ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ - WEBINTELLIGENCE

Ανάπτυξη Web-Based Συστήματος Αναζήτησης Αγγελιών Με Χρήση Facets

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΩΝΣΤΑΝΤΙΝΟΥ ΜΗΤΣΑΡΑΚΗ

Επιβλέπων : Μιχάλης Σαλαμπάσης
Καθηγητής, ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Ιούλιος 2021

Η σελίδα αυτή είναι σκόπιμα λευκή.

ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ – WEB INTELLIGENCE

Ανάπτυξη Web-Based Συστήματος Αναζήτησης Αγγελιών Με Χρήση Facets

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΩΝΣΤΑΝΤΙΝΟΥ ΜΗΤΣΑΡΑΚΗ

Επιβλέπων : Μιχάλης Σαλαμπάσης
Καθηγητής ΔΙ.ΠΑ.Ε.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή στις Click here to enter a date..

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Όνομα Επώνυμο Choose an item. ΔΙ.ΠΑ.Ε.	Όνομα Επώνυμο Choose an item. ΔΙ.ΠΑ.Ε.	'Ονομα Επώνυμο Choose an item. ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Ιούλιος 2021

(Υπογραφή)

ΚΩΝΣΤΑΝΤΙΝΟΣ ΜΗΤΣΑΡΑΚΗΣ

.....

ΚΩΝΣΤΑΝΤΙΝΟΣ ΜΗΤΣΑΡΑΚΗΣ

ΜΗΧΑΝΙΚΟΣ ΠΛΗΡΟΦΟΡΙΚΗΣ Α.Τ.ΕΙ.Θ.

Περίληψη

Η αναζήτηση και η ανάκτηση της κατάλληλης πληροφορίας που να ικανοποιεί τις πληροφοριακές ανάγκες των χρηστών αποτελεί αντικείμενο εκτεταμένης έρευνας. Η παρούσα διπλωματική εργασία πραγματεύεται την αναζήτηση πληροφορίας σε ημιδομημένο κείμενο με χρηση facets, μέσα από ένα οργανωμένο τρόπο παρεχόμενων κριτηρίων. Με την κατάλληλη βιβλιογραφική επισκόπηση των σχετιζόμενων με την αναζήτηση πληροφορίας κλάδων, παρουσιάζεται αρχικά το θεωρητικό υπόβαθρο για την υλοποίηση αντίστοιχης εφαρμογής προσανατολισμένης σε αγγελίες. Τα δεδομένα των αγγελιών αυτών περιλαμβάνουν ποικιλία τύπων και μεταδεδομένων, όπως ελεύθερο κείμενο, κατηγορίες/υποκατηγορίες δομών και ιδιότητες (facets). Η υλοποίηση της εφαρμογής βασίστηκε σε μία ολιστική προσέγγιση του θέματος με τη μέθοδο από κάτω προς τα πάνω. Σε αυτό το πλαίσιο, ευρευνήθηκε ο τρόπος καταγραφής, αποθήκευσης και ευρετηρίασης της πληροφορίας με το Elasticsearch, έγινε ο απαραίτητος μετασχηματισμός των δεδομένων, επιλύθηκαν διάφορα τεχνικά προβλήματα για τη βελτιστοποίηση του συστήματος, και τέλος δόθηκε έμφαση στη διεπαφή και στην παρουσίαση της πληροφορίας στον τελικό χρήστη μέσα από τις επιλογές πεδίων, φίλτρων και facets. Η τελική εφαρμογή είναι αυξημένης λειτουργικότητας σε σύγκριση με μία αντίστοιχη παραδοσιακή εφαρμογή και έχει ως αποτέλεσμα την αποτελεσματικότερη διαχείριση και αναζήτηση πληροφορίας.

Λέξεις Κλειδιά: Ανάκτηση Πληροφορίας, Πολύπλερη Αναζήτηση, Ημιδομημένα Δεδομένα, Elasticsearch, Εμπειρία Χρήστη, Ικανοποίηση Χρήστη

Η σελίδα αυτή είναι σκόπιμα λευκή.

Abstract

Searching and retrieving information relevant to the specific information needs of users has been the subject of extensive research. This dissertation studies how to search for information in semi-structured text using facets, based on an organized way of available criteria. After conducting a literature review in scientific domains related to information searching, the essential theoretical background was formulated for the implementation of an information-searching application for advertisements. The data of these advertisements include a variety of types and metadata, such as free text, categories/subcategories of structures and facets. The implementation follows a holistic bottom-up approach. In this context, this dissertation examines the way of registration, storing and indexing information using Elasticsearch, makes the essential data transformation, solves various technical problems aiming at system optimization and finally places emphasis on user interface and the presentation of the final results to the end user through the use of fields, filters and facets. The proposed application has enhanced functionality compared to a traditional searching application and shows a more efficient and effective way of information management and retrieval.

Keywords: Information Retrieval, Faceted Search, Semi-structured Data, Elasticsearch, User Experience, User Satisfaction

Η σελίδα αυτή είναι σκόπιμα λευκή.

Πίνακας περιεχομένων

Ευρετήριο Εικόνων

Ευρετήριο Πινάκων

1

Εισαγωγή

1.1 Αναζήτηση πληροφορίας στο διαδίκτυο

Ο σκοπός της διπλωματικής αυτής εργασίας είναι η παρουσίαση τρόπων που στοχεύουν στη βελτίωση της αναζήτησης πληροφορίας σε ημιδομημένο κείμενο, και πιο συγκεκριμένα στον τομέα των αγγελιών. Το πλήθος των δεδομένων, η ποικιλία των τύπων και των δομών που απαρτίζουν την πληροφορία, η ετερογένεια των διαφορετικών αποδεκτών τιμών που μπορεί αυτή να λάβει, καθώς και η ανακολουθία των διαφορετικών συστημάτων/ιστοσελίδων αναζήτησης, καθιστούν το πεδίο αυτό εξαιρετικά ενδιαφέρον τόσο από ερευνητική όσο και από τεχνολογική σκοπιά.

1.2 Αντικείμενο διπλωματικής

Η εκπόνηση της παρούσας διπλωματικής εργασίας έγινε στο πλαίσιο του Προγράμματος Μεταπτυχιακών Σπουδών “Ευφυείς Τεχνολογίες Διαδικτύου” του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων, του Διεθνούς Πανεπιστημίου Ελλάδος. Ο τίτλος της είναι “Ανάπτυξη web-based συστήματος αναζήτησης αγγελιών με χρήση facets” και καλείται να ευρενήσει το πεδίο αναζήτησης πληροφορίας με χρήση συγκεκριμένων μεθόδων και τεχνικών που άπτονται τόσο την αποθήκευση της πληροφορίας όσο και την παρουσίασή της.

Σκοπός της διπλωματικής είναι η ολιστική προσέγγιση των συστημάτων αναζήτησης από τις διαστάσεις που άπτονται του εν λόγω πεδίου. Πιο συγκεκριμένα, μελετούνται θέματα Ανάκτησης Πληροφορίας (Information Retrieval – IR), Σχεδιασμού Παρουσίασης Πληροφορίας (Information Design – ID), Εμπειρίας Χρήστη (User Experience – UX), Διεπαφής Χρήστη (User Interface – UI) και Ικανοποίησης Χρήστη (User Satisfaction). Επίσης, διερευνάται ο κατάλληλος τρόπος αναζήτησης πληροφορίας με χρήση πολύπλευρης αναζήτησης (faceted search), ώστε σε συνδυασμό με τις υπόλοιπες τεχνικές αναζήτησης να επιτευχθεί ένα περιβάλλον που με εύκολο και διαισθητικό τρόπο να εξυπηρετεί τις πληροφοριακές ανάγκες του τελικού χρήστη.

Πέρα από την έρευνα, έγινε υλοποίηση διαδικτυακής εφαρμογής αναζήτησης αγγελιών, η οποίο περιλαμβάνει πλήθος επιλογών από αυτές που αναφέρονται στη βιβλιογραφική επισκόπηση (φίλτρα, facets, αναζήτηση κειμένου κ.ά.), και ενσωματώνει λειτουργίες καταχώρησης, ανάκτησης και παρουσίασης της πληροφορίας.

1.2.1 Συνεισφορά

Η συνεισφορά της διπλωματικής συνοψίζεται τόσο στη συγκέντρωση και μελέτη του υλικού της αντίστοιχης θεματολογίας όσο και στην παραγωγή λογισμικού. Οι δυνατότητες της εφαρμογής αυτής βασίζονται στη μαζική και στην ανά περίπτωση ευρετηρίαση εγγράφων ελεύθερου κειμένου, στην ανάλυση και στον διαχωρισμό της πληροφορίας ανά τύπο δεδομένων, στην εξεύρεση λύσεων για την ενσωμάτωση μεθόδων ανάκτησης πληροφορίας στα αποθηκευμένα έγγραφα, καθώς και στη δημιουργία κατάλληλης αρχιτεκτονικής που να υποστηρίζει την πολύπλευρη αναζήτηση σε σύνολα δεδομένων.

1.2.2 Σχετικές εργασίες

Κατά τη διάρκεια υλοποίησης της παρούσας διπλωματικής, έγινε εκτενής έρευνα στους τομείς της ανάκτησης πληροφοριών, της παρουσίασης πληροφορίας, της βελτίωσης εμπειρίας χρήστη, καθώς και στην αναζήτηση πληροφορίας με χρήση διαστάσεων (facets) τόσο σε θεωρητικό όσο και σε πρακτικό επίπεδο. Ακολούθως, αναφέρονται συγκεκριμένα πονήματα που θεωρείται ότι είχαν τη μεγαλύτερη και τη σημαντικότερη επιρροή στην υλοποίησης της παρούσας διπλωματικής, χωρίς όμως να παραβλέπεται και η συμβολή των υπόλοιπων πηγών οι οποίες ασφαλώς και αναφέρονται μέσα στο κείμενο της εργασίας.

Πολύ σημαντική βοήθεια στην κατανόηση του προβλήματος παρείχε το πόνημα [ST09] στο οποίο παρουσιάζεται η πολύπλευρη αναζήτηση τόσο από τη σκοπιά των διαστάσεων όσο και από τη σκοπιά του τρόπου χρήσης των αντίστοιχων φίλτρων. Η διδακτορική διατριβή στο [Ka11], παρουσιάζει πολύ σημαντικό υλικό όσον αφορά τη σκοπιά της συνάθροισης των αποτελεσμάτων, ενώ το πόνημα [Pr07] είναι πολύ διαφωτιστικό για την κατανόηση των

αρχών που διέπουν τον σχεδιασμού παρουσίασης της πληροφορίας. Φυσικά, δεν μπορεί να παραληφθεί και το βιβλίο [MRS09] στο οποίο γίνεται εκτενής και σε βάθος ανάλυση του πεδίου της ανάκτησης πληροφορίας, όπως επίσης και το βιβλίο [RT12] στο οποίο αναλύονται οι διάφοροι τρόποι παρουσίασης της πληροφορίας για την επίτευξη του βέλτιστου αποτελέσματος.

Στο πεδίο της εφαρμογής, η έρευνα στόχευσε σε παρόμοια συστήματα που κάνουν χρήση πολύπλευρης αναζήτησης. Η εργασία [Fp13] παρέχει σημαντικές πληροφορίες για την επίλυση συχνών προβλημάτων στην αναζήτηση με χρήση facets. Στο διαδικτυακό έργο [Project-A], παρουσιάζεται ένα παρεμφερές σύστημα ανάκτησης πληροφορίας με χρήση facets για προκαθορισμένες όμως διαστάσεις, ενώ σε επίπεδο κώδικα μελετήθηκε η επέκταση Elasticquent του Laravel [Elasticquent].

1.3 Οργάνωση κειμένου

Στο Κεφάλαιο 2 παρουσιάζεται η σχετική έρευνα για τα πεδία ανάκτησης πληροφορίας, σχεδιασμού παρουσίασης πληροφορίας, εμπειρίας χρήστη, διεπαφής χρήστη, καθώς και ικανοποίησης χρήστη, πεδία δηλαδή που σχετίζονται με την αναζήτηση πληροφορίας.

Στο Κεφάλαιο 3 περιγράφονται τα διαθέσιμα δεδομένα, η προσέγγιση που ακολουθήθηκε για την ανάκτηση της πληροφορίας, καθώς και ο τρόπος παρουσίασής της στον τελικό χρήστη.

Στο Κεφάλαιο 4 περιγράφεται η υλοποίηση των σημαντικότερων σημείων της προτεινόμενης προσέγγισης.

Το Κεφάλαιο 5 περιγράφει τεχνικές λεπτομέρειες για τις πλατφόρμες και τις τεχνολογίες που χρησιμοποιήθηκαν, όπως και τη διαδικασία εγκατάστασης του απαραίτητου λογισμικού.

Τέλος, στο Κεφάλαιο 6, συνοψίζονται τα αποτελέσματα της διπλωματικής εργασίας και προδιαγράφονται πιθανές μελλοντικές επεκτάσεις για τη βελτίωση του συστήματος.

2

Θεωρητικό υπόβαθρο

Στο κεφάλαιο αυτό γίνεται αναφορά στο απαραίτητο θεωρητικό υπόβαθρο πάνω στο οποίο βασίστηκε η διπλωματική και κρίθηκε αναγκαίο να γίνει σχετική μελέτη και έρευνα. Παρουσιάζονται κυρίως θεμελιώδεις έννοιες και βασικά χαρακτηριστικά που έχουν συνάφεια με τη λειτουργία των χαρακτηριστικών της εφαρμογής που υλοποιήθηκε.

2.1 Ανάκτηση Πληροφορίας

Το πλήθος της πληροφορίας που είναι διαθέσιμη είτε σε ιδιωτικά είτε σε δημόσια αποθετήρια είναι πλέον ασύλληπτο [BBK16], [RHG+13]. Τις περισσότερες φορές, το ζητούμενο πλέον δεν είναι η ύπαρξη της πληροφορίας, αλλά το πώς η πληροφορία αυτή θα ανακτηθεί με σωστό, οργανωμένο και επιθεωρήσιμο τρόπο, ώστε να είναι όσο το δυνατόν εγγύτερα στις ανάγκες του χρήστη. Ο κυριότερος στόχος της ανάκτησης της πληροφορίας (Information Retrieval — IR) είναι η εύρεση των αποτελεσμάτων που ικανοποιούν το ερώτημα του χρήστη, όμως επίσης σημαντική είναι και η συνολική ικανοποίηση εμπειρίας του χρήστη (User Experience — UX) που έθεσε το ερώτημα, καθώς δεν είναι αρκετή μόνο η ανάκτηση των αποτελεσμάτων, αλλά η ευκολία στον τρόπο αναζήτησης και η παρουσίασή τους με τρόπο εύκολο, εύληπτο και σε κατανοητή μορφή [MS10].

Η παραπάνω πρόκληση, γίνεται ακόμα μεγαλύτερη, καθώς όχι μόνο η αναζήτηση γίνεται σε μεγάλο πλήθος δεδομένων, αλλά και σε δεδομένα με διαφορετική μορφή και τύπο. Δεδομένα με προκαθορισμένη μορφή και σημασιολογία, όπως τα δεδομένα μίας βάσης δεδομένων είναι

δομημένου τύπου, ενώ δεδομένα που δεν έχουν είτε προκαθορισμένο μοντέλο ή δεν ταιριάζουν σε σχεσιακούς πίνακες είναι αδόμητου τύπου. Στα αδόμητα συνήθως συμπεριλαμβάνονται μηνύματα απλού κειμένου, ιστοσελίδες, μηνύματα ηλεκτρονικού ταχυδρομείου κ.ά. τα οποία δεν έχουν σημασιολογικώς εμφανή δομή. Στην ενδιάμεση κατηγορία κατατάσσονται δεδομένα τα οποία δεν είναι σαφώς ορισμένα αλλά έχουν κάποια δομή και/ή αναγνωριστικά, ώστε να έχουν κάποιου είδους ιεραρχία στα δεδομένα που εμπεριέχουν, όπως JavaScript Object Notation (JSON). Το γεγονός ότι τα δεδομένα αυτά δεν έχουν κάποια προκαθορισμένη δομή τα καθιστά ιδιαιτέρως ευέλικτα και επεκτάσιμα ως προς την κλιμάκωση. Για τον λόγο αυτόν, η ανάκτηση πληροφορίας από μία σχεσιακή βάση δεδομένων όπου τα πεδία είναι σαφώς ορισμένα είναι εντελώς διαφορετική από ότι από ένα σύνολο ιστοσελίδων στο οποίο εμπεριέχεται κείμενο, και μάλιστα όχι μόνο σε ακαθόριστη και μη προσδιορίσιμη εκ των προτέρων δομή, αλλά και σε διαφορετικές φυσικές γλώσσες [RHG+13]. Ο μετασχηματισμός της πληροφορίας γίνεται μέσω της διαδικασίας εξαγωγής πληροφορίας (Information Extraction) όπου το μη δομημένο κείμενο μετασχηματίζεται σε δομημένη πληροφορία, μέσα από τον εντοπισμό συγκεκριμένων τμημάτων δεδομένων τα οποία βρίσκονται σε έγγραφα κειμένου και ακολούθως μέσω της εξαγωγής τους σε ενιαίες ή αλληλοσχετιζόμενες οντότητες [Bj12].

