



## Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής»

### Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	<b>Εφαρμογή υποβολής ερωτημάτων σε MySQL βάσεις δεδομένων</b>  <b>Application for querying MySQL databases</b>
Όνοματεπώνυμο Φοιτητή	<b>Μαρία Ελεάνα Μούκα</b>
Πατρώνυμο	<b>Ιωάννης</b>
Αριθμός Μητρώου	<b>ΜΠΣΠ 13066</b>
Επιβλέπων	<b>Ευθύμιος Αλέπης , Επίκουρος Καθηγητής</b>

Ημερομηνία Παράδοσης **Νοέμβριος 2015**

---

**Τριμελής Εξεταστική Επιτροπή**

(υπογραφή)

(υπογραφή)

(υπογραφή)

Μαρία Βίββου  
Καθηγητής

Γεώργιος Τσιχριντζής  
Καθηγητής

Ευθύμιος Αλέπης  
Επίκουρος Καθηγητής

## Περιεχόμενα

Πίνακας Εικόνων.....	4
Περίληψη.....	5
Abstract .....	5
Εισαγωγή.....	6
Κεφάλαιο 1: Εξέλιξη των διαδικτυακών εφαρμογών.....	7
Η ιστορία του διαδικτύου.....	7
Διαδικτυακές Εφαρμογές.....	8
Κεφάλαιο 2: Διαδικτυακές εφαρμογές βασισμένες σε βάσεις δεδομένων .....	10
Ο ρόλος των Βάσεων Δεδομένων .....	10
Βάσεις Δεδομένων .....	12
Πλεονεκτήματα.....	13
Κεφάλαιο 3: Σχεδίαση της εφαρμογής.....	16
Απαιτήσεις.....	16
Λειτουργικές Απαιτήσεις.....	16
Μη Λειτουργικές Απαιτήσεις.....	16
Σχεδιασμός.....	16
Data Tier.....	17
Business Tier.....	27
Presentation Tier.....	30
Κεφάλαιο 4: Υλοποίηση της εφαρμογής.....	33
Λειτουργία της εφαρμογής.....	33
Data Layer.....	34
Business Logic Layer.....	35
Συνδεσιμότητα με την Βάση Δεδομένων (JDBC) .....	35
Λειτουργικότητα (JAVA).....	35
Application Server (JSP – Tomcat).....	36
Javascript.....	37
AJAX.....	38
Presentation Layer.....	40
HyperText Markup Language.....	40
Cascading Style Sheet.....	40
javascript.....	41
jQuery.....	42
Χρήση της εφαρμογής .....	43
Κεφάλαιο 5: Συμπεράσματα.....	48
Βιβλιογραφία.....	50



## Πίνακας Εικόνων

Εικόνα 1: Αρχιτεκτονική τριών επιπέδων .....	11
Εικόνα 2: Η αρχιτεκτονική 3 επιπέδων στην εφαρμογή.....	14
Εικόνα 3: Σχεδίαση της εφαρμογής.....	16
Εικόνα 4: information_schema.....	27
Εικόνα 5: Επιλογή ΣΔΒΔ .....	28
Εικόνα 6: Δημιουργία Ερωτήματος.....	29
Εικόνα 7: Εκτέλεση Ερωτήματος.....	30
Εικόνα 8: Διεπαφή επιλογής database server.....	30
Εικόνα 9: Διεπαφή Δημιουργίας Ερωτήματος (1) .....	31
Εικόνα 10: Διεπαφή Δημιουργίας Ερωτήματος (2) .....	31
Εικόνα 11: Διεπαφή δημιουργίας ερωτήματος (3).....	32
Εικόνα 12: Διεπαφή παρουσίασης αποτελέσματος ερωτήματος .....	32
Εικόνα 13: Λειτουργία AJAX .....	39
Εικόνα 14: Παράδειγμα CSS εκφράσεων.....	41
Εικόνα 15: CSS σε αρχείο .....	41
Εικόνα 16: Επιλογή Database Server .....	43
Εικόνα 17: Εσφαλμένη προσπάθεια σύνδεσης στον database server .....	43
Εικόνα 18: Επιλογή Βάσης Δεδομένων .....	44
Εικόνα 19: Επιλογή πίνακα .....	44
Εικόνα 20: Επιλογές πεδίων και τύπου join .....	45
Εικόνα 21: Ομαδοποίηση, συναρτήσεις, ταξινόμηση αποτελέσματος .....	45
Εικόνα 22: Επιλογή συνθηκών ερωτήματος.....	46
Εικόνα 23: Προβολή αποτελέσματος .....	47

## Περίληψη

Σκοπός της παρούσας εργασίας είναι η ανάπτυξη ενός διαδικτυακού εργαλείου ανάκτησης πληροφοριών από ένα σύστημα διαχείρισης βάσεων δεδομένων. Ο χρήστης του συστήματος που θα αναπτυχθεί, θα είναι σε θέση με τον χειρισμό φιλικών γραφικών διεπαφών να υποβάλλει σε ένα ΣΔΒΔ ερωτήματα SQL για την ανάκτηση δεδομένων από τις βάσεις δεδομένων που είναι εγκατεστημένες σε αυτό. Ο χρήστης της εφαρμογής δεν θα απαιτείται να κατέχει υψηλό επίπεδο γνώσης της SQL. Θα χρειάζεται επίσης να γνωρίζει μόνο βασικά χαρακτηριστικά της δομής της βάσης δεδομένων. Η ανάπτυξη της υπ' όψη εφαρμογής είναι αφορμή για την μελέτη των τεχνολογιών που χρειάζεται να χρησιμοποιηθούν για την υλοποίηση εφαρμογών που χειρίζονται δεδομένα από βάσεις δεδομένων.

## Abstract

The aim of this thesis is the development of an online data retrieval tool from a database management system. The user of this system, will be able to user a user friendly graphical interface to submit to a DBMS SQL queries, in order to retrieve required data from databases that are kept on it. The user of the application will not be required to possess a high level of knowledge of SQL. He/she also need to know only basic features of the database structure. The development of the application leads to deep study of technologies need to be used for the system implementation.

## Εισαγωγή

Η ανάπτυξη των τεχνολογιών της πληροφορική και των τηλεπικοινωνιών οδήγησε τους πάσης φύσεως και μεγέθους οργανισμούς στο να αναπτύξουν πληροφοριακά συστήματα που να βασίζονται σε αυτές. Η αρχιτεκτονική ανάπτυξης που ακολουθήθηκε στις περισσότερες περιπτώσεις ήταν αυτή των τριών επιπέδων. Η αρχιτεκτονική αυτή διαχωρίζει τα δεδομένα, τις λειτουργίες και τις διεπαφές των πληροφοριακών συστημάτων. Στο κατώτερο επίπεδο της τοποθετείται η οργάνωσης των δεδομένων. Η συνηθέστερη προσέγγιση στα παραδοσιακά πληροφοριακά συστήματα που ακολουθείται για την οργάνωση των δεδομένων είναι οι βάσεις δεδομένων. Πρόκειται για ώριμη πλέον αντιμετώπιση του ζητήματος του χειρισμού των δεδομένων που χαρακτηρίζεται για την υψηλού επιπέδου αξιοπιστία και εξασφάλιση που προσφέρει. Στην παρούσα εργασία επιχειρείται η ανάπτυξη ενός εργαλείου υποβολής αιτήσεων ανάκτησης δεδομένων από σχεσιακές βάσεις δεδομένων που είναι αποθηκευμένες σε ένα Σχεσιακό Σύστημα Βάσεων Δεδομένων.

Το υπόλοιπο της παρούσας εργασίας είναι δομημένο ως εξής:

- Κεφάλαιο 1: Στο κεφάλαιο αυτό επιχειρείται μία σύντομη αναδρομή στην εξέλιξη των διαδικτυακών εφαρμογών σε σχέση με την ανάπτυξη των τεχνολογιών του διαδικτύου. Επίσης γίνεται αναφορά σε επιπτώσεις των εξελίξεων αυτών σε ποικίλους τομείς της ανθρώπινης δραστηριότητας.
- Κεφάλαιο 2: Οι παράγραφοι του 2<sup>ου</sup> κεφαλαίου εξειδικεύονται στην δομή των διαδικτυακών εφαρμογών που βασίζονται σε βάσεις δεδομένων. Γίνεται εκτενείς αναφορά στην αρχιτεκτονική των τριών επιπέδων. Στα πλαίσια της αναφοράς αυτής περιγράφονται οι προσεγγίσεις που έχουν επικρατήσει στις υλοποιήσεις κάθε επιπέδου της αρχιτεκτονικής. Καταγράφεται επίσης και η διαδικασία που ακολουθείται γενικά για την ανάπτυξη τέτοιου είδους εφαρμογών.
- Κεφάλαιο 3: Στοιχεία της σχεδίασης της εφαρμογής περιγράφονται στο κεφάλαιο αυτό. Αναλύονται οι απαιτήσεις της εφαρμογής, επιλέγεται η καταλληλότερη αρχιτεκτονική πάνω στην οποία θα βασιστεί η υλοποίηση της και περιγράφεται η σχεδίαση του τρόπου υλοποίησης κάθε επιπέδου της.
- Κεφάλαιο 4: Στο κεφάλαιο αυτό περιγράφονται οι επιλογές που έγιναν ώστε να υλοποιηθούν όσα καθορίστηκαν στην φάση της σχεδίασης. Καταδεικνύεται επίσης και ο τρόπος με τον οποίο μπορεί ένας χρήστης να εκμεταλλευτεί τις δυνατότητες της εφαρμογής.
- Κεφάλαιο 5: Στο κεφάλαιο αυτό καταγράφονται τα συμπεράσματα που προέκυψαν από την ανάπτυξη της εφαρμογής και από την μελέτη που απαιτήθηκε.

## Κεφάλαιο 1: Εξέλιξη των διαδικτυακών εφαρμογών

### Η ιστορία του διαδικτύου

Το Internet (International Network) αναπτύχθηκε ως δίκτυο υπολογιστών και διασυνδεδεμένων δικτύων. Είναι ένα δίκτυο που συντίθεται από διασυνδεδεμένα δίκτυα. Κάθε συνδεδεμένη συσκευή στο διαδίκτυο μπορεί μεταφέρει δεδομένα σε οποιονδήποτε άλλη συσκευή που επίσης είναι συνδεδεμένη στο διαδίκτυο. Σήμερα καλύπτει τον σύνολο της υδρογείου

Το Internet στην σημερινή του μορφή είναι εξέλιξη του ARPANET, ενός δικτύου που άρχισε να αναπτύσσεται πειραματικά για στρατιωτικού κυρίως σκοπούς, στα τέλη της δεκαετίας του 1960. Σκοπός του ήταν η διασύνδεση του Υπουργείου Άμυνας των ΗΠΑ με διάφορους στρατιωτικούς και ερευνητικούς οργανισμούς αξιολογώντας παράλληλα μία καινοτόμα μεθοδολογία μετάδοσης δεδομένων, της μεταγωγής πακέτων. Η μεθοδολογία αυτή επέτρεπε σε πολλές συνδέσεις να χρησιμοποιούν ταυτόχρονα τις ίδιες τηλεπικοινωνιακές γραμμές για αποστολή και λήψη δεδομένων. Επίσης παρέχει την δυνατότητα διατήρηση της επικοινωνίας μεταξύ δύο ακραίων ανταποκριτών ακόμα και αν ορισμένα από τα ενδιάμεσα συστήματα βρίσκονταν προσωρινά εκτός λειτουργίας..

Το 1973 στα πλαίσια του ερευνητικού προγράμματος Interneting Project έγινε η διασύνδεση ανομοιογενών δικτύων και την ανταλλαγή δεδομένων μεταξύ τους που οδήγησε στην παρουσίαση του Internet Protocol και του Transmission Control Protocol (TCP) που αφορούσε την αποστολή και λήψη μηνυμάτων απαλλαγμένων από σφάλματα. Το TCP καθόρισε τις προδιαγραφές για τη μεταφορά αρχείων μεταξύ υπολογιστών (File Transfer Protocol - FTP) και το ηλεκτρονικό ταχυδρομείο (E-mail). Το αμέσως επόμενο χρονικό διάστημα και το ARPANET επεκτάθηκε αλλά και αλλά δίκτυα συνδέθηκαν στις απολήξεις του. Η κατάσταση αυτή διατηρήθηκε για περίπου δύο δεκαετίες με αποτέλεσμα το ARPANET να λάβει μεγάλες διαστάσεις και να χρειαστεί να διασπαστεί σε δύο τμήματα – ένα για ερευνητικούς σκοπούς και ένα για στρατιωτικούς. Το 1983 ο συνδυασμός των TCP και IP προτάθηκε ως πρότυπο από το Υπουργείο Άμυνας των ΗΠΑ. Το 1985, το National Science Foundation (NSF) δημιούργησε ένα νέο γρήγορο δίκτυο, το NSFNET. Το δίκτυο αυτό ήταν βασισμένο στο TCP/IP και διασύνδεε πέντε μεγάλων δυνατοτήτων ηλεκτρονικούς υπολογιστές με την υπόλοιπη επιστημονική κοινότητα. Το δίκτυο αυτό επίσης επεκτάθηκε καθώς αρκετά εκπαιδευτικά ιδρύματα συνδέθηκαν σε αυτό. Στην συνέχεια περισσότεροι εκπαιδευτικοί και μη οργανισμοί δημιούργησαν δικτυακές υποδομές τις οποίες συνδέσαν στο παγκόσμιο δίκτυο. Με τον τρόπο αυτό άρχισε να διαμορφώνεται το διαδίκτυο. Το ARPANET μετά τις εξελίξεις αυτές εγκαταλείφθηκε. Στις αρχές της δεκαετίας του 1980 παρουσιάστηκε ο παγκόσμιος ιστός από το εργαστήριο CERN. Αναπτύχθηκε από τον Tim Berners-Lee και αποτέλεσε μία μέθοδο διασύνδεσης δικτυακών κόμβων για την ανταλλαγή δεδομένων με την μορφή υπερκειμένου μέσω του διαδικτύου. Το WWW ενίσχυσε την ανάπτυξη του διαδικτύου αφού έδωσε την δυνατότητα να παρουσιαστούν περισσότερες εφαρμογές του διαδικτύου που απευθύνονταν σε ένα μεγάλο εύρος χρηστών ξεφεύγοντας από τα στενά όρια των στρατιωτικών και εκπαιδευτικών σκοπών.

Από τα μέσα της δεκαετίας του 1990 μέχρι και σήμερα το διαδίκτυο εξελίσσεται με γοργούς ρυθμούς. Οι τεχνολογίες της πληροφορικής και των τηλεπικοινωνιών στηρίζονται σε αυτό. Η πρόσβαση στο διαδίκτυο έγινε προσιτή στο μεγαλύτερο ποσοστό του παγκόσμιου πληθυσμού καθώς αποτελεί επιδίωξη πολλών κυβερνήσεων παγκοσμίως. Η σύνδεση δεν απαιτεί πλέον την ύπαρξη ηλεκτρονικού υπολογιστή αλλά μπορεί να επιτευχθεί και από άλλες συσκευές (smartphones, κινητά τηλέφωνα, τηλεοράσεις κτλ). Οι εφαρμογές καλύπτουν πλέον το σύνολο της ανθρώπινης δραστηριότητας. Οι τελευταίες μορφές του WEB (WEB 2.0 και WEB 3.0) έθεσαν τον χρήστη ταυτόχρονα και δημιουργό του περιεχομένου και το περιεχόμενο «εξυπνότερο» με την προσθήκη στοιχείων σημασιολογίας σε αυτό. Στο σύγχρονο περιβάλλον η απουσία ενός οργανισμού – ακόμα και φυσικού προσώπου – από το διαδίκτυο μπορεί να παρομοιαστεί με αναλφαβητισμό. Το γεγονός αυτό καθιστά τις διαδικτυακές εφαρμογές ακούοντως σημαντικές για κάθε πτυχή της ανθρώπινης δραστηριότητας.



## Διαδικτυακές Εφαρμογές

Οι διαδικτυακές εφαρμογές χαρακτηρίζονται από την διάθεση των υπηρεσιών που προσφέρουν μέσω του διαδικτύου. Αυτό σημαίνει ότι δεν απαιτούν από τον τελικό χρήστη τους να έχει εγκατεστημένη την εφαρμογή στον εξοπλισμό τους. Απαιτούν να έχει σύνδεση στο διαδίκτυο και στην συσκευή που χρησιμοποιεί έναν browser. Οι διεπαφές τους είναι ιστοσελίδες που προβάλλονται στην οθόνη του τελικού χρήστη και μέσω των στοιχείων τους δίνουν την δυνατότητα για εισαγωγή και λήψη δεδομένων. Οι ιστοσελίδες είναι ανεπτυγμένες σε HTML και ανάλογα με τον τρόπο που δημιουργούνται διακρίνονται σε στατικές και δυναμικές:

- **Στατικές Ιστοσελίδες:** Πρόκειται για ιστοσελίδες που το περιεχόμενο τους παρουσιάζεται στο χρήστη ακριβώς στη μορφή που βρίσκεται αποθηκευμένο στον εξυπηρετητή. Είναι γραμμένες αποκλειστικά σε γλώσσα HTML. Χαρακτηριστικό τους είναι η αδυναμία του χρήστη να επέμβει στο περιεχόμενο της. Ο μόνος τρόπος για να μεταβληθεί το περιεχόμενο στις στατικές ιστοσελίδες είναι να τροποποιηθεί το αντίστοιχο αρχείο στον εξυπηρετητή.
- **Δυναμικές Ιστοσελίδες:** Οι ιστοσελίδες αυτές δημιουργούνται δυναμικά κατά την αίτηση του χρήστη στον server για ανάκτηση τους. Αποτέλεσε εξέλιξη των εφαρμογών του διαδικτύου που έδωσε την δυνατότητα να διατεθούν υψηλής λειτουργικότητας εφαρμογές. Στην παραδοσιακή τους μορφή χρησιμοποιούν μηχανισμούς ανάκτησης δεδομένων από διάφορες πηγές όπως βάσεις δεδομένων, αρχεία και μετασχηματισμού τους σε μορφή υπερκειμένου. Έτσι στο server διαμορφώνεται έγγραφο HTML το οποίο και επιστρέφει στον αιτούντα χρήστη μέσω του browser. Με τον τρόπο αυτό το περιεχόμενο δεν διατηρείται πάντα το ίδιο αλλά μεταβάλλεται ανάλογα με τις μεταβολές στις πηγές από τις οποίες προέρχεται. Για την ανάπτυξη δυναμικών ιστοσελίδων χρησιμοποιούνται ειδικές γλώσσες προγραμματισμού όπως η php, η jsp και η asp. Χρησιμοποιείται επίσης η γλώσσα σεναρίων javascript για να προσδώσει στοιχεία διαδραστικότητας στις ιστοσελίδες αφού μπορεί να εκτελεί σεναρία στην μεριά του χρήστη δίνοντας του την ευκαιρία να αλληλοεπιδρά με τις διαδικτυακές εφαρμογές μέσω προγραμματισμένων συμβάντων.

Κατά βάση η ανάπτυξη διαδικτυακών εφαρμογών βασίστηκε στην δημιουργία δυναμικών ιστοσελίδων που τις περισσότερες φορές επικοινωνούν με βάσεις δεδομένων. Η θεματολογία που καλύπτουν πλέον επεκτείνεται στο σύνολο της ανθρώπινης δραστηριότητας. Οι κυριότερες θεματικές περιοχές που καλύπτουν είναι:

- **Εκπαίδευση:** Έχουν αναπτυχθεί πληθώρα εφαρμογών για εκπαιδευτικούς σκοπούς. Η μορφές τους εκτείνονται από απλή παράθεση γνώσεων μέχρι προβολή πολυμέσων και στοιχείων εικονικής πραγματικότητας και τεχνητής νοημοσύνης. Είναι πολύ διαδεδομένη και η χρήση διαδικτυακών εκπαιδευτικών πλατφορμών σύγχρονης και ασύγχρονης τηλεεκπαίδευσης από εκπαιδευτικά ιδρύματα όλων των βαθμίδων.
- **Επικοινωνία:** Αποτελούν τις ευρύτερα χρησιμοποιημένες εφαρμογές. Μέσω αυτών δίνεται η ευκαιρία να επικοινωνούν μεταξύ τους ποικιλοτρόπως.
- **Διασκέδαση:** Θεωρείται η πιο κερδοφόρα βιομηχανία παραγωγής διαδικτυακών εφαρμογών. Με την χρήση διαδικτυακών εφαρμογών και λόγο των δυνατοτήτων που έχουν οι χρήστες τους μπορούν να ψυχαγωγηθούν με οποιοδήποτε τρόπο ταιριάζει στα ενδιαφέροντα του προφίλ τους.
- **Εμπόριο:** Εδώ και αρκετά χρόνια το ηλεκτρονικό εμπόριο διαθέτει σημαίνουσα θέση στην οικονομική δραστηριότητα. Οι εμπορικές συναλλαγές που διενεργούνται μέσω διαδικτύου δεν περιορίζονται μόνο στην αγορά αγαθών από φυσικά πρόσωπα αλλά επεκτείνονται και στις διεπιχειρησιακές σχέσεις αλλά και σε όλο το φάσμα των εμπορικών – οικονομικών συναλλαγών. Οι εφαρμογές του ηλεκτρονικού εμπορίου διακρίνονται πλέον από ισχυρούς μηχανισμούς marketing που κάνουν χρήση τεχνητής νοημοσύνης αλλά και αξιόπιστες διαδικασίες ασφαλείας.
- **Δημόσια Διοίκηση:** Σήμερα όλο και περισσότερες διαδικασίες της δημόσιας διοίκησης εκτελούνται με την χρήση εφαρμογών διαδικτύου. Αυτές είναι υπεύθυνες για την μείωση της γραφειοκρατίας στην διεκπεραίωση των υποθέσεων μεταξύ οργανισμών και φυσικών

προσώπων με φορείς του δημοσίου αλλά και την ελαχιστοποίηση του χρόνου αναμονής για την ολοκλήρωση των απαραίτητων διαδικασιών.

