

Πτυχιακή Εργασία

**Σχεδίαση και υλοποίηση βιβλιογραφικής βάσης
δεδομένων για τη διατήρηση πηγών/βιβλιογραφίας
και της διεπαφής της με τον χρήστη**

Ευάγγελος Δημόπουλος - Π2018100

Επιβλέπουσα: Σωτηροπούλου Άννα

Κέρκυρα/Σεπτέμβριος/2024

1 Πίνακας Περιεχομένων

1	Εισαγωγή	4
1.1	Αντικείμενο και Σκοπός της Εργασίας.....	4
1.2	Κίνητρα για την Ανάπτυξη του Συστήματος.....	4
1.3	Δομή της Εργασίας	5
2	Θεωρητικό Υπόβαθρο.....	7
2.1	Βασικές Έννοιες.....	7
2.2	Υπάρχοντα Συστήματα και Τεχνολογίες.....	8
2.3	Τεχνολογίες και Εργαλεία Ανάπτυξης.....	12
2.4	Γλώσσες Προγραμματισμού	12
2.5	Διεθνή Πρότυπα Βιβλιογραφίας	15
2.5.1	Ιστορικό Ανάπτυξης των Προτύπων	15
2.5.2	Σκοπός και Λειτουργία των Διεθνών Προτύπων	16
2.5.3	Παρουσίαση Κύριων Διεθνών Προτύπων Βιβλιογραφίας	16
2.5.4	Πρότυπο APA (American Psychological Association)	16
2.5.5	Harvard Style	17
2.5.6	Πρότυπο IEEE (Institute of Electrical and Electronics Engineers)	17
2.5.7	Πρότυπο Vancouver	18
2.5.8	Πρότυπο MLA (Modern Language Association)	18
2.6	Συγκριτική Ανάλυση των Διεθνών Προτύπων.....	19
3	Ανάλυση Απαιτήσεων	20
3.1	Λειτουργικές Απαιτήσεις.....	20
3.2	Μη Λειτουργικές Απαιτήσεις	21

3.3	Χρήστες του Συστήματος	23
4	Σχεδίαση του Συστήματος	25
4.1	Ανάλυση του Συστήματος	25
4.2	Διάγραμμα Οντοτήτων-Συσχετίσεων (ER)	26
4.3	Σχεδίαση Διεπαφής Χρήστη	38
5	Υλοποίηση του Συστήματος.....	45
5.1	Περιβάλλον Ανάπτυξης και Τεχνολογίες	45
5.1.1	Βάση Δεδομένων	47
5.1.2	Περιβάλλον Ανάπτυξης.....	49
5.2	Βήματα Υλοποίησης	51
5.2.1	Ανάλυση των Απαιτήσεων	51
5.2.2	Σχεδιασμός της Βάσης Δεδομένων	51
5.2.3	Ανάπτυξη του Back-End Συστήματος	52
5.2.4	Ανάπτυξη της Διεπαφής Χρήστη (Front-End)	53
5.2.5	Έλεγχος και Επικύρωση (Testing)	53
5.2.6	Τελική Εγκατάσταση	53
6	Αξιολόγηση και Αποτελέσματα.....	54
6.1	Μέθοδοι Αξιολόγησης	54
6.2	Αποτελέσματα Αξιολόγησης	55
7	Συμπεράσματα και Μελλοντικές Επεκτάσεις.....	56
7.1	Συμπεράσματα	56
7.2	Μελλοντικές Επεκτάσεις	57
	Βιβλιογραφία	58
	Παράρτημα Α - Κώδικας	60

1 Εισαγωγή

1.1 Αντικείμενο και Σκοπός της Εργασίας

Η παρούσα εργασία επικεντρώνεται στην ανάπτυξη μιας βιβλιογραφικής βάσης δεδομένων που έχει ως στόχο τη διατήρηση και τη διαχείριση δεδομένων που χρησιμοποιούνται ως αναφορές σε επιστημονικά άρθρα, εργασίες και βιβλία. Η διαχείριση των βιβλιογραφικών αναφορών αποτελεί κρίσιμο στοιχείο της ερευνητικής διαδικασίας, καθώς διευκολύνει την οργάνωση και την ανάκτηση πληροφοριών, υποστηρίζοντας έτσι την παραγωγή υψηλής ποιότητας ακαδημαϊκού έργου.

Ο σκοπός της ανάπτυξης του βιβλιογραφικού συστήματος είναι διττός. Πρώτον, να παρέχει ένα ολοκληρωμένο εργαλείο που επιτρέπει στους ερευνητές, φοιτητές και ακαδημαϊκούς να αποθηκεύουν, να οργανώνουν και να αναζητούν βιβλιογραφικές αναφορές με αποδοτικό τρόπο. Δεύτερον, να υποστηρίξει πολλαπλούς τύπους βιβλιογραφικών αναφορών, όπως άρθρα, βιβλία, διατριβές, τεχνικές αναφορές και διαδικτυακές πηγές, προσφέροντας παράλληλα λειτουργίες όπως η δημιουργία βιβλιογραφικών καταλόγων και η εξαγωγή δεδομένων σε διάφορες μορφές.

1.2 Κίνητρα για την Ανάπτυξη του Συστήματος

Η ανάγκη για την ανάπτυξη ενός συστήματος διαχείρισης βιβλιογραφικών δεδομένων προκύπτει από την αυξανόμενη πολυπλοκότητα και τον όγκο των βιβλιογραφικών αναφορών σε ακαδημαϊκά και ερευνητικά περιβάλλοντα. Οι υπάρχουσες λύσεις, όπως το BibTeX και το Mendeley, παρά τις δυνατότητες που προσφέρουν, συχνά περιορίζονται ως προς την ευχρηστία και την επεκτασιμότητά τους. Ένα νέο σύστημα μπορεί να ενσωματώσει σύγχρονες τεχνολογίες και να προσφέρει βελτιωμένη απόδοση, ασφάλεια και ευελιξία, καλύπτοντας τις ανάγκες των χρηστών με πιο αποδοτικό τρόπο.

Επιπλέον, η ανάγκη για αποτελεσματική διαχείριση βιβλιογραφικών δεδομένων είναι ιδιαίτερα εμφανής σε μεγάλα ερευνητικά έργα και ακαδημαϊκά ιδρύματα, όπου η διατήρηση της ακεραιότητας και της ακρίβειας των βιβλιογραφικών αναφορών είναι ζωτικής σημασίας. Η ανάπτυξη ενός νέου συστήματος μπορεί να συμβάλει στη βελτίωση της ποιότητας των ακαδημαϊκών εργασιών και να υποστηρίξει την ευρύτερη ερευνητική κοινότητα.

1.3 Δομή της Εργασίας

Η παρούσα εργασία είναι δομημένη με τέτοιο τρόπο ώστε να προσφέρει μια ολοκληρωμένη προσέγγιση στην ανάπτυξη ενός συστήματος διαχείρισης βιβλιογραφικών αναφορών.

Αρχικά, στο κεφάλαιο της Εισαγωγής, γίνεται μια γενική παρουσίαση του αντικειμένου της εργασίας και των στόχων που θέτει. Αναλύονται τα κίνητρα που οδήγησαν στην ανάπτυξη του συστήματος, καθώς και η ανάγκη για την ύπαρξη ενός τέτοιου εργαλείου στην ακαδημαϊκή και ερευνητική κοινότητα.

Στη συνέχεια, το κεφάλαιο του Θεωρητικού Υπόβαθρου εστιάζει στις βασικές έννοιες και τεχνολογίες που σχετίζονται με το αντικείμενο της εργασίας. Παρουσιάζονται οι βασικές αρχές των τεχνολογιών ιστού και της διαχείρισης βάσεων δεδομένων, καθώς και μια ανασκόπηση των υπάρχοντων συστημάτων διαχείρισης βιβλιογραφικών δεδομένων, όπως το BibTeX και το Mendeley. Επίσης, αναλύονται οι γλώσσες προγραμματισμού και τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη του συστήματος.

Το κεφάλαιο της Ανάλυσης Απαιτήσεων προσδιορίζει τις λειτουργικές και μη λειτουργικές απαιτήσεις του συστήματος. Αναλύονται οι βασικές λειτουργίες που πρέπει να υλοποιηθούν, καθώς και οι απαιτήσεις απόδοσης, ασφάλειας και επεκτασιμότητας. Επίσης, γίνεται μια κατηγοριοποίηση των χρηστών του συστήματος και περιγράφονται οι ανάγκες και οι απαιτήσεις τους.

Η Σχεδίαση του Συστήματος αποτελεί το επόμενο κεφάλαιο, όπου παρουσιάζεται η αρχιτεκτονική του συστήματος και αναλύονται οι επιμέρους λειτουργίες και υπηρεσίες. Περιλαμβάνονται το διάγραμμα οντοτήτων-συσχετίσεων (ER), η μετατροπή του σε σχεσιακό σχήμα και η περιγραφή των πινάκων και των σχέσεων. Επίσης, παρουσιάζεται η σχεδίαση της διεπαφής χρήστη με τη βοήθεια mockups και wireframes.

Το κεφάλαιο της Υλοποίησης του Συστήματος περιγράφει το περιβάλλον ανάπτυξης και τις τεχνολογίες που χρησιμοποιήθηκαν. Αναλύονται τα βήματα υλοποίησης και περιγράφονται τα προβλήματα και οι προκλήσεις που αντιμετωπίστηκαν κατά την ανάπτυξη, καθώς και οι τρόποι

αντιμετώπισής τους. Παρουσιάζονται παραδείγματα κώδικα για να υποστηριχθούν οι περιγραφές.

Στο κεφάλαιο της Αξιολόγησης και των Αποτελεσμάτων, περιγράφονται οι μέθοδοι που χρησιμοποιήθηκαν για την αξιολόγηση του συστήματος και παρουσιάζονται τα αποτελέσματα από τη δοκιμή του. Γίνεται ανάλυση της απόδοσης και της αποτελεσματικότητας του συστήματος, καθώς και των δυνατοτήτων βελτίωσης.

Το κεφάλαιο των Συμπερασμάτων και των Μελλοντικών Επεκτάσεων συνοψίζει τα αποτελέσματα της εργασίας και προτείνει μελλοντικές βελτιώσεις και επεκτάσεις του συστήματος. Περιγράφονται οι δυνατότητες για περαιτέρω ανάπτυξη και οι προοπτικές για την ενσωμάτωση νέων λειτουργιών.

Η εργασία κλείνει με τη Βιβλιογραφία, όπου παρατίθενται όλες οι πηγές που χρησιμοποιήθηκαν, και το Παράρτημα, το οποίο περιλαμβάνει επιπλέον πληροφορίες και οδηγίες χρήσης του συστήματος.

2 Θεωρητικό Υπόβαθρο

Το κεφάλαιο αυτό παρέχει μια εκτενή και ολοκληρωμένη κατανόηση των βασικών εννοιών και τεχνολογιών που σχετίζονται με την ανάπτυξη ενός συστήματος διαχείρισης βιβλιογραφικών δεδομένων. Αρχικά, θα εξεταστούν οι βασικές έννοιες που αφορούν τις τεχνολογίες ιστού και τη διαχείριση βάσεων δεδομένων. Στη συνέχεια, θα παρουσιαστούν υπάρχοντα βιβλιογραφικά συστήματα και τεχνολογίες, αναλύοντας τα πλεονεκτήματα και τα μειονεκτήματά τους. Τέλος, θα εξεταστούν οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη του συστήματος.

2.1 Βασικές Έννοιες

Στην ενότητα αυτή, θα αναλυθούν οι βασικές έννοιες που σχετίζονται με τις τεχνολογίες ιστού και τη διαχείριση βάσεων δεδομένων. Αυτές οι έννοιες αποτελούν τη βάση για την κατανόηση της λειτουργίας του βιβλιογραφικού συστήματος και της ανάπτυξής του.

Τεχνολογίες Ιστού

Οι τεχνολογίες ιστού αναφέρονται στις τεχνολογίες που χρησιμοποιούνται για την ανάπτυξη και την παροχή περιεχομένου στο Διαδίκτυο. Οι κύριες τεχνολογίες ιστού περιλαμβάνουν την HTML (Hypertext Markup Language), την CSS (Cascading Style Sheets) και την JavaScript. Η HTML χρησιμοποιείται για τη δημιουργία της δομής των ιστοσελίδων, η CSS για τον καθορισμό της εμφάνισής τους και η JavaScript για την προσθήκη διαδραστικότητας και δυναμικών λειτουργιών. Επίσης, άλλες τεχνολογίες όπως το Ajax (Asynchronous JavaScript and XML) και η PHP (Hypertext Preprocessor) παίζουν σημαντικό ρόλο στην ανάπτυξη διαδραστικών και δυναμικών ιστοσελίδων.

Η HTML είναι η βάση κάθε ιστοσελίδας, παρέχοντας τη δομή και το περιεχόμενο μέσω μιας σειράς στοιχείων και χαρακτηριστικών. Η CSS συμπληρώνει την HTML προσθέτοντας στυλιστικές ιδιότητες που καθορίζουν την εμφάνιση των σελίδων, ενώ η JavaScript επιτρέπει την προσθήκη δυναμικών στοιχείων, όπως φόρμες επικοινωνίας, κινούμενες εικόνες και άλλα διαδραστικά χαρακτηριστικά. Το Ajax επιτρέπει την ασύγχρονη επικοινωνία με τον διακομιστή, επιτρέποντας την ενημέρωση μέρους της ιστοσελίδας χωρίς να απαιτείται ανανέωση ολόκληρης της σελίδας.

Τέλος, η PHP είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται ευρέως για την ανάπτυξη διακομιστών και την ενσωμάτωση δυναμικών λειτουργιών σε ιστοσελίδες.

Διαχείριση Βάσεων Δεδομένων

Η διαχείριση βάσεων δεδομένων είναι η διαδικασία αποθήκευσης, οργάνωσης και διαχείρισης δεδομένων με τρόπο που επιτρέπει την εύκολη ανάκτηση και ενημέρωση. Οι βάσεις δεδομένων χρησιμοποιούνται ευρέως σε πολλές εφαρμογές, συμπεριλαμβανομένων των συστημάτων διαχείρισης βιβλιογραφικών αναφορών. Τα συστήματα διαχείρισης βάσεων δεδομένων (DBMS) όπως το MySQL, το PostgreSQL και το SQLite παρέχουν τα εργαλεία και τις λειτουργίες για τη διαχείριση δεδομένων με αποδοτικό και ασφαλή τρόπο.

Το MySQL είναι ένα από τα πιο δημοφιλή συστήματα διαχείρισης βάσεων δεδομένων, γνωστό για την αξιοπιστία, την απόδοση και την ευκολία χρήσης του. Χρησιμοποιείται ευρέως σε πολλές εφαρμογές ιστού και υποστηρίζει μεγάλες κλίμακες δεδομένων και χρήστες. Το PostgreSQL είναι ένα ακόμα ισχυρό DBMS, το οποίο προσφέρει προηγμένα χαρακτηριστικά όπως υποστήριξη για πολύπλοκα ερωτήματα και προσαρμοσμένες λειτουργίες. Το SQLite είναι ένα ελαφρύ σύστημα διαχείρισης βάσεων δεδομένων, ιδανικό για εφαρμογές που απαιτούν μικρή αποθηκευτική δυνατότητα και χαμηλή κατανάλωση πόρων.

2.2 Υπάρχοντα Συστήματα και Τεχνολογίες

Σε αυτή την ενότητα, θα παρουσιαστούν μερικά από τα υπάρχοντα συστήματα διαχείρισης βιβλιογραφικών δεδομένων και οι τεχνολογίες που χρησιμοποιούν. Θα γίνει ανάλυση των πλεονεκτημάτων και των μειονεκτημάτων τους, προκειμένου να κατανοήσουμε τις δυνατότητες και τους περιορισμούς τους.

BibTeX

Το BibTeX είναι ένα εργαλείο που χρησιμοποιείται ευρέως για τη διαχείριση βιβλιογραφικών αναφορών σε κείμενα LaTeX. Παρέχει μια τυποποιημένη μορφή για την αποθήκευση βιβλιογραφικών δεδομένων και επιτρέπει την εύκολη εισαγωγή και εξαγωγή τους σε ακαδημαϊκές εργασίες. Ένα από τα πλεονεκτήματα του BibTeX είναι η ευρεία υποστήριξή του σε εργαλεία επεξεργασίας κειμένου και η δυνατότητα δημιουργίας προσαρμοσμένων μορφών

αναφοράς. Ωστόσο, το BibTeX περιορίζεται από τη γραμμική δομή των αρχείων του, γεγονός που μπορεί να δυσκολεύει τη διαχείριση μεγάλων βιβλιογραφικών βάσεων.

Το BibTeX επιτρέπει την εισαγωγή βιβλιογραφικών αναφορών από μια ποικιλία πηγών, όπως άρθρα, βιβλία, διατριβές και τεχνικές αναφορές, και υποστηρίζει μια σειρά από διαφορετικούς τύπους βιβλιογραφικών εγγραφών. Παρά το γεγονός ότι το BibTeX είναι πολύ ισχυρό, η γραμμική του δομή μπορεί να δυσκολέψει τη διαχείριση και την αναζήτηση βιβλιογραφικών δεδομένων σε μεγάλα έργα, καθώς οι χρήστες πρέπει να θυμούνται ή να αναζητούν τις εγγραφές με βάση τις ετικέτες τους.

Mendeley

Το Mendeley είναι ένα σύγχρονο εργαλείο διαχείρισης βιβλιογραφικών δεδομένων που προσφέρει πλούσιες δυνατότητες, όπως η αυτόματη εξαγωγή μεταδεδομένων από αρχεία PDF, η ενσωμάτωση με προγράμματα επεξεργασίας κειμένου και η δυνατότητα συνεργασίας με άλλους ερευνητές. Ένα από τα κύρια πλεονεκτήματα του Mendeley είναι η δυνατότητα συγχρονισμού δεδομένων μέσω του cloud, που επιτρέπει στους χρήστες να έχουν πρόσβαση στις βιβλιογραφικές αναφορές τους από οπουδήποτε. Παρ' όλα αυτά, το Mendeley έχει και ορισμένα μειονεκτήματα, όπως η εξάρτηση από σύνδεση στο διαδίκτυο για τον πλήρη έλεγχο των δεδομένων και οι περιορισμοί στις δυνατότητες εξαγωγής δεδομένων.