Στο σημείο αυτό, είναι σημαντικό να γίνει η διαφοροποίηση ανάμεσα στον κλάδο της ανάκτησης πληροφορίας και σε αυτόν της εξόρυξης γνώμης (opinion mining), καθώς η εξόρυξη γνώμης πρόκειται για τον συνδυασμό της ανάκτησης πληροφορίας και της γλωσσικής ανάλυσης ενός κειμένου που δημιουργήθηκε από τους ίδιους τους χρήστες (user generated content) με σκοπό τον εντοπισμό της γνώμης του χρήστη/συγγραφέα και την εξαγωγή συμπερασμάτων για τη θετικότητα ή/και αρνητικότητα αυτών [RAIS13].

Εμβαθύνοντας, η ανάκτηση πληροφορίας περιγράφεται ως η διαδικασία εύρεσης περιεχομένου μη δομημένου τύπου από μεγάλες συλλογές εγγράφων και κατά τέτοιον τρόπο που να ικανοποιεί την απαίτηση στόχευσης επιθυμητής πληροφορίας μέσω συγκεκριμένων ερωτημάτων, διαδικασία στην οποία εμπεριέχονται και οι μέθοδοι και οι διαδικασίες περιήγησης, φιλτραρίσματος ή περαιτέρω επεξεργασίας των ανακτηθέντων εγγράφων. Με τον όρο έγγραφο (document) νοείται η μονάδα στην οποία βασίζεται ένα σύστημα ανάκτησης πληροφορίας, ενώ ένα σύνολο εγγράφων απαρτίζουν μία συλλογή (collection). Οι χρήστες αναζητούν πληροφορία μέσα από τα έγγραφα χρησιμοποιώντας ερωτήματα (queries), τα οποία είναι οτιδήποτε μεταφέρει ο χρήστης στο σύστημα αναζήτησης, προκειμένου να επικοινωνήσει τις ανάγκες της απαιτούμενης πληροφορίας [MRS09]. Το σύνολο της διαδικασίας αυτής αναπαριστάται σε τρία επίπεδα [He01]:

- Επίπεδο αναπαράστασης εγγράφων (Document Representation Level). Με τον υπολογισμό των βαρών των όρων της συλλογής, καθορίζεται ο βαθμός συνάφειας με τις εκφραζόμενες έννοιες που διατίθενται στο ευρετήριο.
- Επίπεδο αναπαράστασης ερωτημάτων (Query representation level). Συσχετίζοντας το ερώτημα του χρήστη με τα βάρη των όρων παρέχεται με μεγαλύτερη ακρίβεια η πληροφοριακή ανάγκη για την ανάκτηση των σχετικών εγγράφων.
- Επίπεδο αναπαράστασης αξιολόγησης (Evaluation representation level). Το σύστημα αξιολογεί και διακρίνει τα πιο σχετικά έγγραφα χρησιμοποιώντας τα βάρη συσχετίζοντας τους όρους ερωτήματος και αυτούς των εγγράφων.

Η κατηγοριοποίηση των ερωτηματών γίνεται σε τρεις κατηγορίες, α) τα πληροφοριακά/ενημερωτικά (informational), β) τα πλοήγησης (navigational), γ) τα συναλλαγής (transactional). Με τα πληροφοριακά ερωτήματα γίνεται αναζήτηση για τον εντοπισμό περιεχομένου που σχετίζεται με ένα συγκεκριμένο θέμα προκειμένου να αντιμετωπιστεί μια πληροφοριακή ανάγκη του χρήστη. Τα ερωτήματα πλοήγησης αποσκοπούν στην εύρεση ενός συγκεκριμένου σημείου στο διαδίκτυο (π.χ. ιστοσελίδα) όταν ο χρήστης πιθανόν γνωρίζει ή υποθέτει την ύπαρξη του συγκεκριμένου προορισμού, εξερευνώντας τα αποτελέσματα για το εγγύτερο αποτέλεσμα στην επιθυμητή πληροφορία κατευθυνόμενος σε αυτήν. Τα ερωτήματα συναλλαγής αφορούν την εύρεση πληροφορίας και κατεύθυνση του χρήστη προς μία τοποθεσία με σκοπό την εκτέλεση κάποιας συναλλαγής (π.χ. κράτηση ή διαδικτυακή αγορά) [JBS08]. Παρά τις διάφορες τεχνικές και την εκτεταμένη χρήση τους, παραλείπεται ένα μεγάλο ποσοστό σχετικών αποτελεσμάτων προερχόμενο από ερωτήματα. Αυτό συμβαίνει λόγω του σημασιολογικού κενού ανάμεσα στην πραγματική έννοια που επιθυμούν οι χρήστες και στην αναπαριστώμενη, από τα συστήματα αναζήτησης, πληροφορία, καθώς και λόγω της μικρής αλληλεπιδραστικότητας των συστημάτων [ST09].

Στο πεδίο της ανάκτησης της πληροφορίας υπάρχουν διάφορα θέματα προς διερεύνηση, με τα πιο σημαντικά να είναι ο καθορισμός της συνάφειας/σχετικότητας (relevance) των αποτελεσμάτων σε σχέση με τις απαιτήσεις των χρηστών, η αξιολόγηση (evaluation) των επιστρεφόμενων αποτελεσμάτων, ο τρόπος επικοινωνίας, καθώς και ο καθορισμός των πληροφοριακών αναγκών (information needs) μεταξύ του χρήστη και του συστήματος αναζήτησης [CMS15].

2.1.1 Βασικά στοιχεία αξιολόγησης συστημάτων ανάκτησης πληροφορίας

Οι σημερινές τεχνολογικές εξελίξεις επιτρέπουν τη δημιουργία, τη διάδοση και την αποθήκευση τεράστιας ποσότητας πληροφορίας με εκθετικό τρόπο. Η αξιολόγηση των συστημάτων αναζήτησης γίνεται κυρίως με τη χρήση μετρικών, οι κυριότερες εκ των οποίων είναι η αποτελεσματικότητα (effectiveness) και η αποδοτικότητα (efficiency). Με την

αποτελεσματικότητα μετριέται η ικανότητα του συστήματος αναζήτησης στην εύρεση των σχετικών πληροφοριών, ενώ με την αποδοτικότητα μετριέται το πόσο γρήγορα γίνεται αυτό [BS08]. Οπότε, για ένα δεδομένο ερώτημα με ορισμένο βαθμό συνάφειας, η αποτελεσματικότητα ορίζεται ως το μέτρο του κατά πόσο η κατάταξη που παράγεται από μια μηχανή αναζήτησης αντιστοιχεί στην κατάταξη βάσει της συνάφειας που οι χρήστες θεωρούν ότι έχει. Η αποδοτικότητα ορίζεται με βάση τις απαιτήσεις του χρόνου και του υπολογιστικού χώρου που απαιτεί ο αλγόριθμος για την παραγωγή της κατάταξης [CMS15].

Κατά την ανάκτηση της πληροφορίας, ένα έγγραφο θεωρείται συναφές αν παρέχει την επιθυμητή πληροφορία και ικανοποιεί τον χρήστη. Για τη μέτρηση της αποτελεσματικότητας των συστημάτων αυτών, χρησιμοποιούνται κυρίως δύο μεγέθη [MRS09]:

- Η ακρίβεια (precision), η οποία δηλώνει το ποσοστό των ανακτηθέντων αποτελεσμάτων που είναι σχετικά με τις ανάγκες του χρήστη.

$$\text{Ακρίβεια} = \frac{\#\,(\text{σχετικά αποκτηθέντα έγγραφα})}{\#\,(\text{αποκτηθέντα έγγραφα})} = \text{P}(\text{σχετικά}|\text{αποκτηθέντα})$$

- Η ανάκληση (recall), η οποία δηλώνει το ποσοστό των σχετικών εγγράφων της συλλογής που ανακτήθηκαν.

$$\text{Ανάκληση} = \frac{\#\,(\text{σχετικά αποκτηθέντα έγγραφα})}{\#\,(\text{σχετικά έγγραφα})} = \text{P}(\text{αποκτηθέντα}|\text{σχετικά})$$

Σημαντική έννοια η οποία σχετίζεται με τα παραπάνω, είναι ο βαθμός λεπτομέρειας της ευρετηρίασης των εγγράφων (indexing granularity), όπου επιλέγεται το μέγεθος του κάθε εγγράφου. Για παράδειγμα, ένα βιβλίο μπορεί να ευρετηριοποιηθεί ως ένα ενιαίο έγγραφο ή κάθε κεφάλαιό του να θεωρηθεί ξεχωριστό έγγραφο. Αν ο βαθμός λεπτομέρειας είναι υψηλός, τότε αυτό έχει ως αποτέλεσμα αυξημένα false positives, οπότε και περιορισμένη ακρίβεια, ενώ αν είναι πολύ χαμηλός αυτό συνεπάγεται περιορισμένη ανάκληση [GRG18].

Η αποδοτικότητα των συστημάτων αυτών είναι υψίστης σημασίας, καθώς αυτή καθορίζει τον βαθμό κλιμάκωσής τους σε αυξημένες απαιτήσεις ποσότητας πληροφορίας [BS08]. Επίσης, η αποδοτικότητα έχει σημαντική οικονομική διάσταση στους παρόχους μηχανών αναζήτησης, όπου για κάθε ερώτημα συμμετέχουν πλήθος υπολογιστικών συστημάτων και υποδομών, οπότε και η παραμικρή βελτίωση μπορεί να έχει σημαντικό οικονομικό όφελος στα κόστη των μηχανημάτων και της ενέργειας, βελτιώνοντας ταυτόχρονα το ενεργειακό αποτύπωμα [HMT17]. Μάλιστα, τα προηγούμενα χρόνια το συγκεκριμένο θέμα έχει γίνει αντικείμενο εκτεταμένης έρευνας εστιάζοντας στη βελτίωση της οργάνωσης, της συμπίεσης και της πρόσβασης σε ευρετήρια αναζήτησης, αυξάνοντας σημαντικά την αλγοριθμική αποδοτικότητα και επιτρέποντας έτσι στις μηχανές αναζήτησης να ανταποκρίνονται στις αυξανόμενες ανάγκες των χρηστών [HS12].

Για τη μέτρηση της αποδοτικότητας του συστήματος ανάκτησης πληροφοριών μπορούν να χρησιμοποιηθούν μετρικές όπως [CMS15]:

- Χρόνος ευρετηρίασης (elapsed indexing time): Ο χρόνος που απαιτείται για τη δημιουργία ενός ευρετηρίου σε ένα σύστημα.
- Χρόνος επεξεργασίας ευρετηρίου (indexing processor time): Ο χρόνος που απαιτείται από την κεντρική μονάδας επεξεργασίας για τη δημιουργία ενός ευρετηρίου.
- Διεκπεραιωτικότητα ερωτήματος (query throughput): Ο αριθμός των ερωτημάτων που επεξεργάζονται ανά δευτερόλεπτο, μετρική η οποία είναι και η πιο συχνά χρησιμοποιούμενη για την αποδοτικότητα του συστήματος.
- Καθυστέρηση απόκρισης ερώτηματος (query latency): Ο χρόνος αναμονής του χρήστη έως την απόκριση του συστήματος.
- Ευρετηρίαση σε προσωρινό χώρο (indexing temporary space): Το ποσοστό του προσωρινού χώρου στον δίσκο που χρησιμοποιείται στη δημιουργία ενός ευρετηρίου.
- Μέγεθος ευρετηρίου (index size): Το μέγεθος του χώρου αποθήκευσης που απαιτείται για την αποθήκευση του ευρετηρίου.

2.1.2 Διαστάσεις συστημάτων ανάκτησης πληροφορίας

Τα συστήματα ανάκτησης πληροφορίας μπορούν να διακριθούν σε συγκεκριμένες κατηγορίες βάσει της διάστασης (dimension) στην οποία λειτουργούν, όπως περιεχομένου (content), κλίμακας εφαρμογής (application) και σκοπού (task).

Στην πρώτη κατηγορία η ανάκτηση γίνεται από έγγραφα διαφορετικής δομής και μέσων, όπως κείμενο, φωτογραφίες, βίντεο κ.λπ. [CMS15].

Η διάσταση της κλίμακας εφαρμογής χωρίζεται σε επιμέρους κατηγορίες. Η πρώτη κατηγορία είναι η διαδικτυακή αναζητήση (web search), όπου η αναζήτηση γίνεται σε δισεκατομμύρια έγγραφα που είναι αποθηκευμένα σε εκατομμύρια υπολογιστές. Στον αντίποδα βρίσκεται η αναζήτηση και ανάκτηση προσωπικών πληροφοριών (personal information retrieval / desktop search), όπως τα έγγραφα τα οποία βρίσκονται σε έναν τοπικό υπολογιστή, ενώ στο ενδιάμεσο βρίσκεται η τρίτη κατηγορία που αφορά την επιχειρησιακή αναζήτηση (enterprise search), όπου οι συλλογές μπορεί να είναι εταιρικά έγγραφα, πατέντες, ερευνητικά άρθρα κ.λπ. [MRS09]. Παραδείγματα τέτοιων εφαρμογών είναι η κατηγοριοποίηση κειμένου, η εξαγωγή περιλήψεων, η εξαγωγή πληροφορίας, η ανίχνευση θέματος κ.ά. [BCC10]. Η καθετοποιημένη αναζήτηση (vertical search), εξειδικεύεται σε συγκεκριμένους τομείς και θέματα. Τέλος, υπάρχει και η αναζήτηση μεταξύ ομοτίμων (peer-to-peer search), όπου η αναζήτηση λαμβάνει χώρα σε ένα δίκτυο υπολογιστικών κόμβων χωρίς να υπάρχει κεντροποιημένος έλεγχος (π.χ. εφαρμογές διαμοιρασμού μουσικής).

Η αναζήτηση βάσει σκοπού καθορίζεται από τον λόγο για τον οποίο ο χρήστης εκκινεί τη διαδικασία της αναζήτησης. Η πιο συνηθισμένη είναι αυτή που βασίζεται σε ένα ερώτημα χρήστη (ad hoc search), όπου περιλαμβάνονται και οι αναζητήσεις φιλτραρίσματος (filtering), κατηγοριοποίησης (classification) και απάντησης ερώτησης (question answering). Κατά την αναζήτηση φιλτραρίσματος γίνεται η ανίχνευση συγκεκριμένων εγγράφων τα οποία ικανοποιούν το ενδιαφέρον του χρήστη, κατά την αναζήτηση κατηγοριοποίησης τα αποτελέσματα λαμβάνουν ετικέτες ανά κατηγορία, ενώ στην αναζήτηση απάντησης ερωτημάτων ο χρήστης επιζητεί μία άμεση απάντηση σε μία συγκεκριμένη ερώτηση (π.χ. "Ποια είναι η πρωτεύουσα της Ελλάδας") [CMS15].

Αναζήτηση βάσει περιεχομένου	**Αναζήτηση βάσει κλίμακας εφαρμογής**	**Αναζήτηση βάσει σκοπού**
Κείμενο	Διαδικτυακή αναζήτηση	Ad hoc search
Φωτογραφίες	Αναζήτηση προσωπικών πληροφοριών	Ταξινόμηση
Βίντεο	Επιχειρησιακή αναζήτηση	Φιλτράρισμα
Βίντεο	Καθετοποιημένη αναζήτηση	Απάντηση ερώτησης
Ηχητικά ντοκουμέντα	Αναζήτηση μεταξύ ομοτίμων	

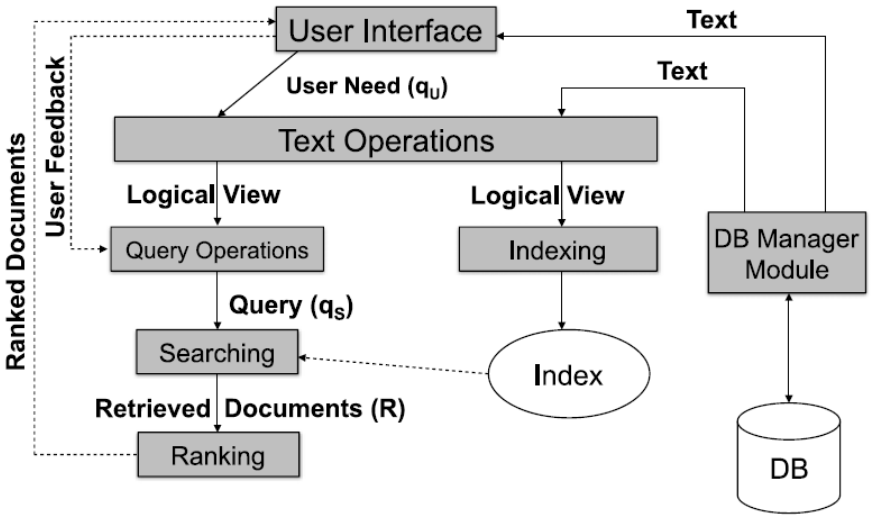
Πίνακας 1: Παραδείγματα κατηγοριών αναζήτησης ανά διάσταση.

Το πιο χαρακτηριστικό παράδειγμα ανάκτησης πληροφορίας είναι οι διαδικτυακές μηχανές αναζήτησης (web search engines) όπως είναι η Google, η Bing κ.ά. Στόχος τους είναι η ανάκτηση εγγράφων από το διαδίκτυο που αντιστοιχούν στις προθέσεις ερωτήματος του χρήστη (κείμενο, εικόνες, βίντεο κ.ά.) [LC14] και η ικανοποίηση των χρηστών από τις στιγμιαίες και στιγμιαία ακριβείς απαντήσεις για τα ερωτήματα που θέτουν [BCC10].