Σήμερα οι διαδικτυακές εφαρμογές κατά βάση στοχεύουν στην όσο το δυνατόν ταχύτερη περαίωση των διαδικασιών που επιμελούνται αλλά και την βελτίωση της εμπειρίας του χρήστη κατά τον πρόσβαση στις διεπαφές τους. Έχουν για αυτόν το σκοπό υιοθετηθεί καινοτόμες τεχνολογίες που εξυπηρετούν τους σκοπούς αυτούς, οι σημαντικότερες εκ των οποίων είναι:

- Asynchronous JavaScript and XML (AJAX): Το AJAX είναι μία συλλογή από τεχνολογίες που χρησιμοποιούνται στην δημιουργία διαδικτυακών εφαρμογών. Η ενσωμάτωση τους δίνει την δυνατότητα ενημέρωσης μέρους του περιεχομένου μίας ιστοσελίδας χωρίς να είναι απαραίτητο να γίνει εκ νέου αίτηση και ανάκτηση της από τον web server. Τα δεδομένα που ανακτώνται με την χρήση AJAX μηχανισμών μπορεί να είναι αποθηκευμένα σε αρχείο κειμένου, σε αρχείο XML ή σε μία βάση δεδομένων. Τις περισσότερες φορές η ανάκτηση του περιεχομένου απαιτεί την εκτέλεση λειτουργιών που έχουν αναπτυχθεί σε μία γλώσσα προγραμματισμού ανάπτυξης δυναμικών ιστοσελίδων
- HTML5: Είναι εξέλιξη της HTML που χαρακτηρίζεται από την εισαγωγή μηχανισμών για την υποστήριξη χρηστών με ειδικές ανάγκες στην και την εισαγωγή tags που εμπεριέχουν και σημασιολογικό χαρακτήρα. Στην έκδοση αυτή της HTML προστέθηκαν tags για την υποστήριξη διάδρασης του χρήστη με τις διεπαφές αλλά και την χρήση και διαχείριση πολυμεσικών στοιχείων.
- Rich Internet Applications (RIA): Αποτελούν μία προσέγγιση στην ανάπτυξη διαδικτυακών εφαρμογών με την διαλειτουργικότητας και υποστήριξη υπερκειμένου. Χαρακτηρίζονται από client - side αρχιτεκτονική που μεταφέρει μεγάλο μέρος της επεξεργασίας στην μεριά του client. Δεν αποφεύγεται η ανάγκη επικοινωνίας με τον server αλλά γίνεται με ασύγχρονο τρόπο δίνοντας στον τελικό χρήστη της αίσθηση της αυτόνομης λειτουργίας της εφαρμογής. Οι εφαρμογές που αναπτύσσονται με τις μεθόδους RIA παρουσιάζουν πλούσιες και λειτουργικές διεπαφές χρήστη. Τα διακινούμενα δεδομένα ανταλλάσσονται ταχέως με τον server. Υποστηρίζουν την ενσωμάτωση κάθε είδους πολυμέσων στις διεπαφές. Σαν τεχνολογία έγινε αρκετά δημοφιλής λόγω της ικανότητάς της να διαμορφώνει τις διεπαφές των διαδικτυακών εφαρμογών με τρόπο τέτοιο που να προσομοιάζουν σε αντίστοιχες desktop.

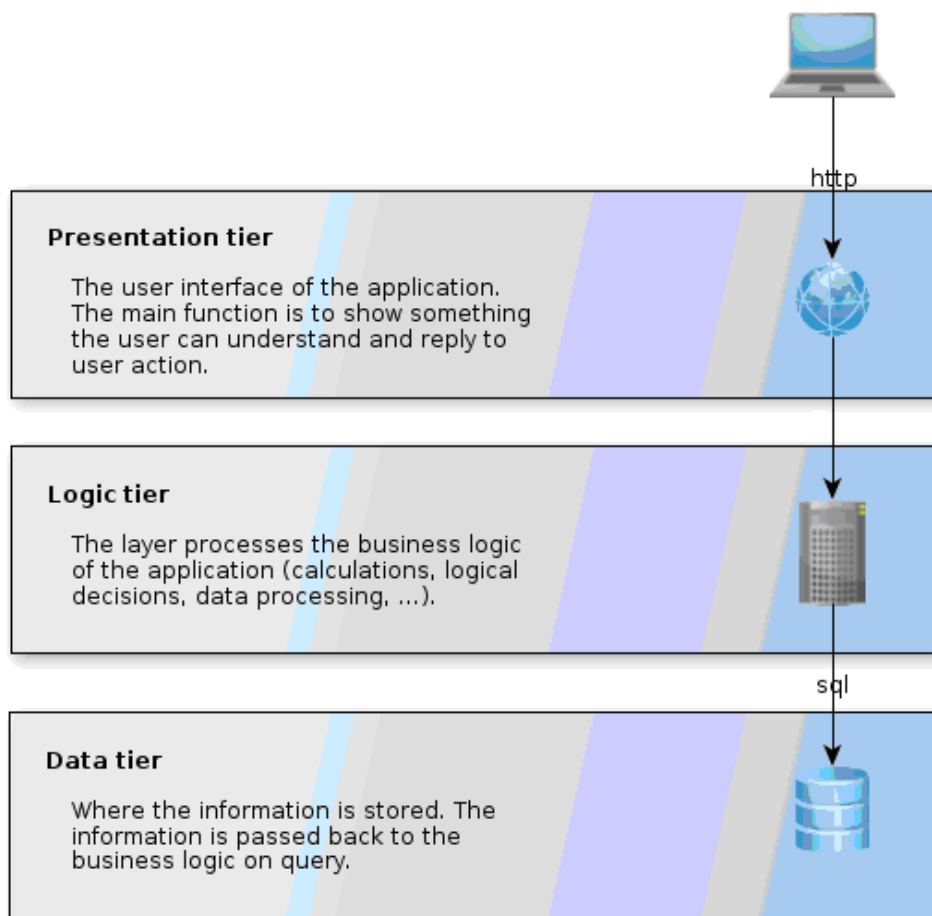
## Κεφάλαιο 2: Διαδικτυακές εφαρμογές βασισμένες σε βάσεις δεδομένων

### Ο ρόλος των Βάσεων Δεδομένων

Όπως έχει ήδη αναφερθεί, οι περισσότερες διαδικτυακές εφαρμογές αναπτύσσονται ακολουθώντας την αρχιτεκτονική των τριών επιπέδων (3-tier). Η αρχιτεκτονική 3-tier αποτελείται από τρία επίπεδα (tiers) που αποτελούν λογικό και όχι φυσικό διαχωρισμό των δεδομένων που διαχειρίζεται η εφαρμογή, των λειτουργιών που εκτελεί και των διεπαφών που προβάλλονται στον χρήστη. Τα τρία επίπεδα είναι τα εξής:

- **Data Tier:** Στο επίπεδο αυτό τοποθετούνται τα δεδομένα. Οι μηχανισμοί του επιπέδου αυτού έχουν σαν αντικείμενο την οργάνωση τους, την προετοιμασία τους για την μετάβαση τους στις λειτουργίες του ανωτέρου επίπεδο αλλά και την λήψη τους στις μορφές που επιστρέφονται από το ανώτερο επίπεδο για την κατάλληλη αποθήκευσή τους. Περιλαμβάνονται υπηρεσίες που έχουν σχέση με την ανάκτηση, τροποποίηση και την αποθήκευση δεδομένων. Η οργάνωση των δεδομένων μπορεί να γίνει σε βάσεις δεδομένων, σε σύστημα αρχείων ή σε κάποια μορφή XML περιγραφής.
- **Business Tier:** Στο επίπεδο αυτό της αρχιτεκτονικής τοποθετείται η λειτουργικότητα της εφαρμογής. Σε αυτό το επίπεδο τοποθετούνται οι λειτουργικές μονάδες της εφαρμογής. Συνήθως αυτές υλοποιούνται με προγράμματα γραμμένα σε κάποια γλώσσα προγραμματισμού ανάπτυξης δυναμικών ιστοσελίδων.
- **Presentation Tier:** Είναι το πλησιέστερο προς τον τελικό χρήστη επίπεδο της αρχιτεκτονικής και περιλαμβάνει τις διεπαφές της. Καθορίζει δηλαδή τον τρόπο με τον οποίο επικοινωνούν ο χρήστης με την εφαρμογή (πως εισάγονται οι παράμετροι των λειτουργιών της και πως παρουσιάζονται τα αποτελέσματα τους).

Σχηματικά η αρχιτεκτονική των τριών επιπέδων παρουσιάζεται στην επόμενη εικόνα:



**Εικόνα 1: Αρχιτεκτονική τριών επιπέδων**

Οι εφαρμογές που είναι ανεπτυγμένες με την αρχιτεκτονική αυτή παρουσιάζουν μία σειρά από πλεονεκτήματα όπως:

- Οι μεταβολές που πραγματοποιούνται στην σχεδίαση ή την υλοποίηση σε κάποιο από τα επίπεδα της αρχιτεκτονικής δεν επηρεάζει τα άλλα επίπεδα της. Κατά συνέπεια τυχόν αναβαθμίσεις δεν οδηγούν σε καθολική αλλαγή της δομής και των λειτουργιών της εφαρμογής.
- Οι λειτουργίες της εφαρμογής αναπτύσσονται και λειτουργούν σε κεντρικές διατάξεις και όχι στις διατάξεις των χρηστών. Έτσι τυχόν αλλαγές είναι άμεσα ορατές σε όλους τους χρήστες χωρίς να χρειάζεται να προβούν σε κάποια ενέργεια οι ίδιοι.
- Τα συστατικά της εφαρμογής μπορούν να διαμοιραστούν σε πολλαπλές δικτυωμένες διατάξεις, επιτρέποντας έτσι την εξισορρόπηση του φόρτου εργασίας του κάθε μία από αυτές.
- Εκμηδενίζονται οι περιορισμοί πολλαπλής πρόσβασης στα δεδομένα αφού οι επικοινωνίες των λειτουργικών μονάδων της εφαρμογής με τις βάσεις δεδομένων είναι σε θέση να διαχειρίζονται πολλαπλές προσβάσεις.

Οι βάσεις δεδομένων τοποθετούνται στο κατώτερο μέρος της αρχιτεκτονικής. Αποτελούν τρόπο οργάνωσης των δεδομένων. Τα συστήματα διαχείρισης τους – προγράμματα που χρησιμοποιούνται για εισαγωγή, τροποποίηση, διαγραφή και ανάκτηση δεδομένων καθώς και άλλων εργασιών διαχείρισης – προσφέρουν μηχανισμούς για την επικοινωνία τους με το ανώτερο επίπεδο της λειτουργικότητας.

## Βάσεις Δεδομένων

Η προσέγγιση των βάσεων δεδομένων αποτελεί έναν οργανωμένο τρόπο αποθήκευσης δεδομένων και πρόσβασής σε αυτά. Τα συστήματα βάσεων δεδομένων είναι ολοκληρωμένα συστήματα που περιλαμβάνουν:

- Τα αποθηκευμένα δεδομένα
- Λογισμικό διαχείρισης τους
- Μηχανισμούς ασφαλείας των δεδομένων

Ανάλογα με τον τρόπο με τον οποίο οργανώνουν την αποθήκευση των δεδομένων τα συστήματα βάσεων δεδομένων διακρίνονται σε διαφορετικές κατηγορίες όπως:

- **Ιεραρχικές:** Τα δεδομένα είναι οργανωμένα σε ιεραρχικές μεταξύ τους συσχετίσεις.
- **Αντικειμενοστραφείς:** Τα δεδομένα είναι οργανωμένα σε κλάσεις αντικειμένων οι οποίες μπορεί να σχηματίζουν ιεραρχίες ή να συσχετίζονται με διάφορους τρόπους μεταξύ τους.
- **Σχεσιακές:** Τα δεδομένα είναι οργανωμένα σε δισδιάστατους πίνακες όπου κάθε γραμμή περιγράφει μία αποθηκευμένη οντότητα και κάθε στήλη περιλαμβάνει τις τιμές των χαρακτηριστικών των οντοτήτων που περιγράφονται από τον πίνακα. Οι πίνακες μπορεί να συσχετίζονται μεταξύ τους.
- **Αντικειμενοσχεσιακές:** Πρόκειται για συνδυασμό των προσεγγίσεων των αντικειμενοστραφών και των σχεσιακών. Η αποθήκευση των δεδομένων γίνεται σε αντικείμενα κλάσεων και ο χειρισμός τους με όρους σχεσιακών βάσεων δεδομένων.

Τα δεδομένα που υπάρχουν στις βάσεις δεδομένων είναι :

- **Ολοκληρωμένα (Integrated):** Τα δεδομένα πρέπει να είναι αποθηκευμένα σε ομοιόμορφα οργανωμένα σύνολα αρχείων. Χαρακτηριστικό είναι το γεγονός ότι τα ίδια δεδομένα δεν επαναλαμβάνονται αλλά ούτε υπάρχουν πλεονάζοντα δεδομένα.
- **Καταμεριζόμενα (Shared):** Πρόσβαση στα ίδια δεδομένα μπορεί να έχουν ταυτόχρονα περισσότεροι του ενός χρήστες ή εφαρμογές. δηλ. να μπορούν περισσότεροι του ενός χρήστες να βλέπουν και να μοιράζονται τα ίδια δεδομένα την ίδια χρονική στιγμή.

Τα πλεονεκτήματα της υλοποίησης του Data Layer με χρήση βάσης δεδομένων είναι:

- Ο περιορισμός της πολλαπλής αποθήκευσης των ίδιων στοιχείων (redundancy).
- Ο καταμερισμός (sharing) των ίδιων στοιχείων σ' όλους τους χρήστες της εφαρμογής.
- Η ομοιομορφία (uniformity) στον χειρισμό και την αναπαράσταση των δεδομένων.
- Η επιβολή ισχυρών κανόνων ασφαλείας (security) στην χρήση των δεδομένων.
- Η διατήρηση της ακεραιότητας (integrity) και της αξιοπιστίας (reliability) των δεδομένων.
- Η ανεξαρτησία της διαχείρισης των δεδομένων (data independence) και των προγραμμάτων από τον φυσικό τρόπο αποθήκευσης των τους.

Η πλέον διαδεδομένη μορφή βάσεων δεδομένων είναι οι σχεσιακές. Τα δεδομένα συνδέονται μεταξύ τους με σχέσεις (relations), οι οποίες προκύπτουν από τα κοινά πεδία που υπάρχουν σε διαφορετικά αρχεία. Τα αρχεία αποκαλούνται πίνακες (tables), οι εγγραφές γραμμές (rows) και τα πεδία στήλες (columns). Η ύπαρξη μιας κοινής τιμής στα πεδία δύο αρχείων καθορίζει και μια σχέση μεταξύ των γραμμών διαφορετικών πινάκων. Οι σχεσιακές βάσεις δεδομένων έχουν το πλεονέκτημα ότι είναι λογικά κατανοητές, είναι ευέλικτες στον χειρισμό τους και εύκολες στις αλλαγές σε δομές και δεδομένα. Αναλυτικότερα τα δεδομένα στις σχεσιακές βάσεις δεδομένων είναι οργανωμένα στις ακόλουθες στοιχειώδεις μορφές:

- **Πεδίο (Field),** είναι το μικρότερο κομμάτι δεδομένων στο οποίο μπορούμε να αναφερθούμε και περιέχει ένα μόνο χαρακτηριστικό ή ιδιότητα ενός στοιχείου της βάσης δεδομένων.
- **Εγγραφή (Record),** είναι ένα σύνολο από διαφορετικά πεδία που περιέχει όλες τις πληροφορίες για ένα στοιχείο της βάσης δεδομένων.
- **Αρχείο (File),** είναι ένα σύνολο από πολλά παρόμοια στοιχεία (εγγραφές) της βάσης δεδομένων.

- Πρωτεύον Κλειδί (Primary Key), είναι ένα πεδίο ή συνδυασμός πεδίων που χαρακτηρίζει μοναδικά μια εγγραφή.
- Κλειδί (Key), είναι ένα πεδίο που δεν έχει κατ' ανάγκη μοναδική τιμή και που μπορούμε να το χρησιμοποιήσουμε για να κάνουμε αναζήτηση σ' ένα αρχείο.
- Ξένο Κλειδί (Foreign Key), είναι ένα πεδίο που έχει το ίδιο σύνολο τιμών με το πρωτεύον κλειδί ενός άλλου αρχείου.

## Πλεονεκτήματα

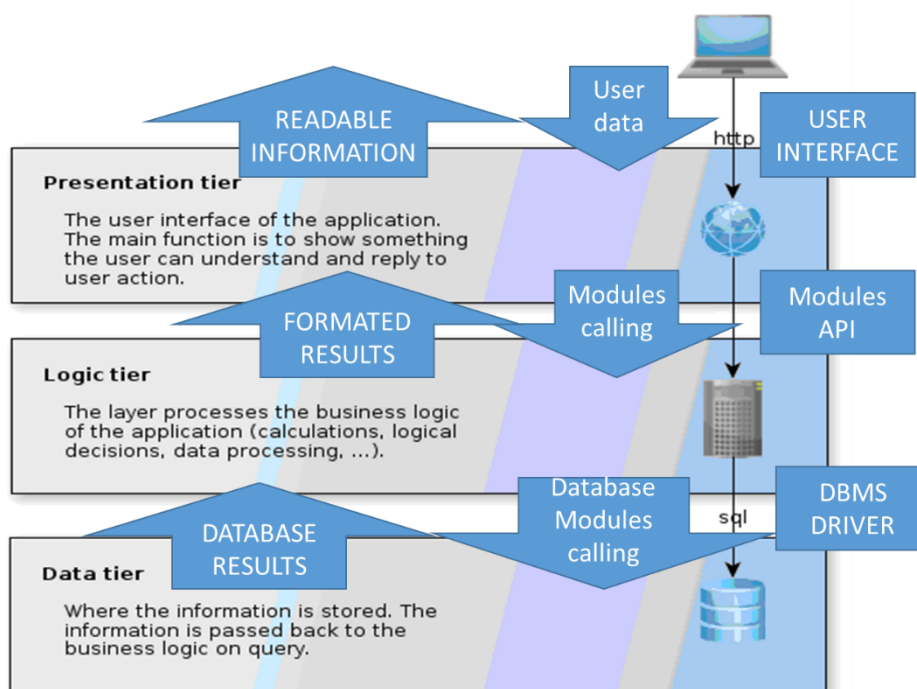
Οι διαδικτυακές εφαρμογές που βασίζονται σε βάσεις δεδομένων αναπτύσσονται στα πρότυπα της αρχιτεκτονικής των τριών επιπέδων και το κατώτερο επίπεδο της υλοποιείται με συστήματα βάσεων δεδομένων. Με τον τρόπο αυτό οι εφαρμογές αυτές ευεργετούνται τόσο από τα πλεονεκτήματα της αρθρωτής φύσης της αρχιτεκτονικής όσο και από τα πλεονεκτήματα της χρήσης βάσεων δεδομένων.

Για την υλοποίηση του επιπέδου δεδομένων εγκαθίσταται ένας database server ο οποίος υποστηρίζει ένα συγκεκριμένο Σύστημα Διαχείρισης Βάσης Δεδομένων. Κάθε ΣΔΒΔ διαθέτει μηχανισμούς για:

- Την διαχείριση των δεδομένων (καταχώρηση, διαγραφή, τροποποίηση, ανάκτηση)
- Την διαχείριση των δομών αποθήκευσης των δεδομένων.
- Την ασφάλεια των δεδομένων (αντίγραφα ασφαλείας, ακεραιότητα δεδομένων, αξιοπιστία δεδομένων, έλεγχος πρόσβασης)
- Την παροχή κατάλληλων οδηγιών για την σύνδεση λειτουργικών μονάδων σε αυτά με σκοπό τον χειρισμό των δεδομένων.
- Την διάθεση APIs για την απευθείας πρόσβαση στον χειρισμό των δεδομένων αλλά και σε εργασίες διαχείρισης του συστήματος.
- Την διάθεση εφαρμογών για την απευθείας πρόσβαση στον χειρισμό των δεδομένων αλλά και σε εργασίες διαχείρισης του συστήματος.

