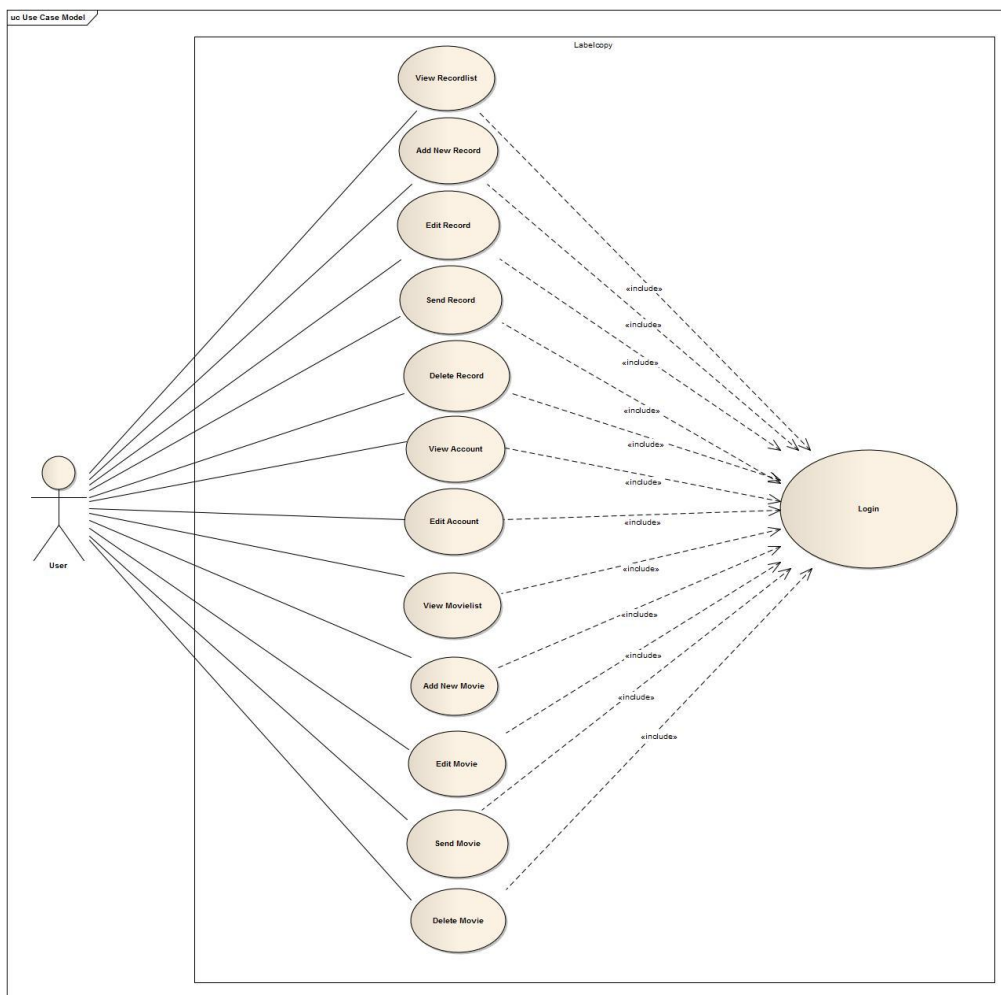
5.6 Διάγραμμα περιπτώσεων χρήσης

Παρακάτω παρουσιάζεται το διάγραμμα περιπτώσεων χρήσης της εφαρμογής. Για να πραγματοποιηθούν όλα τα actions στην εφαρμογή, απαιτείται η εγγραφή χρήστη και η σύνδεσή του στο σύστημα.

Οι περιπτώσεις χρήσης που εμφανίζονται στο παρακάτω διάγραμμα είναι:

- η «view recordlist»
- η «add new record»
- η «edit record»
- η «send record»
- η «delete record»
- η «view account»
- η «edit account»
- η «view movielist»
- η «add new movie»
- η «edit movie»
- και η «send movie»

Όλες οι περιπτώσεις αφορούν εγγεγραμμένους χρήστες και για αυτό απαιτείται ο χρήστης να συνδεθεί στο σύστημα. Δεν υπάρχουν περιπτώσεις χρήσης για μη εγγεγραμμένους χρήστες.