Οι μηχανές αυτές αναζήτησης αποτελούνται από χιλιάδες φυσικά μηχανήματα, λειτουργούν συνεργατικά απαλοίφοντας τα διπλότυτα και τα περιττά αποτελέσματα και παράγουν μία ταξινομημένη λίστα αποτελεσμάτων βάσει βαθμολογίας, η οποία περιλαμβάνει εκτός από τους όρους αναζήτησης, τις περιλήψεις των αποτελεσμάτων, καθώς και τους συνδέσμους οι οποίοι οδηγούν στα αποτελέσματα αυτά. Για την επίτευξη του παραπάνω αποτελέσματος, χρησιμοποιούνται τεχνικές παράλληλης επεξεργασίας, προσωρινής αποθήκευσης (caching) και αντιγραφής (replication), αξιοποιώντας συχνά χρησιμοποιούμενα ή/και κοινά ερωτήματα με στόχο την αποδοτικότερη και αποτελεσματικότερη χρήση των υπολογιστικών πόρων και την ταχύτερη εξυπηρέτηση του τελικού χρήστη [BCC10].

2.1.3 Διαδικασία ανάκτησης πληροφορίας

Η διαδικασία ανάκτησης πληροφορίας (Information Retrieval Process) από έγγραφα που περιέχουν φυσική γλώσσα, παρουσιάζεται στην παρακάτω εικόνα:



Εικόνα 1: Η διαδικασία ανάκτησης πληροφορίας [BBE+13].

Τα βήματα της διαδικασίας είναι ως ακολούθως:

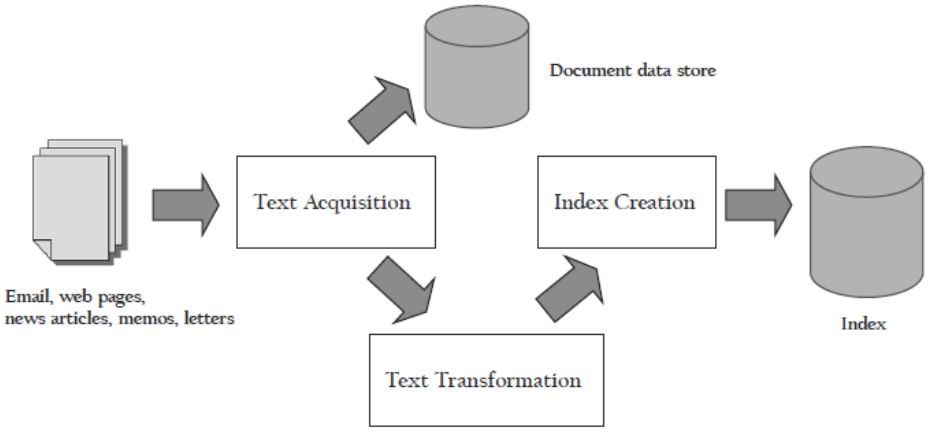
1. Εισάγεται από τον χρήστη η επιθυμητή πληροφορία προς ανάκτηση, με τη μορφή ερωτήματος βασισμένο σε όρους αναζήτησης (q_u), π.χ. λέξεις κλειδιά σε φόρμα.
2. Το ερώτημα q_u αναλύεται και μετασχηματίζεται μέσω ενός συνόλου λειτουργιών που έχουν ως αποτέλεσμα ένα επεξεργασμένο ερώτημα q'_u.
3. Γίνεται περαιτέρω επεξεργασία του ερωτήματος q'_u (q_s), ώστε να αναπαριστάται σε μορφή που να είναι κατανοητή από το σύστημα IR.
4. Ανακτούνται τα σχετικά αποτελέσματα (R) μέσα από ένα σύνολο εγγράφων.
5. Το σύνολο των ανακτηθέντων εγγράφων ταξινομείται βάσει σχετικότητας.
6. Ο χρήστης εξετάζει το σύνολο των ταξινομημένων εγγράφων για την εύρεση της χρήσιμης προς αυτόν πληροφορίας, και βάσει των ενεργειών του, το σύστημα ανατροφοδοτείται σχετικά με τη συνάφεια των αποτελεσμάτων.

Η διαδικασία της αναζήτησης γίνεται σε μία ήδη ευρετηριοποιημένη συλλογή, καθώς θα ήταν αδύνατη η διεκπεραίωση ερωτήματος δυναμικής αναζήτησης (on the fly) τη στιγμή της εισαγωγής, λόγω του μεγέθους της συλλογής, κάτι που γίνεται εύκολα κατανοητό στις περιπτώσεις διαδικτυακής αναζήτησης. Γενικά, η αναζήτηση αποτελείται από δύο επί μέρους διαδικασίες, την ευρετηρίαση (indexing) και τη διαδικασία ερωτήματος (query process).

2.1.4 Διαδικασία ευρετηρίασης

Κατά τη διαδικασία ευρετηρίασης, λαμβάνουν χώρα τρεις διαδικασίες: α) η ανάκτηση του κειμένου (text acquisition), β) ο μετασχηματισμός του κειμένου (text transformation), και γ) η ευρετηρίαση των εγγράφων (indexing) [BBE+13]. Απαραίτητη προϋπόθεση είναι το κείμενο να γίνει διαθέσιμο (text acquisition) είτε στατικά μέσω έτοιμης συλλογής εγγράφων είτε δυναμικά μέσω περιηγητή (crawler), ώστε η ευρετηρίαση να μπορεί να εφαρμοστεί στα

έγγραφα. Κατά τη διαδικασία αυτή, κάθε λέξη/όρος περνάει από τη διαδικασία μετασχηματισμού και προστίθεται σε ένα λεξικό όρων ευρετηρίου (index vocabulary) στο οποίο περιέχονται όλοι οι όροι (index terms) που θεωρούνται χρήσιμοι. Μέσω αυτού, επιτρέπεται η γρήγορη και αποτελεσματικότερη ανάκτηση πληροφορίας. Η αρχιτεκτονική της ευρετηρίασης πρέπει να λαμβάνει υπόψη όχι μόνο το μεγάλο πλήθος εγγράφων κατά τη δημιουργία του ευρετηρίου, αλλά και ότι τα ευρετήρια τροποποιούνται και ανανεώνονται με νέο περιεχόμενο, οπότε και τα ευρετήρια πρέπει να είναι σχεδιασμένα για πολύ γρήγορη επεξεργασία και να επιδέχονται συμπίεσης για ακόμα μεγαλύτερη αποτελεσματικότητα.

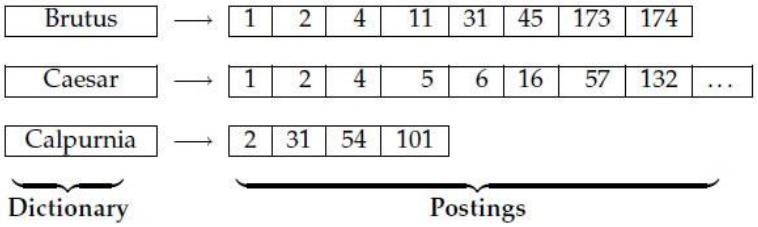


Εικόνα 2: Η διαδικασία ευρετηριοποίησης [CMS15].

Κατά αυτόν τον τρόπο, μόλις ο χρήστης εισάγει τους όρους αναζήτησης (querying) θα γίνει το αντίστοιχο ταίριασμα με τους όρους που υπάρχουν ήδη στο ευρετήριο (query matching) και θα βρεθούν τα έγγραφα στα οποία αντιστοιχούν οι συγκεκριμένοι όροι [CMS15].

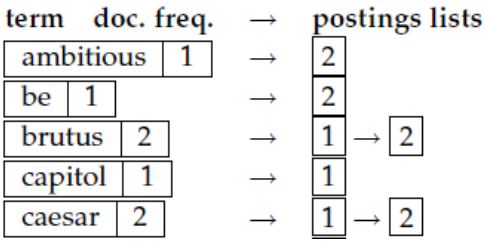
2.1.4.1 Ανεστραμμένο ευρετήριο

Ένας πολύ αποτελεσματικός τρόπος ευρετηρίασης, είναι το ανεστραμμένο ευρετήριο (inverted index), το οποίο αποτελείται από το λεξικό όρων και τη λίστα εγγράφων (postings list ή inverted list). Περιέχει αλφαβητικά ταξινομημένους όλους τους χρήσιμους όρους των εγγράφων της συλλογής, αντιστοιχισμένους με όλα τα έγγραφα στα οποία περιέχονται, και κάθε έγγραφο λαμβάνει έναν σειριακό προσδιοριστικό αριθμό (docID) [MRS09], [HS12].



Εικόνα 3: Λεξικό και λίστα εγγράφων [MRS09].

Οι εμφανίσεις του ίδιου όρου συγχωνεύονται ανά έγγραφο και οι όροι ομαδοποιούνται.



Εικόνα 4: Συχνότητα εμφάνισης όρων και λίστα εγγράφων [MRS09].

Καταγράφονται επιπλέον στατιστικά, όπως η συχνότητα των εγγράφων που περιέχεται κάθε όρος, κάτι που χρησιμοποιείται κατά τη διαδικασία κατάταξης των αποτελεσμάτων.

2.1.4.2 Ευρετήριο θέσης

Οι περισσότερες μηχανές αναζήτησης υποστηρίζουν τη δυνατότητα ερωτημάτων φράσης (phrase queries), η οποία είναι εύκολα κατανοητή από τους χρήστες. Για την παροχή όμως αυτής της λειτουργικότητας δεν αρκούν οι δυνατότητες της απλής λίστας εγγράφων. Μία προσέγγιση για τον χειρισμό των ερωτημάτων αυτών είναι να θεωρούνται όλοι οι διαδοχικοί όροι ενός εγγράφου ως φράση δύο λέξεων (biwords). Στο μοντέλο αυτό (biword index), κάθε φράση θεωρείται ως ένας όρος, ενώ η επεξεργασία μακρύτερων φράσεων γίνεται με τη διαδοχική τους διάσπαση. Για παράδειγμα, το ερώτημα *"διεθνές πανεπιστήμιο ελλάδος θεσσαλονίκη"* διασπάται σε: *"διεθνές πανεπιστήμιο " AND "πανεπιστήμιο ελλάδος" AND "ελλάδος θεσσαλονίκη"*.

Το πρόβλημα σε αυτήν την προσέγγιση είναι η ύπαρξη πολλών false positives, καθώς οι όροι του αρχικού ερωτήματος πιθανόν να βρίσκονται διάσπαρτοι μέσα στο έγγραφο. Η γενική ιδέα του ευρετηρίου δύο λέξεων επεκτείνεται και για μεταλύτερες προτάσεις, δημιουργώντας το ευρετήριο φράσης (phrase index). Οι αναζητήσεις ερωτημάτων ενός όρου συνήθως δεν χειρίζονται από ευρετήρια δύο λέξεων, οπότε απαιτείται η ύπαρξη ενός ευρετηρίου μονών λέξεων. Κάτι τέτοιο μειώνει την πιθανότητα false positive σε ευρετηριοποιημένες φράσεις με πάνω από τρεις όρους, με μειονέκτημα όμως την αύξηση του μεγέθους του ευρετηρίου.

Λόγω των παραπάνω, συνήθως γίνεται χρήση του ευρετηρίου θέσης (positional index). Στην περίπτωση αυτή, ο κάθε όρος αποθηκεύται με τη μορφή docID: (position1, position2, …), όπου κάθε θέση είναι ένας δείκτης στο έγγραφο.

διεθνές, 98.851:
(1, 4: (7, 18, 33, 72);
2, 3: (1, 17, 74))

Στο παραπάνω παράδειγμα, ο όρος "διεθνές" έχει συχνότητα 98.851 και εμφανίζεται 4 φορές στο έγγραφο 1 στις θέσεις 7, 18, 33 και 72. Για την επεξεργασία ενός ερωτήματος φράσης με το μοντέλο αυτό, χρειάζεται και πάλι η πρόσβαση στο ανεστραμμένο ευρετήριο για κάθε όρο ξεχωριστά, ξεκινώντας από τον λιγότερα συχνό και μετά ελέγχεται όχι μόνο η ύπαρξη όλων

των όρων στο έγγραφο αλλά και η εγγύτητά τους. Η ίδια μεθοδολογία χρησιμοποιείται για οποιονδήποτε βαθμό εγγύτητας, κάτι που δεν είναι εφικτό με τα ευρετήρια δύο λέξεων. Μία διάσταση που πρέπει να ληφθεί υπόψη κατά την υιοθέτηση αυτού του μοντέλου είναι η αυξημένη απαίτηση για αποθηκευτικό χώρο για τις λίστες όρων, ενώ παράλληλα αυξάνεται η πολυπλοκότητα της τομής των λιστών όρων, αφού ο αριθμός των προς έλεγχο αντικειμένων δεν περιορίζεται πλέον από τον αριθμό των εγγράφων αλλά από τον συνολικό αριθμό των θέσεων στη συλλογή εγγράφων. Παρόλα αυτά, θεωρείται ότι η λειτουργικότητα αυτή είναι απαραίτητη, καθώς η χρήση αυτών των ερωτημάτων πλέον θεωρείται δεδομένη [MRS09].

Οι δύο παραπάνω τεχνικές ευρετηρίων μπορούν να συνδυαστούν με επιτυχία. Σε περιπτώσεις όπου τα ερωτήματα αποτελούνται από φράσεις που επαναλαμβάνονται συχνά, δεν είναι αποδοτικό να γίνεται συνέχεια η συγχώνευση των λιστών θέσης. Με τον συνδυασμό των δύο προσεγγίσεων χρησιμοποιείται για ορισμένα ερωτήματατα ένα ευρετήριο φράσης ή απλά ένα ευρετήριο δύο λέξεων και για άλλα ερωτήματα χρησιμοποιείται ευρετήριο δεικτών. Ερωτήματα που είναι συχνά χρησιμοποιούμενα σε πρόσφατες αναζητήσεις των χρηστών αποτελούν καλές επιλογές για να συμπεριληφθούν στο ευρετήριο θέσης. Άλλο κριτήριο είναι το υπολογιστικό κόστος των ερωτημάτων, όπου ερωτήματα αποτελούνται από κοινές μεμονωμένες λέξεις αλλά η επιθυμητή φράση προς αναζήτηση είναι σπάνια. Έτσι, η προσθήκη της φράσης “Ελληνικό Πανεπιστήμιο” σε ένα ευρετήριο θέσης θα βελτιώσει την αποτελεσματικότητα σε σχετικά μικρό βαθμό, αφού και οι δύο όροι που απαρτίζουν τη φράση περιέχονται στα περισσότερα έγγραφα ως έγκυρα αποτελέσματα, ενώ η προσθήκη της φράσης “Operation Paperclip” θα έχει ως αποτέλεσμα τη βελτίωση της αποτελεσματικότητας κατά πολύ μεγαλύτερο βαθμό.

Πέρα από τα ερωτήματα αναζήτησης που παρουσιάστηκαν παραπάνω, υπάρχουν και πιο προχωρημένα, όπως αυτά που αναζητούν όρους που κάποιοι χαρακτήρες μπορεί να παίρνουν πολλές τιμές (μπανταρές – wildcard). Η λειτουργικότητα αυτή είναι ιδιαιτέρως χρήσιμη στις περιπτώσεις που οι χρήστες: α) δεν γνωρίζουν επακριβώς τον ορθογραφικό τύπο του όρου, β) όταν επιθυμούν να ανακτήσουν όλες τις παραλλαγές του, γ) όταν οι χρήστες δεν γνωρίζουν αν η μηχανή αναζήτησης χρησιμοποιεί αποκοπή καταλήξεων, και δ) οι χρήστες δεν γνωρίζουν τη σωστή απόδοση ενός ξένου όρου ή φράσης. Μία προσέγγιση για την προσθήκη της λειτουργικότητας αυτής είναι η χρήση Β-δέντρων (B-tree) όπου γίνεται αναζήτηση της αλληλουχία των χαρακτήρων ανά περίπτωση και επιστρέφεται το τμήμα που κάθε φορά χρειάζεται για την εύρεση της απαιτούμενης πληροφορίας [MRS09].

