

Ατομική Διπλωματική Εργασία

**CREATING AN INTERACTIVE WEBSITE SUPPORT
CULTURAL HERITAGE OF CYPRUS IN ARCHITECTURE**

Αντρέας Λύμπουρας

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2013

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**CREATING AN INTERACTIVE WEBSITE SUPPORT CULTURAL
HERITAGE OF CYPRUS IN ARCHITECTURE**

Αντρέας Λύμπουρας

Επιβλέπων Καθηγητής

Δρ. Γιώργος Μ. Σαμάρας

Συν-Επιβλέπων Καθηγητής

Δρ. Παναγιώτης Αντρέου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2013

Ευχαριστίες

Ευχαριστώ τον καθηγητή μου Δρ. Γιώργο Σαμάρα, ο οποίος υπήρξε ο επιβλέπων καθηγητής της διπλωματικής μου εργασίας, για τη συνεχή υποστήριξη, που μου παρείχε κατά τη διάρκεια της εκπόνησής της. Επίσης τον ευχαριστώ για την ευκαιρία που μου έδωσε να δημιουργήσω αυτή την εργασία, από την οποία αποκόμισα αρκετά εφόδια, τόσο για τη συνέχιση των ακαδημαϊκών μου σπουδών, όσο και της επαγγελματικής μου σταδιοδρομίας. Επιπλέον, θα ήθελα να ευχαριστήσω τον συν-επιβλέποντα καθηγητή της διπλωματικής μου εργασίας Δρ. Παναγιώτη Αντρέου, για τη συνεχή καθοδήγηση, τις ιδέες, τις συμβουλές και την τεχνική βοήθεια, που μου πρόσφερε, καθ' όλη τη διάρκεια εκπόνησης της διπλωματικής εργασίας.

Ευχαριστώ επίσης, το Διδακτορικό φοιτητή Μάριο Μπέλκ για τη σχεδιαστική βοήθεια και υποστήριξη, καθώς και το Διδακτορικό φοιτητή του τμήματος Αρχιτεκτονικής, Πέτρο Φωκαΐδη για τις σχεδιαστικές του ικανότητες και για την άψογη συνεργασία μας, για την ανάπτυξη του συστήματος.

Τέλος, ευχαριστώ την οικογένεια μου και ιδιαίτερα τους γονείς μου Γιώργο και Τριανταφυλλιά για την ψυχολογική και οικονομική στήριξη, που μου προσέφεραν, καθ' όλη τη διάρκεια των σπουδών μου.

Περίληψη

Η ανάπτυξη των διαδικτυακών εφαρμογών υπερτερούν έναντι των Desktop εφαρμογών στα σημερινά δεδομένα, καθότι υπάρχουν πιο πολλά πλεονεκτήματα στη χρήση διαδικτυακών εφαρμογών, έναντι των Desktop εφαρμογών. Πρώτα απ' όλα μια διαδικτυακή εφαρμογή έχει πολύ λίγες απαιτήσεις, χρειάζεται μόνο ένα φυλλομετρητή, δίνοντας έτσι τη δυνατότητα να τρέξει και σε κινητή συσκευή. Επιπρόσθετα δεν απαιτείται η εγκατάσταση κάποιου προγράμματος σε υπολογιστή, ούτε συνεχή updates με κάθε αλλαγή, όπως γίνεται σε μια Desktop εφαρμογή.

Στόχος της παρούσας ατομικής διπλωματικής εργασίας είναι ο σχεδιασμός και η ανάπτυξη ενός λογισμικού, το οποίο θα επιτρέπει τους εξουσιοδοτημένους χρήστες να επεξεργάζονται τα δεδομένα της ιστοσελίδας τους, όπως τα Projects, Actors, Medias και τους Users. Θα υπάρχουν επίσης και οι απλοί χρήστες, οι οποίοι θα μπορούν να περιηγηθούν στον ιστιακό τόπο, κάνοντας αναζήτηση στο σύστημα, καθώς και να έχουν το δικό τους προσωπικό λογαριασμό, όπου θα αποθηκεύουν κάποια Projects, Actors ή Media για περεταίρω διερεύνηση.

Ο βασικότερος σκοπός της είναι να βοηθήσει το εργαστήριο Mesarch του Τμήματος Αρχιτεκτονικής του Πανεπιστημίου Κύπρου, να οργανώσει τις πληροφορίες που έχει για τα διάφορα αρχιτεκτονικά Projects.

Περιεχόμενα

Κεφάλαιο 1-Εισαγωγή	7
1.1 Υποκίνηση Έρευνας.....	7
1.2 Σκιαγράφηση Έρευνας.....	8
1.3 Συνεισφορά Εργασίας	8
1.4 Σκελετός Διπλωματικής Εργασίας.....	9
1.5 Ορισμοί, Ακρώνυμα και συντομογραφίες.....	9
Κεφάλαιο 2 -Σχετική εργασία	11
2.1 Υφιστάμενα συστήματα	11
2.2 Σύγκριση υπαρχόντων καθιερωμένων συστημάτων	19
2.3 Ανοικτά Θέματα/Συζήτηση.....	25
2.4 Σχετικά άρθρα	26
Κεφάλαιο 3 -Το σύστημα Mesarch	28
3.1 Ανάλυση απαιτήσεων.....	29
3.1.1 Χαρακτηριστικά Χρηστών	29
3.1.2 Διεπαφές Συστήματος	30
3.1.3 Περιορισμοί Συστήματος	34
3.2 Σενάρια Χρήσης (Use Cases).....	35
3.3 Λειτουργικότητα Συστήματος (Sequence).....	43
3.4 Λεπτομερή Σχεδίαση (Class Diagrams).....	51
3.5 Συστατικά Συστήματος	53
3.6 Προδιαγραφές Συστήματος.....	54
3.6.1 Περιορισμοί σχεδιασμού.....	55
3.6.2 Χαρακτηριστικά του συστήματος λογισμικού	55
3.6.3 Αξιοπιστία (Reliability)	55
3.6.4 Διαθεσιμότητα (Availability).....	56
3.6.5 Ασφάλεια (Security).....	57
3.6.6 Συντηρησιμότητα (Maintainability).....	57

3.6.7	Φορητότητα (Portability)	58
Κεφάλαιο 4 -Σχεδίαση Συστήματος Mesarch		59
4.1	Εισαγωγή.....	59
4.2	Αρχιτεκτονική Συστήματος.....	59
4.3	Εργαλεία Σχεδιασμού.....	60
4.3.1	Εργαλεία μοντελοποίησης.....	60
4.3.2	Εργαλεία ανάπτυξης πρωτοτύπων	61
4.4	Βάση Δεδομένων.....	62
4.4.1	ER diagram	62
4.5	Διαπροσωπεία Συστήματος.....	70
4.5.1	Πρωτότυπα σχεδίασης GUI	70
4.5.2	Graphical User Interface (GUI).....	76
Κεφάλαιο 5- Υλοποίηση Συστήματος Mesarch		77
5.1	Εισαγωγή.....	77
5.2	Εργαλεία Υλοποίησης.....	77
5.2.1	Τεχνολογίες	77
5.2.2	Εργαλεία ανάπτυξης λογισμικού.....	78
5.3	Βαση Δεδομένων.....	79
5.4	Διαπροσωπεία Συστήματος.....	85
Κεφάλαιο 6 -Βιβλιογραφία		88

Κεφάλαιο 1-Εισαγωγή

- 1.1 Υποκίνηση Έρευνας
 - 1.2 Σκιαγράφηση Έρευνας
 - 1.3 Συνεισφορά Εργασίας
 - 1.4 Σκελετός Διπλωματικής Εργασίας
 - 1.5 Ορισμοί, Ακρώνυμα και συντομογραφίες
-

Στο κεφάλαιο αυτό γίνεται μία εισαγωγή στο ερευνητικό πεδίο, με το οποίο ασχολείται η παρούσα διπλωματική εργασία. Αρχικά ορίζεται εν συντομία το πρόβλημα και ακολούθως παρουσιάζεται ο σκοπός της υλοποίησης του CMS συστήματος.

1.1 Υποκίνηση Έρευνας

Στο τμήμα αρχιτεκτονικής του πανεπιστημίου Κύπρου, υπάρχει το εργαστήριο Mesarch που ασχολείται με έρευνες στη σύγχρονη ιστορία της μοντέρνας αρχιτεκτονικής στην ανατολική Μεσόγειο. Το εργαστήριο θέλει να ψηφιοποιήσει τις πληροφορίες που έχει για ορισμένα έργα (Projects), τους εμπλεκόμενους σ' αυτά (Actors) καθώς και εικόνες, βίντεο κλπ (Medias). Θέλει επίσης να διαχειρίζεται αυτόνομα ένα δικτυακό τόπο, στον οποίο θα προσθέτει διάφορες πληροφορίες (εικόνες, περιγραφές, τίτλους κλπ) για Projects, Actors και Medias, τα οποία θα εμφανίζονται με συγκεκριμένο τρόπο στον απλό χρήστη. Εκτός από τον απλό χρήστη υπάρχει και ο χρήστης με λογαριασμό, ο οποίος μπορεί να φυλάει κάποια αντικείμενα για περαιτέρω διερεύνηση. Επίσης υπάρχουν και άλλοι χρήστες όπως system admins, webmasters, data admins, data entry-level 1, data entry-level 2, collaborator και collaborator-data entry, οι οποίοι μπορούν να χειριστούν διάφορες λειτουργίες του συστήματος.

1.2 Σκιαγράφηση Έρευνας

Ως πρώτη φάση της ανάπτυξης του πληροφοριακού συστήματος Mesarch, τέθηκε η γενική ιδέα υλοποίησης ενός συστήματος με τους συγκεκριμένους στόχους του. Ακολούθως έγινε μια μελέτη σκοπιμότητας, κατά πόσο θα ήταν εφικτή μια τέτοια υλοποίηση. Στη συνέχεια μας δόθηκαν οι απαιτήσεις του συστήματος και μέσα από μια διαδικασία πρωτοτυποποίησης, συναντήσεων και δημιουργίας Use Case και Sequence διαγραμμάτων, έγινε η συλλογή όλων των απαιτήσεων. Ακολούθως έγινε ο σχεδιασμός του συστήματος, μέσω της χρήσης UML διαγραμμάτων. Έγινε χρήση Class και ER διαγραμμάτων, καθώς και σχεσιακό μοντέλο για την αναπαράσταση του συστήματος, μέσω γραφικών, αφηρημένων μοντέλων. Στη φάση υλοποίησης του συστήματος το σύστημα εξελίχτηκε σιγά-σιγά, μέσω σχολίων και παροτρύνσεων από τον συν-επιβλέποντα καθηγητή μου, και των παρατηρήσεων των μελών του εργαστηρίου Mesarch. Τέλος έγινε η εγκατάσταση του συστήματος και η λειτουργία του, ενώ μέσω της συντήρησής του, θα γίνονται αλλαγές και θα διορθώνονται οποιαδήποτε λάθη.

Καθ' όλη την διάρκεια ανάπτυξης του συστήματος, έγινε χρήση του αυξητικού μοντέλου με το μοντέλο καταρράκτη, σε κάθε φάση.

1.3 Συνεισφορά Εργασίας

Η παρούσα διπλωματική εργασία εξυπηρετεί τις ανάγκες του εργαστηρίου MesArch, του Τμήματος Αρχιτεκτονικής, του Πανεπιστημίου Κύπρου, μέσω μίας διαδικτυακής εφαρμογής, η οποία θα χρησιμοποιείται για τη διαχείριση χρήσιμων δεδομένων, καθώς και για τις έρευνες του Τμήματος. Μπορεί επίσης να έχει πιο ευρεία χρήση, επηρεάζοντας την αρχιτεκτονική κοινότητα και το ευρύ κοινό.

Η ιστοσελίδα αυτή θα μπορεί να χρησιμοποιηθεί από οποιονδήποτε επιθυμεί να πάρει πληροφορίες, σχετικές για τη σύγχρονη ιστορία της μοντέρνας αρχιτεκτονικής στην Ανατολική Μεσόγειο.

Η εφαρμογή είναι όμορφη και απλή στη χρήση, ενώ εξίσου απλή είναι και η διαδικασία της ανάπτυξης και της συντήρησής της.

1.4 Σκελετός Διπλωματικής Εργασίας

Το πρώτο κεφάλαιο της ατομικής διπλωματικής εργασίας αποτελεί μια εισαγωγή (Κεφάλαιο 1-Εισαγωγή) στο ερευνητικό υπόβαθρο της εργασίας. Προσδιορίζεται το πρόβλημα και διασαφηνίζεται ο σκοπός της εργασίας αυτής.

Στο δεύτερο κεφάλαιο γίνεται μια έρευνα σε σχέση με υφιστάμενα καθιερωμένα συστήματα. (Κεφάλαιο 2 -Σχετική εργασία)

Στο τρίτο κεφάλαιο γίνεται η ανάλυση των απαιτήσεων της διαδικτυακής εφαρμογής (Ανάλυση απαιτήσεων), καθώς και τα διάφορα UML διαγράμματα, όπως σενάρια χρήσης (Σενάρια Χρήσης (Use Cases)), Sequence (Λειτουργικότητα Συστήματος (Sequence)) και Class διαγράμματα (Λεπτομερή Σχεδίαση (Class Diagrams)).

Στο τέταρτο κεφάλαιο παρουσιάζεται η αρχιτεκτονική του συστήματος (Αρχιτεκτονική Συστήματος), τα εργαλεία σχεδιασμού (Εργαλεία Σχεδιασμού), η βάση δεδομένων (Βάση Δεδομένων) και η διαπροσωπεία του συστήματος (Διαπροσωπεία Συστήματος), όσον αφορά τη σχεδίαση του συστήματος Mesarch.

Στο πέμπτο κεφάλαιο παρουσιάζονται τα εργαλεία υλοποίησης (Εργαλεία Υλοποίησης), η βάση δεδομένων (Βάση Δεδομένων), και η διαπροσωπεία του συστήματος (Διαπροσωπεία Συστήματος), όσον αφορά την υλοποίηση του συστήματος Mesarch.

1.5 Ορισμοί, Ακρόνυμα και συντομογραφίες

- ΒΔ: Βάση Δεδομένων.
- ΔΕ: Διαδικτυακή Εφαρμογή.

- URL (Uniform Resource Locator): Καθορίζει τη διεύθυνση μιας ιστοσελίδας.
- HTML (Hyper Text Markup Language): Προτεινόμενη γλώσσα προγραμματισμού για τη δημιουργία της διαδικτυακής εφαρμογής.
- CMS (Content Management Systems): Είναι διαδικτυακές εφαρμογές, που επιτρέπουν την online τροποποίηση του περιεχομένου ενός δικτυακού τόπου.

Κεφάλαιο 2 -Σχετική εργασία

2.1 Υφιστάμενα συστήματα

2.2 Σύγκριση υπαρχόντων καθιερωμένων συστημάτων

2.3 Ανοικτά Θέματα/Συζήτηση

2.4 Σχετικά άρθρα

Αρχικά, θα γίνεται μια αναφορά στο τι είναι στατική, δυναμική και CMS ιστοσελίδα και στη συνέχεια μια αναφορά σε υπάρχοντες στατικές, δυναμικές και CMS ιστοσελίδες, κάνοντας μια σύγκριση αυτών, βάση των λειτουργιών και τεχνικών χαρακτηριστικών του καθενός. Στο τέλος γίνεται μια πιο σαφή διευκρίνιση, όσον αφορά τα πλεονεκτήματα και μειονεκτήματα της κάθε ιστοσελίδας. Τέλος καταλήγουμε στους λόγους, που πάρθηκε η απόφαση για την επιλογή αυτού του είδους εφαρμογής, καθώς και στα πλεονεκτήματα της Mesarch εφαρμογής έναντι άλλων.

2.1 Υφιστάμενα συστήματα

Στο παρόν στάδιο, υπάρχουν διάφορα συστήματα δημιουργίας εμφάνισης ηλεκτρονικού περιεχομένου, τα οποία μπορούν να χωριστούν σε δύο βασικές κατηγορίες: τις στατικές ιστοσελίδες και τις δυναμικές.

Οι **δυναμικές ιστοσελίδες** είναι οι ιστοσελίδες, που έχουν δυναμικό περιεχόμενο. Σαν δυναμικό περιεχόμενο ορίζονται οι πληροφορίες που εμφανίζονται στο site, δίνοντας στον ιδιοκτήτη της ιστοσελίδας την ικανότητα να μπορεί να ενημερώνει, να διαγράφει και να προσθέτει δεδομένα στη βάση δεδομένων, όπου ως συνήθως καταχωρούνται οι πληροφορίες αυτές. Μερικά παραδείγματα από δυναμικά διαδικτυακά συστήματα είναι τα

συστήματα διαχείρισης περιεχομένου, (Content Management Systems-CMS). Τα CMS μπορούν να χρησιμοποιηθούν για ειδήσεις (εφημερίδες, περιοδικά, πρακτορεία ειδήσεων, παρουσιάσεις προϊόντων, online υποστήριξη, αγγελίες και ανακοινώσεις, διαφημίσεις, δελτία τύπου κλπ.

Τα Συστήματα Διαχείρισης Περιεχομένου- ΣΔΠ, (Content Management Systems-CMS) είναι διαδικτυακές εφαρμογές, που επιτρέπουν την online τροποποίηση του περιεχομένου ενός δικτυακού τόπου. Οι διαχειριστές, μέσω του διαδικτύου, ενημερώνουν το περιεχόμενο του ΣΔΠ, το οποίο είναι εγκατεστημένο σ' ένα διακομιστή (server). Οι αλλαγές αυτές γίνονται αυτόματα διαθέσιμες πάλι μέσω του διαδικτύου, σε όλους τους επισκέπτες και χρήστες του δικτυακού τόπου.

Η Διαχείριση Περιεχομένου Ιστοσελίδων - ΔΠΙ (Web Content Management system - WCM) είναι ένα σύστημα CMS, σχεδιασμένο για να απλοποιήσει τη δημοσίευση του δικτυακού περιεχομένου των websites και να επιτρέπει στους δημιουργούς περιεχομένου να υποβάλουν περιεχόμενο, χωρίς να απαιτεί τεχνική γνώση HTML ή μεταφόρτωση των αρχείων.

Οι **στατικές ιστοσελίδες** είναι οι ιστοσελίδες, στις οποίες οι εμφανίσεις της πληροφορίας προς τους χρήστες, είναι προκαθορισμένες και στατικές. Αυτού του τύπου ιστοσελίδες εμφανίζουν την ίδια πληροφορία σε όλους τους χρήστες. Παρόλα αυτά, ο διαχειριστής της ιστοσελίδας, μπορεί να κάνει περιοδικές ενημερώσεις (updates), όπως προσθήκη κειμένου, εικόνων ή άλλες αλλαγές, όσον αφορά τη διεπαφή, τα οποία όμως απαιτούν διαδικτυακές γνώσεις και προγραμματιστικές επιδεξιότητες. Οι στατικές ιστοσελίδες μπορούν να έχουν και δυναμική συμπεριφορά, την οποία χειρίζεται εξολοκλήρου ο χρήστης (μέσω του φυλλομετρητή). Αυτές μπορεί να είναι κάποια JavaScript χαρακτηριστικά, όπως μεγένθυση μιας εικόνας, ή εμφάνιση κάποιου δυναμικού μηνύματος (alert). Γενικά, οι επισκέπτες-χρήστες της ιστοσελίδας, δεν μπορούν να ελέγχουν εξολοκλήρου τι είδους πληροφορία να λαμβάνουν, παρά μόνο να βλέπουν την πληροφορία, την οποία ο διαχειριστής ιστοσελίδας αποφάσισε να προσφέρει σ' αυτούς. Παρόλα αυτά σελίδες, οι οποίες είναι αποθηκευμένες σε βάση δεδομένων και εξυπηρετούν πελάτες, μέσω server, πάλι μπορεί να θεωρηθούν ως

στατικές σελίδες, εφόσον δεν αλλάζει το περιεχόμενο τους και παρουσιάζεται το ίδιο σε όλους, ακριβώς όπως αποθηκεύτηκε.

Στατικές ιστοσελίδες - Παραδείγματα

Η **Schott glass** ιστοσελίδα (<http://www.schott.com>) είναι μια παγκόσμια, Γερμανική εταιρεία, η οποία παράγει ειδικά προϊόντα από γυαλί. Δίνει ιδέες σε διάφορους τομείς, όπως οικιακές συσκευές, φαρμακευτικά είδη, ηλεκτρονικά, οπτικά, αυτοκινητοβιομηχανία και πολλές άλλες βιομηχανίες. Στην ιστοσελίδα τους βρίσκονται τα διάφορα projects τους χωρισμένα σε κατηγορίες (αρχιτεκτονικά, οικιακές συσκευές, οπτικά, φωτισμό και άλλα). Φαίνονται επίσης διάφορες δημοσιεύσεις τους και βίντεο, ενώ δίνεται η δυνατότητα να κατεβούν μερικά περιοδικά (magazines) και δικά τους projects. Επίσης δίνονται πληροφορίες για την εταιρεία, για ορισμένα είδη γυαλιών, καθώς και για τις έρευνές τους. Γενικά η σελίδα διαχωρίζεται σε Products, About, Events and News και Worldwide addresses και πληροφορίες για την εταιρεία. Πρόκειται για μια αρκετά μεγάλη ιστοσελίδα, με πολλές πληροφορίες. Παρόλα αυτά είναι υλοποιημένη μόνο σε HTML με μεγάλο αριθμό html αρχείων, γεγονός που την καθιστά στατική.

Η **Hiroshi Sugimoto** ιστοσελίδα (<http://www.sugimotohiroshi.com>) είναι μια προσωπική ιστοσελίδα του Hiroshi Sugimoto, η οποία παρουσιάζει το Portfolio με τις δουλειές, που έχει ασχοληθεί μέχρι τώρα. Περιλαμβάνει επίσης τη βιογραφία του, τη βιβλιογραφία που έχει συγγράψει, καθώς και τα events, που έχουν σχέση με αυτόν. Πρόκειται για μια μικρού μεγέθους ιστοσελίδα, με λίγες πληροφορίες. Είναι υλοποιημένη μόνο σε HTML, γεγονός που την καθιστά στατική.

Η **Agilitynut** ιστοσελίδα (<http://www.agilitynut.com/roadside.html>) είναι μια προσωπική ιστοσελίδα της Debra Jane Seltzer, η οποία περιέχει εικόνες σχετικές με την αρχιτεκτονική του δρόμου (roadside). Η αρχιτεκτονική αυτή είναι χωρισμένη σε κτήρια, αγάλματα και

others. Η ιστοσελίδα έχει πάρα πολλές εικόνες. Είναι υλοποιημένη μόνο σε HTML και όπως γράφει και η ίδια είναι στατική, διότι είναι ο μόνος τρόπος που ξέρει να χειρίζεται και δε θέλει να δώσει 8000 δολάρια για να την κάνει δυναμική.

Η **Great Buildings** ιστοσελίδα (<http://www.greatbuildings.com/gbc.html>) είναι μια ιστοσελίδα, που έχει τα καλύτερα κτήρια, αρχιτέκτονες, αρχιτεκτονικά βιβλία, 3D μοντέλα, κατηγοριοποιημένα σε τύπους αρχιτεκτονικών κτηρίων και σε περιοχές. Πρόκειται για μια αρκετά μεγάλη ιστοσελίδα, με πολλές πληροφορίες. Παρόλα αυτά είναι υλοποιημένη μόνο σε HTML με μεγάλο αριθμό html αρχείων, γεγονός που την καθιστά στατική.

Η **Artlex** ιστοσελίδα (<http://www.artlex.com/>) είναι μια σελίδα, που παρουσιάζει την τέχνη όλων των εποχών του κόσμου. Είναι χωρισμένη σε κατηγορίες βάση ονόματος αλλά και είδους τέχνης. Πρόκειται για μια αρκετά μεγάλη ιστοσελίδα, με πολλές πληροφορίες. Παρόλα αυτά είναι υλοποιημένη μόνο σε HTML με μεγάλο αριθμό html αρχείων, γεγονός που την καθιστά στατική.

Δυναμικές ιστοσελίδες – Παραδείγματα

Το **Archigram** (<http://archigram.westminster.ac.uk/>) είναι μια ιστοσελίδα ενός αρχιτεκτονικού group, η οποία προβάλλει τις πληροφορίες(δουλειά) της (Projects, Drawings, Ephemera, People, Magazines, Shows, Bibliography-Publications) για το διεθνές κοινό για επαγγελματική ή μη μελέτη. Είναι υλοποιημένη σε PHP και αποτελείται από μια βάση δεδομένων, υλοποιημένη σε MySQL, με την οποία και επικοινωνεί, για την εξαγωγή των δεδομένων, τα οποία εμφανίζονται στο φυλλομετρητή. Όμως τα δεδομένα (Projects, Drawings, Ephemera κλπ) αυτής της ιστοσελίδας είναι στατικά, είναι προκαθορισμένα-αποθηκευμένα στη βάση δεδομένων και εμφανίζονται με το συγκεκριμένο τρόπο. Δεν

μπορούν όλοι οι χρήστες του συστήματος να αλλάξουν τα δεδομένα, παρά μόνο ο διαχειριστής της βάσης δεδομένων και της ιστοσελίδας.

Η **Marcel Breurer digital archive** ιστοσελίδα (<http://breuer.syr.edu/>) είναι μια προσωπική ιστοσελίδα του αρχιτέκτονα Marcel Breurer, όπου προβάλλει την δουλειά του. Η ιστοσελίδα του είναι οργανωμένη σε project τελειωμένα και ατέλειωτα, έχοντας πρόσβαση σε drawings, letters και photographs. Είναι υλοποιημένη σε PHP και αποτελείται από μια βάση δεδομένων, υλοποιημένη σε MySQL, με την οποία και επικοινωνεί, για την εξαγωγή των δεδομένων, τα οποία εμφανίζονται στο φυλλομετρητή.