Το Mendeley διευκολύνει την εισαγωγή βιβλιογραφικών δεδομένων από μια ποικιλία πηγών και υποστηρίζει την αυτόματη εξαγωγή μεταδεδομένων από αρχεία PDF, γεγονός που μειώνει τον χρόνο και την προσπάθεια που απαιτείται για την εισαγωγή δεδομένων. Επιπλέον, προσφέρει δυνατότητες αναζήτησης και φιλτραρίσματος που επιτρέπουν στους χρήστες να βρίσκουν γρήγορα τις βιβλιογραφικές αναφορές που χρειάζονται. Παρόλα αυτά, οι περιορισμοί στις δυνατότητες εξαγωγής δεδομένων και η ανάγκη για σύνδεση στο διαδίκτυο μπορεί να αποτελέσουν προβλήματα για ορισμένους χρήστες.

EndNote

Το EndNote είναι ένα άλλο δημοφιλές εργαλείο διαχείρισης βιβλιογραφικών αναφορών που προσφέρει πλούσιες δυνατότητες οργάνωσης και διαχείρισης. Υποστηρίζει πολλές μορφές

αναφοράς και επιτρέπει στους χρήστες να οργανώνουν τις βιβλιογραφικές αναφορές τους σε κατηγορίες και φακέλους. Ένα από τα πλεονεκτήματα του EndNote είναι η ευρεία υποστήριξη του από ακαδημαϊκά ιδρύματα και βιβλιοθήκες. Ωστόσο, η πολυπλοκότητα της εφαρμογής και το κόστος αδειοδότησης μπορεί να αποτελέσουν περιορισμούς για ορισμένους χρήστες.

Το EndNote επιτρέπει στους χρήστες να εισάγουν βιβλιογραφικά δεδομένα από διάφορες πηγές και να τα οργανώνουν με τρόπο που διευκολύνει την ανάκτηση και τη χρήση τους. Οι χρήστες μπορούν να δημιουργήσουν προσαρμοσμένες κατηγορίες και φακέλους για να οργανώσουν τις βιβλιογραφικές αναφορές τους και να χρησιμοποιήσουν προηγμένα εργαλεία αναζήτησης για να βρουν γρήγορα τις αναφορές που χρειάζονται. Παρά τα πλεονεκτήματα αυτά, η πολυπλοκότητα της εφαρμογής και το κόστος αδειοδότησης μπορεί να αποτελέσουν αποτρεπτικούς παράγοντες για ορισμένους χρήστες, ειδικά για μικρά ερευνητικά έργα ή για χρήστες με περιορισμένους πόρους.

Zotero

Το Zotero είναι ένα ελεύθερο και ανοιχτού κώδικα εργαλείο διαχείρισης βιβλιογραφικών δεδομένων που προσφέρει πολλές από τις δυνατότητες των εμπορικών λύσεων, χωρίς το κόστος αδειοδότησης. Το Zotero επιτρέπει στους χρήστες να αποθηκεύουν, να οργανώνουν και να διαμοιράζονται βιβλιογραφικές αναφορές, καθώς και να συλλέγουν δεδομένα από ιστοσελίδες μέσω των διαθέσιμων επεκτάσεων για προγράμματα περιήγησης. Ένα από τα πλεονεκτήματα του Zotero είναι η ενσωμάτωση με πολλαπλά προγράμματα επεξεργασίας κειμένου και η δυνατότητα συγχρονισμού δεδομένων μέσω του cloud [15].

Το Zotero υποστηρίζει μια ποικιλία μορφών αναφοράς και επιτρέπει την εύκολη εισαγωγή βιβλιογραφικών δεδομένων από ιστοσελίδες, βιβλιοθήκες και αρχεία PDF. Οι χρήστες μπορούν να οργανώσουν τις αναφορές τους σε κατηγορίες και να χρησιμοποιήσουν εργαλεία αναζήτησης για να βρουν γρήγορα τις αναφορές που χρειάζονται. Επιπλέον, το Zotero προσφέρει δυνατότητες συνεργασίας, επιτρέποντας στους χρήστες να διαμοιράζονται τις βιβλιογραφικές αναφορές τους με άλλους ερευνητές. Παρά τα πλεονεκτήματα αυτά, το Zotero μπορεί να έχει περιορισμούς σε σύγκριση με εμπορικά εργαλεία όσον αφορά τις δυνατότητες προσαρμογής και την υποστήριξη.

RefWorks

Το RefWorks είναι ένα άλλο εργαλείο διαχείρισης βιβλιογραφικών αναφορών που χρησιμοποιείται ευρέως σε ακαδημαϊκά περιβάλλοντα. Προσφέρει δυνατότητες εισαγωγής, οργάνωσης και διαμοιρασμού βιβλιογραφικών δεδομένων, καθώς και τη δυνατότητα δημιουργίας βιβλιογραφικών καταλόγων σε διάφορες μορφές αναφοράς. Ένα από τα κύρια πλεονεκτήματα του RefWorks είναι η υποστήριξη για πολλαπλά πρότυπα αναφοράς και η δυνατότητα ενσωμάτωσης με προγράμματα επεξεργασίας κειμένου.

Το RefWorks επιτρέπει στους χρήστες να εισάγουν βιβλιογραφικά δεδομένα από διάφορες πηγές και να τα οργανώνουν με τρόπο που διευκολύνει την ανάκτηση και τη χρήση τους. Οι χρήστες μπορούν να δημιουργήσουν προσαρμοσμένες κατηγορίες και φακέλους για να οργανώσουν τις βιβλιογραφικές αναφορές τους και να χρησιμοποιήσουν προηγμένα εργαλεία αναζήτησης για να βρουν γρήγορα τις αναφορές που χρειάζονται. Παρά τα πλεονεκτήματα αυτά, η πολυπλοκότητα της εφαρμογής και το κόστος αδειοδότησης μπορεί να αποτελέσουν αποτρεπτικούς παράγοντες για ορισμένους χρήστες, ειδικά για μικρά ερευνητικά έργα ή για χρήστες με περιορισμένους πόρους.

Citavi

Το Citavi είναι ένα εργαλείο διαχείρισης βιβλιογραφικών αναφορών που προσφέρει πλούσιες δυνατότητες οργάνωσης και διαχείρισης. Υποστηρίζει πολλές μορφές αναφοράς και επιτρέπει στους χρήστες να οργανώνουν τις βιβλιογραφικές αναφορές τους σε κατηγορίες και φακέλους. Ένα από τα πλεονεκτήματα του Citavi είναι η ευρεία υποστήριξή του από ακαδημαϊκά ιδρύματα και βιβλιοθήκες. Ωστόσο, η πολυπλοκότητα της εφαρμογής και το κόστος αδειοδότησης μπορεί να αποτελέσουν περιορισμούς για ορισμένους χρήστες.

Το Citavi επιτρέπει στους χρήστες να εισάγουν βιβλιογραφικά δεδομένα από διάφορες πηγές και να τα οργανώνουν με τρόπο που διευκολύνει την ανάκτηση και τη χρήση τους. Οι χρήστες μπορούν να δημιουργήσουν προσαρμοσμένες κατηγορίες και φακέλους για να οργανώσουν τις βιβλιογραφικές αναφορές τους και να χρησιμοποιήσουν προηγμένα εργαλεία αναζήτησης για να βρουν γρήγορα τις αναφορές που χρειάζονται. Παρά τα πλεονεκτήματα αυτά, η πολυπλοκότητα της εφαρμογής και το κόστος αδειοδότησης μπορεί να αποτελέσουν

αποτρεπτικούς παράγοντες για ορισμένους χρήστες, ειδικά για μικρά ερευνητικά έργα ή για χρήστες με περιορισμένους πόρους.

2.3 Τεχνολογίες και Εργαλεία Ανάπτυξης

Για την ανάπτυξη του βιβλιογραφικού συστήματος, χρησιμοποιήθηκαν διάφορες γλώσσες προγραμματισμού και εργαλεία ανάπτυξης, τα οποία επιλέχθηκαν με βάση την απόδοσή τους, την ευκολία χρήσης και την επεκτασιμότητα.

2.4 Γλώσσες Προγραμματισμού

Οι κύριες γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για την ανάπτυξη του συστήματος περιλαμβάνουν την PHP για τον προγραμματισμό της λογικής του συστήματος και τη διαχείριση των δεδομένων, την JavaScript για την ανάπτυξη της διεπαφής χρήστη και την HTML/CSS για την παρουσίαση των δεδομένων [13]. Η επιλογή της PHP βασίστηκε στη μεγάλη ευελιξία και την ευρεία υποστήριξη που προσφέρει για την ανάπτυξη δυναμικών ιστοσελίδων και εφαρμογών. Η PHP είναι μια server-side γλώσσα προγραμματισμού που εκτελείται στον διακομιστή και μπορεί να διαχειριστεί αποτελεσματικά αιτήματα χρηστών και να αλληλεπιδρά με βάσεις δεδομένων, όπως η MySQL.

Η PHP είναι ιδιαίτερα γνωστή για την απλότητα της σύνταξής της και την ισχυρή της κοινότητα, που παρέχει πληθώρα βιβλιοθηκών και εργαλείων για την ανάπτυξη διαδικτυακών εφαρμογών. Πλαίσια όπως το Laravel και το CodeIgniter επιτρέπουν στους προγραμματιστές να αναπτύσσουν εφαρμογές πιο γρήγορα και με λιγότερη πολυπλοκότητα, ενώ παρέχουν ενσωματωμένα εργαλεία για διαχείριση χρηστών, ασφάλεια και επικοινωνία με βάσεις δεδομένων. Η PHP μπορεί επίσης να ενσωματωθεί εύκολα με άλλες τεχνολογίες όπως το JavaScript για την ανάπτυξη διαδραστικών διεπαφών και τη δημιουργία δυναμικών ιστοσελίδων.

Η συνεργασία της PHP με HTML, MySQL και CSS αποτελεί έναν από τους πιο δημοφιλείς συνδυασμούς για την ανάπτυξη διαδικτυακών εφαρμογών, προσφέροντας ένα ολοκληρωμένο και ευέλικτο περιβάλλον ανάπτυξης. Η αρμονική συνύπαρξη αυτών των τεχνολογιών επιτρέπει τη δημιουργία ισχυρών και διαδραστικών εφαρμογών, καλύπτοντας όλες τις ανάγκες τόσο από την πλευρά του διακομιστή όσο και από την πλευρά του πελάτη [14].

Συνεργασία της PHP με HTML

Η PHP (Hypertext Preprocessor) αποτελεί μια server-side γλώσσα προγραμματισμού, η οποία εκτελείται στον διακομιστή πριν αποσταλεί το αποτέλεσμα στον πελάτη. Αντίθετα, η HTML (HyperText Markup Language) είναι μια client-side γλώσσα σήμανσης που χρησιμοποιείται για τη δημιουργία της δομής και του περιεχομένου των ιστοσελίδων. Η συνεργασία της PHP με την HTML είναι ιδιαίτερα αποτελεσματική, καθώς επιτρέπει τη δημιουργία δυναμικού περιεχομένου που μπορεί να προσαρμόζεται ανάλογα με τις ανάγκες του χρήστη και τις απαιτήσεις της εφαρμογής. Για παράδειγμα, η PHP μπορεί να ανακτά δεδομένα από μια βάση δεδομένων και να τα ενσωματώνει σε στοιχεία HTML, όπως πίνακες και λίστες, επιτρέποντας την εμφάνιση πληροφοριών σε πραγματικό χρόνο.

Συνεργασία της PHP με MySQL

Η MySQL αποτελεί ένα από τα πιο διαδεδομένα συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) και συνεργάζεται άψογα με την PHP για τη διαχείριση και αποθήκευση δεδομένων. Η PHP διαθέτει ενσωματωμένες επεκτάσεις, όπως το mysqli (MySQL Improved) και το PDO (PHP Data Objects), που επιτρέπουν την εκτέλεση ερωτημάτων, την ανάκτηση δεδομένων και τη διαχείριση των δεδομένων με ασφαλή και αποδοτικό τρόπο. Αυτή η συνεργασία επιτρέπει τη δημιουργία εφαρμογών που υποστηρίζουν δυναμικά χαρακτηριστικά, όπως φόρμες εγγραφής, φόρουμ, συστήματα διαχείρισης περιεχομένου (CMS) και ηλεκτρονικά καταστήματα (e-shops). Οι δυνατότητες αλληλεπίδρασης της PHP με τη MySQL διευκολύνουν τη σύνθετη διαχείριση χρηστών, την εκτέλεση πολύπλοκων ερωτημάτων και την αποτελεσματική αποθήκευση και ανάκτηση πληροφοριών.

Συνεργασία της PHP με CSS

Η CSS (Cascading Style Sheets) είναι μια γλώσσα στυλ που χρησιμοποιείται για την περιγραφή της εμφάνισης και της διάταξης των στοιχείων HTML σε μια ιστοσελίδα. Μέσω της CSS, οι προγραμματιστές μπορούν να καθορίσουν την αισθητική των ιστοσελίδων, συμπεριλαμβανομένων των χρωμάτων, των γραμματοσειρών, των περιθωρίων και της διάταξης των στοιχείων. Η PHP μπορεί να ενσωματωθεί με την CSS για να δημιουργήσει δυναμικά στυλ, που προσαρμόζονται ανάλογα με τις συνθήκες ή τις επιλογές του χρήστη. Για παράδειγμα, η

PHP μπορεί να χρησιμοποιηθεί για την παραγωγή δυναμικών αρχείων CSS, τα οποία προσαρμόζουν την εμφάνιση της ιστοσελίδας σε πραγματικό χρόνο, ανταποκρινόμενα στις προτιμήσεις και τις ανάγκες του χρήστη. Αυτό επιτρέπει τη δημιουργία προσωποποιημένων και προσαρμόσιμων ιστοσελίδων που βελτιώνουν την εμπειρία του χρήστη.

Συνεργασία των PHP, HTML, MySQL και CSS σε Web Applications

Ο συνδυασμός PHP, HTML, MySQL και CSS προσφέρει ένα ισχυρό και ολοκληρωμένο περιβάλλον ανάπτυξης για την κατασκευή σύγχρονων διαδικτυακών εφαρμογών. Η PHP αναλαμβάνει τη διαχείριση της λογικής του συστήματος και της αλληλεπίδρασης με τη βάση δεδομένων, η HTML παρέχει τη δομή και τη διάταξη του περιεχομένου, η MySQL διαχειρίζεται την αποθήκευση και την ανάκτηση δεδομένων, ενώ η CSS καθορίζει την αισθητική παρουσίαση της ιστοσελίδας. Αυτή η συνεργασία επιτρέπει την ανάπτυξη δυναμικών και διαδραστικών εφαρμογών, που είναι εύκολες στη συντήρηση, επεκτάσιμες και φιλικές προς τον χρήστη.

Η χρήση αυτών των τεχνολογιών είναι ιδιαίτερα διαδεδομένη σε δημοφιλείς πλατφόρμες όπως το WordPress, το Joomla και το Drupal, καθώς και σε ηλεκτρονικά καταστήματα όπως το Magento και το OpenCart. Η αρμονική συνεργασία τους επιτρέπει την υλοποίηση προηγμένων λειτουργιών, την επεξεργασία μεγάλων όγκων δεδομένων και τη δημιουργία ελκυστικών και χρηστικών διεπαφών. Ως εκ τούτου, αυτές οι τεχνολογίες αποτελούν βασικά εργαλεία για την ανάπτυξη σύγχρονων και λειτουργικών διαδικτυακών εφαρμογών, καλύπτοντας τις απαιτήσεις τόσο των χρηστών όσο και των επιχειρήσεων.

Βάσεις Δεδομένων

Για τη διαχείριση των βιβλιογραφικών δεδομένων, χρησιμοποιήθηκε το PostgreSQL, ένα σύγχρονο σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων. Το PostgreSQL επιλέχθηκε λόγω της αξιοπιστίας του, της επεκτασιμότητάς του και της δυνατότητας υποστήριξης πολύπλοκων ερωτημάτων και δομών δεδομένων. Επιπλέον, η χρήση του ORM (Object-Relational Mapping) του Django διευκόλυνε τη διαχείριση της βάσης δεδομένων και την αλληλεπίδραση με αυτήν μέσω της Python.

Το PostgreSQL είναι γνωστό για την ισχυρή υποστήριξη για πολύπλοκα ερωτήματα και τη δυνατότητα διαχείρισης μεγάλων κλιμάκων δεδομένων. Παρέχει δυνατότητες για την ανάπτυξη προσαρμοσμένων λειτουργιών και την υποστήριξη πολλαπλών τύπων δεδομένων, καθιστώντας το ιδανικό για τη διαχείριση βιβλιογραφικών δεδομένων. Το ORM του Django επιτρέπει την ανάπτυξη κώδικα που είναι ανεξάρτητος από το συγκεκριμένο DBMS που χρησιμοποιείται, διευκολύνοντας την ανάπτυξη και τη συντήρηση του κώδικα [7].

Με την κατανόηση των παραπάνω βασικών εννοιών και τεχνολογιών, γίνεται σαφές πως η ανάπτυξη ενός συστήματος διαχείρισης βιβλιογραφικών δεδομένων απαιτεί τη χρήση σύγχρονων εργαλείων και μεθοδολογιών, προκειμένου να επιτευχθεί η καλύτερη δυνατή απόδοση και ευχρηστία.

2.5 Διεθνή Πρότυπα Βιβλιογραφίας

Τα διεθνή πρότυπα βιβλιογραφίας αποτελούν σύνολα κατευθυντήριων γραμμών που έχουν καθιερωθεί για την ομοιογενή καταγραφή και παρουσίαση των πηγών σε επιστημονικές και ακαδημαϊκές εργασίες. Η ανάγκη για την τυποποίηση των βιβλιογραφικών αναφορών προκύπτει από την επιθυμία της επιστημονικής κοινότητας να διασφαλίσει την ακρίβεια, την αξιοπιστία και τη δυνατότητα επαλήθευσης των πληροφοριών που περιέχονται σε μία εργασία. Τα πρότυπα αυτά περιλαμβάνουν, μεταξύ άλλων, το APA, το Harvard, το IEEE, το Vancouver και το MLA. Κάθε πρότυπο έχει αναπτυχθεί για να καλύψει τις ανάγκες διαφορετικών επιστημονικών κλάδων και να εξυπηρετήσει τις ειδικές απαιτήσεις τους [1].

2.5.1 Ιστορικό Ανάπτυξης των Προτύπων

Τα πρώτα εγχειρίδια που προτείνουν συγκεκριμένους τρόπους καταγραφής βιβλιογραφικών πηγών εμφανίστηκαν στα τέλη του 19ου αιώνα, σε μια περίοδο όπου η ακαδημαϊκή δημοσίευση άρχισε να γίνεται πιο συστηματική. Η ανάγκη για ακριβείς και επαληθεύσιμες αναφορές αυξήθηκε σταδιακά, καθώς η επιστημονική κοινότητα επιδίωξε να δημιουργήσει ένα κοινό πλαίσιο επικοινωνίας που να διασφαλίζει την ποιότητα και τη διαφάνεια των επιστημονικών έργων. Με την πάροδο των ετών και με την εξάπλωση των ακαδημαϊκών εκδόσεων σε παγκόσμιο επίπεδο, δημιουργήθηκαν συγκεκριμένα πρότυπα που υιοθετήθηκαν από ακαδημαϊκούς

οργανισμούς και εκδότες, τα οποία εξελίχθηκαν και συνεχίζουν να εξελίσσονται για να καλύπτουν τις απαιτήσεις της σύγχρονης έρευνας.