2.1.4.3 Διαδικασία μετασχηματισμού κειμένου

Ιδιαίτερα σημαντική είναι η διαδικασία μετασχηματισμού κειμένου, καθώς εφαρμόζεται τόσο κατά τη δημιουργία του ανεστραμμένου ευρετηρίου όσο και κατά την εισαγωγή των

ερωτημάτων από τον χρήστη. Η διαδικασία αυτή απαρτίζεται από επί μέρους διεργασίες, όπως περιγράφονται ακολούθως:

- Διαχωρισμός όρων (tokenization). Η διεργασία αυτή εξαρτάται από τη φυσική γλώσσα που είναι γραμμένοι οι όροι αναζήτησης, οι οποίοι διαχωρίζονται σε τμήματα (tokens) βάσει κανόνων (διαχωριστικό κενό, απαλοιφή σημείων στίξης κ.λπ.) με τελικό αποτέλεσμα μία σειρά από όρους [MRS09], [CMS15], [BBE+13].
- Απόρριψη κοινών λέξεων (stop word removal). Οι πολύ κοινές λέξεις που εμφανίζονται σχεδόν σε όλα τα έγγραφα και σε μεγάλη συχνότητα, δεν προσδίδουν κάποια αξία ως όροι αναζήτησης και για τον λόγο αυτόν πολλές φορές εξαιρούνται εντελώς από το ευρετήριο. Η χρήση και το μέγεθος αυτών των λιστών εξαρτάται από τη φυσική γλώσσα, καθώς και τη φύση του συστήματος ανάκτησης δεδομένων. Αρκετές μηχανές αναζήτησης δεν τις απορρίπτουν, επειδή αυτό έχει ως αποτέλεσμα τη μείωση της ανάκλησης, λόγω του ότι οι προθέσεις είναι σημαντικές ειδικά στην αποσαφήνιση ερωτημάτων με χρήση φράσεων (phrase queries) [MRS09], [BBE+13].
- Κανονικοποίηση. Η διαδικασία αυτή έχει ως αποτέλεσμα την αντιστοίχιση όρων με άλλους, οι οποίοι έχουν μικρές διαφορές μεταξύ τους (π.χ. ΔΙΠΑΕ και ΔΙ.ΠΑ.Ε.), και την ομαδοποίησή τους σε ισοδύναμες κλάσεις (equivalence classes). Εναλλακτική προσέγγιση είναι οι λίστες συνωνύμων, όπου γίνεται αντιστοίχιση όρων και κάθε ένας μπορεί να ταιριάζει σε πολλές λίστες εγγράφων (π.χ. αυτοκίνητο και αμάξι).
- Μετατροπή σε πεζά/κεφαλαία (capitalization/case-folding). Συχνά, όλοι ή μερικοί από τους χαρακτήρες των όρων αναζήτησης μετατρέπονται σε πεζούς. Συνήθως μετατρέπονται οι αρχικοί χαρακτήρες μίας πρότασης, καθώς αυτές οι λέξεις τις περισσότερες φορές γράφτηκαν με κεφαλαία, απλά έμφαση. Αντίθετα όμως, οι λέξεις γραμμένες με κεφαλαία οι οποίες βρίσκονται στο εσωτερικό των προτάσεων, παραμένουν ως έχουν, καθώς συνήθως γράφονται έτσι σκοπίμως για λόγους σημασιολογικής και εννοιολογικής διαφοροποίησης. Η ανάπτυξη μοντέλων εύρεσης της ακριβής μετατροπής ή διατήρησης των χαρακτήρων ονομάζεται truecasing.
- Αποκοπή καταλήξεων και λημματοποίηση (stemming and lemmatization). Οι διαδικασίες αυτές εξαρτώνται από τη φυσική γλώσσα του κειμένου και είναι πολύ σημαντικές, καθώς ομαδοποιούνται οι λέξεις που βασίζονται σε ένα κοινό θέμα και ελαχιστοποιούνται οι διαφορετικές μορφές και τα παράγωγα των λέξεων (π.χ. βιβλίο, βιβλίων → βιβλίο). Με την αποκοπή των καταλήξεων απαλοίφονται οι καταλήξεις (π.χ. βουνό, βουνίσιος → "βουν") διατηρώντας μόνο το θέμα, προκειμένου στην απεικόνιση των όρων να χρησιμοποιείται μία μόνο λέξη, ανεξαρτήτως πτώσεων, εγκλίσεων κ.ά. Με τη λημματοποίηση αφαιρούνται οι παράγωγες καταλήξεις, έτσι ώστε να διατηρηθεί το "λήμμα" της λέξης το οποίο βασίζεται σε κάποιο λεξικό.

- Στάθμιση βαρύτητας όρων (term weighting). Πρόκειται για την τελική φάση της επεξεργασίας κειμένου και αντικατοπτρίζει τη σημαντικότητα κάθε όρου στα έγγραφα. Χρησιμοποιείται για τον υπολογισμό της βαθμολογίας κατάταξης (ranking score), με χρήση στατιστικών εμφάνισης των όρων και συνδυάζοντας τη συχνότητα εμφάνισής τους στα έγγραφα (term frequency - tf) και τη συχνότητα εμφάνισης των όρων σε όλη τη συλλογή των εγγράφων (inverse document frequency - idf).

Όρος	**df_t**	**idf_t**
πώληση	22511	2,21
αυτοκίνητο	2584	3,05
ευκαιρία	38774	1,45
μάρκα	31558	1,57

Πίνακας 2: Συχνότητα όρων και στάθμιση βαρύτητας.

Για ένα σύνολο εγγράφων N μίας συλλογής, το μέγεθος idf ενός όρου t, ορίζεται ως:

$$\text{idf}_t = \log \frac{N}{\text{df}_t}$$

Η βαρύτητα ενός όρου t σε ένα έγγραφο d, δίνεται από τον τύπο:

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \text{ x } \text{idf}_t$$

Όπως φαίνεται από τον παραπάνω τύπο, δίνεται μεγαλύτερο βάρος σε όρους οι οποίοι εμφανίζονται σε λίγα έγγραφα, μικρότερο σε όσους εμφανίζονται πολύ συχνά, ενώ σχεδόν ασήμαντο, όταν ο όρος εμφανίζεται σχεδόν σε όλα τα έγγραφα [MRS09]. Επίσης, η βαρύτητα μπορεί να είναι δυαδικής μορφής, με την τιμή “0” να αντιστοιχεί στην απουσίας του όρου και την τιμή “1” στην αντίθετη περίπτωση [BBE+13].

2.1.5 Διαδικασία ερωτήματος

Η αναζήτηση της πληροφορίας ολοκληρώνεται με τη διαδικασία του ερωτήματος των όρων και εφαρμόζεται πάνω στο ευρετήριο το οποίο έχει δημιουργηθεί σε προηγούμενη φάση.

Αρχικά, εισάγεται το ερώτημα του χρήστη μέσω μίας διεπαφής του συστήματος, χρησιμοποιώντας τελεστές για τον καθορισμό του ερωτήματος και της βαρύτητας κάθε όρου που το απαρτίζει (π.χ. OR, AND, ταίριασμα ολόκληρης φράσης [match phrase]). Το ερώτημα μετασχηματίζεται (query transformation) με παρόμοιες τεχνικές που εφαρμόζονται κατά τον μετασχηματισμό κειμένου (tokenizing, stop words removal, stemming κ.ά.), έτσι ώστε το κείμενο να είναι συμβατό και συγκρίσιμο με τους όρους του ευρετηρίου. Στο στάδιο αυτό, χρησιμοποιούνται τεχνικές όπως ο έλεγχος ορθογραφίας (spell checking), η υπόδειξη ερωτήματος (query suggestion) και η επέκταση ερωτήματος (query expansion), οι οποίες προέρχονται από ανάλυση πληροφορίας βασισμένη στο σύνολο των εγγράφων της συλλογής, σε προηγούμενα ανακτηθέντα έγγραφα, σε έγγραφα που χρησιμοποιεί ο χρήστης ή σε έγγραφα που έχουν ήδη αναγνωριστεί ως χρήσιμα (relevance feedback) [CMS15].

Μία αρκετά συνηθισμένη τεχνική σύγκρισης λέξεων είναι αυτή της επεξεργασίας απόστασης (edit distance), σύμφωνα με την οποία υπολογίζεται ο αριθμός των λειτουργιών/επεξεργασιών που απαιτούνται για τον μετασχηματισμό μίας λέξης σε κάποια άλλη. Με την τεχνική της απόστασης Damerau-Levenshtein απαριθμείται ο ελάχιστος αριθμός εισαγωγών, διαγραφών, αντικαταστάσεων ή μεταθέσεων μονών χαρακτήρων που απαιτούνται για τον μετασχηματισμό [VJ15]. Έχουν αναπτυχθεί διάφορες τεχνικές για την επιτάχυνση των υπολογισμών μεταξύ των ανορθόγραφων λέξεων. Αυτές περιλαμβάνουν τον περιορισμό της σύγκρισης α) σε λέξεις που ξεκινούν μόνο με το ίδιο γράμμα (τα ορθογραφικά λάθη σπάνια γίνονται στο πρώτο γράμμα της λέξης), β) σε λέξεις που είναι του ίδιου ή παρόμοιου μήκους, (τα ορθογραφικά λάθη σπάνια αλλάζουν το μήκος της λέξης), και γ) σε λέξεις που ακούγονται το ίδιο [CMS15].

2.1.6 Διαδικασία κατάταξης

Επόμενο στάδιο, είναι η εφαρμογή της διαδικασίας κατάταξης και παρουσίασης των αποτελεσμάτων, κάτι που είναι ιδιαίτερα σημαντιόή, καθώς τα αποτελέσματα πρέπει όχι μόνο να ληφθούν σε μικρό χρονικό διάστημα, αλλά και να είναι σχετικά με την επιθυμητή πληροφορία. Η κατάταξη εξαρτάται από τον αλγόριθμο κατάταξης του μοντέλου ανάκτησης, όπου στην πιο απλή μορφή του, η βαθμολογία του εγγράφου υπολογίζεται ως εξής:

$$\text{Βαθμολογία εγγράφου} = \sum_{i} q_i.d_i$$

Σύμφωνα με τον παραπάνω τύπο, υπολογίζεται το άθροισμα όλων των όρων του λεξικού της συλλογής, με q_i τη βαρύτητα του κάθε όρου και d_i τη βαρύτητα του όρου στο έγγραφο, όπου η βαρύτητα κάθε όρου είναι παρόμοια με αυτήν του tf.idf [CMS15]. Εφαρμόζονται τεχνικές βελτιστοποίησης της συνολικής απόδοσης, όπως χρήση προσωρινής αποθήκευσης (caching) ή μερικής αντιγραφής με επιλογή αντιγράφου (partial replication with replica selection). Η επιδίωξη και των δύο τεχνικών είναι η μείωση του χρόνου απόκρισης ερωτήματος αναζητώντας λιγότερο κείμενο, διατηρώντας παράλληλα την ίδια αποτελεσματικότητα. Όταν τα ερωτήματα επαναλαμβάνονται ή σχετίζονται με τα ίδια έγγραφα τότε έχουν τοπικότητα, και αποθηκεύοντας τα αντίστοιχα ερωτήματα, η αναζήτηση περιορίζεται στα αντίγραφα προσωρινής αποθήκευσης ή μερικής αντιγραφής βελτιώνοντας έτσι τον χρόνο απόκρισης [LM03]. Επίσης, υπάρχουν διάφορες τεχνικές στη χρήση του αλγόριθμου υπολογισμού κατάταξης, στοχεύοντας στη μείωση του χρόνου υπολογισμού, με κάποιες από αυτές τις τεχνικές να θεωρούνται ασφαλείς (safe), όπου η κατάταξη δεν αλλάζει είτε εφαρμοστεί είτε όχι, ενώ κάποιες θεωρούνται μη ασφαλείς (unsafe) και θυσιάζουν την ακρίβεια της κατάταξης για μεγαλύτερη ταχύτητα απόκρισης [Bew95].

2.1.7 Διαδικασία αξιολόγησης

Η διαδικασία της αξιολόγησης (evaluation), στοχεύει στη μέτρηση της αποτελεσματικότητας και της αποδοτικότητας του συστήματος με σκοπό τη μελλοντική βελτίωσή του.

Θεωρείται χρήσιμη η καταγραφή της συμπεριφοράς των χρηστών (logging) για την ανάλυση των προτιμήσεων των χρηστών (click through data) και τον συνολικό χρόνο που αφιερώθηκε για την επίσκεψή τους στα αποτελέσματα (dwell time). Κατά αυτόν τον τρόπο, μετριέται η αποτελεσματικότητα των αλγορίθμων και γίνεται η συγκριτική τους αξιολόγηση [CMS15]. Πολύ σημαντικό μέγεθος στο οποίο βασίζονται οι αναλύσεις αυτές, είναι η ικανοποίηση των χρηστών (user satisfaction). Η ταχύτητα της απόκρισης, η κατανοησιμότητα της παρεχόμενης πληροφορίας (π.χ. snippets), καθώς και το μέγεθος του ευρετηρίου, αποτελούν παράγοντες που επηρεάζουν αυτό το μέγεθος [MRS09]. Μία επίσης, σημαντική πτυχή είναι η ανάλυση της αναζήτησης για κάθε ερώτημα χρήστη, ώστε να γίνεται χρήση του ιστορικού των προτιμήσεων, και να προτείνονται εξατομικευμένα αποτελέσματα αναζήτησης [Bj12].

Το βασικότερο στοιχείο που απαιτείται για τον προσδιορισμό της σχετικότητας των αποτελεσμάτων είναι μία συλλογή, η οποία αποτελείται από τρία βασικά στοιχεία: α) μία συλλογή εγγράφων, β) ένα σύνολο πληροφοριακών αναγκών, και γ) την πληροφόρηση που να προσδιορίζει ποια έγγραφα σχετίζονται με τις συγκεκριμένες πληροφοριακές ανάγκες (ground truth). Η αξιολόγηση γίνεται από την πλευρά των πληροφοριακών αναγκών των χρηστών και τον βαθμό ικανοποίησής τους από το παραγώμενο αποτέλεσμα [Uj11].

Από την πλευρά των χρηστών, σημαντική μέθοδος αξιολόγησης της σχετικότητας είναι η ανατροφοδότηση βάσει αποτελεσμάτων (relevance feedback), όπου χρήστες επιλέγουν τα πιο σχετικά από τα ανακτηθέντα έγγραφα και έπειτα από επαναϋπολογισμό βασισμένο στις επιλογές αυτές, το σύστημα επιστρέφει νέα αποτελέσματα [MRS09]. Η συνάφεια των αποτελεσμάτων εξαρτάται από τον αλγόριθμο και μπορεί να είναι είτε απόλυτη (boolean, relevant / non relevant) είτε με κατάταξη (ranked), όπου η συνάφεια θεωρείται ως σχετική.

2.1.8 Δυαδικό μοντέλο και μοντέλα κατάταξης

Το δυαδικό (Boolean) μοντέλο είναι ένα από τα παλαιότερα και πιο απλά IR μοντέλα, όπου τα έγγραφα αντιμετωπίζονται απλά ως ένα σύνολο από λέξεις (bag of words), και κάθε ένα είτε είναι σχετικό με το ερώτημα είτε όχι, καθώς δεν υπάρχει κανένας άλλος βαθμός συνάφειας, ενώ δεν λαμβάνεται υπόψη η συχνότητα εμφάνισης των όρων στα έγγραφα. Όλη η συλλογή περιγράφεται από έναν πίνακα εμφάνισης όρων στα έγγραφα (term-document incidence matrix), όπου κάθε γραμμή αντιστοιχεί σε έναν όρο και κάθε στήλη σε ένα έγγραφο. Σε κάθε κελί του πίνακα η τιμή “1” δηλώνει ότι ο όρος εμφανίζεται στο έγγραφο, ενώ η τιμή “0” δηλώνει την απουσία της.

	Documents																	
Terms	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}	d_{12}	d_{13}	d_{14}	d_{15}	d_{16}	d_{17}	d_{18}
acorn	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
barbaric	0	0	1	1	0	0	0	0	1	1	1	1	1	1	0	1	0	1
beautiful	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1

Εικόνα 5: Πίνακας εμφάνισης όρων στα έγγραφα [GRG18].

Οι όροι του ερωτήματος μπορούν να συνδυαστούν με τους τελεστές “AND”, “OR” και “NOT” που εφαρμόζονται πάνω στα έγγραφα. Το μοντέλο αυτό είναι πολύ απλό στην υλοποίησή του και δίνει πολύ καλά αποτελέσματα στην ανάκτηση συναφών εγγράφων, με την προϋπόθεση ότι ο χρήστης γνωρίζει το λεξιλόγιο του θέματος και μπορεί να χρησιμοποιήσει ερωτήματα με δυαδικούς τελεστές. Είναι αρκετά βολικό για θεματολογίες όπως η νομική έρευνα, όπου η υψηλή ανάκληση είναι προτιμητέα έναντι της χαμηλής ακρίβειας. Παρόλα αυτά, συχνά οι χρήστες δεν γνωρίζουν ακριβώς το λεξικό του ευρετηρίου, ενώ δεν υποστηρίζεται επιστροφή αποτελεσμάτων βάσει κατάταξης συνάφειας [GRG18].

Με το μοντέλο αυτό δεν υπάρχει εγγενώς η δυνατότητα για επιστροφή αποτελεσμάτων σε ανοχή σε ορθογραφικά λάθη, ενώ υπάρχει συχνά η απαίτηση για την αναζήτηση σύνθετων λέξεων ή φράσεων οι οποίες υποδηλώνουν έννοια, κάτι που απαιτεί την επαύξηση των δυνατοτήτων του ευρετηρίου για την καταγραφή της εγγύτητας των όρων στα έγγραφα. Επιπλέον σημαντικός περιορισμός αποτελεί το γεγονός ότι στο μοντέλο αυτό, δεν γίνεται η προσθήκη δυνατότητας για επαύξηση της βαρύτητας σε έγγραφα στα οποία υπάρχει μεγαλύτερη συχνότητα εμφάνισης όρων, οπότε απαιτείται η ύπαρξη του πίνακα συχνότητας όρων στη λίστα εγγράφων. Τέλος, με το μοντέλο αυτό δεν δύναται να υπάρχει βαθμολογία για τα επιστρεφόμενα αποτελέσματα για την αντίστοιχη αξιολόγηση από τον χρήστη.

