



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

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Κατασκευή Component Στίχων για την πλατφόρμα Joomla. Lyrics Component Development for Joomla Platform.
Όνοματεπώνυμο Φοιτητή	Τρισετάκης Δημήτριος
Πατρώνυμο	Εμμανουήλ
Αριθμός Μητρώου	ΜΠΣΠ/ 12069
Επιβλέπων	Μαρία Βίββου, Καθηγήτρια

Ημερομηνία Παράδοσης **Οκτώβριος 2016**





Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

(υπογραφή)

Γεώργιος Τσιχριντζής
Καθηγητής

(υπογραφή)

Ευθύμιος Αλέπης
Επίκουρος Καθηγητής



Περίληψη

Στις μέρες μας όλο και περισσότεροι προγραμματιστές στρέφονται στην κατασκευή ιστοσελίδων μέσω κάποιας πλατφόρμας CMS. Τώρα πια θεωρείται απαραίτητο ένας προγραμματιστής που ασχολείται με την κατασκευή ιστοσελίδων να έχει γνώση της λειτουργίας τους, καθώς και τη δυνατότητα να μπορεί να το παραμετροποιήσει και να το επεκτείνει στις ανάγκες της κάθε περίπτωσης.

Το κύριο πλεονέκτημα μιας πλατφόρμας CMS σε σχέση με μια σελίδα εξ ολοκλήρου από το μηδέν είναι η αποδοτικότητα και η ανταγωνιστικότητα που μπορεί κάποιος να έχει αν το έφτιαχνε από την αρχή. Παρατηρούμε ότι ο χρόνος κατασκευής μιας σελίδας μπορεί να μειωθεί πάνω από το 70%.

Τα δεύτερο μεγάλο πλεονέκτημα είναι η επεκτασιμότητα. Οι πελάτες μιας εταιρείας έχουν αυξανόμενες ανάγκες και δεν είναι δυνατόν να καλυφθούν άμεσα. Με τις επεκτάσεις των CMS μπορούν να καλυφθούν όλο και περισσότερες ανάγκες οι οποίες μπορούν να καλύψουν και τον πιο απαιτητικό χρήστη.

Άλλο ένα πολύ σημαντικό πλεονέκτημα των CMS είναι η ασφάλεια τους απέναντι σε πιθανές κακόβουλες ενέργειες απέναντι τους. Μπορεί από την μία τα περισσότερα CMS να αποτελούν λογισμικά ανοιχτού κώδικα όπου δίνουν πατήματα στις αδυναμίες τους, ωστόσο από την άλλη κάθε χρήστης είναι ένας δυναμικός συνεργάτης που θα βοηθήσει είτε στην εύρεση κάποιας αδυναμίας ή στην αντιμετώπιση ή λύση της.

Τέλος μεγάλο ρόλο παίζει το κομμάτι της υποστήριξης/βιβλιογραφίας. Οι κοινότητες των CMS μπορούν να βοηθήσουν κάποιο χρήστη να πάρει τα κατάλληλα εφόδια για να μπορέσει να λύσει οποιοδήποτε πρόβλημα του παρουσιαστεί. Περισσότερα για τα πλεονεκτήματα ή μειονεκτήματα της πλατφόρμας θα αναλυθούν παρακάτω.

Στη συγκεκριμένη εργασία θα παρουσιαστεί αναλυτικά μια επέκταση για την πλατφόρμα Joomla που είναι μοναδική. Στη συγκεκριμένη πλατφόρμα θα μπορεί ο χρήστης με πολύ εύκολο τρόπο να φτιάξει μια ιστοσελίδα με στίχους, όπου θα μπορεί να βάλει όλα τα στοιχεία ενός τραγουδιού από τραγουδιστή, στιχουργό, άλμπουμ, κλπ.

Η συγκεκριμένη επέκταση ήρθε σαν ανάγκη από την κατασκευή της ιστοσελίδας του Κώστα Βίρβου και θα έχει ως αποτέλεσμα την πλήρη ανάπτυξη και τις ίδιες ιστοσελίδας στο συγκεκριμένο κομμάτι.





Abstract

Nowadays more and more developers are turning to web design through a CMS platform. Now considered necessary a developer involved in web design to have knowledge of their functioning and the possibility can customize and extend the needs of each case.

The main advantage of a CMS platform versus a page entirely from scratch is the efficiency and competitiveness that someone can have if he made from scratch. We note that a page construction time can be reduced by over 70%.

The second major advantage is scalability. A company's customers have increasing needs can not be met immediately. With the CMS extensions can cover more and more needs that can satisfy the most demanding user.

Another very important advantage of CMS is their security against possible malicious action against them. Can the one most to be open-source software taps which give the weaknesses however but then every user is a potential partner to help or find some weakness or treatment or solution.

Finally a big role plays the part of the support / literature. The CMS of communities' can help a user to get the right skills to be able to solve any problem occurred. More about the advantages or disadvantages of the platform will be analyzed below.

In this work will be presented in detail a new extension for Joomla platform that is unique. In this platform the user with an easy way to make a website with lyrics where you can put all the elements of a song by singer, songwriter, album, etc.

This expansion came as a need for the construction of Kostas Site Virvos and will result in the full development and the same site of the current track.



Ευχαριστίες

Θα ήθελα να ευχαριστήσω την καθηγήτριά μου και κόρη του στιχουργού, Μαρία Βίρβου για την ευκαιρία αυτή που μου έδωσε να περιηγηθώ στο χώρο και στο χρόνο μέσα από σπάνια ντοκουμέντα μιας εποχής καθώς και να κατασκευάσω μια σύγχρονη βάση δεδομένων για όλα τα έργα του μεγάλου μας στιχουργού.

Τέλος την κοπέλα μου, την οικογένειά μου και τους φίλους μου για τη στήριξη που μου παρέιχαν και την υπομονή που επέδειξαν όλα τα χρόνια της εκπαίδευσής μου ώστε να μείνω αφοσιωμένος στους στόχους μου.

Οκτώβριος 2016

Τριτσετάκης Δημήτριος



Περιεχόμενα

1.	Ανάλυση απαιτησεων	9
1.1.	Στοχοι.....	9
2.	Η γλώσσα κατασκευης (php, html, css).....	10
2.1.	Php.....	10
2.2.	Πλεονεκτήματα της php	10
2.3.	Html.....	11
2.4.	Css.....	11
3.	Η βάση δεδομενων (mysql, php my admin).....	12
3.1.	Mysql	12
3.2.	Πλεονεκτήματα της mysql.....	12
3.3.	Phpmyadmin.....	13
4.	Το προτυπο model-view controller (mvc)	14
4.1.	Γενικά	14
4.2.	Περιγραφή του προτυπου.....	14
4.3.	Αρχιτεκτονικες των web application frameworks: mvc.....	15
5.	Η πλατφορμα (joomla).....	17
5.1.	Ιστορια	17
5.2.	Χαρακτηριστικα του joomla.....	17
5.3.	Βασικα συστατικα λογισμικου	18
5.4.	Αρχιτεκτονικη 3 επιπεδων	18
5.5.	Δομη του joomla	20
6.	Κατασκευη joomla component.....	21
6.1.	Εισαγωγή.....	21
6.2.	Joomla component	21
6.3.	Ανάλυση δομης com_lyrics.....	22
6.4.	Ροη δεδομενων.....	24
6.5.	Περιπτώσεις χρήσης.....	26
6.6.	Εικονες τραγουδιων	28
6.7.	Διαγραμμα βασης δεδομενων	29
7.	Οδηγίες χρήσης	31
7.1.	Δημιουργία νέου τραγουδιού	31
7.2.	Δημιουργία νέου Στιχουργού	35
7.3.	Δημιουργία νέου Συνθέτη	36
7.4.	Δημιουργία νέου Τραγουδιστη.....	37
7.5.	Δημιουργία νέας Εκτέλεσης.....	39
7.6.	Δημιουργία νέου Album	41
7.7.	Δημιουργία νέου Έτους	42
8.	Συμπεράσματα.....	43
9.	Βιβλιογραφία	44



1. ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

1.1. ΣΤΟΧΟΙ

Πλατφόρμα

Η πλατφόρμα που επιλέχθηκε για να φτιάξουμε την επέκταση είναι το CMS Joomla. Είναι το δημοφιλέστερο σύστημα διαχείρισης περιεχομένου με τα περισσότερα πλεονεκτήματα, εξαιρετικά ευέλικτο και αποτελεσματικό!

Η εφαρμογή αυτή μπορεί να αξιοποιηθεί είτε για προσωπική είτε για εταιρική χρήση με απίστευτες δυνατότητες επέκτασης. Στην παρούσα διπλωματική θα γίνει εκτενής ανάλυση στο σύστημα αυτό, διότι αποτελεί αντικείμενο της μελέτης μας.

Τεχνολογίες

Γνωρίζοντας ότι η πλατφόρμα που θα χρησιμοποιήσουμε είναι γραμμένη σε PHP, HTML και CSS και αποθηκεύει τα δεδομένα του στη βάση MySQL, εύκολα καταλαβαίνουμε ότι η επέκτασή μας θα είναι σε γλώσσα PHP, HTML και CSS και η βάση δεδομένων θα κατασκευαστεί ως επέκταση της MYSQL βάσης του JOOMLA.

Λειτουργικός Σχεδιασμός

Στο συγκεκριμένο κομμάτι θα γίνει ιδιαίτερη ανάλυση στις απαιτήσεις που είχε το project καθόλη τη διάρκεια του development καθώς και στις νέες λειτουργίες που προστίθενται κάθε φορά σε κάθε νέα έκδοση.

Τέλος θα γίνει ειδική μνεία στα προβλήματα που δημιουργήθηκαν (Bugs) στον κώδικα, καθώς και τον τρόπο που επιλύθηκαν.

Γραφιστικός Σχεδιασμός

Το design και για να μπορεί η σελίδα να είναι λειτουργική σε όλες τις διαστάσεις (responsive) χρησιμοποιήθηκε το CSS και JS Framework Bootstrap, μια συλλογή εργαλείων ανοιχτού κώδικα για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών. Είναι το πιο δημοφιλές πρόγραμμα στο GitHub και έχει χρησιμοποιηθεί από τη NASA και το MSNBC, μεταξύ άλλων.



2. Η ΓΛΩΣΣΑ ΚΑΤΑΣΚΕΥΗΣ (PHP, HTML, CSS)

2.1. PHP

Η PHP είναι μια πολύ δημοφιλής γλώσσα συγγραφής σεναρίων στην πλευρά του server (server-side scripting language) καθώς είναι γρήγορη, διαθέτει πολλές ενσωματωμένες συναρτήσεις και το σημαντικότερο, είναι δωρεάν.

Πολλές εταιρείες που παρέχουν υπηρεσίες Web hosting υποστηρίζουν την PHP και υπάρχει μια μεγάλη κοινότητα χρηστών της PHP που είναι διαθέσιμη να βοηθήσει οποιονδήποτε συναντήσει κάποιο πρόβλημα.

Η PHP δημιουργήθηκε από μια σειρά από σενάρια (scripts) της Perl (ή της διεπαφής CGI – Common Gateway Interface) που γράφτηκαν από τον Rasmus Lerdorf το 1995. Η PHP αρχικά σήμαινε Personal Home Page και γράφτηκε για να καταγράφει τον αριθμό των επισκέψεων σε μια ιστοσελίδα. Στην πορεία άλλαξε το ακρωνύμιό της και σημαίνει πλέον Hypertext Preprocessor.

Η PHP είναι ένα εργαλείο για τη δημιουργία δυναμικών ιστοσελίδων (dynamic web pages) και η παρουσία της είναι απόλυτα διαφανής (αόρατη) στον τελικό χρήστη. Μια ιστοσελίδα που περιέχει κάποιον κώδικα σε PHP υφίσταται προεπεξεργασία από τη μηχανή της PHP, που αποκαλείται διερμηνευτής (interpreter), και τα αποτελέσματα αυτής της επεξεργασίας στέλνονται πίσω στον Web server και από εκεί στον φυλλομετρητή του χρήστη (επισκέπτη) της ιστοσελίδας ως καθαρός HTML κώδικας.

Καθώς μόνο τα αποτελέσματα της επεξεργασίας του PHP κώδικα είναι που στέλνονται στον φυλλομετρητή, ο κώδικας που τα δημιούργησε παραμένει κρυφός και συνεπώς πολύ πιο ασφαλής. Αυτό το είδος της προεπεξεργασίας αποκαλείται server-side scripting και οι PHP ιστοσελίδες αποκαλούνται δυναμικές.

2.2. Πλεονεκτήματα της PHP

Κάποιοι από τους βασικούς ανταγωνιστές της PHP είναι ο Perl, Microsoft Active Server Page (ASP), Java Server Pages (JSP) και Allaire Cold Fusion.

Σε σύγκριση με αυτά τα προϊόντα, η PHP έχει πολλά πλεονεκτήματα όπως:

- **Υψηλή Απόδοση**
Η PHP είναι πολύ αποτελεσματική. Με ένα φθινό διακομιστή μπορείτε να εξυπηρετήσετε εκατομμύρια επισκέψεων καθημερινά. Οι δοκιμές που δημοσιεύθηκαν από την Zend Technologies δείχνουν ότι η PHP ξεπερνά τους ανταγωνιστές της.
- **Διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων δεδομένων**
Η PHP έχει εγγενείς συνδέσεις για πολλά συστήματα βάσεων δεδομένων. Εκτός από την MySQL μπορείτε να συνδεθείτε με τις βάσεις δεδομένων PostgreSQL , mSQL, Oracle , dbm, filePro , Informix , InterBase , Sybase , κ.α
- **Ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες Web διαδικασίες**
Επειδή η PHP σχεδιάστηκε για να χρησιμοποιείται στο Web, έχει πολλές ενσωματωμένες βιβλιοθήκες , που εκτελούν πολλές χρήσιμες λειτουργίες σχετικές με το Web. Μπορείτε να δημιουργήσετε εικόνες GIF δυναμικά , να συνδεθείτε με άλλες υπηρεσίες δικτύων , να στείλετε ηλεκτρονικό ταχυδρομείο , να δουλέψετε με cookies και να δημιουργήσετε PDF έγγραφα και όλα αυτά με λίγες γραμμές κώδικα.
- **Μηδενικό κόστος**
Η PHP είναι δωρεάν. Μπορείτε να κατεβάσετε την τελευταία έκδοση της από το <http://www.php.net> , χωρίς χρέωση.
- **Ευκολία μάθησης και χρήσης**
Η σύνταξη της PHP βασίζεται σε άλλες γλώσσες προγραμματισμού , βασικά στην C και στην Perl .



- **Φορητότητα**
Η PHP είναι διαθέσιμη για πολλά λειτουργικά συστήματα . Μπορείτε να γράψετε κώδικα PHP για δωρεάν συστήματα τύπου Unix, όπως LINUX και FreeBSD, για εμπορικές εκδόσεις του UNIX, όπως το Solaris και το IRIX ή για διαφορετικές εκδόσεις των Microsoft Windows. Ο κώδικας σας συνήθως θα δουλεύει χωρίς αλλαγές στα συστήματα που τρέχουν την PHP.
- **Διαθεσιμότητα του κώδικα προέλευσης**
Έχετε πρόσβαση στον κώδικα προέλευσης της PHP. Αντίθετα με εμπορικά, κλειστά προγράμματα, αν υπάρχει κάτι που θέλετε να αλλάξετε ή να προσθέσετε στη γλώσσα μπορείτε να το κάνετε. Δεν χρειάζεται να περιμένετε τον κατασκευαστή να εμφανίσει διορθώσεις. Δεν θα ανησυχείτε αν ο κατασκευαστής θα σταματήσει να υπάρχει ή αν θα σταματήσει να υποστηρίζει το προϊόν.

2.3. HTML

Η HTML είναι το ακρωνύμιο των λέξεων HyperText Markup Language (γλώσσα μορφοποίησης υπερκειμένου) και είναι η βασική γλώσσα δόμησης σελίδων του World Wide Web. Είναι μία γλώσσα προγραμματισμού. Χρησιμοποιείται για τη σήμανση ενός τμήματος κειμένου και την καλύτερη εμφάνισή του. Επιτρέπει την ενσωμάτωση ήχου και εικόνων στις web σελίδες. Αρχικά είχε κατασκευασθεί με σκοπό μόνο την μορφοποίηση κειμένου, αλλά μεγάλωσε και ενσωμάτωσε σχεδιαστικές τεχνικές κ.α.

Η γλώσσα χρησιμοποιεί ένα αριθμό από tags για την μορφοποίηση κειμένου, για την δημιουργία συνδέσμων (links) μετάβασης ανάμεσα των σελίδα, για την εισαγωγή εικόνων, ήχου κ.α.