Οι λειτουργικές μονάδες που χρειάζεται να κάνουν χρήση των αποθηκευμένων στην βάση δεδομένων ή να καταχωρήσουν δεδομένα σε αυτή, χρησιμοποιούν κατάλληλα APIs των οδηγιών ώστε να αποκτήσουν πρόσβαση στις αντίστοιχες λειτουργίες του ΣΔΒΔ. Μέσω των APIs αυτών μεταβιβάζονται οι κατάλληλες εντολές στην βάση δεδομένων. Με την εκτέλεση των εντολών αυτών δημιουργείται η απόκριση του ΣΔΒΔ το οποίο επίσης μέσω του API μεταβιβάζεται στο ανώτερο επίπεδο στην λειτουργική μονάδα που το κάλεσε. Το αποτέλεσμα αυτό είναι σε μορφή διαχειρίσιμη από το επίπεδο λειτουργιών. Από την περιγραφή αυτή καθίσταται φανερό ότι ο τρόπος υλοποίησης των λειτουργικών μονάδων δεν επηρεάζει τον τρόπο υλοποίησης της οργάνωσης των δεδομένων αρκεί να παρέχονται οδηγοί για την μεταξύ τους επικοινωνία. Το ίδιο ισχύει και για το αντίστροφο αφού η υλοποίηση του τρόπου οργάνωσης των δεδομένων επίσης δεν επηρεάζει τον τρόπο υλοποίησης των λειτουργιών.

Αφού οι λειτουργικές μονάδες ολοκληρώσουν τις δοσοληψίες με την βάση δεδομένων με τον τρόπο που αναφέρθηκε παραπάνω, χρειάζεται να οδηγήσουν τα αποτελέσματα των δοσοληψιών αυτών στο ανώτερο επίπεδο που είναι υπεύθυνο για την διαμόρφωση των διεπαφών της εφαρμογής σε μορφή αναγνωρίσιμη από τον άνθρωπο. Οι διεπαφές συνήθως αναπτύσσονται σε ιστοσελίδες γραμμένες σε HTML. Έτσι οι λειτουργικές μονάδες θα πρέπει να προετοιμάσουν τα αποτελέσματα ώστε να παραληφθούν από το ανώτερο επίπεδο στην μορφή εκείνη που είναι κατάλληλη για να ενσωματωθεί στις διεπαφές της. Παρόμοια διαδικασία εκτελείται και στην αντίστροφη κατεύθυνση. Από τις διεπαφές του ανωτέρου επιπέδου της εφαρμογής συλλέγονται τα δεδομένα του χρήστη και οδηγούνται στις λειτουργικές μονάδες του μεσαίου επιπέδου όπου υφίστανται κατάλληλη επεξεργασία. Σχηματικά η λειτουργία των διαδικτυακών εφαρμογών με τον τρόπο αυτό φαίνεται στην παρακάτω εικόνα:



Εικόνα 2: Η αρχιτεκτονική 3 επιπέδων στην εφαρμογή

Η συνεργασία των διεπαφών με τις λειτουργικές μονάδες, όπως συνοπτικά περιεγράφηκε παραπάνω, παρέχει αντίστοιχα πλεονεκτήματα με την επικοινωνία λειτουργικών μονάδων και βάσεων δεδομένων. Η υλοποίηση του ανωτέρου επιπέδου της αρχιτεκτονικής δεν επηρεάζει τον τρόπο υλοποίησης των λειτουργικών μονάδων της και αντίστροφα ο τρόπος με τον οποίο υλοποιούνται οι λειτουργικές μονάδες επίσης δεν επηρεάζει τον τρόπο που αναπτύσσονται οι διεπαφές.

Ο τρόπος ανάπτυξης των διαδικτυακών εφαρμογών που βασίζονται σε βάσεις δεδομένων παρέχουν γενικότερα τα ακόλουθα πλεονεκτήματα:

- Είναι εύκολο να αναβαθμιστούν: Οι ανάπτυξη των τεχνολογιών του διαδικτύου δημιουργεί διαρκώς νέες προκλήσεις όσον αφορά την ανάπτυξη διαδικτυακών εφαρμογών. Κατά συνέπεια συχνά παρίσταται η ανάγκη για την υλοποίηση εκ νέου κάποιου από τα επίπεδα της αρχιτεκτονικής (του συνόλου ή μέρους). Στο επίπεδο των δεδομένων αυτό μπορεί να γίνει με την εγκατάσταση μίας νέας έκδοσης των εφαρμογών που χρησιμοποιούνται, με την καλύτερη οργάνωση της φυσικής αποθήκευσης των δεδομένων ακόμα και με την πλήρη μετάβαση σε τελείως διαφορετική φιλοσοφία διατήρησης των δεδομένων (πχ την μετάβαση από σχεσιακές βάσεις δεδομένων σε αντικειμενοστραφείς). Στο επίπεδο των εφαρμογών μπορεί να συμβεί με την υιοθέτηση νέων καινοτόμων τεχνολογιών και μεθοδολογιών που αποσκοπούν στην βελτίωση της αποδοτικότητας, της προσθαφείρεση λειτουργιών αλλά και την καθολική μετάβαση από μία τεχνολογία ή τεχνοτροπία σε μία άλλη. Τέλος συχνά παρατηρούνται αλλαγές στις διεπαφές των διαδικτυακών εφαρμογών οι οποίες είτε περιορίζονται σε απλές εικαστικές παρεμβάσεις είτε έχουν να κάνουν με την διευκόλυνση χρήσης των εφαρμογών (χειρισμός από άτομα με ειδικές ανάγκες, ηλικιωμένους, ανθρώπους μη εξοικειωμένους με την χρήση ηλεκτρονικών υπολογιστών). Όλες αυτές οι αλλαγές μπορεί να συμβαίνουν χωρίς να είναι απαραίτητο να συμβούν αλλαγές και στα άλλα επίπεδα της αρχιτεκτονικής.
- Ευκολότερη διαχείριση: Η διαχείριση των εφαρμογών γίνεται ευκολότερη αφού οι εργασίες της μπορούν να διαιρεθούν στα αντίστοιχα επίπεδα της αρχιτεκτονικής. Έτσι η διαχείριση των διαδικτυακών εφαρμογών διακρίνεται στην διαχείριση της βάσης δεδομένων, των λειτουργικών μονάδων και των διεπαφών. Η ενασχόληση με κάθε είδος διαχείρισης εξειδικεύεται και κατά συνέπεια γίνεται αποδοτικότερη και πιο αξιόπιστη.

- **Τυποποίηση:** Η ανάπτυξη των εφαρμογών κατά αυτόν τον τρόπο ευνοεί την τυποποίηση της όλης διαδικασίας. Είναι εφικτό να δημιουργηθούν πρότυπα ανάπτυξης ανά επίπεδο της αρχιτεκτονικής τα οποία να μπορούν να επαναχρησιμοποιούνται – με τις κατάλληλες προσαρμογές - σε άλλες εφαρμογές. Χαρακτηριστικό παράδειγμα είναι η χρήση των Open Source Content Management Systems όπου προτυποποιημένες διαδικτυακές εφαρμογές που βασίζονται σε βάσεις δεδομένων είναι δυνατόν μέσα από απλές προσαρμογές και προσθήκες να ικανοποιήσουν οποιεσδήποτε απαιτήσεις.



## Κεφάλαιο 3: Σχεδίαση της εφαρμογής

### Απαιτήσεις

#### Λειτουργικές Απαιτήσεις

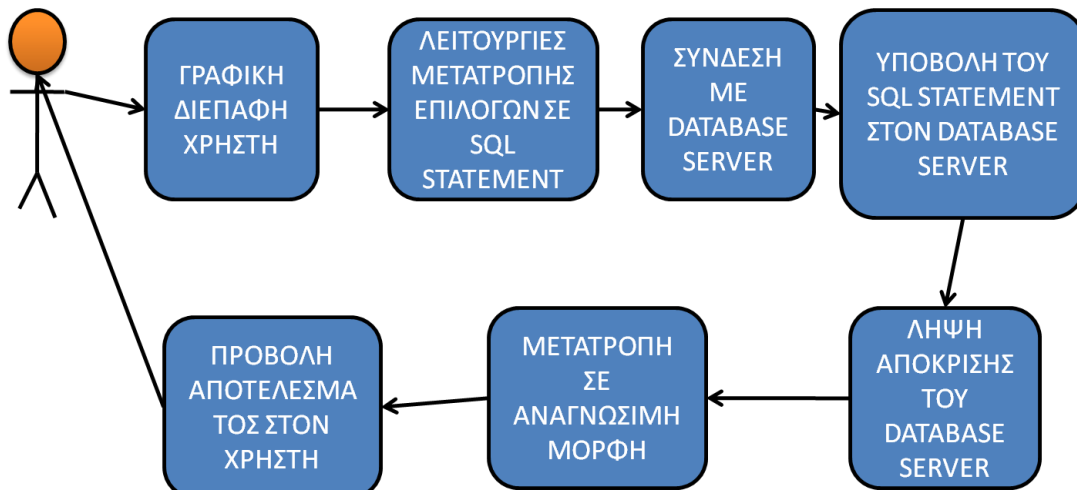
- Η δυνατότητα υποβολή sql ερωτημάτων σε βάση δεδομένων εγκατεστημένη σε δεδομένο database server χωρίς να είναι απαραίτητη η γνώση της sql μέσω jdbc διασύνδεσης.
- Η δυνατότητα χειρισμού της εφαρμογής από προσωπικό που δεν έχει υψηλό επίπεδο εξοικείωσης με τις βάσεις δεδομένων και γενικότερα με την πληροφορική.

#### Μη Λειτουργικές Απαιτήσεις

- Δυνατότητα σύνδεσης με Σύστημα Διαχείρισης Βάσης Δεδομένων.
- Δυνατότητα σύνδεσης σε κάθε βάση δεδομένων MySql σε οποιονδήποτε database server.

### Σχεδιασμός

Η εφαρμογή χρειάζεται να παρέχει στον τελικό χρήστη ένα κατάλληλο γραφικό περιβάλλον που να του δίνει την δυνατότητα να επιλέγει τα δεδομένα που θέλει να του προβληθούν εφαρμόζοντας επί αυτών τα επίσης επιθυμητά φίλτρα. Τις επιλογές του αυτές ο χρήστης θα τις υποβάλλει στο σύστημα όπου με κατάλληλες λειτουργίες θα μετατρέπονται σε sql statement. Στην συνέχεια θα επιτυγχάνεται σύνδεση στην επιλεγμένη βάση δεδομένων στον database server με jdbc διασύνδεση και υποβολή του sql statement. Η απόκριση του server θα υφίσταται επεξεργασία από κατάλληλες λειτουργίες, θα μετατρέπεται σε αναγνώσιμη από τον χρήστη μορφή και θα αποστέλλεται στο τερματικό του χρήστη. Η διαδικασία αυτή φαίνεται σχηματικά παρακάτω.



Εικόνα 3: Σχεδίαση της εφαρμογής

Επιπλέον θεωρήθηκε σαν επιπλέον απαίτηση η δυνατότητα της πρόσβασης στον database server από απόσταση ώστε να μην είναι αναγκαία η φυσική παρουσία του χρήστη στον χώρο του database server.

Με βάση τα παραπάνω επιλέχθηκε 3-tier αρχιτεκτονική για την ανάπτυξη της εφαρμογής. Στις επόμενες παραγράφους φαίνεται το πώς σχεδιάστηκε η εφαρμογή σε κάθε επίπεδο της αρχιτεκτονικής της.

## Data Tier

Στο κατώτερο επίπεδο της αρχιτεκτονικής αναπτύσσεται στο λεξικό δεδομένων του συστήματος διαχείρισης βάσης δεδομένων στο οποίο τις βάσεις θα υποβάλλουν ερωτήματα οι λειτουργίες της εφαρμογής. Δεδομένου ότι το σχήμα της βάσης δεδομένων παρέχεται έτοιμο από την υλοποίηση του συστήματος και ότι η εφαρμογή είναι προσανατολισμένη σε ΣΔΒΔ MySQL, η βάση δεδομένων με τον οποίο θα υλοποιηθεί το Data Tier θα είναι το information\_schema. Πρόκειται για μία βάση δεδομένων που περιλαμβάνεται σε κάθε ΣΔΒΔ MySQL και στην οποία οργανώνονται δεδομένα σχετικά με την δομή, οργάνωση και περιεχόμενο όλων των υπολοίπων βάσεων δεδομένων που περιλαμβάνονται στο σύστημα. Οι πίνακες που περιλαμβάνει η βάση δεδομένων είναι:

CHARACTER\_SETS: Περιλαμβάνει δεδομένα σχετικά με τα character sets που υποστηρίζονται. Η γραμμογράφηση του φαίνεται στην επόμενη εικόνα:

<b>INFORMATION_SCHEMA Name</b>	<b><u>SHOW</u> Name</b>	<b>Remarks</b>
<b>CHARACTER_SET_NAME</b>	<b>Charset</b>	
<b>DEFAULT_COLLATE_NAME</b>	<b>Default collation</b>	
<b>DESCRIPTION</b>	<b>Description</b>	MySQL extension
<b>MAXLEN</b>	<b>Maxlen</b>	MySQL extension

COLLATIONS: Περιλαμβάνει δεδομένα σχετικά με τα collations που υποστηρίζονται. Η γραμμογράφηση του φαίνεται στην επόμενη εικόνα:

<b>INFORMATION_SCHEMA Name</b>	<b><u>SHOW</u> Name</b>	<b>Remarks</b>
<b>COLLATION_NAME</b>	<b>Collation</b>	
<b>CHARACTER_SET_NAME</b>	<b>Charset</b>	MySQL extension
<b>ID</b>	<b>Id</b>	MySQL extension
<b>IS_DEFAULT</b>	<b>Default</b>	MySQL extension
<b>IS_COMPILED</b>	<b>Compiled</b>	MySQL extension
<b>SORTLEN</b>	<b>Sortlen</b>	MySQL extension

COLLATION\_CHARACTER\_SET\_APPLICABILITY: Προσδιορίζει το ποια character sets είναι κατάλληλα για collations. Η γραμμογράφηση του πίνακα φαίνεται στην επόμενη εικόνα.

<b>INFORMATION_SCHEMA Name</b>	<b><u>SHOW</u> Name</b>	<b>Remarks</b>
<b>COLLATION_NAME</b>	<b>Collation</b>	
<b>CHARACTER_SET_NAME</b>	<b>Charset</b>	

COLUMNS: Περιλαμβάνει πληροφορίες για πεδία που περιλαμβάνονται στους πίνακες των βάσεων δεδομένων του ΣΔΒΔ. Είναι από τους πίνακες με αρκετό ενδιαφέρον για την εφαρμογή διότι περιλαμβάνει πληροφορίες που μπορεί να επηρεάζουν τον τρόπο με τον οποίο θα χρησιμοποιηθούν τα

πεδία που θα επιλέγονται για τα ερωτήματα που θα σχηματίζονται από τους χρήστες. Η γραμμογράφηση του πίνακα έχει όπως παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
TABLE_CATALOG		NULL
TABLE_SCHEMA		
TABLE_NAME		
COLUMN_NAME	Field	
ORDINAL_POSITION		see notes
COLUMN_DEFAULT	Default	
IS_NULLABLE	Null	
DATA_TYPE	Type	
CHARACTER_MAXIMUM_LENGTH	Type	
CHARACTER_OCTET_LENGTH		
NUMERIC_PRECISION	Type	
NUMERIC_SCALE	Type	
CHARACTER_SET_NAME		
COLLATION_NAME	Collation	
COLUMN_TYPE	Type	MySQL extension
COLUMN_KEY	Key	MySQL extension
EXTRA	Extra	MySQL extension
PRIVILEGES	Privileges	MySQL extension
COLUMN_COMMENT	Comment	MySQL extension

COLUMN\_PRIVILEGES: Περιλαμβάνει πληροφορίες για δικαιώματα χρηστών σε πεδία που περιλαμβάνονται στους πίνακες των βάσεων δεδομένων του ΣΔΒΔ.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
GRANTEE		'user_name'@'host_name' value
TABLE_CATALOG		NULL
TABLE_SCHEMA		
TABLE_NAME		
COLUMN_NAME		
PRIVILEGE_TYPE		
IS_GRANTABLE		

**KEY\_COLUMN\_USAGE:** Περιλαμβάνει πληροφορίες για πεδία κλειδιά των πινάκων των βάσεων δεδομένων του ΣΔΒΔ στα οποία επιβάλλονται περιορισμοί. Η γραμμογράφηση του φαίνεται παρακάτω

<b>INFORMATION_SCHEMA Name</b>	<b>SHOW Name</b>	<b>Remarks</b>
CONSTRAINT_CATALOG		NULL
CONSTRAINT_SCHEMA		
CONSTRAINT_NAME		
TABLE_CATALOG		
TABLE_SCHEMA		
TABLE_NAME		
COLUMN_NAME		
ORDINAL_POSITION		
POSITION_IN_UNIQUE_CONSTRAINT		
REFERENCED_TABLE_SCHEMA		
REFERENCED_TABLE_NAME		
REFERENCED_COLUMN_NAME		

**ENGINES:** Στο πίνακα αυτόν καταχωρούνται πληροφορίες για τους τρόπους αποθήκευσης των πινάκων που είναι διαθέσιμοι στο ΣΔΒΔ. Η γραμμογράφηση του πίνακα φαίνεται παρακάτω.

<b>INFORMATION_SCHEMA Name</b>	<b>SHOW Name</b>	<b>Remarks</b>
ENGINE	Engine	MySQL extension
SUPPORT	Support	MySQL extension
COMMENT	Comment	MySQL extension
TRANSACTIONS	Transactions	MySQL extension
XA	XA	MySQL extension
SAVEPOINTS	Savepoints	MySQL extension

**EVENTS:** Πίνακας που περιλαμβάνει πληροφορίες για προγραμματισμένα συμβάντα που είναι να γίνουν στο ΣΔΒΔ. Η γραμμογράφηση του πίνακα φαίνεται στην επόμενη εικόνα.

<b>INFORMATION_SCHEMA Name</b>	<b><u>SHOW</u> Name</b>	<b>Remarks</b>
EVENT_CATALOG		NULL, MySQL extension
EVENT_SCHEMA	Db	MySQL extension
EVENT_NAME	Name	MySQL extension
DEFINER	Definer	MySQL extension
TIME_ZONE	Time zone	MySQL extension
EVENT_BODY		MySQL extension
EVENT_DEFINITION		MySQL extension
EVENT_TYPE	Type	MySQL extension
EXECUTE_AT	Execute at	MySQL extension
INTERVAL_VALUE	Interval value	MySQL extension
INTERVAL_FIELD	Interval field	MySQL extension
SQL_MODE		MySQL extension
STARTS	Starts	MySQL extension
ENDS	Ends	MySQL extension
STATUS	Status	MySQL extension
ON_COMPLETION		MySQL extension
CREATED		MySQL extension
LAST_ALTERED		MySQL extension
LAST_EXECUTED		MySQL extension
EVENT_COMMENT		MySQL extension
ORIGINATOR	Originator	MySQL extension
CHARACTER_SET_CLIENT	character_set_client	MySQL extension
COLLATION_CONNECTION	collation_connection	MySQL extension
DATABASE_COLLATION	Database Collation	MySQL extension

GLOBAL\_STATUS και SESSION\_STATUS: Πίνακες που παρέχουν πληροφορίες για την κατάσταση του ΣΔΒΔ. Η γραμμογράφηση τους φαίνεται παρακάτω.

<b>INFORMATION_SCHEMA Name</b>	<b><u>SHOW</u> Name</b>	<b>Remarks</b>
VARIABLE_NAME	Variable_name	
VARIABLE_VALUE	Value	

GLOBAL\_VARIABLES και SESSION\_VARIABLES: Οι πίνακες αυτοί περιλαμβάνουν πληροφορίες για μεταβλητές που χρησιμοποιούνται για την περιγραφή της κατάστασης του ΣΔΒΔ. Η μορφή τους φαίνεται στην παρακάτω εικόνα.

<b>INFORMATION_SCHEMA Name</b>	<b><u>SHOW</u> Name</b>	<b>Remarks</b>
VARIABLE_NAME	Variable_name	
VARIABLE_VALUE	Value	

**PARTITIONS:** Πρόκειται για πίνακα που περιλαμβάνει πληροφορίες για partitions πινάκων των βάσεων του ΣΔΒΔ. Η δομή του φαίνεται παρακάτω.