Η **Sadler Brown Architecture** ιστοσελίδα (<http://www.sadlerbrown.co.uk/>) είναι η ιστοσελίδα μιας Αγγλικής αρχιτεκτονικής εταιρείας, όπου προβάλλονται τα Projects της. Η ιστοσελίδα αυτή έχει πληροφορίες όσον αφορά το προσωπικό, τα νέα, το προφίλ της εταιρείας αλλά και τα projects, που έκανε μέχρι τώρα. Αυτή είναι υλοποιημένη σε Microsoft Visual C#, με μια απλή βάση δεδομένων για εμφάνιση των Projects, σε Microsoft SQL Server. Τα δεδομένα της ιστοσελίδας αυτής, επίσης εμφανίζονται στατικά.

Δυναμικές CMS ιστοσελίδες - Παραδείγματα

Η **Let us design/build** ιστοσελίδα (<http://www.letusserveyou.com/>) είναι μια προσωπική ιστοσελίδα του Joel Harner, ο οποίος προσφέρει υπηρεσίες σπιτιού, όπως ανακαινίσεις και συντηρήσεις. Η ιστοσελίδα αυτή χωρίζεται στις κατηγορίες: υπηρεσίες τις οποίες προσφέρει, τα πιο διάσημα σπίτια που άλλαξε (portfolio), επεξήγηση των κατηγοριών για διατήρηση-συντήρηση σπιτιών. Πρόκειται για μια σελίδα Joomla-based υλοποιημένη σε PHP με MySQL βάση δεδομένων. Είναι μια δυναμική ιστοσελίδα, όπου κάποιοι admin χρήστες μπορούν να αλλάξουν το περιεχόμενό της δυναμικά.

Η **Group Home Deal** ιστοσελίδα (<http://www.grouphomedeal.com/>) είναι η ιστοσελίδα μιας εταιρείας, η οποία κάνει προσφορές σε πελάτες της, σε ήδη υπάρχοντα projects στην Ινδία, ανάλογα με το πόσα άτομα θα κάνουν join κάποιο project. Η ιστοσελίδα αποθηκεύει τα projects αυτά, τους πελάτες της, οι οποίοι εγγράφονται στα projects (investors), τους ιδιοκτήτες γης (landowners) και τους χρηματομεσίτες (brokers) . Επίσης αποθηκεύει και τις συμφωνίες, που έκλεισαν μέχρι τώρα, καθώς και τα group προσφορών που δημιουργήθηκαν. Πρόκειται για μια σελίδα Joomla-based, υλοποιημένη σε PHP με MySQL βάση δεδομένων, με την οποία και επικοινωνεί, για την εξαγωγή των δεδομένων, τα οποία εμφανίζονται στο φυλλομετρητή. Πρόκειται για μια δυναμική ιστοσελίδα, όπου κάποιοι admin χρήστες μπορούν να αλλάξουν το περιεχόμενό της δυναμικά.

Η **Archdaily** ιστοσελίδα (<http://www.archdaily.com/>) είναι μια ιστοσελίδα όπου προβάλλονται τα τελευταία νέα στον κόσμο της αρχιτεκτονικής, όπως projects, products, events, interviews και competitions. Προσφέρει γνώση, έμπνευση και εργαλεία στους επισκέπτες της αλλά και σε αρχιτέκτονες, οι οποίοι μπορούν να εγγραφούν για να ανεβάσουν δικά τους project, news, awards or events. Μπορεί να κατηγοριοποιηθεί σε έξι κύρια θέματα: selected works, Arch Daily software, books & magazines, Arch Daily classics, interviews and their building of the year competition. Πρόκειται για μια σελίδα Joomla-based, υλοποιημένη σε PHP με MySQL βάση δεδομένων. Είναι μια δυναμική ιστοσελίδα, όπου κάποιοι admin χρήστες μπορούν να αλλάξουν το περιεχόμενό της δυναμικά.

Η **Architizer** ιστοσελίδα (<http://www.architizer.com/>) είναι μια κοινωνική δικτύωση, που ενώνει τους διάφορους αρχιτέκτονες ανά τον κόσμο. Αρχιτέκτονες αλλά και εταιρείες μπορούν να εγγραφούν και να ανεβάσουν τη δουλειά τους, μοιράζοντας την με άλλους αρχιτέκτονες, αλλά και πιθανούς πελάτες. Η σελίδα είναι χωρισμένη σε projects, people, firms και products. Επίσης το Architizer δίνει τη δυνατότητα στους relationship-builders να συμμετέχουν σε competitions, list jobs και να βλέπουν νέα της σελίδας. Πρόκειται για μια σελίδα Joomla-based υλοποιημένη σε PHP με MySQL βάση δεδομένων. Είναι μια

δυναμική ιστοσελίδα, όπου κάποιοι admin χρήστες μπορούν να αλλάξουν το περιεχόμενο της δυναμικά.

Η **Homedit** ιστοσελίδα (<http://www.homedit.com/>) παρέχει τη δυνατότητα στο χρήστη να βρει τα τελευταία νέα, άρθρα και θέματα για την αρχιτεκτονική, την εσωτερική διακόσμηση (interior design) και τις επιπλώσεις. Υπάρχουν πάνω από 10,000 άρθρα με περισσότερες από 60,000 εικόνες και ιδέες. Μπορεί να κατηγοριοποιηθεί σε 9 κύριες κατηγορίες: apartments, appliances, architecture, bathrooms, chairs, interiors, lighting, living rooms και το how-to. Επιπρόσθετα οι χρήστες μπορούν να υποβάλουν μια αρχιτεκτονική ιδέα, στέλνοντας απλά ένα e-mail στον ιδιοκτήτη της σελίδας, με μια περιγραφή και μερικές εικόνες. Πρόκειται για μια σελίδα Word Press-based, υλοποιημένη σε PHP με MySQL βάση δεδομένων. Είναι μια δυναμική ιστοσελίδα, όπου κάποιοι admin χρήστες μπορούν να αλλάξουν το περιεχόμενο της δυναμικά.

Η **Archello** ιστοσελίδα (<http://www.archello.com/>), επιτρέπει σε ανθρώπους αλλά και εταιρείες να μοιράσουν την εμπειρία τους στο αρχιτεκτονικό περιβάλλον. Άνθρωποι και εταιρείες μπορούν να ενωθούν μεταξύ τους στα projects, προϊόντα(products) και υλικά(materials). Κάνοντας αυτές τις ενώσεις μεταξύ τους μπορούν να δείξουν το προφίλ τους στην αρχιτεκτονική βιομηχανία. Υπάρχουν άνθρωποι (people entity) ενωμένοι σε εταιρείες και projects ενωμένα πάνω σ' αυτές. Ο κάθε χρήστης επίσης μπορεί να προσθέσει νέα υλικά και projects στον λογαριασμό του, τα οποία μπορούν να σχολιάσουν άλλοι εγγεγραμμένοι χρήστες αλλά και να τα αποθηκεύσουν ως αγαπημένα. Μπορεί να κατηγοριοποιηθεί σε projects, products, materials, people και companies. Πρόκειται για μια σελίδα Word Press-based υλοποιημένη σε PHP με MySQL βάση δεδομένων. Είναι μια δυναμική ιστοσελίδα, όπου κάποιοι admin χρήστες μπορούν να αλλάξουν το περιεχόμενο της δυναμικά.

Η **Contemporist** ιστοσελίδα (<http://www.contemporist.com>) παρουσιάζει το σύγχρονο πολιτισμό και επικεντρώνεται στην αρχιτεκτονική, το σχεδιασμό, την τέχνη και τα ταξίδια.

Πρόκειται για μια σελίδα Word Press-based, υλοποιημένη σε PHP με MySQL βάση δεδομένων. Είναι μια δυναμική ιστοσελίδα, όπου κάποιοι admin χρήστες μπορούν να αλλάξουν το περιεχόμενό της δυναμικά.

2.2 Σύγκριση υπαρχόντων καθιερωμένων συστημάτων

Τεχνικά χαρακτηριστικά

		Web page	Prog.Language	Database
Static	P1.Schott	http://www.schott.com	html	-
	P2.Hiroshi Sugimoto	http://www.sugimotohirosi.com	html	-
	P3.Agilitynut	http://www.agilitynut.com/roadside.html	html	-
	P4.Great Buildings	http://www.greatbuildings.com/gbc.html	html	-
	P5.Artlex	http://www.artlex.com/	html	-
Dynamic	P6.Archigram	http://archigram.westminster.ac.uk	php	MySQL
	P7.Marcel Breurer	http://breuer.syr.edu/	php	MySQL
	P8.Sadler Brown Architecture	http://www.sadlerbrown.co.uk/	ASP.NET	-
Dynamic - CMS	P9.Let us design/build	http://www.letusserveyou.com/	php	MySQL
	P10.GroupHomeDeal	http://www.grouphomedea.com/	php	MySQL
	P11.Arch daily	http://www.archdaily.com/	php	MySQL
	P12.Architizer	http://www.architizer.com/	php	MySQL

	P13.Homedit	http://www.homedit.com/	php	MySQL
	P14.Archello	http://www.archello.com/	php	MySQL
	P15.Contemporist	http://www.contemporist.com	php	MySQL

Οντότητες

		Entities
Static	P1.Schott	Products, About, Events, News, Worldwide address and info
	P2.Hiroshi Sugimoto	Biography, Bibliography, Event
	P3.Agilitynut	Buildings, Statues, Others
	P4.Great Buildings	Buildings, Architects, Books, 3D Models
	P5.Artlex	Name and Architectural type categories
Dynamic	P6.Archigram	Projects, Drawings, Ephemera, People, Magazines, Shows, Bibliography-Publications
	P7.Marcel Breurer	Drawings, Photographs, Letters
	P8.Sadler Brown Architecture	Projects, News, People
Dynamic - CMS	P9.Let us design/build	Portfolio, Services
	P10.GroupHomeDeal	Projects, Investors, Land owners, Brokers, Deals made
	P11.Arch daily	Projects, Products, Events, News, Interviews, Competitions
	P12.Architizer	Projects, People, Firms, Products
	P13.Homedit	Apartments, Appliances, Architecture, Bathrooms, Chairs, Interiors, Lighting, Living rooms, How-to
	P14.Archello	Projects, Products, Materials, People
	P15.Contemporist	Architecture, Design, Art, Travel

Λειτουργικότητες

		search	adv search	filters	login	admin panel	sitemap on page	sitemap	other languages	follow on network	share on network
Static	P1.Schott	x			x		x	x	x	x	x
	P2.Hiroshi Sugimoto										
	P3.Agilitynut	x									
	P4.Great Buildings	x	x								
	P5.Artlex	x									
Dynamic	P6.Archigram	x	x	x				x			
	P7.Marcel Breurer	x	x	x			x				
	P8.Sadler Brown										
Dynamic- CMS	P9.Let us design/build					x		x		x	x
	P10.GroupHomeDeal	x	x		x	x			x		
	P11.Arch daily	x	x		x	x	x		x		x
	P12.Architizer	x	x	x	x	x					
	P13.Homedit	x				x				x	
	P14.Archello	x	x	x	x	x					x
	P15.Contemporist	x				x				x	x

Λειτουργικότητες - συνέχεια

		AJAX	print page	videos	download	Google map	network of architects	workplace	send email	comments on projects	project recommend
Static	P1.Schott	x	x	x	x						
	P2.Hiroshi Sugimoto										
	P3.Agilitynut										
	P4.Great Buildings					x			x		
	P5.Artlex										
Dynamic	P6.Archigram	x							x		
	P7.Marcel Breurer										
	P8.Sadler Brown	x									
Dynamic- CMS	P9.Let us design/build								x		
	P10.GroupHomeDeal	x					x	x			
	P11.Arch daily	x		x				x	x	x	x
	P12.Architizer						x	x	x		x
	P13.Homedit								x	x	
	P14.Archello	x				x	x	x		x	
	P15.Contemporist									x	

Βλέποντας αναλυτικά τα πιο πάνω είδη συστήματος και παρουσιάζοντας μερικά παραδείγματα από τα διαφορετικά είδη, συμπεραίνουμε ότι και οι δύο κατηγορίες ιστοσελίδων που υπάρχουν, είναι χρήσιμες και μπορούν να χρησιμοποιηθούν ανάλογα με τις απαιτήσεις του πελάτη.

Για παράδειγμα οι **στατικές ιστοσελίδες** είναι πιο οικονομικές, καθότι απαιτούνται απλές γνώσεις για την υλοποίησή τους, επίσης η συντήρησή τους είναι πιο φθηνή, αφού συνήθως δεν έχουν βάση δεδομένων και είναι σταθερές, χωρίς αλλαγές. Τα δεδομένα τους είναι στατικά, δεν αλλάζουν μέχρι να ανεβεί νέο version στο διακομιστή (server) τους, δείχνοντας έτσι σταθερή πληροφορία στους χρήστες τους. Παρόλα αυτά εμφανίζονται γρήγορα από τον server, αφού οι σελίδες τους είναι απλές και δεν χρειάζεται επιπλέον εργασία από το server για να τις εμφανίσει. Όλες οι σελίδες μιας στατικής ιστοσελίδας δημιουργούνται από ειδικούς προγραμματιστές ιστοσελίδων. Η δημιουργία μιας νέας σελίδας ή η προσθήκη μιας νέας λειτουργικότητας είναι χρονοβόρο και δύσκολο έργο, που απαιτεί εξειδικευμένες γνώσεις.

Από την άλλη οι **δυναμικές ιστοσελίδες** προσφέρουν πιο πολλές λειτουργίες στους χρήστες, όπως ενημέρωση του περιεχομένου από οπουδήποτε, ταυτόχρονη ενημέρωση από πολλούς χρήστες και διαφορετικούς υπολογιστές, τροποποιούνται με πιο εύκολο και γρήγορο τρόπο και γενικά αποτελούν ένα πιο ολοκληρωμένο σύστημα για τη συνεργασία των χρηστών του συστήματος, χωρίς να απαιτούνται εξειδικευμένες γνώσεις προγραμματισμού ιστοσελίδων. Είναι αυτό που λέμε data base driven type ιστοσελίδες, όπου ο server θα δημιουργήσει μια σελίδα στο χρήστη, ανάλογα με το τι ζήτησε να δει. Όπως για παράδειγμα σε ένα online κατάστημα, όταν ένα αντικείμενο πωληθεί, ενημερώνεται ο διακομιστής (server) για να αλλάξει η βάση δεδομένων, και στο επόμενο αίτημα του χρήστη θα δει την αλλαγμένη έκδοση (version) της σελίδας. Οι δυναμικές ιστοσελίδες είναι πιο αργές και πιο ακριβές να αναπτυχθούν, ενώ η συντήρηση του διακομιστή τους (server), κοστίζει περισσότερο, λόγω της χρήσης βάσης δεδομένων. Εδώ ο server έχει να κάνει πιο πολύ δουλειά και συμφέρει πολύ για μεγάλες ιστοσελίδες..

Η διαφορά στην ταχύτητα, σε σχέση με τις στατικές, δεν είναι μεγάλη μέχρι και είκοσι διαδικτυακές σελίδες, λόγω των επιπλέον λειτουργιών που χρειάζεται μια δυναμική

ιστοσελίδα), ενώ είναι αισθητή για περισσότερες από πενήντα διαδικτυακές σελίδες. Ειδικότερα, για περισσότερες από εκατό σελίδες, η διαφορά είναι τεράστια, διότι έχουμε και μικρότερο φόρτο στο διακομιστή (server) και χρήση λιγότερου χώρου, λόγω του ότι δεν υπάρχουν πολλές επαναλαμβανόμενες στατικές σελίδες (η ανάπτυξη των διαδικτυακών σελίδων γίνεται δυναμικά).

Μια CMS (Content Management System) ιστοσελίδα μπορεί να έχει διαφόρους χρήστες, οι οποίοι μπορούν να έχουν πρόσβαση σε ορισμένες διαδικτυακές σελίδες του συστήματος. Ονομάζονται Content Management Systems, διότι κάποιοι χρήστες έχουν τη δυνατότητα να αλλάζουν το περιεχόμενο των διαδικτυακών σελίδων προσθέτοντας, διαγράφοντας ή μπορούν να επεξεργάζονται τη σελίδα. Αυτό την κάνει ακόμα πιο δυναμική ιστοσελίδα.

Τα είδη των συστημάτων διαφοροποιούνται, ανάλογα με τις υπηρεσίες που προσφέρουν, την προγραμματιστική γλώσσα με την οποία είναι υλοποιημένα, αλλά και την αρχιτεκτονική τους, όπως για παράδειγμα είναι ο server που χρησιμοποιεί το καθένα.

Συνήθως ο διακομιστής (server) είναι είτε Microsoft Server είτε LAMP Server.

Η Microsoft Server τεχνολογία αποτελείται από ASP.NET, ή MSSQL τεχνολογίες. Όντας ιδιόκτητο πρόγραμμα έχει επιπλέον κόστος για την υιοθέτησή του, το οποίο πρέπει να ληφθεί υπόψη για την επιλογή πλατφόρμας, στην οποία η ιστοσελίδα θα τρέξει.

Οι LAMP Server τεχνολογία είναι βασισμένες σε Linux, Apache, PHP, MySQL τεχνολογίες. Το κύριο πλεονέκτημα της χρήσης ενός τέτοιου συστήματος είναι ότι είναι ανοικτού κώδικα (open source) και διαθέτουν δωρεάν λογισμικό, οπότε δεν υπάρχει επιπλέον κόστος. Οι LAMP Servers είναι πολύ δυνατοί servers, όντας σταθεροί και πολύ ασφαλισμένοι.

2.3 Ανοικτά Θέματα/Συζήτηση

Η Mesarch διαδικτυακή εφαρμογή είναι δυναμική CMS ιστοσελίδα, υλοποιημένη σε Microsoft Server τεχνολογία, ASP.NET με SQL Server για τη βάση δεδομένων. Η επιλογή

για δημιουργία μιας δυναμικής και όχι στατικής ιστοσελίδας, έγινε λόγω του ότι το εργαστήριο Mesarch ήθελε να μπορεί να ενημερώνεται το περιεχομένο της από οπουδήποτε, να έχει τη δυνατότητα για ταυτόχρονη ενημέρωση από πολλούς χρήστες και διαφορετικούς υπολογιστές και γενικά να μπορεί να διαχειρίζεται μόνη της το όλο σύστημα, καθώς θα υπάρχουν αρκετές αλλαγές-τροποποιήσεις στις πληροφορίες του συστήματος. Επίσης αφού τα μέλη του εργαστηρίου δεν έχουν εξειδικευμένες γνώσεις προγραμματισμού ιστοσελίδων, μια δυναμική ιστοσελίδα θα τους βοηθούσε πολύ.

Η επιλογή χρήσης ASP.NET τεχνολογίας, λήφθηκε λόγω του ότι είναι η ιστοσελίδα καθίσταται πιο ασφαλισμένη από διάφορους κακοπροαίρετους, που θέλουν να κάνουν κακό στη βάση δεδομένων ή να κλέψουν κάποιες προσωπικές πληροφορίες από αυτή. Η ASP.NET προνόησε για διάφορα είδη επιθέσεων από τους κακοπροαίρετους χρήστες, όπως τα SQL Injection attacks, τα οποία προσπαθούν να εκμεταλλευτούν τις αδυναμίες της SQL και μέσω queries να καταστρέψουν ή να υποκλέψουν πληροφορίες. Επίσης η ASP.NET δημιουργήθηκε από την Microsoft, ενώ η PHP είναι open source, δημιουργημένη από διάφορους εθελοντές, κάνοντας την πιο αξιόπιστη από την PHP.

Ειδικότερα η ιστοσελίδα επιλέχθηκε να υλοποιηθεί στην c#.NET αντί VB.NET, γλώσσα προγραμματισμού, λόγω του ότι είχα εμπειρίες με αυτή τη γλώσσα, αφού είχα υλοποιήσει παρόμοια συστήματα σ' αυτή.

Η κύρια διαφορά με τις άλλες CMS ιστοσελίδες, που υπάρχουν είναι ότι η παρούσα ιστοσελίδα είναι φτιαγμένη ακριβώς στα μέτρα, που τη χρειάζονταν το εργαστήριο Mesarch. Μέσω των συναντήσεων και των προτύπων ιστοσελίδων καταλήξαμε σε αυτό το συγκεκριμένο template, το οποίο δημιουργήθηκε για τους συγκεκριμένους σκοπούς, που το χρειάζονταν το εργαστήριο.

2.4 Σχετικά άρθρα

Το “Computational Historiography: Data Mining in a Century of Classics Journals” (Mimno, 2012) είναι μια δημοσίευση, η οποία μιλά για τις υπολογιστικές μεθόδους

ανάλυσης κειμένου (κάποιων journals), που μπορούν να εφαρμοστούν για την υπολογιστική ιστογραφία στις ανθρωπιστικές συλλογές. Πιο συγκεκριμένα εξηγεί, μέσω casestudies, πώς μπορούν να αντιμετωπιστούν οι διάφορες δυσκολίες γύρω από το θέμα .

To “Interactive multimedia ethnography: Archiving workflow, interface aesthetics and metadata” (Matthews&Aston, 2012) είναι μια δημοσίευση, η οποία μιλά για μια προσέγγιση για ένα end-to-end σύστημα για την ψηφιακή αρχειοθέτηση, κάνοντας χρήση metadata για την ευκολότερη αλληλεπίδραση του χρήστη με το σύστημα. Το άρθρο προτείνει νέα interface εργαλεία, τα οποία είναι εύκολα, οικονομικά και γρήγορα, εκμεταλλευόμενα τα metadata, open content και τα description standards. Το σύστημα που προτείνει το άρθρο κάνει χρήση σχετικά φτηνού λογισμικού και διαδικτυακής τεχνολογίας, αλλά επίσης είναι επεκτάσιμο για μεγαλύτερες συλλογές media.

To “Computing similarity between items in a digital library of cultural heritage” (Aletras, Stevenson, Clough, 2012) είναι μια δημοσίευση, η οποία μιλά για αυτοματοποιημένες τεχνικές κατηγοριοποίησης όμοιων αντικειμένων. Αυτό μπορεί να βοηθήσει τους χρήστες να εξερευνήσουν και να ανακαλύψουν δεδομένα από τις ψηφιακές συλλογές-βιβλιοθήκες πολιτιστικής κληρονομιάς. Ο συνδυασμός και των δύο τεχνικών για υπολογισμό ομοιότητας, το knowledge-based και corpus-based, δίνουν τα καλύτερα αποτελέσματα και στα δύο πειράματα, που περιγράφονται στο άρθρο. Το knowledge-based κάνει χρήση εξωτερικών πηγών γνώσης και το corpus-based αναλύει τη συχνότητα των λέξεων σε έγγραφα.

Κεφάλαιο 3 -Το σύστημα Mesarch

- 3.1 Ανάλυση απαιτήσεων
 - 3.1.1 Χαρακτηριστικά Χρηστών
 - 3.1.2 Διεπαφές Συστήματος
 - 3.1.2.1 Διεπαφές Χρήστη
 - 3.1.2.2 Διεπαφές Λογισμικού
 - 3.1.2.3 Διεπαφές Επικοινωνιών
 - 3.1.3 Περιορισμοί Συστήματος
- 3.2 Σενάρια Χρήσης (Use Cases)
- 3.3 Λειτουργικότητα Συστήματος (Sequence)
- 3.4 Λεπτομερή Σχεδίαση (Class Diagrams)
- 3.5 Συστατικά Συστήματος
- 3.6 Προδιαγραφές Συστήματος
 - 3.6.1 Περιορισμοί σχεδιασμού
 - 3.6.2 Χαρακτηριστικά συστήματος λογισμικού
 - 3.6.3 Αξιοπιστία (Reliability)
 - 3.6.4 Διαθεσιμότητα (Availability)
 - 3.6.5 Ασφάλεια (Security)
 - 3.6.6 Συντηρησιμότητα (Maintainability)
 - 3.6.7 Φορατότητα (Portability)

Στο κεφάλαιο αυτό καθορίζεται το σύστημα Mesarch. Πιο συγκεκριμένα αναλύονται οι απαιτήσεις, οι προδιαγραφές του συστήματος, κάνοντας αναφορά στα χαρακτηριστικά χρηστών, στις διεπαφές του συστήματος και στους περιορισμούς του. Επιπλέον στο κεφάλαιο αυτό παρουσιάζονται τα σενάρια χρήσης, τα sequence και class διαγράμματα.

3.1 Ανάλυση απαιτήσεων

3.1.1 Χαρακτηριστικά Χρηστών

Υπάρχουν τέσσερα είδη χρηστών που διαχειρίζονται τα δεδομένα, είναι οι **απλοί χρήστες με λογαριασμό και χωρίς λογαριασμό, system admins, webmasters, data admins, data entry-level 1, data entry-level 2, collaborator και collaborator-data entry.**

Οι **data admins, data entry users level 1, data entry users level 2 και collaborators-data entry** είναι οι χρήστες οι οποίοι μπορούν να δημιουργήσουν και να επεξεργαστούν δεδομένα στο σύστημα. Όλοι εκτός των **data admins** πρέπει να στέλνουν τα αντικείμενα που δημιουργούν προς έγκριση (approval). Μπορούν επίσης να βλέπουν private και archived πληροφορίες του συστήματος.