Όταν ένας Web Browser ανοίγει ένα αρχείο HTML τα στοιχεία (tags) μεταφράζονται σε κατάλληλα χαρακτηριστικά με αποτελέσματα στην εμφάνιση και στην λειτουργικότητα της συγκεκριμένης σελίδας.

Η δημιουργία αρχείων HTML είναι πολύ απλή. Αρκεί να “τρέξουμε” έναν οποιοδήποτε διορθωτή κειμένου text, όπως το Notepad των Windows. Να γράψουμε τον κώδικα HTML που επιθυμούμε και να το αποθηκεύσουμε σε ένα αρχείο με κατάληξη .htm ή .html.

Συνήθως αποθηκεύουμε με επέκταση .html όταν θα δημοσιεύσουμε τα αρχεία μας σε διακομιστή Unix. Αντίθετα ο εξυπηρετητής Windows NT καταλαβαίνει τα αρχεία και σαν .htm .

2.4. CSS

Η CSS (Cascading Style Sheets-Διαδοχικά Φύλλα Στυλ) ή (αλληλουχία φύλλων στυλ) είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης.

Χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστοτόπου.

Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στυλιστικά μια ιστοσελίδα δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και δίνει περισσότερες δυνατότητες σε σχέση με την html.

Τα μεγαλύτερα πλεονεκτήματά της είναι τα εξής:

- Δυνατότητα για τον ταυτόχρονο έλεγχο της διάταξης (layout) πολλών εγγράφων από μόνο ένα φύλλο στυλ (stylesheet).
- Πιο ακριβής έλεγχος της διάταξης των ιστοσελίδων.
- Εφαρμογή διαφορετικής διάταξης σε διαφορετικές εξόδους όπως screen, print κ.α.
- Αναρίθμητες προηγμένες και εξεζητημένες τεχνικές.



3. Η ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ (MYSQL, PHP MY ADMIN)

3.1. MySQL

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων που μετρά περισσότερες από 11 εκατομμύρια εγκαταστάσεις. Ο πηγαίος κώδικας της είναι διαθέσιμος με ελεύθερη άδεια χρήσης.

Η ονομασία MySQL περιέχει δύο στοιχεία. Το My προέρχεται από το όνομα της κόρης του συνιδρυτή του συστήματος Μόντι Βιδένιους (Monty Widenius) και το SQL αναφέρεται στη γλώσσα SQL (Structured Query Language), μια γλώσσα υπολογιστών που σχεδιάστηκε ανεξάρτητα από την MySQL για τη διαχείριση δεδομένων, σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (MySQL, PostgreSQL, Oracle κλπ).

Το πρότυπο της γλώσσας SQL αναπτύχθηκε ξεχωριστά από τις υλοποιήσεις συστημάτων διαχείρισης βάσεων δεδομένων (όπως της MySQL, της PostgreSQL, της Oracle κλπ).

Θεωρητικά ένας κώδικας SQL θα πρέπει να τρέχει με τον ίδιο τρόπο και σε μια βάση MySQL όπως και σε μια βάση PostgreSQL. Κάθε υλοποίηση ενός συστήματος διαχείρισης σχεσιακών βάσεων δεδομένων μπορεί να διαφέρει με κάποιο άλλο παρέχοντας επιπλέον δυνατότητες, βελτιωμένα χαρακτηριστικά. Στην πράξη η μεταφορά ενός κώδικα SQL από μια βάση σε μια διαφορετική υλοποίηση μπορεί να παρουσιάσει ασυμβατότητες ή θα πρέπει να γίνουν τροποποιήσεις στον κώδικα.

Η MySQL είναι δημοφιλής βάση δεδομένων για διαδικτυακά προγράμματα και ιστοσελίδες.

Χρησιμοποιείται σε κάποιες από τις πιο διαδεδομένες διαδικτυακές υπηρεσίες, όπως το Flickr, το YouTube, η Wikipedia, το Google, το Facebook και το Twitter.

3.2. Πλεονεκτήματα της MySQL

Μερικοί από τους κύριους ανταγωνιστές της MySQL είναι οι PostgreSQL, Microsoft SQL και Oracle αλλά έχει πολλά πλεονεκτήματα όπως θα δούμε παρακάτω:

- **Απόδοση**
Η MySQL είναι χωρίς αμφιβολία γρήγορη. Μπορείτε να δείτε στη σελίδα δοκιμών <http://web.mysql.com/benchmark.html> που αποκλείουν κάθε ανταγωνισμό.
- **Χαμηλό κόστος**
Η MySQL είναι διαθέσιμη δωρεάν, με άδεια ανοικτού κώδικα (OpenSource) ή με χαμηλό κόστος, αν πάρετε εμπορική άδεια, αν απαιτείται από την εφαρμογή σας.
- **Ευκολία Χρήσης**
Οι περισσότερες μοντέρνες βάσεις δεδομένων χρησιμοποιούν SQL. Αν έχετε χρησιμοποιήσει ένα άλλο σύστημα διαχείρισης βάσεων δεδομένων δεν θα έχετε πρόβλημα να προσαρμοστείτε σε αυτό.
- **Φορητότητα**
Η MySQL μπορεί να χρησιμοποιηθεί σε πολλά διαφορετικά συστήματα Unix όπως επίσης και στα Microsoft Windows.
- **Κώδικας Προέλευσης**
Όπως και με την PHP, μπορείτε να πάρετε και να τροποποιήσετε τον κώδικα προέλευσης της MySQL.
- **Νέες εκδόσεις**
Η MySQL αναπτύσσεται συνεχώς και έρχεται με νέες εντυπωσιακές λειτουργίες που είναι ικανή να υποστηρίξει πολύ μεγάλα projects με υψηλή αξιοπιστία.



3.3. PhpMyAdmin

Το PhpMyAdmin είναι ένα εργαλείο γραμμένο σε php με το οποίο διαχειριζόμαστε τις βάσεις δεδομένων που έχουμε μέσω web. Το phpMyAdmin μπορεί να χειρίζεται πλήρως βάσεις δεδομένων, πίνακες, πεδία πινάκων αλλά και ένα ολόκληρο MySQL Server.

Υποστηρίζει 54 γλώσσες, μεταξύ των οποίων και τα ελληνικά και είναι λογισμικό ανοιχτού κώδικα.

- Δυνατότητες του PhpMyAdmin
- Δημιουργεί και να διαγράφει βάσεις δεδομένων
- Δημιουργεί, τροποποιεί, διαγράφει, αντιγράφει και μετονομάζει πίνακες
- Κάνει συντήρηση της βάσης
- Προσθέτει, διαγράφει και τροποποιεί πεδία πινάκων
- Εκτελεί ερωτήματα SQL ακόμα και ομαδικά (batch)
- Διαχειρίζεται κλειδιά σε πεδία
- Φορτώνει αρχεία κειμένου σε πίνακες
- Δημιουργεί και διαβάζει πίνακες (που προέρχονται από dump βάσης)
- Εξάγει και εισάγει δεδομένα σε μορφή CVS, XML και LATEX
- Διαχειρίζεται πολλούς διακομιστές
- Διαχειρίζεται τους χρήστες MySQL και τα δικαιώματά τους
- Ελέγχει την αναφορική δραστηριότητα των δεδομένων των MyISAM πινάκων
- Δημιουργεί PDF γραφικών του layout της βάσης δεδομένων
- Εκτελεί αναζητήσεις σε όλη τη βάση δεδομένων ή μέρος αυτής
- Υποστηρίζει πίνακες InnoDB και ξένα κλειδιά
- Υποστηρίζει MySQLi, μια βελτιωμένη επέκταση του MySQL

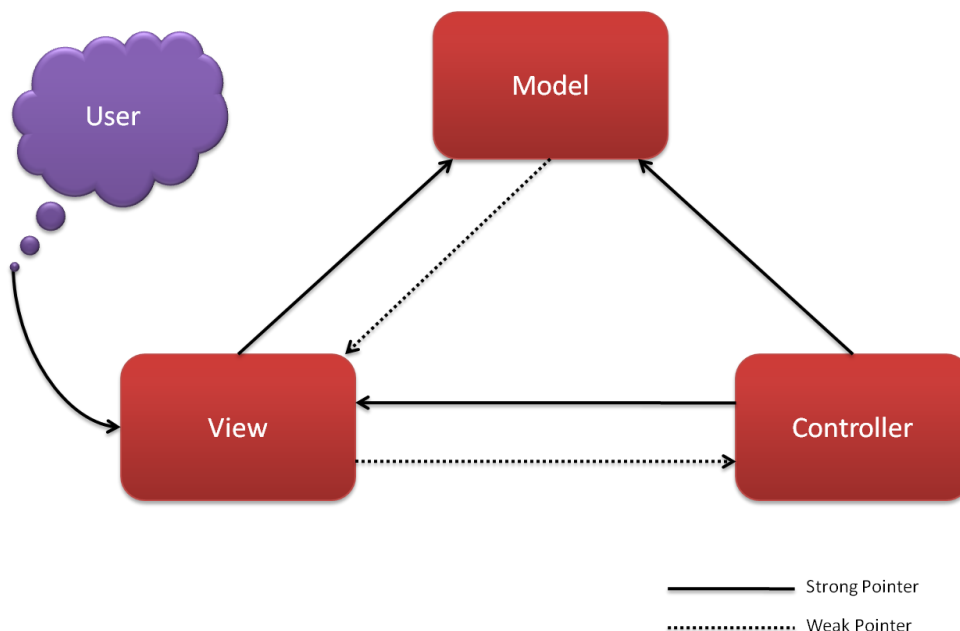


4. ΤΟ ΠΡΟΤΥΠΟ MODEL-VIEW CONTROLLER (MVC)

4.1. Γενικά

Το Model-View-Controller (MVC) είναι ένα αρχιτεκτονικό πρότυπο που χρησιμοποιείται στην τεχνολογία λογισμικού και συνήθως στη δημιουργία web εφαρμογών. Η επιτυχής χρήση του προτύπου απομονώνει τη λογική της web εφαρμογής από το user interface με το οποίο αλληλεπιδρά ο χρήστης, έχοντας έτσι ως αποτέλεσμα μία εφαρμογή πιο ευέλικτη, όπου είναι ευκολότερο να μεταβάλλει κάποιος είτε τον τρόπο παρουσίασης της εφαρμογής της είτε τον υποκείμενο κώδικα που εφαρμόζει λειτουργίες στα δεδομένα, χωρίς τα επιμέρους τμήματα να αλληλοεπηρεάζονται.

Στην MVC αρχιτεκτονική, το model αναπαριστά την πληροφορία, τα δεδομένα που η εφαρμογή χρησιμοποιεί. Το view αντιστοιχεί σε τρόπο παρουσίασης των δεδομένων του model στο interface του χρήστη π.χ.: text boxes, checkboxes κ.λπ., Τέλος ο controller διαχειρίζεται τη σύνδεση των δεδομένων του model με τη λογική του προγράμματος και καθορίζει ποια από τα δεδομένα του model θα παρουσιαστούν από το view στο interface του χρήστη.



Σχήμα 1 : Ένα απλό διάγραμμα που παρουσιάζει τη σχέση-σύνδεση ανάμεσα στα model view controller. Οι συνεχόμενες γραμμές αντικατοπτρίζουν άμεση σύνδεση ενώ οι διακεκομμένες την έμμεση.

4.2. ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΤΥΠΟΥ

Το πρότυπο Model-view-controller είναι τόσο ένα αρχιτεκτονικό πρότυπο όσο και ένα σχεδιαστικό πρότυπο, αναλόγως που χρησιμοποιείται.

Σαν αρχιτεκτονικό πρότυπο

Είναι συνηθισμένο μία εφαρμογή να διασπάται σε χωριστά επίπεδα τα οποία τρέχουν σε διαφορετικούς υπολογιστές: Επίπεδο Παρουσίασης(UI), Επίπεδο Λογικής, και επίπεδο Πρόσβασης στα Δεδομένα. Στο MVC το επίπεδο παρουσίασης χωρίζεται περαιτέρω σε view και controller.

Το πρότυπο MVC συχνά συναντάται σε web εφαρμογές, όπου το view είναι απλά μία html σελίδα, και ο controller είναι ο κώδικας που περισυλλέγει δυναμικά δεδομένα και δημιουργεί το



περιεχόμενο μέσα στο HTML. Τέλος το model αντιπροσωπεύει το περιεχόμενο, που συνήθως βρίσκεται αποθηκευμένο σε κάποια βάση δεδομένων ή σε XML κόμβους(data) καθώς επίσης και τους κανόνες(metadata) που μετατρέπουν το περιεχόμενο αυτό ανάλογα με την αλληλεπίδραση του χρήστη.

Αν και για το πρότυπο MVC υπάρχει μια μεγάλη ποικιλία εκδόσεων, ένας γενικός κύκλος λειτουργίας του θα μπορούσε να περιγραφεί όπως παρακάτω:

1. Ο χρήστης αλληλεπιδρά με το User Interface με κάποιο τρόπο.
2. Ο controller χειρίζεται το event εισόδου από το User Interface, συνήθως μέσω κάποιου event handler.
3. Ο controller ειδοποιεί το model για το event που έλαβε χώρα, συνήθως μεταβάλλοντας την κατάσταση του model με κάποιο τρόπο.
4. Το view χρησιμοποιεί το model(έμμεσα) για να δημιουργήσει το κατάλληλο νέο User Interface. Το view παίρνει επίσης δεδομένα από το model. Το model όμως δεν έχει άμεση γνώση του view.
5. Το user interface περιμένει για επόμενη ενέργεια του χρήστη, η οποία ξεκινά ένα νέο κύκλο.

Αποσυνδέοντας το model και το view, το πρότυπο MVC μειώνει κατά πολύ την πολυπλοκότητα της αρχιτεκτονικής μιας web εφαρμογής ενώ ταυτόχρονα αυξάνει την προσαρμοστικότητά και την ευελιξία της.

Σαν πρότυπο σχεδίασης

Το πρότυπο MVC περιλαμβάνει πολύ περισσότερα τμήματα της αρχιτεκτονικής μιας εφαρμογής από τα συνηθισμένα ενός σχεδιαστικού προτύπου.

Model

Πρόκειται για μία domain-specific αναπαράσταση της πληροφορίας πάνω στην οποία ενεργεί η εφαρμογή. Η λογική της εφαρμογής (domain-logic) προσθέτει νόημα στα ανεπεξέργαστα δεδομένα.

Πολλές εφαρμογές χρησιμοποιούν έναν μόνιμο (σταθερό) αποθηκευτικό μηχανισμό (όπως π.χ.: μία βάση δεδομένων) για να αποθηκεύουν τα δεδομένα τους.

View

Παρουσιάζει τα δεδομένα του model σε μία μορφή κατάλληλη για αλληλεπίδραση, συνήθως user interface στοιχεία. Είναι δυνατόν πολλαπλά διαφορετικά views να υπάρχουν για ένα μοναδικό model για διαφορετικούς σκοπούς.

Controller

Επεξεργάζεται και ανταποκρίνεται σε events, που συνήθως προκαλούνται από επιλογές του χρήστη και είναι πιθανόν να επιδρά και να αλλάζει το model.

4.3. ΑΡΧΙΤΕΚΤΟΝΙΚΕΣ ΤΩΝ WEB APPLICATION FRAMEWORKS: MVC.

Πολλά frameworks ακολουθούν το Model-View-Controller(MVC) αρχιτεκτονικό πρότυπο για να διαχωρίσουν το data model από το business rules και το user interface.

Push-based vs. Pull-based.

Τα περισσότερα MVC frameworks ακολουθούν push-based αρχιτεκτονική. Χρησιμοποιούν ενέργειες που επιτελούν την απαραίτητη επεξεργασία και μετά «σπρώχνουν» τα δεδομένα στο επίπεδο παρουσίασης. Μια εναλλακτική είναι η pull-based αρχιτεκτονική όπου το framework ξεκινά με το επίπεδο παρουσίασης το οποίο «τραβά» αποτελέσματα από πιθανούς πολλούς controllers όπου και όπως χρειάζεται.



Χαρακτηριστικά ασφαλείας των framework

Κάποια web application Frameworks παρέχουν authentication και authorization επιτρέποντας έτσι στους web servers να αναγνωρίζουν τους χρήστες των εφαρμογών και να απαγορεύουν την είσοδο σε λειτουργίες ανάλογα με κάποια κριτήρια.

Web Template System

Οι δυναμικές ιστοσελίδες συνήθως αποτελούνται από ένα στατικό και ένα δυναμικό τμήμα που είναι κώδικας που δημιουργεί HTML. Ο κώδικας που δημιουργεί το HTML βασίζεται σε μεταβλητές που ανήκουν σε κάποιο template ή σε κώδικα. Τα περιεχόμενα που δημιουργούνται μπορεί να προέρχονται από κάποια βάση δεδομένων μειώνοντας έτσι τις σελίδες σε ένα web-site.