<b>INFORMATION_SCHEMA Name</b>	<b>SHOW Name</b>	<b>Remarks</b>
TABLE_CATALOG		MySQL extension
TABLE_SCHEMA		MySQL extension
TABLE_NAME		MySQL extension
PARTITION_NAME		MySQL extension
SUBPARTITION_NAME		MySQL extension
PARTITION_ORDINAL_POSITION		MySQL extension
SUBPARTITION_ORDINAL_POSITION		MySQL extension
PARTITION_METHOD		MySQL extension
SUBPARTITION_METHOD		MySQL extension
PARTITION_EXPRESSION		MySQL extension
SUBPARTITION_EXPRESSION		MySQL extension
PARTITION_DESCRIPTION		MySQL extension
TABLE_ROWS		MySQL extension
AVG_ROW_LENGTH		MySQL extension
DATA_LENGTH		MySQL extension
MAX_DATA_LENGTH		MySQL extension
INDEX_LENGTH		MySQL extension
DATA_FREE		MySQL extension
CREATE_TIME		MySQL extension
UPDATE_TIME		MySQL extension
CHECK_TIME		MySQL extension
CHECKSUM		MySQL extension
PARTITION_COMMENT		MySQL extension
NODEGROUP		MySQL extension
TABLESPACE_NAME		MySQL extension

**PLUGINS:** Πίνακας που περιλαμβάνει για Plugins εγκατεστημένα στο ΣΔΒΔ. Η δομή του έχει ως εξής:

<b>INFORMATION_SCHEMA Name</b>	<b>SHOW Name</b>	<b>Remarks</b>
PLUGIN_NAME	<b>Name</b>	MySQL extension
PLUGIN_VERSION		MySQL extension
PLUGIN_STATUS	<b>Status</b>	MySQL extension
PLUGIN_TYPE	<b>Type</b>	MySQL extension
PLUGIN_TYPE_VERSION		MySQL extension
PLUGIN_LIBRARY	<b>Library</b>	MySQL extension
PLUGIN_LIBRARY_VERSION		MySQL extension
PLUGIN_AUTHOR		MySQL extension
PLUGIN_DESCRIPTION		MySQL extension
PLUGIN_LICENSE		MySQL extension

**PROCESSLIST:** Είναι ένας πίνακας που αποθηκεύει πληροφορίες σχετικές με τα threads που τρέχουν στο ΣΔΒΔ. Η δομή του φαίνεται παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
ID	Id	MySQL extension
USER	User	MySQL extension
HOST	Host	MySQL extension
DB	db	MySQL extension
COMMAND	Command	MySQL extension
TIME	Time	MySQL extension
STATE	State	MySQL extension
INFO	Info	MySQL extension

PROFILING: Περιλαμβάνει πληροφορίες σχετικές με την κατάσταση λειτουργίας του ΣΔΒΔ. Η δομή του φαίνεται παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
QUERY_ID	Query_ID	
SEQ		
STATE	Status	
DURATION	Duration	
CPU_USER	CPU_user	
CPU_SYSTEM	CPU_system	
CONTEXT_VOLUNTARY	Context_voluntary	
CONTEXT_INVOLUNTARY	Context_involuntary	
BLOCK_OPS_IN	Block_ops_in	
BLOCK_OPS_OUT	Block_ops_out	
MESSAGES_SENT	Messages_sent	
MESSAGES_RECEIVED	Messages_received	
PAGE_FAULTS_MAJOR	Page_faults_major	
PAGE_FAULTS_MINOR	Page_faults_minor	
SWAPS	Swaps	
SOURCE_FUNCTION	Source_function	
SOURCE_FILE	Source_file	
SOURCE_LINE	Source_line	

REFERENTIAL\_CONSTRAINTS: Πίνακας που περιλαμβάνει πληροφορίες που σχετίζονται με περιορισμούς αναφοράς. Οι πληροφορίες αυτές είναι χρήσιμες κατά την δόμηση των ερωτημάτων από τον χρήστη ειδικότερα όσον αφορά τους περιορισμούς ξένων κλειδιών. Η δομή του πίνακα έχει ως εξής:

INFORMATION_SCHEMA Name	SHOW Name	Remarks
CONSTRAINT_CATALOG		NULL
CONSTRAINT_SCHEMA		
CONSTRAINT_NAME		
UNIQUE_CONSTRAINT_CATALOG		NULL
UNIQUE_CONSTRAINT_SCHEMA		
UNIQUE_CONSTRAINT_NAME		
MATCH_OPTION		
UPDATE_RULE		
DELETE_RULE		
TABLE_NAME		
REFERENCED_TABLE_NAME		

ROUTINES: Ο πίνακας αυτός περιλαμβάνει πληροφορίες για αποθηκευμένες ρουτίνες. Η γραμμογράφηση του φαίνεται στην επόμενη εικόνα.

INFORMATION_SCHEMA Name	mysql.proc Name	Remarks
SPECIFIC_NAME	specific_name	
ROUTINE_CATALOG		NULL
ROUTINE_SCHEMA	db	
ROUTINE_NAME	name	
ROUTINE_TYPE	type	{PROCEDURE FUNCTION}
DTD_IDENTIFIER		data type descriptor
ROUTINE_BODY		SQL
ROUTINE_DEFINITION	body or body_utf8	see Notes
EXTERNAL_NAME		NULL
EXTERNAL_LANGUAGE	language	NULL
PARAMETER_STYLE		SQL
IS_DETERMINISTIC	is_deterministic	
SQL_DATA_ACCESS	sql_data_access	
SQL_PATH		NULL
SECURITY_TYPE	security_type	
CREATED	created	
LAST_ALTERED	modified	
SQL_MODE	sql_mode	MySQL extension
ROUTINE_COMMENT	comment	MySQL extension
DEFINER	definer	MySQL extension
CHARACTER_SET_CLIENT		MySQL extension
COLLATION_CONNECTION		MySQL extension
DATABASE_COLLATION		MySQL extension

SCHEMATA: Ο πίνακας αυτός περιλαμβάνει πληροφορίες για τις βάσεις δεδομένων που περιλαμβάνονται στο ΣΔΒΔ. Είναι από τους πίνακες που χρησιμοποιεί αρκετά η εφαρμογή για την επιλογή των βάσεων δεδομένων στις οποίες θα υποβληθούν τα ερωτήματα. Η δομή του πίνακα φαίνεται παρακάτω.



INFORMATION_SCHEMA Name	SHOW Name	Remarks
CATALOG_NAME		NULL
SCHEMA_NAME	Database	
DEFAULT_CHARACTER_SET_NAME		
DEFAULT_COLLATION_NAME		
SQL_PATH		NULL

SCHEMA\_PRIVILEGES: Ο πίνακας αυτός περιέχει πληροφορίες σχετικά με τα δικαιώματα χρηστών στις βάσεις δεδομένων του ΣΔΒΔ. Η δομή του φαίνεται παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
GRANTEE		'user_name'@'host_name' value, MySQL extension
TABLE_CATALOG		NULL, MySQL extension
TABLE_SCHEMA		MySQL extension
PRIVILEGE_TYPE		MySQL extension
IS_GRANTABLE		MySQL extension

STATISTICS: Πίνακας που περιλαμβάνει πληροφορίες για Indexes. Η γραμμογράφηση του φαίνεται παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
TABLE_CATALOG		NULL
TABLE_SCHEMA		= Database
TABLE_NAME	Table	
NON_UNIQUE	Non_unique	
INDEX_SCHEMA		= Database
INDEX_NAME	Key_name	
SEQ_IN_INDEX	Seq_in_index	
COLUMN_NAME	Column_name	
COLLATION	Collation	
CARDINALITY	Cardinality	
SUB_PART	Sub_part	MySQL extension
PACKED	Packed	MySQL extension
NULLABLE	Null	MySQL extension
INDEX_TYPE	Index_type	MySQL extension
COMMENT	Comment	MySQL extension

TABLES: Πρόκειται για πίνακα που χρησιμοποιείται αρκετά από την εφαρμογή και παρέχει πληροφορίες που αφορούν τους πίνακες των βάσεων δεδομένων του ΣΔΒΔ. Η γραμμογράφηση του πίνακα φαίνεται στην επόμενη εικόνα.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
TABLE_CATALOG		NULL
TABLE_SCHEMA	Table_...	
TABLE_NAME	Table_...	
TABLE_TYPE		
ENGINE	Engine	MySQL extension
VERSION	Version	The version number of the table's .frm file, MySQL extension
ROW_FORMAT	Row_format	MySQL extension
TABLE_ROWS	Rows	MySQL extension
AVG_ROW_LENGTH	Avg_row_length	MySQL extension
DATA_LENGTH	Data_length	MySQL extension
MAX_DATA_LENGTH	Max_data_length	MySQL extension
INDEX_LENGTH	Index_length	MySQL extension
DATA_FREE	Data_free	MySQL extension
AUTO_INCREMENT	Auto_increment	MySQL extension
CREATE_TIME	Create_time	MySQL extension
UPDATE_TIME	Update_time	MySQL extension
CHECK_TIME	Check_time	MySQL extension
TABLE_COLLATION	Collation	MySQL extension
CHECKSUM	Checksum	MySQL extension
CREATE_OPTIONS	Create_options	MySQL extension
TABLE_COMMENT	Comment	MySQL extension

TABLE\_CONSTRAINTS: Πίνακας στον οποίο αποθηκεύονται πληροφορίες σχετικές με περιορισμούς που υποβάλλονται για τους πίνακες. Η δομή του φαίνεται παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
CONSTRAINT_CATALOG		NULL
CONSTRAINT_SCHEMA		
CONSTRAINT_NAME		
TABLE_SCHEMA		
TABLE_NAME		
CONSTRAINT_TYPE		

TABLE\_PRIVILEGES: Πίνακας που περιλαμβάνει πληροφορίες σχετικές με δικαιώματα χρηστών σε πίνακες των βάσεων της ΣΑΒΔ. Η δομή του πίνακα φαίνεται παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
GRANTEE		'user_name'@'host_name' value
TABLE_CATALOG		NULL
TABLE_SCHEMA		
TABLE_NAME		
PRIVILEGE_TYPE		
IS_GRANTABLE		

TRIGGERS: Είναι πίνακας που περιλαμβάνει πληροφορίες για triggers που έχουν δημιουργηθεί. Η γραμμογράφηση του πίνακα φαίνεται στην επόμενη εικόνα.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
TRIGGER_CATALOG		NULL
TRIGGER_SCHEMA		
TRIGGER_NAME	Trigger	
EVENT_MANIPULATION	Event	
EVENT_OBJECT_CATALOG		NULL
EVENT_OBJECT_SCHEMA		
EVENT_OBJECT_TABLE	Table	
ACTION_ORDER		0
ACTION_CONDITION		NULL
ACTION_STATEMENT	Statement	
ACTION_ORIENTATION		ROW
ACTION_TIMING	Timing	
ACTION_REFERENCE_OLD_TABLE		NULL
ACTION_REFERENCE_NEW_TABLE		NULL
ACTION_REFERENCE_OLD_ROW		OLD
ACTION_REFERENCE_NEW_ROW		NEW
CREATED	Created	
SQL_MODE	sql_mode	MySQL extension
DEFINER	Definer	MySQL extension
CHARACTER_SET_CLIENT	character_set_client	MySQL extension
COLLATION_CONNECTION	collation_connection	MySQL extension
DATABASE_COLLATION	Database Collation	MySQL extension

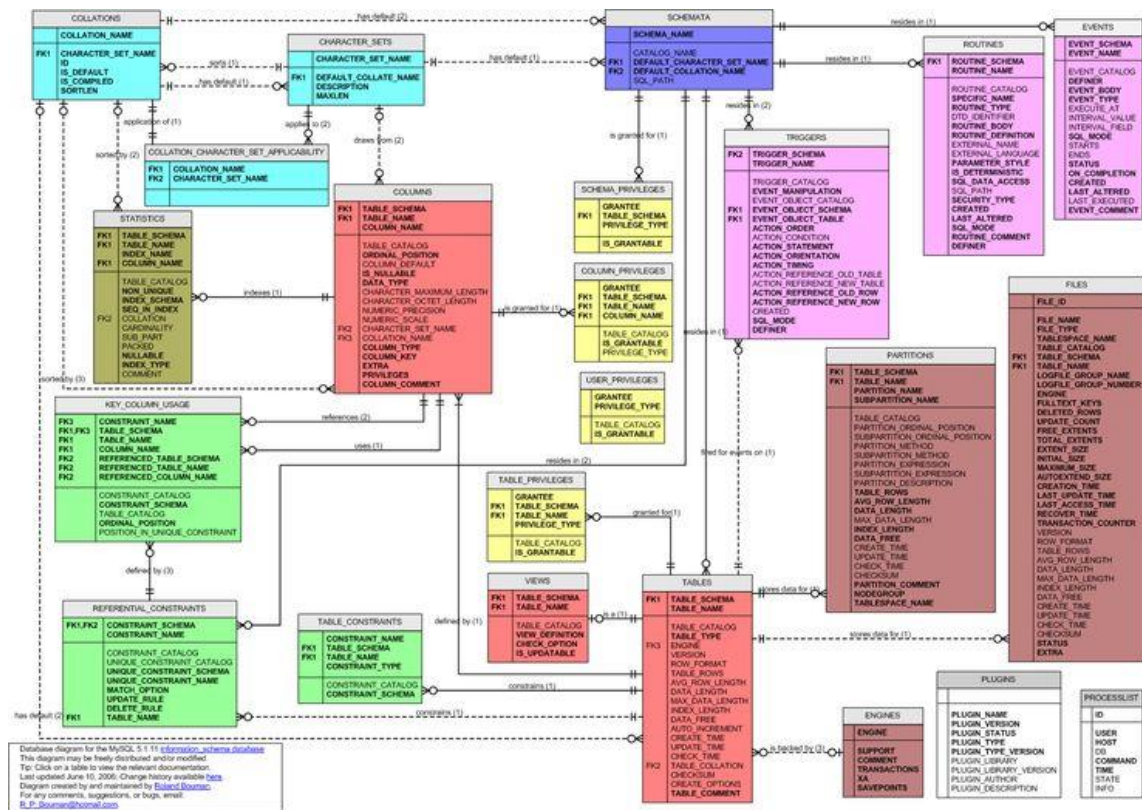
USER\_PRIVILEGES: Πίνακας που περιλαμβάνει τα δικαιώματα των χρηστών στο ΣΔΒΔ. Η γραμμογράφηση του πίνακα φαίνεται στην επόμενη εικόνα.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
GRANTEE		'user_name'@'host_name' value, MySQL extension
TABLE_CATALOG		NULL, MySQL extension
PRIVILEGE_TYPE		MySQL extension
IS_GRANTABLE		MySQL extension

VIEWS: Είναι πίνακας που περιλαμβάνει πληροφορίες για τα VIEWS που έχουν οριστεί στις βάσεις δεδομένων του ΣΔΒΔ. Η δομή του πίνακα φαίνεται παρακάτω.

INFORMATION_SCHEMA Name	SHOW Name	Remarks
TABLE_CATALOG		NULL
TABLE_SCHEMA		
TABLE_NAME		
VIEW_DEFINITION		
CHECK_OPTION		
IS_UPDATABLE		
DEFINER		
SECURITY_TYPE		
CHARACTER_SET_CLIENT		MySQL extension
COLLATION_CONNECTION		MySQL extension

Το σχεσιακό μοντέλο της βάσης δεδομένων φαίνεται στην παρακάτω εικόνα. Σε αυτήν αποτυπώνονται οι πίνακες της καθώς και οι μεταξύ τους σχέσεις (Πρωτεύοντα και Ξένα Κλειδιά).

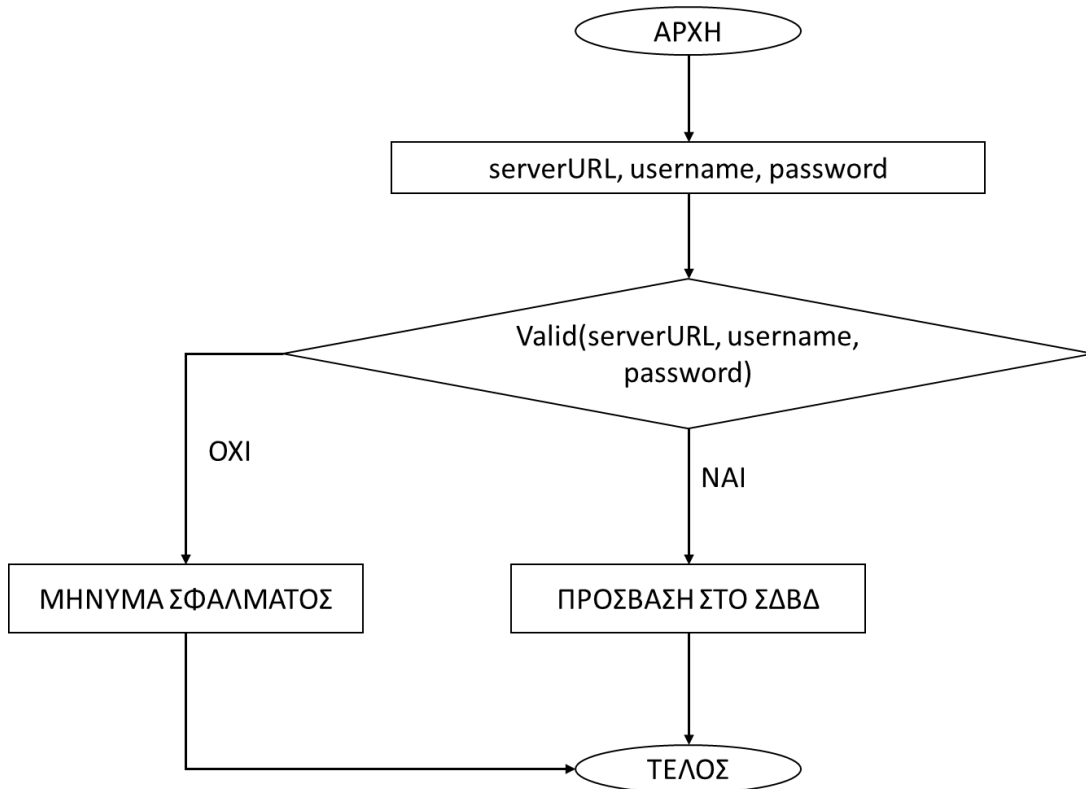


Εικόνα 4: information\_schema

## Business Tier

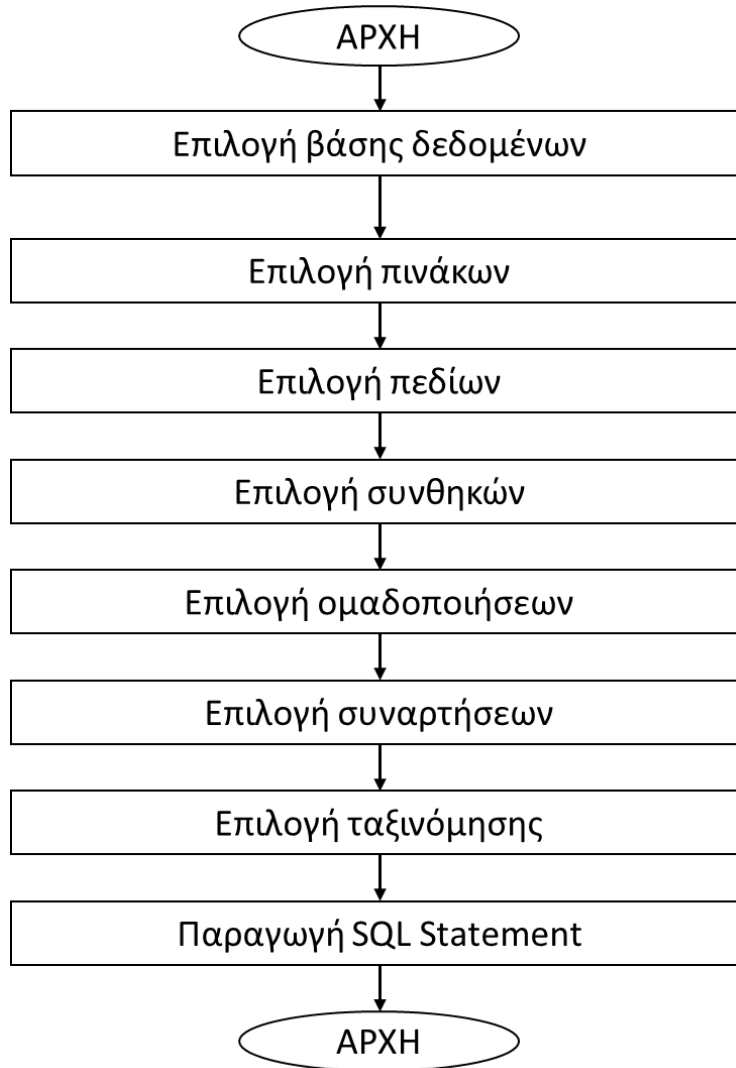
Στο επίπεδο αυτό ορίζονται οι λειτουργίες που θα εκτελεί η εφαρμογή. Αυτές είναι:

- Επιλογή Συστήματος Διαχείρισης βάσης δεδομένων: Κατά την λειτουργία αυτή λαμβάνεται σαν είσοδος η θέση (URL) του ΣΔΒΔ στο οποίο είναι επιθυμητό να αποκατασταθεί μία σύνδεση και τα σχετικά credentials. Αποτέλεσμα της είναι είτε η παραχώρηση δυνατότητας πρόσβασης είτε η άρνηση της. Η λειτουργία παριστάνεται στο επόμενο διάγραμμα ροής.