Οι **data admins και data entry users level 1** είναι οι χρήστες που μπορούν να δημιουργήσουν, να επεξεργαστούν και να διαγράψουν (αυτά που έβαλαν οι ίδιοι) υπάρχοντα δεδομένα στο σύστημα. Επίσης μπορούν να κάνουν approve τα δεδομένα, που εισάγουν άλλοι.

Τις κρυμμένες πληροφορίες του συστήματος (Mesarch Notes, Other Info κλπ), μπορούν να τις βλέπουν μόνο οι **data admins, data entry users level 1 και data entry users level 2.**

Όσον αφορά την επεξεργασία στα Thesaurus lists, οι **data admins και data entry users level 1** μπορούν να τα επεξεργαστούν όλα, ενώ οι **data entry users level 2 και collaborators-data entry** μόνο τα Main construction system, Secondary construction system, media's format και media's keywords πεδία.

Οι **system admins** μπορούν να κάνουν backup / restore το σύστημα ή τα δεδομένα της σελίδας, ενώ οι **webmasters** να επεξεργάζονται τους χρήστες του συστήματος. Επίσης οι **data admins και data entry-level 1**, μπορούν να επεξεργάζονται τους χρήστες (διαγραφή, αλλαγή ρόλου, αλλαγή ονόματος κλπ.), να δημιουργήσουν, να επεξεργαστούν και να διαγράψουν οτιδήποτε στο σύστημα, αλλά επίσης και να εισάγουν νέους χρήστες στο σύστημα, να διαγράψουν τους υπάρχοντες χρήστες, να κάνουν approve τα δεδομένα, που

εισάγουν άλλοι και γενικά μπορούν να κάνουν τα πάντα, μέχρι και να διαγράψουν ολόκληρη τη βάση δεδομένων.

Απλοί χρήστες χωρίς λογαριασμό: Μπορούν απλά να κινούνται-σερφάρουν στην ιστοσελίδα, χωρίς επιπλέον προτερήματα (Navigating).

Απλοί χρήστες με λογαριασμό: Μπορούν να σερφάρουν, όπως οι χρήστες χωρίς λογαριασμό (account), αλλά επίσης να αποθηκεύουν κάποια projects, media, actors για το Workplace τους, έτσι ώστε να τα βρίσκουν πιο εύκολα και γρήγορα. (Navigating + saving date in My Workplace).

3.1.2 Διεπαφές Συστήματος

3.1.2.1 Διεπαφές Χρήστη

Διεπαφή απλού χρήστη: Ο απλός χρήστης μπορεί να δει τα διάφορα projects, media, actors, να κάνει αναζήτηση στη σελίδα, να δημιουργήσει νέο λογαριασμό, και γενικά να περιηγηθεί σε αυτήν.

Διεπαφή συνδεδεμένου χρήστη: Μπορεί να κάνει ότι και ο απλός ο χρήστης, αλλά επίσης να αποθηκεύει κάποια projects, media, actors στο λογαριασμό του, για να τα βρει πιο εύκολα και γρήγορα. (έχει το δικό του Workplace).

Διεπαφή administrator: Μπορούν να δουν ότι βλέπει και ένας απλός χρήστης. Υπάρχουν διάφορα είδη χρηστών, που μπορούν να έχουν πρόσβαση ο καθένας στα όρια των αναθετόντων ρόλων του. Μπορούν να δημιουργηθούν δυναμικά νέοι τύποι ρόλων εφαρμοσμένοι στις ανάγκες του συστήματος. Ανάλογα με τον χρήστη μπορούν να δουν τις διάφορες φόρμες εισαγωγής δεδομένων, όπως φόρμες για εισαγωγή project, media, actor, ή τα thesaurus lists, ή να δουν επιπλέον private ή archived πληροφορίες του συστήματος ή ακόμα και private projects, actors ή media. Επίσης μπορούν να δουν και να επεξεργαστούν

τους διάφορους χρήστες, που υπάρχουν στο σύστημα, καθώς και τα διάφορα αντικείμενα έτοιμα για έγκριση (approved).

Φόρμες και schemas τα οποία δημιουργήθηκαν στη βάση δεδομένων μπορούν να είναι είτε **ACTIVE** είτε **INACTIVE**.

Στην κατηγορία ACTIVE υπάρχουν τα:

PENDING – τα οποία είναι ατελείωτα projects ή actors.

APPROVED – τα οποία είναι τελειωμένα και εγκεκριμένα projects, ή actors.

3.1.2.2 Διεπαφές Λογισμικού

3.1.2.2.1 MICROSOFT VISUAL STUDIO 2010

- Όνομα: Microsoft Visual Studio 2010
- Αριθμός Έκδοσης: 2010
- Πηγή: Microsoft
- Τεκμηρίωση: <http://msdn.microsoft.com/en-us/vcsharp/default.aspx>
- Σκοπός: Το λογισμικό χρησιμοποιήθηκε για τον προγραμματισμό του web application. Η επιλογή γλώσσας ήταν η C#.Ο λόγος επιλογής αυτής της προγραμματιστικής γλώσσας ήταν ότι παρέχει μια πολύ καλή διεπαφή και μπορεί εύκολα να χρησιμοποιηθεί. Προβάλλει μια χρήσιμη εργαλειοθήκη για δημιουργία φορμών, συμβουλές για τον προγραμματιστή, πολύ καλή διαχείριση λαθών και είναι βασισμένη στη χρήση γεγονότων. Υποστηρίζει επιπλέον παράλληλο προγραμματισμό, αντικειμενοστραφή προγραμματισμό και μπορεί εύκολα να συνδυαστεί με τον MSSQL Server.

3.1.2.2.2 SQL SERVER

- Όνομα: MS SQL Server.
- Αριθμός Έκδοσης: 2010.
- Πηγή: Microsoft Corporation.
- Τεκμηρίωση: <http://www.microsoft.com/sqlserver/en/us/default.aspx>
- Σκοπός: Το λογισμικό αυτό βοήθησε στη δημιουργία της Βάση Δεδομένων, που χρησιμοποιεί η εφαρμογή. Ο λόγος που επιλέχθηκε ο MSSQL Server ως η Βάση Δεδομένων είναι η συμβατότητά του με το front system, το οποίο θα υλοποιηθεί σε NET Framework.

3.1.2.2.3 ADOBE FLASH PROFESIONAL CS3

- Όνομα: Adobe flash professional CS3
- Αριθμός Έκδοσης: CS3
- Πηγή: Adobe
- Τεκμηρίωση: <https://www.adobe.com/cfusion/tdrc/index.cfm?product=flashpro>
- Σκοπός: Με αυτό το λογισμικό έγιναν όλες οι εικόνες για την διαδικτυακή εφαρμογή του συστήματος, όπως βελάκια, εικόνες για το πιθανό logo του εργαστηρίου κλπ.

3.1.2.2.4 ADOBE FIREWORKS CS4

- Όνομα: Adobe fireworks CS4
- Αριθμός Έκδοσης: CS4
- Πηγή: Adobe

- Τεκμηρίωση:
<http://www.adobe.com/cfusion/tdrc/index.cfm?loc=en&product=fireworks>
- Σκοπός: Με αυτό το λογισμικό δημιουργήθηκαν διάφορες εικόνες logo για τη εφαρμογή, έτσι ώστε να επιλεγεί η κατάλληλη.

3.1.2.2.5 ADOBE DREAMWEAVER CS4

- Όνομα: Adobe dream weaver CS4
- Αριθμός Έκδοσης: CS4
- Πηγή: Adobe
- Τεκμηρίωση: <http://www.adobe.com/cfusion/tdrc/index.cfm?product=dreamweaver>
- Σκοπός: Με αυτό το λογισμικό δημιουργήθηκαν τα διάφορα πρωτότυπα για την εφαρμογή του συστήματος, έτσι ώστε οι πελάτες να έχουν μια πρώτη επαφή με το σύστημα, καθώς επίσης και για επιλογή καταλλήλου-αρεστού επικαλύμματος (template) για τη διαδικτυακή εφαρμογή.

3.1.2.2.6 ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ

- Όνομα: Windows XP, Windows Vista, Windows 7
- Έκδοση: Home Edition, Professional Edition, Ultimate Edition
- Πηγή: Microsoft

3.1.2.3 Hardware Interfaces

Για την λειτουργία της εφαρμογής, μπορεί να χρησιμοποιηθεί οποιαδήποτε ηλεκτρονική συσκευή, από desktop PC, laptop, tablet, smart phone κλπ, η οποία έχει πρόσβαση με το διαδίκτυο και ένα φυλλομετρητή. Έτσι αν είναι φορητή η συσκευή θα μπορεί από οπουδήποτε βρίσκεται ο χρήστης, να επεξεργάζεται τα δεδομένα ή να κάνει περιήγηση στη διαδικτυακή εφαρμογή.

3.1.2.4 Διεπαφές Επικοινωνιών

Η εφαρμογή θα είναι διαδικτυακή, οπότε απαραίτητη προϋπόθεση για να λειτουργήσει είναι μια σύνδεση με το διαδίκτυο, μέσω οποιουδήποτε φυλλομετρητή. Επίσης, το λειτουργικό σύστημα πρέπει να υλοποιεί οποιαδήποτε τεχνολογία δικτύου, όπως είναι η Wi-Fi (IEEE 802.11) ή το Ethernet (IEEE 802.3).

3.1.3 Περιορισμοί Συστήματος

Το σύστημα αποτελείται από τρία βασικά χαρακτηριστικά: τον Server, τη διεπαφή του Author και τη διεπαφή του Administrator.

Προαπαιτούμενο για την υποστήριξη των διεπαφών του απλού χρήστη και των χρηστών που επεξεργάζονται δεδομένα, είναι η πρόσβαση στο Διαδίκτυο. Η πρόσβαση μπορεί να είναι είτε ασύρματη ή με καλώδιο.

Η βάση δεδομένων πρέπει να είναι πάντα σταθερή και να προβάλλει πάντα ασφάλεια στα δεδομένα, λόγω των πληροφοριών, που υπάρχουν από την ελεγχόμενη πρόσβαση. Επίσης, ο Server πρέπει να έχει και αυτός ελεγχόμενη πρόσβαση.

Γενικά, το σύστημα πρέπει να είναι απλό. Το GUI πρέπει να είναι εύκολο και σταθερό, έχοντας το ίδιο πρότυπο σε όλες τις οθόνες, όπως και τα ονόματα, θα πρέπει να είναι περιγραφικά.

3.2 Σενάρια Χρήσης (Use Cases)

General use case scenario



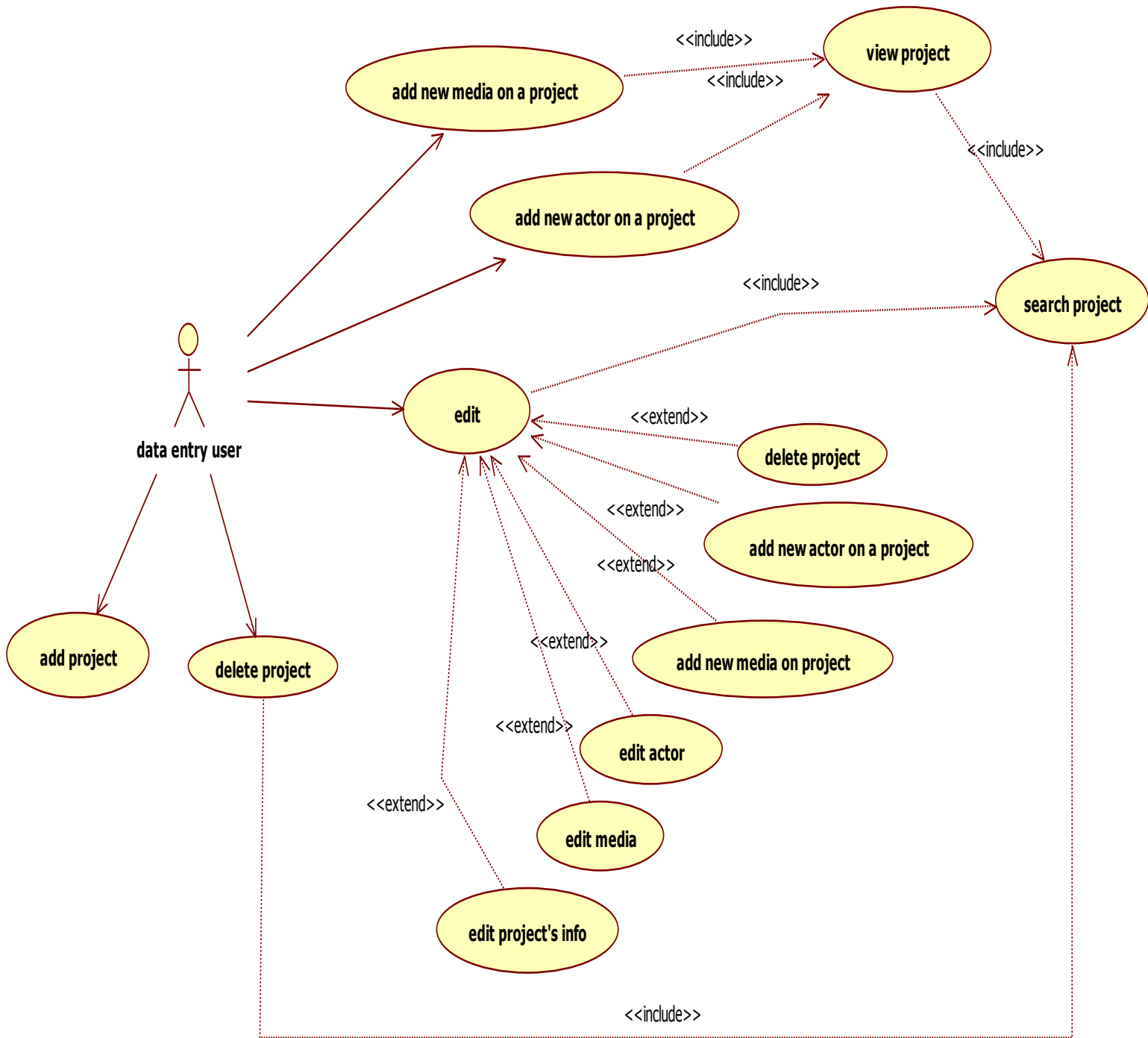
Το πιο πάνω διάγραμμα περιγράφει γενικά τις βασικές λειτουργίες, που μπορούν να κάνουν οι βασικοί χρήστες του συστήματος. Οι απλοί χρήστες συνδεδεμένοι στο λογαριασμό τους και αυτοί που δεν είναι, κληρονομούν από τον απλό χρήστη, χωρίς λογαριασμό. Οι data entry χρήστες κληρονομούν από τους συνδεδεμένους χρήστες και οι administrators από τους data entry χρήστες.

Ένας απλός χρήστης μπορεί να δει τα projects, actors, medias, αφού πρώτα κάνει filter ή search. Μπορεί επίσης να δει πληροφορίες για την σελίδα και το εργαστήριο, που είναι υπεύθυνο για αυτήν (about info), αλλά και βλέποντας ένα project να δει πληροφορίες για media, actors σχετικά με το project. Ένας χρήστης μη συνδεδεμένος στο σύστημα μπορεί, είτε να συνδεθεί βάζοντας τα συνθηματικά του, είτε να κάνει νέο λογαριασμό. Ένας συνδεδεμένος χρήστης, μπορεί να αλλάξει τις πληροφορίες του, να αποθηκεύσει projects, actors, medias (αφού πρώτα τα βρει μέσω search ή filtering), στο λογαριασμό του (workspace).

Ένας data entry user μπορεί να διαχειριστεί projects, actors, medias και thesaurus δεδομένα.

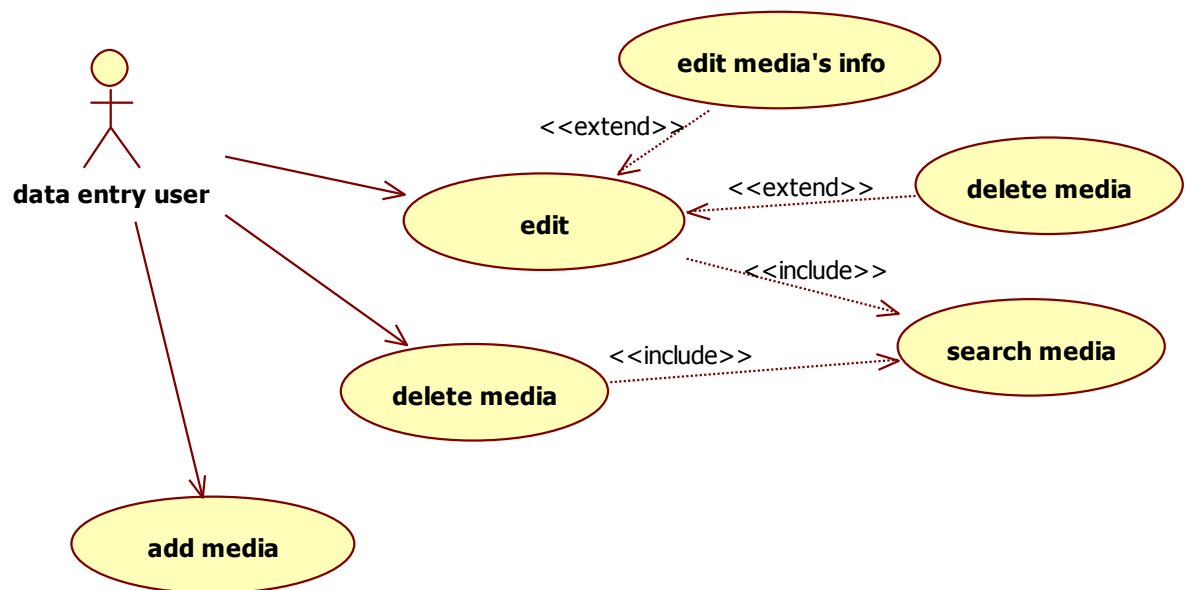
Ένας admin μπορεί επιπλέον να διαχειριστεί τους χρήστες του συστήματος.

Manage projects:



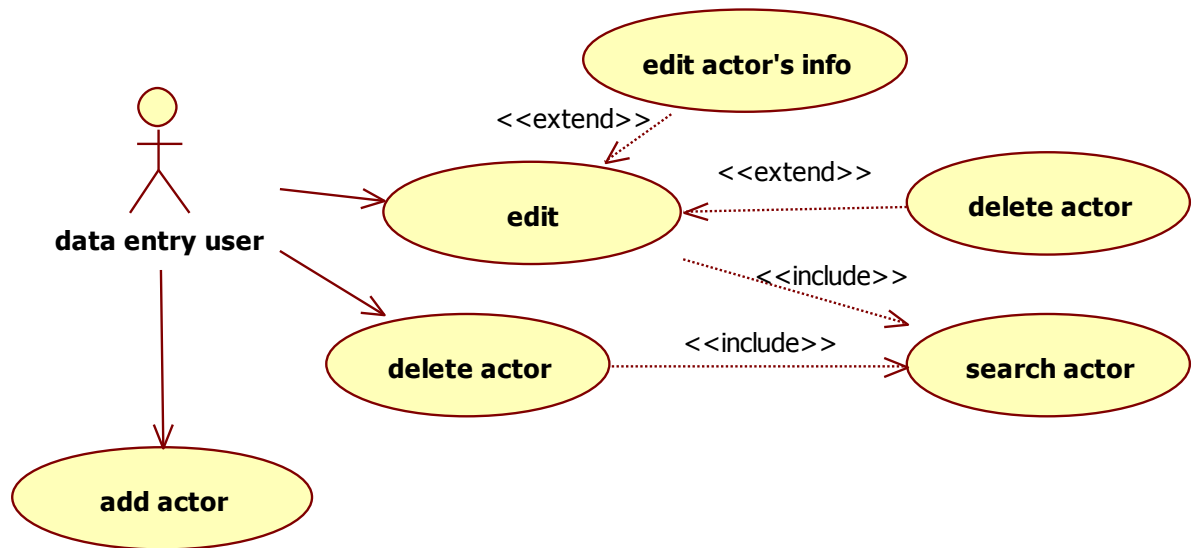
Στο πιο πάνω διάγραμμα περιγράφεται η βασική λειτουργία διαχείρισης των projects. Ο υπεύθυνος διαχειριστής μπορεί να προσθέσει ένα νέο project. Μπορεί επίσης να το διαγράψει αφού κάνει search. Μετά το search βλέποντας ένα project μπορεί επίσης να προσθέσει νέο actor, ή media πάνω σε αυτό. Κάνοντας edit κάποιο project (αφού πρώτα γίνει search), ο data entry χρήστης μπορεί να διαγράψει το project, να προσθέσει κάποιο actor ή media, να ενημερώσει τις πληροφορίες του project αυτού, αλλά και να επεξεργαστεί τις πληροφορίες των media και actors που έχουν σχέση με το project.

Manage media:



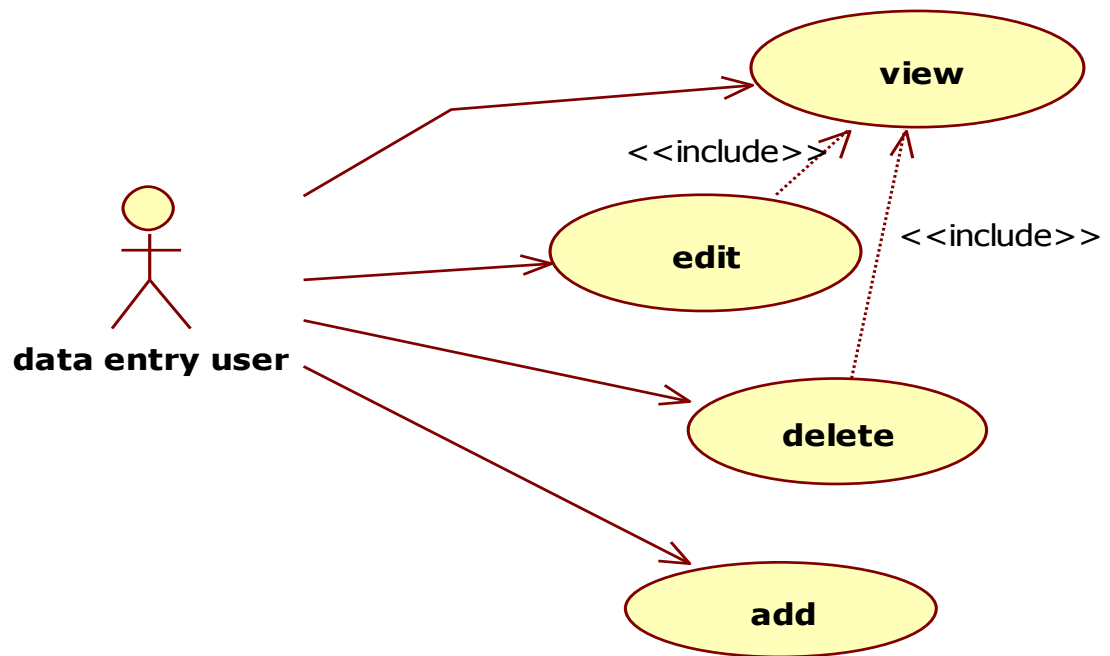
Στο πιο πάνω διάγραμμα περιγράφεται η βασική λειτουργία διαχείρισης των medias. Ο υπεύθυνος διαχειριστής μπορεί να προσθέσει ένα νέο media. Μπορεί επίσης να το διαγράψει αφού κάνει search. Κάνοντας edit κάποιο media (αφού πρώτα γίνει search), ο data entry χρήστης μπορεί να διαγράψει το media αυτό και να ενημερώσει τις πληροφορίες του.

Manage actors:



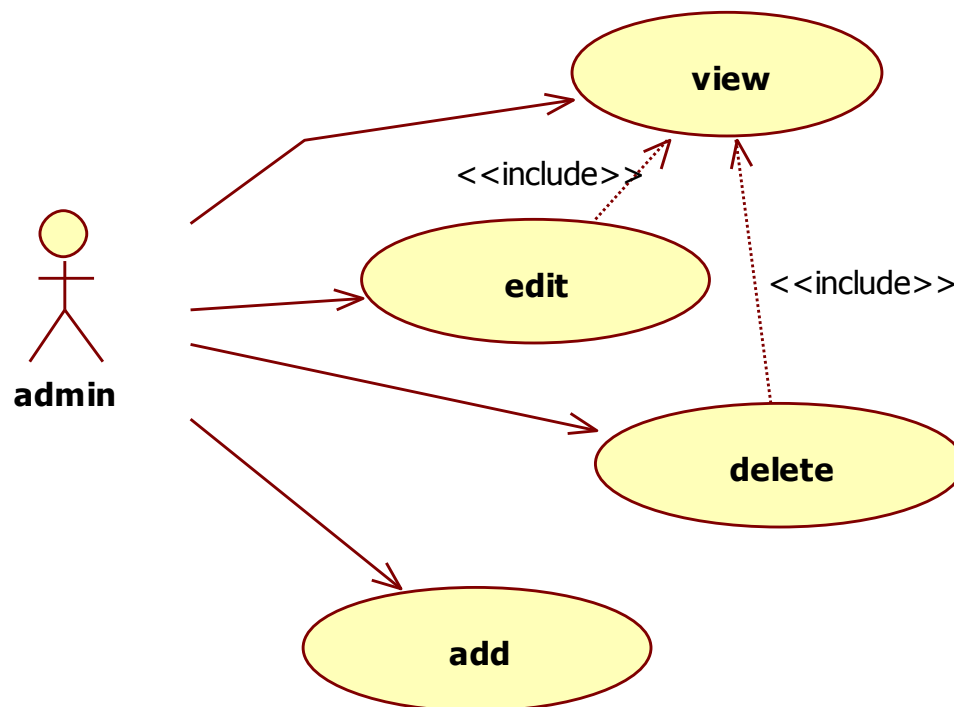
Στο πιο πάνω διάγραμμα περιγράφεται η βασική λειτουργία διαχείρισης των actors. Ο υπεύθυνος διαχειριστής μπορεί να προσθέσει ένα νέο actor. Μπορεί επίσης να το διαγράψει αφού κάνει search. Κάνοντας edit κάποιο actor (αφού πρώτα γίνει search), ο data entry χρήστης μπορεί να διαγράψει τον actor αυτό και να ενημερώσει τις πληροφορίες του.