Για παράδειγμα, ένα μεσιτικό γραφείο με 500 σπίτια προς πώληση, σε ένα static web-site θα έπρεπε να διαθέτει 500 σελίδες, μία για κάθε σπίτι, για να διαθέτει όλη την πληροφορία στο δίκτυο. Αντιθέτως, σε ένα dynamic web-site ο μεσίτης απλά θα συνδέει την δυναμική σελίδα στη βάση που αποτελείται από τις 500 εγγραφές (κάθε μία από τις οποίες αποτελεί ένα από τα προς πώληση σπίτια).

Ajax framework

Ο Ajax (Asynchronous Javascript and XML) είναι μία τεχνική ανάπτυξης διαδραστικών εφαρμογών web. Ο σκοπός είναι η δημιουργία web σελίδων που ανταποκρίνονται ανταλλάσσοντας ελάχιστη πληροφορία με το server, ούτως ώστε να μη χρειάζεται εξολοκλήρου επαναφόρτωση(reloading) της σελίδας κάθε φορά που ο χρήστης ζητά μια αλλαγή.

Κατ' επέκτασίν η τεχνική αυτή στόχο έχει την αύξηση της διαδραστικότητας, της ταχύτητας απόκρισης και της χρησιμοποίησης της ιστοσελίδας. Πρόκειται στην ουσία για ένα framework που χρησιμοποιείται για την ανάπτυξη web εφαρμογών που χρησιμοποιούν μια γκάμα τεχνολογιών για την κατασκευή δυναμικών web σελίδων στην πλευρά του client. Τα δεδομένα διαβάζονται από κάποιο server ή επίσης στέλνονται στο server με Javascript requests.

Παρόλ' αυτά κάποια επεξεργασία συνήθως είναι απαραίτητη στην πλευρά του server, όπως αποθήκευση πληροφορίας κ.λπ. Αυτή η επεξεργασία είναι γίνεται πιο εύκολη με τη χρήση κάποιου framework. Ο στόχος λοιπόν του Ajax framework είναι να συνδέσει αποδοτικά clientside και server-side λειτουργίες.

Οφέλη των frameworks

Ένα Ajax framework έχει χαρακτηριστεί ως μεσάζον ανάμεσα στον client και το server. Οι Ajax τεχνολογίες σκοπό έχουν την μείωση της καθυστέρησης που αντιλαμβάνεται ο χρήστης όταν η σελίδα προσπαθεί να προσπελάσει το server. Ένα framework διευκολύνει έναν προγραμματιστή Ajax τεχνολογιών σε δύο επίπεδα. Στον client υπάρχουν συναρτήσεις που στέλνουν requests στο server και στον server υπάρχουν κατάλληλες συναρτήσεις που επεξεργάζονται τα requests και επαναποστέλνουν τα δεδομένα πίσω στο browser του client.

Παραδείγματα Javascript Ajax frameworks

Τα JavaScript Ajax Frameworks είναι browser-side frameworks που συχνά χρησιμοποιούνται στην ανάπτυξη web εφαρμογών με χρήση του Ajax. Παρακάτω παρατίθενται κάποια γνωστά Ajax frameworks που χρησιμοποιούνται για τη δημιουργία web εφαρμογών με δυναμική σύνδεση ανάμεσα στον client και το server.

- Dojo Toolkit: Open Source DHTML toolkit γραμμένο σε JavaScript
- jQuery: ένα JavaScript framework λειτουργεί σαν Ajax framework αλλά παρέχει και επιπρόσθετες λειτουργίες.



5. Η ΠΛΑΤΦΟΡΜΑ (Joomla)

5.1. ΙΣΤΟΡΙΑ

Το Joomla αποτελεί παιδί ενός άλλου γνωστού CMS, του Mambo του οποίου υπήρξε fork. Με τον όρο fork στη Μηχανική Λογισμικού αναφερόμαστε σε ένα project όπου μια ομάδα developers παίρνει μία νόμιμη κópια ενός λογισμικού και αρχίζει ανεξάρτητη υλοποίηση επάνω σε αυτό. Η διαδικασία ξεκίνησε τον Αύγουστο του 2005, όταν τα πνευματικά δικαιώματα του Mambo ανήκαν στην εταιρεία Miro Intl Pty Ltd. Η εταιρεία δημιουργεί έναν μη κερδοσκοπικό οργανισμό με σκοπό να χρηματοδοτήσει το project και να το προστατέψει από πιθανές αγωγές.

Η ομάδα υλοποίησης εκφράζει ρητά τις ενστάσεις της και τις διαφωνίες της σχετικά με διάφορους όρους που έχει θέσει η εταιρεία οι οποίοι κατά τη γνώμη των μελών της αντιβαίνουν με τους κανόνες χρήσης και διάδοσης του λεγόμενου λογισμικού ανοιχτού πηγαίου κώδικα (open source). Η ομάδα υλοποίησης λοιπόν του Joomla (που τότε δεν είχε ακόμη καν όνομα) δημιουργεί ένα website με ονομασία 'OpenSourceMatters' όπου εκφράζει την αντίθεσή της με τους όρους της Miro καθώς και τις θέσεις της σχετικά με την ανάγκη ύπαρξης ελεύθερου λογισμικού.

Η κίνηση αυτή προσελκύει πολύ κόσμο ο οποίος ενθαρρύνει την ομάδα υλοποίησης ώστε αυτή να κινηθεί προς την κατεύθυνση λανσαρίσματος του fork στο ευρύ κοινό. Μη έχοντας ακόμη επίσημο όνομα για το project, στις 18 Αυγούστου του 2005 ο Andrew Blake (επικεφαλής της ομάδας υλοποίησης) καλεί τους φίλους της κίνησης να προτείνουν ονόματα για το project. Την 1η του Σεπτεμβρίου του 2005 ανακοινώνεται στο κοινό πως το όνομα που επιλέχθηκε ήταν το Joomla, Αραβική λέξη η μεταφράζεται στην Ελληνική ως 'συνολικά'.

Λίγο αργότερα τον ίδιο μήνα, στις 16, έχουμε το επίσημο λανσάρισμα της πρώτης έκδοσης του Joomla, ενώ τις επόμενες εβδομάδες ακολουθεί το λανσάρισμα του επίσημου λογοτύπου και του εγχειριδίου χρήσης. Η πρώτη έκδοση του Joomla (1.0.0) ήταν στην ουσία ένα αντίγραφο του Mambo 4.5.2.3 στο οποίο είχαν γίνει μικροδιορθώσεις σε bugs και είχαν εφαρμοστεί patches σε γνωστές 'τρύπες' ασφαλείας. Η δεύτερη έκδοση του Joomla που είναι η 1.5.0 λανσάρεται στις 22 Ιανουαρίου του 2008 και εισάγει πολλές αλλαγές στον τρόπο λειτουργίας της πλατφόρμας αλλά και νέα λειτουργικότητα σε σχέση με τον προκάτοχό της. Από την πρώτη έκδοσή του μέχρι και σήμερα το Joomla έχει γίνει εξαιρετικά δημοφιλές και σύμφωνα με την επίσημη αναφορά Open Source CMS Market Share Report, από τον Οκτώβριο του 2009 το Joomla είναι το πιο δημοφιλές Web CMS.

5.2. ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΟΥ JOOMLA

Το Joomla είναι ένα Web CMS ανοιχτού πηγαίου κώδικα, που σημαίνει πως μπορείτε να το χρησιμοποιήσετε τόσο για τη δημιουργία sites μη κερδοσκοπικού χαρακτήρα όσο και εμπορικών χωρίς να απαιτείται η καταβολή κάποιου χρηματικού ποσού στους developers της πλατφόρμας ή η αγορά κάποιας άδειας χρήσης. Αυτό είναι και ένα από τα μεγαλύτερα πλεονεκτήματά του έναντι άλλων Web CMS, αν και σήμερα υπάρχει όπως προαναφέρθηκε πληθώρα open source Web CMS για όλες τις ανάγκες και όλες τις απαιτήσεις. Αποτελείται από έναν πυρήνα στον οποίο ενσωματώνεται η συνολική λειτουργικότητα υπό τη μορφή modules, components, plugins και templates. Κάποια από τα βασικότερα χαρακτηριστικά του είναι τα εξής:

- Υποστηρίζει διαχείριση και πιστοποίηση (authentication) χρηστών
- Υποστηρίζει διαφορετικές γλώσσες (internationalization)
- Είναι πλήρως επεκτάσιμο
- Υποστηρίζει χρήση web services
- Επιτρέπει στους χρήστες να συνεισφέρουν στο περιεχόμενο κάνοντας χρήση WYSIWYG editors
- Έχει ενσωματωμένο σύστημα βοήθειας προς τον χρήστη



- Περιέχει ενσωματωμένο FTP Manage
- Επιτρέπει την εύκολη εναλλαγή προτύπων (templates)
- Υποστηρίζει χρήση URL που είναι φιλικά προς τις μηχανές αναζήτησης
- Υποστηρίζει πληθώρα χρήσιμων λειτουργιών όπως:
 - Δημοψηφίσματα (polls)
 - Φόρμες επικοινωνίας
 - Νέα με τη μορφή RSS Feeds
 - Διαχείριση επαφών και email
 - Προβολή σελίδων ως pdf αρχεία
 - Σελίδες φιλικές στην εκτύπωση

5.3. ΒΑΣΙΚΑ ΣΥΣΤΑΤΙΚΑ ΛΟΓΙΣΜΙΚΟΥ

Όλες οι σοβαρές επιχειρησιακές εφαρμογές λογισμικού αποτελούνται από τέσσερα βασικά συστατικά. Τα συστατικά αυτά από κάτω προς τα πάνω (lower to higher level) είναι τα εξής:

1. **Αποθήκευση δεδομένων (data storage):**
Πρόκειται για το κομμάτι της εφαρμογής που είναι υπεύθυνο για την αποθήκευση των δεδομένων. Κάθε εφαρμογή διαχειρίζεται δεδομένα τα οποία και αποθηκεύει είτε σε μία βάση δεδομένων είτε σε κάποιας μορφής αρχείο. Ο τρόπος με τον οποίο αποθηκεύονται τα δεδομένα αυτά είναι ευθύνη του συγκεκριμένου συστατικού.
2. **Λογική πρόσβασης δεδομένων (data access logic):**
Το συγκεκριμένο συστατικό της εφαρμογής είναι υπεύθυνο να παρέχει έναν αξιόπιστο μηχανισμό ελέγχου πρόσβασης των δεδομένων της ανάλογα με τα δικαιώματα του κάθε χρήστη, χωρίς ο μηχανισμός αυτός να είναι πολύπλοκος. Στην περίπτωση που τα δεδομένα αποθηκεύονται σε μια σχεσιακή βάση δεδομένων τον ρόλο αυτόν αναλαμβάνει το RDBMS, ενώ στην περίπτωση που τα δεδομένα αποθηκεύονται σε κάποιο αρχείο ο μηχανισμός υλοποιείται είτε από την ίδια την εφαρμογή, είτε από το file system είτε χρησιμοποιείται συνδυασμός και των δύο.
3. **Επιχειρησιακή λογική (application logic):**
Πρόκειται για το καθαρά προγραμματιστικό κομμάτι της εφαρμογής που περιγράφει τις διαδικασίες και τον τρόπο λειτουργίας της επιχείρησης-οργανισμού που επιθυμεί να τις αυτοματοποιήσει κάνοντας χρήση της εν λόγω εφαρμογής.
4. **Λογική παρουσίασης (presentation logic):**
Το συστατικό της εφαρμογής που περιγράφει τον τρόπο με τον οποίο υλοποιείται η διάδραση του χρήστη με αυτήν. Συνήθως πρόκειται για κάποιας μορφής GUI που υλοποιείται σε μία high-level γλώσσα προγραμματισμού και κάνει χρήση του event-driven μοντέλου προγραμματισμού, ή ενός web interface στο οποίο ο χρήστης έχει πρόσβαση μέσω ενός web browser.

Ανάλογα με την αρχιτεκτονική που χρησιμοποιείται από τους μηχανικούς για την υλοποίηση της εφαρμογής, τα παραπάνω συστατικά μπορεί να υλοποιούνται σε ένα ή σε περισσότερα επίπεδα.

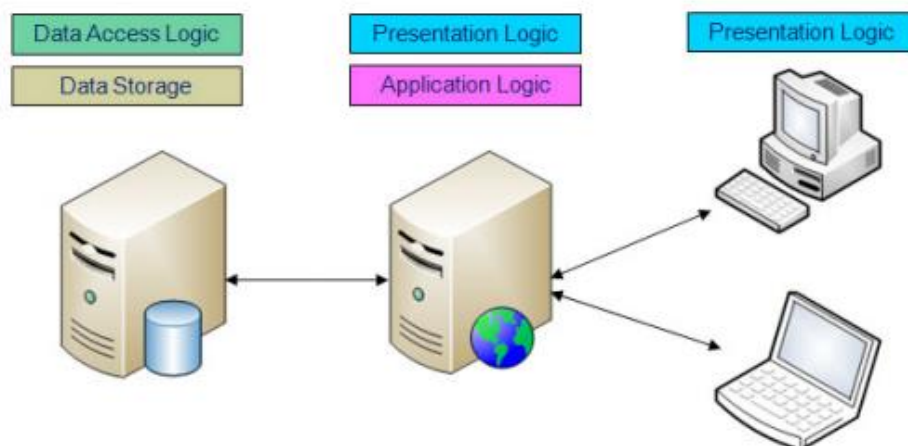
5.4. ΑΡΧΙΤΕΚΤΟΝΙΚΗ 3 ΕΠΙΠΕΔΩΝ

Οι σύγχρονες διαδικτυακές εφαρμογές κάνουν χρήση μιας αρχιτεκτονικής γνωστής ως multi-tiered που στα Ελληνικά σημαίνει αρχιτεκτονική πολλών επιπέδων. Η συγκεκριμένη αρχιτεκτονική έχει κατά κάποιον τρόπο αντικαταστήσει την παραδοσιακή client/server αρχιτεκτονική όπου η συνολική λειτουργικότητα μοιράζεται σε 2 επίπεδα και πλέον 'σπάει' σε 3 ή περισσότερα επίπεδα (n tiers). Από τις πιο γνωστές και επιτυχημένες αρχιτεκτονικές της



κατηγορίας αυτής είναι η Java EE, η οποία κάνει χρήση τεσσάρων επιπέδων. Οι διαδικτυακές εφαρμογές γραμμένες σε PHP χρησιμοποιούν ένα επίπεδο λιγότερο, δηλαδή συνολικά τρία. Αυτό σημαίνει πως τα συστατικά που περιγράψαμε στην προηγούμενη υποενότητα μοιράζονται σε τρία ξεχωριστά επίπεδα τα οποία είναι το Database tier, το Web tier και το Client tier.

- Database tier**
 Το επίπεδο αυτό υλοποιεί τόσο τον μηχανισμό αποθήκευσης των δεδομένων όσο και τον μηχανισμό έλεγχου πρόσβασης σε αυτά κάνοντας χρήση ενός RDBMS που συνήθως είναι ένας MySQL Server.
- Web tier**
 Στο επίπεδο αυτό υλοποιείται η επιχειρησιακή λογική της εφαρμογής αλλά και μεγάλο κομμάτι της παρουσίασης κάνοντας χρήση ενός συνόλου αρχείων που μπορεί να είναι αρχεία PHP, HTML, CSS, JavaScript, εικόνες κ.α. Ένας Web server (συνήθως ένας Apache) σε συνεργασία με τα πακέτα της PHP που βρίσκονται εγκατεστημένα στο συγκεκριμένο επίπεδο αναλαμβάνουν να τροφοδοτήσουν τον web browser του χρήστη με HTML, τη γλώσσα δηλαδή που μπορεί να ερμηνεύσει.
- Client tier**
 Το επίπεδο αυτό βρίσκεται στον υπολογιστή του χρήστη και συγκεκριμένα στον web browser που χρησιμοποιεί για να αλληλοεπιδράσει με την εφαρμογή. Για αυτόν ακριβώς τον λόγο στο σχήμα 1 που ακολουθεί βλέπουμε πως κομμάτι της παρουσίασης βρίσκεται υλοποιημένο στο επίπεδο αυτό. Το τμήμα της παρουσίασης που υλοποιείται στο Client tier είναι ο ίδιος ο web browser, ένα λογισμικό δηλαδή μέσω του οποίου ο χρήστης είναι σε θέση να ερμηνεύσει και να προβάλλει την HTML που του αποστέλλει ο Web server.



Σχήμα 2 : Περιγράφεται η αρχιτεκτονική 3 επιπέδων της PHP και συγκεκριμένα, ο φυσικός διαχωρισμός. Στην περίπτωση αυτή τα 3 διαφορετικά επίπεδα της αρχιτεκτονικής βρίσκονται εγκατεστημένα σε διαφορετικά μηχανήματα.