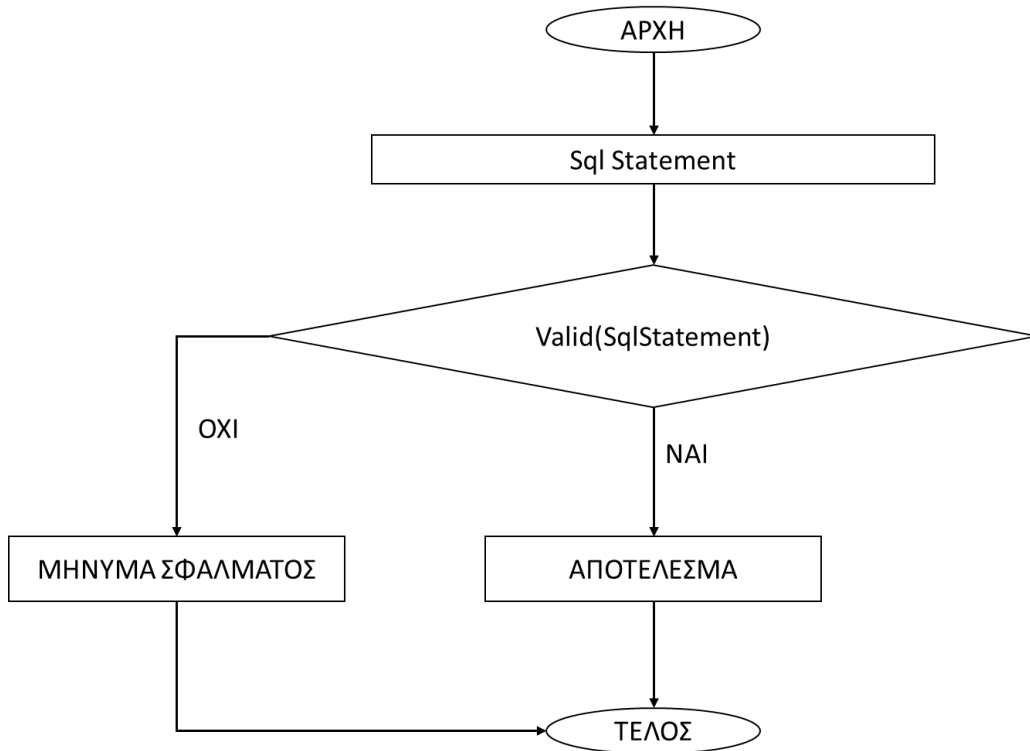
Εικόνα 5: Επιλογή ΣΔΒΔ

- Σύνθεση Ερωτήματος: Με την λειτουργία αυτή συντίθεται το ερώτημα που θα προωθηθεί στην βάση δεδομένων. Οι παράμετροι είναι η βάση δεδομένων που θα εξεταστεί, οι πίνακες που συμμετέχουν, τα πεδία που είναι επιθυμητό να επιστραφούν, πιθανές ομαδοποιήσεις, ταξινομήσεις, συναρτήσεις που θα πρέπει να εφαρμοστούν στα επιστρεφόμενα πεδία και συνθήκες που θα πρέπει να ικανοποιούνται. Η έξοδος της λειτουργίας αυτής είναι ένα SQL Statement. Στο επόμενο διάγραμμα φαίνεται σχηματικά η λειτουργία.



Εικόνα 6: Δημιουργία Ερωτήματος

- Εκτέλεση ερωτήματος και παραγωγή του αποτελέσματος: Στην λειτουργία αυτή ελέγχεται η ορθότητα του ερωτήματος. Αν είναι ορθό τότε επιστρέφεται το αποτέλεσμά του. Σε διαφορετική περίπτωση επιστέφεται κατάλληλο μήνυμα σφάλματος. Η λειτουργία περιγράφεται σχηματικά στο ακόλουθο διάγραμμα ροής.



Εικόνα 7: Εκτέλεση Ερωτήματος

### Presentation Tier

Στο επίπεδο αυτό σχεδιάζονται οι διεπαφές της εφαρμογής. Οι διεπαφές που χρειάζεται να σχεδιαστούν είναι τρεις. Κάθε μία από αυτές υποστηρίζει τις λειτουργίες του παραπάνω επιπέδου. Έτσι σχεδιάστηκαν οι παρακάτω διεπαφές:

- Επιλογή ΣΔΒΔ: Πρόκειται για μία απλή φόρμα με τρία TEXT BOXES που χρησιμοποιούνται για να μπορεί ο χρήστης να καταχωρήσει την τοποθεσία του server και τα credentials που απαιτούνται για την σύνδεση. Αν τα στοιχεία που δίνει είναι σωστά τότε οδηγείται στην οθόνη σχηματισμού του ερωτήματος. Σε διαφορετική περίπτωση ενημερώνεται με κατάλληλο μήνυμα σφάλματος. Το παρακάτω σκαρίφημα περιγράφει την μορφή της διεπαφής.

Host	User	Password
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="CONNECT"/>		

Εικόνα 8: Διεπαφή επιλογής database server

- Σχηματισμός Ερωτήματος: Πρόκειται για τη πιο πολύπλοκη διεπαφή. Σε αυτή ο χρήστης πρέπει να προσδιορίσει ένα αρκετά μεγάλο πλήθος παραμέτρων για τον σχηματισμό του ερωτήματος. Τα στοιχεία αυτά ομαδοποιούνται σε καρτέλες ώστε να μην χρειάζεται να προβαίνει ο χρήστης σε συνεχή scrolling. Σε κάθε καρτέλα ομαδοποιήθηκαν συναφή παράμετροι. Επιπλέον επειδή κάθε ΣΔΒΔ μπορεί να φιλοξενεί πολλές βάσεις δεδομένων, κάθε μία από τις οποίες μπορεί να περιλαμβάνει πολλούς πίνακες με πολλά πεδία και πολλούς περιορισμούς επ’ αυτών επιλέχθηκε οι επιλογές που θα προβάλλονται για επιλογή τιμής σε κάθε παράμετρο να γίνεται μετά την επιλογή σχετικών παραμέτρων που σχετίζονται με αυτή. Για παράδειγμα οι επιλογές των πινάκων γίνεται αφού επιλεγεί η επιθυμητή βάση δεδομένων (από αυτούς που περιλαμβάνονται σε αυτή). Αφού ο χρήστης κάνει όλες του τις επιλογές μπορεί να στείλει το ερώτημα στην βάση δεδομένων για εκτέλεση με κλικ σε πλήκτρο. Επίσης θα χρησιμοποιούνται διαφορετικοί χρωματισμοί για να επισημαίνονται τα πρωτεύοντα και τα ξένα κλειδιά μεταξύ των πεδίων που παρουσιάζονται στην διεπαφή. Το παρακάτω σκαρίφημα καταδεικνύει το πώς σχεδιάστηκε η διεπαφή.

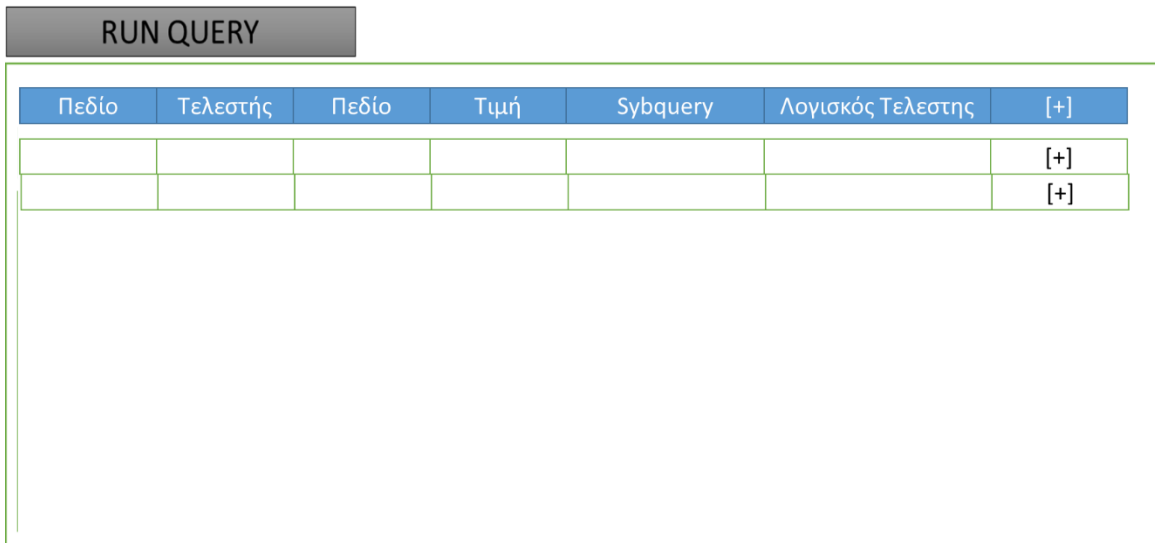
RUN QUERY		
Database	Table	Field

Εικόνα 9: Διεπαφή Δημιουργίας Ερωτήματος (1)

RUN QUERY		
Function	Group By Fields	Field

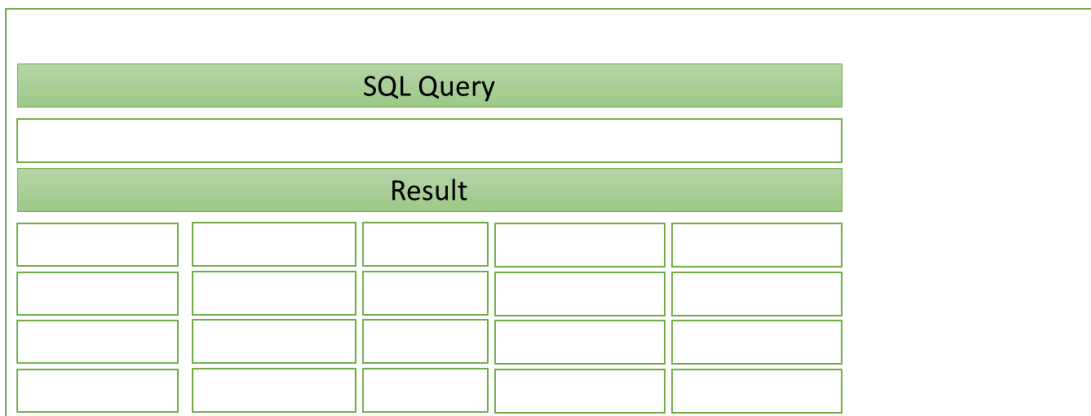
Εικόνα 10: Διεπαφή Δημιουργίας Ερωτήματος (2)





**Εικόνα 11: Διεπαφή δημιουργίας ερωτήματος (3)**

- Παρουσίαση Αποτελέσματος ερωτήματος: Στην διεπαφή αυτή, ανάλογα με την ορθότητα του ερωτήματος προβάλλεται ένα μήνυμα σφάλματος ή ένας πίνακας με τις εγγραφές που ταιριάζουν με τα κριτήρια του ερωτήματος. Παράλληλα εμφανίζεται και το SQL Statement που εκτελέστηκε στην βάση δεδομένων. Στο παρακάτω σκαρίφημα παρουσιάζεται η διεπαφή παρουσίασης του ερωτήματος.



**Εικόνα 12: Διεπαφή παρουσίασης αποτελέσματος ερωτήματος**

## Κεφάλαιο 4: Υλοποίηση της εφαρμογής

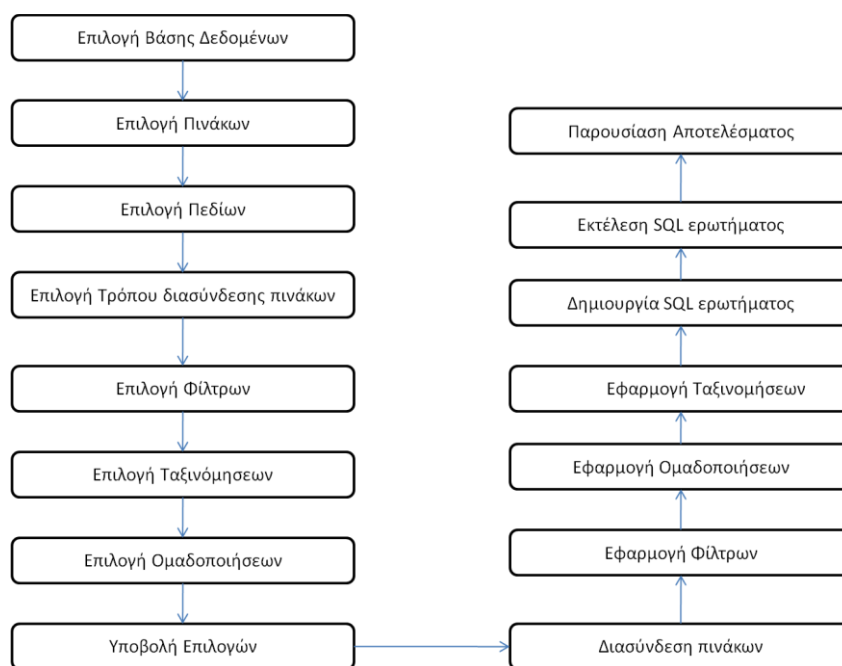
### Λειτουργία της εφαρμογής.

Η λειτουργικότητα της εφαρμογής μπορεί να διαιρεθεί σε επιμέρους λειτουργίες ως εξής:

- Επιλογή Στοιχείων Ερωτήματος
- Επιλογή βάσης δεδομένων: Από την βάση δεδομένων information\_schema λαμβάνονται με κατάλληλο query οι βάσεις δεδομένων που είναι εγκατεστημένες στον database server και προβάλλονται σε combo box από όπου ο χρήστης μπορεί να επιλέξει αυτήν που τον ενδιαφέρει. Με την επιλογή του ενεργοποιείται διαδικασία ανάκτησης των πινάκων της βάσης.
- Επιλογή πινάκων: Αφού ο χρήστης επιλέξει την επιθυμητή βάση δεδομένων, ενεργοποιείται η διαδικασία ανάκτησης των πινάκων που περιέχονται στην επιλεγμένη βάση δεδομένων με κατάλληλο query στην βάση δεδομένων information\_schema. Τα ονόματα των πινάκων καθίστανται διαθέσιμα για επιλογή από τον χρήστη σε combo box με χρήση της τεχνολογίας AJAX. Ο χρήστης έχει την δυνατότητα να επιλέξει πολλαπλούς πίνακες. Κάθε φορά που επιλέγεται ένας πίνακας «τρέχει» η διαδικασία ανάκτησης των πεδίων που περιέχονται στον πίνακα αυτόν.
- Επιλογή πεδίων για προβολή: Κάνοντας την επιλογή των επιθυμητών πινάκων ο χρήστης, μπορεί και βλέπει τα πεδία που έχουν οι επιλεγμένοι πίνακες σε combo box, μετά την εκτέλεση κατάλληλου ερωτήματος στην βάση δεδομένων information\_schema . Τα πεδία είναι σημειωμένα με κόκκινο χρώμα αν είναι πρωτεύοντα κλειδιά, με πράσινο χρώμα αν είναι ξένα κλειδιά και με μαύρο χρώμα τα υπόλοιπα. Όταν ο χρήστης
- Επιλογή τρόπου συμμετοχής πινάκων στο αποτέλεσμα: Με την επιλογή των πινάκων που θα συμμετέχουν στο ερώτημα του χρήστη, σε κατάλληλο σημείο της ιστοσελίδας (με την βοήθεια AJAX) εμφανίζονται checkboxes όπου ο χρήστης καλείται να καθορίσει ποιοι από τους πίνακες θα έχουν ολική συμμετοχή στην συνένωση τους. Επίσης υπάρχει checkbox όπου μπορεί να επιλέξει αν επιθυμεί την παραγωγή του καρτεσιανού γινομένου
- Επιλογή πεδίων για ταξινόμηση: Με την επιλογή των πεδίων που θα εμφανίζονται στο αποτέλεσμα του ερωτήματος, σε καθορισμένο σημείο της ιστοσελίδας εμφανίζονται τα ονόματα των πεδίων και δίπλα τους ένα text box όπου ο χρήστης καταγράφει την σειρά των πεδίων με την οποία θέλει να γίνει η ταξινόμηση του αποτελέσματος.
- Επιλογή πεδίων για ομαδοποίηση: Με την επιλογή των πεδίων που θα εμφανίζονται στο αποτέλεσμα του ερωτήματος, σε καθορισμένο σημείο της ιστοσελίδας εμφανίζονται τα ονόματα των πεδίων και δίπλα τους ένα check box όπου ο χρήστης σημειώνει αν επιθυμεί στο πεδίο αυτό να γίνει ομαδοποίηση του αποτελέσματος.
- Επιλογή Συναρτήσεων Ομαδοποίησης: Με την επιλογή των πεδίων που θα λάβουν μέρος στο αποτέλεσμα δημιουργούνται ζεύγη από combo boxes στα οποία εμφανίζονται τα πεδία των συμμετεχόντων πινάκων και συναρτήσεις ομαδοποίησης όπου ο χρήστης θέλει να εφαρμόσει.
- Επιλογή Φίλτρων Αποτελέσματος: Με μια σειρά από ελεγκτήρια που εμφανίζονται κάθε φορά που ο χρήστης θέλει να εισάγει ένα νέο φίλτρο στο ερώτημα του δίνεται η δυνατότητα να περάσει ο χρήστης τα κριτήρια των φίλτρων του.
- Δημιουργία του Ερωτήματος
- Δημιουργία του Select: Από το αντίστοιχο combo box της φόρμας επιλογής των στοιχείων του ερωτήματος συλλέγονται τα πεδία που θα εμφανίζονται στο αποτέλεσμα.
- Δημιουργία του From: Τοποθετείται ο πίνακας που θα συμμετέχει στο ερώτημα – αν είναι μοναδικός – ή το δημιουργηθώ join αν συμμετέχουν πολλοί πίνακες. Στην περίπτωση που έχει επιλεγεί η παραγωγή καρτεσιανού γινομένου τότε τοποθετούνται οι πίνακες όπως επιλέχθηκαν από τον αντίστοιχο combo box της φόρμας επιλογής στοιχείων ερωτήματος.
- Δημιουργία του Join: Το join δημιουργείται ως εξής:

- Γίνεται έλεγχος αν οι πίνακες που επιλέχθηκαν συνδέονται με ξένα κλειδιά.
  - Για όσους πίνακες συνδέονται με ξένα κλειδιά γίνεται έλεγχος αν είναι επιθυμητή η ολική τους συμμετοχή ώστε να γίνει outer join ή αν δεν είναι επιθυμητή ώστε να γίνει inner join.
- Δημιουργία του Where: Τα επιλεγέντα φίλτρα ελέγχονται και δημιουργείται με βάση αυτά το where του ερωτήματος.
  - Δημιουργία του Group By: Στο group by του ερωτήματος μπαίνουν τα πεδία που έχουν επιλεγεί στο αντίστοιχο combo box.
  - Δημιουργία του Order By: Στο order by του ερωτήματος μπαίνουν τα πεδία που έχουν επιλεγεί στο αντίστοιχο combo box και αφού πρώτα έχουν ταξινομηθεί με την σειρά που έχει σημειώσει ο χρήστης.
  - Προβολή του αποτελέσματος: Το ερώτημα που δημιουργήθηκε αποδίδεται στον database server. Η απόκριση λαμβάνεται από τον database server, μετασχηματίζεται σε html πίνακα και παρουσιάζεται στον τελικό χρήστη. Αν υπάρχει οποιοδήποτε λάθος στο ερώτημα εμφανίζεται κατάλληλο μήνυμα.