Manage thesaurus:



Στο πιο πάνω διάγραμμα περιγράφεται η βασική λειτουργία διαχείρισης των thesaurus lists. Ο υπεύθυνος διαχειριστής μπορεί να προσθέσει ένα νέο thesaurus entry. Μπορεί επίσης να το διαγράψει, αφού κάνει view την ανάλογη ενότητά του. Κάνοντας edit κάποιο thesaurus entry (αφού πρώτα γίνει view), ο data entry χρήστης, μπορεί να ενημερώσει τις πληροφορίες του (όνομα).

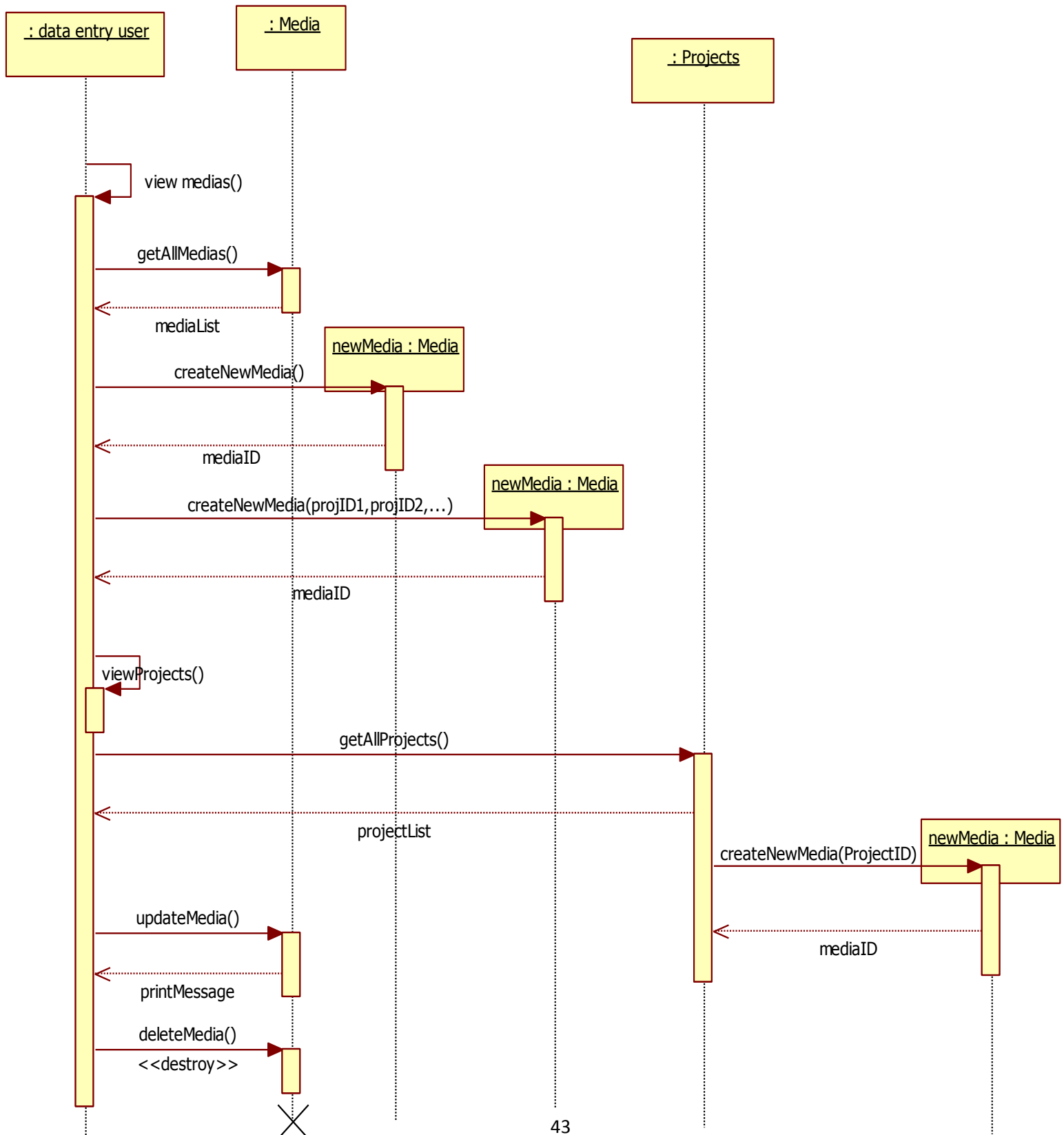
Manage users:



Στο πιο πάνω διάγραμμα περιγράφεται η βασική λειτουργία διαχείρισης των χρηστών. Ο υπεύθυνος διαχειριστής μπορεί να προσθέσει ένα νέο χρήστη. Μπορεί επίσης να τον διαγράψει αφού του κάνει view όλους τους χρήστες. Κάνοντας edit κάποιο χρήστη (αφού πρώτα γίνουν view όλοι) ο data entry χρήστης μπορεί να ενημερώσει τις πληροφορίες του.

3.3 Λειτουργικότητα Συστήματος (Sequence)

Data entry user – Media:



Στο πιο πάνω διάγραμμα περιγράφονται οι βασικές λειτουργίες, που μπορεί να κάνει ένας διαχειριστής media (data entry χρήστης) για την εισαγωγή, ενημέρωση και διαγραφή ενός media.

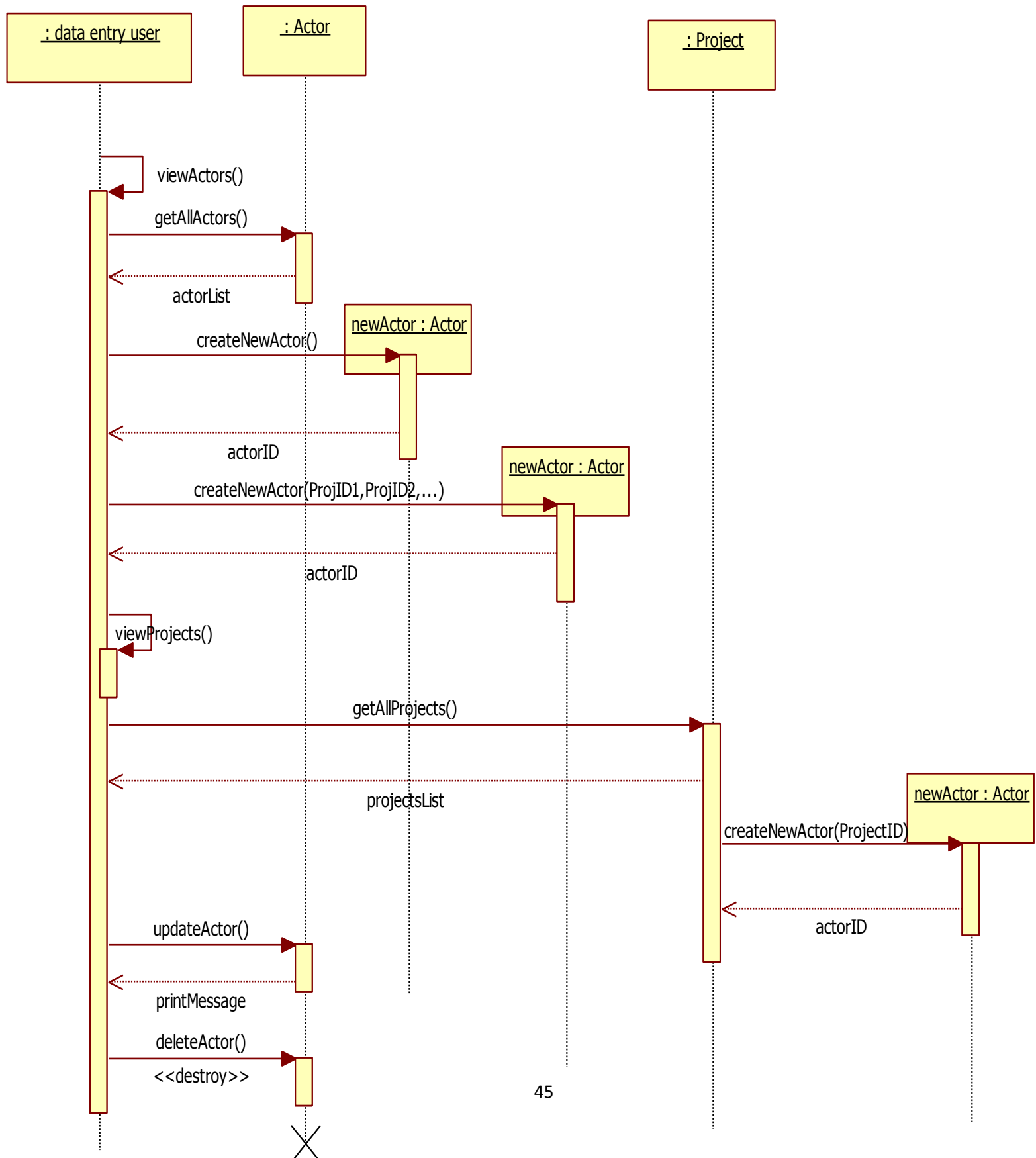
Ο υπεύθυνος χρήστης μπορεί να δει όλα τα medias τα οποία θα επιστρέψει η κλάση media μέσω της συνάρτησης getAllMedias().

Για την εισαγωγή ενός media, ο χρήστης μπορεί να δημιουργήσει νέο ανεξάρτητο media μέσω της συνάρτησης createNewMedia(), αλλά και επίσης να δημιουργήσει κάποιο media αναθέτοντάς το σε κάποια projects, μέσω της createNewMedia(projID1, projID2,...). Μπορεί επίσης να δημιουργήσει ένα media όντας σε ένα project, το οποίο θα ανατεθεί αυτόματα στο project αυτό.

Για την ενημέρωση ενός media, μπορεί να καλέσει τη συνάρτηση updateMedia(), στην οποία θα δώσει και όλα τα δεδομένα για ενημέρωση.

Για την διαγραφή ενός media, μπορεί να καλέσει την συνάρτηση deleteMedia(), η οποία θα διαγράψει το media.

Data entry user – Actor:



Στο πιο πάνω διάγραμμα περιγράφονται οι βασικές λειτουργίες, που μπορεί να κάνει ένας διαχειριστής actor (data entry χρήστης) για την εισαγωγή, ενημέρωση και διαγραφή ενός actor.

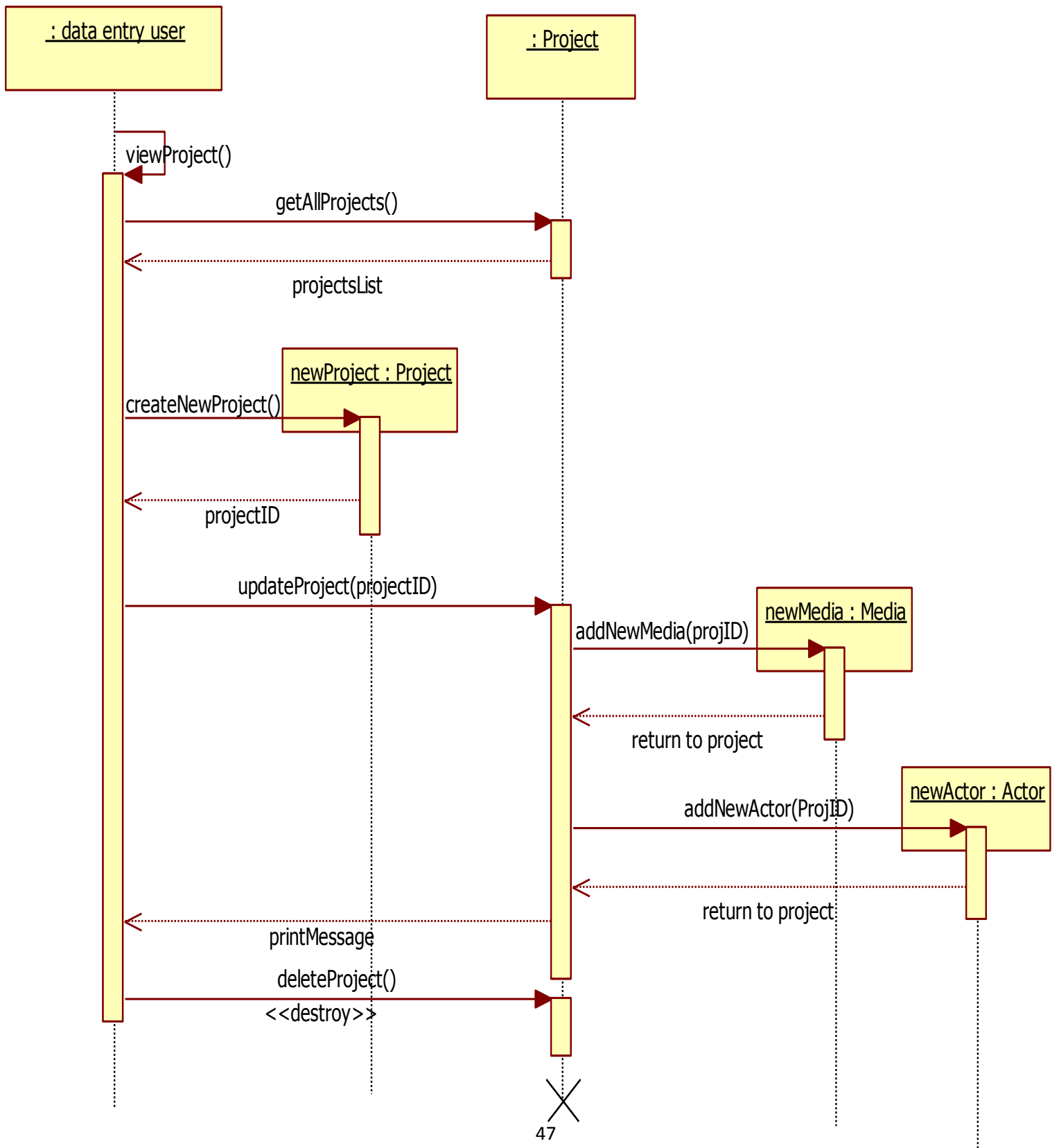
Ο υπεύθυνος χρήστης μπορεί να δει όλους τους actors, τους οποίους θα επιστρέψει η κλάση Actor, μέσω της συνάρτησης getAllActors().

Για την εισαγωγή ενός actor, ο χρήστης μπορεί να δημιουργήσει νέο ανεξάρτητο actor, μέσω της συνάρτησης createNewActor(), αλλά και να δημιουργήσει επίσης κάποιον actor αναθέτοντάς τον σε κάποια projects, μέσω της createNewActor(projID1, projID2,...). Μπορεί επίσης να δημιουργήσει έναν actor όντας σε ένα project, το οποίο θα ανατεθεί αυτόματα στο project αυτό.

Για την ενημέρωση ενός actor μπορεί να καλέσει την συνάρτηση updateActor(), στην οποία θα δώσει και όλα τα δεδομένα για ενημέρωση.

Για τη διαγραφή ενός actor μπορεί να καλέσει την συνάρτηση deleteActor(), η οποία θα διαγράψει τον actor.

Data entry user – Project:



Στο πιο πάνω διάγραμμα περιγράφονται οι βασικές λειτουργίες, που μπορεί να κάνει ένας διαχειριστής project (data entry χρήστης) για την εισαγωγή, ενημέρωση και διαγραφή ενός actor.

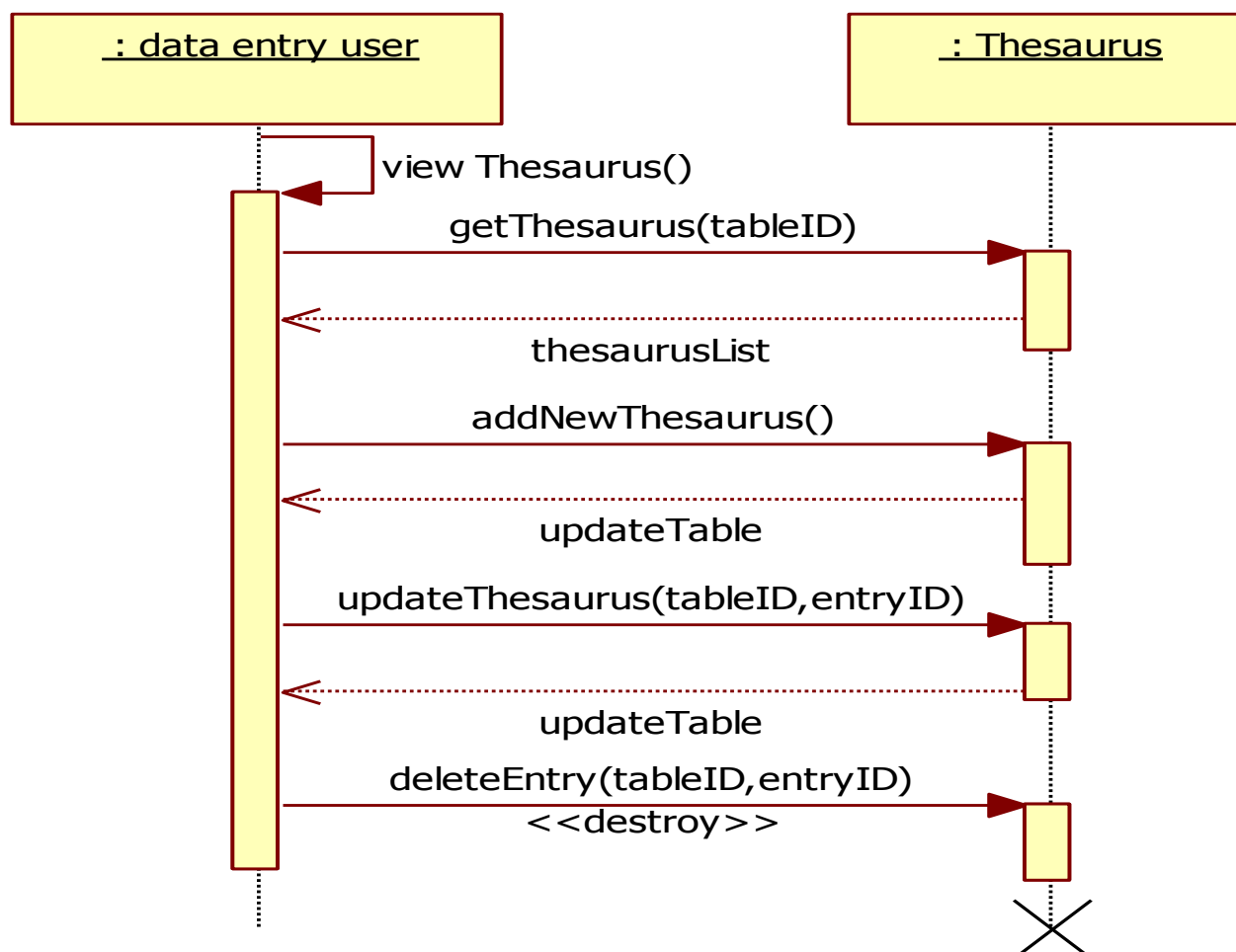
Ο υπεύθυνος χρήστης μπορεί να δει όλα τα project, τα οποία θα επιστρέψει η κλάση Project, μέσω της συνάρτησης getAllProjects().

Για την εισαγωγή ενός project, ο χρήστης μπορεί να δημιουργήσει νέο project, μέσω της συνάρτησης createNewProject().

Για την ενημέρωση ενός project, μπορεί να καλέσει τη συνάρτηση updateProject (projectID), στην οποία θα δώσει και όλα τα δεδομένα για ενημέρωση. Στη φάση ενημέρωσης του project, μπορεί επίσης να προσθέσει νέο actor ή media.

Για την διαγραφή ενός project, μπορεί να καλέσει την συνάρτηση deleteProject(), η οποία θα διαγράψει το project.

Data entry user – Thesaurus:

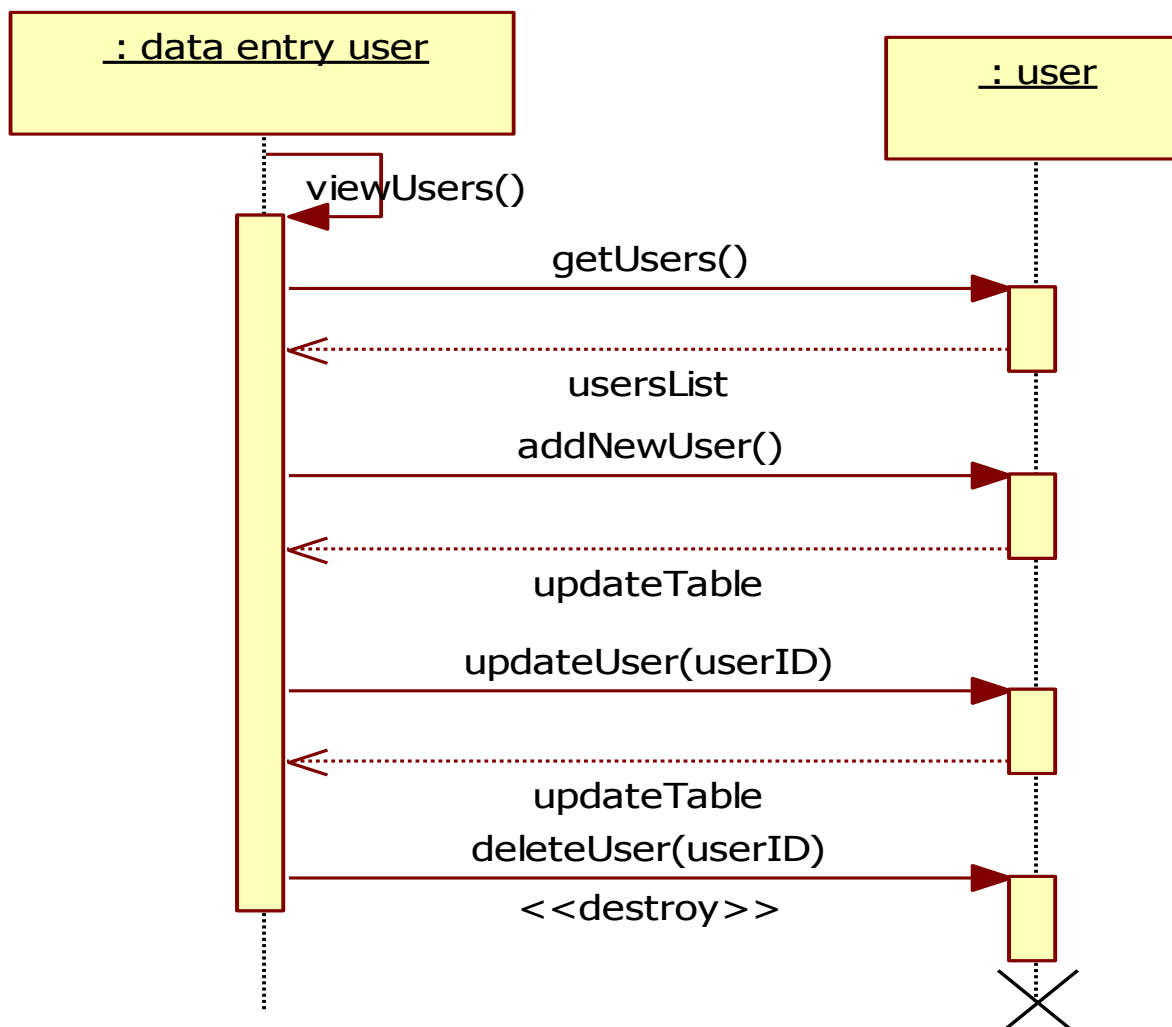


Στο πιο πάνω διάγραμμα περιγράφονται οι βασικές λειτουργίες, που μπορεί να κάνει ένας data entry χρήστης για την εμφάνιση, εισαγωγή, ενημέρωση και διαγραφή ενός thesaurus entry.

Έχοντας το tableID μπορείς να δεις ένα thesaurus table.

Το update ή delete ενός thesaurus entry, μπορεί να γίνει βλέποντας το αντίστοιχο table του και έχοντας το entryID του.