Εικόνα 51 – Διάγραμμα περιπτώσεων χρήσης της εφαρμογής



5.7 Επεξήγηση τμημάτων κώδικα

Παρουσίαση τμήματος κώδικα από τη πλευρά του εξυπηρετητή (server). Στο παρακάτω τμήμα κώδικα που βρίσκεται στο αρχείο index.php, αρχικά δηλώνουμε ότι θα χρησιμοποιήσουμε το slim framework. Έπειτα ορίζουμε κάποιες σταθερές τις οποίες θα χρησιμοποιήσει το σύστημα για την επικοινωνία με τη βάση και την αποστολή emails. Αρχικοποιούμε ένα instance του Slim και ορίζουμε τους http headers που θα δεχόμαστε.

```
/* Step 1: Require the Slim Framework */  
  
Require 'Slim/Slim.php';  
  
define("DB_USER", "labelcopyUser"); /*enter DB username*/  
define("DB_PASS", "123456"); /*enter DB password*/  
define("HOST", "127.0.0.1"); /*enter host address*/  
define("DATABASE", "labelcopy"); /*enter DB name*/  
define("ADMINEMAIL","email@email.com"); /*user email for sending email*/  
define("ADMINEMAILPASS","enterpasswordhere"); /*password for email above*/  
  
/* initialize slim */  
  
\Slim\Slim::registerAutoloader();  
  
$app = new \Slim\Slim();  
  
/* set response headers */  
  
$app->response->headers->set('Access-Control-Allow-Origin', '*');  
  
$app->response->headers->set('Access-Control-Allow-Methods', 'GET, POST, PUT, OPTIONS');  
  
$app->response->headers->set('Access-Control-Allow-Headers', 'Origin, X-Requested-With, Content-Type, Accept');
```

Στη συνέχεια θα αναφερθούμε στη συνάρτηση login με σκοπό να εξηγήσουμε τον τρόπο με τον οποίο γίνεται η είσοδος του χρήστη στο σύστημα. Ο χρήστης καταχωρώντας το username και το password του, κάνει ένα http post request στον διακομιστή (server) αποστέλλοντας τα στοιχεία του. Η login συνάρτηση παραλαμβάνει τα στοιχεία του χρήστη και ελέγχει εάν τα στοιχεία αυτά αντιπροσωπεύουν κάποιον λογαριασμό στο σύστημα. Εάν γίνει η αυθεντικοποίηση του χρήστη, παράγεται ένα μοναδικό token το οποίο επιστρέφεται στη client πλευρά του συστήματος και η client πλευρά πρέπει να το συμπεριλαμβάνει στους headers της σε κάθε κλήση προς το σύστημα. Το τμήμα κώδικα που παρουσιάζεται αναφέρεται στη λειτουργία που μόλις εξηγήσαμε.



```

$app->post('/login', function() {
    $request = \Slim\Slim::getInstance() -> request();
    $username = $request->post('username'); // get username from request
    $pass = $request->post('password'); // get password from request
    try{
        $db = getConnection(); // connect to database

        $sql = "select u.id from users as u inner join account as a on a.userid = u.id where
u.username=:username and u.password=:password limit 1";

        $stmt = $db->prepare($sql);
        $stmt->bindParam("username", $username);

// generate sha512 hash from given password
        $stmt->bindParam("password", hash('sha512', $username . $pass . $username));
        $stmt->execute();
        $userId = $stmt->fetchObject();
        $retVal = new stdClass();
        $retVal->isAuth = ($userId != null);
        if ($userId) { // user is authenticated
            $retVal->expires_in = 3600; // 60 min in sec – token expiration
            $retVal->token = bin2hex(openssl_random_pseudo_bytes(16)); // generate token

// save token to database
            $sql = "insert into AuthTokens (userid, token) values (:userId, :token)";
            $stmt = $db->prepare($sql);
            $stmt->bindParam("userId", $userId->id);
            $stmt->bindParam("token", $retVal->token);
            $stmt->execute();
        }
        echo json_encode($retVal); // return token to client side
    } catch (PDOException $e) { return ; } });