2.5.2 Σκοπός και Λειτουργία των Διεθνών Προτύπων

Ο πρωταρχικός σκοπός των διεθνών προτύπων βιβλιογραφίας είναι η καθιέρωση ενός συστήματος που επιτρέπει την ακριβή και συνεπή καταγραφή των πηγών, διευκολύνοντας την επαλήθευση των δεδομένων και την αναπαραγωγή της έρευνας. Μέσω της χρήσης αυτών των προτύπων, οι συγγραφείς αποδίδουν τις πηγές με τρόπο που ενισχύει τη διαφάνεια και την εγκυρότητα της ερευνητικής τους εργασίας, ενώ παράλληλα προστατεύονται από την κατηγορία της λογοκλοπής. Επιπλέον, τα πρότυπα αυτά καθοδηγούν τους αναγνώστες ώστε να εντοπίζουν εύκολα τις πηγές που χρησιμοποιήθηκαν, να αξιολογούν την ποιότητα των πληροφοριών και να προχωρούν σε περαιτέρω αναζητήσεις εάν αυτό είναι απαραίτητο [6].

2.5.3 Παρουσίαση Κύριων Διεθνών Προτύπων Βιβλιογραφίας

2.5.4 Πρότυπο APA (American Psychological Association)

Το πρότυπο APA αναπτύχθηκε από την Αμερικανική Ψυχολογική Εταιρεία (American Psychological Association) και χρησιμοποιείται κυρίως στις κοινωνικές και συμπεριφορικές επιστήμες. Το πρότυπο αυτό είναι σχεδιασμένο να προσφέρει σαφήνεια και απλότητα στην καταγραφή των πηγών, επιτρέποντας την εύκολη αναγνώριση των αναφορών στο κείμενο και τη βιβλιογραφία [2].

Ιστορικό και Εξέλιξη: Το APA πρωτοεμφανίστηκε το 1929 και έχει υποστεί πολλές αναθεωρήσεις, με την πιο πρόσφατη να είναι η 7η έκδοση, η οποία δημοσιεύθηκε το 2020. Η ανάγκη για τυποποιημένη μορφή αναφοράς προέκυψε από την πολυπλοκότητα των παραπομπών και την ανάγκη για ακριβή καταγραφή των πηγών σε επιστημονικές εργασίες.

Δομή Αναφοράς: Οι αναφορές στο κείμενο περιλαμβάνουν το επώνυμο του συγγραφέα και το έτος δημοσίευσης σε παρένθεση, π.χ., (Smith, 2020). Στη βιβλιογραφία, η αναφορά έχει τη μορφή: Smith, J. (2020). *Title of the book*. Publisher.

Χρήση και Πεδία Εφαρμογής: Το APA χρησιμοποιείται κυρίως σε εργασίες ψυχολογίας, κοινωνιολογίας, εκπαίδευσης και άλλων κοινωνικών επιστημών. Είναι ιδιαίτερα αποτελεσματικό

στην παρουσίαση εμπειρικών δεδομένων και την καταγραφή πηγών όπως άρθρα περιοδικών, βιβλία και ψηφιακές πηγές.

2.5.5 Harvard Style

Το Harvard είναι ένα γενικό σύστημα αναφοράς που δεν έχει ένα ενιαίο εγχειρίδιο, αλλά ακολουθεί κοινές αρχές για την καταγραφή των πηγών. Παρόλο που δεν υπάρχει ένας συγκεκριμένος οργανισμός που να είναι υπεύθυνος για τη διατήρηση και αναθεώρηση του, το Harvard είναι εξαιρετικά δημοφιλές σε πολλούς επιστημονικούς κλάδους.

Ιστορικό και Εξέλιξη: Η χρήση του Harvard ξεκίνησε στα μέσα του 20ού αιώνα και υιοθετήθηκε ευρέως από τα περισσότερα ακαδημαϊκά ιδρύματα λόγω της ευελιξίας του. Αν και δεν υπάρχει συγκεκριμένη έκδοση, οι βασικές του αρχές παραμένουν σταθερές.

Δομή Αναφοράς: Στο κείμενο, οι αναφορές έχουν τη μορφή (Συγγραφέας, Έτος), π.χ., (Brown, 2015). Στη βιβλιογραφία, η μορφή είναι: Brown, A. (2015). *Title of the book*. Publisher.

Χρήση και Πεδία Εφαρμογής: Το Harvard χρησιμοποιείται σε ευρύ φάσμα επιστημονικών τομέων, από τις ανθρωπιστικές μέχρι τις φυσικές επιστήμες. Το σύστημα αυτό είναι κατάλληλο για εργασίες που περιλαμβάνουν πηγές από διάφορες κατηγορίες, όπως άρθρα, βιβλία και διαδικτυακές πηγές.

2.5.6 Πρότυπο IEEE (Institute of Electrical and Electronics Engineers)

Το πρότυπο IEEE εφαρμόζεται κυρίως στις επιστήμες της μηχανικής και της πληροφορικής. Βασίζεται σε αριθμητικές παραπομπές στο κείμενο και παρέχει μια σαφή και ακριβή δομή για την καταγραφή των τεχνικών και επιστημονικών πηγών [5].

Ιστορικό και Εξέλιξη: Το IEEE καθιερώθηκε για να εξυπηρετήσει τις ανάγκες της τεχνικής και επιστημονικής κοινότητας, ειδικά στις δημοσιεύσεις που αφορούν ηλεκτρολόγους μηχανικούς και ειδικούς πληροφορικής. Το πρότυπο αυτό έχει εξελιχθεί για να καλύψει την ανάγκη για λεπτομερή και ακριβή καταγραφή των πηγών.

Δομή Αναφοράς: Στο κείμενο, οι αναφορές γίνονται με αριθμητική σειρά, π.χ., [1], [2], κ.λπ. Στη βιβλιογραφία, η πλήρης αναφορά περιλαμβάνει το όνομα του συγγραφέα, τον τίτλο του άρθρου, το όνομα του περιοδικού, τον αριθμό του τεύχους, το έτος δημοσίευσης και τις σελίδες.

Χρήση και Πεδία Εφαρμογής: Το IEEE χρησιμοποιείται ευρέως στις τεχνικές επιστήμες, όπως η μηχανική, η πληροφορική και οι τηλεπικοινωνίες. Είναι ιδανικό για εργασίες με πολλές παραπομπές σε τεχνικά άρθρα και συνεδριακές εργασίες.

2.5.7 Πρότυπο Vancouver

Το πρότυπο Vancouver είναι ευρέως διαδεδομένο στις ιατρικές και βιολογικές επιστήμες. Χρησιμοποιεί αριθμητικές παραπομπές και είναι κατάλληλο για την παρουσίαση κλινικών και επιστημονικών δεδομένων [4].

Ιστορικό και Εξέλιξη: Το Vancouver αναπτύχθηκε από τη Διεθνή Επιτροπή Ιατρικών Συντακτών το 1978 για να δημιουργηθεί ένα κοινό πρότυπο για τη συγγραφή ιατρικών άρθρων. Το πρότυπο αυτό έχει αναγνωριστεί από την ιατρική κοινότητα και χρησιμοποιείται παγκοσμίως.

Δομή Αναφοράς: Στο κείμενο, οι αναφορές γίνονται με αριθμητική σειρά, π.χ., [1]. Στη βιβλιογραφία, η αναφορά περιλαμβάνει το όνομα του συγγραφέα, τον τίτλο του άρθρου, το όνομα του περιοδικού, το έτος δημοσίευσης, τον αριθμό του τεύχους και τις σελίδες.

Χρήση και Πεδία Εφαρμογής: Το Vancouver είναι ιδανικό για ιατρικά και βιολογικά άρθρα, κλινικές μελέτες και έρευνες. Είναι αποδεκτό από τα περισσότερα ιατρικά περιοδικά και συνεδριακές δημοσιεύσεις.

2.5.8 Πρότυπο MLA (Modern Language Association)

Το πρότυπο MLA χρησιμοποιείται κυρίως στις ανθρωπιστικές επιστήμες, ιδιαίτερα στις φιλολογικές και γλωσσολογικές σπουδές. Εστιάζει στην ευανάγνωστη παρουσίαση των λογοτεχνικών και κειμενικών πηγών [3].

Ιστορικό και Εξέλιξη: Το MLA δημιουργήθηκε από την Modern Language Association και έχει ενημερωθεί πολλές φορές, με την πιο πρόσφατη έκδοση να είναι η 8η, που κυκλοφόρησε το 2016.

Δομή Αναφοράς: Στο κείμενο, η αναφορά περιλαμβάνει το επώνυμο του συγγραφέα και τον αριθμό της σελίδας, π.χ., (Smith 23). Στη βιβλιογραφία, η μορφή είναι: Smith, John. *Title of the Book*. Publisher, 2020.

Χρήση και Πεδία Εφαρμογής: Το MLA είναι κατάλληλο για τις ανθρωπιστικές και φιλολογικές σπουδές, όπου η καταγραφή των σελίδων και η ακριβής απόδοση των κειμενικών πηγών είναι ζωτικής σημασίας.

2.6 Συγκριτική Ανάλυση των Διεθνών Προτύπων

Κάθε πρότυπο βιβλιογραφίας προσφέρει διαφορετικές προσεγγίσεις στην καταγραφή των πηγών. Το APA και το Harvard προσφέρουν ευελιξία και είναι κατάλληλα για τις κοινωνικές επιστήμες, ενώ το IEEE και το Vancouver είναι ιδανικά για τις τεχνικές και ιατρικές επιστήμες λόγω της αριθμητικής τους προσέγγισης. Το MLA, από την άλλη πλευρά, είναι εξειδικευμένο για τις ανθρωπιστικές σπουδές. Παρά τις διαφορές τους, όλα τα πρότυπα αποσκοπούν στη διασφάλιση της σαφήνειας και της ακρίβειας στην απόδοση των πηγών, επιτρέποντας στους αναγνώστες να εντοπίζουν και να αξιολογούν την εγκυρότητα των πληροφοριών.

Τα διεθνή πρότυπα βιβλιογραφίας αποτελούν θεμελιώδες εργαλείο για την ακαδημαϊκή κοινότητα. Η σωστή χρήση τους εξασφαλίζει την αξιοπιστία και την ποιότητα της επιστημονικής έρευνας, προωθώντας την ακαδημαϊκή ακεραιότητα και διευκολύνοντας την επιστημονική επικοινωνία. Κάθε πρότυπο έχει τις δικές του εφαρμογές και χαρακτηριστικά, και η επιλογή του κατάλληλου προτύπου εξαρτάται από το επιστημονικό πεδίο και τις απαιτήσεις της εκάστοτε δημοσίευσης.

3 Ανάλυση Απαιτήσεων

Το κεφάλαιο αυτό επικεντρώνεται στην ανάλυση των απαιτήσεων για την ανάπτυξη του βιβλιογραφικού συστήματος διαχείρισης δεδομένων. Η ανάλυση απαιτήσεων είναι κρίσιμη για την επιτυχή υλοποίηση οποιουδήποτε συστήματος, καθώς καθορίζει τις λειτουργικές και μη λειτουργικές προδιαγραφές που πρέπει να καλυφθούν. Θα εξετάσουμε τις βασικές λειτουργικές απαιτήσεις, τις μη λειτουργικές απαιτήσεις και τις κατηγορίες των χρηστών του συστήματος, περιγράφοντας τις ανάγκες και τις απαιτήσεις τους.

3.1 Λειτουργικές Απαιτήσεις

Οι λειτουργικές απαιτήσεις αφορούν τις συγκεκριμένες λειτουργίες που πρέπει να υποστηρίζει το σύστημα για να καλύψει τις ανάγκες των χρηστών του. Αυτές οι απαιτήσεις περιλαμβάνουν την εισαγωγή, αποθήκευση, οργάνωση, ανάκτηση και διαμοιρασμό βιβλιογραφικών δεδομένων, καθώς και τη διαχείριση χρηστών και τη δημιουργία βιβλιογραφικών καταλόγων.

Εισαγωγή Δεδομένων

Το σύστημα πρέπει να υποστηρίζει την εισαγωγή βιβλιογραφικών δεδομένων από διάφορες πηγές, όπως αρχεία PDF, ιστοσελίδες και βάσεις δεδομένων. Η αυτόματη εξαγωγή μεταδεδομένων από αρχεία PDF είναι ιδιαίτερα σημαντική για την εξοικονόμηση χρόνου και την αποφυγή λαθών κατά την εισαγωγή δεδομένων.

Αποθήκευση και Οργάνωση Δεδομένων

Τα βιβλιογραφικά δεδομένα πρέπει να αποθηκεύονται σε μια δομημένη βάση δεδομένων, που να επιτρέπει την εύκολη οργάνωση και κατηγοριοποίησή τους. Το σύστημα πρέπει να υποστηρίζει τη δημιουργία προσαρμοσμένων κατηγοριών και φακέλων για την οργάνωση των δεδομένων, καθώς και τη δυνατότητα επισύναψης σημειώσεων και ετικετών σε κάθε εγγραφή.

Ανάκτηση Δεδομένων

Η ανάκτηση δεδομένων είναι μια βασική λειτουργία του συστήματος, που πρέπει να υποστηρίζει γρήγορη και ακριβή αναζήτηση βιβλιογραφικών αναφορών με βάση διάφορα κριτήρια, όπως ο τίτλος, ο συγγραφέας, το έτος δημοσίευσης και οι λέξεις-κλειδιά. Επιπλέον, το

σύστημα πρέπει να παρέχει προηγμένα εργαλεία φιλτραρίσματος και ταξινόμησης για την ευκολότερη εύρεση των ζητούμενων αναφορών.

Διαμοιρασμός Δεδομένων

Το σύστημα πρέπει να υποστηρίζει τη δυνατότητα διαμοιρασμού βιβλιογραφικών αναφορών με άλλους χρήστες, τόσο εντός όσο και εκτός του οργανισμού. Αυτή η δυνατότητα είναι ιδιαίτερα σημαντική για τη συνεργασία σε ερευνητικά έργα και για την ανταλλαγή πληροφοριών μεταξύ ερευνητών.

Διαχείριση Χρηστών

Η διαχείριση χρηστών περιλαμβάνει τη δυνατότητα δημιουργίας, επεξεργασίας και διαγραφής λογαριασμών χρηστών, καθώς και τη διαχείριση των δικαιωμάτων πρόσβασης. Το σύστημα πρέπει να υποστηρίζει πολλαπλά επίπεδα πρόσβασης, ανάλογα με το ρόλο του χρήστη, όπως διαχειριστές, ερευνητές και επισκέπτες.

Δημιουργία Βιβλιογραφικών Καταλόγων

Το σύστημα πρέπει να παρέχει τη δυνατότητα δημιουργίας βιβλιογραφικών καταλόγων σε διάφορες μορφές αναφοράς, όπως το APA, το MLA και το Chicago. Οι χρήστες πρέπει να μπορούν να εξάγουν τους βιβλιογραφικούς καταλόγους σε μορφές όπως το BibTeX, το EndNote και το RIS, διευκολύνοντας έτσι την ενσωμάτωσή τους σε ακαδημαϊκές εργασίες και δημοσιεύσεις.

3.2 Μη Λειτουργικές Απαιτήσεις

Οι μη λειτουργικές απαιτήσεις αφορούν τα χαρακτηριστικά ποιότητας του συστήματος, όπως η απόδοση, η ασφάλεια, η επεκτασιμότητα και η ευχρηστία. Αυτές οι απαιτήσεις είναι εξίσου σημαντικές με τις λειτουργικές, καθώς επηρεάζουν την αποδοτικότητα και την εμπειρία των χρηστών.

Απόδοση

Το σύστημα πρέπει να είναι σε θέση να διαχειρίζεται μεγάλες ποσότητες βιβλιογραφικών δεδομένων χωρίς σημαντική καθυστέρηση. Η ταχύτητα απόκρισης στις αναζητήσεις και οι

λειτουργίες ανάκτησης δεδομένων πρέπει να είναι βελτιστοποιημένες για να παρέχουν γρήγορα αποτελέσματα στους χρήστες.

Ασφάλεια

Η ασφάλεια των δεδομένων είναι κρίσιμη για τη διαχείριση βιβλιογραφικών αναφορών, ειδικά όταν αυτές περιλαμβάνουν ευαίσθητες πληροφορίες. Το σύστημα πρέπει να υποστηρίζει προηγμένες λειτουργίες ασφάλειας, όπως η κρυπτογράφηση δεδομένων, η διαχείριση ταυτοποίησης και εξουσιοδότησης χρηστών, και η καταγραφή των δραστηριοτήτων των χρηστών [10].

Επεκτασιμότητα

Το σύστημα πρέπει να είναι σχεδιασμένο με τρόπο που να επιτρέπει την εύκολη επεκτασιμότητα, ώστε να μπορεί να διαχειρίζεται αυξανόμενο όγκο δεδομένων και χρήστες χωρίς να επηρεάζεται η απόδοση. Η υποστήριξη για την προσθήκη νέων λειτουργιών και η δυνατότητα ενσωμάτωσης με άλλα συστήματα είναι επίσης σημαντικές.

Ευχρηστία

Η ευχρηστία του συστήματος είναι κρίσιμη για την αποδοτική χρήση του από τους χρήστες. Το σύστημα πρέπει να έχει μια φιλική προς τον χρήστη διεπαφή, να παρέχει καθοδηγητικά μηνύματα και να είναι εύκολο στη χρήση χωρίς να απαιτείται εκτενής εκπαίδευση. Η διεπαφή χρήστη πρέπει να είναι σχεδιασμένη με βάση τις αρχές της εργονομίας και της ευχρηστίας, διευκολύνοντας την πλοήγηση και την αλληλεπίδραση με το σύστημα.

Αξιοπιστία

Το σύστημα πρέπει να είναι αξιόπιστο και να λειτουργεί απρόσκοπτα χωρίς διακοπές ή σφάλματα. Η ανθεκτικότητα σε αστοχίες και η δυνατότητα ανάκαμψης από σφάλματα είναι κρίσιμα στοιχεία για την αξιοπιστία του συστήματος. Επίσης, η διαθεσιμότητα των δεδομένων πρέπει να είναι συνεχής και χωρίς διακοπές.