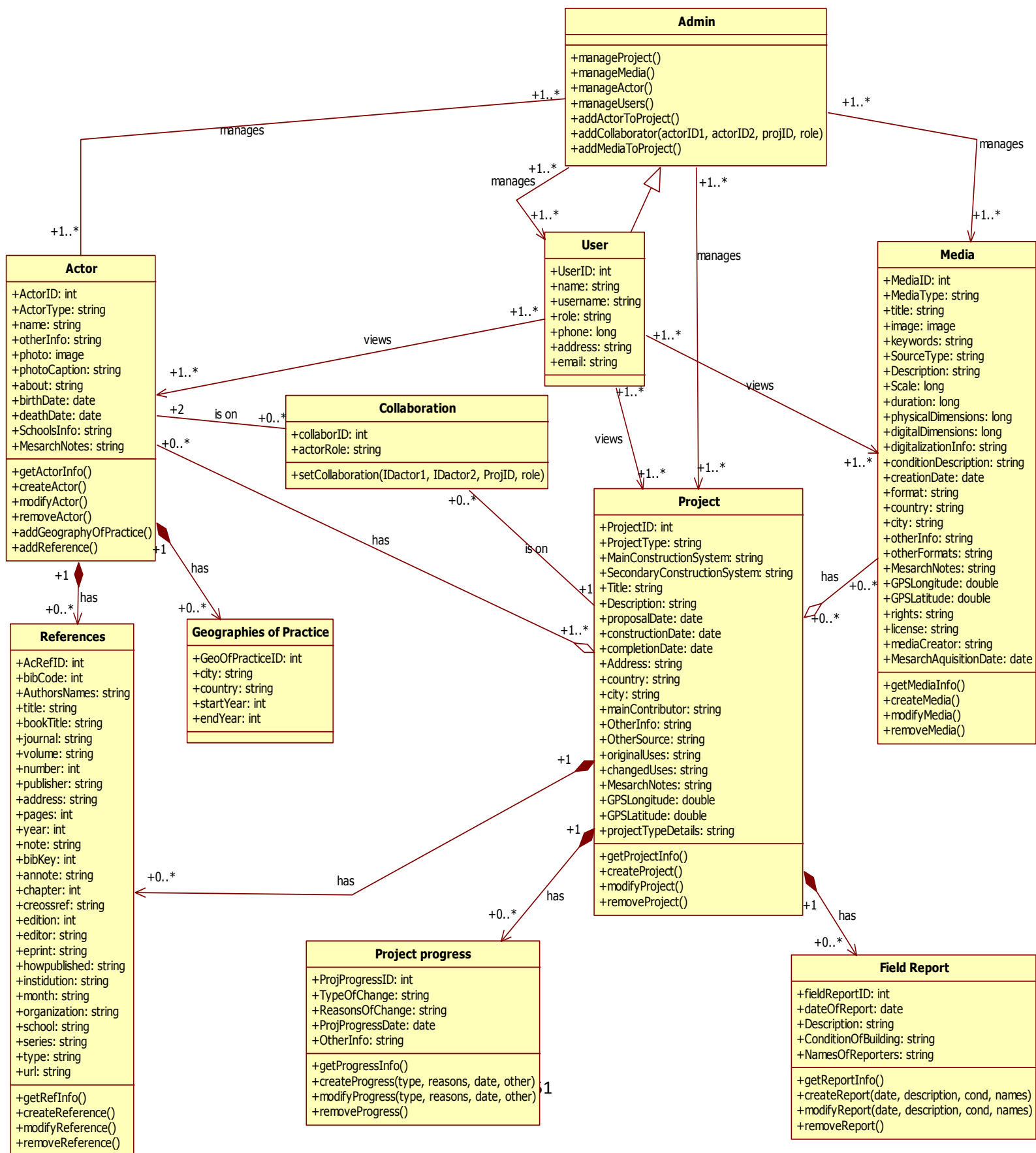
Admin – User:



Στο πιο πάνω διάγραμμα περιγράφονται οι βασικές λειτουργίες, που μπορεί να κάνει ένας data entry χρήστης, για την εμφάνιση, εισαγωγή, ενημέρωση και διαγραφή ενός user.

Για να κάνεις update ή delete ένα user, μπορείς να το κάνεις, ενώ βλέπεις τους users και έχοντας το userID του.

3.4 Λεπτομερή Σχεδίαση (Class Diagrams)



Το πιο πάνω διάγραμμα αποτελεί το class διάγραμμα του συστήματος Mesarch. Δείχνει αφαιρετικά τις διάφορες κλάσεις που μπορεί να έχει το σύστημα μαζί με τα χαρακτηριστικά και τις συναρτήσεις τους.

Το πιο πάνω διάγραμμα δείχνει επίσης τις σχέσεις που μπορεί να έχουν η μια κλάση με την άλλη:

- Ένα Project μπορεί να έχει πολλά Field reports, αλλά ένα συγκεκριμένο Field report ανήκει σε ένα μόνο Actor.
- Ένα Project μπορεί να έχει πολλούς Actors, αλλά και ένας Actor μπορεί να ανήκει σε πολλά Projects.
- Ένας Actor μπορεί να έχει πολλά Geographies of practice, αλλά ένα συγκεκριμένο Geography of practice ανήκει σε ένα μόνο Actor.
- Ένας Actor μπορεί να έχει πολλά References, αλλά ένα συγκεκριμένο Reference ανήκει σε ένα μόνο Actor.
- Ένα collaboration σε ένα Project γίνεται έχοντας 2 Actors και θέτοντας το είδος (actor Role) του collaboration τους. Όμως ένας Actor μπορεί να έχει πολλά collaborations με διάφορους Actors σε διάφορα Projects.
- Ένας Project μπορεί να έχει πολλά References, αλλά ένα συγκεκριμένο Reference ανήκει σε ένα μόνο Project.
- Ένα Project μπορεί να έχει πολλά Media, αλλά και ένα Media μπορεί να ανήκει σε πολλά Projects.
- Ένας υπεύθυνος του συστήματος (Administrator) μπορεί να διαχειριστεί όλα τα tables και να τα αλλάξει, εφόσον μπορεί να διαχειριστεί τις βασικές κλάσεις Project, Actor, Media, Users.
- Ένας User μπορεί να δει πληροφορίες από τα tables.

3.5 Συστατικά Συστήματος

Το σύστημα Mesarch έχει αρκετές κλάσεις, αφού η ιδέα της asp.NET είναι η δημιουργία κλάσεων για κάθε νέα σελίδα. Όμως μπορούμε να ξεχωρίσουμε κάποιες κλάσεις-συστατικά στο σύστημα, που τα χρησιμοποιούν άλλες κλάσεις και που οι χρήση τους είναι πιο γενική από τα άλλα.

Μπορούμε να πούμε ότι το βασικό template του συστήματος χρησιμοποιείται από όλες τις σελίδες της ιστοσελίδας, οι οποίες κληρονομούν από το βασικό Master Page.

Εκτός από το βασικό Master Page, έχουμε και άλλα Master Pages, τα οποία κληρονομούν μεν από το βασικό, το επεκτείνουν και κληρονομούνται και από μια ομάδα από άλλες σελίδες. Αυτά είναι το Search Master Page, Users Master Page, Admin Web From, Master Actor, Master Project, και το Admin Master Page (για τα thesaurus).

Παράλληλα έχουμε μια κλάση Global, η οποία είναι static, δηλαδή μπορείς να καλέσεις τις συναρτήσεις και χαρακτηριστικά της, χωρίς να δημιουργήσεις αντικείμενο. Σε αυτήν φυλάμε καταστάσεις, όπως το όνομα του χρήστη, ο οποίος έκανε Login στο σύστημα, καθώς επίσης και τον ρόλο του χρήστη (admin, data entry user, κλπ).

Μια άλλη κλάση, που είναι γενικής μορφής, είναι η κλάση Logoff, την οποία καλούν όλες οι σελίδες στέλνοντάς της το Absolute URL τους. Αυτή αποφασίζει αν θα κάνει ανακατεύθυνση (redirect) στην κύρια σελίδα (homepage) ή θα μείνει στην ίδια σελίδα, αποσυνδέοντας το χρήστη (logout).

Υπάρχουν και κάποιες βοηθητικές κλάσεις, οι οποίες βοηθούν τη φόρτωση του περιεχομένου μιας ιστοσελίδας, ασύγχρονα (χωρίς πλήρη επαναφόρτιση της παρούσας σελίδας - AJAX). Οι κλάσεις αυτές καλούνται μέσω Javascript/AJAX κώδικα, δίνοντας τους κάποιες παραμέτρους, όπως κάποια ταυτότητα κάποιου project. Αυτές οι κλάσεις λοιπόν, αναλαμβάνουν να αλλάξουν το περιεχόμενο ενός <div>tag, ανάλογα με τις παραμέτρους που παίρνουν.

Οι κλάσεις αυτές είναι:

DescriptionActorHelper, descriptionFieldReportHelper, descriptionImageHelper, descriptionMediaHelper, descriptionMesarchLabHelper, descriptionProjectHelper, descriptionReferenceHelper, filterImageHelper, leftRightBtns, searchHelper και mediaImageHelper.

3.6 Προδιαγραφές Συστήματος

Πιο κάτω, περιγράφονται λειτουργικές και μη λειτουργικές απαιτήσεις, καθώς και οποιεσδήποτε άλλες απαιτήσεις υπάρχουν.

Υπάρχουν κάποιες στατικές και κάποιες δυναμικές απαιτήσεις επιδόσεων, όσον αφορά το λογισμικό μας.

Για τις στατικές απαιτήσεις του συστήματος, θα χρειαστεί να αναφέρουμε ότι πρέπει να υποστηρίζει ταυτόχρονη χρήση πολλών χρηστών (χιλιάδες), αφού πρόκειται για μια ιστοσελίδα με αρκετούς χρήστες σε παγκόσμιο επίπεδο. Όσον αφορά τον τύπο πληροφοριών, που θα διαχειρίζεται το σύστημα, θα είναι περισσότερο χαρακτήρες και εικόνες (medias και actors).

Μία από τις δυναμικές απαιτήσεις του συστήματός είναι ο αριθμός των αιτήσεων και δεδομένων, που πρέπει να επεξεργάζεται σε μια συγκεκριμένη χρονική περίοδο. Η εφαρμογή μας θα πρέπει να επεξεργάζεται γρήγορα και άμεσα, σε πολύ μικρό χρονικό διάστημα (λιγότερο από 5 δευτερόλεπτα), τα αιτήματα των χρηστών, αφού πρόκειται για ένα σύστημα, που διευκολύνει τον χρήστη, τόσο στην αναζήτηση κάποιας οντότητας του συστήματος, όσο και στην είσοδο δεδομένων για δημιουργία κάποιας οντότητας (project, media, actor, user, thesaurus entry).

3.6.1 Περιορισμοί σχεδιασμού

- Η εφαρμογή να μπορεί να λειτουργεί σε οποιονδήποτε φυλλομετρητή και ειδικά στους τρεις βασικούς (Internet Explorer, Google Chrome, Mozilla Fire fox, Safari).
- Να παρέχει αξιόπιστες πληροφορίες και η διεπαφή του χρήστη με την εφαρμογή , να είναι εύκολη ως προς την κατανόηση και τη χρήση.
- Να είναι αποδοτικό, δηλαδή να παρέχει τις πληροφορίες σε μικρό χρόνο απόκρισης (λιγότερο από 5 δευτερόλεπτα).
- Οι εγγραφές νέων χρηστών να γίνονται με ευκολία, χωρίς πολλές πληροφορίες.
- Η είσοδος δεδομένων για τη δημιουργία των οντοτήτων, θα πρέπει να γίνεται όσο πιο γρήγορα και απλά γίνεται.
- Τέλος η εξαγωγή πληροφοριών, θα πρέπει να γίνεται γρήγορα, απλά και καινοτόμα, δηλαδή η εμφάνιση των εικόνων ενός project, media, actor και των πληροφοριών του, να γίνονται γρήγορα.

3.6.2 Χαρακτηριστικά του συστήματος λογισμικού

Τα χαρακτηριστικά του συστήματος λογισμικού, που θα αναφέρουμε πιο κάτω μπορούν να θεωρηθούν και σαν απαιτήσεις του συστήματος.

3.6.3 Αξιοπιστία (Reliability)

Η εφαρμογή πρέπει να είναι όσο το δυνατό πιο αξιόπιστη. Για να επιτευχθεί αυτό πρέπει με την εισαγωγή οποιωνδήποτε δεδομένων και εγγραφών για τη δημιουργία μιας οντότητας(project, media, actor, user, thesaurus entry) να γίνονται οι απαραίτητοι έλεγχοι να τη σωστή εισαγωγή τους. Αν για παράδειγμα ο κωδικός κάποιου χρήστη πρέπει να αποτελείται από χαρακτήρες και αριθμούς, καθώς επίσης να έχει το λιγότερο 5 χαρακτήρες

θα πρέπει να γίνεται έλεγχος ότι εισάχθηκε με τους σωστούς όρους ο κωδικός. Ακόμα αν κάπου πρέπει να δοθεί ένας αριθμός ή μια ημερομηνία, θα πρέπει να γίνεται ο έλεγχος για την ορθότητα του τύπου δεδομένων, που δόθηκε.

Επιπρόσθετα, με την εισαγωγή δεδομένων η αποθήκευσή τους θα πρέπει να γίνεται στους σωστούς-κατάλληλους πίνακες της Βάσης Δεδομένων.

Τέλος, η σωστή εμφάνιση δεδομένων, στις κατάλληλες καρτέλες και πεδία, είναι ένα θέμα αξιοπιστίας της ιστοσελίδας, καθώς πρέπει να δείχνει τα κατάλληλα δεδομένα στα κατάλληλα πεδία.

3.6.4 Διαθεσιμότητα (Availability)

Η διαθεσιμότητα, που πρέπει να παρέχει το σύστημα στους χρήστες για προβολή και επεξεργασία των δεδομένων είναι αυστηρή. Απαγορεύεται να χαθεί η οποιαδήποτε πληροφορία.

Για τη διασφάλιση της διαθεσιμότητας της πληροφορίας, πρέπει να τηρηθούν οι πιο κάτω σημαντικές παράμετροι:

Φύλαξη δεδομένων σε εφεδρικά αντίγραφα ασφάλειας. Σε περίπτωση σφάλματος ή κατάρρευσης του συστήματος, η φύλαξη δεδομένων σε εφεδρικά αντίγραφα ασφαλείας προσφέρει υποστήριξη και γρήγορη αποκατάσταση της κανονικής λειτουργίας του συστήματος.

Χρήση τεχνολογιών, που να αυξάνουν τη διαθεσιμότητα των δεδομένων, καθώς και τη μείωση σφαλμάτων.

Άρα η διαθεσιμότητα των πληροφοριών είναι δεδομένη ανά πάσα στιγμή προς το χρήστη, με οποιαδήποτε αλλαγή γίνει στο σύστημα, είτε προσθήκη νέων λειτουργιών, είτε με αναβάθμιση υφιστάμενων.

3.6.5 Ασφάλεια (Security)

Ο κώδικας της ιστοσελίδας θα πρέπει να περιέχει ελέγχους, που θα προστατεύουν την βάση δεδομένων από SQL injection attacks. Αυτού του είδους οι επιθέσεις, προσπαθούν να εκμεταλλευτούν αδυναμίες της SQL και μέσω queries από το login form ή από τα query strings των url, προσπαθούν να υποκλέψουν δεδομένα του συστήματος ή και να το καταστρέψουν. Τόσο η βάση όσο και το script, στο οποίο θα είναι γραμμένη η ιστοσελίδα, πρέπει να συμπεριλάβουν ελέγχους, που θα αποτρέπουν αυτού του είδους τις επιθέσεις. Επίσης πρέπει να ελέγχεται η πιθανότητα να μην μπορεί κάποιος, είτε από λάθος, είτε επίτηδες να δημιουργεί πολλά προφίλ, διότι αυτό εμπεριέχει τον κίνδυνο δημιουργίας ανεπιθύμητου μεγέθους πίνακα, με αποτέλεσμα τη μείωση απόδοσης του συστήματος.. Επίσης ο κωδικός και το user name, πρέπει να κρυπτογραφούνται κατά την αποστολή τους από τον χρήστη στο server, έτσι ώστε να μην μπορεί κάποιος να τους διαβάσει, μέσω snooping. Τέλος ο server θα πρέπει να είναι σε θέση να αναγνωρίζει τις απόπειρες για σπάσιμο κωδικών, όπως brute force attacks, δηλ. πολλές συνεχόμενες προσπάθειες για login, μέσα σε σύντομο χρονικό διάστημα.

3.6.6 Συντηρησιμότητα (Maintainability)

Η σχεδίαση της βάσης δεδομένων στη μορφή τουλάχιστον 3NF, θα βοηθήσει στη διατήρηση της ακεραιότητας και της ορθότητας των δεδομένων, που βρίσκονται στη βάση. Ο λόγος είναι ότι δεν θα προκαλούνται ανωμαλίες κατά την εισαγωγή/διαγραφή δεδομένων στη βάση. Ο κώδικας θα βρίσκεται σε ένα SVN repository, έτσι ώστε να φαίνονται οι αλλαγές σε κάθε έκδοση και να μπορούμε να θυμόμαστε τι έχει γίνει σε κάθε έκδοση του κώδικα. Σε περίπτωση που κάτι πάει στραβά, με την νέα έκδοση του κώδικα, θα μπορούμε να γυρίσουμε την ιστοσελίδα στην προηγούμενη σταθερή έκδοση. Ο κώδικας πρέπει να γραφτεί βάση όλων των αρχών προγραμματισμού, μερικές από τις οποίες είναι επεξηγηματικά σχόλια, όπου χρειάζεται, σπάσιμο του κώδικα σε πολλά αρχεία κλπ. Αρχεία κώδικα με πάνω από χίλιες γραμμές είναι συνήθως δυσνόητα. Γενικά, οποιοσδήποτε είναι

λίγο σχετικός με το αντικείμενο, θα πρέπει να είναι σε θέση να διαβάσει και να κατανοήσει τον κώδικα εύκολα και να μπορεί να τον επεκτείνει.

3.6.7 Φορητότητα (Portability)

Η ιστοσελίδα θα γραφτεί σε asp.NET έτσι ώστε να έχουμε την ασφάλεια που προσφέρει η Microsoft. Θα γίνουν δοκιμές και θα ελεγχθεί ότι η ιστοσελίδα δουλεύει και εμφανίζεται σωστά, σε όλους τους διάσημους browser, οι οποίοι είναι οι Internet Explorer, Mozilla Firefox, Google Chrome και Safari. Οι browser αυτοί χρησιμοποιούνται από την μεγάλη πλειοψηφία των χρηστών, συνεπώς αν η εφαρμογή δουλεύει σωστά σε αυτούς τους browsers, είμαστε σχεδόν σίγουροι ότι όλοι οι χρήστες της ιστοσελίδας θα βλέπουν σωστά το περιεχόμενο. Επίσης αυτοί οι browser έχουν και εκδόσεις για φορητές συσκευές, άρα μπορούμε να είμαστε σίγουροι ότι ικανοποιούμε και του χρήστες φορητών συσκευών.

Κεφάλαιο 4 -Σχεδίαση Συστήματος Mesarch

- 4.1 Εισαγωγή
 - 4.2 Αρχιτεκτονική Συστήματος
 - 4.3Εργαλεία Σχεδιασμού
 - 4.3.1 Εργαλεία μοντελοποίησης
 - 4.3.2 Εργαλεία ανάπτυξης πρωτοτύπων
 - 4.4Βάση Δεδομένων
 - 4.4.1 ER diagram
 - 4.5 Διαπροσωπεία Συστήματος
 - 4.5.1 Πρωτότυπα σχεδίασης GUI
 - 4.5.2 Graphical User Interface (GUI)
-

4.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζεται η αρχιτεκτονική του συστήματος, οι τεχνολογίες, που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος. Επιπλέον, παρουσιάζονται τα εργαλεία στα οποία πραγματοποιήθηκε η σχεδίαση του συστήματος.

4.2 Αρχιτεκτονική Συστήματος

Το σύστημα αποτελείται από δύο υποσυστήματα, τη Βάση δεδομένων, και την ιστοσελίδα. Η ιστοσελίδα μπορεί να διαιρεθεί σε άλλα δύο υποσυστήματα, το σύστημα διαχείρισης δεδομένων (Administration Panel) και το σύστημα εμφάνισης δεδομένων. Η αρχιτεκτονική που χρησιμοποιείται είναι το client – server μοντέλο, όπου ως clients λειτουργούν τα δύο υποσυστήματα Administration Panel και το σύστημα εμφάνισης δεδομένων, ενώ στην

πλευρά του server, βρίσκεται ο Data base server και ο IIS, όπου στον τελευταίο φιλοξενούνται τα συστήματα Administration Panel και το σύστημα εμφάνισης δεδομένων.

Σύστημα εμφάνισης δεδομένων: Το λογισμικό το οποίο διαχειρίζεται την εμφάνιση των πληροφοριών στο φυλλομετρητή.

Σύστημα διαχείρισης δεδομένων (Administration Panel): Το λογισμικό το οποίο διαχειρίζονται οι Administrators του συστήματος. Υπάρχουν δύο είδη χρηστών, που διαχειρίζονται τα δεδομένα, οι **administrators** και οι **data entry users**.

Σχεσιακή Βάση Δεδομένων: Η σχεσιακή βάση δεδομένων χρησιμοποιείται για την αποθήκευση των δεδομένων και των δύο υποσυστημάτων. Οτιδήποτε αφορά δεδομένα των χρηστών του συστήματος, αποθηκεύονται και ανακτώνται από αυτή.

4.3 Εργαλεία Σχεδιασμού

4.3.1 Εργαλεία μοντελοποίησης

Microsoft Excel 2010

Το Microsoft Excel είναι μια εφαρμογή λογιστικών φύλλων, που αναπτύχθηκε από τη Microsoft. Διαθέτει υπολογισμό, γραφικά εργαλεία, πίνακες και άλλα. Χρησιμοποιήθηκε για τη δημιουργία του πίνακα σύγκρισης των διάφορων ιστοσελίδων και για τη δημιουργία των πινάκων του Relational Model.

StarUML

Το Star UML project είναι η δημιουργία ενός εργαλείου μοντελοποίησης λογισμικού, καθώς και πλατφόρμα που σε βοηθά να αναπτύξεις διάφορα αφηρημένα μοντέλα του συστήματος. Κάθε μοντέλο αναπαριστά διαφορετική πτυχή του συστήματος. Βασικά είναι η αναπαράσταση του συστήματος, μέσω γραφικών μοντέλων UML Unified Modeling

Language), το οποίο με βοήθησε στην υλοποίηση των use case, sequence και class διαγραμμάτων.

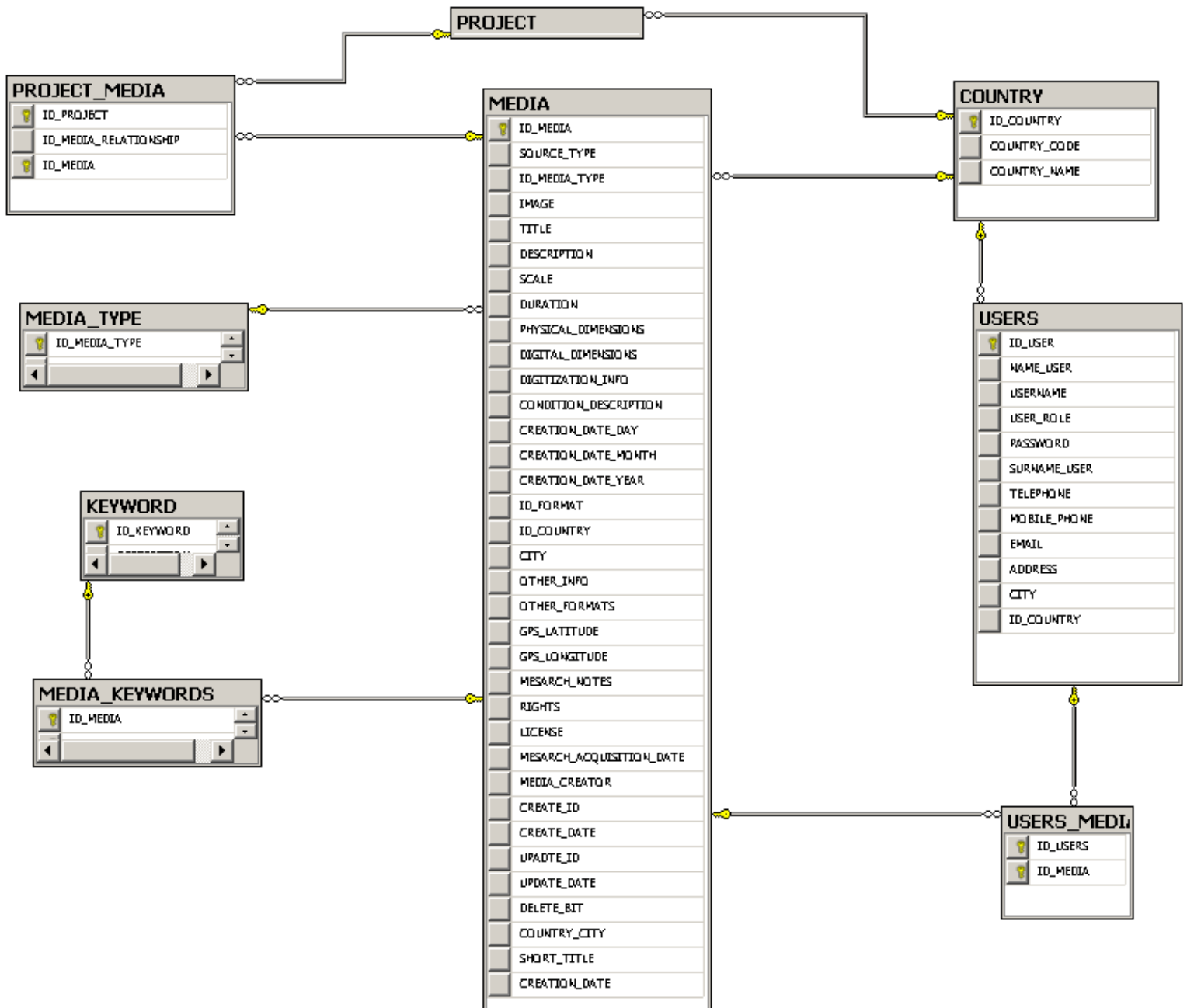
4.3.2 Εργαλεία ανάπτυξης πρωτοτύπων

Adobe programs (Adobe Flash CS4 Professional, Adobe Fireworks CS4, Adobe Dream weaver CS4): Το Adobe Dream weaver είναι ένα πρόγραμμα για σχεδιασμό ιστοσελίδων και το χρησιμοποιήθηκε για τα αρχικά πρωτότυπα, που δόθηκαν στους πελάτες για την επιλογή του βασικού template της ιστοσελίδας. Το Adobe Fireworks και Flash Professional CS4, είναι βοηθητικά εργαλεία γραφικών, τα οποία χρησιμοποιήθηκαν για την δημιουργία των λογότυπων (logos) της ιστοσελίδας, καθώς και για το σχεδιασμό ορισμένων εικόνων, όπως κάποια βέλη (arrows).