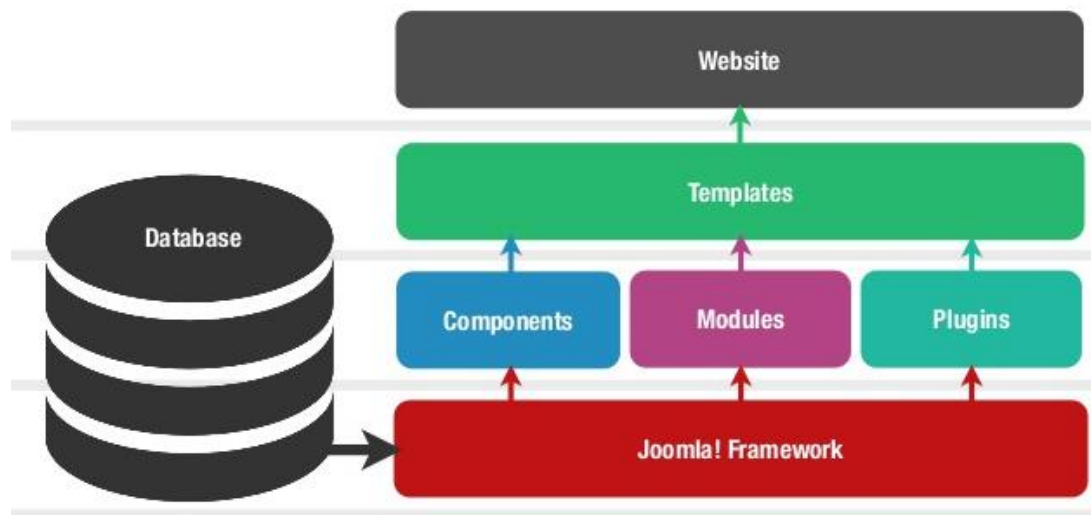
Στον αντίποδα υπάρχει και ο λογικός διαχωρισμός, μία διάταξη όπου τα 3 επίπεδα της αρχιτεκτονικής μπορεί να βρίσκονται εγκατεστημένα στο ίδιο μηχανήμα. Τα 3 αυτά επίπεδα μπορεί να βρίσκονται εγκατεστημένα στο ίδιο μηχανήμα, εξακολουθούν όμως να είναι ξεχωριστά και να έχουν διακριτούς ρόλους.



5.5. ΔΟΜΗ ΤΟΥ JOOMLA

Το Joomla αποτελείται από πολλά και διαφορετικά συστατικά μέρη που επιτελούν συγκεκριμένες λειτουργίες το καθένα. Κάθε συστατικό του Joomla εξαιτίας της διαφορετικής του λειτουργίας αποτελεί από μόνο του ένα αυτόνομο τμήμα.

Εάν κάνουμε μια προεπισκόπηση του συστήματος θα δούμε ότι απαρτίζεται από 3 βασικά επίπεδα που αποτελούν τους κύριους μοχλούς και φαίνονται στην παρακάτω εικόνα.



Σχήμα 3 : Δομή και αρχιτεκτονική του Joomla CMS. Αρχικά βλέπουμε τα core functions, στην συνέχεια τις 3 βασικές δομές (Components, Modules, Plugins) και τέλος το template που αποτελεί την απεικόνιση της εφαρμογής σε HTML (View)

Είναι το κεντρικό, κατασκευαστικό επίπεδο, το επίπεδο των εφαρμογών και το επίπεδο των επεκτάσεων.

Το πρώτο επίπεδο αφορά το επίπεδο της βάσης του Framework. Στο επίπεδο αυτό βρίσκονται τα συστατικά στοιχεία του Joomla, όπως:

- Τις βιβλιοθήκες που απαιτούνται από το πλαίσιο του Joomla ή που εγκαθίστανται από εξωτερικούς προγραμματιστές.
- Το πλαίσιο λειτουργίας του Joomla.
- Τα πρόσθετα που επεκτείνουν τη λειτουργία που είναι διαθέσιμη στο πλαίσιο λειτουργίας του Joomla.

Το δεύτερο είναι το επίπεδο των εφαρμογών που αποτελείται από 3 βασικές εφαρμογές:

- **Εγκατάσταση**
Η εφαρμογή αυτή είναι υπεύθυνη για την εγκατάσταση του Joomla σε ένα web server και τη διαγραφή της όταν ολοκληρωθεί.
- **Διαχείριση**
Η εφαρμογή αυτή είναι υπεύθυνη για τον διαχειριστή που δουλεύει από πίσω και διαχειρίζεται το δικτυακό τόπο και το περιεχόμενό του.
- **Ο δικτυακός τόπος**
Η εφαρμογή αυτή είναι υπεύθυνη για την παρουσίαση του δικτυακού τόπου.
- **Απομακρυσμένη πρόσβαση**
Η εφαρμογή αυτή υποστηρίζει τη διαχείριση του δικτυακού τόπου του Joomla απομακρυσμένα.



Το τρίτο επίπεδο περιέχει όλα τα απαραίτητα εργαλεία που βοηθούν στην τελική προβολή της Web-Based εφαρμογής ώστε να είναι εμφανής σε τρίτους. Πιο συγκεκριμένα στα template και στα τελικά view που θα μετατρέψουν το logic σε HTML.

6. ΚΑΤΑΣΚΕΥΗ JOOMLA COMPONENT

6.1. ΕΙΣΑΓΩΓΗ

Όπως αναλύσαμε και πιο πριν το Joomla CMS είναι ένα πανίσχυρο προγραμματιστικό εργαλείο για την ανάπτυξη web-based Applications που εξελίσσετε από πολλές κοινότητες προγραμματιστών σε πολλούς και διάφορους τομείς με σκοπό την απόκτηση πολλών λειτουργιών που θα κάνουν τις προγραμματιστικές απαιτήσεις πιο απλές και κατανοητές.

Μια από τις βασικές του λειτουργίες είναι η κατασκευή Component τα οποία υπάγονται στην φιλοσοφία του και τα καθιστούν συνεργάσιμα με όλες τις λειτουργίες που μας δίνουν οι βιβλιοθήκες του.

Με την ανάπτυξη ενός μοναδικού component από την αρχή με τις βάσεις που εμείς του ορίσουμε μπορούμε να δημιουργήσουμε ότι ακριβώς θέλουμε και να παρουσιάσουμε κάθε μορφής δεδομένα έχοντας σαν όφελος ότι αναφέραμε πιο πάνω.

Για τις δικές μας απαιτήσεις θα κατασκευάσουμε ένα component με το οποίο θα διαχειριζόμαστε τις απαιτήσεις μιας εφαρμογής που αφορά τους στίχους των τραγουδιών, ενός ή πολλών στιχουργών, ενός ή πολλών τραγουδιστών, ενός ή πολλών συνθετών με όλες τις επιμέρους πληροφορίες που θα μας δοθούν όπως έτος έκδοσης, τύπος τραγουδιού και πολλά άλλα τα οποία θα αναφέρομε παρακάτω σχηματικά αλλά και πιο εκτεταμένα.

6.2. JOOMLA COMPONENT

Component ονομάζουμε το κύριο στοιχείο μιας ιστοσελίδας το οποίο είναι επίσης ένα στοιχείο του μενού. Πιο περιγραφικά όταν επιλέξουμε ένα στοιχείο του μενού το οποίο αναφέρεται στο component θα “φορτωθεί” μια σελίδα η οποία εμπεριέχει το component που ορίζεται από το στοιχείο του μενού.

Να σημειωθεί ότι επειδή η πλατφόρμα Joomla είναι δομημένη με το πρότυπο MVC όπως αναφέραμε και πιο πάνω οπότε αν ρίξουμε μια ματιά και στην δομή των αρχείων του μπορούμε να κατανοήσουμε ότι τα κυρίως συστατικά και λειτουργίες του με μια απλή ματιά.

Με έναν έλεγχο στους υποφακέλους των components θα δούμε καθένα από τα βασικά στοιχεία της Joomla: com_banners, com_contact και για άρθρα com_content, com_finder, com_mailto και com_media, com_newfeeds, com_search, com_users, com_weblinks και com_wrappers. Όλοι οι φάκελοι των component ξεκινάνε με το com_ και έπειτα μπαίνει το όνομα του component. Αυτό είναι ένα παράδειγμα της ισχυρής σύμβασης ονομασίας των φακέλων που χρησιμοποιούνται στο Joomla.

Οι περισσότεροι από αυτούς τους τύπους των στοιχείων του μενού αντιστοιχούν ακριβώς σε ένα από τους φακέλους Joomla. Αυτοί οι φάκελοι περιλαμβάνουν τις επαφές, τα άρθρα (χρησιμοποιώντας το φάκελο com_content), news feeds, αναζήτηση, χρήστες, σύνδεση κλπ.



Name	Date modified	Type
administrator	10/28/2011 7:58 AM	File Folder
cache	10/28/2011 7:58 AM	File Folder
cli	10/28/2011 7:58 AM	File Folder
components	10/28/2011 7:58 AM	File Folder
images	10/28/2011 7:58 AM	File Folder
includes	10/28/2011 7:58 AM	File Folder
installation	10/28/2011 7:58 AM	File Folder
language	10/28/2011 7:58 AM	File Folder
libraries	10/28/2011 7:58 AM	File Folder
logs	10/28/2011 7:58 AM	File Folder
media	10/28/2011 7:58 AM	File Folder
modules	10/28/2011 7:58 AM	File Folder
plugins	10/28/2011 7:58 AM	File Folder
templates	10/28/2011 7:58 AM	File Folder
tmp	10/28/2011 7:58 AM	File Folder
htaccess.txt	4/7/2011 8:47 AM	Text Document
index.php	2/21/2011 11:45 AM	PHP File
joomla.xml	10/17/2011 7:59 AM	XML File
LICENSE.txt	12/12/2009 7:44 AM	Text Document
README.txt	9/25/2011 12:00 PM	Text Document
robots.txt	9/20/2011 6:37 AM	Text Document
web.config.txt	4/7/2011 8:30 AM	Text Document

Εικόνα 1: Πιο αναλυτικά οι δομή των φακέλων με τα αρχεία που δίνουν ζωή στην πλατφόρμα του Joomla.

Πιο γενικά και αφαιρετικά μπορούμε να αναφέρουμε πως το component Lyrics μπορεί να παρομοιαστεί με μια μεγάλη βιβλιοθήκη η οποία κατηγοριοποιεί τα τραγούδια με τα υπόλοιπα χαρακτηριστικά τους τα οποία είναι τα εξής:

- Συνθέτης
- Τραγουδοποιός
- Τραγουδιστής
- Εκτέλεση του τραγουδιού
- Χρονολογία
- Άλμπουμ

Κάθε ένα τραγούδι μπορεί να συνδέεται με πάνω από τραγουδοποιούς, συνθέτες, τραγουδιστές, εκτελεστές, άλμπουμ καθώς και χρονολογίες.

6.3. ΑΝΑΛΥΣΗ ΔΟΜΗΣ COM_LYRICS

Σε αυτό το κομμάτι θα περιγράψουμε τις λειτουργίες που χρειαζόμαστε για να κατασκευάσουμε το νέο Joomla Component που θα κατασκευάσουμε.

Τραγούδια

Στο τμήμα αυτό υπάρχει μια λίστα με όλα τα τραγούδια που υπάρχουν στο component Lyrics. Στην ουσία το κάθε τραγούδι είναι ένα ξεχωριστό πεδίο της λίστας το οποίο συνοδεύεται από τα χαρακτηριστικά του τα οποία είναι: οι στίχοι, η φωτογραφία του άλμπουμ στο οποίο ανήκει το τραγούδι, ο τραγουδοποιός, ο συνθέτης, το άλμπουμ στο οποίο ανήκει και η χρονολογία του.



Όπως είναι εμφανές στην περίπτωση που έχουμε ένα τραγούδι από ένα δεύτερο τραγουδιστή ή σε ένα δεύτερο άλμπουμ θα δημιουργήσουμε ένα έξτρα πεδίο στη λίστα όπου θα υπάρχει πάλι ο ίδιος τίτλος τραγουδιού αλλά θα συνδέεται με διαφορετικό τραγουδιστή, συνθέτη ή άλμπουμ, αναλόγως ποιο ισχύει την κάθε φορά.

Στιχουργοί

Στο πεδίο αυτό υπάρχουν οι τραγουδοποιοί. Σε αντίθετη περίπτωση όπου θα υπήρχε πλήθος τραγουδοποιών θα είχαμε χρήση μιας λίστας με τα ονόματα τους.

Έτσι σε κάθε τραγούδι θα μπορούσαμε να επιλέξουμε το όνομα του τραγουδοποιού, αυτή η τεχνική ακολουθείται στο συνθέτη, στους τραγουδιστές κτλ.

Συνθέτες

Σε αυτό το τμήμα βρίσκονται οι συνθέτες των τραγουδιών. Υπάρχει μια λίστα με τα ονόματα των συνθετών. Εδώ είναι σημαντικό να τονίσουμε πώς το κάθε τραγούδι μπορεί να έχει πάνω από έναν συνθέτη, επίσης ο κάθε συνθέτης συνδέεται με πάνω από ένα τραγούδια τα οποία έχει συνθέσει.

Εδώ βλέπουμε μια σύνδεση της λογικής “ένα προς πολλά και πολλά προς ένα”. Αυτό θα μας βοηθήσει να κατανοήσουμε πιο εύκολα τη λογική της σύνδεσης.

Τραγουδιστές

Σε αυτό το τμήμα βρίσκονται οι τραγουδιστές του κάθε τραγουδιού. Υπάρχει μια λίστα με τα ονόματα των τραγουδιστών. Θα πρέπει να τονίσουμε πώς το κάθε τραγούδι μπορεί να έχει πάνω από έναν τραγουδιστή, επίσης ο κάθε τραγουδιστής συνδέεται με πάνω από ένα τραγούδια τα οποία έχει τραγουδήσει.

Εκτελέσεις Τραγουδιών

Σε αυτό το τμήμα βρίσκονται οι εκτελέσεις των τραγουδιών. Το κάθε τραγούδι έχει εκτελεστεί από πλήθος καλλιτεχνών. Στην ουσία στο τμήμα αυτό συνδέεται το τραγούδι με τον τραγουδιστή.

Πιο αναλυτικά το κάθε τραγούδι έχει πολλές εκτελέσεις οι οποίες αριθμούνται, επίσης επιλέγουμε την χρονολογία εκτέλεσης, το όνομα του τραγουδιστή, το βίντεο καθώς και το είδος του εκτελεστή (εάν είναι βασικός ή εάν εκτελεί τα φωνητικά)

Άλμπουμ (Δίσκοι)

Στο τμήμα αυτό βρίσκονται τα άλμπουμ των τραγουδιών. Το κάθε τραγούδι είναι τμήμα ενός άλμπουμ ή μπορεί να είναι τμήμα περισσότερων από ένα άλμπουμ.

Πρακτικά στο τμήμα των τραγουσιών όπου και εμπλουτίζουμε το κάθε τραγούδι με τα χαρακτηριστικά του, συμπληρώνουμε και το άλμπουμ στο οποίο περιλαμβάνεται. Εάν αυτό το τραγούδι περιέχεται και σε δεύτερο ή τρίτο άλμπουμ τότε θα πρέπει να δημιουργηθεί δεύτερο ή αντίστοιχα και τρίτο στοιχείο για το συγκεκριμένο τραγούδι.

Χρονολογίες

Στο τμήμα αυτό εισάγουμε τις χρονολογίες του κάθε τραγουδιού και του κάθε άλμπουμ. Υπάρχει μια λίστα με όλες τις χρονολογίες και στη συνέχεια τις συνδέουμε με το τραγούδι ή την εκτέλεση που μας ενδιαφέρει.

Να σημειωθεί ότι αυτό το τμήμα θα έχει λογική στην σωστή κατηγοριοποίηση των τραγουδιών που θα μας βοηθήσει μετέπειτα στην αναζήτηση και στο φιλτράρισμα των τραγουδιών και είναι ο βασικός λόγος που το πεδίο αυτό έχει γίνει δυναμικό.