Συμβατότητα

Το σύστημα πρέπει να είναι συμβατό με διάφορες πλατφόρμες και προγράμματα επεξεργασίας κειμένου, όπως το Microsoft Word, το LaTeX και άλλες εφαρμογές που χρησιμοποιούνται ευρέως στην ακαδημαϊκή κοινότητα. Η υποστήριξη για πρότυπα αναφοράς και μορφές εξαγωγής δεδομένων είναι επίσης σημαντική για τη διασφάλιση της διαλειτουργικότητας με άλλες εφαρμογές.

3.3 Χρήστες του Συστήματος

Οι χρήστες του συστήματος μπορούν να κατηγοριοποιηθούν σε διάφορες ομάδες, ανάλογα με τις ανάγκες και τις απαιτήσεις τους. Καθεμία από αυτές τις ομάδες χρηστών έχει διαφορετικές ανάγκες και απαιτεί συγκεκριμένες λειτουργίες από το σύστημα [11].

Ερευνητές

Οι ερευνητές είναι η κύρια ομάδα χρηστών του συστήματος και έχουν ανάγκη για την αποθήκευση, οργάνωση και ανάκτηση βιβλιογραφικών δεδομένων. Οι ερευνητές χρειάζονται προηγμένα εργαλεία αναζήτησης και φιλτραρίσματος για την εύκολη ανάκτηση των βιβλιογραφικών αναφορών, καθώς και τη δυνατότητα δημιουργίας βιβλιογραφικών καταλόγων σε διάφορες μορφές αναφοράς.

Φοιτητές

Οι φοιτητές χρησιμοποιούν το σύστημα κυρίως για τη διαχείριση των βιβλιογραφικών αναφορών που απαιτούνται για τις ακαδημαϊκές τους εργασίες και τις διατριβές. Χρειάζονται εύκολα στη χρήση εργαλεία εισαγωγής δεδομένων και δυνατότητες οργάνωσης και ανάκτησης που θα τους βοηθήσουν να διαχειρίζονται τις βιβλιογραφικές αναφορές τους αποτελεσματικά.

Διαχειριστές

Οι διαχειριστές του συστήματος έχουν την ευθύνη για τη διαχείριση των χρηστών, την ασφάλεια των δεδομένων και τη συντήρηση του συστήματος. Χρειάζονται προηγμένα εργαλεία διαχείρισης χρηστών, καθώς και δυνατότητες παρακολούθησης και καταγραφής δραστηριοτήτων για τη διασφάλιση της ασφάλειας και της ακεραιότητας των δεδομένων.

Εκδότες και Βιβλιοθήκες

Οι εκδότες και οι βιβλιοθήκες μπορούν να χρησιμοποιήσουν το σύστημα για τη διαχείριση και τη διάθεση βιβλιογραφικών δεδομένων. Χρειάζονται δυνατότητες οργάνωσης και κατηγοριοποίησης δεδομένων, καθώς και εργαλεία διαμοιρασμού για την εύκολη πρόσβαση και ανάκτηση των βιβλιογραφικών αναφορών από τους χρήστες.

Με την κατανόηση των παραπάνω απαιτήσεων, το σύστημα μπορεί να σχεδιαστεί και να αναπτυχθεί με τρόπο που να καλύπτει τις ανάγκες όλων των ομάδων χρηστών, παρέχοντας τα απαραίτητα εργαλεία και λειτουργίες για την αποδοτική διαχείριση βιβλιογραφικών δεδομένων.

4 Σχεδίαση του Συστήματος

Η σχεδίαση ενός συστήματος διαχείρισης βιβλιογραφικών αναφορών είναι ένα κρίσιμο στάδιο που περιλαμβάνει την ανάλυση του ER διαγράμματος, τη μετατροπή του σε σχεσιακό σχήμα, και τη σχεδίαση της διεπαφής χρήστη. Σε αυτό το κεφάλαιο, θα εξετάσουμε λεπτομερώς την αρχιτεκτονική του συστήματος, τις οντότητες και τις σχέσεις τους, και τη δομή της βάσης δεδομένων.

4.1 Ανάλυση του Συστήματος

Η ανάλυση του συστήματος περιλαμβάνει την περιγραφή των κύριων λειτουργιών και των απαιτήσεων του συστήματος, καθώς και τον προσδιορισμό των βασικών στοιχείων της αρχιτεκτονικής του.

Το σύστημα χρησιμοποιεί μια σχεσιακή βάση δεδομένων για να οργανώσει τα δεδομένα. Οι κύριες οντότητες της βάσης δεδομένων περιλαμβάνουν τους συγγραφείς, τα άρθρα, τα βιβλία, τις διπλωματικές εργασίες, τις ιστοσελίδες, και τους οργανισμούς. Οι σχέσεις μεταξύ αυτών των οντοτήτων είναι θεμελιώδεις για την οργάνωση των βιβλιογραφικών δεδομένων και την υποστήριξη των λειτουργιών του συστήματος [8].

- **Σχέσεις Πολλά προς Πολλά:** Οι σχέσεις πολλών προς πολλών, ειδικά μεταξύ συγγραφέων και δημοσιεύσεων, διαχειρίζονται μέσω ενδιάμεσων πινάκων, εξασφαλίζοντας την πλήρη καταγραφή της συμμετοχής των συγγραφέων σε διάφορα έργα.
- **Σχέσεις Πολλά προς Ένα:** Πολλές οντότητες, όπως τα άρθρα και οι ιστοσελίδες, συνδέονται με περιοδικά, εκδότες, και οργανισμούς μέσω σχέσεων πολλά προς ένα, διευκολύνοντας την αναγνώριση της προέλευσης των δεδομένων.

Κύριες Λειτουργίες του Συστήματος:

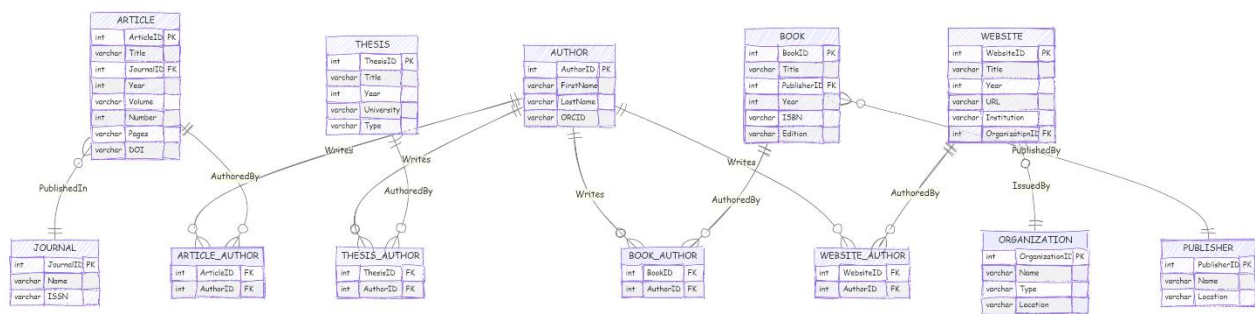
1. **Εισαγωγή Βιβλιογραφικών Δεδομένων:** Το σύστημα θα πρέπει να υποστηρίζει την εισαγωγή δεδομένων από διάφορες πηγές, όπως αρχεία PDF και βάσεις δεδομένων.
2. **Οργάνωση Δεδομένων:** Οι χρήστες θα μπορούν να οργανώνουν τις βιβλιογραφικές αναφορές τους σε κατηγορίες και φακέλους, προσθέτοντας ετικέτες και σημειώσεις.

3. **Ανάκτηση Δεδομένων:** Το σύστημα θα πρέπει να προσφέρει προηγμένα εργαλεία αναζήτησης και φιλτραρίσματος, επιτρέποντας την εύκολη και γρήγορη ανάκτηση των αναφορών.
4. **Διαμοιρασμός Δεδομένων:** Το σύστημα πρέπει να υποστηρίζει τη δυνατότητα διαμοιρασμού βιβλιογραφικών αναφορών με άλλους χρήστες.
5. **Διαχείριση Χρηστών:** Η διαχείριση χρηστών περιλαμβάνει τη δημιουργία, επεξεργασία και διαγραφή λογαριασμών, καθώς και την ανάθεση ρόλων και δικαιωμάτων πρόσβασης.

4.2 Διάγραμμα Οντοτήτων-Συσχετίσεων (ER)

Το Διάγραμμα Οντοτήτων-Συσχετίσεων (ER) αποτελεί ένα από τα πιο σημαντικά εργαλεία στη φάση της σχεδίασης βάσεων δεδομένων. Αντικατοπτρίζει τη δομή της βάσης δεδομένων μέσω της απεικόνισης των οντοτήτων και των σχέσεων τους, προσφέροντας μια οπτική αναπαράσταση που διευκολύνει την κατανόηση και την ανάλυση του συστήματος. Στο πλαίσιο της διαχείρισης βιβλιογραφικών αναφορών, το ER διάγραμμα βοηθάει στην κατηγοριοποίηση και οργάνωση των βιβλιογραφικών δεδομένων, καθιστώντας σαφές ποια δεδομένα χρειάζονται αποθήκευση και πώς αυτά συνδέονται μεταξύ τους [8].

Στο διάγραμμα αυτό, κάθε οντότητα αντιπροσωπεύει έναν τύπο αντικειμένου για τον οποίο αποθηκεύονται δεδομένα. Για παράδειγμα, οι οντότητες μπορεί να περιλαμβάνουν άρθρα, βιβλία, περιοδικά, συγγραφείς και εκδότες. Οι σχέσεις μεταξύ των οντοτήτων καθορίζουν τον τρόπο με τον οποίο αυτά τα δεδομένα συνδέονται. Οι σχέσεις μπορούν να είναι "ένα προς πολλά" ή "πολλά προς πολλά", ανάλογα με το πως συνδέονται τα δεδομένα μεταξύ τους.



Εικόνα 1: Το ER διάγραμμα με όλες τις οντότητες και τις σχέσεις

Η ανάλυση του ER διαγράμματος μας επιτρέπει να εντοπίσουμε τις απαιτήσεις αποθήκευσης δεδομένων και να σχεδιάσουμε το σχεσιακό σχήμα της βάσης δεδομένων που θα χρησιμοποιηθεί στην υλοποίηση του συστήματος. Αυτό διασφαλίζει ότι η βάση δεδομένων θα είναι καλά δομημένη, επεκτάσιμη και ικανή να υποστηρίξει τις λειτουργικές απαιτήσεις του συστήματος

Κύριες Οντότητες

Authors (Συγγραφείς):

Η οντότητα Authors αποτελεί βασικό στοιχείο της βιβλιογραφικής βάσης δεδομένων, καθώς κάθε επιστημονικό έργο απαιτεί την απόδοση σε έναν ή περισσότερους συγγραφείς. Αυτή η οντότητα αποθηκεύει πληροφορίες για τους συγγραφείς, επιτρέποντας την καταγραφή και την παρακολούθηση των συνεισφορών τους. Η χρήση του ORCID (Open Researcher and Contributor ID) διασφαλίζει την ακριβή ταυτοποίηση των ερευνητών σε παγκόσμιο επίπεδο, διευκολύνοντας την αναγνώριση των συνεισφορών τους σε διάφορες δημοσιεύσεις.

Πεδία:

- **AuthorID**: Μοναδικός Αναγνωριστικός Αριθμός Συγγραφέα. Χρησιμοποιείται για την ταυτοποίηση κάθε συγγραφέα στη βάση δεδομένων.
- **FirstName**: Όνομα Συγγραφέα. Καταγράφει το μικρό όνομα του συγγραφέα.
- **LastName**: Επώνυμο Συγγραφέα. Καταγράφει το επώνυμο του συγγραφέα.

- ORCID: Open Researcher and Contributor ID. Ένας μοναδικός αριθμός που χρησιμοποιείται για την αναγνώριση των συγγραφέων παγκοσμίως.

Articles (Άρθρα):

Η οντότητα Articles αποθηκεύει δεδομένα για άρθρα που δημοσιεύονται σε επιστημονικά περιοδικά. Τα άρθρα είναι κεντρικά στην επιστημονική έρευνα και η καταγραφή τους είναι απαραίτητη για την αναγνώριση των ερευνητικών επιτευγμάτων και την αναφορά τους. Η χρήση του DOI (Digital Object Identifier) επιτρέπει την ασφαλή και μόνιμη πρόσβαση στα άρθρα, εξασφαλίζοντας την αναγνωσιμότητα και την αξιοπιστία τους.

Πεδία:

- ArticleID: Μοναδικός Αναγνωριστικός Αριθμός Άρθρου. Χρησιμοποιείται για την ταυτοποίηση κάθε άρθρου.
- Title: Τίτλος Άρθρου. Καταγράφει τον τίτλο του άρθρου όπως δημοσιεύθηκε.
- JournalID: Αναγνωριστικός Αριθμός Περιοδικού. Σύνδεση με την οντότητα Journals, δείχνοντας το περιοδικό όπου δημοσιεύθηκε το άρθρο.
- Year: Έτος Δημοσίευσης. Δείχνει το έτος κατά το οποίο δημοσιεύθηκε το άρθρο.
- Volume: Όγκος Περιοδικού. Καταγράφει τον όγκο του περιοδικού στον οποίο δημοσιεύθηκε το άρθρο.
- Number: Αριθμός Τεύχους Περιοδικού. Καταγράφει τον αριθμό του τεύχους του περιοδικού.
- Pages: Σελίδες Άρθρου. Καταγράφει το εύρος των σελίδων όπου εμφανίζεται το άρθρο.
- DOI: Ψηφιακός Αναγνωριστής Αντικειμένου. Μοναδικός αριθμός που χρησιμοποιείται για την άμεση και μόνιμη αναγνώριση του άρθρου.

Books (Βιβλία):

Η οντότητα Books περιλαμβάνει πληροφορίες για βιβλία που εκδίδονται από εκδοτικούς οίκους. Τα βιβλία αποτελούν πηγή έγκυρης πληροφόρησης και η σωστή καταγραφή τους βοηθά στην

οργάνωση της βιβλιογραφίας. Το ISBN (International Standard Book Number) χρησιμοποιείται για την παγκόσμια ταυτοποίηση των βιβλίων, διευκολύνοντας την εύρεση και αναφορά τους.

Πεδία:

- BookID: Μοναδικός Αναγνωριστικός Αριθμός Βιβλίου. Χρησιμοποιείται για την ταυτοποίηση κάθε βιβλίου.
- Title: Τίτλος Βιβλίου. Καταγράφει τον τίτλο του βιβλίου.
- PublisherID: Αναγνωριστικός Αριθμός Εκδότη. Σύνδεση με την οντότητα Publishers, καθορίζει τον εκδότη του βιβλίου.
- Year: Έτος Έκδοσης. Δείχνει το έτος κατά το οποίο εκδόθηκε το βιβλίο.
- ISBN: Διεθνής Αριθμός Βιβλίου. Ένας μοναδικός αριθμός που χρησιμοποιείται για την ταυτοποίηση του βιβλίου.
- Edition: Έκδοση Βιβλίου. Καταγράφει την συγκεκριμένη έκδοση του βιβλίου.

Journals (Περιοδικά):

Η οντότητα Journals αποθηκεύει πληροφορίες για επιστημονικά περιοδικά. Τα περιοδικά είναι μέσα δημοσίευσης επιστημονικών άρθρων και η σωστή καταγραφή τους είναι απαραίτητη για την αναγνώριση και την αναφορά των άρθρων. Το ISSN (International Standard Serial Number) είναι ένας μοναδικός αριθμός που χρησιμοποιείται για την αναγνώριση των περιοδικών σε διεθνές επίπεδο.

Πεδία:

- JournalID: Μοναδικός Αναγνωριστικός Αριθμός Περιοδικού. Χρησιμοποιείται για την ταυτοποίηση κάθε περιοδικού.
- Name: Όνομα Περιοδικού. Καταγράφει το όνομα του περιοδικού.
- ISSN: Διεθνής Αριθμός Αναγνώρισης Περιοδικού. Ένας μοναδικός αριθμός που αναγνωρίζει το περιοδικό.

Publishers (Εκδότες):

Η οντότητα Publishers περιλαμβάνει πληροφορίες για τους εκδοτικούς οίκους. Οι εκδότες παίζουν σημαντικό ρόλο στη διάδοση των βιβλίων και άλλων δημοσιεύσεων. Η καταγραφή των εκδοτών είναι κρίσιμη για την οργάνωση των βιβλιογραφικών δεδομένων και την αναγνώριση των εκδοτικών δραστηριοτήτων.

Πεδία:

- PublisherID: Μοναδικός Αναγνωριστικός Αριθμός Εκδότη. Χρησιμοποιείται για την ταυτοποίηση κάθε εκδότη.
- Name: Όνομα Εκδότη. Καταγράφει το όνομα του εκδότη.
- Location: Τοποθεσία Εκδότη. Καταγράφει τη γεωγραφική τοποθεσία ή τη διεύθυνση του εκδοτικού οίκου.

Theses (Διπλωματικές Εργασίες):

Η οντότητα Theses καταγράφει διπλωματικές και διδακτορικές εργασίες. Αυτές οι εργασίες είναι σημαντικές ακαδημαϊκές συνεισφορές και η σωστή καταγραφή τους είναι απαραίτητη για την αναφορά και την αναγνώρισή τους. Η αναφορά σε διπλωματικές και διδακτορικές εργασίες υποστηρίζει την ακαδημαϊκή έρευνα και παρέχει πρόσβαση σε νέες ιδέες και ευρήματα.

Πεδία:

- ThesisID: Μοναδικός Αναγνωριστικός Αριθμός Διπλωματικής Εργασίας. Χρησιμοποιείται για την ταυτοποίηση κάθε εργασίας.
- Title: Τίτλος Εργασίας. Καταγράφει τον τίτλο της διπλωματικής ή διδακτορικής εργασίας.
- Year: Έτος Ολοκλήρωσης. Δείχνει το έτος κατά το οποίο ολοκληρώθηκε η εργασία.
- University: Πανεπιστήμιο. Καταγράφει το πανεπιστήμιο στο οποίο υποβλήθηκε η εργασία.
- Type: Τύπος Εργασίας. Περιγράφει αν η εργασία είναι διπλωματική ή διδακτορική.

Website (Ιστοσελίδα):

Η οντότητα Website είναι η νέα προσθήκη που αντικαθιστά την οντότητα Documents. Καταγράφει πληροφορίες για ιστοσελίδες, περιλαμβάνοντας δεδομένα όπως τον τίτλο της ιστοσελίδας, τη διεύθυνση URL, το έτος δημοσίευσης ή τελευταίας ενημέρωσης, και το ίδρυμα που φιλοξενεί την ιστοσελίδα. Η καταγραφή αυτών των πληροφοριών είναι κρίσιμη για την αναφορά διαδικτυακών πηγών, καθώς οι ιστοσελίδες αποτελούν συχνή πηγή πληροφόρησης στη σύγχρονη βιβλιογραφία.