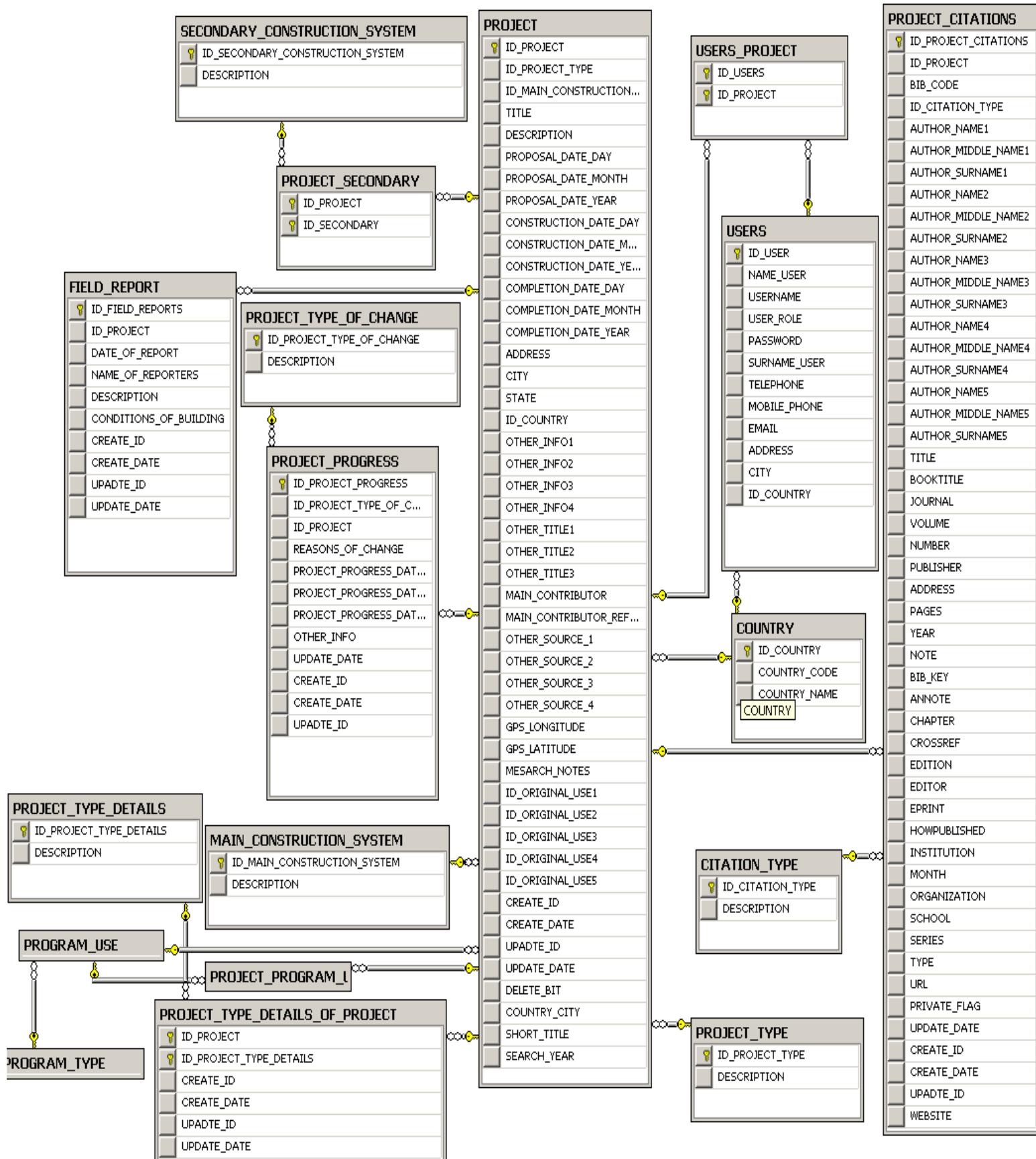
4.4 Βάση Δεδομένων

4.4.1 ER diagram

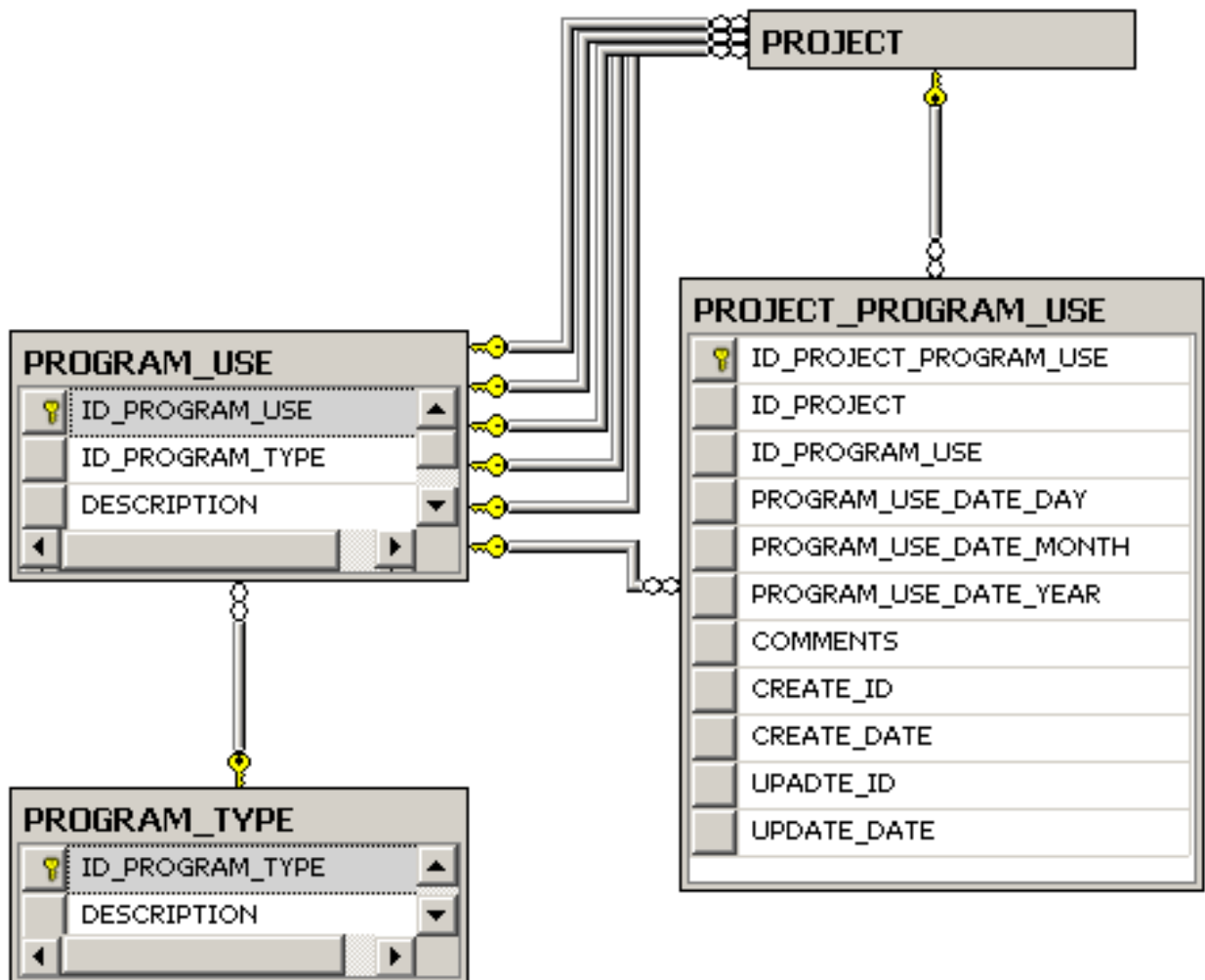
Media relations



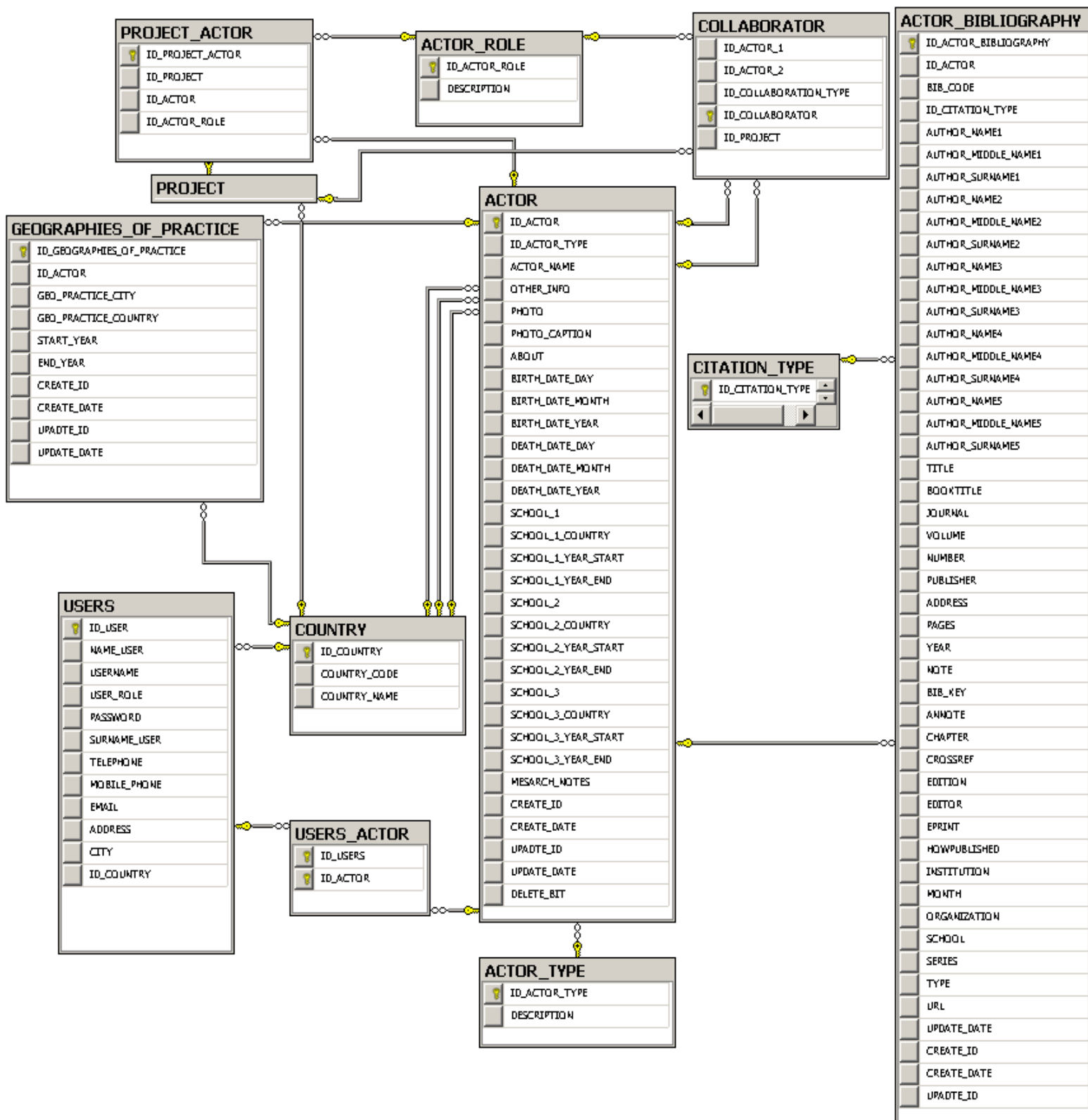
Project relations



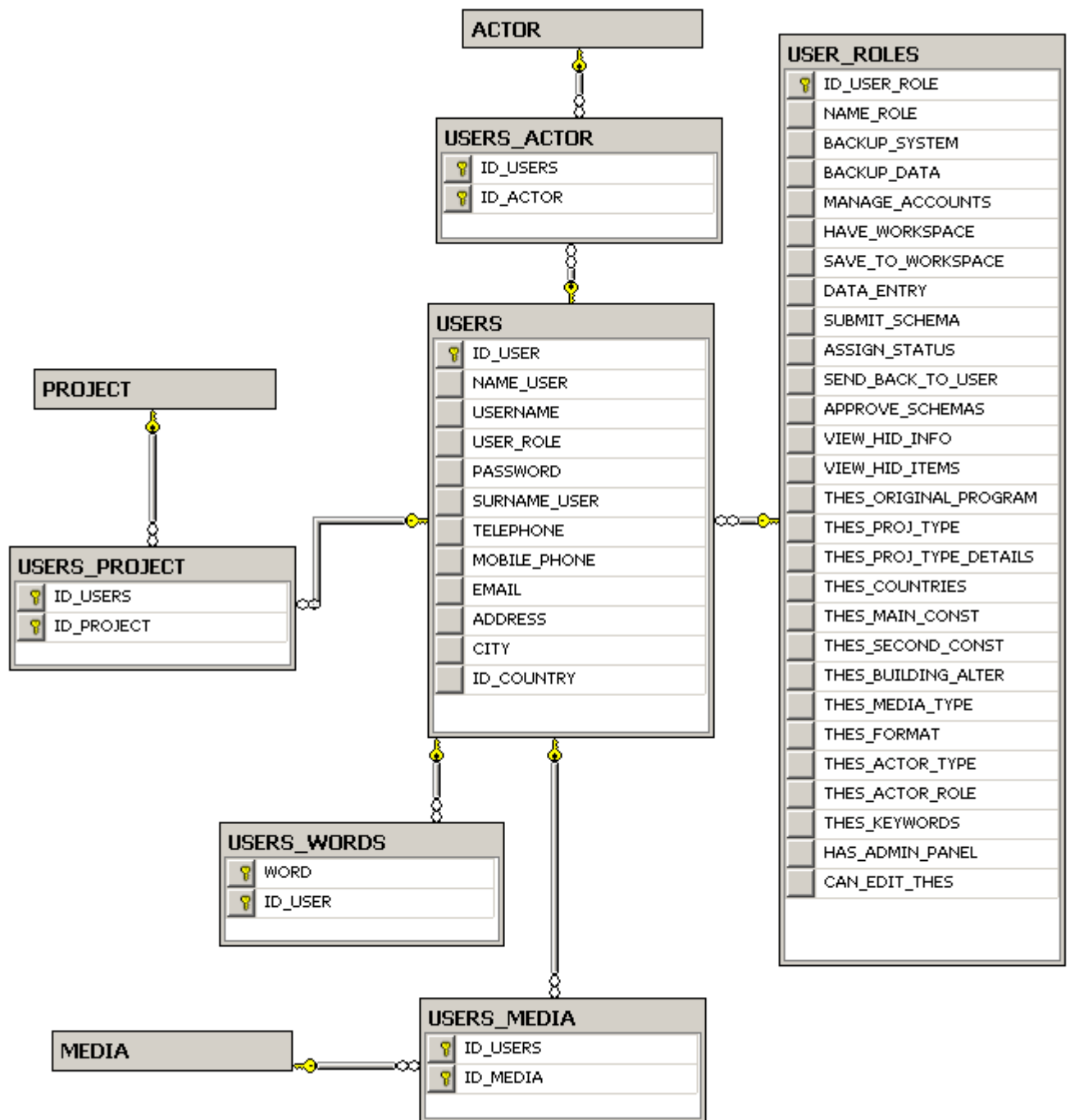
Changed use - project



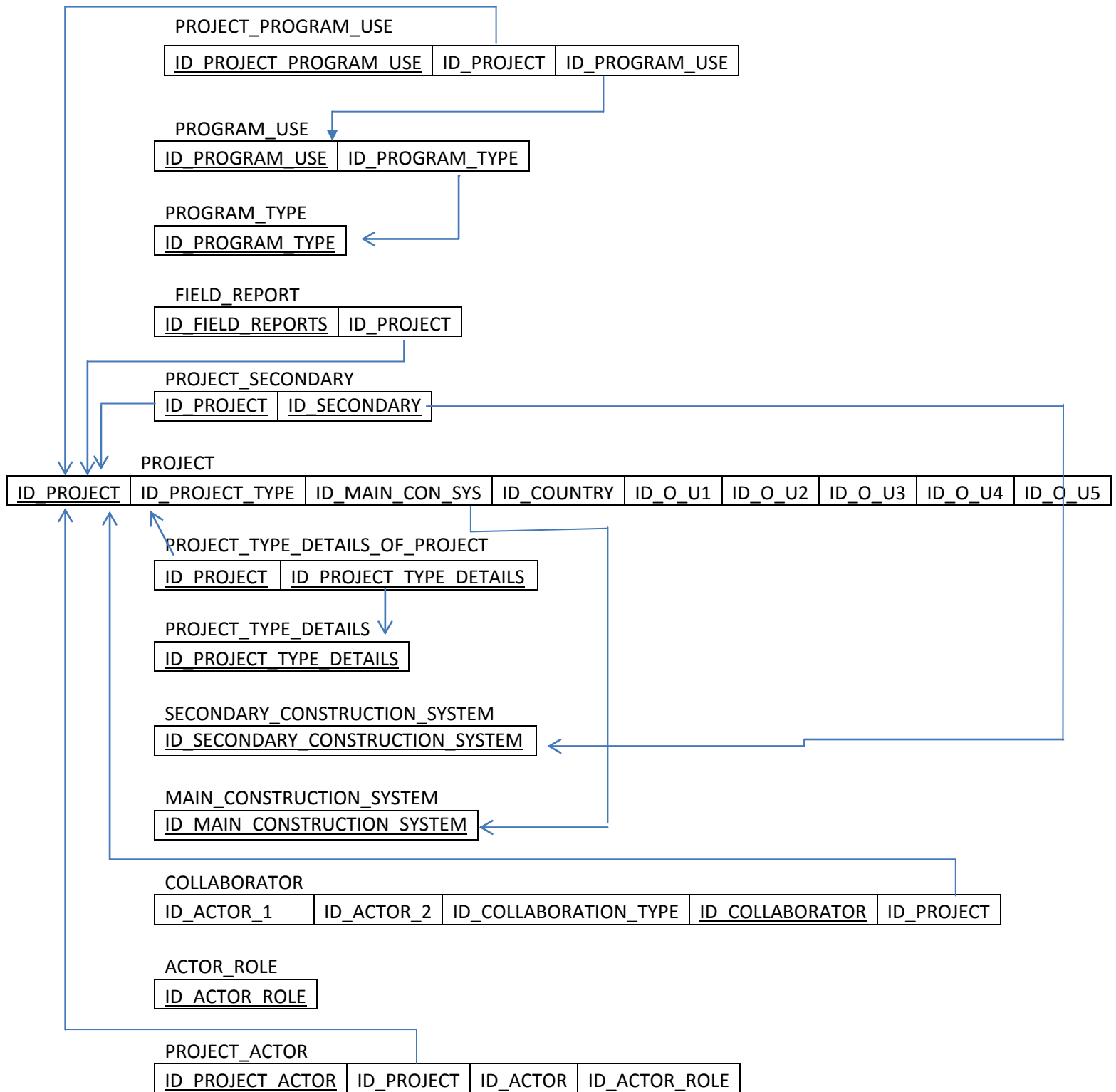
Actor relations

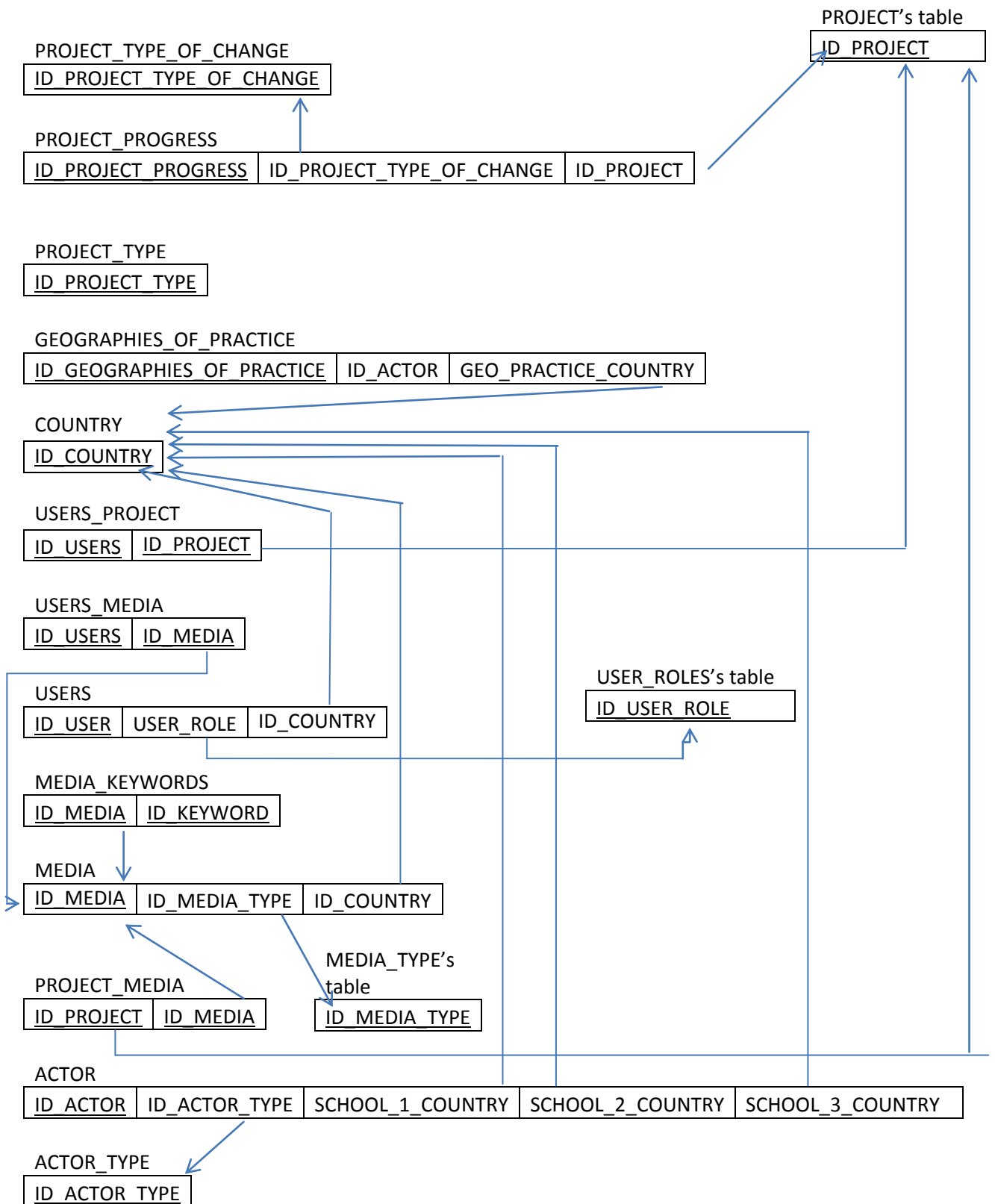


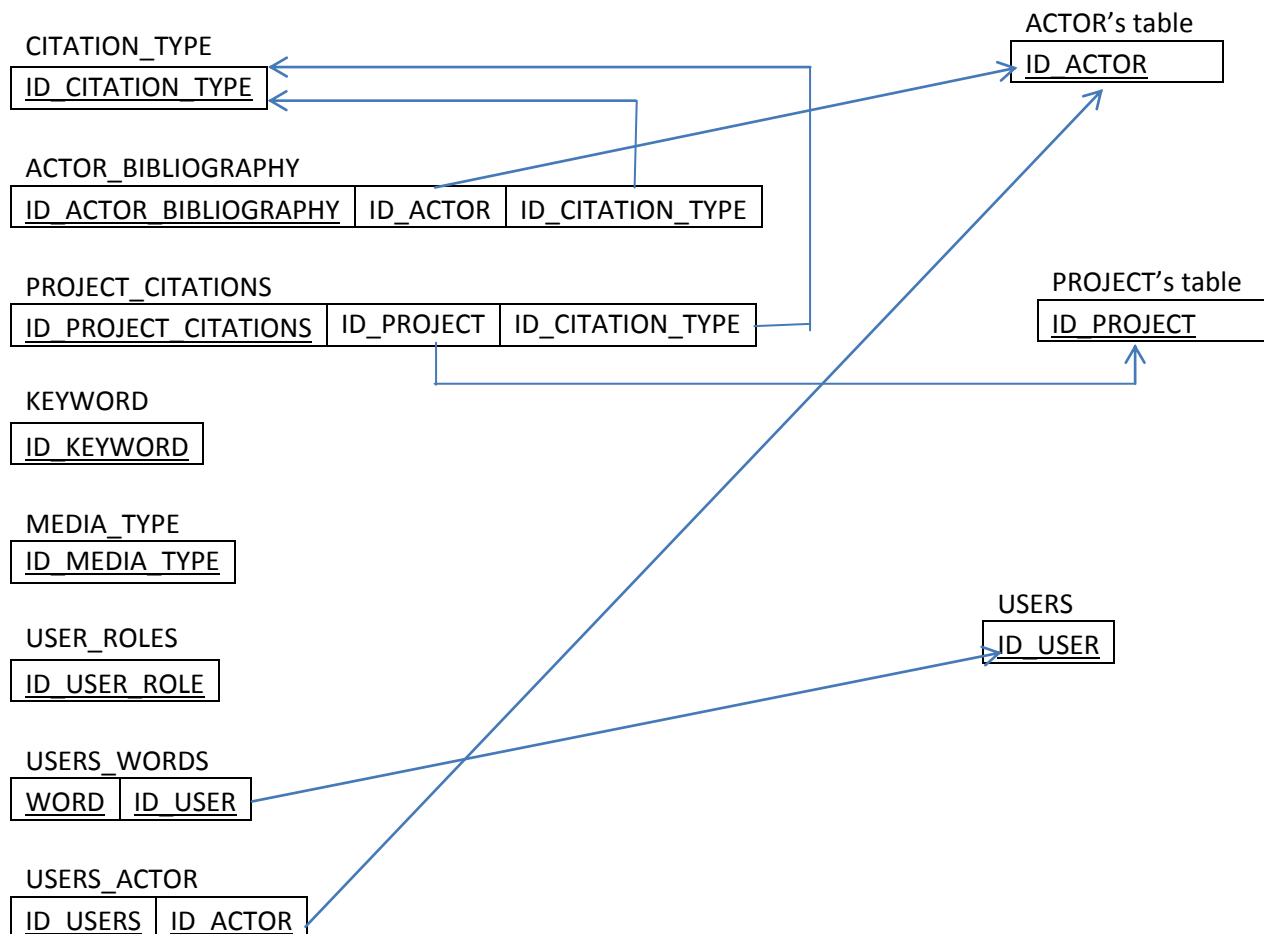
Users' relations



Relational Model – Only keys





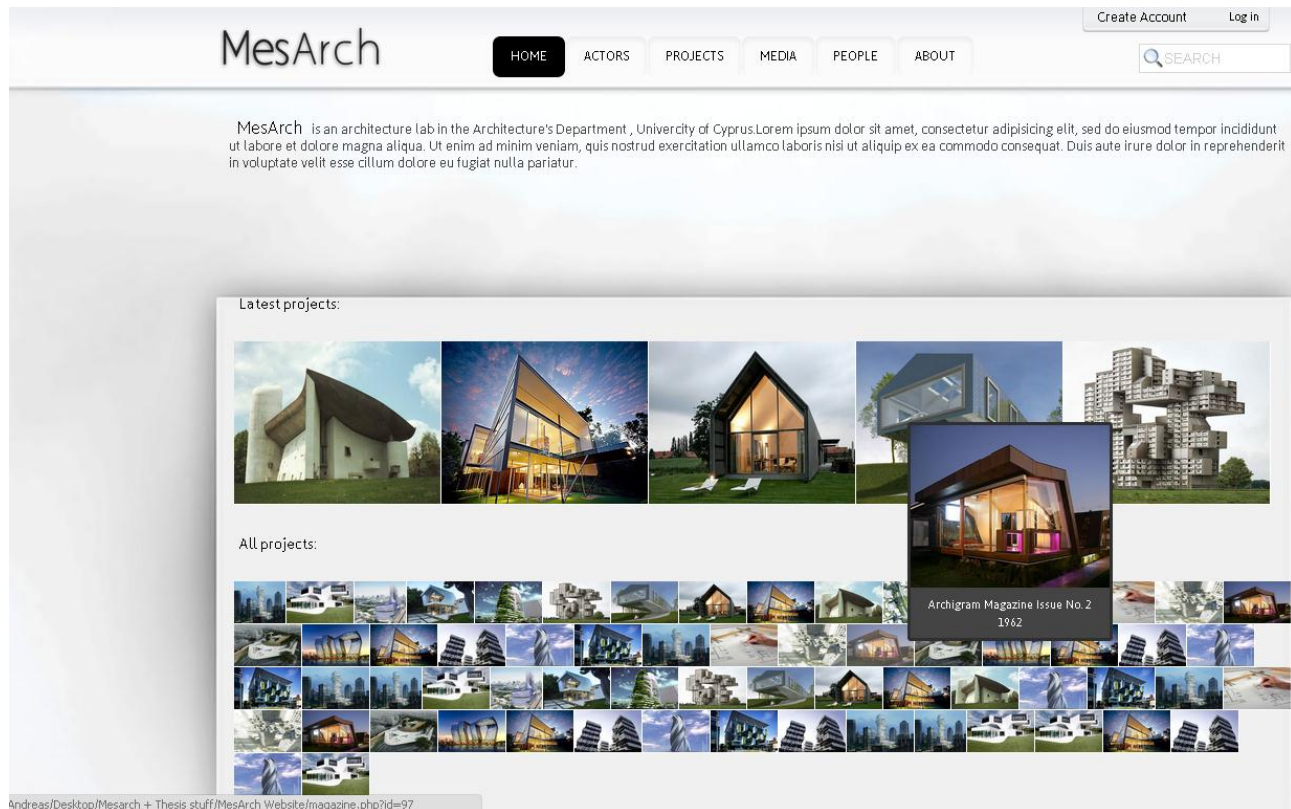


Στο πιο πάνω σχεσιακό μοντέλο παρουσιάζονται τα κύρια και δευτερεύοντα κλειδιά κάθε table, που υπάρχει στη βάση δεδομένων. Παρουσιάζονται επίσης ποια κλειδιά αναφέρουν και ποια αναφέρονται, σε κάθε σχέση των tables.

4.5 Διαπροσωπεία Συστήματος

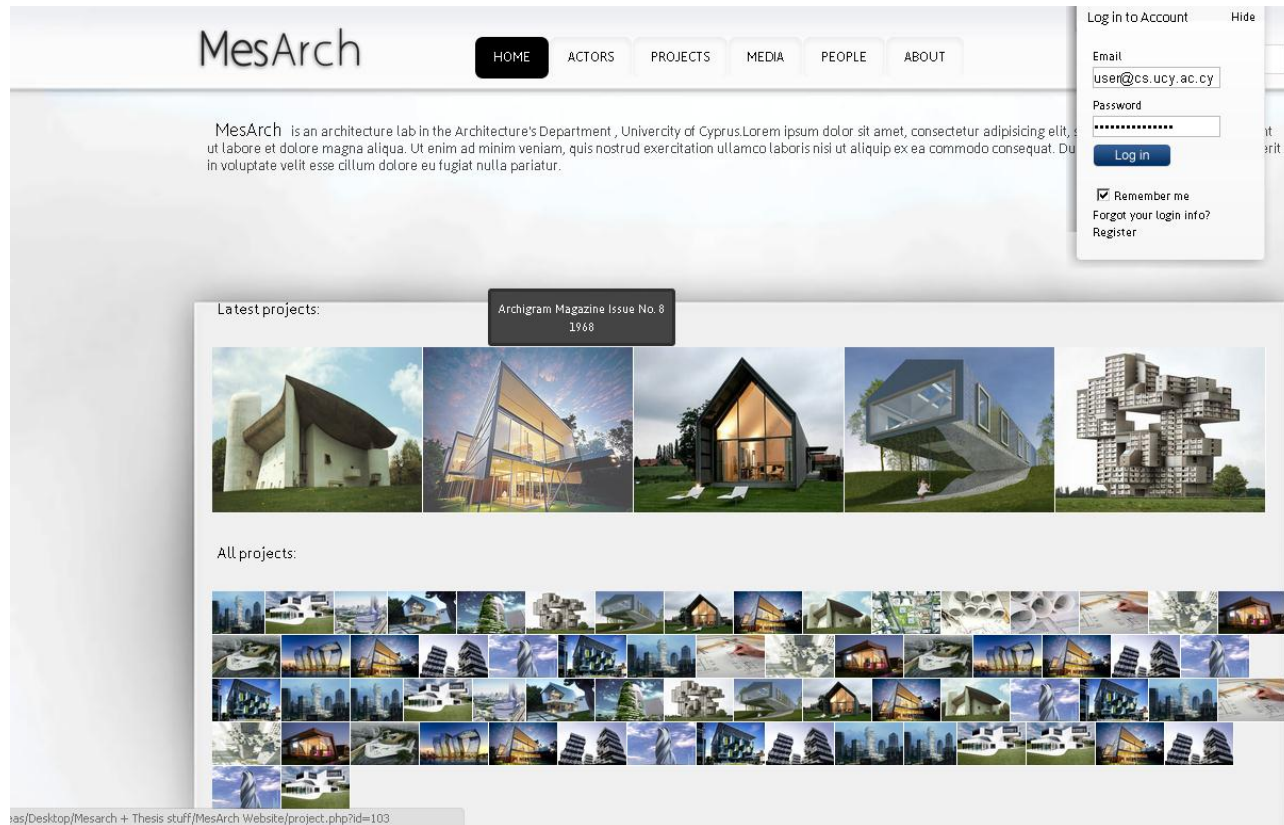
4.5.1 Πρωτότυπα σχεδίασης GUI

Πρωτοτυποποίηση αρχικής σελίδας με hover πάνω σε ένα project.



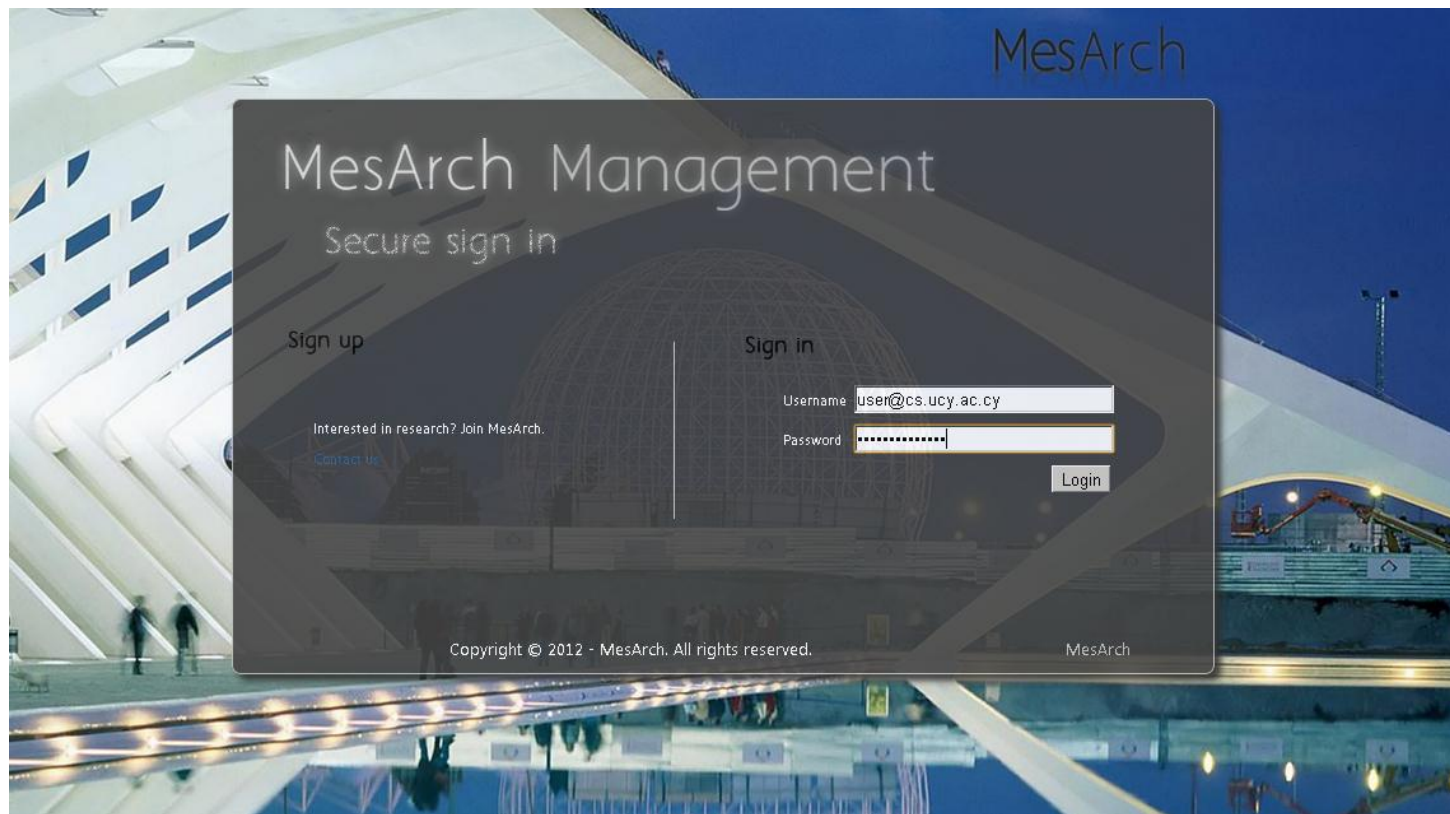
Αρχική σελίδα του πρωτοτύπου, όπου φαίνονται οι διάφορες εικόνες των project του συστήματος. Κάνοντας hover πάνω σε ένα project το μεγεθύνει γράφοντας και μερικές πληροφορίες για αυτό, όπως όνομα, αριθμός project και ημερομηνία κατασκευής. Υπάρχει