Η συνολική λειτουργικότητα της εφαρμογής περιγράφεται σχηματικά παρακάτω



Εικόνα : Λειτουργία της Εφαρμογής

## Data Layer

Η εφαρμογή βασίστηκε σε σχεσιακές βάσεις δεδομένων MySQL. Η MySQL αποτελεί open source λογισμικό, ευρέως διαδεδομένο και πολύ αξιόπιστο. Το μηδενικό κόστος εγκατάστασης, η αξιοπιστία της, η πληρότητα της και η διευρυμένη κοινότητα χρήσης και υποστήριξης της την καθιστούν εκ των κορυφαίων παραγόντων στην αγορά των RDBMS. Υπάρχει πλήθος εκδόσεων του MySQL RDBMS και εργαλείων διαχείριση του. Το πιο ολοκληρωμένο και εύκολα εγκαταστάσιμο είναι αυτό που εμπεριέχεται στο πακέτο xampp και που περιλαμβάνει επίσης τον apache web server, τον php application server και τον filezilla ftp server. Είναι διαθέσιμο για download από <http://www.apachefriends.org/en/xampp.html>. Σε ένα ΣΔΒΔ MySQL πάντα περιλαμβάνεται μία βάση δεδομένων στην οποία αποθηκεύονται όλα εκείνα τα απαραίτητα

στοιχεία για την περιγραφή όλων των υπολοίπων βάσεων δεδομένων. Η βάση δεδομένων αυτή είναι η `information_schema`, η δομή της οποίας περιεγράφηκε στην προηγούμενη ενότητα.

Στην λειτουργία της επιλογής ΣΔΒΔ για την υποβολή ερωτημάτων γίνεται προσπάθεια για σύνδεση σε αυτό με `credentials` που δίνει ο χρήστης. Αν αυτά σχετίζονται επιτυχώς με ήδη αποθηκευμένα στους αντίστοιχούς πίνακες της βάσης δεδομένων `information_schema`. Παράλληλα για την πρόσβαση σε όλες τις βάσεις δεδομένων ελέγχονται τα δικαιώματα του εισερχομένου χρήστη.

Στην λειτουργία της δημιουργίας του ερωτήματος χρειάζεται να γίνεται ή ανάκτηση των διαθέσιμων πινάκων της επιλεγμένης βάσης δεδομένων. Μετά την επιλογή των πινάκων ανακτώνται τα πεδία τους και οι περιορισμοί που έχουν οριστεί σε αυτά. Όλες αυτές οι πληροφορίες ανακτώνται από τους αντίστοιχους πίνακες στην βάση δεδομένων `information_schema`.

Τέλος στην λειτουργία της εκτέλεσης του ερωτήματος και την δημιουργία του αποτελέσματος, το διαμορφωθέν `sql statement` εκτελείται και δημιουργείται ο `result set` από την βάση δεδομένων που έχει επιλεγεί. Σε περίπτωση σφάλματος το ΣΔΒΔ αποκρίνεται με κατάλληλο μήνυμα σφάλματος.

## Business Logic Layer

### Συνδεσιμότητα με την Βάση Δεδομένων (JDBC)

Η διασύνδεση των λειτουργικών μονάδων του επιπέδου Business Logic και της βάσης δεδομένων επιτυγχάνεται με την χρήση του Java Database Connectivity - JDBC. Η τεχνολογία JDBC (μας επιτρέπει να έχουμε πρόσβαση σε μία σχεσιακή βάση δεδομένων μέσα από το πρόγραμμά μας. Η τεχνολογία βασίζεται στην ύπαρξη οδηγών (`drivers`), οι οποίοι είναι αρμόδιοι για την επικοινωνία μεταξύ του προγράμματος και της βάσης. Με τη χρήση του κατάλληλου οδηγού συνδεόμαστε με τη βάση, στέλνουμε εντολές και λαμβάνουμε αποτελέσματα. Οι απαραίτητες κλάσεις ορίζονται από τη βιβλιοθήκη `java.sql`. Για την εφαρμογή χρησιμοποιήθηκε ο `Mysql-JDBC-connector` ο οποίος είναι διαθέσιμος δωρεάν από <http://dev.mysql.com/downloads/connector/j/3.1.html>

### Λειτουργικότητα (JAVA)

Οι βασικές λειτουργίες της υποβολής των `sql` ερωτημάτων και της ανάκτησης των αποκρίσεων από και προς την βάση δεδομένων επιτυγχάνονται από κατάλληλες Java κλάσεις. Η Java είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού που σχεδιάστηκε από την εταιρεία πληροφορικής Sun Microsystems. Χαρακτηριστικό της είναι η ανεξαρτησία του λογισμικού που αναπτύσσεται σε Java από το λειτουργικό σύστημα στο οποίο τρέχουν και γενικότερα το περιβάλλον στο οποίο εντάσσονται. Τα προγράμματα. Αυτό οφείλεται στο γεγονός ότι κάθε Java πρόγραμμα μεταγλωττίζεται μέσω του κατάλληλο προγράμματος, του `javac`, το οποίο παράγει έναν αριθμό από αρχεία (τύπου `.class`) τα οποία ορίζονται σαν είσοδος σε μια εγκατεστημένη Java Virtual Machine και παράγει από αυτά προγράμματα γραμμένα σε γλώσσα μηχανής που υποστηρίζεται από το υποβόσκων λειτουργικό σύστημα και τον επεξεργαστή. Η Java Virtual Machine εξαρτάται από το περιβάλλον ανάπτυξης των εφαρμογών και έτσι κυκλοφορούν διαφορετικές εκδόσεις που χαρακτηρίζονται από την καταλληλότητα χρήσης σε συγκεκριμένα λειτουργικά συστήματα και είδη επεξεργαστών. Αυτός ο τρόπος παραγωγής εκτελέσιμου κώδικα πέρα από την ανεξαρτησία των προγραμμάτων από το περιβάλλον τους εξασφαλίζει και το υπολογιστικό σύστημα στο οποίο τρέχουν τα δημιουργηθέντα προγράμματα από καταστροφικό για αυτό κώδικα. Αυτό συμβαίνει διότι η εικονική μηχανή ελέγχει και αποτρέπει την εκτέλεση του.

Σε γλώσσα `java` αναπτύχθηκαν δύο κλάσεις:

- `DataBase`: Η κλάση αυτή υλοποιεί τις απαραίτητες λειτουργίες για την επικοινωνία μεταξύ των λειτουργιών της εφαρμογής και της βάσης δεδομένων. Στα χαρακτηριστικά της περιλαμβάνονται:
  - `server`: Μεταβλητή που κρατάει την θέση του `database server`

- databaseSchema: Η τρέχουσα βάση δεδομένων.
  - user: Μεταβλητή που κρατάει το username για την σύνδεση στον database server.
  - password: Μεταβλητή που κρατάει το password για την σύνδεση στο database server.
  - driver: Μεταβλητή που κρατάει τον driver που χρειάζεται για την σύνδεση στην βάση δεδομένων.
- Οι μέθοδοι της κλάσης περιλαμβάνουν:
- Constructor: Δημιουργεί ένα αντικείμενο database.
  - setDatabase: Καταχωρεί τιμές στις μεταβλητές – χαρακτηριστικά του αντικειμένου και ελέγχει αν είναι ορθά (επιστρέφει true σε θετική περίπτωση).
  - executeDBQuery: Εκτελεί ένα ερώτημα στην βάση δεδομένων και επιστρέφει το αποτέλεσμα σε html μορφή.
  - executeDBSubQuery: Εκτελεί ένα υποερώτημα στην βάση δεδομένων και επιστρέφει το αποτέλεσμα σε ένα άλλο ερώτημα σε html μορφή.
  - executeSql: Εκτελεί ένα ερώτημα στην βάση δεδομένων Information\_schema και επιστρέφει το αποτέλεσμα σε html μορφή.
  - isStringInArray: Ελέγχει να μία συμβολοσειρά εμπεριέχεται σε έναν πίνακα.
  - joinClause: Δημιουργεί ένα join statement
  - isPrimaryKey: Ελέγχει αν ένα πεδίο είναι πρωτεύον κλειδί.
  - isForeignKey: Ελέγχει αν ένα πεδίο είναι ξένο κλειδί.
- Sorting: Κλάση που χρησιμοποιείται για την ταξινόμηση πίνακα συμβολοσειρών.

## Application Server (JSP – Tomcat)

Η ανίχνευση των ενεργειών του τελικού χρήστη από το επίπεδο παρουσίασης, η μετατροπή τους sql statement, η προώθηση του στις λειτουργίες των Java κλάσεων, η λήψη των αποκρίσεων των Java κλάσεων και η μετατροπή τους σε αναγνώσιμη μορφή είναι αποτέλεσμα των λειτουργιών που εκτελούνται εντός των ιστοσελίδων Java Server Pages και της επεξεργασία τους από τον tomcat application server.

Η γλώσσα προγραμματισμού Java Server Pages (JSP) είναι μία τεχνολογία δυναμικών σελίδων, που παρέχει εργαλεία για την ανάπτυξη προγραμμάτων διακομιστών. Στην JSP είναι ότι ο κώδικας (κώδικας Java) είναι τοποθετημένος μέσα στην ιστοσελίδα κατά τέτοιο τρόπο που να μεταβάλλει δυναμικά το περιεχόμενο της HTML σελίδας. Ο κώδικας Java ενσωματώνεται στον κώδικα HTML με την χρήση των χαρακτήρων <% και %>. Ανάμεσα από τα tags αυτά καταγράφεται και εκτελείται java κώδικας ο οποίος παράγει κώδικα HTML που μπορεί να αναγνωρίσει και να χειριστεί ο web server. Ο τρόπος με τον οποίο ένας JSP- διακομιστής επεξεργάζεται ένα JSP κείμενο είναι να εξάγει τον κώδικα Java, να δημιουργήσει ένα servlet και μετά να επιστρέφει την σελίδα που αντιστοιχεί στον HTML κώδικα συμπληρωμένο με τον HTML κώδικα που παρήγαγε το servlet.

Ο Apache Tomcat είναι ένα σύνολο προγραμμάτων και λειτουργικών μονάδων που επιτρέπει να εξυπηρετούνται WEB εφαρμογές Java Servlets και JSP (Java Server Pages). Διατίθεται δωρεάν από <http://tomcat.apache.org/>.

Στην παρούσα εφαρμογή αναπτύχθηκαν οι εξής jsp σελίδες:

- Index.jsp: Χρησιμοποιείται για την καταχώρηση από τον χρήστη των στοιχείων σύνδεσης στο ΣΔΒΔ.
- Home.jsp: Χρησιμοποιείται για την σύνθεση των ερωτημάτων προς την βάση δεδομένων.

- Query.jsp: Χρησιμοποιείται για την προβολή του αποτελέσματος υποβληθέντος ερωτήματος.
- tableSelection.jsp: Χρησιμοποιείται για την δημιουργία της λίστας των πινάκων που ανήκουν σε μία βάση δεδομένων. Καλείται με ajax request.
- fieldSelection.jsp: Χρησιμοποιείται για την δημιουργία των πεδίων που περιλαμβάνονται σε έναν πίνακα βάσης δεδομένων. Καλείται με ajax request.
- fieldTableSelection.jsp: Χρησιμοποιείται για την δημιουργία των πεδίων που περιλαμβάνονται σε πίνακες βάσης δεδομένων. Καλείται με ajax request.

## Javascript

Η JavaScript είναι γλώσσα προγραμματισμού η οποία έχει σαν σκοπό την παραγωγή δυναμικού περιεχομένου και την εκτέλεση κώδικα στην πλευρά του client (client-side) σε ιστοσελίδες. Η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στον server και πριν την απόκρισή του στην αίτηση του client, αλλά στον browser του χρήστη. Το χαρακτηριστικό αυτό μειώνει την απαίτηση της συνεχούς επικοινωνίας μεταξύ client-server για την ολοκλήρωση απαιτητικών λειτουργιών.

Η γλώσσα JavaScript χρησιμοποιείται στην ανάπτυξη διαδικτυακών εφαρμογών όταν χρειάζεται να εξυπηρετηθούν κάποιοι από τους ακόλουθους σκοπούς:

- Απόδοση του λιγότερου δυνατού φόρτου επεξεργασίας στην μεριά του server: Η κατανομή της επεξεργασίας που απαιτείται για την ολοκλήρωση των διαφόρων λειτουργιών της εφαρμογής στον client και τον server μπορεί να οδηγήσει στην μείωση του απαιτούμενου προς τούτο χρόνου. Αυτό συμβαίνει διότι μειώνονται οι ανάγκες της επικοινωνίας μεταξύ client και server αλλά και διότι μειώνεται και ο όγκος των ανταλλάσσόμενων μεταξύ τους δεδομένων. Συχνά ο έλεγχος και η επικύρωση των δεδομένων που εισάγονται από τους χρήστες γίνεται από τη μεριά του browser κι έτσι φιλτράρονται εκ των προτέρων τα δεδομένα που αποστέλλονται στον server. Επίσης μπορεί και ολόκληρες διαδικασίες να είναι εφικτό να ολοκληρωθούν στην μεριά του client. Γενικά η τάση είναι να επικοινωνεί ο client με τον server μόνο όταν είναι αδύνατον να ολοκληρωθεί η διαδικασία διαφορετικά. Στην περίπτωση αυτή θα πρέπει ο χρήστης να έχει τις δυνατότητες εκτέλεσης javascript σεναρίων ενεργοποιημένη στον browser του.
- Ο χρόνος δημιουργίας της απόκρισης της εφαρμογής στις εισόδους του χρήστη να είναι μικρή: όταν είναι επιθυμητό να ολοκληρώνονται οι λειτουργίες της εφαρμογής γρήγορα (σχεδόν πάντα) τότε επιδιώκεται οι κλήσεις σε λειτουργίες του server να περιορίζονται στις απολύτως απαραίτητες. Με τον τρόπο αυτό περιορίζονται οι περιπτώσεις όπου απαιτείται η αναμονή για την φόρτωση νέων ιστοσελίδων εκ μέρους των χρηστών.
- Αυτόματη διόρθωση λαθών: Είναι εφικτό να αναπτυχθούν σεναρία σε javascript τα οποία να προβαίνουν σε διορθώσεις των δεδομένων που αποστέλλονται για επεξεργασία στον server. Η δυνατότητα αυτή από την μία εξαλείφει τα φαινόμενα των διαδοχικών εσφαλμένων επικοινωνιών μεταξύ client και server αλλά μετατοπίζει και ένα μέρος της λειτουργικότητας της εφαρμογής στον client εξασφαλίζοντας επιπλέον πόρους από τον server.
- Αυξημένη χρηστικότητα των διεπαφών: Με την javascript είναι δυνατόν να προστεθούν μεγαλύτερες δυνατότητες στην χρήση των διεπαφών εκ μέρους των χρηστών σε σχέση με αυτές που παρέχει από μόνη της η HTML. Η HTML – ακόμα και στην έκδοση HTML5 – είναι γενικά φτωχή στο να μπορεί να διαμορφώσει υψηλού επιπέδου διεπαφές οι οποίες να είναι ελκυστικές αλλά και χρηστικές στον τελικό χρήστη. Στο τομέα αυτό η javascript προσφέρει ισχυρές υπηρεσίες για την διαμόρφωση διεπαφών που να δίνουν στον χρήστη να τις χειριστεί αποδοτικά και ευχάριστα.
- Αυξημένη δυνατότητα αλληλεπίδρασης χρήστη και διεπαφής: Η javascript έχει την δυνατότητα να προγραμματίζει τον τρόπο αντίδρασης του browser σε ένα μεγάλο σχετικά πλήθος από συμβάντα (πχ click, double click, mouse over, key press κα). Το

γεγονός αυτό προσθέτει διαδραστικότητα στις διεπαφές που αναπτύσσονται με HTML. Τα στοιχεία των διεπαφών προγραμματίζονται έτσι ώστε να ανιχνεύουν τις ενέργειες του χρήστη και να παρέχουν την αντίδραση εκείνη που αντιστοιχεί σε αυτές.

Η HTML δεν έχει την δυνατότητα από μόνη της να παρέχει σύγχρονες διεπαφές υψηλού επιπέδου και έτσι σχεδόν πάντα συνδυάζεται με την Javascript για το σκοπό αυτό.

## AJAX

Η Ajax είναι ένας από τους πρόσφατους και σημαντικότερους τρόπους βελτίωσης της online εμπειρίας των χρηστών και δημιουργίας νέας και καινοτομικής λειτουργικότητας web επιτρέποντας σε συγκεκριμένα τμήματα μιας ιστοσελίδας να προβάλλονται χωρίς ανανέωση ολόκληρης της σελίδας, βελτιώνει σημαντικά την εμπειρία των εφαρμογών web. Επιτρέπει επίσης στους προγραμματιστές web να δημιουργούν διαισθητικές και καινοτομικές διαδραστικές διαδικασίες. Αυτό επιτυγχάνεται με την κλήση μία server-side λειτουργίας – γραμμένη σε php, asp, jsp - από μία αντίστοιχη client-side – γραμμένη σε javascript και την τοποθέτηση του αποτελέσματος (xml) σε προκαθορισμένο τμήμα της ιστοσελίδας.

Οι τεχνολογίες που περιέχονται στο AJAX είναι οι εξής:

- HTML
- CSS
- JavaScript
- XML και XLST
- XMLHttpRequest Object
- Document Object Model (DOM)

Το αντικείμενο XMLHttpRequest υποστηρίζεται από την JavaScript και προσφέρει όλες τις μεθόδους, χαρακτηριστικά και συμβάντα που είναι απαραίτητα για την ολοκληρωμένη λειτουργία μίας διαδικασίας που στηρίζεται σε μηχανισμούς AJAX. Η λειτουργία του μηχανισμού συνοπτικά έχει ως εξής:

- Μετά από κάποια ενέργεια του τελικού χρήστη δημιουργείται ένα αντικείμενο XMLHttpRequest (1 και 2).
- Το αντικείμενο κατά την διάρκεια της διαδικασίας περνάει από πέντε διαφορετικές καταστάσεις:
  - 0: δεν υπήρξε αρχικοποίηση του αιτήματος
  - 1: η σύνδεση με τον server έχει εδραιωθεί
  - 2: το αίτημα στάλθηκε στον server
  - 3: γίνεται επεξεργασία του αιτήματος
  - 4: η επεξεργασία έχει τελειώσει και το αποτέλεσμα είναι έτοιμο να σταλεί στον browser
- Για κάθε μία από τις καταστάσεις αυτές μπορεί να προγραμματιστούν ανάλογες ενέργειες.
- Τα στοιχεία του XMLHttpRequest μεταβιβάζονται στον WEB SERVER και εκτελείται η λειτουργία η οποία θα επιστρέψει στον CLIENT το ζητούμενο περιεχόμενο (3).
- Το περιεχόμενο λαμβάνεται από τον CLIENT και τοποθετείται στην ιστοσελίδα (4 – 5 και 6).

Στην παρακάτω εικόνα φαίνεται σχηματικά η λειτουργία του μηχανισμού AJAX σε σύγκριση με την κλασική μέθοδο αίτησης περιεχομένου.

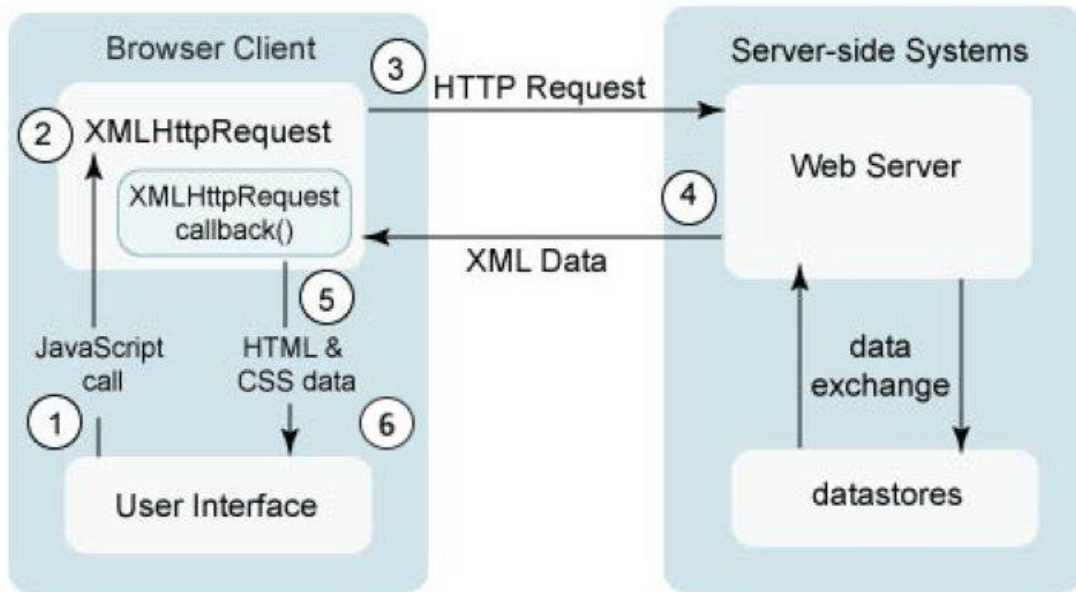


Figure 1: How AJAX Technologies Handle a User Action

Εικόνα 13: Λειτουργία AJAX



## Presentation Layer

### HyperText Markup Language

Η HTML είναι μια XML Based γλώσσα. Αποτελεί υποσύνολο της γλώσσας SGML (Standard Generalized Markup Language) που επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα. Η HTML καθορίζει όχι μόνο το περιεχόμενο που εμφανίζεται στις ιστοσελίδες αλλά και τον τρόπο με τον οποίο αυτό παρουσιάζεται στις ιστοσελίδες. Ορίζει τη δομή και τη θέση των στοιχείων (elements) που αποτελούν μία ιστοσελίδα. Τα στοιχεία αυτά προσδιορίζονται από ετικέτες (tags) της μορφής:

```
<tag_name attribute1="attribute_value" ...>content</tag_name>
```

Οι ετικέτες αυτές παρέχουν πληροφορίες στους browsers σχετικά με το περιεχόμενο, την δομή και την παρουσίαση των ιστοσελίδων. Οι ιδιότητες (attributes) αποδίδουν επιπλέον χαρακτηριστικά στις ετικέτες επεκτείνοντας το σύνολο των πληροφοριών που παρέχουν. Η HTML είναι case insensitive ενώ τα αρχεία που είναι γραμμένα σε html έχουν προέκταση htm ή html.