Πεδία:

- WebsiteID: Μοναδικός Αναγνωριστικός Αριθμός Ιστοσελίδας. Χρησιμοποιείται για την ταυτοποίηση κάθε ιστοσελίδας.
- Title: Τίτλος Ιστοσελίδας. Καταγράφει τον τίτλο της ιστοσελίδας.
- Year: Έτος Δημοσίευσης/Ενημέρωσης. Δείχνει το έτος κατά το οποίο δημοσιεύθηκε ή ενημερώθηκε η ιστοσελίδα.
- URL: Διεύθυνση Διαδικτύου. Καταγράφει τη διεύθυνση URL της ιστοσελίδας.
- Institution: Ίδρυμα. Καταγράφει το όνομα του ιδρύματος ή του οργανισμού που φιλοξενεί την ιστοσελίδα.
- OrganizationID: Αναγνωριστικός Αριθμός Οργανισμού. Συνδέεται με την οντότητα Organizations, αναγνωρίζοντας τον οργανισμό που σχετίζεται με την ιστοσελίδα.

Organizations (Οργανισμοί):

Η οντότητα Organizations καταγράφει πληροφορίες για οργανισμούς που σχετίζονται με την έκδοση ιστοσελίδων ή άλλων επιστημονικών αναφορών. Αυτή η οντότητα περιλαμβάνει ακαδημαϊκά ιδρύματα, ερευνητικά κέντρα, και άλλους φορείς που συμβάλλουν στην παραγωγή και διάδοση της επιστημονικής γνώσης.

Πεδία:

- **OrganizationID:** Μοναδικός Αναγνωριστικός Αριθμός Οργανισμού. Χρησιμοποιείται για την ταυτοποίηση κάθε οργανισμού.
- **Name:** Όνομα Οργανισμού. Καταγράφει το όνομα του οργανισμού.
- **Type:** Τύπος Οργανισμού. Περιγράφει τον τύπο του οργανισμού (π.χ., Πανεπιστήμιο, Ερευνητικό Κέντρο).
- **Location:** Τοποθεσία Οργανισμού. Καταγράφει τη γεωγραφική τοποθεσία ή τη διεύθυνση του οργανισμού.

Σχέσεις:

Οι σχέσεις μεταξύ των οντοτήτων καθορίζουν πώς συνδέονται τα δεδομένα στη βάση και επιτρέπουν την πλήρη και συνεπή καταγραφή των βιβλιογραφικών αναφορών. Οι σχέσεις αυτές είναι κρίσιμες για τη διασφάλιση της ορθής διαχείρισης των δεδομένων και την υποστήριξη της δημιουργίας αναφορών.

Άρθρα - Περιοδικά (Articles - Journals):

- **Τύπος Σχέσης:** Πολλά προς ένα.
- **Περιγραφή:** Κάθε άρθρο δημοσιεύεται σε ένα μόνο περιοδικό, ενώ ένα περιοδικό μπορεί να περιλαμβάνει πολλά άρθρα. Αυτή η σχέση επιτρέπει την παρακολούθηση και την καταγραφή των άρθρων που δημοσιεύονται σε συγκεκριμένα περιοδικά.
- **Υλοποίηση:** Χρησιμοποιείται το ξένο κλειδί JournalID στον πίνακα ARTICLE για να συνδεθεί με το JournalID στον πίνακα JOURNAL, καθορίζοντας το περιοδικό όπου δημοσιεύθηκε το άρθρο.

Βιβλία - Εκδότες (Books - Publishers):

- **Τύπος Σχέσης:** Πολλά προς ένα.

- **Περιγραφή:** Κάθε βιβλίο εκδίδεται από έναν μόνο εκδότη, αλλά ένας εκδότης μπορεί να εκδίδει πολλά βιβλία. Αυτή η σχέση είναι σημαντική για την καταγραφή των βιβλίων που εκδίδονται από συγκεκριμένους εκδότες, υποστηρίζοντας την αναγνώριση των εκδοτικών δραστηριοτήτων.
- **Υλοποίηση:** Το ξένο κλειδί PublisherID στον πίνακα BOOK συνδέεται με το PublisherID στον πίνακα PUBLISHER, υποδεικνύοντας τον εκδότη του βιβλίου.

Ιστοσελίδες - Οργανισμοί (Websites - Organizations):

- **Τύπος Σχέσης:** Πολλά προς ένα.
- **Περιγραφή:** Κάθε ιστοσελίδα μπορεί να σχετίζεται με έναν οργανισμό που την υποστηρίζει ή τη φιλοξενεί. Αυτή η σχέση είναι σημαντική για την καταγραφή της προέλευσης των πληροφοριών που παρουσιάζονται σε ιστοσελίδες και την αναγνώριση των οργανισμών που παρέχουν έγκυρες και αξιόπιστες πηγές πληροφόρησης.
- **Υλοποίηση:** Το ξένο κλειδί OrganizationID στον πίνακα WEBSITE συνδέεται με το OrganizationID στον πίνακα ORGANIZATION, καθορίζοντας τον οργανισμό που σχετίζεται με την ιστοσελίδα.

Σχέσεις Πολλά προς Πολλά μεταξύ Συγγραφέων και Δημοσιεύσεων:

- **Περιγραφή:** Οι ενδιάμεσοι πίνακες (ARTICLE_AUTHOR, BOOK_AUTHOR, THESIS_AUTHOR, WEBSITE_AUTHOR) επιτρέπουν την καταγραφή των πολλών προς πολλές σχέσεων μεταξύ των συγγραφέων και των έργων τους. Αυτές οι σχέσεις εξασφαλίζουν την ευελιξία και την ακρίβεια στην καταγραφή της συμμετοχής των συγγραφέων σε διάφορες δημοσιεύσεις.
- **Υλοποίηση:** Οι ενδιάμεσοι πίνακες χρησιμοποιούν ξένα κλειδιά για να συνδέσουν τους συγγραφείς με τις οντότητες τους (π.χ., άρθρα, βιβλία, εργασίες, ιστοσελίδες), επιτρέποντας τη λεπτομερή καταγραφή των συνεργασιών και των συνεισφορών.

Αναλυτική Περιγραφή Πινάκων:

Παρακάτω παρέχεται ένας συνολικός πίνακας που περιγράφει όλους τους πίνακες της βάσης δεδομένων (οντότητες), περιλαμβάνοντας το όνομα του πίνακα, τα πεδία, τον τύπο δεδομένων κάθε πεδίου, τα κλειδιά (PK, FK), και μια σύντομη περιγραφή για κάθε πεδίο. Ως «PK» Primary Key (Πρωτεύον Κλειδί) ορίζεται ένας μοναδικός αναγνωριστικός αριθμός για την κάθε οντότητα. Ως «FK»: Foreign Key (Ξένο Κλειδί) ορίζεται ένας αναγνωριστικός αριθμός που συνδέεται με άλλον πίνακα, δημιουργώντας σχέσεις μεταξύ των πινάκων.

ΠΙΝΑΚΑΣ	ΠΕΔΙΟ	ΤΥΠΟΣ	ΚΛΕΙΔΙ	ΠΕΡΙΓΡΑΦΗ
ΔΕΔΟΜΕΝΩΝ				
AUTHORS	AuthorID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε συγγραφέα.
	FirstName	varchar		Το όνομα του συγγραφέα.
	LastName	varchar		Το επώνυμο του συγγραφέα.
	ORCID	varchar		Μοναδικός αριθμός ORCID για την αναγνώριση συγγραφέων.
ARTICLES	ArticleID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε άρθρο.
	Title	varchar		Ο τίτλος του άρθρου.
	JournalID	int	FK	Αναγνωριστικός αριθμός του περιοδικού.
	Year	int		Το έτος δημοσίευσης του άρθρου.

<i>BOOKS</i>	Volume	varchar		Ο όγκος του περιοδικού.
	Number	int		Ο αριθμός του τεύχους του περιοδικού.
	Pages	varchar		Το εύρος των σελίδων του άρθρου.
	DOI	varchar		Ο ψηφιακός αναγνωριστής αντικειμένου του άρθρου.
	BookID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε βιβλίο.
	Title	varchar		Ο τίτλος του βιβλίου.
	PublisherID	int	FK	Αναγνωριστικός αριθμός εκδότη.
	Year	int		Το έτος έκδοσης του βιβλίου.
	ISBN	varchar		Ο διεθνής αριθμός βιβλίου.
<i>JOURNALS</i>	Edition	varchar		Η έκδοση του βιβλίου.
	JournalID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε περιοδικό.
	Name	varchar		Το όνομα του περιοδικού.
<i>PUBLISHERS</i>	ISSN	varchar		Ο διεθνής αριθμός αναγνώρισης περιοδικού.
	PublisherID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε εκδότη.
	Name	varchar		Το όνομα του εκδότη.

THESES

Location	varchar		Η τοποθεσία του εκδότη.
ThesisID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε εργασία.
Title	varchar		Ο τίτλος της εργασίας.
Year	int		Το έτος ολοκλήρωσης της εργασίας.
University	varchar		Το πανεπιστήμιο στο οποίο υποβλήθηκε η εργασία.
Type	varchar		Ο τύπος της εργασίας (π.χ., Διπλωματική, Διδακτορική).

WEBSITES

WebsiteID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε ιστοσελίδα.
Title	varchar		Ο τίτλος της ιστοσελίδας.
Year	int		Το έτος δημοσίευσης ή τελευταίας ενημέρωσης της ιστοσελίδας.
URL	varchar		Η διεύθυνση URL της ιστοσελίδας.
Institution	varchar		Το ίδρυμα ή ο οργανισμός που φιλοξενεί την ιστοσελίδα.
OrganizationID	int	FK	Αναγνωριστικός αριθμός οργανισμού.

<i>ORGANIZATIONS</i>	OrganizationID	int	PK	Μοναδικός αναγνωριστικός αριθμός για κάθε οργανισμό.
	Name	varchar		Το όνομα του οργανισμού.
	Type	varchar		Ο τύπος του οργανισμού (π.χ., Πανεπιστήμιο, Ερευνητικό Κέντρο).
	Location	varchar		Η τοποθεσία του οργανισμού.
<i>ARTICLE_AUTHOR</i>	ArticleID	int	FK	Αναγνωριστικός αριθμός άρθρου.
	AuthorID	int	FK	Αναγνωριστικός αριθμός συγγραφέα.
<i>BOOK_AUTHOR</i>	BookID	int	FK	Αναγνωριστικός αριθμός βιβλίου.
	AuthorID	int	FK	Αναγνωριστικός αριθμός συγγραφέα.
<i>THESIS_AUTHOR</i>	ThesisID	int	FK	Αναγνωριστικός αριθμός εργασίας.
	AuthorID	int	FK	Αναγνωριστικός αριθμός συγγραφέα.
<i>WEBSITE_AUTHOR</i>	WebsiteID	int	FK	Αναγνωριστικός αριθμός ιστοσελίδας.
	AuthorID	int	FK	Αναγνωριστικός αριθμός συγγραφέα.

Στον πίνακα παρουσιάζονται και οι βοηθητικοί πίνακες όπου όλα τους τα πεδία είναι δευτερεύοντα κλειδιά.

Σχέσεις:

1. **Articles-Journals:** Μια σχέση "πολλά προς ένα" μέσω του ξένου κλειδιού JournalID.
2. **Books-Publishers:** Μια σχέση "πολλά προς ένα" μέσω του ξένου κλειδιού PublisherID.
3. **Authors-Articles/Books:** Μια σχέση "πολλά προς πολλά" μέσω ενδιάμεσων πινάκων Article_Author και Book_Author.
4. **OnlineSources-Authors:** Μια σχέση "πολλά προς ένα" μέσω του ξένου κλειδιού AuthorID.
5. **Theses-Authors:** Μια σχέση "πολλά προς ένα" μέσω του ξένου κλειδιού AuthorID.
6. **TechReports-Organizations:** Μια σχέση "πολλά προς ένα" μέσω του ξένου κλειδιού OrganizationID.

Ενδιάμεσοι Πίνακες:

1. **Article_Author:** Περιλαμβάνει τα πεδία ArticleID και AuthorID, καθορίζοντας τη σχέση "πολλά προς πολλά" μεταξύ άρθρων και συγγραφέων.
2. **Book_Author:** Περιλαμβάνει τα πεδία BookID και AuthorID, καθορίζοντας τη σχέση "πολλά προς πολλά" μεταξύ βιβλίων και συγγραφέων.

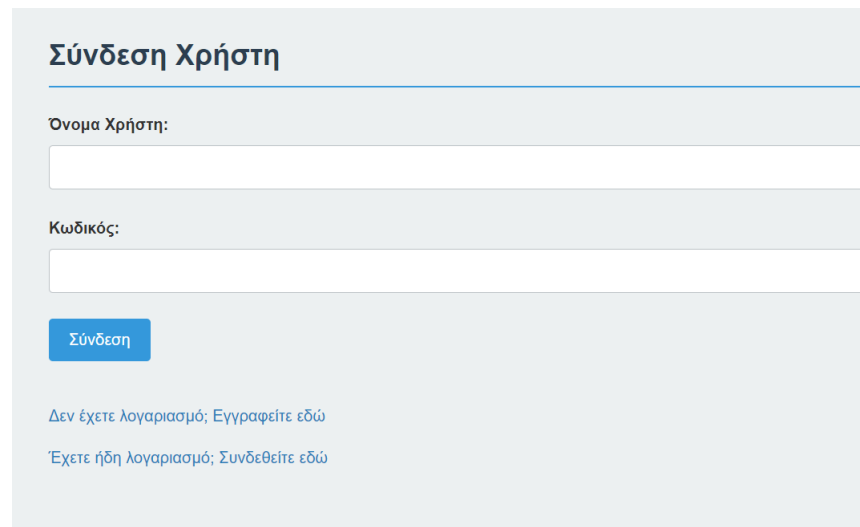
4.3 Σχεδίαση Διεπαφής Χρήστη

Η διεπαφή χρήστη είναι το σημείο επαφής μεταξύ του χρήστη και του συστήματος και πρέπει να σχεδιαστεί με γνώμονα την ευχρηστία και την αποδοτικότητα [8]. Ακολουθούν οι κύριες σελίδες της διεπαφής χρήστη και οι λειτουργίες που παρέχουν.

Κύριες Σελίδες

Σελίδα Σύνδεσης Χρήστη

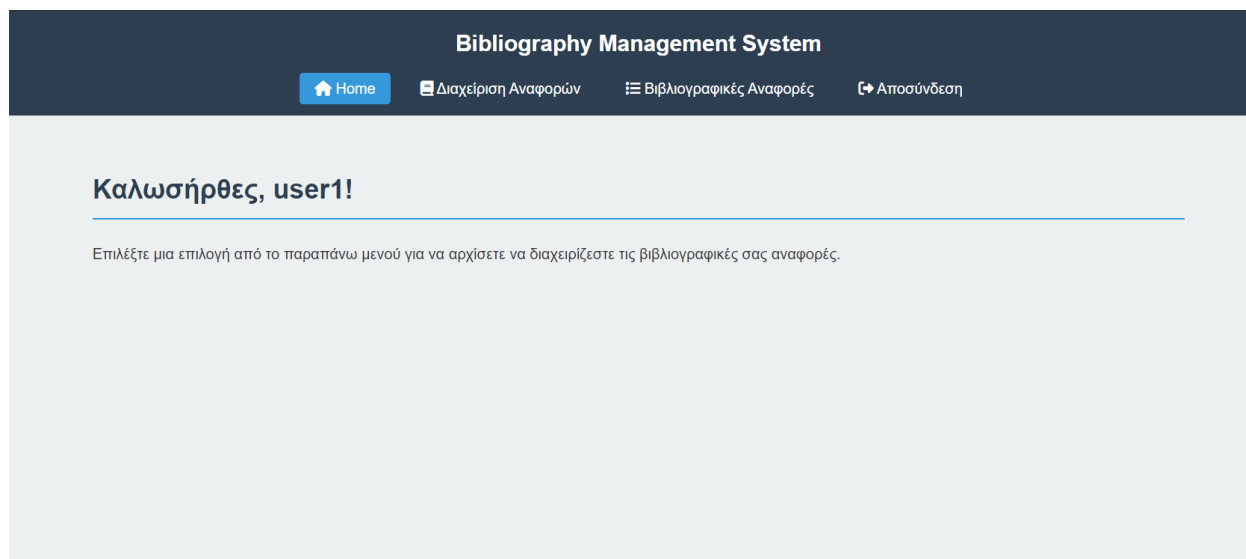
Αυτή η σελίδα αποτελεί το αρχικό σημείο εισόδου του χρήστη στην εφαρμογή. Περιλαμβάνει πεδία για το **όνομα χρήστη** και τον **κωδικό πρόσβασης**. Υπάρχουν δύο σύνδεσμοι: ένας για **εγγραφή** νέου λογαριασμού και ένας για **ανάκτηση κωδικού πρόσβασης**. Οπτικά, η σελίδα είναι απλή και καθαρή, με έμφαση στη λειτουργικότητα.



The screenshot shows a light gray rectangular box representing the login interface. At the top, the title "Σύνδεση Χρήστη" is displayed in a dark blue font, underlined. Below the title, there are two input fields. The first field is labeled "Όνομα Χρήστη:" and the second is labeled "Κωδικός:". Both labels are in a small, dark gray font. Below the second input field, there is a blue button with the text "Σύνδεση" in white. At the bottom of the box, there are two lines of text in a small, dark gray font: "Δεν έχετε λογαριασμό; Εγγραφείτε εδώ" and "Έχετε ήδη λογαριασμό; Συνδεθείτε εδώ".

Κεντρική Σελίδα Χρήστη

Μετά την επιτυχή σύνδεση, ο χρήστης μεταφέρεται στη **σελίδα υποδοχής**. Η σελίδα εμφανίζει μήνυμα καλωσορίσματος με το όνομα του χρήστη (π.χ., "user1") και παρέχει **πλοήγηση** στα βασικά χαρακτηριστικά της εφαρμογής μέσω του επάνω μενού (π.χ., "Manage References", "Reference List" και "Logout").



Σελίδα Διαχείρισης Αναφορών

Αυτή η σελίδα παρουσιάζει μια λίστα με όλες τις αποθηκευμένες **πηγές** που έχει καταχωρήσει ο χρήστης, μαζί με βασικές πληροφορίες όπως **τίτλος, συγγραφείς, έτος, τύπος**. Κάθε αναφορά συνοδεύεται από συνδέσμους για **επεξεργασία** ή **διαγραφή**. Ο χρήστης μπορεί να **προσθέσει νέα πηγή** ή να **εξάγει** τις πηγές του σε μορφή **CSV**. Επίσης, υπάρχει **εργαλείο αναζήτησης** για διευκόλυνση στην πλοήγηση.