επιλογή για Create Account, το οποίο σε παίρνει σε νέα σελίδα και Login, το οποίο πατώντας το, κατεβαίνει ένα panel στην ίδια σελίδα για να εισάγει τα δεδομένα του ο χρήστης. Επίσης φαίνονται στο πάνω μέρος της σελίδας και οι διάφορες ετικέτες (tabs) για περιήγηση στη σελίδα. Υπάρχει και το search input box, όπου γίνονται οι αναζητήσεις στη σελίδα.

Πρωτοτυποποίηση Λειτουργίας εξουσιοδότησης 1



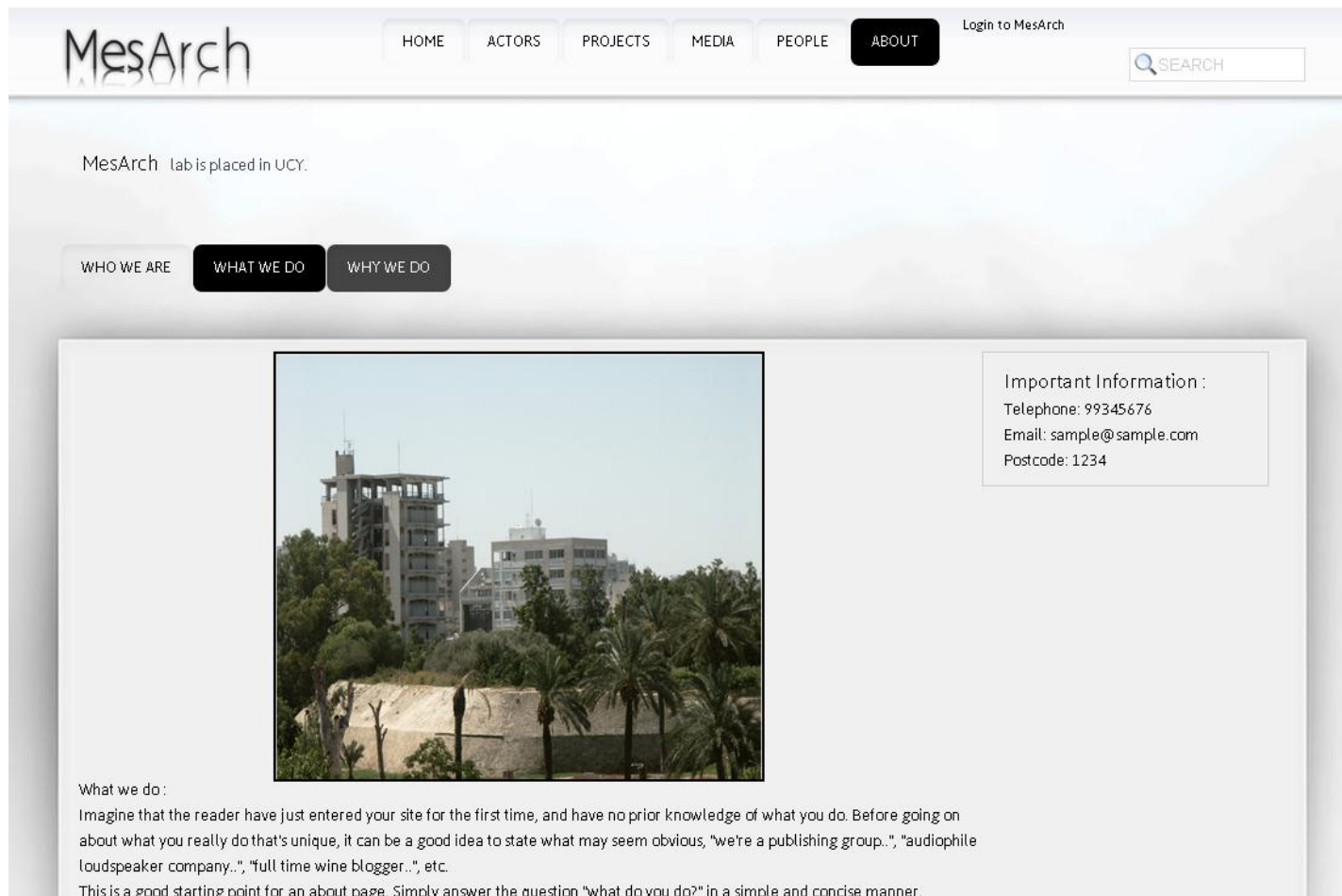
Το panel που κατεβαίνει όταν ο χρήστης πατήσει το Login. Πατώντας το Hide μπορεί να κρύψει και πάλι το panel. Πληκτρολογώντας στο email και password τα στοιχεία του, μπορεί να μπει στο σύστημα πατώντας το κουμπί Login.

Πρωτοτυποποίηση Λειτουργίας εξουσιοδότησης 2



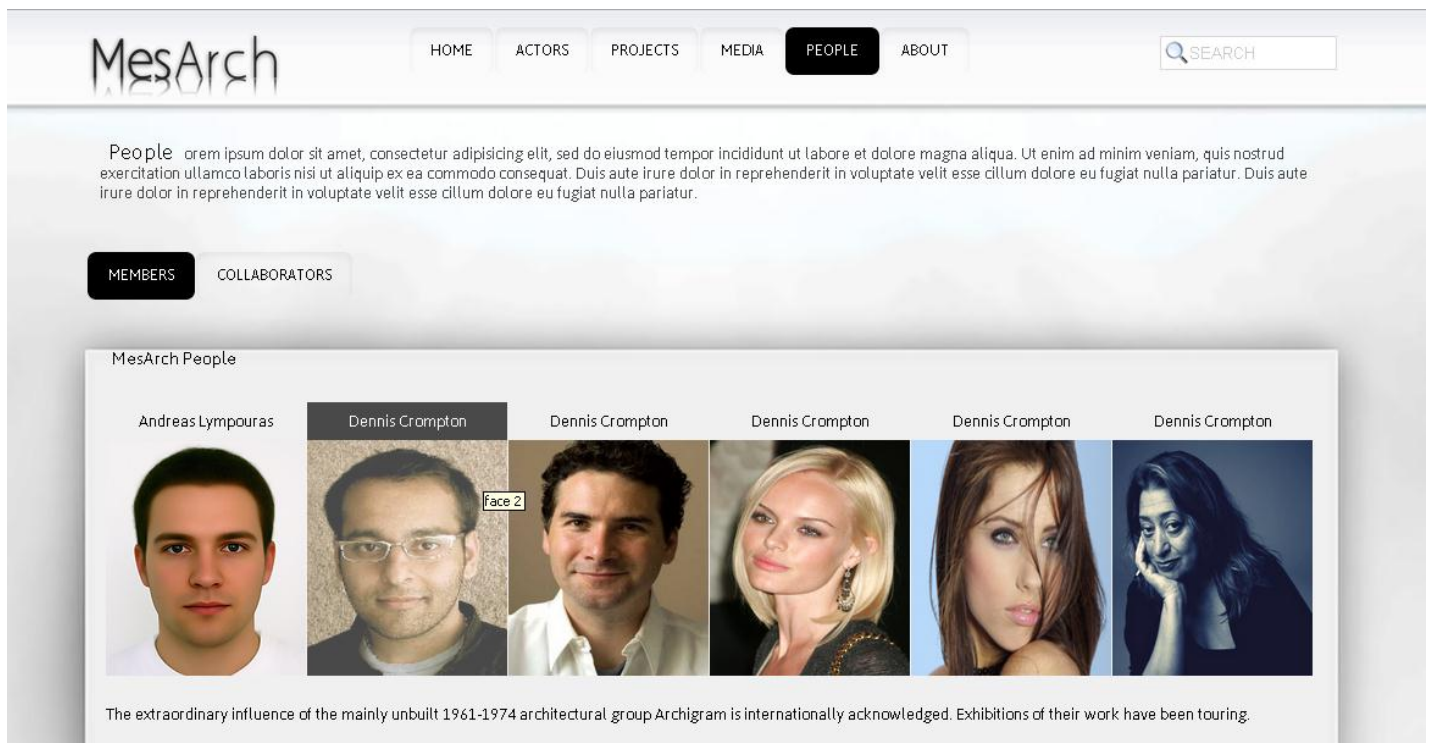
Μια δεύτερη ιδέα για το πώς να είναι η λειτουργία εξουσιοδότησης.

Πρωτοτυποποίηση για το About tab με επιλεγμένο το “What we do” και hover over το “Why we do”.



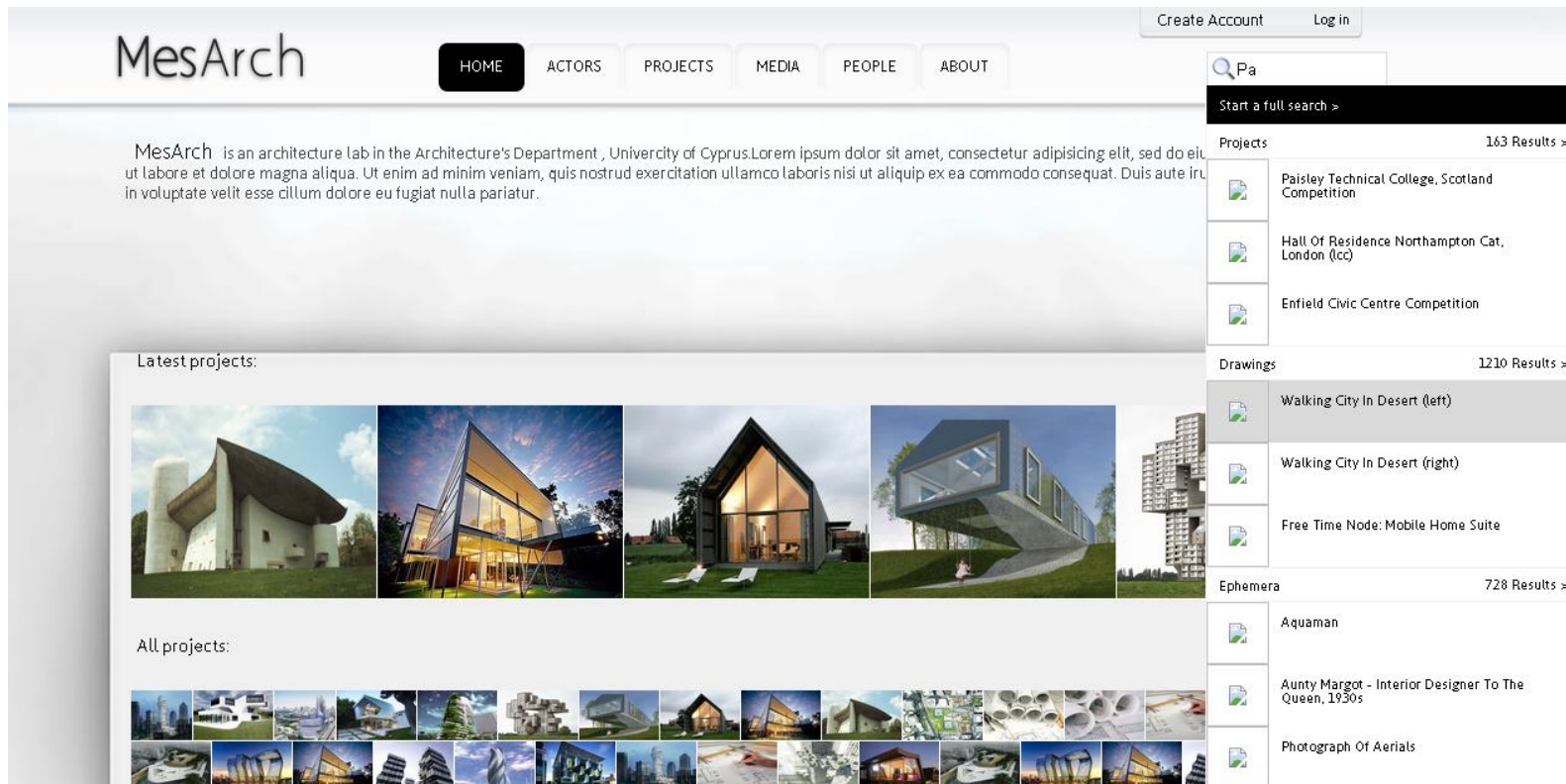
Εδώ είναι επιλεγμένη η ετικέτα (tab) About, όπου ο χρήστης μπορεί να δει διάφορες πληροφορίες για το εργαστήριο Mesarch. Υπάρχουν άλλες ετικέτες σε αυτή τη σελίδα όπως: “Who w eare”, “What we do” και “Why we do”.