```



Αφού έχει πραγματοποιηθεί η σύνδεση του χρήστη στο σύστημα και ο χρήστης έχει παραλάβει το μοναδικό token του, όπως αναφέραμε παραπάνω, σε κάθε νέα κλήση που γίνεται στον server θα πρέπει να αποστέλλεται μαζί και το token μέσα στους headers. Ο server προτού εκτελέσει κάποια κλήση που καλεί ο χρήστης, ελέγχει αν το token είναι ακόμη ενεργό ή αν έχει κάνει expire. Τη διαδικασία αυτή, την επιτυγχάνουμε μέσω του Middleware που μας παρέχει το Slim Framework. Η ιδιότητα του Middleware είναι το γεγονός ότι εκτελεί πρώτα μία συγκεκριμένη διαδικασία και μετά καλεί την κύρια κλήση. Στη δική μας περίπτωση πρώτα θα εκτελεστεί η διαδικασία ελέγχου του Token και αυθεντικοποίησης του χρήστη και μετά θα ακολουθήσει η κλήση την οποία έχει ζητήσει ο χρήστης.

Η κλάση OAuth2Auth είναι η κλάση του authentication και κληρονομεί, όπως προαναφέραμε (extends) από το Middleware του Slim. Αποτελείται ουσιαστικά από 2 συναρτήσεις την FindUser και την Call.

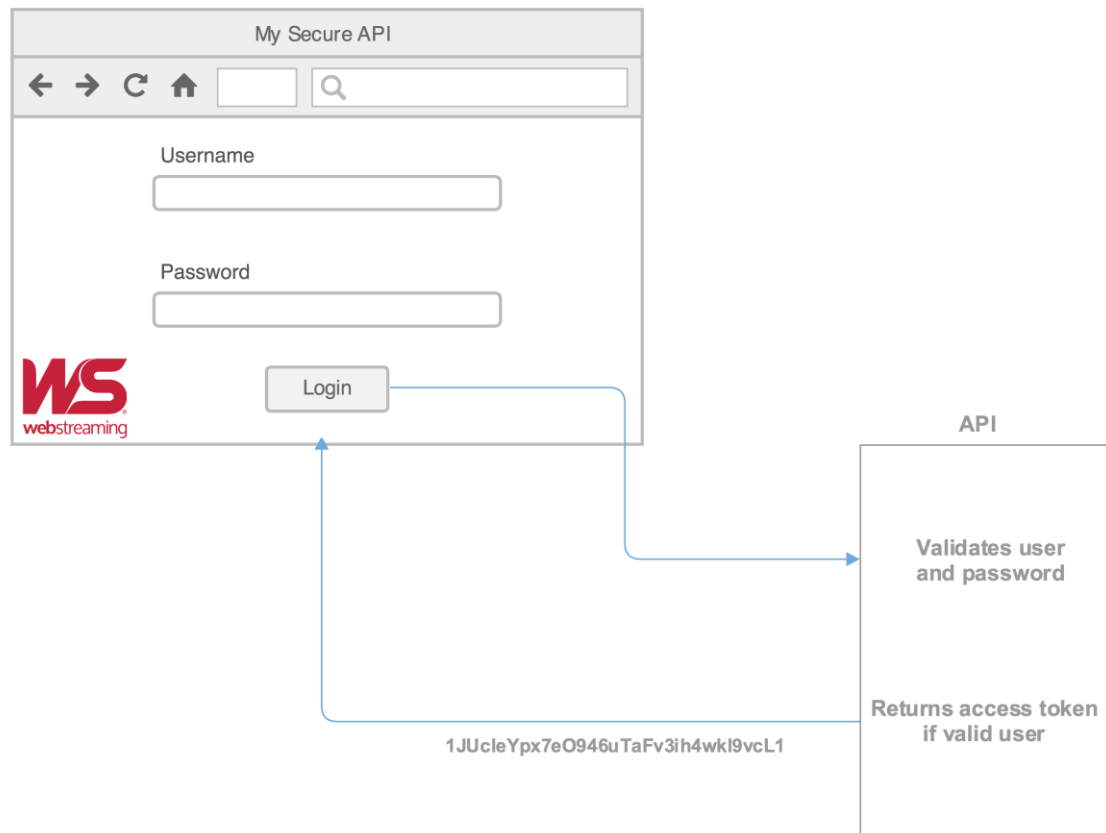
Όταν ο server δέχεται μια κλήση που απαιτεί authentication, καλεί αυτόματα την συνάρτηση Call του OAuth2Auth. Αυτή, διαβάζει από τους headers το αίτημα του χρήστη και με την σειρά της καλεί την FindUser για να ελέγξει αν ο χρήστης έχει ενεργό token. Η FindUser ψάχνει στον πίνακα με τα tokens, το token του χρήστη, και επιστρέφει το id του χρήστη αν το βρει.

```
// Auth via Slim Middleware
```

```
class OAuth2Auth extends \Slim\Middleware {  
    protected function findUser ($db, $token) {  
        $sql = "select u.id  
            from Users as u  
            inner join AuthTokens a on a.UserId = u.id  
            inner join account as ac on ac.userid = u.id  
            where a.token =:token;";  
        $stmt = $db->prepare($sql);  
        $stmt->bindParam("token", $token);  
        $stmt->execute();  
        $userId = $stmt->fetchObject();  
        return $userId->id;  
    }  
}
```



```
public function call() {
    // no auth header
    $app = $this->app;
    if (preg_match('|^' . $this->root . '.*|', $this->app->request->getResourceUri())) {
        if (!isset($app->request()->headers['Authorization'])) {
            die();
        } else {
            try {
                $authHeader = explode(" ", $app->request()->headers('Authorization'))[1];
                $validated_user_id = $this->findUser($this->mysql, $authHeader);
                $this->app->user_id = $validated_user_id;
            } catch (Exception $e) {
                $view = $this->app->view();
                $view->setData("data", array("message" => $e->getMessage()));
            }
        }
    }
    $this->next->call();
}
```