Bibliography Management System

Home

Διαχείριση Αναφορών

Βιβλιογραφικές Αναφορές

Αποσύνδεση

Διαχείριση Αναφορών

Ααζήτηση αναφορών...

Ταξινόμηση κατά τίτλο

Τίτλος	Συγγραφείς	Έτος	Τύπος	Ενέργειες	
Cryptography and Network Security: Principles and Practice	William Stallings	2017	Book	Επεξεργασία	Διαγραφή
Data Security and Privacy Protection Issues in Cloud Computing: A Survey	Jing Wang, Xiaoyan Wang	2020	Journal Article	Επεξεργασία	Διαγραφή
Latest Updates on Quantum Computing	Michael Young	2023	Website	Επεξεργασία	Διαγραφή
Security Enhancements in IoT Networks	Sarah Johnson	2021	Conference Paper	Επεξεργασία	Διαγραφή

Προσθήκη νέας αναφοράς

Εξαγωγή σε CSV

Φόρμα Προσθήκης Βιβλιογραφικής Αναφοράς

Αυτή η φόρμα επιτρέπει την **προσθήκη νέας πηγής** στην εφαρμογή. Ο χρήστης καλείται να συμπληρώσει πεδία όπως **τίτλος, συγγραφείς, έτος, εκδότης, DOI**, ανάλογα με τον τύπο της πηγής που επιλέγει (π.χ. άρθρο, βιβλίο, ιστοσελίδα). Υπάρχει επίσης επιλογή προσθήκης **περισσότερων συγγραφέων**, ενώ η φόρμα έχει σχεδιαστεί με τρόπο που διευκολύνει την οργάνωση της πληροφορίας.

Bibliography Management System
Home
Διαχείριση Αναφορών
Βιβλιογραφικές Αναφορές
Αποσύνδεση

Προσθήκη Νέας Αναφοράς

Τύπος Πηγής:

Journal Article

Τίτλος:

Υποχρεωτικό

Έτος:

Υποχρεωτικό

Συγγραφείς

Όνομα

Επώνυμο

Υποχρεωτικό

Υποχρεωτικό

Διαγραφή

Προσθήκη Συγγραφέα

Journal Name:

Υποχρεωτικό

Τεύχος:

Προαιρετικό

Έκδοση:

Προαιρετικό

Σελίδες:

Προαιρετικό

DOI:

Προαιρετικό

Προσθήκη Αναφοράς

Λίστα Πηγών και Δημιουργία Παραθεμάτων

Σε αυτήν τη σελίδα, ο χρήστης μπορεί να επιλέξει πηγές για δημιουργία **παραθεμάτων** (citations). Προσφέρει τη δυνατότητα επιλογής στυλ παραθέματος (π.χ., **APA, Harvard**), και μετά την επιλογή, το σύστημα δημιουργεί το σωστά μορφοποιημένο παράθεμα για κάθε επιλεγμένη πηγή, εμφανίζοντάς το σε λίστα.

Bibliography Management System

Home

Διαχείριση Αναφορών

Βιβλιογραφικές Αναφορές

Αποσύνδεση

Βιβλιογραφικές Αναφορές

Αποθηκευμένες Αναφορές

☒ Επιλογή Όλων

☒ Cryptography and Network Security: Principles and Practice (2017)
William Stallings

☒ Data Security and Privacy Protection Issues in Cloud Computing: A Survey (2020)
Jing Wang, Xiaoyan Wang

☒ Latest Updates on Quantum Computing (2023)
Michael Young

☒ Security Enhancements in IoT Networks (2021)
Sarah Johnson

APA Style

Επιλεγμένες Αναφορές

Stallings, W. (2017). Cryptography and network security: principles and practice. Pearson.

Wang, J., & Wang, X. (2020). Data security and privacy protection issues in cloud computing: a survey. *IEEE Access*, 8(1), 18631-18649.
<https://doi.org/10.1109/ICCSEE.2012.193>.

Young, M. (2023). Latest updates on quantum computing.

Johnson, S. (2021). Security enhancements in iot networks.

Δημιουργία Αναφορών

43

5 Υλοποίηση του Συστήματος

5.1 Περιβάλλον Ανάπτυξης και Τεχνολογίες

Η ανάπτυξη του συστήματος διαχείρισης βιβλιογραφικών αναφορών πραγματοποιήθηκε με τη χρήση τεχνολογιών που επιλέχθηκαν για την αποδοτικότητά τους, την ευκολία ενσωμάτωσης και την επεκτασιμότητα του συστήματος. Στην παρακάτω ανάλυση αναφέρονται οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν, καθώς και ο ρόλος των αρχείων του κώδικα.

Γλώσσα Προγραμματισμού

Η κύρια γλώσσα προγραμματισμού που χρησιμοποιήθηκε για την ανάπτυξη του συστήματος είναι η PHP μαζί με βασικά στοιχεία HTML. Η PHP προσφέρει δυναμικό χειρισμό δεδομένων και είναι κατάλληλη για την ανάπτυξη διαδραστικών εφαρμογών ιστού, ειδικά σε περιπτώσεις όπου απαιτείται διαχείριση δεδομένων από μια βάση δεδομένων, όπως στην παρούσα εφαρμογή [18].

Κάθε ένα από τα 20 αρχεία που έχουν αναπτυχθεί για τη λειτουργικότητα του συστήματος περιγράφονται παρακάτω:

1. **edit_reference.php**: Αυτό το αρχείο επιτρέπει την επεξεργασία βιβλιογραφικών αναφορών που έχουν ήδη εισαχθεί. Ο χρήστης μπορεί να τροποποιήσει τα δεδομένα μιας αναφοράς και να τα αποθηκεύσει στη βάση δεδομένων.
2. **error_handling.php**: Διαχειρίζεται τα σφάλματα του συστήματος καταγράφοντάς τα σε αρχεία καταγραφής (`error_log.txt`). Αυτό το αρχείο εξασφαλίζει ότι τα σφάλματα διαχειρίζονται με ασφάλεια χωρίς να αποκαλύπτουν ευαίσθητες πληροφορίες στους χρήστες.
3. **footer.php**: Το `footer.php` αποτελεί μέρος της διεπαφής χρήστη και περιλαμβάνει το τυποποιημένο κάτω μέρος της κάθε σελίδας, όπως δικαιώματα πνευματικής ιδιοκτησίας ή άλλες στατικές πληροφορίες.
4. **index.php**: Η κύρια σελίδα της εφαρμογής. Παρουσιάζει ένα σύνολο πληροφοριών στον χρήστη και τον καθοδηγεί να δει ή να διαχειριστεί τις βιβλιογραφικές αναφορές του ανάλογα με το αν είναι συνδεδεμένος.

5. **layout.php**: Οργανώνει τη γενική εμφάνιση και τη δομή των σελίδων του συστήματος. Περιλαμβάνει στοιχεία όπως το header και το footer, εξασφαλίζοντας συνέπεια στην εμφάνιση των σελίδων.
6. **login_register.php**: Περιλαμβάνει τη φόρμα σύνδεσης και εγγραφής χρηστών. Διαχειρίζεται την αυθεντικοποίηση χρηστών και επιτρέπει την πρόσβαση σε προστατευμένες περιοχές του συστήματος.
7. **add_edit_entity.php**: Χρησιμοποιείται για την προσθήκη ή την επεξεργασία δεδομένων οντοτήτων στη βάση δεδομένων. Αφορά όλες τις οντότητες του συστήματος, όπως συγγραφείς, άρθρα, βιβλία κ.λπ.
8. **add_reference.php**: Παρέχει τη δυνατότητα προσθήκης νέων βιβλιογραφικών αναφορών. Μέσω μιας φόρμας, ο χρήστης μπορεί να εισάγει στοιχεία όπως τίτλος, συγγραφείς, έτος δημοσίευσης, και άλλα χαρακτηριστικά μιας βιβλιογραφικής αναφοράς.
9. **bibliography_db.sql**: Το αρχείο SQL που περιλαμβάνει την αρχική δομή της βάσης δεδομένων. Περιέχει τον ορισμό των πινάκων, των σχέσεων και των πρωτεύοντων και ξένων κλειδιών που χρησιμοποιούνται στη βάση δεδομένων.
10. **db_connect.php**: Αυτό το αρχείο διαχειρίζεται τη σύνδεση με τη MySQL βάση δεδομένων. Περιλαμβάνει τον κώδικα για τη σύνδεση με τη βάση δεδομένων, τον χειρισμό σφαλμάτων σύνδεσης και τη διατήρηση της ασφάλειας των δεδομένων.
11. **manage_references.php**: Επιτρέπει στους χρήστες να διαχειρίζονται όλες τις βιβλιογραφικές αναφορές που έχουν εισαγάγει. Ο χρήστης μπορεί να αναζητήσει, να τροποποιήσει ή να διαγράψει αναφορές.
12. **manage_users.php**: Επιτρέπει τη διαχείριση των χρηστών από τον διαχειριστή του συστήματος. Ο διαχειριστής μπορεί να προσθέσει, να τροποποιήσει ή να διαγράψει χρήστες, καθώς και να καθορίσει τα δικαιώματα πρόσβασής τους.
13. **process_entity.php**: Αυτό το αρχείο χρησιμοποιείται για την επεξεργασία δεδομένων από τη φόρμα εισαγωγής και επεξεργασίας οντοτήτων. Τα δεδομένα αποστέλλονται στη βάση δεδομένων για αποθήκευση ή ενημέρωση.

14. **process_login.php**: Υπεύθυνο για την επεξεργασία των στοιχείων σύνδεσης χρηστών και τη διαχείριση της συνεδρίας. Ελέγχει την εγκυρότητα των στοιχείων σύνδεσης και κατευθύνει τους χρήστες στη σωστή σελίδα ανάλογα με τα δικαιώματά τους.
15. **reference_list.php**: Παρέχει έναν πλήρη κατάλογο όλων των βιβλιογραφικών αναφορών που έχουν εισαχθεί στο σύστημα. Προσφέρει τη δυνατότητα στους χρήστες να δουν όλες τις αναφορές και να επιλέξουν να τις επεξεργαστούν ή να τις διαγράψουν.
16. **register.php**: Χρησιμοποιείται για την εγγραφή νέων χρηστών στο σύστημα. Επιτρέπει στους χρήστες να εισάγουν τα στοιχεία τους για να δημιουργήσουν έναν νέο λογαριασμό και αποθηκεύει αυτά τα στοιχεία στη βάση δεδομένων.
17. **search_results.php**: Αυτό το αρχείο διαχειρίζεται τα αποτελέσματα αναζήτησης βιβλιογραφικών αναφορών. Επιτρέπει στους χρήστες να αναζητήσουν συγκεκριμένες αναφορές με βάση διάφορα κριτήρια, όπως ο τίτλος ή ο συγγραφέας.
18. **validate_input.php**: Περιλαμβάνει λειτουργίες για τον έλεγχο της εγκυρότητας των δεδομένων που εισάγονται από τους χρήστες. Εξασφαλίζει ότι τα δεδομένα που αποθηκεύονται είναι έγκυρα και ασφαλή, αποτρέποντας την είσοδο κακόβουλων δεδομένων.
19. **logout.php**: Αυτό το αρχείο διαχειρίζεται την αποσύνδεση των χρηστών. Καταστρέφει τη συνεδρία και επανακατευθύνει τους χρήστες στην κύρια σελίδα μετά την αποσύνδεσή τους.
20. **index.php (τελευταίο αρχείο)**: Το αρχικό σημείο πρόσβασης στο σύστημα, παρέχοντας στον χρήστη την πρώτη αλληλεπίδραση με την εφαρμογή.

Βάση Δεδομένων

5.1.1 Βάση Δεδομένων

Η βάση δεδομένων που χρησιμοποιείται είναι η MySQL, και το αρχείο `bibliography_db.sql` περιέχει τους πίνακες και τις σχέσεις που αποθηκεύουν τις βιβλιογραφικές πληροφορίες. Επιπλέον, οι πίνακες συνδέονται μεταξύ τους μέσω ξένων κλειδιών (foreign keys), τα οποία διασφαλίζουν την ακεραιότητα των δεδομένων [17]. Παρακάτω περιγράφονται οι πίνακες, τα πεδία και οι σχέσεις (ξένα κλειδιά).

ARTICLES: Αποθηκεύει πληροφορίες σχετικά με επιστημονικά άρθρα. Περιλαμβάνει τα πεδία:

- `ArticleID` (int, Primary Key): Μοναδικός αναγνωριστικός αριθμός για κάθε άρθρο.
- `Title` (varchar): Ο τίτλος του άρθρου.
- `Journal` (varchar): Το περιοδικό όπου δημοσιεύθηκε το άρθρο.
- `Year` (int): Το έτος δημοσίευσης του άρθρου.
- `DOI` (varchar): Ο ψηφιακός αναγνωριστικός αριθμός (Digital Object Identifier) του άρθρου.

ARTICLE_AUTHOR: Αυτός ο ενδιάμεσος πίνακας συνδέει άρθρα με συγγραφείς σε σχέση πολλά προς πολλά μέσω των ξένων κλειδιών:

- `ArticleID` (int, Foreign Key): Συνδέεται με τον πίνακα `ARTICLES` (`ArticleID`).
- `AuthorID` (int, Foreign Key): Συνδέεται με τον πίνακα `AUTHORS` (`AuthorID`).

AUTHORS: Αποθηκεύει πληροφορίες σχετικά με συγγραφείς.

- `AuthorID` (int, Primary Key): Μοναδικός αναγνωριστικός αριθμός για κάθε συγγραφέα.
- `FirstName` (varchar): Το όνομα του συγγραφέα.
- `LastName` (varchar): Το επώνυμο του συγγραφέα.
- `ORCID` (varchar): Ο αριθμός ORCID του συγγραφέα για παγκόσμια αναγνώριση.

BOOKS: Περιέχει πληροφορίες για βιβλία.

- `BookID` (int, Primary Key): Μοναδικός αναγνωριστικός αριθμός για κάθε βιβλίο.
- `Title` (varchar): Ο τίτλος του βιβλίου.
- `Publisher` (varchar): Το όνομα του εκδότη.
- `Year` (int): Το έτος έκδοσης του βιβλίου.
- `ISBN` (varchar): Ο Διεθνής Αριθμός Βιβλίου (International Standard Book Number).

BOOK_AUTHOR: Ενδιάμεσος πίνακας που συνδέει βιβλία με συγγραφείς σε σχέση πολλά προς πολλά μέσω των ξένων κλειδιών:

- `BookID` (int, Foreign Key): Συνδέεται με τον πίνακα `BOOKS` (`BookID`).
- `AuthorID` (int, Foreign Key): Συνδέεται με τον πίνακα `AUTHORS` (`AuthorID`).

REFERENCES: Περιέχει γενικές βιβλιογραφικές αναφορές.

- `id` (int, Primary Key): Μοναδικός αναγνωριστικός αριθμός για την αναφορά.
- `user_id` (int, Foreign Key): Συνδέεται με τον πίνακα `USERS` (`UserID`).
- `type` (varchar): Ο τύπος της αναφοράς (π.χ. άρθρο, βιβλίο, κ.λπ.).
- `title` (varchar): Ο τίτλος της αναφοράς.
- `year` (int): Το έτος της αναφοράς.
- `journal` (varchar): Το περιοδικό όπου δημοσιεύθηκε η αναφορά (αν είναι άρθρο).
- `volume` (varchar): Ο τόμος του περιοδικού.
- `issue` (varchar): Το τεύχος του περιοδικού.

REFERENCE_AUTHOR: Συνδέει αναφορές με συγγραφείς μέσω των ξένων κλειδιών:

- `ReferenceID` (int, Foreign Key): Συνδέεται με τον πίνακα `REFERENCES` (`id`).
- `AuthorID` (int, Foreign Key): Συνδέεται με τον πίνακα `AUTHORS` (`AuthorID`).

THESIS_AUTHOR: Συνδέει διπλωματικές/διδακτορικές εργασίες με συγγραφείς μέσω των ξένων κλειδιών:

- `ThesisID` (int, Foreign Key): Συνδέεται με τον πίνακα `THESES` (`ThesisID`).
- `AuthorID` (int, Foreign Key): Συνδέεται με τον πίνακα `AUTHORS` (`AuthorID`).

WEBSITE_AUTHOR: Συνδέει ιστοσελίδες με συγγραφείς μέσω των ξένων κλειδιών:

- `WebsiteID` (int, Foreign Key): Συνδέεται με τον πίνακα `WEBSITES` (`WebsiteID`).
- `AuthorID` (int, Foreign Key): Συνδέεται με τον πίνακα `AUTHORS` (`AuthorID`).

5.1.2 Περιβάλλον Ανάπτυξης

Το σύστημα αναπτύχθηκε σε ένα τοπικό περιβάλλον ανάπτυξης, το οποίο περιλαμβάνει την υποστήριξη για PHP και MySQL, δύο από τις πιο ευρέως χρησιμοποιούμενες τεχνολογίες για τη δημιουργία δυναμικών εφαρμογών ιστού και διαχείρισης βάσεων δεδομένων. Η επιλογή του συγκεκριμένου περιβάλλοντος έγινε λόγω της ευελιξίας και της ευρείας αποδοχής τους στη

δημιουργία εφαρμογών που απαιτούν συνδυασμό ισχυρής διαχείρισης δεδομένων και διαδραστικών δυνατοτήτων χρήστη [8].

PHP και MySQL

Η PHP είναι μια γλώσσα προγραμματισμού ειδικά σχεδιασμένη για ανάπτυξη ιστοσελίδων και διαδραστικών εφαρμογών ιστού. Χρησιμοποιήθηκε για την υλοποίηση όλων των βασικών λειτουργιών του συστήματος, όπως η διαχείριση βιβλιογραφικών αναφορών, η διαχείριση χρηστών, καθώς και η επικοινωνία με τη βάση δεδομένων. Η PHP υποστηρίζει τη δυναμική δημιουργία περιεχομένου και επιτρέπει την ευκολία σύνδεσης με τη **MySQL** βάση δεδομένων για την αποθήκευση και ανάκτηση δεδομένων.

Η **MySQL** επιλέχθηκε ως σύστημα διαχείρισης βάσεων δεδομένων (DBMS) λόγω της αξιοπιστίας και της επεκτασιμότητάς της. Όλες οι βιβλιογραφικές αναφορές, οι συγγραφείς, τα άρθρα, και οι σχέσεις μεταξύ αυτών αποθηκεύονται σε μια σχεσιακή βάση δεδομένων, την οποία διαχειρίζεται η MySQL. Χρησιμοποιήθηκε επίσης για την υλοποίηση των σχέσεων πολλά προς πολλά και την εξασφάλιση της ακεραιότητας των δεδομένων μέσω πρωτευόντων και ξένων κλειδιών.