Πρωτοτυποποίηση για το People tab με hover πάνω σε ένα actor.



Εδώ είναι επιλεγμένη η ετικέτα (tab) People, όπου ο χρήστης μπορεί να δει διάφορες πληροφορίες για τους ανθρώπους που είναι αποθηκευμένοι στο σύστημα, είτε ως αρχιτέκτονες, είτε ως designers κλπ. Χωρίζεται και σε άλλες ετικέτες σε αυτή τη σελίδα όπως: “Members” και “Collaborators”. Όπου members είναι οι κύριοι εμπλεκόμενοι με ένα project, ενώ οι collaborators έμμεσοι συνεργάτες, έχοντας κάποιο είδος συνεργασίας με τους άμεσους.

Πρωτοτυποποίηση για το search της εφαρμογής.



Φαίνεται ο τρόπος εμφάνισης και λειτουργίας του search. Χωρίζεται σε τρεις κατηγορίες: “Projects”, “Drawings” και “Ephemera” και δείχνει τα πρώτα τρία αποτελέσματα της αναζήτησης. Δείχνει επίσης και τα συνολικά αποτελέσματα για την κάθε κατηγορία.

4.5.2 Graphical User Interface (GUI)

Το γραφικό περιβάλλον του συστήματος και η σχεδίασή του είναι ένας από τους σημαντικότερους παράγοντες στην επιτυχία ενός λογισμικού. Το Graphical User Interface (GUI) πρέπει να είναι φιλικό, εύχρηστο αλλά και έξυπνο, έτσι ώστε πάντα να δίνει στο χρήστη την εντύπωση ότι έχει το πάνω χέρι στην πλοήγηση μέσα σε ένα λογισμικό. Πρέπει να βοηθά το χρήστη στη διεκπεραίωση των αιτημάτων του και όχι να τον καθυστερεί ή να τον συγχύζει.

Στο GUI τα κουμπιά θα πρέπει να είναι τοποθετημένα στην ίδια τοποθεσία σε κάθε φόρμα και γενικά να έχει μια αρμονία. Να έχει δηλαδή το δικό του template, έτσι ώστε ο χρήστης μέσα σε μικρό χρονικό διάστημα, να γνωρίζει που βρίσκονται τα κουμπιά. Θα ξέρει από πριν τι να περιμένει σε κάθε φόρμα, έτσι δεν θα του φαίνεται δύσκολο να χειριστεί το λογισμικό.

Ακόμη, δεν πρέπει να υπάρχουν πολλά κουμπιά ή σύνδεσμοι πάνω στη φόρμα, όσο πιο λίγα τόσο το καλύτερο για το χρήστη, ο οποίος σε πολλές περιπτώσεις μπερδεύεται. Είναι ότι καλύτερο και για το χρήστη, αλλά και για τον προγραμματιστή, καθότι κανένας από τους δυο δεν θέλει να κάνει τη ζωή του δύσκολη, πόσο μάλλον ο χρήστης, για τον οποίο προορίζεται το σύστημα.

Τέλος, αν υπάρχει ένας αρμονικός συνδυασμός χρωμάτων στο GUI θα το κάνει ακόμα πιο φιλικό και ευχάριστο. Οι υπερβολές θα κάνουν την εφαρμογή άκομψη και κουραστική και έτσι δεν θα βοηθούν το χρήστη να αποδίδει καλύτερα.

Κεφάλαιο 5- Υλοποίηση Συστήματος Mesarch

5.1 Εισαγωγή

5.2 Εργαλεία Υλοποίησης

5.2.1 Τεχνολογίες

5.2.2 Εργαλεία ανάπτυξης λογισμικού

5.3 Βάση Δεδομένων

5.4 Διαπροσωπεία Συστήματος

5.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζονται οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος. Επιπλέον, παρουσιάζονται τα εργαλεία, στα οποία πραγματοποιήθηκε η ανάπτυξη του λογισμικού.

5.2 Εργαλεία Υλοποίησης

5.2.1 Τεχνολογίες

ASP.Net: Ανήκει στο γενικότερο .NET framework της Microsoft και παρέχει τη δυνατότητα δημιουργίας δυναμικών ιστοσελίδων και ιστιακών εφαρμογών. Επίσης το εργαλείο αυτό είναι υλοποιημένο πάνω στο Common Language Runtime, που επιτρέπει στον προγραμματιστή να γράψει σε ASP.NET, χρησιμοποιώντας οποιαδήποτε .NET γλώσσα.

jQuery: Είναι μια cross-browser JavaScript βιβλιοθήκη, η οποία παρέχει τη δυνατότητα ενσωμάτωσης script κώδικα στον html κώδικα, με σκοπό την απλοποίηση του client-side scripting. Μερικές από τις λειτουργίες του είναι η εύκολη διαχείριση των DOM elements, η δημιουργία animations, η διαχείριση events (e.g. Click event) ή ακόμα και η ανάπτυξη AJAX εφαρμογών. Στο σύστημα που αναπτύχθηκε, γίνεται συχνή χρήση της τεχνολογίας AJAX για το λόγο ότι επιτρέπει την αλλαγή συγκεκριμένης περιοχής του περιεχομένου (αποφεύγοντας τα post backs-refresh) και παράλληλα αυξάνοντας τη διεπαφή του χρήστη (user experience).

5.2.2 Εργαλεία ανάπτυξης λογισμικού

Microsoft SQL Server 2008 R2: Ο Data base Server στον οποίο είναι εγκατεστημένη η βάση δεδομένων, για την αποθήκευση και ανάκτηση των απαραίτητων πληροφοριών των χρηστών.

Microsoft Visual Studio 2010: Το Microsoft Visual Studio 2010 είναι το περιβάλλον ανάπτυξης εφαρμογών, που υποστηρίζουν τεχνολογίες της Microsoft. Υποστηρίζει πολλές γλώσσες προγραμματισμού και δίνει αρκετές λύσεις στον προγραμματιστή. Επιπλέον, οι εφαρμογές που δημιουργούνται στο περιβάλλον αυτό, είναι πλήρως συμβατές με τα λειτουργικά συστήματα της Microsoft

IIS WebServer: Ο IIS web server, όπου θα γίνει η υλοποίηση της εφαρμογής. Ο web server είναι υπεύθυνος να δέχεται τα αιτήματα από τους πελάτες – χρήστες για να τους εξυπηρετήσει.

Adobe programs (Adobe Flash CS4 Professional, Adobe Fireworks CS4): Το Adobe Fireworks και Flash Professional CS4, είναι βοηθητικά εργαλεία γραφικών, τα οποία χρησιμοποιήθηκαν για τη δημιουργία των λογότυπων (logos) της ιστοσελίδας, καθώς και για το σχεδιασμό ορισμένων εικόνων, όπως για παράδειγμα κάποια βέλη (arrows) ή εικόνες επιλογής.

5.3 Βαση Δεδομένων

Stored Procedures

1. `fn_GET_FILTERED_CREATOR(@FROMcreation_date_year INT,@TOcreation_date_year INT,@isUnknown BIT,@project_id INT,@location NVARCHAR(MAX),@IDMediaType INT,@searchString NVARCHAR(MAX))`

Η διαδικασία αυτή επιστρέφει τους media creators (names), καθώς και τον αριθμό των media creators, που επιστραφήκαν, οι οποίοι είναι φιλτραρισμένοι ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο media filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project, τότε θα μειώσει τις επιλογές του για media creators. Παράδειγμα:

▼ Dates

19xx to 19xx Filter »

☐ Unknown

▼ Project

☐ Amathus Hotel 1

☐ Andreas 1

☐ Andreas Palace 1

☐ Testing 1

☐ Testing Project 1

▼ Media Type

☐ Photograph 2

☐ Journal article 1

☐ Presentation drawings 1

▼ Location

☐ Aruba 1

☐ Azerbaijan 1

☐ Bahamas 1

☐ Unknown 1

▼ Creators

☐ Daphne 1

☐ Martin, Andreas 1

☐ Mary 1

☐ Unknown 1

▼ Dates

19xx to 19xx Filter »

☐ Unknown

▼ Project

☒ Amathus Hotel 1

▼ Media Type

☐ Presentation drawings 1

▼ Location

☐ Bahamas 1

▼ Creators

☐ Martin, Andreas 1

2. `fn_GET_FILTERED_LOCATION(@FROMcreation_date_year INT,@TOcreation_date_year INT,@isUnknown BIT,@project_id INT,@creators NVARCHAR(MAX),@IDMediaType INT,@searchString NVARCHAR(MAX))`

Η διαδικασία αυτή επιστρέφει τα location(countries), καθώς και τον αριθμό των location που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο media filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project, τότε θα μειώσει τις επιλογές του για τα locations.

3. `fn_GET_FILTERED_MEDIA(@FROMcreation_date_year INT, @TOcreation_date_year INT,@isUnknown BIT,@project_id INT,@location NVARCHAR(MAX),@creators NVARCHAR(MAX),@IDMediaType INT,@startRow INT,@maxShownItems INT,@sortBy NVARCHAR(MAX),@searchString NVARCHAR(MAX))`

Η διαδικασία αυτή επιστρέφει τα media (names), καθώς και τον αριθμό των media που επιστραφήκαν, τα οποία είναι φιλτραρισμένα, ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο media filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project, τότε θα μειώσει τις επιλογές του για τα medias.

4. `fn_GET_FILTERED_MEDIA_PROJECT(@FROMcreation_date_year INT,@TOcreation_date_year INT,@isUnknown BIT,@location NVARCHAR(MAX),@creators NVARCHAR(MAX),@IDMediaType INT,@searchString NVARCHAR(MAX))`

Η διαδικασία αυτή επιστρέφει τα projects (names), καθώς και τον αριθμό των projects που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο media filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για creator, τότε θα μειώσει τις επιλογές του για τα projects.

5. `fn_GET_FILTERED_MEDIA_TYPE(@FROMcreation_date_year INT,@TOcreation_date_year INT,@isUnknown BIT,@project_id INT,@location NVARCHAR(MAX),@creators NVARCHAR(MAX),@searchString NVARCHAR(MAX))`

Η διαδικασία αυτή επιστρέφει τα media types (names), καθώς και τον αριθμό των media types που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο media filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project, τότε θα μειώσει τις επιλογές του για τα media types.

```
6. fn_GET_FILTERED_PROGRAM_TYPE(@FROMdate_year INT,@TOdate_year
INT,@project_type_id INT,@project_type_details_id INT,@location
NVARCHAR(MAX),@actor_id INT,@searchString NVARCHAR(MAX))
```

Η διαδικασία αυτή επιστρέφει τα program types (names), καθώς και τον αριθμό των program types που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο project filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project, τότε θα μειώσει τις επιλογές του για τα program types.

```
7. fn_GET_FILTERED_PROJECT_ACTOR(@FROMdate_year INT,@TOdate_year
INT,@project_type_id INT,@project_type_details_id INT,@program_type_id INT,@location
NVARCHAR(MAX),@searchString NVARCHAR(MAX))
```

Η διαδικασία αυτή επιστρέφει τους actors (names), καθώς και τον αριθμό των actors που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο project filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project type details τότε θα μειώσει τις επιλογές του για τους actors.

```
8. fn_GET_FILTERED_PROJECT_LOCATION(@FROMdate_year INT,@TOdate_year
INT,@project_type_id INT,@project_type_details_id INT,@program_type_id INT,@actor_id
INT,@searchString NVARCHAR(MAX))
```

Η διαδικασία αυτή επιστρέφει τα locations (countries), καθώς και τον αριθμό των locations που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο project filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project type details, τότε θα μειώσει τις επιλογές του για τα locations.

9. fn_GET_FILTERED_PROJECT_TYPE(@FROMdate_year INT,@TOdate_year INT,@project_type_details_id INT,@program_type_id INT,@location NVARCHAR(MAX),@actor_id INT,@searchString NVARCHAR(MAX))

Η διαδικασία αυτή επιστρέφει τα project types (names), καθώς και τον αριθμό των project types, που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο project filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project type details, τότε θα μειώσει τις επιλογές του για τα project types.

10. fn_GET_FILTERED_PROJECT_TYPE_DETAILS(@FROMdate_year INT,@TOdate_year INT,@project_type_id INT,@program_type_id INT,@location NVARCHAR(MAX),@actor_id INT,@searchString NVARCHAR(MAX))

Η διαδικασία αυτή επιστρέφει τα project type details (names), καθώς και τον αριθμό των project type details, που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο project filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για location, τότε θα μειώσει τις επιλογές του για τα project type details.

11. fn_GET_FILTERED_PROJECTS(@FROMdate_year INT,@TOdate_year INT,@location NVARCHAR(MAX),@project_type_id INT,@project_type_details_id INT,@program_type_id INT,@actor_id INT,@startRow INT,@maxShownItems INT,@sortBy NVARCHAR(MAX),@searchString NVARCHAR(MAX))

Η διαδικασία αυτή επιστρέφει τα projects(names), καθώς και τον αριθμό των projects, που επιστραφήκαν, τα οποία είναι φιλτραρισμένα ανάλογα με τις υπόλοιπες επιλογές του χρήστη στο project filter panel. Αν για παράδειγμα ο χρήστης βάλει φίλτρο για project type details, τότε θα μειώσει τις επιλογές του για τα projects.

12. fn_GET_MEDIA_COUNT(@FROMcreation_date_year INT,@TOcreation_date_year INT,@isUnknown BIT,@project_id INT,@location NVARCHAR(MAX),@creators NVARCHAR(MAX),@IDMediaType INT,@searchString NVARCHAR(MAX))

Η διαδικασία αυτή μετρά και επιστρέφει τον αριθμό των medias, που θα εμφανιστούν βάση των φίλτρων, που έβαλε ο χρήστης.

13. fn_GET_ORIGINAL_USES(@project_id INT)

Η διαδικασία επιστρέφει ένα string των μη κενών original uses, που έχει ένα project. Χρησιμοποιείται στην περίπτωση όπου η επιλογή list view είναι ενεργοποιημένη.

14. fn_GET_PROJECTS_COUNT(@FROMdate_year INT,@TOdate_year INT,@location NVARCHAR(MAX),@project_type_id INT,@project_type_details_id INT,@program_type_id INT,@actor_id INT,@searchString NVARCHAR(MAX))

Η διαδικασία αυτή μετρά και επιστρέφει τον αριθμό των projects, που θα εμφανιστούν βάση των φίλτρων, που έβαλε ο χρήστης.

Scalar-valued Functions

1. fn_COUNTRY_CITY(@ID_COUNTRY INT,@CITY NVARCHAR(MAX))

Χρησιμοποιείται για την υπολογιζόμενη στήλη (computed column) COUNTRY_CITY (project, media) και επιστρέφει unknown σε περίπτωση, που δεν έχει ούτε country ούτε city, μόνο το country ή city στην περίπτωση, που έχει μόνο country ή city και όταν έχει και τα δύο επιστρέφει και τα δύο, χωρισμένα με μια παύλα.

2. fn_CREATION_DATE(@creationDateDay INT,@creationDateMonth INT,@creationDateYear INT)

Χρησιμοποιείται για την υπολογιζόμενη στήλη (computed column) `CREATION_DATE(media)` και επιστρέφει την ημερομηνία σε αναπαράσταση 11 October 1999. Εάν δεν έχει κάποιο από τα `day`, `month`, `year` τότε δεν το εμφανίζει.

3. `fn_GET_CITATION_FOR_WEBSITE(@ID_PROJECT_CITATIONS INT)`

Επιστρέφει ένα string με την αναπαράσταση των citations fields.

4. `fn_PROJECT_SEARCH_DATE(@PROPOSAL_DATE_YEAR SMALLINT,@COMPLETION_DATE_YEAR SMALLINT,@CONSTRUCTION_DATE_YEAR SMALLINT)`

Χρησιμοποιείται για την υπολογιζόμενη στήλη (computed column) `SEARCH_YEAR(project)` και καθορίζει με βάση ποιο year θα γίνεται search στο φιλτράρισμα των projects. Εάν υπάρχει τιμή για το `PROPOSAL_DATE_YEAR`, τότε θα επιστρέψει αυτό, αλλιώς θα επιστρέψει το `COMPLETION_DATE_YEAR`. Ανδενέχειούτε`COMPLETION_DATE_YEAR` τότε επιστρέφει το `CONSTRUCTION_DATE_YEAR`.

5. `fn_SHORTEN_MEDIA_TITLE(@ID_MEDIA INT,@MAX_CHARS INT)`

Χρησιμοποιείται για την υπολογιζόμενη στήλη (computed column) `SHORT_TITLE(media)` και εάν ο τίτλος (title) του media είναι μεγαλύτερος από `@MAX_CHARS` χαρακτήρες, επιστρέφει μόνο τους `@MAX_CHARS` πρώτους και προσθέτει τρεις τελείες στο τέλος.

6. `fn_SHORTEN_PROJECT_TITLE(@ID_PROJECT INT,@MAX_CHARS INT)`

Χρησιμοποιείται για την υπολογιζόμενη στήλη (computed column) `SHORT_TITLE(project)` και εάν ο τίτλος (title) του project είναι μεγαλύτερος από `@MAX_CHARS` χαρακτήρες επιστρέφει μόνο τους `@MAX_CHARS` πρώτους και προσθέτει τρεις τελείες στο τέλος.

7. fn_HAS_ADMIN_PANEL (@ID_USER_ROLE INT)

Χρησιμοποιείται για την υπολογιζόμενη στήλη (computed column) HAS_ADMIN_PANEL (user_roles) και εάν έστω και ένα από τα BACKUP_SYSTEM, BACKUP_DATA, MANAGE_ACCOUNTS, DATA_ENTRY, APPROVE_SCHEMAS, THES_ORIGINAL_PROGRAM, THES_PROJ_TYPE, THES_PROJ_TYPE_DETAILS, THES_COUNTRIES, THES_MAIN_CONST, THES_SECOND_CONST, THES_BUILDING_ALTER, THES_MEDIA_TYPE, THES_FORMAT, THES_ACTOR_TYPE, THES_ACTOR_ROLE, THES_KEYWORDS γνωρίσματα του user_roles πίνακα είναι true, τότε επιστρέφει true, αλλιώς false. Αυτό μας δίνει τη δυνατότητα να θέσουμε τι θέλουμε να έχει το Admin Panel, ώστε ανάλογα με τον ρόλο του χρήστη να του το εμφανίζουμε ή όχι.

8. fn_CAN_EDIT_THESAURUS (@ID_USER_ROLE INT)

Χρησιμοποιείται για την υπολογιζόμενη στήλη (computed column) CAN_EDIT_THES (user_roles) και εάν έστω και ένα από τα THES_ORIGINAL_PROGRAM, THES_PROJ_TYPE, THES_PROJ_TYPE_DETAILS, THES_COUNTRIES, THES_MAIN_CONST, THES_SECOND_CONST, THES_BUILDING_ALTER, THES_MEDIA_TYPE, THES_FORMAT, THES_ACTOR_TYPE, THES_ACTOR_ROLE, THES_KEYWORDS γνωρίσματα του user_roles πίνακα είναι true, τότε επιστρέφει true αλλιώς false. Αυτό μας δίνει την δυνατότητα να θέσουμε αν θα εμφανίζει την επιλογή EDITLISTS/THESAURUS στο χρήστη, ανάλογα με τον ρόλο του.

5.4 Διαπροσωπεία Συστήματος

Η Mesarch διαδικτυακή εφαρμογή είναι δυναμική ιστοσελίδα, υλοποιημένη σε Microsoft Server τεχνολογία, ASP.NET με SQL Server για την βάση δεδομένων. Αποτελείται από HTML κώδικα, AJAX, Javascript, καθώς και τοC# κώδικα στο code behind της ASP.NET.

Το βασικό πλάνο της διαπροσωπείας είναι το ακόλουθο. Δημιουργήθηκε το βασικό template της σελίδας, από το οποίο όλες οι υπόλοιπες «κληρονομούν». Υπάρχουν επιμέρους template, τα οποία κληρονομούν κάποιες σελίδες ως μέρος μιας ομάδας, όπως για παράδειγμα το search Projects, search Actors, search Media, search Users, κληρονομούν

από το γενικό template, υπεύθυνο για το search (Search Master Page). Το ίδιο γίνεται και με τις φόρμες εισαγωγής δεδομένων ξεχωριστά για project και actor, με τα thesaurus και με τους users.

Σε όλα τα hyperlinks του κώδικα της ιστοσελίδας, χρησιμοποιείται η σύμβαση ότι πρέπει να πάνε ένα φάκελο πίσω και μετά στο φάκελο, όπου βρίσκεται ο σύνδεσμος για ανακατεύθυνση και μετά στη σελίδα ανακατεύθυνσης. Για αυτό και υπάρχει μια ομαδοποίηση με όλες τις σελίδες μέσα σε φακέλους.

Στο code behind μπήκε ο κώδικας, που δημιουργεί δυναμικά το περιεχόμενο μιας σελίδας, επικοινωνώντας με τη βάση δεδομένων για την εξαγωγή δεδομένων ή για την εισαγωγή δεδομένων στη βάση.

Υπάρχουν και κάποιες βοηθητικές κλάσεις για να βοηθήσουν στη φόρτωση του περιεχομένου μιας ιστοσελίδας ασύγχρονα (χωρίς πλήρη επαναφόρτιση της παρούσας σελίδας - AJAX). Οι κλάσεις αυτές καλούνται μέσω Javascript/AJAX κώδικα, δίνοντας τους κάποιες παραμέτρους, όπως κάποια ταυτότητα κάποιου project. Αυτές οι κλάσεις λοιπόν αναλαμβάνουν να αλλάξουν το περιεχόμενο ενός <div>tag, ανάλογα με τις παραμέτρους που παίρνουν.

Με την εμφάνιση κάποιου project –αντικειμένου, ο χρήστης έχει τη δυνατότητα να αλλάξει δυναμικά το περιεχόμενο των πληροφοριών, που μπορεί να βλέπει στα δεξιά της σελίδας επιλέγοντας να δει επιπλέον πληροφορίες για το project ή να δει κάποιο media ή actor, που έχει σχέση με το project. Η αλλαγή αυτή των πληροφοριών γίνεται δυναμικά, μέσω της χρήσης AJAX τεχνολογίας, το ίδιο και η εμφάνιση των εικόνων κάποιου project ή media. Το quick search της ιστοσελίδας, γίνεται επίσης με AJAX, κάθε φορά που ο χρήστης πληκτρολογεί ένα χαρακτήρα, γίνεται αμέσως search στη βάση δεδομένων και εμφανίζονται τα αποτελέσματα στην εφαρμογή, χωρίς να χρειάζεται επαναφορά (refresh) ολόκληρης της σελίδας.

Τέλος σε πολλά μέρη του προγράμματος έγινε χρήση javascript κώδικα, το οποίο κάνει πιο εντυπωσιακή την εμπειρία του χρήστη με το σύστημα. Μερικά παραδείγματα χρήσης javascript κώδικα στη Mesarch εφαρμογή είναι το fade-in, fade-out των tabs στο βασικό

template της σελίδας, κάποια show/hide <div> στην εισαγωγή δεδομένων, κάποια slides up and down στο filter panel και στην εμφάνιση κάποιων επιπλέον πληροφοριών.

Κεφάλαιο 6 –Βιβλιογραφία

- Aletras, N., Stevenson, M., & Clough, P. (2012). Computing similarity between items in a digital library of cultural heritage.
- Arch daily*. (n.d.). Ανάκτηση από Arch daily web site: www.archdaily.com/
- Architizer*. (n.d.). Ανάκτηση από Architizer web site: www.architizer.com/en_us
- Artlex*. (n.d.). Ανάκτηση August 1996, από Artlex web site: www.artlex.com/
- Contemporist*. (n.d.). Ανάκτηση από Contemporist web site: www.contemporist.com/
- Great buildings collections*. (n.d.). Ανάκτηση 1997, από Great buildings collections web site: www.greatbuildings.com/gbc.html
- GroupHomeDeal*. (n.d.). Ανάκτηση από GroupHomeDeal web site: www.grouphomedead.com/
- Harris, T., & Morgan, J. (2011). *Marcel Breurer digital archive*. Ανάκτηση από Marcel Breurer digital archive web site: breuer.syr.edu/
- Hiroshi Sugimoto*. (n.d.). Ανάκτηση από Hiroshi Sugimoto web site: www.sugimotohiroshi.com
- Let us design/build*. (n.d.). Ανάκτηση από Let us design/build web site: www.letusserveyou.com/
- Lucian, G. S. (n.d.). *Homedit*. Ανάκτηση από Homedit web site: <http://www.homedit.com/>
- Matthews, P., & Aston, J. (2012). Interactive multimedia ethnography: Archiving workflow, interface aesthetics and metadata.
- Mimno, D. (2012). Computational historiography: Data mining in a century of classics journals.
- Postmus, M. (2009). *Archello*. Ανάκτηση από Archello web site: www.archello.com/en
- Sadler Brown Architecture*. (n.d.). Ανάκτηση από Sadler Brown Architecture web site: www.sadlerbrown.co.uk/default.aspx
- Schott glass made of ideas*. (n.d.). Ανάκτηση από Schott web site: www.schott.com
- Seltzer, D. J. (n.d.). *Agilitynut*. Ανάκτηση 2000, από Agilitynut web site: www.agilitynut.com/roadside.html
- Visnjic, F., & Panfilo, P. (n.d.). *Archigram*. Ανάκτηση από Archigram web site: archigram.westminster.ac.uk