Εικόνα 52 – Διάγραμμα λειτουργίας του Middleware και της υποστήριξης των tokens

Για να καταφέρει το σύστημα να αναγνωρίσει το Middleware που δημιουργήσαμε θα πρέπει να του το δηλώσουμε :

```
$app->add(new OAuth2Auth(getConnection(), '/authorize'));
$app->group('/authorize', function() use ($app) {
    $app->get('/release', function() use ($app) { ...});
    .....
    .....
});
```

Με αυτό τον τρόπο, το Slim αναγνωρίζει πως οποιαδήποτε κλήση προς URL με την μορφή «/authorize/<operation>» θα πρέπει να φέρει το token στους headers της και να ελεγχθεί από το Middleware. Όπως φαίνεται και στο κομμάτι κώδικα παραπάνω, για να πάρει ο χρήστης μια λίστα από «release» θα πρέπει να κάνει μια HTTP GET κλήση προς τον διακομιστή με url : /authorize/release. Στο παραπάνω τμήμα κώδικα, φαίνεται άλλο ένα στοιχείο του Slim, το «group». Το group έχει την έννοια της ομαδοποίησης των routes βάση του prefix που έχουν. Πιο



συγκεκριμένα, εμείς θέλουμε να ομαδοποιήσουμε όλα τα routes τα οποία μας οδηγούν σε διαδικασίες οι οποίες απαιτούν αυθεντικοποίηση του χρήστη. Άρα, δημιουργούμε ένα group που αναγνωρίζει αρχικά το prefix «authorize», και σε αυτό το group ορίζουμε την κάθε συνάρτηση που αναγνωρίζει το υπόλοιπο μέρος του url πχ «/release».

6. Συμπεράσματα και μελλοντικές επεκτάσεις

6.1 Συμπεράσματα

Η εφαρμογή που υλοποιήθηκε, σχεδιάστηκε και δημιουργήθηκε ύστερα από έρευνα στο τρόπο λειτουργίας και αρχειοθέτησης περιεχομένου των εταιρειών που εμπορεύονται πνευματική ιδιοκτησία (δισκογραφικές εταιρείες κλπ). Σκοπός της εφαρμογής είναι η απλοποίηση του αρχείου και η επίσπευση των διαδικασιών διαμοιρασμού της πληροφορίας με σκοπό την καλύτερη αξιολόγηση των τίτλων που έχει στην ιδιοκτησία της η εταιρεία.