Apache HTTP Server

Για την εξυπηρέτηση των αιτήσεων της εφαρμογής και τη φιλοξενία των σελίδων της, χρησιμοποιήθηκε ο **Apache HTTP Server**, ένας από τους πιο διαδεδομένους web servers ανοιχτού κώδικα. Ο Apache είναι υπεύθυνος για τη διαχείριση των αιτήσεων HTTP από τους χρήστες και την προώθησή τους προς τα αρχεία PHP, ώστε να δημιουργείται το δυναμικό περιεχόμενο της εφαρμογής. Χρησιμοποιήθηκε για να διαχειρίζεται τα αιτήματα σύνδεσης, τις αναζητήσεις βιβλιογραφικών αναφορών, και τις ενέργειες που εκτελούν οι χρήστες (όπως η προσθήκη και η επεξεργασία αναφορών).

Εργαλείο Επεξεργασίας Κώδικα (Notepad++)

Η ανάπτυξη του κώδικα έγινε με τη χρήση του **Notepad++**, ενός ελαφρού και ευέλικτου εργαλείου επεξεργασίας κώδικα. Το Notepad++ υποστηρίζει πλήρως τη σύνταξη γλωσσών όπως PHP, HTML, και SQL, επιτρέποντας στον προγραμματιστή να γράψει, να οργανώσει, και να

διαχειριστεί αποδοτικά όλα τα αρχεία του συστήματος. Παρόλο που είναι ένα απλό εργαλείο, παρέχει σημαντικές δυνατότητες όπως η επισήμανση σύνταξης (syntax highlighting), η αναδίπλωση κώδικα (code folding), και η εύκολη πλοήγηση μεταξύ των αρχείων του έργου, καθιστώντας το ιδανικό για τη διαχείριση και την ανάπτυξη μιας τέτοιας εφαρμογής.

Το τοπικό περιβάλλον ανάπτυξης επιτρέπει τη δοκιμή και την παρακολούθηση της λειτουργίας της εφαρμογής σε πραγματικό χρόνο, ενώ παρέχει ασφάλεια και ευκολία στη διαχείριση της βάσης δεδομένων και των αρχείων του συστήματος. Στη συνέχεια, το σύστημα μπορεί να αναπτυχθεί σε έναν απομακρυσμένο διακομιστή όταν η ανάπτυξη ολοκληρωθεί και το σύστημα είναι έτοιμο για χρήση από τους τελικούς χρήστες.

5.2 Βήματα Υλοποίησης

Η υλοποίηση του συστήματος διαχείρισης βιβλιογραφικών αναφορών έγινε μέσα από μία σειρά συγκεκριμένων βημάτων που κάλυψαν τόσο την υποδομή του back-end όσο και την εμπειρία χρήστη στο front-end. Παρακάτω περιγράφονται αναλυτικά τα βήματα που ακολουθήθηκαν, συμπεριλαμβάνοντας τα 20 αρχεία κώδικα που χρησιμοποιήθηκαν.

5.2.1 Ανάλυση των Απαιτήσεων

Το πρώτο βήμα ήταν η ανάλυση των απαιτήσεων του συστήματος. Καθορίστηκαν οι βασικές λειτουργίες του συστήματος, όπως:

- Διαχείριση βιβλιογραφικών αναφορών (εισαγωγή, επεξεργασία, διαγραφή, αναζήτηση).
- Διαχείριση συγγραφέων και δημοσιεύσεων.
- Διαχείριση χρηστών με ρόλους και δικαιώματα πρόσβασης.

5.2.2 Σχεδιασμός της Βάσης Δεδομένων

Η σχεδίαση της βάσης δεδομένων έγινε χρησιμοποιώντας το MySQL και περιλαμβάνει πίνακες για τις οντότητες:

- **AUTHORS, ARTICLES, BOOKS, PUBLISHERS, JOURNALS, THESES, WEBSITES.** Οι σχέσεις μεταξύ αυτών των οντοτήτων (π.χ. `ARTICLE_AUTHOR`, `BOOK_AUTHOR`) υλοποιήθηκαν μέσω ξένων κλειδιών για την εξασφάλιση της ακεραιότητας των δεδομένων.

5.2.3 Ανάπτυξη του Back-End Συστήματος

Το back-end αναπτύχθηκε με τη χρήση PHP και αποτελείται από τα παρακάτω αρχεία:

- **db_connect.php**: Διαχειρίζεται τη σύνδεση με τη βάση δεδομένων MySQL. Ασφαλίζει τη σωστή επικοινωνία της εφαρμογής με τη βάση δεδομένων, διαχειριζόμενη τη σύνδεση και τα queries.
- **add_reference.php** και **edit_reference.php**: Αρχεία που διαχειρίζονται την προσθήκη και επεξεργασία βιβλιογραφικών αναφορών μέσω φορμών. Ο χρήστης μπορεί να εισάγει νέα δεδομένα ή να επεξεργαστεί υπάρχουσες αναφορές.
- **add_edit_entity.php**: Αρχείο που επιτρέπει την προσθήκη και επεξεργασία δεδομένων για όλες τις οντότητες της βάσης δεδομένων (π.χ. συγγραφείς, άρθρα, βιβλία).
- **process_entity.php**: Επεξεργάζεται δεδομένα που λαμβάνονται από τις φόρμες του χρήστη και τα αποθηκεύει στη βάση δεδομένων, είτε προσθέτοντας νέα είτε ενημερώνοντας υπάρχοντα.
- **validate_input.php**: Εξασφαλίζει την ασφάλεια και την εγκυρότητα των δεδομένων που εισάγονται, αποτρέποντας κακόβουλα δεδομένα ή σφάλματα κατά την αποθήκευση.
- **process_login.php** και **logout.php**: Αρχεία που διαχειρίζονται τη σύνδεση και αποσύνδεση χρηστών, επαληθεύοντας τα διαπιστευτήρια και διαχειρίζοντας τις συνεδρίες (sessions).
- **manage_references.php**: Επιτρέπει τη διαχείριση όλων των βιβλιογραφικών αναφορών από τον χρήστη. Ο χρήστης μπορεί να δει, να επεξεργαστεί ή να διαγράψει αναφορές.
- **manage_users.php**: Αρχείο για τη διαχείριση των χρηστών. Ο διαχειριστής μπορεί να προσθέσει, επεξεργαστεί και διαγράψει χρήστες, καθώς και να τους εκχωρήσει δικαιώματα πρόσβασης.
- **error_handling.php**: Διαχειρίζεται την καταγραφή και αναφορά σφαλμάτων σε αρχεία καταγραφής, διασφαλίζοντας ότι τα σφάλματα δεν επηρεάζουν τη λειτουργία της εφαρμογής.

5.2.4 Ανάπτυξη της Διεπαφής Χρήστη (Front-End)

Η διεπαφή χρήστη αναπτύχθηκε με χρήση HTML και CSS, προσφέροντας ένα απλό και λειτουργικό περιβάλλον:

- **index.php**: Η κεντρική σελίδα της εφαρμογής που εμφανίζει μια επισκόπηση των βασικών λειτουργιών και επιτρέπει στον χρήστη να πλοηγηθεί στις βιβλιογραφικές αναφορές.
- **footer.php** και **layout.php**: Αρχεία που παρέχουν το πρότυπο της εμφάνισης (layout) και οργανώνουν τη δομή όλων των σελίδων, ώστε να έχουν μια συνεπή εμφάνιση.
- **search_results.php**: Αρχείο που διαχειρίζεται τα αποτελέσματα αναζήτησης, προσφέροντας τη δυνατότητα στον χρήστη να αναζητήσει βιβλιογραφικές αναφορές με βάση διάφορα κριτήρια.
- **login_register.php** και **register.php**: Περιλαμβάνουν τις φόρμες σύνδεσης και εγγραφής χρηστών, επιτρέποντας νέους χρήστες να εγγραφούν και υπάρχοντες να συνδεθούν.

5.2.5 Έλεγχος και Επικύρωση (Testing)

Ο έλεγχος του συστήματος περιλάμβανε:

Έλεγχος Βάσης Δεδομένων: Δοκιμάστηκε η εισαγωγή, ενημέρωση και διαγραφή δεδομένων, καθώς και οι σχέσεις ξένων κλειδιών για την ακεραιότητα των δεδομένων.

Έλεγχος Λειτουργικότητας: Δοκιμάστηκαν όλες οι λειτουργίες του συστήματος, όπως η διαχείριση αναφορών και χρηστών, για να διασφαλιστεί ότι λειτουργούν χωρίς σφάλματα.

Ασφάλεια Δεδομένων: Ελέγχθηκε η ασφάλεια των δεδομένων, ειδικά σε ό,τι αφορά την αποφυγή SQL injection και άλλων πιθανών απειλών [12].

5.2.6 Τελική Εγκατάσταση

Μετά τον έλεγχο, το σύστημα εγκαταστάθηκε σε έναν web server που υποστηρίζει PHP και MySQL, ώστε να είναι προσβάσιμο από τους χρήστες. Ο Apache HTTP Server χρησιμοποιήθηκε για τη διαχείριση των αιτήσεων και τη φιλοξενία της εφαρμογής. Πραγματοποιήθηκαν οι τελικές ρυθμίσεις ασφαλείας και διαχείρισης της βάσης δεδομένων.

6 Αξιολόγηση και Αποτελέσματα

Η αξιολόγηση του συστήματος διαχείρισης βιβλιογραφικών αναφορών έγινε βάσει συγκεκριμένων μεθόδων και κριτηρίων, τα οποία είχαν ως στόχο να επιβεβαιώσουν τη σωστή λειτουργία και την αποδοτικότητα της εφαρμογής [21]. Στη συνέχεια παρουσιάζονται οι μέθοδοι που χρησιμοποιήθηκαν για την αξιολόγηση του συστήματος και τα αποτελέσματα που προέκυψαν.

6.1 Μέθοδοι Αξιολόγησης

Για την αξιολόγηση του συστήματος, χρησιμοποιήθηκαν οι παρακάτω μέθοδοι:

Λειτουργικός Έλεγχος (Functional Testing): Δοκιμάστηκαν όλες οι βασικές λειτουργίες του συστήματος για να διασφαλιστεί ότι ανταποκρίνονται σωστά στις απαιτήσεις που είχαν τεθεί. Περιλάμβανε την εισαγωγή, επεξεργασία, και διαγραφή βιβλιογραφικών αναφορών, τη διαχείριση συγγραφέων και χρηστών, καθώς και τη λειτουργία αναζήτησης και φιλτραρίσματος αναφορών.

Δοκιμή Σωστής Καταχώρησης Δεδομένων: Ελέγχθηκε αν όλα τα δεδομένα εισάγονται σωστά στη βάση δεδομένων, εστιάζοντας στις σχέσεις πολλά προς πολλά (π.χ. συγγραφείς-άρθρα, συγγραφείς-βιβλία) μέσω των ενδιάμεσων πινάκων (π.χ. ARTICLE_AUTHOR, BOOK_AUTHOR).

Έλεγχος Σφαλμάτων (Error Handling Testing): Ελέγχθηκε η σωστή διαχείριση σφαλμάτων μέσω του αρχείου error_handling.php, διασφαλίζοντας ότι όλα τα σφάλματα καταγράφονται χωρίς να εκτίθενται ευαίσθητες πληροφορίες στον τελικό χρήστη.

Έλεγχος Διεπαφής Χρήστη (UI Testing): Ελέγχθηκε η ευχρηστία της διεπαφής, η πλοήγηση στις σελίδες, και η λειτουργία των φορμών. Δοκιμάστηκαν οι φόρμες σύνδεσης και εγγραφής, καθώς και η πλοήγηση στις σελίδες διαχείρισης βιβλιογραφικών αναφορών και χρηστών.

Ασφάλεια Δεδομένων (Security Testing): Εξετάστηκε η προστασία από επιθέσεις όπως SQL injection και cross-site scripting (XSS), μέσω των ελέγχων που παρέχει το αρχείο validate_input.php. Δοκιμάστηκε η εγκυρότητα των εισαγόμενων δεδομένων για να αποτραπούν πιθανές απειλές ασφαλείας.

Απόδοση (Performance Testing): Ελέγχθηκε η ταχύτητα ανταπόκρισης του συστήματος σε βασικές λειτουργίες, όπως η αναζήτηση βιβλιογραφικών αναφορών και η διαχείριση μεγάλου αριθμού χρηστών και δεδομένων.

6.2 Αποτελέσματα Αξιολόγησης

Από την αξιολόγηση του συστήματος προέκυψαν τα παρακάτω αποτελέσματα:

Λειτουργικότητα: Το σύστημα ανταποκρίνεται σωστά στις απαιτήσεις. Όλες οι βασικές λειτουργίες, όπως η εισαγωγή, επεξεργασία και αναζήτηση βιβλιογραφικών αναφορών, λειτουργούν χωρίς προβλήματα. Οι σχέσεις μεταξύ των οντοτήτων διαχειρίζονται σωστά μέσω των ενδιάμεσων πινάκων.

Χρήση και Πλοήγηση: Η διεπαφή είναι εύχρηστη και προσβάσιμη. Ο χρήστης μπορεί να πλοηγηθεί εύκολα στο σύστημα και να ολοκληρώσει τις εργασίες του χωρίς σφάλματα ή προβλήματα στη χρήση των φορμών.

Ασφάλεια: Το σύστημα είναι ασφαλές απέναντι σε κοινές επιθέσεις (SQL injection, XSS). Η χρήση του `validate_input.php` διασφαλίζει ότι τα δεδομένα που εισάγονται στο σύστημα είναι ασφαλή.

Απόδοση: Η απόδοση του συστήματος είναι ικανοποιητική ακόμη και με μεγάλα σύνολα δεδομένων. Η αναζήτηση και φιλτράρισμα βιβλιογραφικών αναφορών εκτελούνται γρήγορα, ακόμα και όταν υπάρχουν πολλές αναφορές στη βάση δεδομένων.

7 Συμπεράσματα και Μελλοντικές Επεκτάσεις

Το παρόν κεφάλαιο συνοψίζει τα κύρια ευρήματα της υλοποίησης του συστήματος διαχείρισης βιβλιογραφικών αναφορών, παρουσιάζοντας τις επιτυχίες και τις προκλήσεις που αναδείχθηκαν κατά τη διάρκεια της ανάπτυξης. Παράλληλα, προτείνονται πιθανές μελλοντικές επεκτάσεις που θα μπορούσαν να ενισχύσουν τη λειτουργικότητα και την αποδοτικότητα του συστήματος.

7.1 Συμπεράσματα

Η ολοκλήρωση του συστήματος διαχείρισης βιβλιογραφικών αναφορών επιτεύχθηκε με επιτυχία, καλύπτοντας όλες τις βασικές λειτουργικές απαιτήσεις που είχαν καθοριστεί κατά τη φάση της ανάλυσης. Η υλοποίηση των κύριων λειτουργιών, όπως η εισαγωγή, η αποθήκευση, η αναζήτηση και η διαχείριση βιβλιογραφικών δεδομένων, αποδείχθηκε επιτυχής, ενώ ταυτόχρονα εξασφαλίστηκε η ομαλή λειτουργία του συστήματος σε πραγματικό χρόνο. Σημαντικό στοιχείο ήταν η χρήση του συστήματος διαχείρισης βάσεων δεδομένων MySQL, το οποίο, σε συνδυασμό με την κατάλληλη σχεδίαση της βάσης δεδομένων, διασφάλισε την ακεραιότητα των δεδομένων. Η ορθή υλοποίηση των σχέσεων μεταξύ των οντοτήτων μέσω των ξένων κλειδιών απέτρεψε την εμφάνιση ασυνεπειών και κατέστησε το σύστημα αξιόπιστο και ανθεκτικό σε σφάλματα.

Αναφορικά με τη λειτουργικότητα, το σύστημα ανταποκρίθηκε στις προσδοκίες, καθώς όλες οι λειτουργίες, από τη διαχείριση των βιβλιογραφικών αναφορών μέχρι τη διαχείριση των χρηστών, εκτελούνται απρόσκοπτα. Δεν παρατηρήθηκαν δυσλειτουργίες ή σφάλματα κατά την εκτέλεση των διεργασιών, γεγονός που καταδεικνύει την επιτυχία του σχεδιασμού και της υλοποίησης. Επιπλέον, η ασφάλεια και η απόδοση του συστήματος διαπιστώθηκαν να είναι επαρκείς, καθώς η εφαρμογή προσφέρει γρήγορη απόκριση και παράλληλα προστασία απέναντι σε πιθανές επιθέσεις και παραβιάσεις. Η εφαρμογή προηγμένων τεχνικών ασφαλείας, όπως η κρυπτογράφηση δεδομένων και η διαχείριση δικαιωμάτων πρόσβασης, ενίσχυσε περαιτέρω την αξιοπιστία και την ασφάλεια του συστήματος.

Το σύστημα αυτό μπορεί να θεωρηθεί ως ένα αξιόπιστο εργαλείο για τη διαχείριση βιβλιογραφικών δεδομένων, ιδιαίτερα σε περιβάλλοντα όπου απαιτούνται ακρίβεια και ασφάλεια. Η δυνατότητα αποθήκευσης, αναζήτησης και επεξεργασίας βιβλιογραφικών

αναφορών, σε συνδυασμό με την ευελιξία και την επεκτασιμότητα του συστήματος, το καθιστούν κατάλληλο για χρήση σε ακαδημαϊκά και ερευνητικά ιδρύματα. Παρόλο που το σύστημα ανταποκρίθηκε στις βασικές απαιτήσεις και επιδιώξεις του έργου, υπάρχουν περιθώρια για περαιτέρω βελτιώσεις και επεκτάσεις.

7.2 Μελλοντικές Επεκτάσεις

Μελλοντικές επεκτάσεις θα μπορούσαν να περιλαμβάνουν την ενσωμάτωση νέων λειτουργιών, όπως η δυνατότητα αυτόματης εξαγωγής βιβλιογραφικών δεδομένων από διαδικτυακές πηγές και βάσεις δεδομένων, καθώς και η ενσωμάτωση συστημάτων ανάλυσης βιβλιογραφικών δεδομένων για την αναγνώριση τάσεων και τη διευκόλυνση της ερευνητικής διαδικασίας. Επιπλέον, η βελτίωση της διεπαφής χρήστη και η ανάπτυξη εφαρμογών για κινητές συσκευές θα μπορούσαν να ενισχύσουν την προσβασιμότητα και τη χρηστικότητα του συστήματος, καθιστώντας το πιο ευέλικτο και εύχρηστο για τους χρήστες.