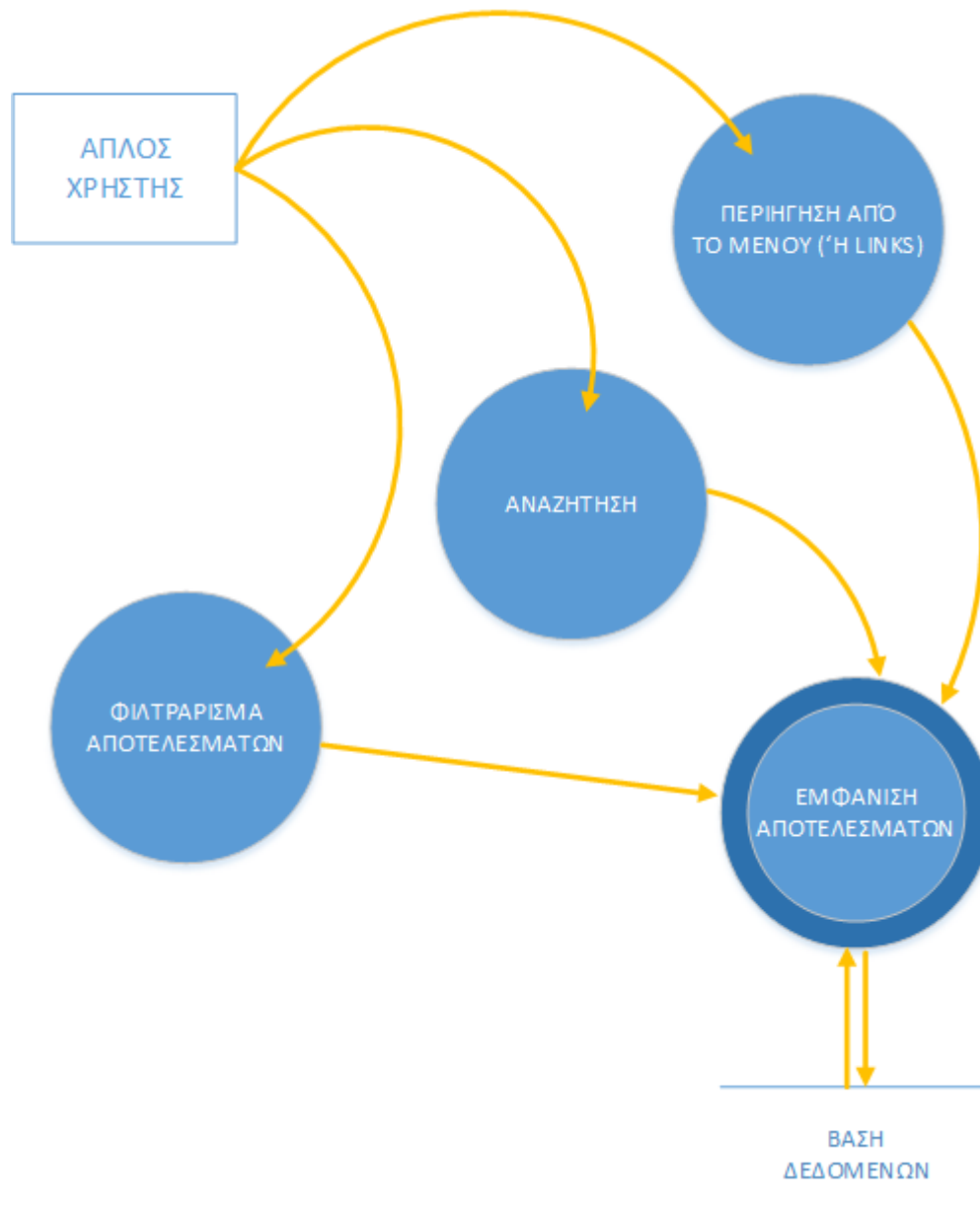
Αναζήτηση

Στο σημείο αυτό ο χρήστης μπορεί να αναζητήσει οποιοδήποτε τραγούδι επιθυμεί με χρήση μιας λέξης κλειδί από τους στίχους ή τον τίτλο. Αυτό είναι πολύ βοηθητικό αν σκεφτείτε απλώς πως στο component μπορεί συμπεριλαμβάνονται εκατοντάδες τραγούδια. (Στην υλοποίηση μας υπάρχουν πάνω από 500 πραγματικά τραγούδια με τα σωστά τους στοιχεία) .



6.4. ΡΟΗ ΔΕΔΟΜΕΝΩΝ

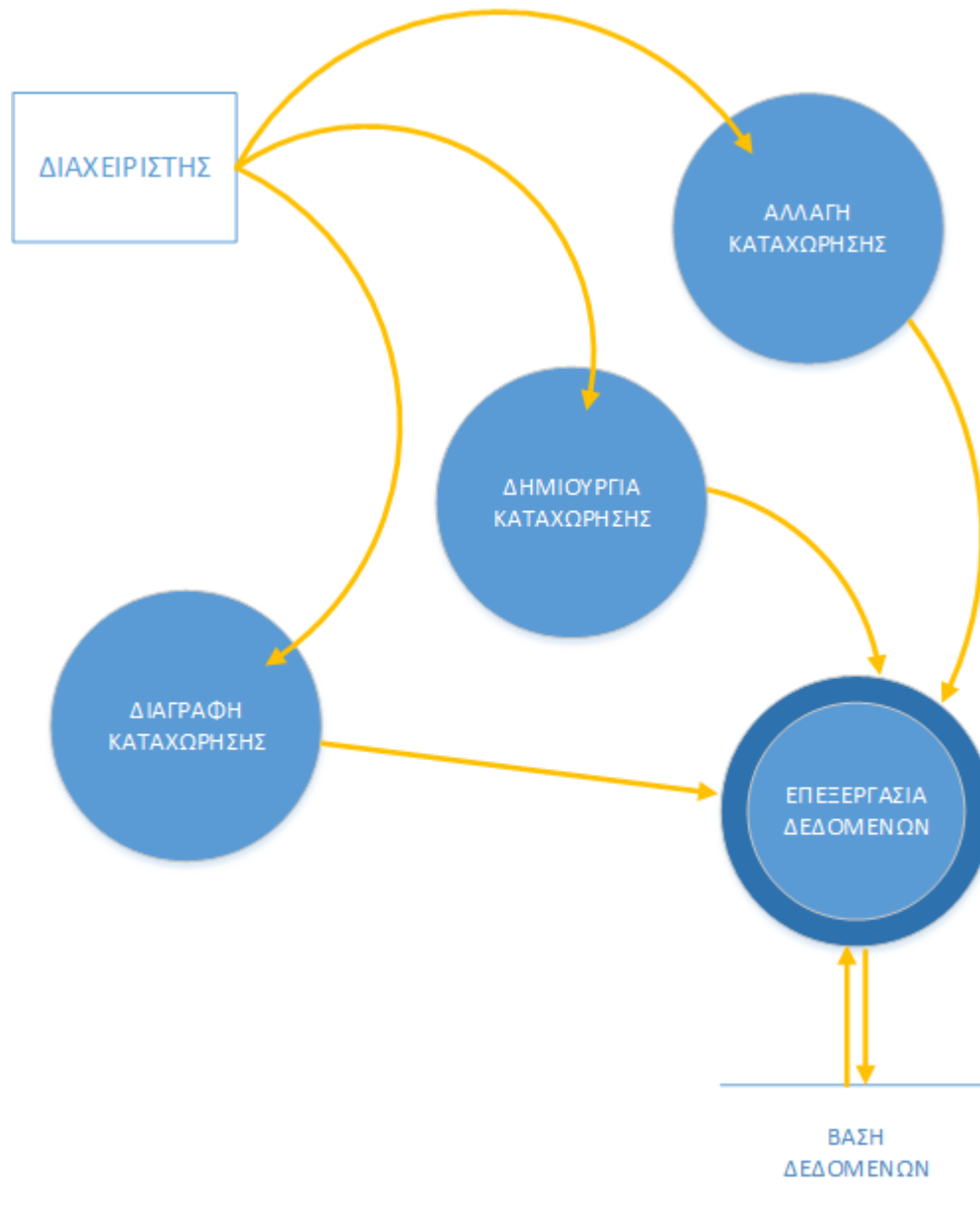
Αρχικά, παραθέτουμε το αρχικό διάγραμμα ροής στο οποίο και φαίνονται οι εργασίες που έχει τη δυνατότητα να εκτελέσει ο απλός χρήστης (User).



Σχέδιο 4 : Διάγραμμα ροής δεδομένων απλού χρήστη



Στη συνέχεια παρουσιάζονται τα Διαγράμματα Ροής δεδομένων για το Διαχειριστή (Administrator) με όλες τις δραστηριότητες που μπορεί να εκτελέσει.



Σχέδιο 5 : Διάγραμμα ροής δεδομένων διαχειριστή



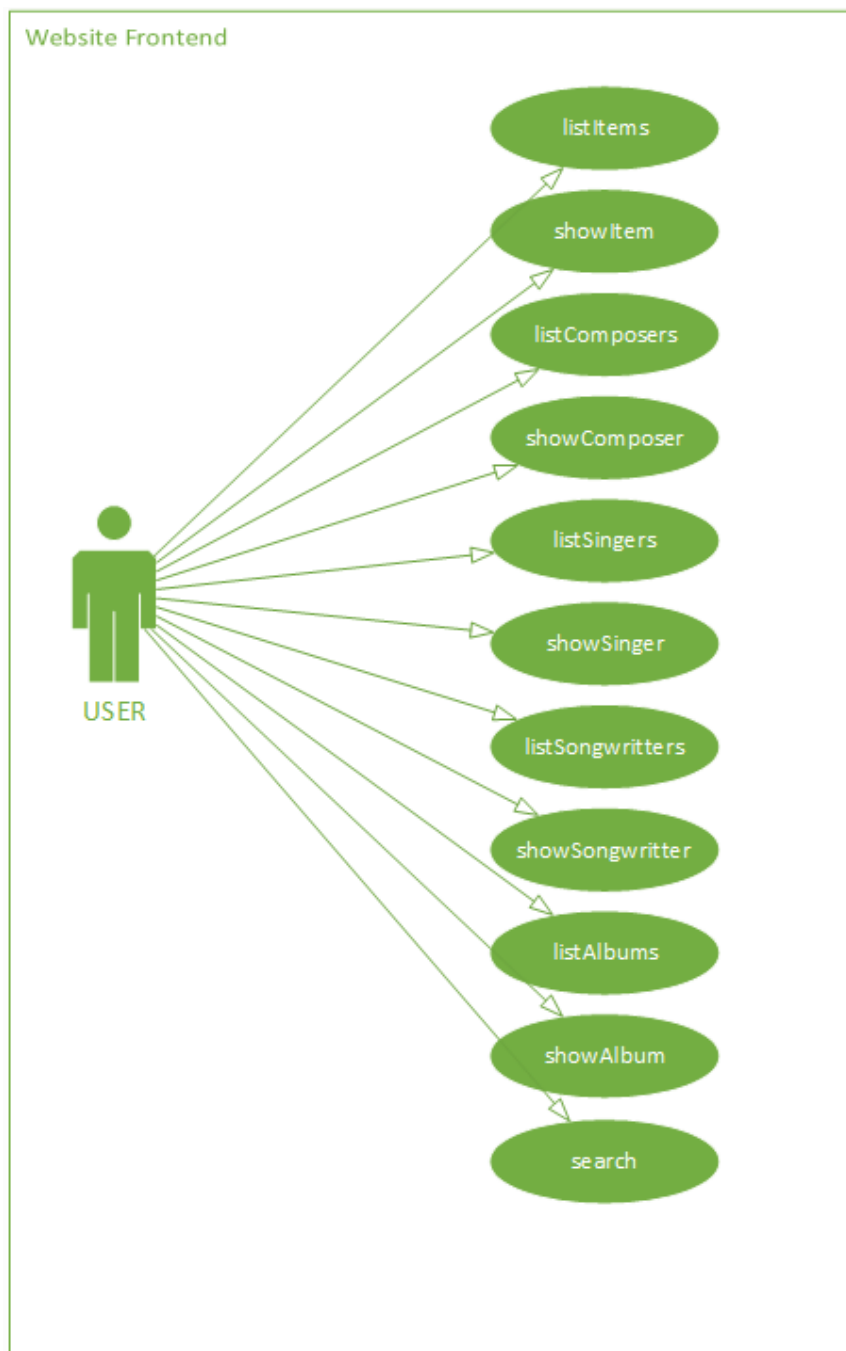
6.5. ΠΕΡΙΠΤΩΣΕΙΣ ΧΡΗΣΗΣ

Για την καλύτερα κατανόηση των παραπάνω απαιτήσεων παραθέτουμε τα διαγράμματα περιπτώσεων χρήσης για τους χρήστες του com_lyrics.

Στο σύστημα μας θα υπάρχουν 2 χρήστες που μπορεί να αλληλεπιδράσουν:

1. Απλός Χρήστης (user)

Ο απλός χρήστης έχει την δυνατότητα πρόσβασης μόνο στο public interface της ιστοσελίδας (Frontend) μέσα από το οποίο μπορεί να χρησιμοποιήσει τις παρακάτω δυνατότητες.

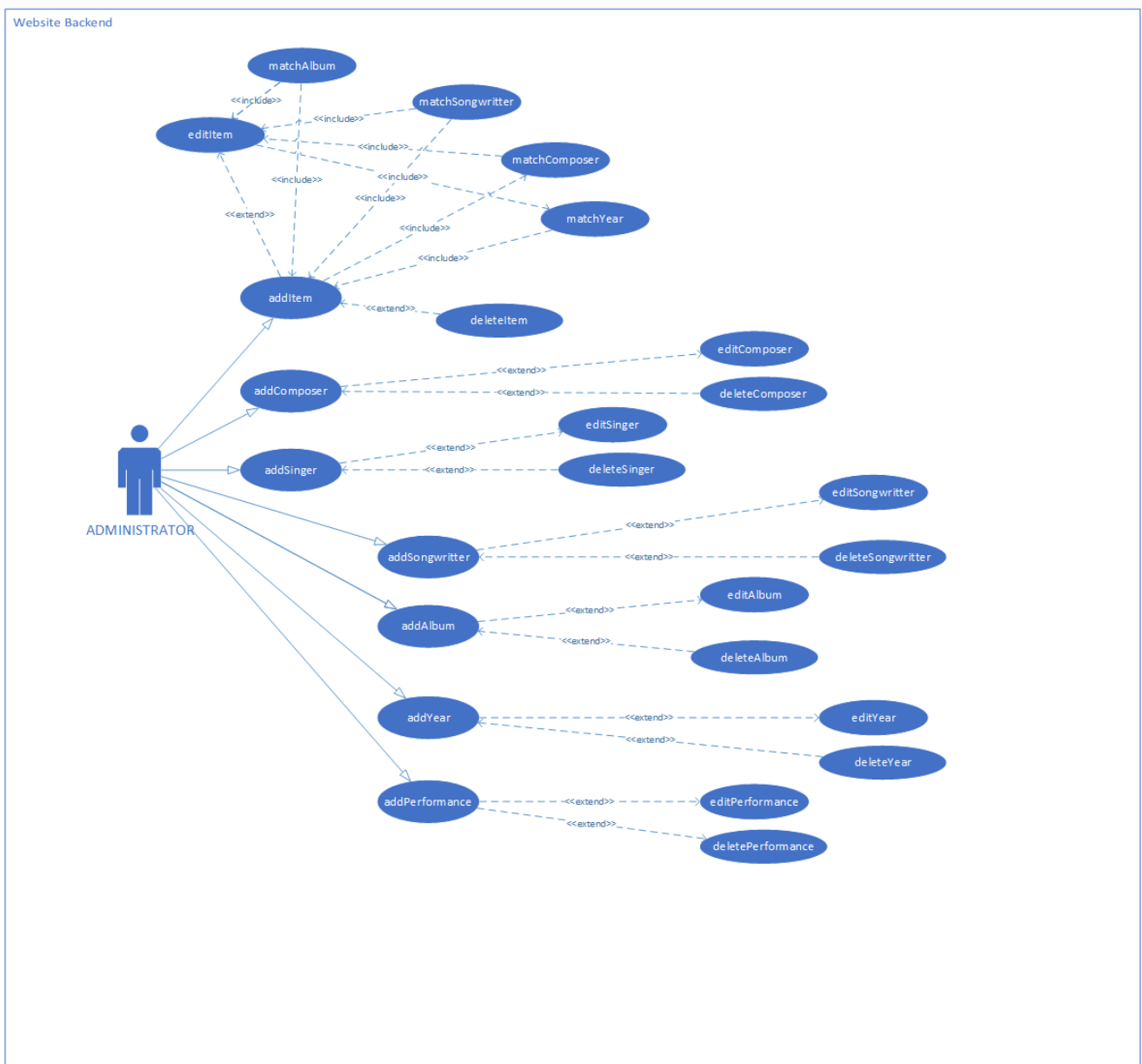


Σχέδιο 6 : Διάγραμμα περίπτωσης απλού χρήστη



2. Διαχειριστής (administrator)

Ο διαχειριστής έχει την δυνατότητα πρόσβασης αρχικά στο public interface σαν απλός χρήστης και κατ' επέκταση στο private interface του συστήματος στο οποίο μπορεί να παραμετροποιήσει τα στοιχεία του com_lyrics.