Στα μοντέλα κατάταξης (ranked), οι χρήστες μπορούν να χρησιμοποιήσουν ερωτήματα ελεύθερου κειμένου χωρίς να απαιτείται η χρήση συγκεκριμένων τελεστών, αφού το σύστημα είναι σε θέση να καθορίσει τα πιο σχετικά έγγραφα. Στα μοντέλα αυτά, δύναται να γίνει χρήση στατιστικών εμφάνισης των όρων, κατηγοριοποίησης (classification), μηχανικής μάθησης (machine learning), ομοιότητας (similarity measure), καθώς και της θέσης που οι όροι αυτοί βρέθηκαν (proximity), ώστε να καθορίζεται αν οι όροι ενός ερωτήματος πρέπει να βρίσκονται ο ένας πλησίον του άλλου μέσα στο έγγραφο, περιορίζοντας τον μέγιστο παρεμβαλλόμενο αριθμό λέξεων. Σε αυτήν την κατηγορία μοντέλων, τα αποτελέσματα της αναζήτησης παρουσιάζονται ως μία ταξινομημένη λίστα, με στόχο τα πιο σχετικά αποτελέσματα να εμφανίζονται πρώτα [MRS09]. Μία ευρέως χρησιμοποιούμενη προσέγγιση για την αξιολόγηση της αποτελεσματικότητας των μοντέλων κατάταξης είναι η μέση τιμή ακρίβειας (average precision), όπου υπολογίζεται η ακρίβεια που αντιστοιχεί στις θέσεις κατάταξης των ανακτηθέντων εγγράφων, καθώς και η μέση ακρίβεια (mean average precision), η οποία είναι ο μέσος όρος των μέσων όρων της ακρίβειας, και υποδηλώνει το πόσο επιτυχημένο είναι το μοντέλο που εκτελεί το ερώτημα [GRG18].

Πέρα από την απλή προσέγγιση της ανάκτησης των ακριβώς πιο σχετικών εγγράφων, πολλές φορές υιοθετούνται διαφορετικές τεχνικές αναλόγως της περίστασης [MRS09]:

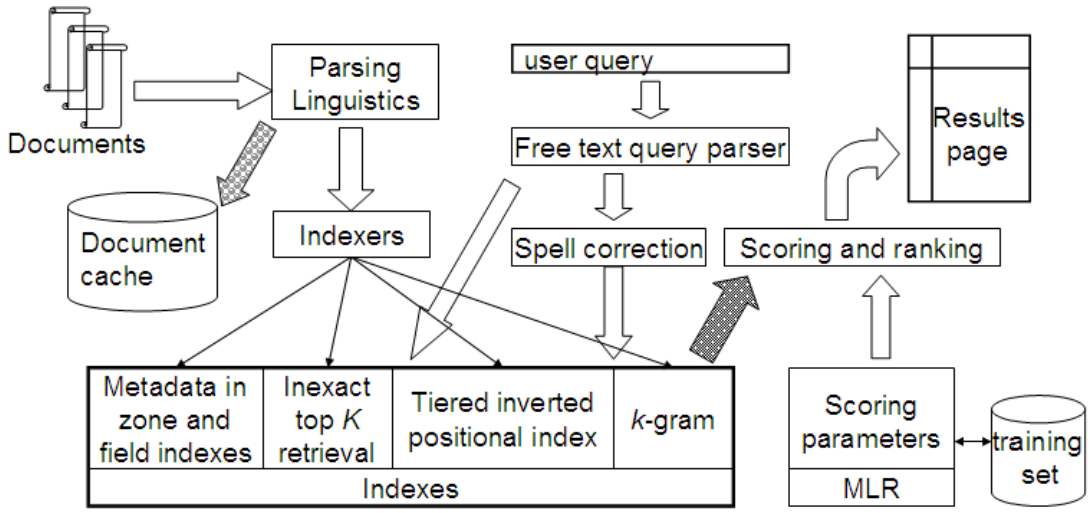
- Εύρεση των εγγράφων που έχουν σχεδόν περίπου το ίδιο σκορ κατάταξης με τα περισσότερο σχετικά έγγραφα (inexact top-K), ώστε να μειώνεται το υπολογιστικό κόστος της αναζήτησης, χωρίς να διαφοροποιείται η ικανοποίηση του χρήστη.
- Για ερωτήματα με πολλούς όρους, πέρα από την επιλογή των εγγράφων που περιέχουν μόνο το σύνολο των όρων, απορρίπτονται τα έγγραφα των οποίων το idf δεν ξεπερνάει ένα συγκεκριμένο κατώφλι (index elimination), αποφεύγοντας έτσι το κόστος ελέγχου των λιστών με όρους χαμηλού idf.
- Μία άλλη προσέγγιση είναι η χρήση βέλτιστων λιστών (champion lists), όπου για κάθε όρο προϋπολογίζεται το σύνολο εκείνων των εγγράφων (r) που έχουν την υψηλότερη βαρύτητα. Έπειτα, για κάθε ερώτημα επιλέγεται η ένωση από τις βέλτιστες λίστες με κάθε έναν από τους όρους που περιέχονται στο ερώτημα και οι υπολογισμοί γίνονται μόνο σε έγγραφα τα οποία προέκυψαν.
- Με την τεχνική χρήσης του στατικού σκορ ποιότητας (static quality score) υπάρχει ένα σκορ ποιότητας g(d) για κάθε έγγραφο d που κυμαίνεται στις τιμές ανάμεσα στο “0” και στο “1” και το οποίο είναι ανεξάρτητο απο το ερώτημα και επομένως σταθερό (static). Η συνολική κατάταξη προέρχεται από τον συνδυασμό του σταθερού ανά έγγραφο σκορ και της εξαρτώμενης από το ερώτημα βαθμολογίας. Τέτοιο παράδειγμα είναι ερωτήματατα σχετικά με κριτικές προϊόντων από σχόλια χρηστών.

2.1.9 Σύνοψη διαδικασίας ανάκτησης πληροφορίας

Ανακεφαλαιώνοντας, στα μοντέλα ανάκτηση πληροφορίας, χρησιμοποιούνται κυρίως τα παρακάτω χαρακτηριστικά [CMS15]:

- Ύπαρξη όρου (term occurrence): Αν ο όρος υπάρχει ή όχι στο έγγραφο.
- Συχνότητα όρου: Ο αριθμός των φορών που ο όρος εμφανίζεται στο έγγραφο.
- Ανεστραμμένη συχνότητα εμφάνισης των όρων στη συλλογή (inverse document frequency): Ο δείκτης σημαντικότητας ενός όρου για όλη τη συλλογή των εγγράφων.
- Μέγεθος εγγράφου (document length): Ο αριθμός των όρων σε ένα έγγραφο.
- Εγγύτητα όρων (term proximity): Η ύπαρξη μοτίβων εμφάνισης όρων σε ένα έγγραφο για τη διαπίστωση αλληλοεξαρτήσεων όρων.

Στην παρακάτω εικόνα, παρουσιάζεται ένα ολοκληρωμένο σύστημα αναζήτησης, σύμφωνα με όσα περιγράφηκαν έως τώρα.



Εικόνα 6: Ένα ολοκληρωμένο σύστημα αναζήτησης [MRS09].

Τα έγγραφα περνούν από τη γλωσσική επεξεργασία, δημιουργούνται τα ευρετήρια και οι περιλήψεις αποτελεσμάτων, όπου γίνεται εμφανές στον χρήστη ο λόγος για τον οποίο το έγγραφο ταιριάζει στο ερώτημα. Αντίγραφα των token χρησιμοποιούνται για τη δημιουργία μίας τράπεζας ευρετηρίων στην οποία αποθηκεύονται μεταδεδομένα για κάθε έγγραφο, ευρετήρια θέσης (positional indexes), ευρετήρια ορθογραφικού ελέγχου (spelling correction) κ.ά. Το ερώτημα ελεύθερου κειμένου χρησιμοποιείται στη βάση αυτών των ευρετηρίων και σε περίπτωση που δεν παραχθεί ικανοποιητικός αριθμός αποτελεσμάτων, χρησιμοποιούνται τα ευρετήρια ορθογραφικού ελέγχου ώστε να ανακτηθούν επιπλέον αποτελέσματα. Για τα ανακτηθέντα έγγραφα υπολογίζεται η βαθμολογία κατάταξής τους και τέλος εμφανίζονται ταξινομημένα στην αντίστοιχη σελίδα αποτελεσμάτων [MRS09].

2.1.10 Εύρεση εγγράφων στο διαδίκτυο

Ιδιαίτερα σημαντική λειτουργία για την οποία αξίζει να γίνει ξεχωριστή αναφορά είναι η εύρεση νέων εγγράφων που απαρτίζουν συλλογές στο διαδίκτυο. Η εύρεση νέων εγγράφων-ιστοσελίδων βασίζεται στα προγράμματα ανίχνευσης ιστού (web crawlers / spiders) των οποίων η λειτουργία είναι η γρήγορη και αποτελεσματική συλλογή ιστοσελίδων ακολουθούμενες από τους υπερσυνδέσμους που διασυνδέονται. Ο βασικός αλγόριθμος είναι απλός, ένα πρόγραμμα διατρέχει ένα δεδομένο σύνολο από διαδικτυακές διευθύνσεις (Uniform Resource Locators - URLs), εξάγει τους υπερσυνδέσμους που υπάρχουν σε αυτές, τους αποθηκεύει στο σύνολο συνόρου (URL frontier), και επαναληπτικά διατρέχει τις ιστοσελίδες που αντιστοιχούν στους υπερσυνδέσμους αυτούς. Με τη συνεχή προσπέλαση νέων ιστοσελίδων, οι διευθύνσεις του αρχικού συνόλου διαγράφονται από το σύνολο διευθύνσεων συνόρων και αντικαθιστούνται με νέες, εφόσον αυτές δεν υπάρχουν ήδη στο ευρετήριο. Κατά τη διάρκεια της παραπάνω διαδικασίας πρέπει να γίνονται σεβαστοί οι

κανόνες που έχουν τεθεί στους εξυπηρετητές σχετικά με την προσπέλασή τους, κάτι το οποίο συνήθως επιτυγχάνεται με τη χρήση του πρωτοκόλλου Robots Exclusion Protocol [ON10].

Η συμπεριφορά των προγραμμάτων αυτών καθορίζεται από τον συνδυασμό πολιτικών, όπως: α) της επιλογής (selection) που ορίζει τις ιστοσελίδες προς προσπέλαση, β) της επαναπροσπέλασης (re-visit), στην οποία ορίζεται η συχνότητα προσπέλασης των ιστοσελίδων αναλόγως των αλλαγών σε αυτές, γ) της ευγένειας (politeness), σύμφωνα με την οποία αποφεύγεται η προσπέλαση πολλών ιστοσελίδων της ίδιας διεύθυνσης για την αποφυγή υπερφόρτωσης του εξυπηρετητή, και δ) της παραλληλοποίησης (parallelization), η οποία καθορίζει τον συντονισμό των κατανεμημένων προγραμμάτων ανίχνευσης [MRS09].

Πρέπει να σημειωθεί ότι δεν είναι προσπελάσιμα όλα τα τμήματα του διαδικτύου, καθώς πολύ μεγάλο τμήμα του παραμένει κρυφό (hidden/deep Web). Το μεγαλύτερο μέρος του περιεχομένου της συγκεκριμένης κατηγορίας εμπίπτει α) σε προσωπικές ιστοσελίδες οι οποίες είναι ιδιωτικές, β) σε αποτελέσματα φορμών όπου τα αποτελέσματα είναι ορατά μόνο μετά την υποβολή δεδομένων, και γ) ιστοσελίδες που έχουν δημιουργηθεί με χρήση κάποιας γλώσσας σεναρίου κατά την προβολή (script language), έτσι ώστε κατά τη στιγμή της επίσκεψης του crawler στην ιστοσελίδα, δεν υπάρχει ορατό περιεχόμενο [CMS15].

2.2 Πολύπλευρη αναζήτηση

Λόγω των αναρίθμητων αποτελεσμάτων που μπορούν να προκύψουν κατά τη διάρκεια της αναζήτησης, η ανάκτηση της πληροφορίας από τις μηχανές αναζήτησης πρέπει να γίνεται με έξυπνο τρόπο, ενώ η επιστροφή της πληροφορίας προς τον χρήστη πρέπει να αφορά μόνο πληροφορία που είναι χρήσιμη και ενδιαφέρουσα προς αυτόν. Μελέτες έχουν δείξει ότι η επένδυση στην αύξηση της εμπειρίας χρήστη μπορούν να αποφέρουν απόδοση από 2 έως και 100 φορές την αρχική τους αξία. Στην επίτευξη των παραπάνω, σημαντικό ρόλο παίζει η χρήση των όψεων/πτυχών/πλευρών/διαστάσεων (facets), οι οποίες είναι συγκεκριμένα χαρακτηριστικά/παράμετροι (features) που βοηθούν τον χρήστη να κατανοήσει καλύτερα το εύρος της αναζήτησης [Bj12].

Οι όψεις είναι ουσιαστικά ανεξάρτητες ιδιότητες (attributes) ή διαστάσεις, σύμφωνα με τις οποίες είναι δυνατή η ταξινόμηση ενός αντικειμένου [RT12] ή κατηγορίες οι οποίες έχουν συγκεκριμένο νόημα και οργανώνονται με τέτοιον τρόπο που να αντικατοπτρίζουν της σχετικές έννοιες σε έναν συγκεκριμένο τομέα [NH14]. Η έννοια αυτή, είναι η κεντρική ιδέα της θεωρίας των όψεων (facet theory) και η πολύπλευρη αναζήτηση (faceted search) είναι η εφαρμογή αυτής της θεωρίας στο διαδικτυακό περιβάλλον. Σύμφωνα με αυτήν, τα ερωτήματα μη δομημένου κειμένου συνδυάζονται με την πολύπλευρη πλοήγηση [MIAT+18], όπου μέσω της διαδραστικής, ευρετικής και επαναλαμβανόμενης διαδικασίας αναζήτησης τα αποτελέσματα περιορίζονται προοδευτικά μόνο στα πιο σχετικά [WLZZ+13].

Με την πολύπλευρη αναζήτηση, προσφέρεται η δυνατότητα βελτίωσης της εμπειρίας αναζήτησης, καθώς παρέχεται ένα ευέλικτο πλαίσιο που επιτρέπει στους χρήστες να υλοποιήσουν αναζητήσεις που κυμαίνονται από την απλή ανάκτηση δυαδικής απάντησης (υπάρχει / δεν υπάρχει) έως πολύπλοκες διερευνητικές αναζητήσεις για την επίλυση προβλημάτων. Η μέθοδος αυτή είναι πλέον η κυρίαρχη τάση στην αλληλεπίδραση μεταξύ των μηχανών αναζήτησης και των χρηστών, ιδίως στο ηλεκτρονικό εμπόριο, ενώ έρευνες έχουν δείξει ότι μέσω της πολύπλευρης αναζήτησης παρέχεται πιο αποτελεσματική υποστήριξη αναζήτησης πληροφοριών στους χρήστες από ότι μέσω της συμβατικής αναζήτησης με λέξεις κλειδιά [RT12]. Για παράδειγμα, τα εμπορεύματα μπορούν να ερευτηριοποιηθούν σε διαφορετικές διαστάσεις όπως στυλ, χρώμα, μέγεθος κ.λπ. και οι χρήστες να προβούν σε αναζήτηση με συνδυασμό και αναδιάταξη των διαστάσεων [WLL12]. Ο τρόπος αναζήτησης αυτός, υιοθετήθηκε από τις ιστοσελίδες διαδικτυακού εμπορίου (π.χ. eBay, Amazon) λόγω του παρεχόμενου τρόπου σταδιακής βελτίωσης των αποτελεσμάτων [NH14], καθώς πλήθος ερευνών και μετρήσεων έχουν επανειλημμένως δείξει ότι πλεονεκτούν σε χρηστικότητα έναντι άλλων τρόπων αναζήτησης [SH15].

Τα ερωτήματα με τη χρήση όψεων επιστρέφουν άμεσα τα αποτελέσματα και η αλληλεπίδραση μεταξύ του χρήστη και της μηχανής αναζήτησης είναι χωρίς διακοπές και περισσότερο στοχευμένη, καθώς συμβάλουν στην καλύτερη κατανόηση των όρων και του θέματος, ενώ επίσης, βελτιώνουν την εμπειρία χρήστη λόγω της αμεσότητας της αλληλεπίδρασης και της επιστροφής των αποτελεσμάτων [MIAT+18]. Ταυτόχρονα, οι χρήστες μπορούν να οδηγηθούν στην πληροφορία βάσει του τι θυμούνται σχετικά με την πληροφορία που αναζητούν, ενώ οι διεπαφές όψεων βοηθούν την αύξηση της φιλικότητας προς τους χρήστες ως προς το να μην νιώθουν αποπροσανατολισμένοι [NH14].

Σε σύγκριση με την παραδοσιακή αναζήτηση, η κυριότερη διαφορά είναι η ευρετηρίαση των πόρων. Στην παραδοσιακή αναζήτηση, η ευρετηρίαση και η αναζήτηση βασίζονται σε όρους, χωρίς να καθορίζεται η σημασιολογία της φράσης αναζήτησης, ενώ στην αναζήτηση με διαστάσεις η ευρετηρίαση γίνεται με τη χρήση διαφορετικών διαστάσεων, όπου κάθε μία έχει διαφορετική σημασιολογική πληροφορία. Ταυτόχρονα, η φράση αναζήτησης συσχετίζεται με κάποια διάσταση, οπότε και αποκτά και την εκάστοτε σημασιολογία.

Μια διάσταση μπορεί να περιγραφεί ως ένα σύνολο βασικών λέξεων ή διαστάσεων μικρότερης λεπτομέρειας. Η τιμή μίας διάστασης (facet value) είναι ένα ανεξάρτητο αντικείμενο που περιγράφει μία συγκεκριμένη πτυχή ενός πόρου [WLL12]:

$$V = \{term_1, term_2, \ldots term_i, \ldots, term_n\}$$

Τέτοιο παράδειγμα είναι οι τιμές των χρωμάτων που μπορούν να περιγράψουν ένα προϊόν.