Η δομή των html αρχείων περιλαμβάνει δύο τμήματα:

- **Επικεφαλίδα (head):** Το τμήμα αυτό οροθετείται από ετικέτες <head>...</head>. Σε αυτό περιλαμβάνονται πληροφορίες για το ίδιο το html έγγραφο. Αυτές οι πληροφορίες (μετα-πληροφορίες) μπορεί να περιλαμβάνουν τον τίτλο της ιστοσελίδας, εξωτερικά αρχεία τα οποία είναι απαραίτητα για την ορθή προβολή της, το charset που χρησιμοποιεί κα. Οι κυριότερες ετικέτες που περιλαμβάνονται στην επικεφαλίδα είναι:
  - <title>: το πιο συχνά χρησιμοποιούμενο και το πλέον απαραίτητο
  - <base>: ορίζει την κύρια τοποθεσία του αρχείου και χρησιμεύει ως αναφορά για όλα τα pathnames και links
  - <link>: ορίζει τη σχέση του παρόντος αρχείου με ένα άλλο αρχείο
  - <meta>: παρέχει πληροφορίες για το αρχείο, όπως λέξεις-κλειδιά
  - <script>: προστίθεται αν θέλουμε να χρησιμοποιήσουμε JavaScript, VBScript
  - <style>: για ενσωμάτωση CSS (Cascading Style Sheets)
- **Κυρίως Σώμα (body):** Σε αυτό το τμήμα περιλαμβάνεται το κυρίως περιεχόμενο της ιστοσελίδας. Είναι το περιεχόμενο αυτό το οποίο εμφανίζεται στην ιστοσελίδα. Οροθετείται από τις ετικέτες <body>...</body>.

Προκειμένου οι browsers να μπορούν να αποδίδουν με ακρίβεια το περιεχόμενο των ιστοσελίδων που προσδιορίζεται με την html καθορίζεται σαφείς προδιαγραφές. Σήμερα οι προδιαγραφές αυτές ορίζονται από την html5 στην οποία έχουν πλέον συμπεριληφθεί στοιχεία που ενισχύουν την διαδραστικότητα των ιστοσελίδων αλλά και στοιχεία που ευνοούν την χρήση των ιστοσελίδων από ανθρώπους με ειδικές ανάγκες. Οι προδιαγραφές αυτές είναι απαραίτητες στο νεοδιαμορφωθέν περιβάλλον του διαδικτύου με την μεγάλη διαθεσιμότητα διαφορετικών browsers αλλά και την χρήση πολλών ειδών συσκευών για την πρόσβαση στις ιστοσελίδες.

### Cascading Style Sheet

Η CSS (Cascading Style Sheets είναι μια αποτελεί μία γλώσσα για την ανάπτυξη στυλ της εμφάνισης εγγραφών που έχουν γραφεί με μία XML based γλώσσα. Συνηθέστερα χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός HTML εγγράφου. Με την CSS καθορίζονται τα χρώματα, οι γραμματοσειρές, η μορφή των συστατικών της ιστοσελίδας. Αποτελεί έναν ολοκληρωμένο μηχανισμό για τον καθορισμό της ομοιόμορφης εμφάνισης των ιστοσελίδων ενός δικτυακού τόπου. Η πρόταση CSS αρχικά ξεκίνησε από



## jQuery

Η jQuery είναι client-side βιβλιοθήκη γραμμένη σε γλώσσα Javascript για την ενσωμάτωση ελκυστικών διαδραστικών χαρακτηριστικών στις ιστοσελίδες. Παρουσιάστηκε από τον John Resig το 2006 και αναπτύχθηκε από την Google. Η δυνατότητα που παρείχε στους προγραμματιστές για εύκολη προσθήκη στοιχείων διαδραστικότητας σε ιστοσελίδες την κατέστησε γρήγορα αρκετά δημοφιλής.

Παρέχει ένα εύχρηστο API που επιτρέπει την επικοινωνία με την DOM, ενεργεί αιτήματα AJAX και δημιουργεί αρκετά εντυπωσιακά εφέ. Τα χαρακτηριστικά της είναι:

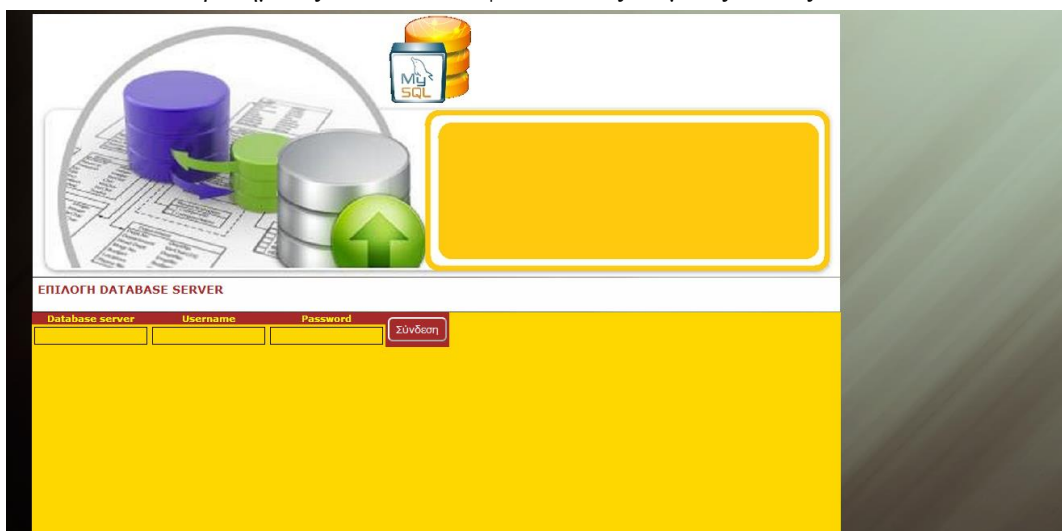
- Χειρίζεται την HTML/A
- Χειρίζεται τα A
- Χειρίζεται συμβάντα της HTML
- Παρουσιάζει effects και animations
- Χειρίζεται τις AJAX αιτήσεις.
- Υποστηρίζει εξειδικευμένες λειτουργίες.

Τα στοιχεία του A χαρακτηρίζονται μοναδικά σε ένα html έγγραφο από το αναγνωριστικό (id) που τους αποδίδεται, όπως επίσης από την κλάση (class) στην οποία ανήκουν ή από το ίδιο tag (π.χ. div, ul, κ.λ.π.). Η βιβλιοθήκη jQuery δίνει την δυνατότητα εύρεσης των διάφορων στοιχείων του A βάσει των ιδιοτήτων τους αλλά και της ιεραρχίας τους μέσα στο έγγραφο. Η διάσχιση του DOM (Traversing) γίνεται μέσω καταλλήλων συναρτήσεων. Η διαχείριση των events ακολουθεί τα πρότυπα του W3C και εφαρμόζονται κατά τον ίδιο τρόπο σε όλους τους γνωστούς και δημοφιλείς browsers. Η διαχείριση τους γίνεται με την χρήση συναρτήσεων και τον ορισμό μέσα σ' αυτές callback συναρτήσεων για την εφαρμογή μιας σειράς βημάτων που επιθυμεί ο εκάστοτε προγραμματιστής να γίνουν όταν πραγματοποιηθεί το event. Αντίστοιχα, υποστηρίζεται η χρήση effects όπως η απόκρυψη ή εμφάνιση στοιχείων του DOM, η ολίσθησή τους ή ακόμα και η εφαρμογή fade in και fade out. Σε όλα τα effects, η ταχύτητα με την οποία εφαρμόζονται είναι ρυθμίσιμη. Παρέχεται και ένας αριθμός συναρτήσεων για τον χειρισμό τιμών του A που αφορούν ένα στοιχείο βάσει του ονόματος του πεδίου αλλά και την μετατροπή τους. Παρέχεται επίσης έκδοση που περιέχει συναρτήσεις για τον χειρισμό animations και visual effects. Τέλος σημαντικό χαρακτηριστικό της χρήσης της βιβλιοθήκης JQuery είναι η αποδοτική προσαρμογή της τεχνολογίας AJAX στους δικτυακούς τόπους.

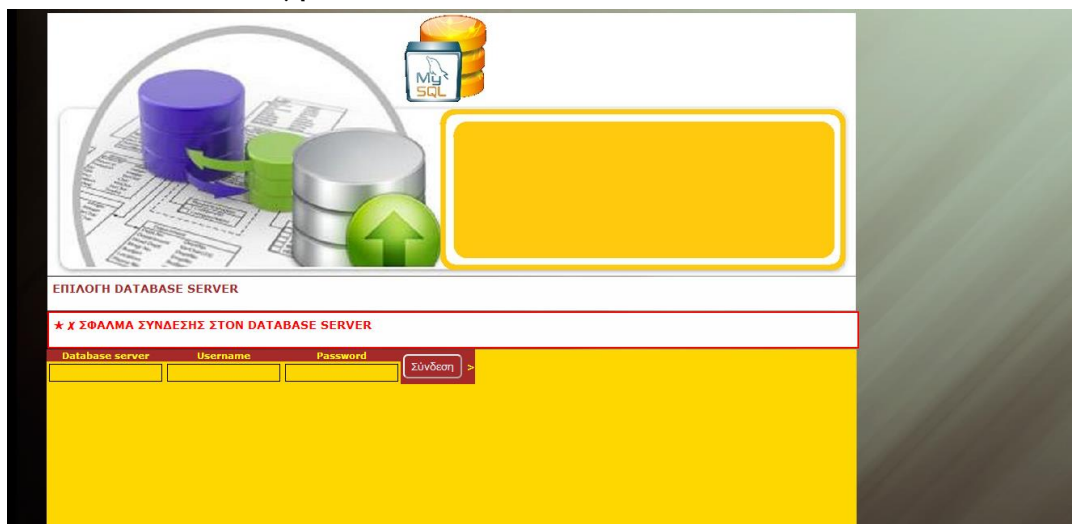
## Χρήση της εφαρμογής

Ο χρήστης της εφαρμογής μπορεί να εκτελέσει τρεις λειτουργίες με τις διεπαφές της εφαρμογής. Οι λειτουργίες αυτές είναι μέρος της διαδικασίας υποβολής ερωτημάτων σε βάσεις δεδομένων που είναι εγκατεστημένες σε ένα MySQL ΣΔΒΔ. Οι λειτουργίες αυτές είναι:

- Σύνδεση σε ΣΔΒΔ: Ο χρήστης της εφαρμογής έχει την δυνατότητα να επιλέξει τον database server στις βάσεις δεδομένων του οποίου θα εκτελέσει τα ερωτήματα. Με την είσοδο του στην εφαρμογή προβάλλεται μία φόρμα στην οποία ο χρήστης καθορίζει την θέση του database server (socket) καθώς και το username και password για είσοδο σε αυτόν. Αν τα στοιχεία που δίνει είναι λανθασμένα εμφανίζεται ένα μήνυμα σφάλματος. Διαφορετικά μεταφέρεται στην επόμενη λειτουργία που είναι η σύνθεση του ερωτήματος. Η διαδικασία φαίνεται στις επόμενες εικόνες.



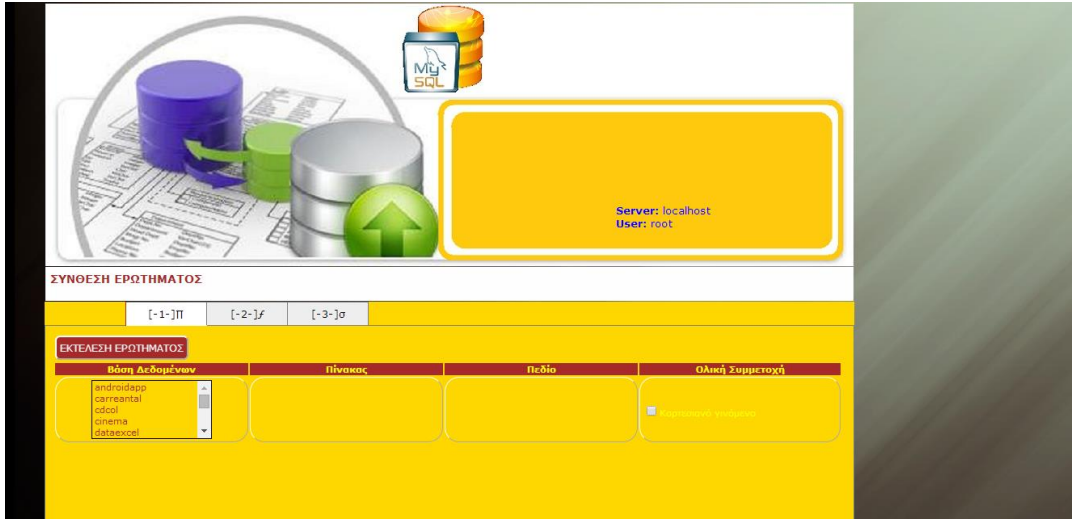
Εικόνα 16: Επιλογή Database Server



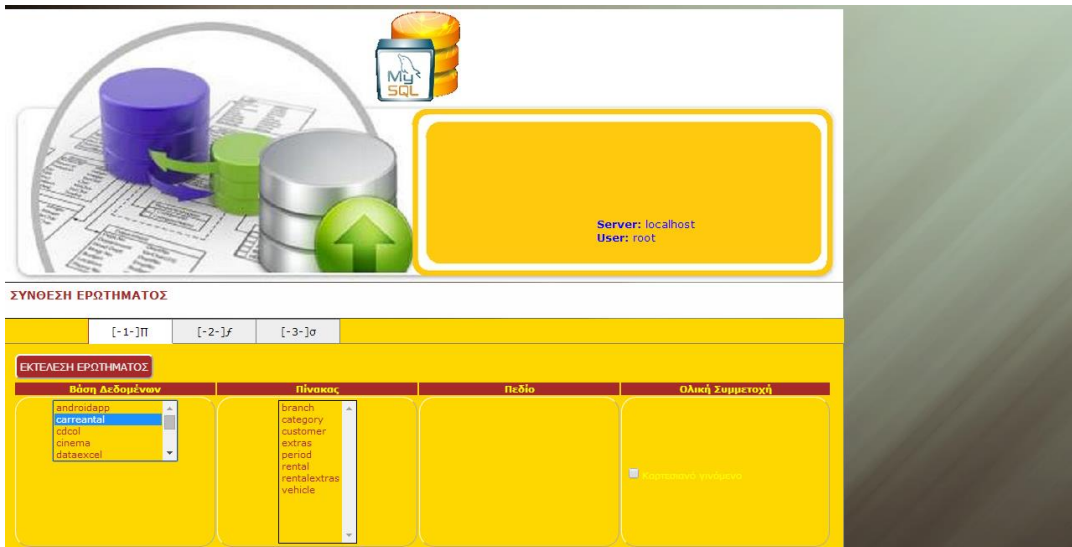
Εικόνα 17: Εσφαλμένη προσπάθεια σύνδεσης στον database server

Σύνθεση του ερωτήματος: Αφού περάσει με επιτυχία την οθόνη σύνδεσης με το ΣΔΒΔ ο χρήστης, οδηγείται στην οθόνη σύνθεσης του ερωτήματος προς την βάση δεδομένων. Η οθόνη αυτή αποτελείται από τρεις καρτέλες στις οποίες δίνεται η δυνατότητα να επιλεγούν τα κριτήρια του ερωτήματος. Αρχικά στην πρώτη καρτέλα πρέπει να επιλέξει σε ποια βάση θα αναζητήσει πληροφορίες. Με την επιλογή του ταυτόχρονα εμφανίζονται προς επιλογή και οι πίνακες που περιέχονται στην επιλεγμένη βάση δεδομένων. Η προβολή των πινάκων γίνεται με χρήση της τεχνολογίας AJAX χωρίς να είναι απαραίτητη η

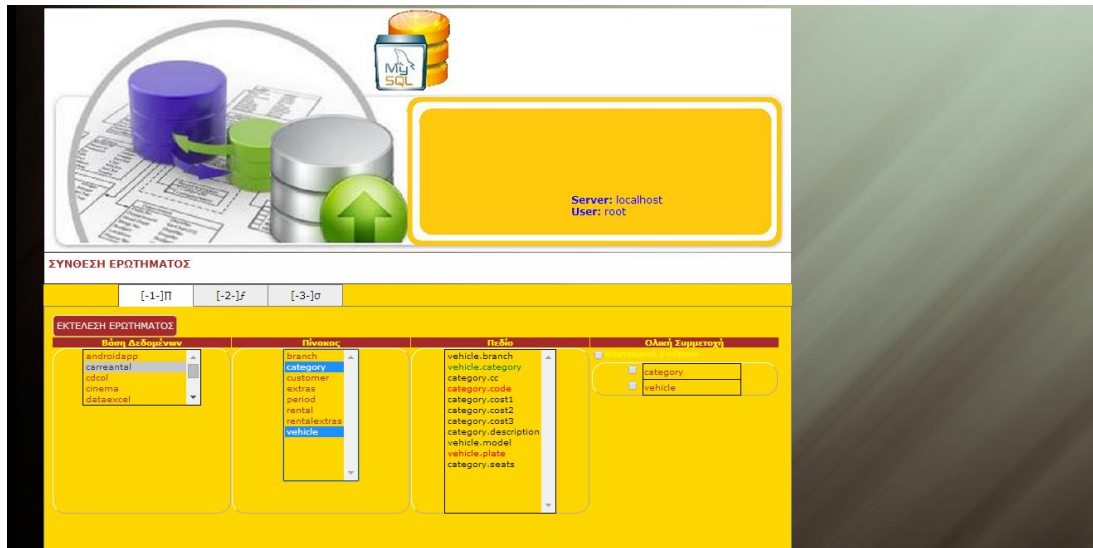
επαναφόρτωση ολόκληρης της σελίδας. Επίσης χρειάζεται να επιλέξει – στην περίπτωση που έχει επιλέξει πολλαπλούς πίνακες – αν επιθυμεί να προβληθεί το καρτεσιανό γινόμενο ή να γίνει join στα ξένα κλειδιά των πινάκων. Στην περίπτωση του join επιλέγει αν κάποιος από τους πίνακες θα έχει ολική συμμετοχή στην παραγωγή του αποτελέσματος. Τα διαθέσιμα προς επιλογή για παρουσίαση πεδία εμφανίζονται μετά την επιλογή των πινάκων με την τεχνολογία AJAX. Με κόκκινο χρώμα σημειώνονται τα πεδία που αποτελούν κύρια κλειδιά ενώ με πράσινο τα πεδία που αποτελούν ξένα κλειδιά. Ο χρήστης έπειτα επιλέγει τα πεδία που θέλει να του παρουσιαστούν στο αποτέλεσμα. Οι επιλογές της πρώτης καρτέλας φαίνονται στις επόμενες εικόνες:



Εικόνα 18: Επιλογή Βάσης Δεδομένων



Εικόνα 19: Επιλογή πίνακα



Εικόνα 20: Επιλογές πεδίων και τύπου join

Στην επόμενη καρτέλα και αφού έχουν γίνει οι επιλογές βάσης δεδομένων, πινάκων και πεδίων, ο χρήστης καλείται να επιλέξει πεδία στα οποία επιθυμεί να πραγματοποιηθεί ομαδοποίηση, συναρτήσεις που ενδεχομένως να επιθυμεί να εφαρμοστούν επί των αποτελεσμάτων καθώς και αν επιθυμεί τα αποτελέσματα να επιστραφούν ταξινομημένα και ως προς ποιο πεδίο. Οι διαθέσιμες επιλογές φαίνονται στην επόμενη εικόνα:

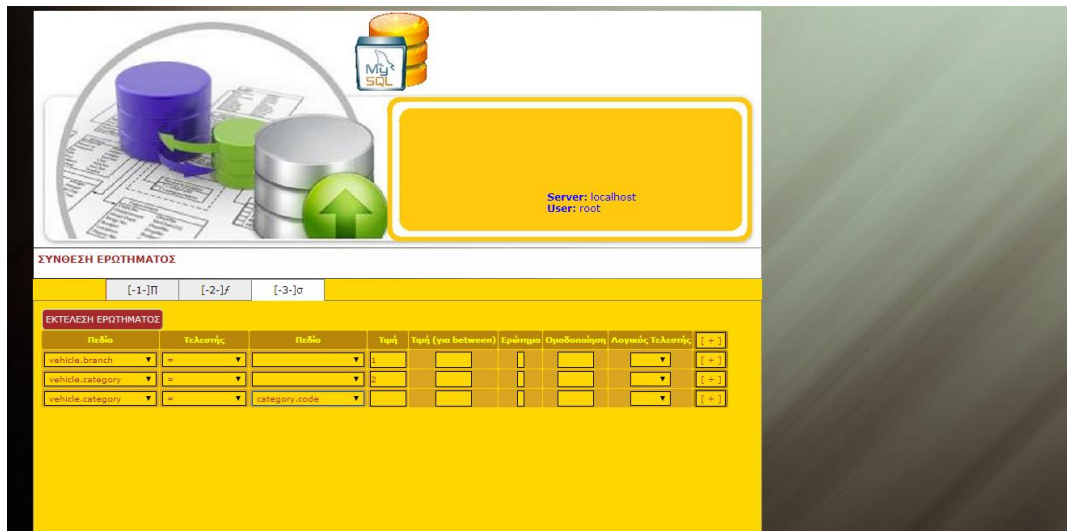


Εικόνα 21: Ομαδοποίηση, συναρτήσεις, ταξινόμηση αποτελέσματος

Στην τρίτη καρτέλα ο χρήστης μπορεί να επιλέξει τις συνθήκες που πρέπει να ικανοποιούν τα αποτελέσματα. Μπορεί να επιλέξει να προστεθούν αρκετές συνθήκες κάνοντας κλικ κάθε φορά στο πλήκτρο [+]. Οι επιλογές που είναι διαθέσιμες στον χρήστη για κάθε συνθήκη είναι:

- Πεδίο: Επιλέγει το πεδίο όπου θα εφαρμοστεί το φίλτρο.
- Τελεστής: Ο τελεστής του φίλτρου.
- Πεδίο: Το δεξί μέρος του τελεστή αν είναι άλλο πεδίο επιλέγεται από εκεί.
- Τιμή: Αν το δεξί μέρος είναι τιμή τότε πληκτρολογείται εκεί.
- Τιμή (για between): Η δεύτερη τιμή αν ο τελεστής είναι between.