Σχέδιο 7 : Διάγραμμα περίπτωσης διαχειριστή



6.6. ΕΙΚΟΝΕΣ ΤΡΑΓΟΥΔΙΩΝ

ΑΝΑΓΚΕΣ

Στις απαιτήσεις του στοιχείου που δημιουργήσαμε είχαμε την ανάγκη ύπαρξης φωτογραφιών για το κάθε τραγούδι, μια σχετικά μικρή φωτογραφία που θα απεικόνιζε κάποια πληροφορία του τραγουδιού, συνήθως το εξώφυλλο του άλμπουμ που φιλοξένησε το τραγούδι.

Η πιο απλή λύση θα ήταν η να αποθηκεύσουμε την διαδρομή του αρχείου της εικόνας στην βάση δεδομένων μας και όταν θα χρειαζόμασταν την εικόνα με ένα απλό ερώτημα στην βάση θα μπορούσαμε να έχουμε το δεδομένο μας. Γιατί όμως να υπερφορτώνουμε την βάση μας με περιττά ερωτήματα ενώ θα μπορούσαμε να το επιτύχουμε με μια απλή εντολή PHP.

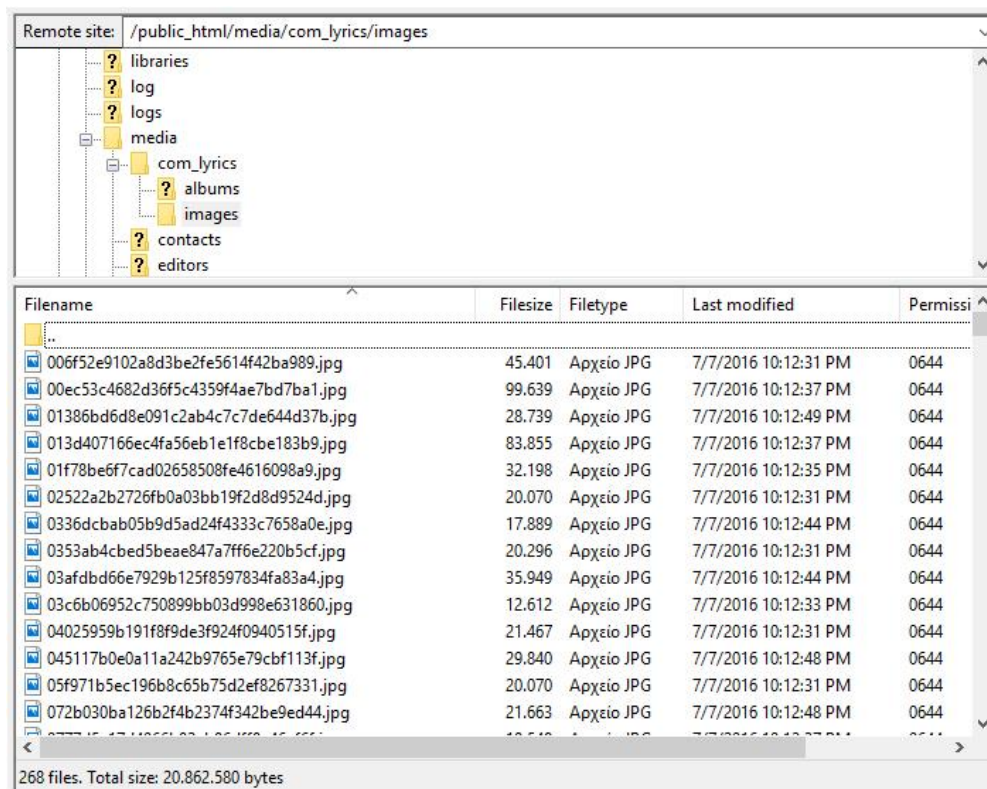
Για να αποθηκεύσουμε τις εικόνες μας, λοιπόν, αφού ξέραμε ότι το κάθε τραγούδι θα έχει μία και μόνο φωτογραφία δημιουργήσαμε ένα φάκελο στα media του Joomla, δηλαδή που το Joomla αποθηκεύει κάθε αρχείο Media για οποιοδήποτε λειτουργία του, έναν υποφάκελο com_lyrics όπου κατ' επέκταση αυτό έχει τους υποφάκελους αναλόγως του view όπου και χρησιμοποιούμε (item, album, composer, singer κλπ.) και η εξάρτηση τους με το κάθε item γίνεται με το id του. Δηλαδή η εικόνα του τραγουδιού με id = 1 βρίσκεται στον φάκελο /public_html/media/com_lyrics/item με όνομα 1.jpg , αντίστοιχα η εικόνα του album με id = 53 θα βρίσκεται στον φάκελο /public_html/media/com_lyrics/album με όνομα 53.jpg .

ΑΣΦΑΛΕΙΑ

Ωστόσο επειδή αυτό θα πρόδιδε πολύ την δομή και θα ήταν πολύ εύκολο για κάποιον να προβεί σε κακόβουλες ενέργειες απέναντι στο component μας χρησιμοποιήσαμε την συνάρτηση md5 της php και «κρύβουμε» το όνομα του αρχείου με την δομή

Filename = md5(Image<id>)

Με αυτόν τον τρόπο όταν θέλουμε να καλέσουμε την εικόνα του αντίστοιχου στοιχείου απλά ζητάμε μέσα στον Controller μας την εικόνα με το παραπάνω όνομα μαζί με έναν λογικό έλεγχο αν υπάρχει το αρχείο, αν αυτό δεν είναι διαθέσιμο τότε εμφανίζουμε μια γενική εικόνα ότι δεν υπάρχει διαθέσιμη εικόνα.



Εικόνα 2 : Δομή αρχείων εικόνας για το αντίστοιχο στοιχείο του com_lyrics

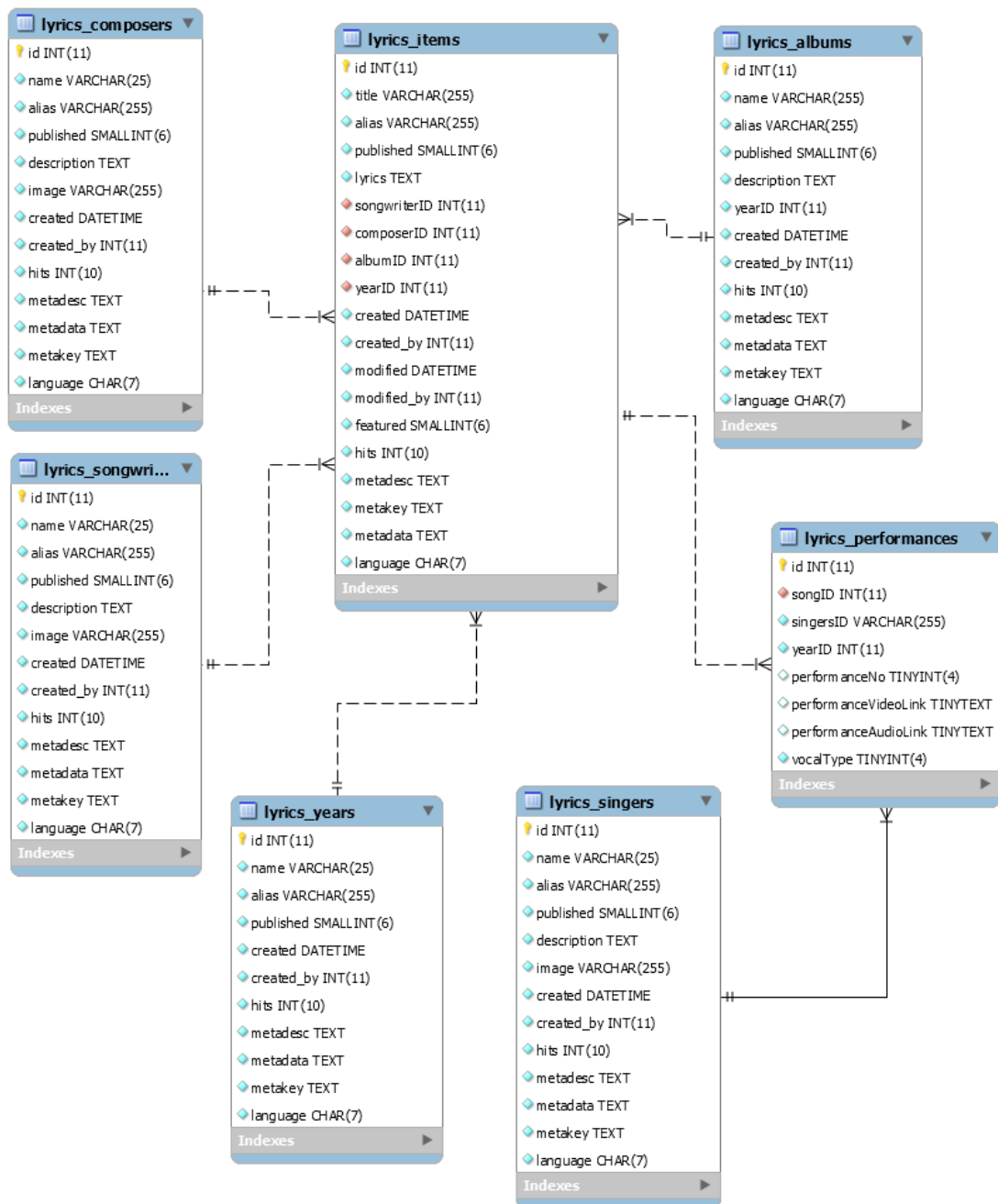


NO IMAGE
FOUND

Εικόνα 3 : Εικόνα μη διαθέσιμου αρχείου εικόνας για το αντίστοιχο στοιχείο του com_lyrics

6.7. ΔΙΑΓΡΑΜΜΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

Στο παρακάτω σχήμα μπορούμε να δούμε το διάγραμμα της βάσης δεδομένων που χρησιμοποιήθηκε για την κατασκευή του Joomla Component. Σε αυτούς τους πίνακες έχουν αποθηκευτεί όλα τα δεδομένα που εμφανίζονται στην ιστοσελίδα καθώς και οποιαδήποτε άλλη πληροφορία θα χρειαστεί στο μέλλον για την σωστή του λειτουργία.



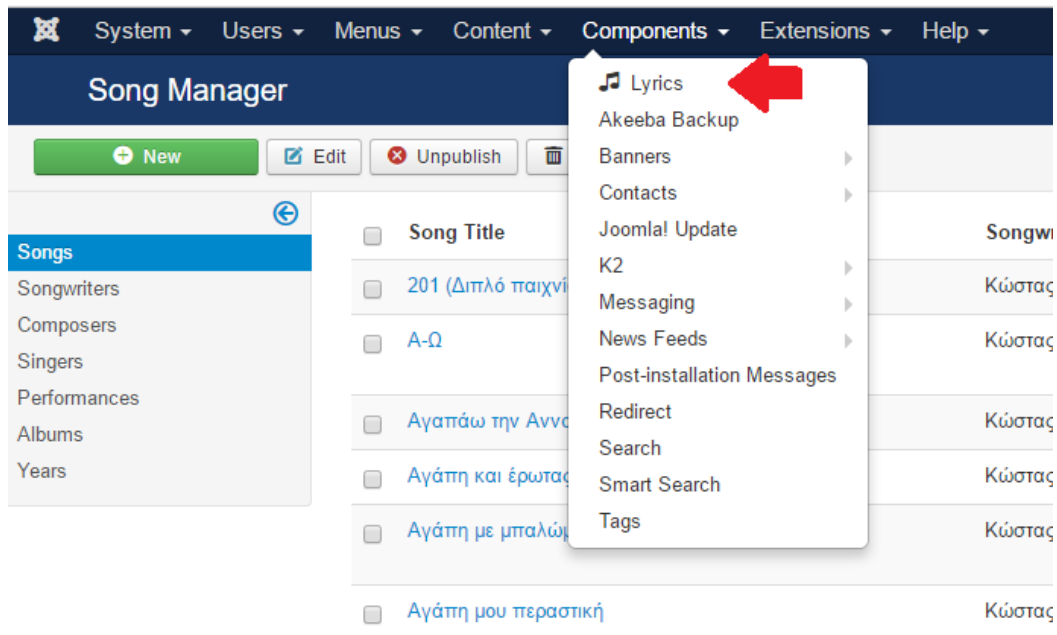
Σχήμα 8 : Δομή βάσης δεδομένων του `com_lyrics`



7. ΟΔΗΓΙΕΣ ΧΡΗΣΗΣ

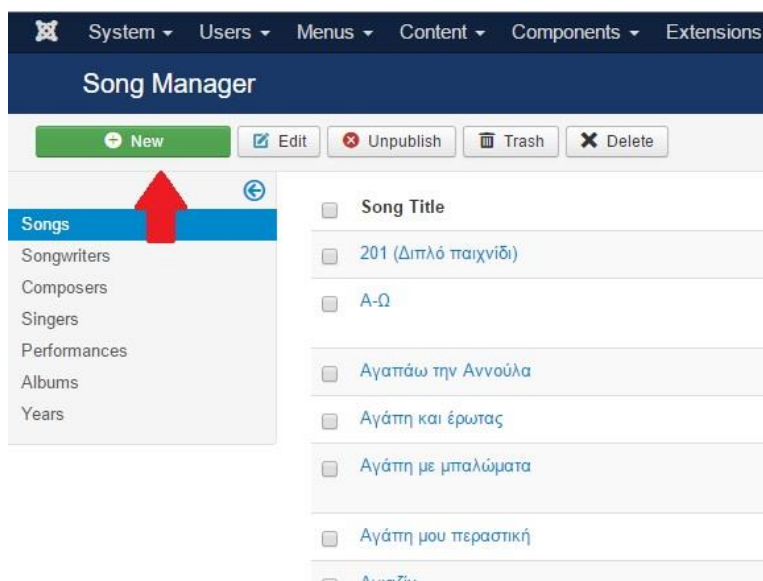
7.1. Δημιουργία νέου τραγουδιού

1. Κάνουμε login στην διαχειριστική περιοχή (<http://mysite.com/administrator>) με τους κωδικούς που μας έχουν δοθεί.
2. Από το top menu -> Components -> Lyrics για να έχουμε πρόσβαση στο Component



Εικόνα 7.1 : Top Menu

3. Πατάμε στο “New” πάνω αριστερά



Εικόνα 7.2 : Νέο Τραγούδι



Συμπληρώνουμε τα απαραίτητα στοιχεία, τα οποία είναι:

- Song Title: ο τίτλος του τραγουδιού
- Lyrics: οι στίχοι του τραγουδιού

Εικόνα 7.3 : Εισαγωγή δεδομένων τραγουδιού

Στην καρτέλα Song Details συμπληρώνουμε:

- Songwriter: το όνομα του συνθέτη από μια λίστα
- Composer: το όνομα του στιχουργού από μια λίστα
- Album: το όνομα του album
- Year: η χρονολογία του άλμπουμ από μια λίστα
- Από το «επιλογή αρχείου» πάνω αριστερά, επιλέγουμε τη φωτογραφία που θέλουμε.



System Users Menus Content Components Extensions Help

Edit Song

Save Save & Close Save & New Close

Song Title * Αγάπη και έρωτας Alias agapi-kai-erotas

Lyrics Song Details Publishing

Songwriter Κώστας Βίρβος ← ↓ Επιλογή αρχείου Δεν επιλέχθηκε κανένα αρχείο.

Quick Add Songwriter No Image Available

Composer Χάρης Καλούδης ←

Quick Add Composer

Album -----

Quick Add Album

Εικόνα 7.4 : Εισαγωγή δεδομένων τραγουδιού 2

System Users Menus Content Components Extensions Help

Edit Song

Save Save & Close Save & New Close

Song Title * Αγάπη και έρωτας Alias agapi-kai-erotas

Lyrics Song Details Publishing

Created Date 2015-09-05 08:03:21 📅 Meta Description

Created By Super User 👤 Meta Keywords

Modified Date 2015-09-05 08:04:42 Robots Use Global

Modified By Super User Author

Hits 14 Content Rights

Εικόνα 7.5 : Εισαγωγή δεδομένων τραγουδιού 3

Έτσι δημιουργείται ένα τραγούδι.



Η παρουσίαση του θα είναι σε αυτή τη μορφή.

Αριστερά βλέπουμε το εξώφυλλο του δίσκου, δεξιά πάνω τον τίτλο, από κάτω τις εκτελέσεις και πιο κάτω τους στίχους.

Κώστας Βίρβος

Αναζήτηση...