Συνοψίζοντας, το σύστημα διαχείρισης βιβλιογραφικών αναφορών που αναπτύχθηκε αποτελεί ένα σταθερό και λειτουργικό εργαλείο που ανταποκρίνεται στις σύγχρονες απαιτήσεις διαχείρισης βιβλιογραφικών δεδομένων. Παρά τα επιτυχή αποτελέσματα της υλοποίησης, οι προτεινόμενες επεκτάσεις και βελτιώσεις θα μπορούσαν να προσφέρουν πρόσθετη αξία, διασφαλίζοντας την μακροχρόνια βιωσιμότητα και την αποτελεσματική χρήση του συστήματος σε ένα ευρύ φάσμα εφαρμογών.

Βιβλιογραφία

- [1] M. D. Gordon and R. R. Dufault, Guide to Research and Publishing: The Essentials of Writing and Citing Sources, 2nd ed. New York: Academic Press, 2017.
- [2] American Psychological Association, Publication Manual of the American Psychological Association, 7th ed. Washington, DC: American Psychological Association, 2020.
- [3] Modern Language Association, MLA Handbook, 8th ed. New York: Modern Language Association of America, 2016.
- [4] International Committee of Medical Journal Editors, Recommendations for the Conduct, Reporting, Editing, and Publication of Scholarly Work in Medical Journals (ICMJE Recommendations), 2020.
- [5] IEEE, IEEE Editorial Style Manual. Piscataway, NJ: IEEE, 2018.
- [6] R. Pears and G. Shields, Cite Them Right: The Essential Referencing Guide, 11th ed. London: Palgrave Macmillan, 2019.
- [7] R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, 7th ed. Boston: Pearson, 2015.
- [8] J. Casteel, SQL All-in-One For Dummies, 3rd ed. Hoboken, NJ: Wiley, 2017.
- [9] M. Gruber and S. Guelich, PHP and MySQL Web Development, 4th ed. Indianapolis, IN: Addison-Wesley, 2009.
- [10] W. Stallings and L. Brown, Computer Security: Principles and Practice, 4th ed. Boston: Pearson, 2018.
- [11] M. Bishop, Computer Security: Art and Science, 2nd ed. Boston: Addison-Wesley, 2018.
- [12] M. Chapple and J. M. Stewart, CISSP (ISC)² Certified Information Systems Security Professional Official Study Guide, 9th ed. Indianapolis, IN: Sybex, 2021.
- [13] M. K. Williams and K. A. Lane, *PHP & MySQL: Server-side Web Development*, 1st ed. Indianapolis, IN: Pearson, 2017.

- [14] M. McGrath, HTML, CSS & JavaScript Web Publishing in One Hour a Day, 7th ed. Indianapolis, IN: Sams Publishing, 2015.
- [15] S. Ghosh, Managing Bibliographies with Zotero: A Practical Guide, 1st ed. Bangalore: Packt Publishing, 2021.
- [16] T. Berners-Lee and M. Fischetti, Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web, 1st ed. New York: HarperCollins, 1999.
- [17] D. Wells and J. Williams, SQL for Data Scientists: A Beginner's Guide to Database Management and Data Wrangling, 1st ed. Berkeley, CA: Apress, 2020.
- [18] M. Garofalakis, J. Gehrke, and R. Rastogi, Data Stream Management: Processing High-Speed Data Streams, 1st ed. Berlin: Springer, 2016.
- [19] B. Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, 20th ed. Indianapolis, IN: Wiley, 2015.
- [20] E. Hatcher and O. Gospodnetic, Lucene in Action: Covers Apache Lucene 3.0, 2nd ed. Greenwich, CT: Manning Publications, 2010.
- [21] J. Dean and S. Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, 1st ed. Redmond, WA: Microsoft Research, 2004.
- [22] R. C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, 1st ed. Boston: Prentice Hall, 2008.

Παράρτημα Α- Κώδικας

Index.php

```
<?php

// Prevent caching

header("Cache-Control: no-store, no-cache, must-revalidate, max-age=0");

header("Cache-Control: post-check=0, pre-check=0", false);

header("Pragma: no-cache");


// Include necessary files

require_once 'db_connect.php';


// Start the session if it hasn't been started already

if (session_status() == PHP_SESSION_NONE) {

    session_start();

}


// Check if the user is logged in

$is_logged_in = isset($_SESSION['username']);

$is_admin = $is_logged_in && ($_SESSION['role'] === 'admin');

?>


<!DOCTYPE html>
```

```
<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Bibliography Management System</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <div class="container">

        <header>

            <h1>Bibliography Management System</h1>

            <nav>

                <ul>

                    <?php if ($is_logged_in): ?>

                        <li><a href="index.php">Home</a></li>

                        <li><a href="manage_references.php">Manage References</a></li>

                        <!-- Removed the Search References link -->

                        <!-- <li><a href="search.php">Search References</a></li> -->

                        <li><a href="reference_list.php">Reference List</a></li>

                    <?php if ($is_admin): ?>

                        <li><a href="manage_users.php">Manage Users</a></li>

                    <?php endif; ?>

                </ul>

            </nav>

        </header>

    </div>

</body>

</html>
```

```

        <li><a href="logout.php">Logout</a></li>

<?php else: ?>

        <li><a href="login_register.php">Login</a></li>

        <li><a href="login_register.php?register=true">Register</a></li>

<?php endif; ?>

</ul>

</nav>

</header>

<main>

<?php if ($is_logged_in): ?>

        <h2>Welcome, <?php echo htmlspecialchars($_SESSION['username']); ?>!</h2>

        <p>Select an option from the menu above to start managing your bibliographic
references.</p>

<?php else: ?>

        <h2>Welcome to the Bibliography Management System</h2>

        <p>Please log in or register to start managing your bibliography references.</p>

<?php endif; ?>

</main>

<footer>

        <p>&copy; 2024 Bibliography Management System. All rights reserved.</p>

```

```
<p>Developed for academic use.</p>

</footer>

</div>

</body>

</html>
```

Add_reference.php

```
<?php

// Prevent caching

header("Cache-Control: no-store, no-cache, must-revalidate, max-age=0");

header("Cache-Control: post-check=0, pre-check=0", false);

header("Pragma: no-cache");


include 'db_connect.php';

include 'validate_input.php';


// Start the session if it hasn't been started already

if (session_status() == PHP_SESSION_NONE) {

    session_start();

}
```

```

// Check if the user is logged in

if (!isset($_SESSION['username']) || !isset($_SESSION['user_id'])) {

    header('Location: login_register.php');

    exit();

}


$page_title = "Add New Reference";

$message = "";


if ($_SERVER['REQUEST_METHOD'] === 'POST') {

    error_log("Form submitted: " . print_r($_POST, true));


    $type = validate_input($_POST['type']);

    $title = validate_input($_POST['title']);

    $year = validate_input($_POST['year'], 'int');

    $journal = isset($_POST['journal']) ? validate_input($_POST['journal']) : null;

    $volume = isset($_POST['volume']) ? validate_input($_POST['volume']) : null;

    $issue = isset($_POST['issue']) ? validate_input($_POST['issue']) : null;

    $pages = isset($_POST['pages']) ? validate_input($_POST['pages']) : null;

    $publisher = isset($_POST['publisher']) ? validate_input($_POST['publisher']) : null;

    $doi = isset($_POST['doi']) ? validate_input($_POST['doi']) : null;

```

```

$user_id = $_SESSION['user_id']; // Get the user ID from the session

try {

    // Insert reference tied to the current user

    $stmt = $pdo->prepare("INSERT INTO `references` (type, title, year, journal, volume, issue,
pages, publisher, doi, user_id) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)");

    $stmt->execute([$type, $title, $year, $journal, $volume, $issue, $pages, $publisher, $doi,
$user_id]);

    $referenceId = $pdo->lastInsertId();

    if (isset($_POST['authors']) && is_array($_POST['authors'])) {

        foreach ($_POST['authors'] as $author) {

            $firstName = validate_input($author['first_name']);

            $lastName = validate_input($author['last_name']);

            $stmt = $pdo->prepare("INSERT INTO authors (FirstName, LastName) VALUES (?, ?) ON
DUPLICATE KEY UPDATE AuthorID = LAST_INSERT_ID(AuthorID)");

            $stmt->execute([$firstName, $lastName]);

            $authorId = $pdo->lastInsertId();

            $stmt = $pdo->prepare("INSERT INTO reference_author (ReferenceID, AuthorID) VALUES
(?, ?)");

            $stmt->execute([$referenceId, $authorId]);

```

```

    }

}

$message = "Reference added successfully!";

header('Location: reference_list.php');

exit();

} catch (PDOException $e) {

    error_log("PDO Exception: " . $e->getMessage());

    $message = "Error: " . $e->getMessage();

}

}

ob_start();

?>

<div class="content-wrapper">

    <div class="container">

        <?php if (!empty($message)): ?>

            <div class="message"><?php echo $message; ?></div>

        <?php endif; ?>

    </div>

    <h2>Add New Reference</h2>

```

```
<form method="post" id="referenceForm">

  <div class="form-group">

    <label for="type">Type of Source:</label>

    <select name="type" id="type" required>

      <option value="Journal Article">Journal Article</option>

      <option value="Book">Book</option>

      <option value="Conference Paper">Conference Paper</option>

      <option value="Website">Website</option>

    </select>

  </div>

  <div class="form-group">

    <label for="title">Title:</label>

    <input type="text" name="title" id="title" required>

  </div>

  <div class="form-group">

    <label for="year">Year:</label>

    <input type="number" name="year" id="year" required>

  </div>

  <div id="authors-container">
```

```

<h3>Authors</h3>

<div id="authors-list">

  <div class="author-row">

    <div class="form-group">

      <label for="first_name_0">First Name</label>

      <input type="text" id="first_name_0" name="authors[0][first_name]" required>

    </div>

    <div class="form-group">

      <label for="last_name_0">Last Name</label>

      <input type="text" id="last_name_0" name="authors[0][last_name]" required>

    </div>

    <button type="button" class="remove-author" disabled>Remove</button>

  </div>

</div>

<button type="button" id="add-author">Add Author</button>

</div>

<!-- Journal/Conference Paper specific fields -->

<div class="form-group journal-required" style="display: none;">

  <label for="journal">Journal Name:</label>

  <input type="text" name="journal" id="journal">

</div>

```

```
<div class="form-group journal-required" style="display: none;">

  <label for="volume">Volume:</label>

  <input type="text" name="volume" id="volume">

</div>
```

```
<div class="form-group journal-required" style="display: none;">

  <label for="issue">Issue:</label>

  <input type="text" name="issue" id="issue">

</div>
```

```
<div class="form-group journal-required" style="display: none;">

  <label for="pages">Pages:</label>

  <input type="text" name="pages" id="pages">

</div>
```

```
<!-- Book specific field -->
```

```
<div class="form-group book-required" style="display: none;">

  <label for="publisher">Publisher:</label>

  <input type="text" name="publisher" id="publisher">

</div>
```

```
<!-- Website specific fields -->

<div class="form-group website-required" style="display: none;">

  <label for="url">Website URL:</label>

  <input type="url" name="url" id="url">

</div>


<div class="form-group website-required" style="display: none;">

  <label for="access_date">Access Date:</label>

  <input type="date" name="access_date" id="access_date">

</div>


<div class="form-group">

  <label for="doi">DOI:</label>

  <input type="text" name="doi" id="doi">

</div>


<div class="form-group">

  <input type="submit" value="Add Reference">

</div>

</form>

</div>

</div>
```

<script>

```
// JavaScript for dynamically showing/hiding fields based on source type

document.addEventListener('DOMContentLoaded', function() {

  const typeSelect = document.getElementById('type');

  const journalFields = document.querySelectorAll('.journal-required');

  const bookFields = document.querySelectorAll('.book-required');

  const conferenceFields = document.querySelectorAll('.conference-required');

  const websiteFields = document.querySelectorAll('.website-required');


  function toggleFields() {

    const selectedType = typeSelect.value;


    // Hide all fields initially

    journalFields.forEach(field => field.style.display = 'none');

    bookFields.forEach(field => field.style.display = 'none');

    conferenceFields.forEach(field => field.style.display = 'none');

    websiteFields.forEach(field => field.style.display = 'none');


    // Remove required attribute and tooltips from all fields

    document.querySelectorAll('input').forEach(input => {
```

```

input.required = false;

input.removeAttribute('placeholder');

input.removeAttribute('title');

});

// Display and configure tooltips for required/optional fields based on the selected type
if (selectedType === 'Journal Article') {

    // Show fields for Journal Article

    journalFields.forEach(field => field.style.display = 'block');


    // Required fields

    document.getElementById('title').required = true;

    document.getElementById('title').setAttribute('placeholder', 'Required');

    document.getElementById('title').setAttribute('title', 'Required for Journal Article');


    document.getElementById('journal').required = true;

    document.getElementById('journal').setAttribute('placeholder', 'Required');

    document.getElementById('journal').setAttribute('title', 'Required for Journal Article');


    document.getElementById('year').required = true;

    document.getElementById('year').setAttribute('placeholder', 'Required');

    document.getElementById('year').setAttribute('title', 'Required for Journal Article');

```

```

// Optional fields

document.getElementById('volume').setAttribute('placeholder', 'Optional');

document.getElementById('volume').setAttribute('title', 'Optional for Journal Article');

document.getElementById('issue').setAttribute('placeholder', 'Optional');

document.getElementById('issue').setAttribute('title', 'Optional for Journal Article');

document.getElementById('pages').setAttribute('placeholder', 'Optional');

document.getElementById('pages').setAttribute('title', 'Optional for Journal Article');

document.getElementById('doi').setAttribute('placeholder', 'Optional');

document.getElementById('doi').setAttribute('title', 'Optional for Journal Article');

}

else if (selectedType === 'Book') {

    // Show fields for Book

    bookFields.forEach(field => field.style.display = 'block');


    // Required fields

    document.getElementById('title').required = true;

    document.getElementById('title').setAttribute('placeholder', 'Required');

    document.getElementById('title').setAttribute('title', 'Required for Book');


    document.getElementById('publisher').required = true;

    document.getElementById('publisher').setAttribute('placeholder', 'Required');

```

```

document.getElementById('publisher').setAttribute('title', 'Required for Book');

document.getElementById('year').required = true;

document.getElementById('year').setAttribute('placeholder', 'Required');
document.getElementById('year').setAttribute('title', 'Required for Book');


// Optional fields

document.getElementById('edition').setAttribute('placeholder', 'Optional');
document.getElementById('edition').setAttribute('title', 'Optional for Book');
document.getElementById('isbn').setAttribute('placeholder', 'Optional');
document.getElementById('isbn').setAttribute('title', 'Optional for Book');
document.getElementById('pages').setAttribute('placeholder', 'Optional');
document.getElementById('pages').setAttribute('title', 'Optional for Book');
document.getElementById('doi').setAttribute('placeholder', 'Optional');
document.getElementById('doi').setAttribute('title', 'Optional for Book');
}

else if (selectedType === 'Conference Paper') {

    // Show fields for Conference Paper

    conferenceFields.forEach(field => field.style.display = 'block');


    // Required fields

    document.getElementById('title').required = true;

```

```

document.getElementById('title').setAttribute('placeholder', 'Required');

document.getElementById('title').setAttribute('title', 'Required for Conference Paper');


document.getElementById('conference_name').required = true;

document.getElementById('conference_name').setAttribute('placeholder', 'Required');

document.getElementById('conference_name').setAttribute('title', 'Required for
Conference Paper');


document.getElementById('year').required = true;

document.getElementById('year').setAttribute('placeholder', 'Required');

document.getElementById('year').setAttribute('title', 'Required for Conference Paper');


// Optional fields

document.getElementById('pages').setAttribute('placeholder', 'Optional');

document.getElementById('pages').setAttribute('title', 'Optional for Conference Paper');

document.getElementById('doi').setAttribute('placeholder', 'Optional');

document.getElementById('doi').setAttribute('title', 'Optional for Conference Paper');

document.getElementById('publisher').setAttribute('placeholder', 'Optional');

document.getElementById('publisher').setAttribute('title', 'Optional for Conference
Paper');

}

else if (selectedType === 'Website') {

```

```

// Show fields for Website

websiteFields.forEach(field => field.style.display = 'block');


// Required fields

document.getElementById('title').required = true;

document.getElementById('title').setAttribute('placeholder', 'Required');

document.getElementById('title').setAttribute('title', 'Required for Website');


document.getElementById('url').required = true;

document.getElementById('url').setAttribute('placeholder', 'Required');

document.getElementById('url').setAttribute('title', 'Required for Website');


document.getElementById('access_date').required = true;

document.getElementById('access_date').setAttribute('placeholder', 'Required');

document.getElementById('access_date').setAttribute('title', 'Required for Website');


// Optional fields

document.getElementById('publisher').setAttribute('placeholder', 'Optional');

document.getElementById('publisher').setAttribute('title', 'Optional for Website');

document.getElementById('doi').setAttribute('placeholder', 'Optional');

document.getElementById('doi').setAttribute('title', 'Optional for Website');

}

```

```

}

typeSelect.addEventListener('change', toggleFields);

// Initialize form fields on page load

toggleFields();

});

// JavaScript for dynamically adding/removing authors

document.addEventListener('DOMContentLoaded', function() {

    const authorsList = document.getElementById('authors-list');

    const addAuthorButton = document.getElementById('add-author');

    let authorCount = 1; // Start counting from the second author

    function createAuthorRow(index) {

        const row = document.createElement('div');

        row.className = 'author-row';

        row.innerHTML = `

            <div class="form-group">

                <label for="first_name_${index}">First Name</label>

                <input type="text" id="first_name_${index}" name="authors[${index}][first_name]"

required placeholder="Required" title="Required">

```

```

</div>

<div class="form-group">

  <label for="last_name_${index}">Last Name</label>

  <input type="text" id="last_name_${index}" name="authors[${index}][last_name]"
required placeholder="Required" title="Required">

</div>

<button type="button" class="remove-author">Remove</button>

`;

return row;
}

```

```

addAuthorButton.addEventListener('click', function() {

  const newAuthorRow = createAuthorRow(authorCount);

  authorsList.appendChild(newAuthorRow);

  authorCount++;

});

authorsList.addEventListener('click', function(e) {

  if (e.target.classList.contains('remove-author')) {

    const authorRow = e.target.closest('.author-row');

    if (authorsList.children.length > 1) {

      authorRow.remove(); // Remove the author row if there is more than one

```

```
    } else {  
        alert("You must have at least one author.");  
    }  
}  
});  
});
```

```
</script>
```

```
<?php
```

```
$content = ob_get_clean();
```

```
include 'layout.php';
```

```
?>
```