Οι διαστάσεις που αποτελούνται από ένα σύνολο βασικών όρων ονομάζονται βασικές (basic):

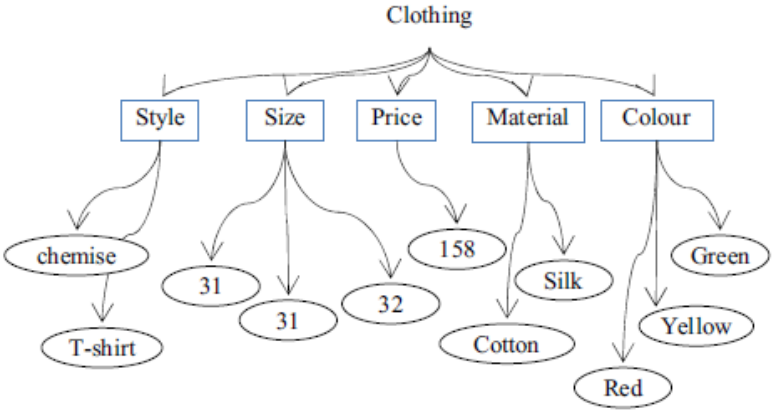
$$BF = \{V_1, V_2, \ldots V_i, \ldots, V_n\}$$

Σύνθετες (complex) είναι αυτές που αποτελούνται από άλλες, μικρότερες σε λεπτομέρεια:

$$CF = \{BF_1, BF_2, \ldots BF_i, \ldots, BF_n\}$$

Τα δέντρα διαστάσεων (facets tree) αποτελούνται από διαφορετικής λεπτομέρειας διαστάσεις, στα οποία οι χαμηλότερου βαθμού διαστάσεις έχουν τις δικές τους τιμές:

$$FT = \{CF, BF, V\}$$



Εικόνα 7: Παράδειγμα δέντρου διαστάσεων [WLL12].

Οι ερευνητές έχουν κυρίως να απαντήσουν σε δύο ερωτήματα: α) πως να γίνει η επιλογή των διαστάσεων με αυτόματο τρόπο, και β) με ποιον τρόπο να γίνει η ευρετηρίαση των διαστάσεων που προκύπτουν από ένα σύνολο ιστοσελίδων [WLL12]. Οι διαστάσεις πολλές φορές προέρχονται από προϋπάρχοντα πεδία στα μεταδεδομένα του κάθε στοιχείου ή δημιουργούνται χειροκίνητα από την ανάλυση των στοιχείων [CJB14].

Σε πολλά συστήματα, η αντιστοίχιση των διαστάσεων των πόρων καθορίζεται χειρωνακτικά. Κάτι τέτοιο είναι σχετικά εύκολο σε συγκεκριμένες εφαρμογές που αφορούν το διαδικτυακό εμπόριο, καθώς είναι εύκολος ο προσδιορισμός των διαφορετικών χαρακτηριστικών και των τιμών που μπορούν να λάβουν, αφού κάθε εμπόρευμα έχει τις σχετικές προδιαγραφές οι οποίες μπορούν να βρεθούν εύκολα. Στον αντίποδα όμως, βρίσκονται συστήματα που έχουν μη δομημένη πληροφορία, όπως άρθρα, ιστοσελίδες, πατέντες, αγγελίες κ.λπ., των οποίων οι διαστάσεις είναι δύσκολο να καθοριστούν με τον προηγούμενο τρόπο [WLL12]. Για την ευρεία ανάπτυξη τέτοιων συστημάτων πρέπει να υλοποιηθούν τεχνικές για αυτόματη δημιουργία διαστάσεων, μέσω του προσδιορισμού των χρήσιμων διαστάσεων και της δημιουργίας της ιεραρχίας για κάθε αναγνωρισμένη διάσταση.

Οι υφιστάμενοι μέθοδοι αυτόματης εξαγωγής διαστάσεων χωρίζονται σε τρεις κατηγορίες που αντιστοιχούν στους τύπους διαφορετικών δεδομένων, οι οποίοι είναι μη δομημένα, ημιδομημένα και δομημένα. Ενδεικτικά, μία τέτοια τεχνική είναι ο προσδιορισμός των σημαντικών φράσεων σε κάθε έγγραφο, η επέκταση κάθε φράσης με χρήση φράσεων στο ίδιο εννοιολογικό περιεχομένου μέσω εξωτερικών πηγών (π.χ. Wikipedia), η δημιουργία των διαστάσεων, και τέλος η σύγκριση της κατανομής των όρων με την αρχική βάση δεδομένων

για τον προσδιορισμό των όρων που μπορούν να χρησιμοποιηθούν για την επιλογή των διαστάσεων [WI12]. Άλλοι προτεινόμενοι από τη βιβλιογραφία τρόποι, είναι α) η επιλογή όρων που συναντώνται συχνότερα από ένα συγκεκριμένο κατώφλι (threshold), β) η χρήση της διασποράς των όρων σε μία συλλογή, και γ) η χρήση πιθανοτικών μεθόδων (probabilistic) με χρήση υπολογισμού κατανομής όρων [WLZZ+13].

Τα βασικά χαρακτηριστικά οποιασδήποτε συλλογής εγγράφων βασισμένη σε διαστάσεις είναι α) η ιεραρχία των διαστάσεων, και β) η χαρτογράφηση των εγγράφων στην ιεραρχία αυτή [YGN+08]. Με τη δημιουργία της ιεραρχίας στοχεύεται η ανακάλυψη των σχέσεων "είναι" ("is-a") και "αποτελεί μέρος" ("part-of") μεταξύ των όρων της συλλογής [Pr07]. Μέσω της ιεραρχίας αυτής, υποστηρίζεται η αναζήτηση πάνω σε μεγάλα σύνολα στοιχείων. Συνήθως, η περιήγηση υποστηρίζεται από μία μοναδική ιεραρχία ή ταξονομία (taxonomy), βάσει της οποίας οργανώνεται θεματικά το περιεχόμενο της συλλογής, κάτι που σπάνια είναι συνεπές με τα περιεχόμενα αυτής [WI12]. Δεδομένου ότι η αντιστοίχιση μεταξύ των διαστάσεων των οντοτήτων και των ιδιοτήτων του μοντέλου Οντοτήτων – Συσχετίσεων με αυτά των βασικών διαστάσεων, εξαρτώνται από το εκάστοτε πεδίο εργασίας/αναζήτησης, δεν υπάρχουν προκαθορισμένα σύνολα διαστάσεων που να είναι αποδεκτά για κάθε περίπτωση. Μία εκ των κατηγοριοποιήσων δημιουργίας διαστάσεων που έχουν προταθεί είναι η παρακάτω [ST09]:

- P (Οντότητα/Ποιος – Personality/Who), η οποία περιγράφει τη φύση του αντικειμένου και θεωρείται ως η βασική διάσταση.
- M (Υλικό/Τι – Matter/What), στην οποία περιγράφεται το υλικό του αντικειμένου.
- E (Ενέργεια/Πως – Energy/How), όπου περιγράφονται οι διαδικασίες που σχετίζονται με το αντικείμενο.
- S (Χώρος/Που – Space/Where) που υποδηλώνει που βρίσκεται/υπάρχει το αντικείμενο.
- T (Χρόνος/Πότε – Time/When), σύμφωνα με την οποία περιγράφεται πότε συμβαίνει ή υπάρχει το αντικείμενο προς κατηγοριοποίηση.

Η βασική απαίτηση κατά την τροφοδότηση εγγράφων σε συλλογή βασισμένη σε διαστάσεις, είναι η συσχέτιση κάθε εγγράφου με τις αντίστοιχες διαστάσεις και η διασφάλιση της συνοχής της ταξονομίας των διαστάσεων και των μεταξύ τους ιεραρχικών σχέσεων. Υπάρχουν δύο προσεγγίσεις για την τροφοδότηση εγγράφων σε πολύπλευρης μορφής συλλογές: α) η ύπαρξη μίας ολοκληρωμένης ταξονομίας διαστάσεων πριν την ευρετηρίαση που γνωστή στο σύστημα, και β) το σύστημα να μαθαίνει την ταξονομία κατά τη διάρκεια της ευρετηρίασης των εγγράφων [YGN+08]. Η δημουργία και συντήρηση ταξονομιών με μη αυτόματο τρόπο είναι χρονοβόρα και απαιτεί μεγάλο φόρτο εργασίας, ενώ το ίδιο ισχύει και για την αναγνώριση των διαφορετικών διαστάσεων των εγγράφων [WLZZ+13].

Όσον αφορά την αλληλεπίδραση χρήστη, η τυπική αλληλεπίδραση χρήστη μέσω διεπαφής όψεων περιλαμβάνει τη βελτίωση ενός ερωτήματος αναζήτησης ή την πλοήγηση μέσω πολλαπλών ανεξάρτητων ιεραρχιών. Η διαδικασία αυτή αποσκοπεί στην περιγραφή των δεδομένων είτε με μεγαλύτερη λεπτομέρεια (refinement) είτε γενικεύοντάς τα (generalization) [YGN+08], αποσκοπώντας στην επίλυση του προβλήματος της παροχής υπερπληθώρας δεδομένων (information overload problem), σύμφωνα με το οποίο οι χρήστες πρέπει να αφιερώσουν πολύ χρόνο στην επιλογή της παρεχόμενης πληροφορίας από τα επιστρεφόμενα αποτελέσματα. Το πρόβλημα αυτό γίνεται περισσότερο κατανοητό, αναλογιζόμενοι το γεγονός ότι αποτελέσματα διαφορετικής θεματολογίας τείνουν να αναμιγνύονται σε μία λίστα αποτελεσμάτων. Αυτό λύνεται με την πολύπλευρη αναζήτηση παρέχοντας στους χρήστες τη δυνατότητα καλύτερης εξερεύνησης του πληροφοριακού χώρου και βελτιώνοντας σταδιακά τις επιλογές της αναζήτησης σε κάθε διάσταση [NH14]. Το σύνολο των επιλογών κάθε στιγμή είναι γνωστό ως πλαίσιο πλοήγησης (navigational context) και αντιστοιχεί στην εκάστοτε θέση του χρήστη στον πληροφοριακό χώρο.

Μία βασική αρχή της πολύπλευρης αναζήτησης είναι η ελαχιστοποίηση της πιθανότητας παραγωγής μηδενικών αποτελεσμάτων, καθοδηγώντας τους χρήστες σε παραγωγικές επιλογές πλοήγησης [WLZZ+13]. Στην πράξη, αυτό σημαίνει ότι πρέπει να εμφανίζονται ως ενεργές μόνο οι επιλογές που οδηγούν στην παραγωγή αποτελεσμάτων, ενώ όσες οδηγούν σε αδιέξοδα πρέπει να είναι ανενεργές ή μη εμφανείς [RT12]. Με αυτόν τον τρόπο λαμβάνεται ταυτόχρονα υπόψη η δημοτικότητα των εννοιολογικών στοιχείων από τα οποία απαρτίζονται τα δεδομένα [SH15]. Με τον επανακαθορισμό των διαστάσεων, παρουσιάζονται στον χρήστη πιθανές βελτιώσεις μέσω της εμφάνισης υποκατηγοριών ή της συνάθροισης των αποτελεσμάτων που πληρούν τα κριτήρια τόσο του ερωτήματος κειμένου όσο και αυτά των διαστάσεων. Με την ποσοτική επισκόπηση του πλήθους και της ποικιλίας των διαθέσιμων δεδομένων, παρέχεται η καθοδήγηση στους χρήστες σχετικά με το μονοπάτι στο οποίο μπορούν να εμβαθύνουν για την καλύτερη στόχευση των αποτελεσμάτων και ενισχύουν την αναλυσιμότητα των ευρετηριοποιημένων πολυδιάστατων δεδομένων [YGN+08].

Πρέπει όμως να ληφθεί υπόψη ότι ορισμένες φορές, η χρήση των διαστάσεων είναι πιθανόν να οδηγήσει σε ελάχιστα αποτελέσματα με πολύ γρήγορο ρυθμό κάτι που μπορεί να έχει αρνητικό αντίκτυπο στην εμπειρία του χρήστη και για αυτόν τον λόγο, θα πρέπει να γίνεται προσεκτική επιλογή των διαστάσεων που θα εμφανίζονται [NH14], καθώς η μείωση της παραγωγής αποτελεσμάτων μπορεί να φτάσει ακόμα και σε λογαριθμικό βαθμό σε σχέση με τις επιλεγμένες διαστάσεις [SH15]. Επίσης, άλλος ένας περιορισμός της πολύπλευρης αναζήτησης είναι ότι ενώ οι χρήστες μπορούν να βελτιώσουν προοδευτικά τη στόχευση ενός ερωτήματος, παρόλα αυτά δεν μπορούν να το επεκτείνουν. Για παράδειγμα, με την εισαγωγή του όρου “ελληνική επανάσταση” σε ένα πεδίο αναζήτησης και τη χρήση διαστάσεων όπως

“έτος”, “ήρωας” μπορούν να βρεθούν συγκεκριμένα αποτελέσματα, αλλά δεν είναι δυνατή η γενίκευση (zoom out) σε μεγαλύτερο πεδίο παγκόσμιας ιστορίας [NH14].

Σε προσαρμοστικές διεπαφές αναζήτησης (adaptive faceted search interfaces), τα στοιχεία ελέγχου προσαρμόζονται δυναμικά στα πραγματικά δεδομένα, τα οποία περιορίζονται από τις εκάστοτε επιλογές του χρήστη [Vp09]. Οι τρόποι πλοήγησης σε μία τέτοια διεπαφή διακρίνονται: α) στη συγκεκριμενοποίηση του ερωτήματος (zoom-in), β) στη γενίκευση του ερωτήματος (zoom-out), γ) στην αντικατάσταση μέρους του ερωτήματος (shift), δ) στην αντικατάσταση του συνόλου του ερωτήματος (pivot), ε) στην αποσυνδετική επιλογή (disjunctive selection) πολλαπλών εννοιών μέσα σε μία διάσταση (slice-and-dice), και στ) στον καθορισμό του ερωτήματος με χρήση διάστασης τιμών (range selection). Η τροποποίηση των ερωτημάτων και η μετάβαση από μία κατάσταση σε άλλη ονομάζεται ως σύνδεσμος πλοήγησης (navigation link) που απαρτίζεται από την επιλογή (selection) και τον τρόπο λειτουργίας πλοήγησης (navigation mode), κάτι που σημαίνει ότι η ίδια επιλογή μπορεί με χρήση διαφορετικών τρόπων να οδηγήσει στα ίδια αποτελέσματα [ST09].

Η επιλογή και αποεπιλογή των φίλτρων αποτελεί τη σημαντικότερη λειτουργία στην πολύπλευρη αναζήτηση. Οι επιλογές των διαστάσεων μπορούν να είναι μονής (single-select) ή πολλαπλής επιλογής (multi-select), όπου στην πρώτη περίπτωση, οι τιμές των όψεων είναι αμοιβαίως αποκλειόμενες (mutually exclusive), ώστε σε κάθε δεδομένη στιγμή να μπορεί να εφαρμοστεί μόνο μία, ενώ στη δεύτερη μπορούν να έχουν πολλαπλές τιμές. Οι όψεις πολλαπλής επιλογής μπορούν να είναι είτε διαζευκτικής λογικής (OR) είτε συμπλεκτικής (AND). Στην πρώτη κατηγορία θεωρείται ότι οι τιμές συνδυάζονται αποσυμπλεκτικά (disjunctively), ώστε το αντικείμενο να ανήκει μόνο σε μία κατηγορία (π.χ. έτος πρώτης προβολής μίας ταινίας), ενώ στη δεύτερη οι τιμές θεωρείται ότι συνδυάζονται ταυτόχρονα (conjunctively), ώστε το αντικείμενο να μπορεί να ανήκει σε πολλαπλές κατηγορίες (π.χ. πολλοί συμμετέχοντες ηθοποιοί σε μία ταινία). Είναι σύνηθες, οι τιμές που εφαρμόζονται σε διαφορετικές όψεις, να θεωρούνται ότι εφαρμόζονται ταυτόχρονα (π.χ. “Σκηνοθέτης = Στήβεν” AND “Είδος = Περιπέτεια” AND “Έτος = 2015”), ενώ οι τιμές που εφαρμόζονται στην ίδια όψη να θεωρούνται ότι εφαρμόζονται αποσυμπλεκτικά (π.χ. “Έτος = 2015” OR “Έτος = 2016” OR “Έτος = 2017”). Κάτι τέτοιο όμως, επιδέχεται διαφοροποίησης αναλόγως των αναγκών και χαρακτηριστικών της εκάστοτε εφαρμογής, ενώ οι συνδυασμοί των επιλογών των όψεων τροποποιούνται δυναμικά με τις διάφορες επιλογές του χρήστη [RT12].

Οι διεπαφές που επιτρέπουν μόνο μία επιλογή ανά διάσταση υποστηρίζουν τη λειτουργία shift με τον πιο εύκολο τρόπο, καθώς η αντικατάσταση του ερωτήματος γίνεται με ένα μόνο κλικ. Αν επιτρέπονται πολλές τιμές ανά διάσταση (slice-and-dice), τότε πρέπει να γίνεται διάκριση ανάμεσα στη λειτουργεία zooming-in προσθέτοντας την επιλεγμένη τιμή στο ενεργό φίλτρο και στη λειτουργία shift, αντικαθιστώντας τις προηγούμενες τιμές από τη

συγκεκριμένη διάσταση. Η λειτουργία pivoting, συνηθως υποστηρίζεται απευθείας μέσω των λεπτομερειών κάθε αποτελέσματος, με συνήθη πρακτική τη μετάβαση σε νέα αποτελέσματα με την επιλογή τιμών που εμφανίζονται στα μεταδεδομένα των αποτελεσμάτων.