6.2 Μελλοντικές επεκτάσεις και βελτιώσεις

Όπως κάθε εφαρμογή που υλοποιείται, έτσι και αυτή δέχεται περαιτέρω βελτιώσεις και προσθήκες καθώς είναι πολλές οι δυνατότητες που μας προσφέρει η τεχνολογία των web services.

Αρχικά θα μπορούσαμε να πούμε ότι η εφαρμογή θα δεχόταν νέους τύπους αρχείων πνευματικής ιδιοκτησίας. Αυτά μπορεί να είναι:

- Δημοσιεύσεις
- Βιβλία και συγγράμματα
- Ποιήματα
- Παραμύθια
- Τηλεοπτικές σειρές
- Ντοκιμαντέρ
- Εκδόσεις λογισμικού

Κάθε τύπος από τους παραπάνω εξυπηρετεί και ένα διαφορετικό κομμάτι των πνευματικών δικαιωμάτων. Αυτό θα μπορούσε να οδηγήσει στο να εδραιωθεί η εφαρμογή που υλοποιήθηκε και να γίνει απαραίτητη στους επαγγελματίες του χώρου είτε μέσω του web interface είτε μέσω ενσωμάτωσης αυτής σε τρίτες εφαρμογές.

Κάθε κατηγορία πνευματικής ιδιοκτησίας, συνδέεται με μεγαλύτερες εταιρείες που διαχειρίζονται την πληροφορία αλλά και τα οικονομικά θέματα των δικαιωμάτων. Αυτές οι εταιρείες θα μπορούν να συνδεθούν στο σύστημά μας μέσω του api και να λαμβάνουν τα αρχεία απευθείας στο σύστημά τους.

Απαραίτητη προϋπόθεση να εισάγουν τα σωστά στοιχεία στα header types και να περνάνε μέσα από τη συνάρτηση auth όλες τις κλήσεις που κάνουν στο σύστημα. Όπως ήδη έχει αναφερθεί



παραπάνω η εφαρμογή καλύπτει κάποια πρωτόκολλα ασφαλείας και είναι έτοιμη ώστε να επεκταθεί και να συνδεθεί με μεγαλύτερα συστήματα.

Μία ακόμη μελλοντική προσθήκη θα μπορούσε να είναι η ύπαρξη ρόλου διαχειριστή που θα μπορεί να λαμβάνει πληροφορίες για όλες τις καταχωρήσεις του συστήματος, από όλους τους χρήστες και θα επιβλέπει τη λειτουργία του συστήματος. Επίσης θα μπορούσε να δημιουργηθεί σύνδεσμος διαμοιρασμού της πληροφορίας με μετρητή αντίστροφης μέτρησης ώστε να μπορεί κάποιος να δει την πληροφορία για λίγα λεπτά και μετά να «κλειδώσει» ξανά.

Μία ήδη σημαντική προσθήκη που στο μέλλον θα μπορούσε να αναπτυχθεί περισσότερο είναι η προσθήκη και τρίτης γλώσσας για το περιεχόμενο και η απόπειρα να γίνει η εφαρμογή localized για χώρες που δεν χρησιμοποιούν τα αγγλικά.

Αν και η εφαρμογή το landing page και το web interface είναι πλήρως συμβατά με φορητές συσκευές, θα μπορούσε στο μέλλον να δημιουργηθεί μία εφαρμογή για φορητές συσκευές που θα έκανε χρήση και των μετρητών των συσκευών και των data περιαγωγής για εξαγωγή χρήσιμων αποτελεσμάτων.