ΑΡΧΙΚΗ ΔΙΣΚΟΓΡΑΦΙΑ ΣΥΝΕΝΤΕΥΞΕΙΣ ΑΦΙΕΡΩΜΑΤΑ ΧΡΟΝΙΚΟ ΕΠΙΚΟΙΝΩΝΙΑ

Νανούρισμα (Κοιμήσου αγγελούδι μου) (1962)

Στιχουργός **Κώστας Βίρβος**
 Συνθέτης **Μίκης Θεοδωράκης**

Εκτελέσεις

1	Δέσποινα Μπεμπεδέλη (1962)	Video
2	Γιάννης Πάριος (1985)	Video
3	Χορωδία Τερψιχόρης Παπαστ (1961)	Video
4	Σαββίνα Πανάντου (NoYear)	
5	Γιώτα Λύδια (1955)	Video
6	Γιάννης Μπέζος (2002)	Video
7	Κερασσία Σαμαρά (2006)	Video
8	Γλυκερία, Μίκης Θεοδωράκης (NoYear)	Video
9	Ελένη Βουδουράκη, Μίκης Θεοδωράκης (2013)	Video

[f Share](#)
[t Tweet](#)
[g+1](#)

ΕΠΙΛΟΓΕΣ

ΕΚΤΥΠΩΣΙΜΗ ΜΟΡΦΗ

ΑΠΟΣΤΟΛΗ ΜΕ EMAIL

ΑΝΑΖΗΤΗΣΗ

Επιλέξτε Συνθέτης

Επιλέξτε Χρονολογία

Κοιμήσου αγγελούδι μου, παιδί μου νάνι νάνι
 Να μεγαλώσεις γρήγορα, σαν τ' αψηλό πλατάνι
 Να γίνεις άντρας στο κορμί και στο μυαλό
 Και να `σαι πάντα μεσ' το δρόμο τον καλό

Κοιμήσου αγγελούδι μου γλυκά με το τραγούδι μου

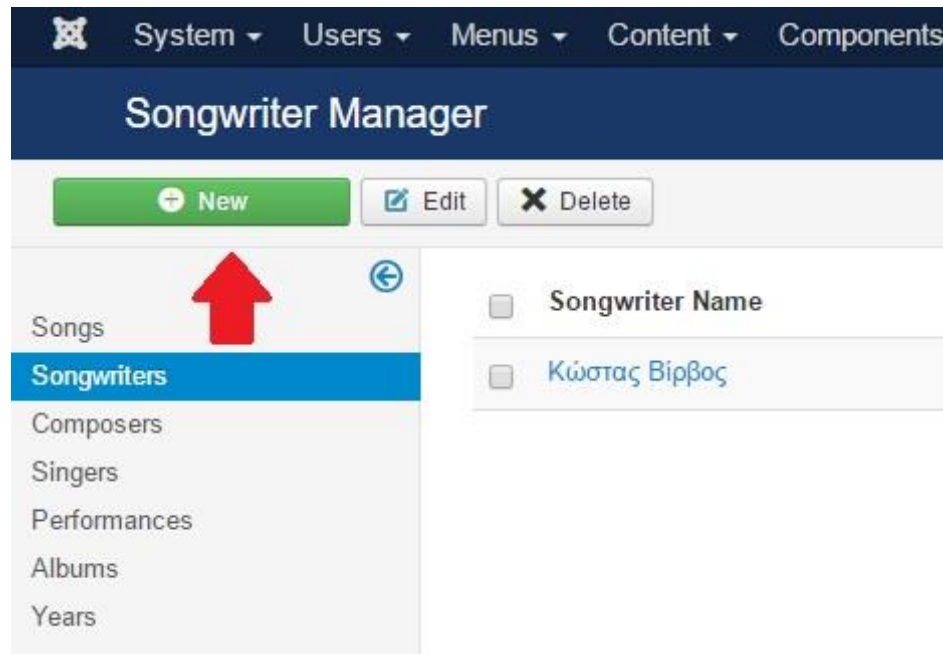
Κοιμήσου περιστέρι μου να γίνεις σαν ασάλι
 Να γίνει κι η καρδούλα σου σαν του Χριστού μεγάλη
 Για να μην πεις μεσ' τη ζωή σου δεν μπορώ
 κι αν πρέπει ακόμα να σηκώσεις και σταυρό

Εικόνα 7.6 : Παρουσίαση δεδομένων τραγουδιού στο Front-End



7.2. Δημιουργία νέου Στιχουργού

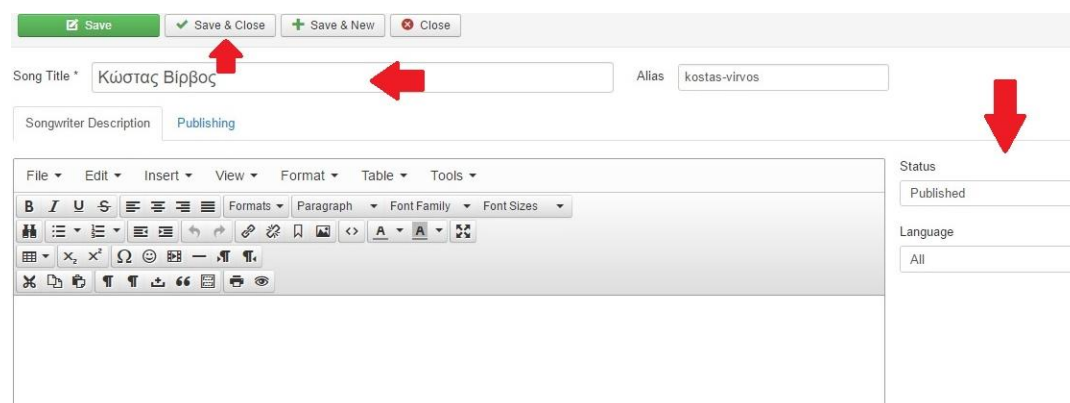
Επιλέγουμε “New” πάνω αριστερά αφού πρώτα πάμε στο Songwriters από τη λίστα



Εικόνα 7.7 : Νέος Στιχουργός

Στη συνέχεια συμπληρώνουμε τα απαραίτητα στοιχεία.

- Title, Description (αν είναι απαραίτητο) και Status.
- Τέλος πατάμε Save&Close



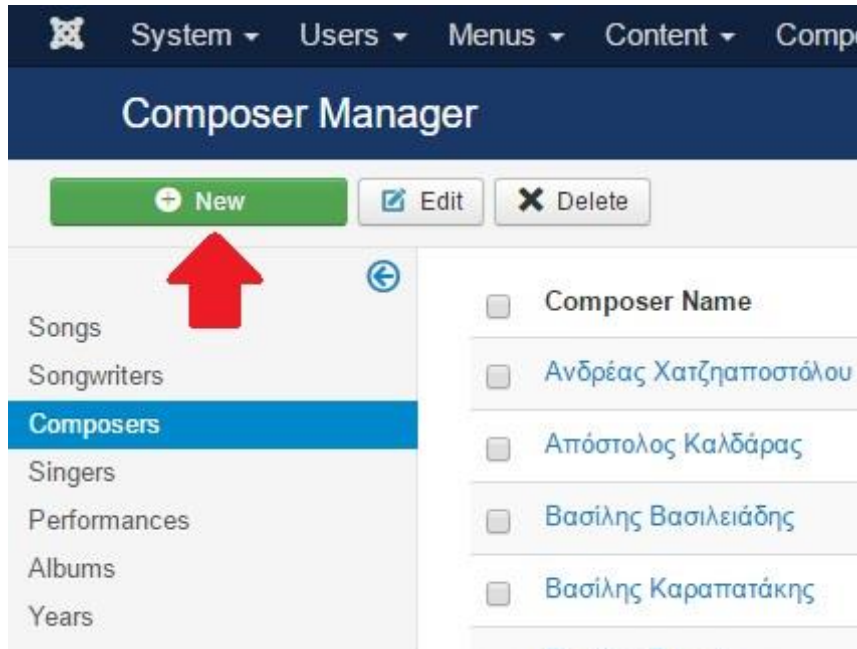
Εικόνα 7.8: Εισαγωγή δεδομένων στιχουργού

Έτσι δημιουργούμε ένα Στιχουργό.



7.3. Δημιουργία νέου Συνθέτη

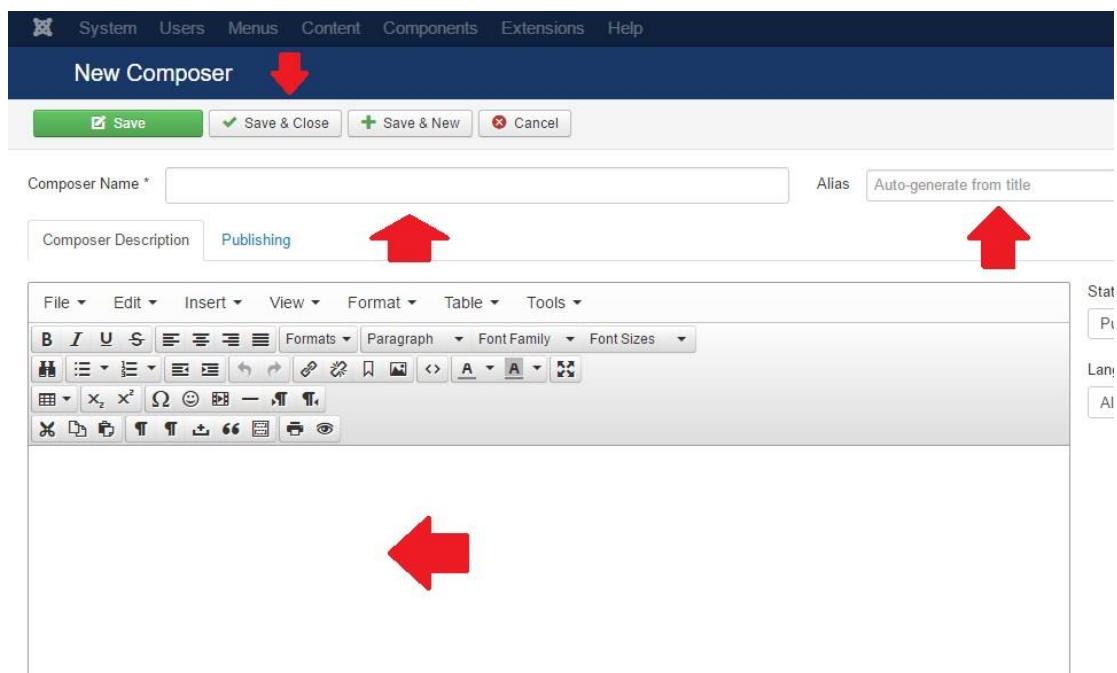
Επιλέγουμε “New” πάνω αριστερά αφού πρώτα επιλέξουμε Composer από τη λίστα.



Εικόνα 7.9 : Νέος Συνθέτης

Στη συνέχεια συμπληρώνουμε τα παρακάτω στοιχεία:

- Composer name: όνομα συνθέτη
- Description: εάν θέλουμε να προσθέσουμε μια περιγραφή

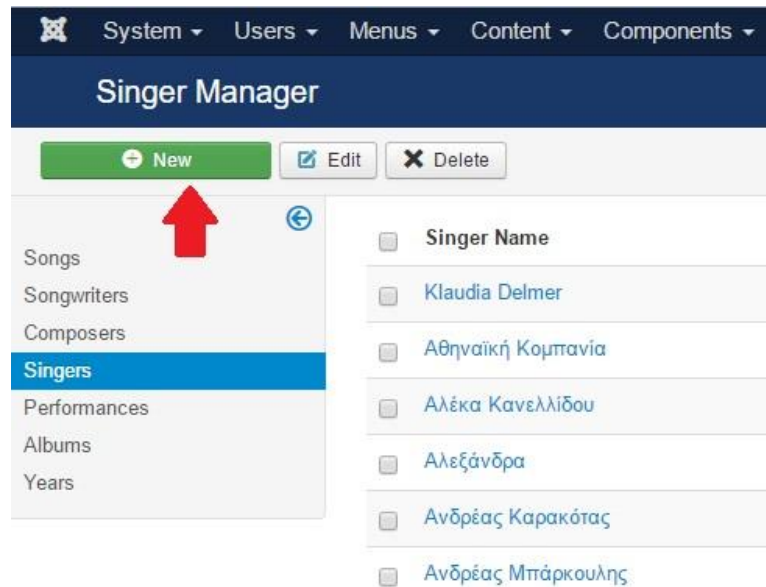


Εικόνα 7.10: Εισαγωγή δεδομένων συνθέτη



7.4. Δημιουργία νέου Τραγουδιστή

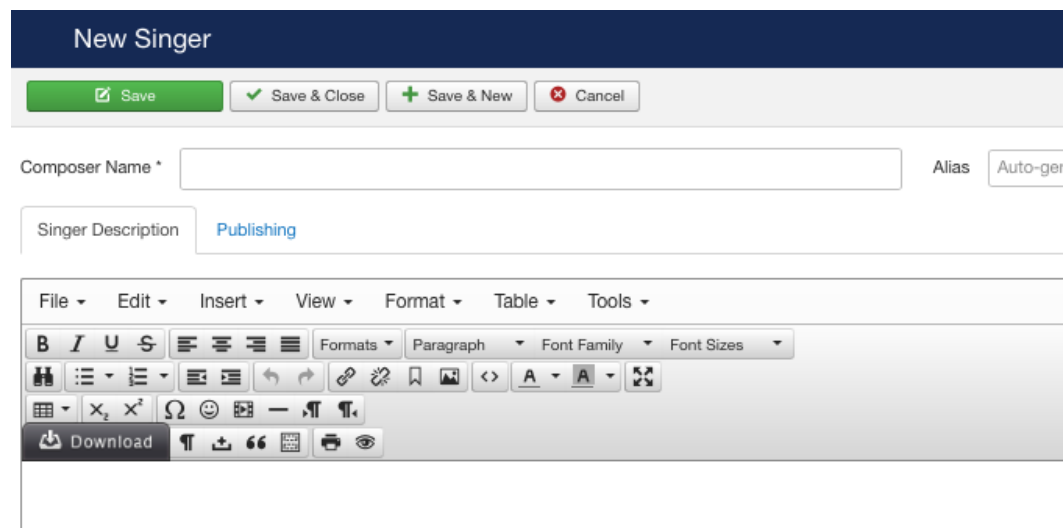
Επιλέγουμε “New” πάνω αριστερά αφού πρώτα επιλέξουμε Singer από τη λίστα.



Εικόνα 7.11 : Νέος Τραγουδιστής

Συμπληρώνουμε τα παρακάτω:

- Singer name
- Description (όπου βάζουμε την βιογραφία του)



Εικόνα 7.12 : Εισαγωγή δεδομένων Τραγουδιστής



Η παρουσίαση του θα είναι σε αυτή τη μορφή.

Πάνω βλέπουμε το όνομα του συνθέτη, ακριβώς από κάτω βλέπουμε το βιογραφικό του και τέλος βλέπουμε τα τραγούδια που έχει συνθέσει

The screenshot shows the website 'Κώστας Βίρβος' with a search bar and navigation menu. The main content area displays the biography of Giannis Kalatzis, followed by a table of his compositions.

Γιάννης Καλατζής

Ο Γιάννης Καλατζής γεννήθηκε τον Απρίλη του 1943 στη Θεσσαλονίκη, όπου μαζί με δυο φίλους του, στις αρχές της δεκαετίας του '60, έφτιαξαν το Τρίο Μορένο. Με την προτροπή του Τώνη Μαρούδα, το 1965 καταβαίνουν στην Αθήνα. Λίγο μετά το Τρίο διαλύεται, αλλά σε μια γιορτή κρασιού στο Δασφί, το 1967, ο Γιώργος Μητσάκης τον ακούει να τραγουδά και του προτείνει συνεργασία. Ξεκινούν μαζί στην Πλάκα και λίγο μετά του δίνει τα πρώτα του τραγούδια σε δίσκους 45 στροφών στην Odeon - Parlophone, μετέπειτα Μίνος, εταιρεία που θα στηρίξει με τεράστιες επιτυχίες και θα υπηρετήσει πιστά μέχρι το 1985... Μετά τον Μητσάκη, θα ακολουθήσουν συνεργασίες με τους λαϊκούς συνθέτες Απόστολο Καλδώρα, Μπάμνη Μπακάλη, Κώστα Καπλάνη και με το Θόδωρο Δερβενιώτη που του δίνει "Τα χρυσά κλειδιά" σε στίχους Κώστα Βίρβου. Από το 1968-69 και μετά και μέχρι το 1976-77 συνεργάζεται δισκογραφικά με το Μάνο Λοΐζο και το Λευτέρη Παπαδόπουλο, το Γιώργο Κατσαρό και τον Πυθαγόρα, το Σταύρο Κουγιουμτζή, με τραγούδια κυρίως σε στίχους της Σώτιας Τσώτου αλλά και το Γιάννη Σπανό και το Βασίλη Δημητρίου. Από το 1976 ως το 1981 τραγουδά συνθέσεις του Νίκου Καρβέλα και του Τόλη Βοσκόπουλου ενώ το 1985 συμμετέχει στη μεγάλη συναυλία αφιέρωμα για το Μάνο Λοΐζο στο Ολυμπιακό Στάδιο. Από τότε μέχρι το 2003 απέχει συνειδητά από τη δισκογραφία. Εκείνη τη χρονιά επανέρχεται με το διπλό cd "Τα αυθεντικά", ξανατραγουδώντας δικά του τραγούδια, καθώς και κάποια που είχαν περάσει στη δισκογραφία με άλλες φωνές. Πηγή. Στην ίδια πηγή μπορείτε επίσης να παρακολουθήσετε συνέντευξη με τον καλλιτέχνη.