Η οπτικοποίηση της ιεραρχίας μεταξύ των διαστάσεων γίνεται με διάφορους τρόπους. Στις περιπτώσεις που δεν υπάρχει κάποια ιεραρχία, τότε αρκεί η εμφάνιση μίας ταξινομημένης λίστας των διαστάσεων. Διαφορετικά, οι διαστάσεις μπορούν να εμφανίζονται α) με τη μορφή πτυσσόμενου ή μη δέντρου, β) με τη λειτουργία εμβάθυνσης και αντικατάστασης (zoom and replace), όπου ο χρήστης με κάθε κλικ εμβαθύνει σε εύρος περισσότερων επιλογών τιμών, γ) μέσω πτυσσόμενων πάνελ όπου η ιεραρχία αναπαριστάται με τη μορφή ενός επιπέδου τύπου ακορντεόν, και δ) με τη λειτουργία συνεχούς εμβάθυνσης (continuous zooming), όπου το κάθε επίπεδο ιεραρχίας ξεδιπλώνεται οριζόντια με κάθε επιλογή [ST09].

2.3 Σχεδίαση παρουσίασης πληροφορίας και εμπειρία χρήστη

Η συνολική αποτελεσματικότητα των συστημάτων ανάκτησης πληροφοριών, εξαρτάται όχι μόνο από την ανάκτηση της σωστής πληροφορίας αλλά και από την ευκολία με την οποία μπορούν να χρησιμοποιήσουν το σύστημα. Αυτό συνίσταται στον εύκολο και γρήγορο εντοπισμό της πληροφορίας, την κατανόησή της και τέλος την παροχή δυνατότητας για τη χρήση αυτής, κάτι το οποίο περιγράφεται με το ακρωνύμιο LUNA (Locate, UNderstand, Act [Εντοπισμός, Κατανόηση, Δράση]) [BLLW17]. Η πληροφοριακή ανάγκη δεν ικανοποιείται από ένα και μοναδικό ιδανικό έγγραφο αλλά από το σύνολο και τη συνάθροιση της γνώσης που συλλέγεται κατά τη διάρκεια της ίδιας της διαδικασίας αναζήτησης [RT12].

Αν ο χρήστης δεν μπορεί να εντοπίσει εύκολα την πληροφορία, τότε η επικοινωνία μεταξύ του συστήματος και του χρήστη αποτυγχάνει [BLLW17]. Για τον λόγο αυτόν, σημαντικό ρόλο στη σχεδίαση τέτοιων συστημάτων, έχει η Σχεδιαση Παρουσίασης Πληροφορίας (Information Design), σύμφωνα με την οποία για την ικανοποίηση των πληροφοριακών αναγκών των αποδεκτών της πληροφορίας, πρέπει να γίνεται η ανάλυση, ο σχεδιασμός, η παρουσίαση και η κατανόηση ενός μηνύματος, ανεξαρτήτως μέσου και ικανοποιώντας αισθητικά, οικονομικά και εργονομικά τις τιθέμενες απαιτήσεις του χρήστη. Κατανοώντας ότι τα δεδομένα συχνά είναι περίπλοκα, μη οργανωμένα και μη δομημένα, μέσω της σχεδίασης πληροφορίας δίνεται η δυνατότητα οργάνωσης, διαμόρφωσης και παρουσίασής τους ώστε να γίνονται εύληπτα ως σημαντικές πληροφορίες [Pr07], ενώ σε τελική ανάλυση βασικός στόχος της σχεδίασης πληροφορίας είναι η σαφήνεια της επικοινωνίας και της κατανόησης του μηνύματος από τους αποδέκτες [Pr10].

Έχουν προταθεί διάφορες αρχές σχεδίασης παρουσίασης της πληροφορίας. Για το σύστημα που πραγματεύεται η συγκεκριμένη διπλωματική εργασία λαμβάνονται υπόψη η συνέπεια της σχεδίασης, η εγγύτητα των στοιχείων για την ανάδειξη της σχέσης μεταξύ του περιεχομένου,

η διάκριση των αποτελεσμάτων, η συγκεκριμένη στοίχιση του περιεχομένου, η ιεράρχιση των στοιχείων, η ορθή τήρηση της δομής, η διαισθητική ροή και η σαφήνεια του περιεχομένου [Pr07], καθώς επίσης και η αποφυγή της υπερφόρτωσης πληροφορίας [Bj12].

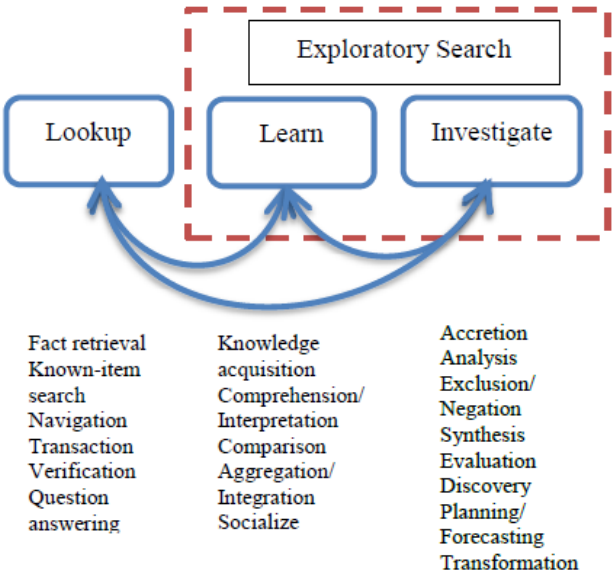
Κατά τον σχεδιασμό τόσο του εσωτερικού συστήματος αναζήτησης όσο και της αντίστοιχης διεπαφής χρήστη, είναι πολύ σημαντικό να γίνει κατανοητό ότι στον πυρήνα όλης αυτής της διαδικασίας βρίσκεται ο χρήστης και όχι το σύστημα αναζήτησης [RT12] και ότι απώτερος σκοπός είναι η ικανοποίηση του χρήστη μέσω της παροχής ικανοποιητικής εμπειρίας (User Experience). Ως εμπειρία χρήστη ορίζεται οι αντιλήψεις και οι αποκρίσεις του χρήστη από την αναμενόμενη χρήση ενός προϊόντος, συστήματος ή υπηρεσίας [ISO 9241-210:2010].

Η σχεδίαση με επίκεντρο τον χρήστη περιλαμβάνει τρία διαφορετικά στάδια: α) την κατανόηση των χρηστών, των αναγκών τους και της ροής εργασίας τους, β) τον σχεδιασμό της αρχιτεκτονικής της πληροφορίας με τις απαραίτητες λειτουργείες και σωστή εμφάνιση, και γ) την αξιολόγηση σχετικά με το αν η προτεινόμενη λύση καλύπτει επιτυχώς τις ανάγκες των χρηστών και τηρεί τη χρηστικότητα.

Η παραπάνω διαδικασία είναι πολύ σημαντική, καθώς έχει καταδειχθεί συσχέτιση ανάμεσα στην ικανοποιητική εμπειρία χρήστη και την αφοσίωση πελατών, ενώ η καλύτερη χρηστικότητα έχει θετικό αντίκτυπο στο κόστος δαπανών ανάπτυξης, εκπαίδευσης και υποστήριξης συστημάτων. Ως παράδειγμα στον τομέα των ακαδημαϊκών, μπορεί θα θεωρηθεί η έλλειψη επιλογών ή η αδυναμία επικοινωνίας εύρεσης μίας σχετικής με την έρευνα τους δημοσίευση, κάτι το οποίο μπορεί να σημαίνει τη διαφορά μεταξύ αποδοχής και απόρριψης μίας δημοσίευσης ή επιχορήγησης [Bj12], οδηγώντας έτσι όχι μόνο στην αναστάτωση του χρήστη που δεν κατέστη δυνατό να βρει τη ζητούμενη πληροφορία, αλλά έχοντας και αρνητικό οικονομικό αντίκτυπο.

Από τη σκοπιά της πρόσβασης στην πληροφορία, υπάρχει η διάκριση σε εστιασμένη αναζήτηση (focalized search) και σε διερευνητική αναζήτηση (exploratory search / browsing). Στην εστιασμένη αναζήτηση, οι χρήστες προσπαθούν να εντοπίσουν σχετικά γρήγορα τα στοιχεία πληροφοριών με βάση το περιεχόμενό τους (π.χ. η αναζήτηση ενός ζωγράφου σε μια εγκυκλοπαίδεια), ενώ στη διερευνητική αναζήτηση οι χρήστες εξερευνούν τις συσχετίσεις ανάμεσα στα αντικείμενα (η αναζήτηση της σχέσης και των αλληλεξαρτήσεων μεταξύ του συγκεκριμένου ζωγράφου με άλλους καλλιτέχνες σε συγκεκριμένη χρονική περίοδο και γεωγραφικό χώρο). Η εστιασμένη αναζήτηση μπορεί να θεωρηθεί πιο άμεση, ενώ η διερευνητική περισσότερο πολύπλοκη και για να είναι αποτελεσματική πρέπει να παρέχει τη δυνατότητα στους χρήστες α) για γρήγορη εύρεση όλων των σχετικών χαρακτηριστικών και των αντικειμένων που τους ενδιαφέρουν, β) να εστιάζουν στα πιο σημαντικά από αυτά απορρίπτοντας όσα αντικείμενα και χαρακτηριστικά δεν καλύπτουν τα κριτήρια επιλογής τους, και γ) να εξερευνούν τα αντικείμενα και τα

χαρακτηριστικά αναλόγως των επιλογών τους οι οποίες μπορούν να τροποποιούνται δυναμικά [ST09].



Εικόνα 8: Η διευρευνητική αναζήτηση [MAI18].

Ο προτεινόμενος σχεδιασμός των συστημάτων πρέπει να έχει ολοκληρώσει το στάδιο κατανόησης των χρηστών, λαμβάνοντας υπόψη: α) τον τύπο του χρήστη (user type), ο οποίος υποδηλώνει το επίπεδο γνώσης και εξειδίκευσής του, β) τον στόχο του χρήστη (goal), ο οποίος δείχνει το επίπεδο βαθμού χρήσης της αναζήτησης (από απλή έως σύνθετη αναζήτηση και ανάλυση αποτελεσμάτων), γ) το περιεχόμενο (content) του πεδίου αναζήτησης, και δ) τη λειτουργία αναζήτησης (search mode) που λαμβάνει υπόψη τα παραπάνω περιλαμβάνοντας την αναζήτηση, τη σύγκριση, την ανάλυση και την αξιολόγηση των αποτελεσμάτων.

Συνοπτικά, το πλαίσιο της αναζήτησης θα μπορούσε να περιγραφεί ως "πού είσαι, ποιός είσαι τι βρίσκεται δίπλα σου και πότε κάνεις την αναζήτηση". Αυτό γίνεται κατανοητό αφού οι χρήστες αναζητούν αποτελέσματα όχι μόνο βάσει θεματικής συνάφειας, αλλά και συσχετισμένα με τη φυσική τοποθεσία στην οποία βρίσκονται, καθώς και της χρονικής στιγμής που δημοσιεύτηκε η πληροφορία που θεωρείται σχετική (freshness). Ο συνδυσμός της φρεσκάδας με το χωρικό πλαίσιο προσιδιάζει με τη λογική των "έκτακτων/τελευταίων ειδήσεων" τα οποία είναι συγκεκριμένα για τον εκάστοτε χρήστη.

Ο βαθμός εμπειρίας των χρηστών μπορεί να διακριθεί στην εξειδίκευση τομέα/πεδίου και την τεχνική εξειδίκευση, με τους έμπειρους χρήστες να συσχετίζουν ευκολότερα την αναζήτηση με την πληροφορία. Η εξειδίκευση σε τομέα καθορίζει την εξοικείωση των χρηστών με ένα συγκεκριμένο θέμα, ενώ η τεχνική εξειδίκευση προσδιορίζει την ικανότητα των χρηστών στη χρήση τεχνικών μέσων, όπως η χρήση υπολογιστή, του διαδικτύου, των μηχανών αναζήτησης, συγκεκριμένων προγραμμάτων κ.λπ. Οι αρχάριοι χρήστες συνήθως ακολουθούν μία αναζήτηση σε εύρος (breadth first), αποφεύγοντας την απομάκρυνσή τους από το αρχικό

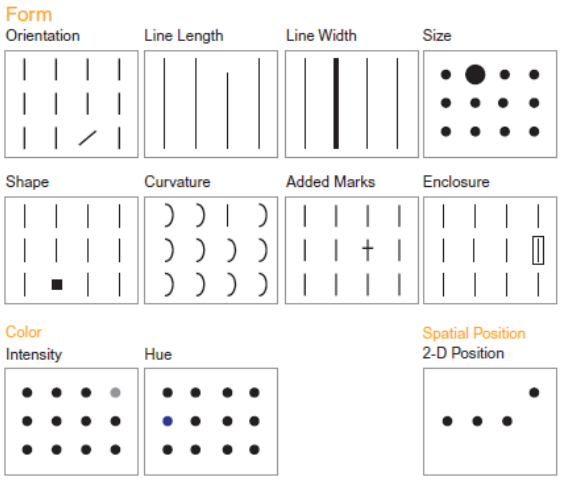
σημείο αναζήτησης, ενώ οι έμπειροι χρήστες εφαρμόζουν μία εις βάθος αναζήτηση (depth first) ακολουθώντας μεγαλύτερο αριθμό συνδέσμων σε καθετοποιημένο επίπεδο, καθώς γνωρίζουν σε μεγαλύτερο βαθμό το αντικείμενο, όπως και το τι ακριβώς επιζητούν[RT12]. Πέρα από τον διαφορετικό βαθμό εμπειρίας των χρηστών και τους διαφορετικούς τρόπους που χρησιμοποιούν κατά την εύρεση πληροφοριών, ο ίδιος χρήστης μπορεί να έχει διαφορετικές ανάγκες ανάλογα με τον τελικό σκοπό που έχει κάθε φορά [Bj12]. Για παράδειγμα, η συμπεριφορά ενός μεσίτη για την αναζήτηση αγγελιών ακινήτων για κάποιον πελάτη μπορεί να είναι πολύ διαφορετική από ότι όταν ψάχνει ακίνητο για τον ίδιο. Κατανοώντας τη συμπεριφορά των χρηστών ενός συγκεκριμένου συστήματος, δίνεται η δυνατότητα βελτιστοποίησης της σχεδίασης βάσει των στρατηγικών και των τρόπων αναζήτησης που επιδέχονται βελτίωσης [RT12].

Η αναζήτηση μπορεί να γίνεται είτε χωρίς περιορισμούς, όπου οι χρήστες εισάγουν το ερώτημά τους ως σύνολο λέξεων-κλειδιών σε ένα πεδίο αναζήτησης (απλή αναζήτηση) είτε με κατευθυνόμενο και δομημένο τρόπο όπου οι χρήστες καθορίζουν τις επιθυμητές τιμές των χαρακτηριστικών στοιχείων που αναζητούν (αναζήτηση με παραμέτρους). Η απλή αναζήτηση είναι συνήθως αρκετή για τους περισσότερους χρήστες, αλλά οι έμπειροι χρήστες προτιμούν να καθορίσουν ακριβώς την αναζήτησή τους και ωφελούνται από τη χρήση προηγμένης αναζήτησης. Ανεξάρτητα από τον προσφερόμενο μηχανισμό αναζήτησης, προτείνεται να παρέχονται οδηγίες για τη βελτίωση των αναζητήσεων και διατύπωσης ερωτημάτων [Vp09], ενώ σημαντικό σημείο είναι ότι ο σχεδιασμός της παρουσίασης της πληροφορίας πρέπει να γίνεται για την ανάδειξη των δεδομένων/αποτελεσμάτων και όχι για την ανάδειξη της σχεδίασης αυτής καθ' ευατής [Pr07]. Επίσης, οι διεπαφές αναζήτησης πρέπει να είναι σχεδιασμένες ώστε να υποβοηθούν τον εύκολο ανασχεδιασμό των ερωτημάτων, παρέχοντας προτάσεις από λίστες σχετικών αποτελεσμάτων, ενώ το βάθος του μονοπατιού αναζήτησης πρέπει να είναι εύληπτο. Στις περιπτώσεις που το κοινό διαθέτει αυξημένη εμπειρία, θα πρέπει να υποστηρίζονται εργαλεία προχωρημένης αναζήτησης και φιλτραρίσματος που να παρέχουν τη δυνατότητα εύκολου περιορισμού των αποτελεσμάτων, όπως αναζήτηση όρων προσανατολισμένη σε συγκεκριμένα πεδία (domain specific), πολύπλευρη αναζήτηση και επιλογή εύρους αποτελεσμάτων. Αυξημένο ρόλο στην κατανόηση των αποτελεσμάτων κειμένου, αποτελούν οι οπτικές επισκοπήσεις και προεπισκοπήσεις (overviews/previews).