7. Πίνακας Εικόνων

Εικόνα 1 – Ο ιστότοπος της υπηρεσίας CRS (Copyright Registration Service)	12
Εικόνα 2 – Σελίδα καταχώρησης περιεχομένου της υπηρεσίας CRS	13
Εικόνα 3 – Ο ιστότοπος της υπηρεσίας ACS (Artists' Collecting Society)	14
Εικόνα 4 – Ο ιστότοπος του οργανισμού DACS (Design & Artists Copyright Society)	15
Εικόνα 5 – Ο ιστότοπος του οργανισμού MPLC (Motion Picture Licensing Corporation)	16
Εικόνα 6 – Ο ιστότοπος της Orchard	17
Εικόνα 7 – Εγγραφή χρήστη	18
Εικόνα 8 - Πριν τη συμπλήρωση των πεδίων	19
Εικόνα 9 – Μετά τη συμπλήρωση των πεδίων	19
Εικόνα 10 – Συμπληρωμένα πεδία κατά την εγγραφή χρήστη	19
Εικόνα 11 – Αγγλική εκδοχή της φόρμας εγγραφής	20
Εικόνα 12 – Η φόρμα σύνδεσης χρήστη στα Ελληνικά	20
Εικόνα 13 – Η φόρμα σύνδεσης χρήστη στα Αγγλικά	21
Εικόνα 14 – Πίνακας ελέγχου συνδεδεμένου χρήστη – Στατιστικά στοιχεία	21
Εικόνα 15 – Στατιστικά στοιχεία καταχωρήσεων	22
Εικόνα 16 – Στατιστικά στοιχεία καταχωρήσεων	22
Εικόνα 17 – Στατιστικά στοιχεία καταχωρήσεων	22
Εικόνα 18 – Προβολή και τροποποίηση προφίλ χρήστη	23
Εικόνα 19 – Προβολή της οθόνης καταχώρησης μουσικού περιεχομένου	24
Εικόνα 20 – Προβολή της λίστας τραγουδιών	25
Εικόνα 21 – Εμφάνιση της λειτουργίας αναζήτησης	25
Εικόνα 22 – Εμφάνιση της επιλογής τροποποίησης της εγγραφής και της σήμανσης tooltip	26
Εικόνα 23 – Εμφάνιση της φόρμας τροποποίησης της εγγραφής	27
Εικόνα 24 – Εμφάνιση παραθύρου με τη πληροφορία του τραγουδιού που θα αποσταλεί	28
Εικόνα 25 – Διαγραφή καταχώρησης	28
Εικόνα 26 – Εμφάνιση της οθόνης καταχώρησης οπτικοακουστικού περιεχομένου	29
Εικόνα 27 – Προβολή της λίστας ταινιών	30
Εικόνα 28 – Εμφάνιση της επιλογής τροποποίησης της εγγραφής και της σήμανσης tooltip	30
Εικόνα 29 – Οθόνη τροποποίησης της εγγραφής ταινίας	31
Εικόνα 30 – Οθόνη αποστολής πληροφοριών ταινίας	32
Εικόνα 31 – Οθόνη αποστολής πληροφοριών ταινίας	32
Εικόνα 32 - Οθόνες της βασικής ιστοσελίδας της εφαρμογής	33
Εικόνα 33 – Οθόνες εμφάνισης στατιστικών στοιχείων για τις εγγραφές της εφαρμογής	34
Εικόνα 34 – Διάγραμμα παρουσίασης της δομής των web services	38
Εικόνα 35 – Παρουσίαση χρήσης της Php	39
Εικόνα 36 – Παρουσίαση της δομής του Slim Framework	41
Εικόνα 37 – Διάφοροι τύποι οθόνης που υποστηρίζονται από το Bootstrap	42
Εικόνα 38 – Παρουσίαση responsive στοιχείων του Bootstrap	43
Εικόνα 39 - Διάγραμμα απόκρισης μεταξύ των εκδόσεων της MySQL	47
Εικόνα 40 – Το user interface του ATOM	48
Εικόνα 41 - Η οθόνη έναρξης της τελευταίας έκδοσης του Photoshop με αναφορά στα 25 χρόνια από τη δημιουργία του δημοφιλούς λογισμικού	49
Εικόνα 42 – Η αρχική οθόνη του XAMPP	50
Εικόνα 43 – Δομή αρχιτεκτονικής MVC	51
Εικόνα 44 – Το σχεδιάγραμμα της βάσης δεδομένων της εφαρμογής – Απεικόνιση πινάκων	52
Εικόνα 45 – Πίνακες βάσης δεδομένων	53
Εικόνα 46 – Δομή του πίνακα users	54
Εικόνα 47 – Δομή του πίνακα account	54
Εικόνα 48 – Δομή του πίνακα AuthTokens	55
Εικόνα 49 – Δομή του πίνακα release	56