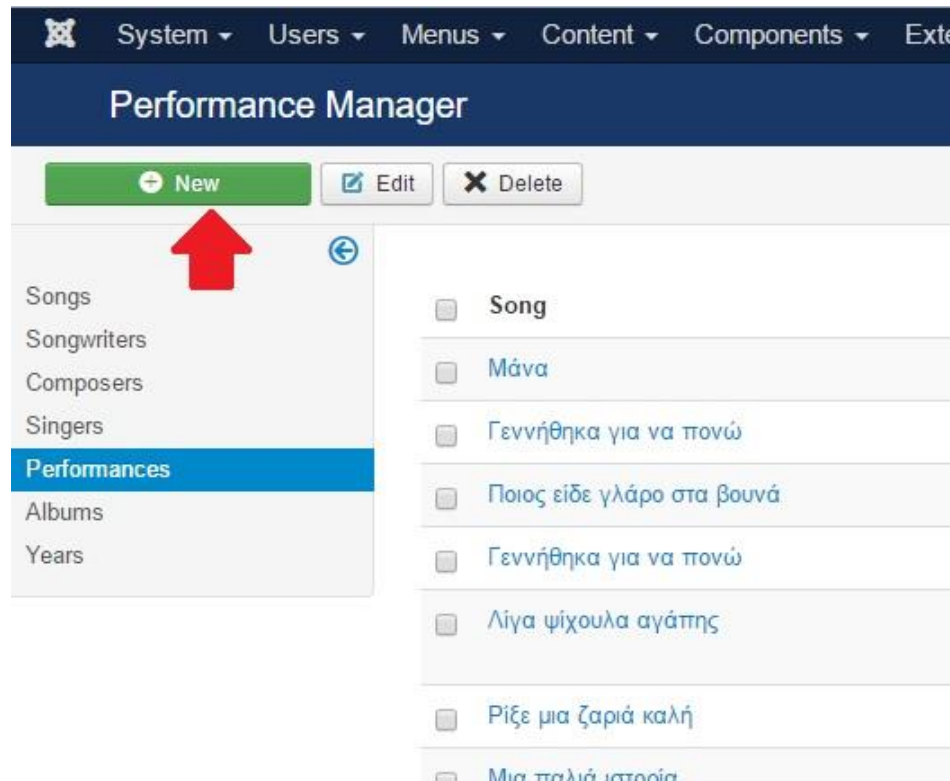
Τίτλος Τραγουδιού	Συνθέτης	Στιχουργός	Έτος
Τα χρυσά κλειδιά	Θόδωρος Δερβενιώτης	Κώστας Βίρβος	1968
Τα χρυσά κλειδιά	Θόδωρος Δερβενιώτης	Κώστας Βίρβος	1968
Τα χρυσά κλειδιά	Θόδωρος Δερβενιώτης	Κώστας Βίρβος	1968
Αγάπη μου περαστική	Θόδωρος Δερβενιώτης	Κώστας Βίρβος	1968
Ο Χατζηαβάτης	Μίμης Πλέσσας	Κώστας Βίρβος	1971

Εικόνα 7.13 – Εμφάνιση Τραγουδιστή στο Front-End



7.5. Δημιουργία νέας Εκτέλεσης

Επιλέγουμε “New” πάνω αριστερά αφού πρώτα επιλέξουμε Performances από τη λίστα.



Εικόνα 7.14 – Νέα Εκτέλεση

Συμπληρώνουμε τα παρακάτω στοιχεία:

- Song: το όνομα του τραγουδιού
- Year: χρονολογία της εκτέλεσης
- Singer name: όνομα τραγουδιστή
- Performance video: το λινκ με το βίντεο του τραγουδιού (αν υπάρχει)
- Performance Audio: το λινκ με το τραγούδι (αν υπάρχει)
- Performance Number: αριθμός εκτέλεσης, στην ουσία βάζουμε αν είναι η πρώτη, δεύτερη κτλ εκτέλεση
- Vocal Type: εδώ επιλέγουμε αν είναι ο βασικός τραγουδιστής ή αν κάνει φωνητικά



Εικόνα 7.15 – Εισαγωγή δεδομένων εκτέλεσης

Έτσι δημιουργούμε μια εκτέλεση. Η οποία φαίνεται στο τραγούδι με την παρακάτω μορφή.

Όπου κάτω από τον τίτλο, τον στιχουργό και τον συνθέτη βλέπουμε τις εκτελέσεις του τραγουδιού.

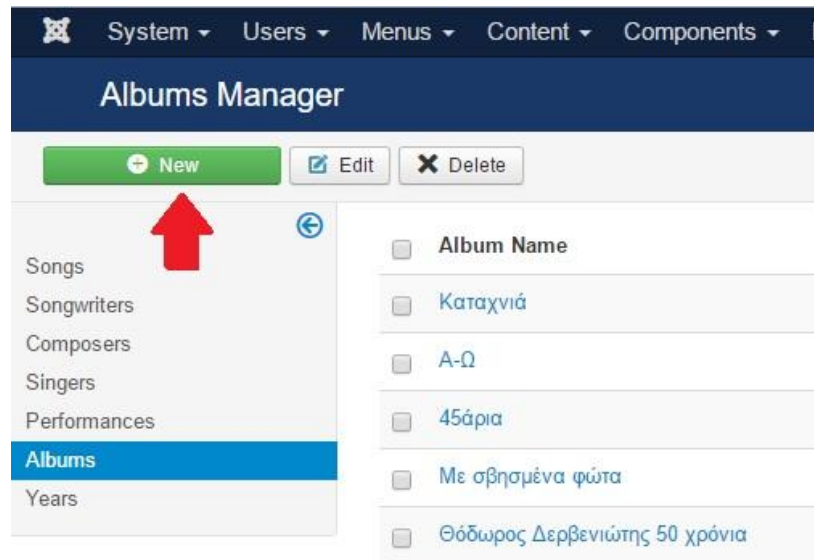
Εκτέλεση	Χρονιά	Καλλιτέχνης	Μορφή
1	1975	Γρηγόρης Μπιθικιώτης	Video
2	1975	Σταμάτης Κόκοτας	Video
3	1975	Βώξ	Video
4	NoYear	Άννα Βίση	Video
5	NoYear	Πέγκυ Ζήνα	Video
6	2007	Βασίλης Παπακωνσταντίνου	Video
7	2013	Αντωνής Ρέμος	Video
8	2012	Γρηγόρης Μπιθικιώτης II, Σταμάτης Κόκοτας, Στέλιος Διονυσίου, Χάρης Βαρβακούρης	Video
9	2014	Ανδρέας Ιμπάρκουλης, Σταμάτης Κόκοτας, Στέλιος Διονυσίου, Χάρης Βαρβακούρης	Video

Εικόνα 7.16 – Εμφάνιση εκτελέσεων τραγουδιού στο Front-End



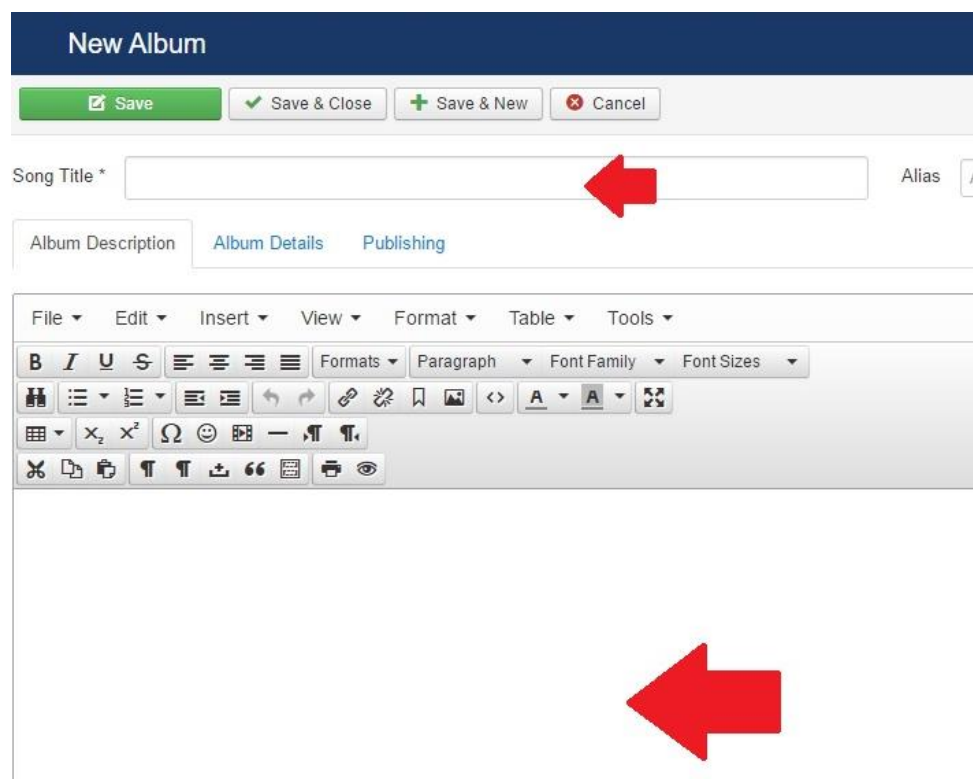
7.6. Δημιουργία νέου Album

Επιλέγουμε “New” πάνω αριστερά αφού πρώτα επιλέξουμε Albums από τη λίστα.



Εικόνα 7.17 – Νέο Album

Στη συνέχεια συμπληρώνουμε το Title όπου μπαίνει ο τίτλος του άλμπουμ



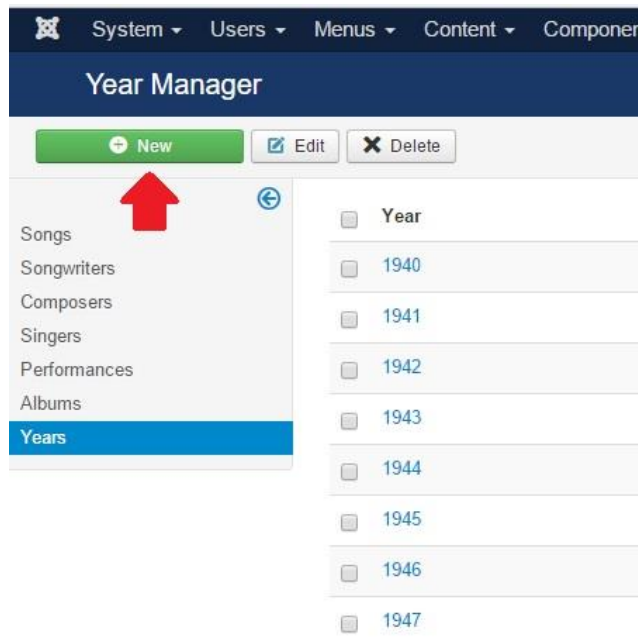
Εικόνα 7.18 : Εισαγωγή δεδομένων Album



Έτσι δημιουργούμε ένα άλμπουμ.

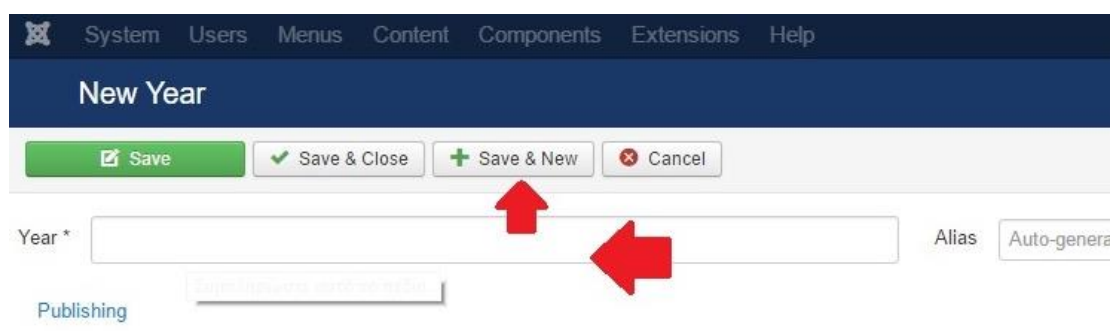
7.7. Δημιουργία νέου Έτους

Επιλέγουμε “New” πάνω αριστερά αφού πρώτα επιλέξουμε Years από τη λίστα.



Εικόνα 7.19 – Νέα Χρονολογία

Για να προσθέσουμε ένα νέο έτος στην ουσία συμπληρώνουμε το Year.



Εικόνα 7.20 – Εισαγωγή Δεδομένων Χρονολογίας

Έτσι προσθέτουμε ένα έτος.



8. Συμπεράσματα

Τα συμπεράσματα που αντλούμε από τη συγκεκριμένη διατριβή είναι ότι ο συνδυασμός από Γλώσσες προγραμματισμού (PHP, MySQL, HTML) και η σωστή γνώση της πλατφόρμας Joomla αποτελεί κλειδί για την δημιουργία οποιουδήποτε web project με όποια απαίτηση και αν υπάρχει. Παρέχονται αρκετές δυνατότητες ώστε να μπορέσουμε να δημιουργήσουμε και να διαμορφώσουμε εύχρηστες εφαρμογές.

Στο μέλλον, θα ήταν χρήσιμο να γίνουν κάποιες βελτιώσεις για την καλύτερη απεικόνιση της βάσης δεδομένων των τραγουδιών και να γίνουν κάποιες προσθήκες που θα βοηθήσουν την ταχύτερη εισαγωγή των δεδομένων από τους διαχειριστές.

Τέλος, αξίζει να σημειωθεί ότι το component που δημιουργήθηκε μπορεί σε άλλη ιστοσελίδα να φιλοξενήσει πολύ μεγαλύτερη βάση τραγουδιών με ακόμη περισσότερους καλλιτέχνες, τραγουδιστές, αλμπουμ κλπ. και να αποτελέσει μια τεράστια βάση δεδομένων.



9. Βιβλιογραφία

Πηγές

- [1] DEVELOPING AN MVC COMPONENT - Joomla Official Docs
(https://docs.joomla.org/J3.x:Developing_an_MVC_Component)
- [2] ΓΕΝΙΚΕΥΜΕΝΗ BROWSER-CENTRIC ΠΑΡΟΥΣΙΑΣΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ -
Διπλωματική Εργασία - Στέλλα Ν. Ανδρουλάκη (2009)
- [3] ΕΙΣΑΓΩΓΗ ΣΤΗ ΔΗΜΙΟΥΡΓΙΑ ΔΥΝΑΜΙΚΩΝ WEBSITES ΜΕ ΤΟ JOOMLA CMS - Digital
Academy (<http://www.dga.gr/web/publications/notes/Joomla.pdf>)
- [4] IMPLEMENTING A WEB-BASED LIBRARY SYSTEM WITH SOCIAL FEATURES —
Dimitris Klisiaris (<https://dklisiaris.gitbooks.io/thesis/content/index.html>)
- [5] JOOMLA PROGRAMMING MANUAL - Gianpiero Fasulo (2013) - Google Commerce Ltd
(<https://play.google.com/store/books/details?id=YlvjAgAAQBAJ&rdid=book-YlvjAgAAQBAJ>)
- [6] JOOMLA! PROGRAMMING - Mark Dexter & Louis Landry (2012) - Google Commerce Ltd
(https://play.google.com/store/books/details/Mark_Dexter_Joomla_Programming?id=TJREIRJICboC)
- [7] DEVELOPER TUTORIAL – JoomlaTutorials.com
(<https://www.joomlatutorials.com/joomla-tutorials/developer-tutorials>)
- [8] OFFICIAL JOOMLA! WEBSITE
(<https://www.joomla.com/>)
- [9] WIKIPEDIA, THE FREE ENCYCLOPEDIA
(<https://el.wikipedia.org>)
- [10] CODECADEMY: LEARN TO CODE
(<https://www.codecademy.com>)

Διαγράμματα

Όλα τα διαγράμματα δημιουργήθηκαν από το εργαλείο Microsoft Visio 2013 (τα πρωτότυπα αρχεία υπάρχουν στην ηλεκτρονική μορφή της εργασίας)

Εικόνες

Όλες οι εικόνες είναι απεικονίσεις από την live ιστοσελίδα (<http://kostasvirvos.gr/>) .