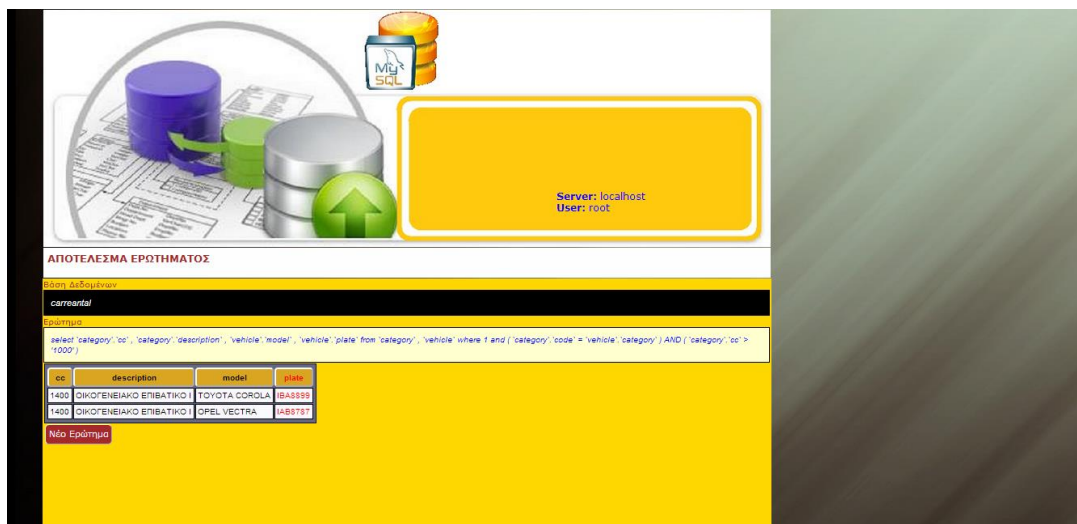
- **Ερώτημα:** Επιλέγεται προκειμένου να παρουσιαστούν οι πλειάδες όπου το πεδίο έχει (ή δεν έχει) τιμές στο σύνολο τιμών που επιστρέφει ένα άλλο ερώτημα. Κάνοντας κλικ ο χρήστης στο πλήκτρο αυτό θα ανοίξει μία νέα οθόνη όπου θα μπορεί να δημιουργήσει αυτό το δευτερεύον ερώτημα.
  - **Ομαδοποίηση:** Παρέχεται δυνατότητα ομαδοποίησης των φίλτρων.
  - **Λογικός Τελεστής:** Εφαρμόζεται λογικός τελεστής ανάμεσα στα φίλτρα.
  - **[+]** : Γίνεται κλικ εκεί προκειμένου να προστεθεί νέο φίλτρο. Η εμφάνιση των ελεγκτηρίων για την προσθήκη του νέου φίλτρου γίνεται με κατάλληλη χρήση εντολών Javascript και css.
- Στην επόμενη εικόνα παρουσιάζεται η καρτέλα καταχώρησης των συνθηκών που πρέπει να ικανοποιούνται από το αποτέλεσμα.



Εικόνα 22: Επιλογή συνθηκών ερωτήματος

Όταν ο χρήστης ολοκληρώσει την καταχώρηση των στοιχείων του ερωτήματος κάνει κλικ στο «ΕΚΤΕΛΕΣΗ ΕΡΩΤΗΜΑΤΟΣ» προκειμένου να εμφανιστεί το αποτέλεσμα του ερωτήματος.

**Προβολή αποτελέσματος:** Στην οθόνη εμφανίζεται η βάση δεδομένων από όπου προέρχονται τα δεδομένα, το sql statement που παράχθηκε από τις επιλογές του χρήστη και τέλος ο πίνακας με το αποτέλεσμα του ερωτήματος. Στο πίνακα αποτελεσμάτων με κόκκινο χρώμα εμφανίζονται τα πρωτεύοντα κλειδιά ενώ τα ξένα κλειδιά με πράσινο χρώμα. Ο χρήστης έχει την δυνατότητα να συνθέσει νέο ερώτημα κάνοντας κλικ στο πλήκτρο «ΝΕΟ ΕΡΩΤΗΜΑ».



Εικόνα 23: Προβολή αποτελέσματος



## Κεφάλαιο 5: Συμπεράσματα

Η σχεδίαση της εφαρμογής πραγματοποιήθηκε ανά επίπεδο της αρχιτεκτονικής της. Αυτό συμβαίνει κάθε φορά που αναπτύσσεται διαδικτυακή εφαρμογή που βασίζεται σε βάσεις δεδομένων. Σε πρώτη φάση σχεδιάστηκε ο τρόπος αποθήκευσης των δεδομένων, δηλαδή η βάση δεδομένων στην οποία είναι οργανωμένα τα δεδομένα που χειρίζεται η εφαρμογή. Στην παρούσα εφαρμογή η βάση δεδομένων αυτή είναι η `information_Schema`. Έτσι δεν χρειάστηκε να αναπτυχθεί μία βάση δεδομένων από την αρχή αλλά έπρεπε να μελετηθεί η δομή και η οργάνωση της `information_schema`. Μέσα από την μελέτη αυτή ο αναλυτής μπορεί να διαπιστώσει τον τρόπο με τον οποίο διατηρούνται όλα εκείνα τα απαραίτητα δεδομένα για την περιγραφή τόσο του ίδιου του ΣΔΒΔ αλλά και των βάσεων που είναι εγκατεστημένες εντός αυτού. Οι πληροφορίες αυτές για την MySQL ήταν αποθηκευμένες με τρόπο τέτοιο που εξασφαλίζει την πλήρη περιγραφή του συστήματος αλλά και των βάσεων δεδομένων. Η `information_schema` είναι μία καλοσχεδιασμένη βάση δεδομένων χωρίς πλεονασμούς και περιττά δεδομένα, με μικρή σχετικά έκταση (μικρό σχετικά πλήθος πινάκων και μικρός όγκος δεδομένων) που έχει την δυνατότητα να διατηρεί όλες τις πληροφορίες περί βάσεων δεδομένων (πίνακες, πεδία, σχέσεις ακεραιότητας, τρόπος φυσικής οργάνωσης αποθήκευσης, χρήστες και δικαιώματα αυτών). Αποτέλεσε εύκολη διαδικασία ο προσδιορισμός της δομής της μέσα από την μελέτη της ώστε στην συνέχεια να είναι εφικτή η ανάπτυξη των λειτουργιών της.

Η σχεδίαση των λειτουργιών προϋπέθετε πρωτίτως το καθορισμό των απαιτήσεων της εφαρμογής. Οι λειτουργικές και μη λειτουργικές απαιτήσεις της εφαρμογής αλλά και οι τύποι χρηστών που πρόκειται να την χρησιμοποιήσουν είναι οι παράμετροι εκείνοι που καθορίζουν το ποιες λειτουργίες πρέπει να αναπτυχθούν και με ποιο τρόπο. Στην παρούσα εφαρμογή ο ορισμός των προϋποθέσεων αυτών ήταν απλός. Οι χρήστες που θα μπορούσαν να χρησιμοποιούσαν την εφαρμογή είναι οι ίδιοι που έχουν δικαίωμα να διαχειρίζονται το υποβόσκειν ΣΔΒΔ. Έτσι δεν χρειάστηκε να γίνει διαχωρισμός των χρηστών σε κατηγορίες αφού αυτό το πραγματοποιεί το ίδιο το ΣΔΒΔ. Αυτό πλέον που χρειάστηκε να καθοριστεί είναι ποιες λειτουργίες θα κάλυπταν τις απαιτήσεις από την εφαρμογή και με ποιόν τρόπο.

Οι διεπαφές της εφαρμογής σχεδιάστηκαν με τρόπο τέτοιο που να διευκολύνουν τον χρήστη στον χειρισμό τους. Αποφεύχθηκε η παράθεση μεγάλου όγκου πληροφοριών στην οθόνη. Όπου χρειάστηκε οι λειτουργίες διασπάστηκαν σε πολλαπλές οθόνες ώστε να είναι ευκολότερα διαχειριστικές από τους χρήστες.

Η ανάπτυξη της εφαρμογής έγινε σταδιακά. Κάθε επίπεδο αναπτύχθηκε ξεχωριστά από το άλλο ακολουθώντας τα όσα σχεδιάστηκαν. Η υλοποίηση ξεκίνησε από το επίπεδο δεδομένων. Σε αυτό - όπως προαναφέρθηκε τοποθετείται μία έτοιμη βάση δεδομένων. Ωστόσο έπρεπε να χρησιμοποιηθεί ο κατάλληλος οδηγός για την επικοινωνία με το επίπεδο της λειτουργικότητας. Εφ' όσον επιλέχθηκε η ανάπτυξη των λειτουργιών με την χρήση Java και jsp αναζητήθηκε ανάλογος οδηγός. Διαπιστώθηκε ότι η MySQL βρίσκεται πλέον σε υψηλό επίπεδο ωριμότητας (γεγονός που πρωτίτως οφείλετε στον επίσης υψηλό βαθμό υιοθέτησης της από την κοινότητα των προγραμματιστών ανάπτυξης εφαρμογών διαδικτύου) και έτσι παρείχε ένα πλήρες σύνολο οδηγιών ικανών να ανταποκριθούν στις ανάγκες κάθε πλατφόρμας λειτουργίας της εφαρμογής. Παράλληλα οι οδηγοί αυτοί παρέχονταν σε εκδόσεις τέτοιες που καθιστούσαν την εγκατάστασή τους απλή διαδικασία αφού δεν απαιτούσαν από τους χρήστες να προβούν σε παραμετροποιήσεις ή άλλες επεμβάσεις στο λογισμικό που εγκαθιστούν.

Στο επίπεδο της λειτουργικότητας όπως προαναφέρθηκε, χρησιμοποιήθηκε η προσέγγιση των δυναμικών ιστοσελίδων για την υλοποίηση των λειτουργιών της εφαρμογής. Για την ανάπτυξη των δυναμικών ιστοσελίδων επιλέχθηκε η λύση της jsp. Ο jsp κώδικας ενσωματώνεται εύκολα στον αντίστοιχο HTML και έτσι ο προγραμματιστής μπορούσε σε απλά βήματα να ενσωματώσει τις λειτουργίες στις διεπαφές της εφαρμογής. Παράλληλα εφαρμόστηκε και η τεχνολογία AJAX, γεγονός που εκτός από την βελτίωση της παρουσίας των λειτουργιών στον τελικό χρήστη (της σύνθεσης του ερωτήματος) βελτίωσε και τους χρόνους απόκρισης της εφαρμογής στις ενέργειες του χρήστη και γενικότερα κατέστησε την εφαρμογή ταχύτερη. Με την AJAX μειώθηκε ο όγκος των διακινουμένων δεδομένων μεταξύ client και server ενώ κάθε φορά μεταβαλλόταν το μέρος εκείνο της διεπαφής που ήταν απαραίτητο και μόνο. Για την ομοιομορφία της ανάπτυξης της εφαρμογής επιλέχθηκε οι λειτουργίες που καλούνται με τον μηχανισμό AJAX να είναι επίσης γραμμένες σε jsp, αν και θα μπορούσε να είναι ανεπτυγμένες σε άλλη γλώσσα προγραμματισμού.

Στο επίπεδο των διεπαφών διαπιστώθηκε ότι η χρήση CSS για την διαμόρφωση της δομής και της εμφάνισης των διεπαφών τα διαχώρισε από το περιεχόμενο. Έτσι είναι εύκολο, αν στο μέλλον παραστεί

ανάγκη, να διαφοροποιηθεί η εμφάνιση των ιστοσελίδων που διαμορφώνουν τις διεπαφές της εφαρμογής χωρίς να είναι απαραίτητη οποιαδήποτε επέμβαση στο περιεχόμενο αυτών. Χρησιμοποιήθηκαν επίσης στοιχεία javascript τα οποία αναβάθμισαν την διαδραστικότητα και την ελκυστικότητα των ιστοσελίδων. Ειδικότερα η βιβλιοθήκη jQuery έδωσε την ευκαιρία στον προγραμματιστή να προσθέσει υψηλού επιπέδου στοιχεία στις διεπαφές που αναβαθμίζουν την εμπειρία του χρήστη με αυτή. Διαπιστώθηκε ότι με την χρήση της jQuery καθίσταται απλή διαδικασία η ενσωμάτωση ευφάνταστων εφέ για την συνοδεία των συμβάντων των χρηστών.

Οι διαδικτυακές εφαρμογές δίνουν την ευκαιρία στους χρήστες τους να χρησιμοποιούν λειτουργίες που είναι εγκατεστημένες σε απόσταση από αυτούς. Οι εφαρμογές δεν είναι εγκατεστημένες στη μεριά του τελικού χρήστη αλλά το γεγονός αυτό δεν εμποδίζει την αποδοτική τους εκτέλεση. Σε αυτό σημαντικό ρόλο παίζει η εξέλιξη που έχει επιτευχθεί τα τελευταία χρόνια στις τεχνολογίες της πληροφορικής και των τηλεπικοινωνιών. Τα δίκτυα και το διαδίκτυο έγιναν ταχύτερα. Η πρόσβαση στο διαδίκτυο έγινε προσιτότερη τόσο λόγω της μείωσης της απαιτούμενης δαπάνης για εξοπλισμό και υπηρεσίες αλλά και λόγω της δυνατότητας πρόσβασης πλέον από περισσότερες συσκευές. Με τις τεχνολογίες του WEB2.0 (μία από τις οποίες είναι και η AJAX) οι διαδικτυακές εφαρμογές αποκτούν μορφή που πλησιάζει αρκετά εκείνη των desktop εφαρμογών. Ο χρήστης δηλαδή όλο και περισσότερο (όσο περισσότερο δηλαδή εφαρμόζονται οι τεχνολογίες του WEB2.0) αποκτά την αίσθηση ότι χειρίζεται desktop εφαρμογές κατά την χρήση διαδικτυακών εφαρμογών.

Στο μέλλον αναμένεται η χρήση των διαδικτυακών εφαρμογών να διατηρήσει την δυναμική της. Η κυρίαρχη τάση είναι η αποκέντρωση των λειτουργιών και η εκτέλεση τους μακριά από τα συστήματα των χρηστών. Άλλωστε όλες οι καινοτόμες τεχνολογίες που προτείνονται την τελευταία δεκαετία είναι προσανατολισμένες στην κατεύθυνση αυτή (cloud computing, semantic web, social networking, user generated content κα). Έτσι όχι μόνο δεν προβλέπεται στο μέλλον να εγκαταλειφθούν το διαδίκτυο ως μέσο λειτουργίας των εφαρμογών αλλά αντιθέτως θεωρείται βέβαιο ότι η έρευνα θα είναι για πολλά χρόνια ακόμα επικεντρωμένη στην βελτίωση τους.

Ένας τρόπος για την βελτίωση τους είναι και η πρόταση αποδοτικότερων τρόπων χειρισμού των δεδομένων. Η διατήρηση αυτών σε βάσεις δεδομένων έχει αρχίσει πλέον να αμφισβητείται. Ήδη έχουν προταθεί τρόποι για την εκμετάλλευση δεδομένων που παρέχονται στο διαδίκτυο χωρίς να απαιτείται η αποθήκευσή τους και η διατήρησή τους σε βάσεις δεδομένων που διαχειρίζονται δεδομένες εφαρμογές (οντολογίες, σημασιολογικός ιστός). Ωστόσο ακόμα και έτσι η διατήρησή τους ως πρωτότυπα προτιμάται να γίνεται σε βάσεις δεδομένων. Μάλιστα η λύση της MySQL, αν και έχουν περάσει πολλά χρόνια από την εμφάνισή της, παραμένει ψηλά στις προτιμήσεις των προγραμματιστών. Η δυναμική της αναμένεται να διατηρηθεί κάτι που καταδεικνύεται και από την απόφαση της ORACLE να επιμεληθεί πλέον την εξέλιξη της.

## Βιβλιογραφία

- [1] Anderson Paul. What is Web 2.0? Ideas, technologies and implications for educations, 2007, <http://www.jisc.ac.uk/media/documents/techwatch/tsw0701b.pdf>
- [2] Καλτσογιάννης Αλέξανδρος, WEB 2.0:Χαρακτηριστικά και επίδρασή του σε επιχειρήσεις, κεντρική διοίκηση και χρήστες, 2007, <http://www.observatory.gr/files/meletes/Web%202.0.pdf>
- [3] Fraternali Piero, Rossi Gustavo, Sánchez-Figueroa Fernando, Rich Internet Applications, 2010, <http://63.84.220.100/csdl/mags/ic/2010/03/mic2010030009.pdf>
- [4] Ostfalia University, AJAX: Rich Internet Applications, 2013, <http://www.upriss.org.uk/wp/w10.pdf>
- [5] Eichorn, Joshua, Understanding Ajax, 2006, Pearson Education
- [6] El.Castro “Εισαγωγή στην HTML με εικόνες” Κλειδάριθμος
- [7] Thomas A.,Powell, Fritz,Schneider, JavaScript, 2001, McGraw-Hill Education - Europe
- [8] Ian Pouncey, Richard York, Beginning CSS Cascading Style Sheets for Web Design, 2011, John Wiley & Sons Inc
- [9] jQuery Cookbook, 2009, O'Reilly Media, Inc, USA
- [10] The Main Features of MySQL. 2013. <http://dev.mysql.com/doc/refman/5.0/en/features.html>
- [11] Oracle, Java Server Pages Technologies, 2010, <http://www.oracle.com/technetwork/java/javaee/jsp/index.html>
- [12] W3Schools, Javascript Tutorial, 2011, <http://www.w3schools.com/js/default.asp>
- [13] W3Schools, CSS Tutorila, 2011,<http://www.w3schools.com/css/>
- [14] W3Schools, HTML Tutorila, 2011,<http://www.w3schools.com/html/>
- [15] Jesse James Garrett, Ajax: A New Approach to Web Applications, 2006, Jesse James Garrett
- [16] MySql, Information schema tables, 2010, <http://dev.mysql.com/doc/refman/5.0/en/information-schema.html>