Εικόνα 50 – Δομή του πίνακα movie	56
Εικόνα 51 – Διάγραμμα περιπτώσεων χρήσης της εφαρμογής	58
Εικόνα 52 – Διάγραμμα λειτουργίας του Middleware και της υποστήριξης των tokens	63



8. Βιβλιογραφικές Πηγές

Introduction to Slim, “a micro framework for PHP”

<http://slimframework.com>

Getting Started with Slim

<http://help.slimframework.com>

Hypertext Transfer Protocol

https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol

Hypertext Transfer Protocol - HTTP/1.1

<http://www.w3.org/Protocols/rfc2616/rfc2616.html>

HTTP: The protocol every web developer must know

<http://net.tutsplus.com/tutorials/tools-and-tips/http-the-protocol-every-web-developer-must-know-part-1/>

David Gourley , HTTP: The Definitive Guide (Definitive Guides) 1st Edition

<http://www.amazon.com/HTTP-The-Definitive-Guide-Guides/dp/1565925092>

IBM (2007), “New to SOA and Web services”

<http://www.128.ibm.com/developerworks/webservices/newto/websvc.html>

MSDN (2001), “XML Web Services Basics”

<http://msdn2.microsoft.com/en-us/library/ms996507.aspx>

Roseindia.net (2007), “Web Services” <http://www.roseindia.net/webservices/webservices.shtml>

Developer.com (2003), “Web Services Tutorial: Understanding XML and XML Schema” □

“Part 1” <http://www.developer.com/services/print.php/2195981>

“Part 2” http://www.developer.com/services/print.php/10928_2195981_2

Developer.com (2002), “Introduction to Web Services”

“Part 1” <http://www.developer.com/services/article.php/1485821>

“Part 2: Architecture” <http://www.developer.com/services/article.php/1495091>

“Part 3: Understanding XML” <http://www.developer.com/services/article.php/1557871>

W3C (2001), “Web Service Definition Language (WSDL)” <http://www.w3.org/TR/wsdl>

Michele E. Davis, Jon A. Phillips (2007). *Learning PHP & MySQL*. O'Reilly

«Interview with Kai Seidler from the XAMPP project». MySQL AB. <http://dev.mysql.com/tech-resources/interviews/kai-seidler-xampp.html>

«apache friends - xampp for windows». Apachefriends.org. 2008-12-19.

<http://www.apachefriends.org/en/xampp-windows.html>

ONJava.com (2001), “Java and Web Services Primer”

<http://www.onjava.com/lpt/a/1025>

W3schools.com (2007), “XML Tutorial” <http://www.w3schools.com/xml/default.asp>



About PHP <http://php.net/manual/en/faq.general.php>

What every web developer must know about url encoding
<http://blog.lunatech.com/2009/02/03/what-every-web-developer-must-know-about-url-encoding>

List of copyright collection societies (global)
https://en.m.wikipedia.org/wiki/List_of_copyright_collection_societies#Global

Trinity tutorials – Build your first web service with php
<https://trinitytuts.com/build-first-web-service-php/>

Angular JS - Getting started
<https://material.angularjs.org/0.11.2/#/getting-started>

Quick introduction to Angular JS
<https://egghead.io/articles/new-to-angularjs-start-learning-here>

Wikipedia, About Angular JS
<https://en.wikipedia.org/wiki/AngularJS>

Wikipedia, About XAMPP
<https://el.wikipedia.org/wiki/XAMPP>

WIPOCOS - Software for Collective Management of Copyright and Related Rights
<http://www.wipo.int/copyright/en/initiatives/wipocos.html>

Digital rights management (DRM)
https://en.wikipedia.org/wiki/Digital_rights_management

The Digital Learning Challenge: Obstacles to Educational Uses of Copyrighted Material in the Digital Age
<http://cyber.law.harvard.edu/media/files/copyrightandeducation.html>

Top online ID3Tag Editor
<http://www.wondershare.com/id3/online-id3-tag-editors.html>

Wikipedia, About Photoshop
<https://el.wikipedia.org/wiki/Photoshop>