Σημαντικό στοιχείο που πρέπει να ληφθεί υπόψη είναι το γεγονός ότι κατά τη διάρκεια της αναζήτησης, η αρχική πληροφοριακή ανάγκη των χρηστών αλλάζει μέσω της τυχαίας ανακάλυψης (serendipity), όπου η πληροφορία εμφανίζεται μέσω μιας τυχαίας εύρεσης, κάποιας απροσδόκητης διορατικότητας ή μίας κατά τύχη ανακάλυψης (information journey model), σε αντίθεση με τις πληροφορίες που προκύπτουν από τις εν γνώσει και συνειδητές δραστηριότητες των χρηστών. Οι πληροφοριακές ανάγκες του χρήστη δεν είναι στατικές

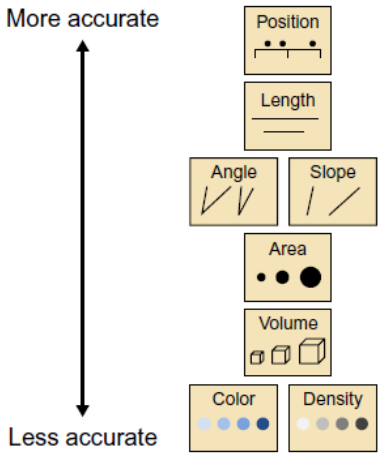
αλλά δυναμική και συνεχώς εξελισσόμενες, κάτι που τονίζει τη σημασία της ανατροφοδοτούμενης αναζήτησης μέσω της συλλογής πληροφοριών (information foraging).

Ανεξαρτήτως εμπειρίας, πρέπει να λαμβάνονται υπόψη οι δυνατότητες και οι περιορισμοί που άπτονται της ανθρώπινης αντίληψης. Γενικά, ο ανθρώπινος εγκέφαλος λαμβάνει πιο εύκολα τις οπτικές πληροφορίες αν αυτές παρουσιάζονται με συγκεκριμένους τρόπους, ειδικά όταν εμπεριέχονται ποσοτικές πληροφορίες, μοτίβα ή πληροφορίες που να πρέπει να διαφέρουν από το φόντο. Σε αυτές τις περιπτώσεις πρέπει να χρησιμοποιούνται τα λεγόμενα προνοητικά χαρακτηριστικά (preattentive attributes).



Εικόνα 9: Η οπτική αντίληψη βάσει προνοητικών χαρακτηριστικών [PT12].

Επίσης, τα όρια της βραχυπρόθεσμης μνήμης που μπορεί να συγκρατήσει το ανθρώπινο μυαλό είναι συγκεκριμένα, καθώς σε αυτήν γίνονται οι ενέργειες της ανάλυσης και της νοηματοδότησης, και υπάρχει όριο για την ποσότητα πληροφορίας που μπορεί να συγκρατηθεί σε κάθε δεδομένη στιγμή. Ακόμα, συγκεκριμένα μοτίβα είναι περισσότερο αποτελεσματικά για την παρουσίαση της πληροφορίας ανά περίπτωση.



Εικόνα 10: Η ακρίβεια της ποσοτικής αντίληψης με τη χρήση διαφορετικών δεικτών [RT12].

Η διδιάστατη θέση και το μήκος είναι δύο αποτελεσματικοί δείκτες ποσοτικής αξίας, οπότε και χρησιμοποιούνται περισσότερο για την παρουσίαση αριθμητικών δεδομένων, ενώ, η

χρήση χρώματος δεν είναι καλός δείκτης για τον σκοπό αυτόν και είναι καλύτερα να χρησιμοποιείται στην κατηγοριοποίηση των δεδομένων [RT12].

Τα παραπάνω, άπτονται του αντικειμένου της διεπαφής χρήστη (User Interface) και της χρηστικότητα (usability) μέσω της οποίας διεκπεραιώνεται η αναζήτηση. Ενδεικτικά, η διεπαφή χρήστη προτείνεται να στοχεύει στα παρακάτω σημεία [MIAT+18]:

- Την ευκολία εκμάθησης εφαρμογής (learnability), δηλαδή τον βαθμό ευκολίας που κάποιος νέος χρήστης δύναται να ολοκληρώσει μια διεργασία μέσω της διεπαφής.
- Την αποτελεσματικότητα (efficiency), δηλαδή την ταχύτητα που οι χρήστες ολοκληρώνουν τις διεργασίες, αφού κατανοήσουν τη λειτουργία της διεπαφής.
- Τον βαθμό ευκολίας απομνημόνευσης της διαδικασίας (memorability), ο οποίος σχετίζεται με την ικανότητα του χρήστη να χρησιμοποιεί σε ικανοποιητικό βαθμό τη διεπαφή, μετά από αρκετό διάστημα μη χρήσης της.
- Την ελαχιστοποίηση του συνόλου των λαθών (errors), στοιχείο σύμφωνα με το οποίο υποδηλώνεται πόσο επιτυχής είναι η διεπαφή, μέσω του πλήθους, του τύπου, της συχνότητας, καθώς και του αν οι χρήστες κατάφεραν να ολοκληρώσουν τις διεργασίες τους παρά τα λάθη που προέκυψαν.
- Την ικανοποίηση του χρήστη (User Satisfaction), η οποία εξαρτάται από τα προαναφερθέντα χαρακτηριστικά.

Συνδυαζόμενο με τα παραπάνω, είναι ότι ο σχεδιασμός της παρουσίασης της πληροφορίας επηρεάζει τη συνολική αφοσίωση χρήστη (User Engagement), η οποία αποτελεί βασική έννοια στον σχεδιασμό διαδικτυακών εφαρμογών. Η αφοσίωση χρήστη αναφέρεται στη συναισθηματική, γνωστική και συμπεριφορική σύνδεση που υπάρχει, ανά πάσα στιγμή και πιθανώς κατά την πάροδο του χρόνου, μεταξύ ενός χρήστη και ενός πόρου. Στον παρακάτω πίνακα παρουσιάζονται τα χαρακτηριστικά που επηρεάζουν το μέγεθος αυτό [AKLP11].

Χαρακτηριστικό	**Ορισμός**
Εστίαση προσοχής (Focused Attention)	Αποκλεισμός άλλων πραγμάτων και ενεργειών κατά τη διάρκεια της αλληλεπίδρασης με το σύστημα
Θετική επίδραση (Positive Affect)	Τα συναισθήματα που βιώθηκαν κατά τη διάρκεια της αλληλεπίδρασης με το σύστημα
Αισθητική (Aesthetics)	Το πόσο γοητευμένος είναι ο χρήστης από την αισθητική της διεπαφής
Ανθετικότητα (Endurability)	Η πιθανότητα εντύπωσης της εμπειρίας στη μνήμη του χρήστη και η προθυμία για επανάληψη ή συστασή της
Καινοτομία (Novelty)	Η παροχή νέων, άγνωστων ή απροσδόκητων εμπειριών
Παροχές και έλεγχος (Richness and Control)	Η πληρότητα της διεπαφής και του ελέγχου της
Φήμη, εμπιστοσύνη και	Η εμπιστοσύνη που δείχνει ο χρήστης για το σύστημα

προσδοκία (Reputation, Trust and Expectation)	
Περιβάλλον χρήστη (User Context)	Τα κίνητρα και τα οφέλη του χρήστη.

Πίνακας 3: Χαρακτηριστικά αφοσίωσης χρήστη [AKLP11].

Ένα καλοσχεδιασμένο σύστημα αποτελεί κύριο παράγοντα για την αφοσίωση του χρήστη και η χρηστικότητα έχει άμεση και θετική σχέση με τον βαθμό εμπιστοσύνης και ικανοποίησης των χρηστών (User Satisfaction) [FGG06], ενώ σε αντίθετη περίπτωση μπορεί να υπάρξει το φαινόμενο της εγκατάληψης χρήσης του συστήματος μετά την πρώτη επίσκεψη (bounce rate) [Google]. Η εντύπωση που δημιουργεί η εμφάνιση ενός συστήματος φαίνεται να συσχετίζεται με τη γενική εντύπωση που έχει ο χρήστης για την ποιότητά του και να επηρεάζει τον βαθμό ικανοποίησης από τη χρήση του [MIAT+18].

Η ικανοποίηση του χρήστη αποτελεί σημαντικό παράγοντα για την αξιολόγηση των συστημάτων ανάκτησης πληροφορίας. Οι γενικοί παράγοντες που επηρεάζουν το μέγεθος αυτό είναι α) η αποτελεσματικότητα του συστήματος, β) η αποτελεσματικότητα του χρήστη, γ) η προσπάθεια που καταβάλλει ο χρήστης, και δ) τα χαρακτηριστικά του χρήστη. Από τη σκοπιά του εξεταζόμενου συστήματος, η ικανοποίηση επηρεάζεται άμεσα από τον χρόνο που απαιτείται για την εύρεση των πληροφοριών που αναζητούνται, καθώς όσο λιγότερος χρόνος αφιερώνεται στην αναζήτηση, τόσο μεγαλύτερη είναι η ικανοποίηση, ενώ όσο περισσότερα αποτελέσματα χρειάζεται να εξεταστούν για τον εντοπισμό της σχετικής πληροφορίας τόσο χαμηλότερη είναι η ικανοποίηση [MS10].

2.3.1 Σχεδίαση όψεων αναζήτησης

Στα συστήματα ανάκτησης πληροφορίας, σημαντική θέση έχει ο τρόπος διαδικασίας αναζήτησης. Κυρίαρχη έννοια εδώ, είναι αυτή της ευχέρειας πρόσβασης (affordance), σύμφωνα με την οποία ο σχεδιασμός ενός αντικειμένου πρέπει να παρέχει πληροφορία για τον τρόπο που λειτουργεί. Την ίδια λογική πρέπει να ακολουθούν οι φόρμες αναζήτησης υποδηλώνοντας τη λειτουργικότητά τους σε κάθε αλληλεπίδραση. Τον πιο σημαντικό ρόλο στη φόρμα αναζήτησης έχει ο τρόπος εισαγωγής του ερωτήματος των όρων αναζήτησης, με προτεινόμενες προτάσεις βελτιώσης τα παρακάτω [RT12]:

- Το πεδίο αναζήτησης όρων πρέπει να είναι αρκετά μεγάλο ώστε να είναι εμφανή όλα τα στοιχεία του ερωτήματος, αλλά ταυτόχρονα να ενθαρρύνεται η επαρκής διατύπωση των πληροφοριακών αναγκών.
- Η χρήση ενός ενδεικτικού κειμένου (placeholder), για βελτίωση της φιλικότητας προς τους χρήστες.

- Αναζήτηση εύρους (scoped search). Η επιλογή από κατηγορίες βελτιώνει τη στόχευση πιο σχετικών αποτελεσμάτων, ειδικά για τους έμπειρους χρήστες.
- Αναζήτηση σε αποτελέσματα (search within). Η δυνατότητα εισαγωγής όρων αναζήτησης σε δεύτερο χρόνο σε προϋπάρχοντα αποτελέσματα, μέσω πεδίου πολύπλευρης αναζήτησης. Σημαντική είναι η αποφυγή μηδενικών αποτελεσμάτων.
- Αυτόματη συμπλήρωση (autocomplete). Η αναγνώριση της πληροφορίας κατά την ανάκλησή της συμβάλλει στην ευκολότερη κατανόησή της όταν έχει συναντηθεί κατά το παρελθόν και μπορεί να ανακληθεί από τη μνήμη του χρήστη. Έτσι, το πρόβλημα ανάκλησης πληροφορίας μετατρέπεται σε θέμα αναγνώρισής της.
- Αυτόματες προτάσεις (autosuggest). Με τον τρόπο αυτόν παρέχονται επιλογές από μία λίστα που δεν έχει τέτοιους περιορισμούς προτείνοντας νέες ιδέες στον χρήστη.
- Άμεσα αποτελέσματα (instant results). Πολλές φορές η παροχή άμεσων αποτελεσμάτων με την επιλογή από συγκεκριμένο σετ όρων παρέχει το βέλτιστο πλήθος αποτελεσμάτων με άμεση απόκριση και αποφεύγοντας ορθογραφικά λάθη.
- Προτάσεις διόρθωσης (did you mean). Συμβάλλει στην αντιμετώπιση εισαγωγής ανορθόγραφων όρων, μέσω της χρήσης αλγορίθμων ορθογραφικού ελέγχου για τη σύγκριση ερωτημάτων με τις διάφορες σωστά ορθογραφικές μορφές της κάθε λέξης.
- Αυτόματη διόρθωση (autocorrect). Σε αυτές τις περιπτώσεις δεν είναι γνωστό εκ των προτέρων η πληροφοριακή ανάγκη του χρήστη αλλά είναι σίγουρα γνωστό το τι δεν θέλει, οπότε η επιλογή αυτή βοηθάει στην καλύτερη απόκριση του συστήματος.
- Μερικό ταίριασμα (partial match). Σε περιπτώσεις που δεν επιστρέφονται αποτελέσματα δεν έχει νόημα η αντικατάσταση ολόκληρου του ερωτήματος, αλλά η παροχή των αποτελεσμάτων που βασίζεται σε μερικό ταίριασμα των όρων.
- Σχετικές αναζητήσεις (related searches). Όπως και η λειτουργία της αυτόματης πρότασης, οι σχετικές αναζητήσεις παρέχουν στον χρήστη νέες ιδέες, ενώ αποσαφηνίζουν τυχόν διφορούμενα ερωτήματα.

Εκτός από τον τρόπο εισαγωγής των όρων αναζήτησης στα αντίστοιχα πεδία, εξίσου σημαντικός είναι και ο τρόπος παρουσίασης του συνόλου των επιστρεφόμενων αποτελεσμάτων, καθώς ο συνδυασμός τους αποτελεί τον πυρήνα της διαδικασίας αναζήτησης. Η παρουσίαση των αποτελεσμάτων δεν πρέπει να είναι ούτε πολύ λεπτομερής σπαταλώντας πολύτιμο χώρο παρουσίασης αλλά ούτε και πολύ σύντομη, καθώς έτσι ελλοχεύει ο κίνδυνος για την παράλειψη ουσιώδης πληροφορίας [RT12].

Όταν υπάρχουν πάρα πολλές διαστάσεις ή το περιβάλλον εργασίας χρήστη είναι πολύ μικρό για την εμφάνιση όλων, τότε πρέπει να εμφανίζονται μόνο ορισμένες από αυτές, κάτι που απαιτεί την κατάταξη των διαστάσεων, έτσι ώστε να επιλέγονται οι πιο σημαντικές αναλόγως των συνθηκών [WLZZ+13], [ST09]. Επίσης, πρέπει να παρέχονται οι περισσότερο

δημοφιλείς τιμές των διαστάσεων περιορίζοντας τις τιμές με χαμηλή εμφάνιση, καθώς υπάρχει η παραδοχή ότι οι πιο συχνές τιμές είναι περισσότερο χρήσιμες [NH14].

Σημαντική περιοχή παρουσίασης πληροφορίας θεωρείται το επάνω μέρος της οθόνης (above the fold) το οποίο είναι ορατό πριν την κύλιση του ποντικιού (scrolling). Τρεις από τις πιο σημαντικές μηχανές αναζήτησης (Google, Bing και Yahoo) ακολουθούν παρόμοιο τρόπο παρουσίασης, αφού από προεπιλογή για κάθε αποτέλεσμα εμφανίζεται ο τίτλος της σελίδας (page title), η διαδικτυακή διεύθυνση (URL) και το κείμενο σύνοψης (snippet), στο οποίο επισημαίνονται οι όροι αναζήτησης που ταιριάζουν με το κείμενο του αποτελέσματος. Οι κανόνες εμφάνισης των αποτελεσμάτων πρέπει να σέβονται τους διαφορετικούς κανόνες του κάθε πλαισίου αναζήτησης, όπως για παράδειγμα ότι στο διαδικτυακό εμπόριο πρέπει να εμφανίζονται οι φωτογραφίες των αποτελεσμάτων, στα άρθρα των ενημερωτικών ιστοσελίδων πρέπει να είναι εμφανής η ημερομηνία δημοσίευσης κ.λπ. [RT12]. Η σύνοψη συνήθως είναι η επικεφαλίδα ή οι πρώτες γραμμές του εγγράφου, ενώ πολλές φορές ελέγχεται και το μέτρο πυκνότητας των λέξεων ερωτήματος (density measure). Για τα αποτελέσματα αναζητήσεων που προέρχονται από διαφημίσεις ακολουθούνται διαφορετικές τεχνικές, όπως η χρήση διαφορετικών βάσεων δεδομένων στις οποίες περιέχεται η σύντομη περιγραφή του αποτελέσματος και του υπερσυνδέσμου σε αυτά. Έπειτα, μέσα από μία διαδικασία δημοπρασιών (bid) ανά όρο αναζήτησης και άλλων παραγόντων επιστρέφονται οι αντίστοιχες περιγραφές των αποτελεσμάτων [CMS15].

2.3.2 Διάταξη (layout)

Ένα από τα ζητήματα στον σχεδιασμό πολύπλευρης αναζήτησης είναι η επιλογή της θέσης του μενού αναζήτησης, με τις επιλογές να είναι κάθετη, οριζόντια και υβριδική. Η κάθετη διάταξη είναι η πιο συνηθισμένη και εξυπηρετεί την κλιμάκωση των επιλογών των όψεων στη δομή της ιστοσελίδας και τη συνοχή με τα παραγώμενα αποτελέσματα, ενισχύοντας τη σχέση μεταξύ των επιλογών και των αποτελεσμάτων. Στην οριζόντια διάταξη, οι επιλογές των όψεων διατάσσονται οριζόντια, συνήθως στο επάνω μέρος της ιστοσελίδας, και καταλαμβάνουν μια πιο κυρίαρχη θέση σε αυτήν, ενθαρρύνοντας την αλληλεπίδραση του χρήστη με τη φόρμα αναζήτησης. Ωστόσο, η διαμόρφωση αυτή δεν κλιμακώνεται σε μεγάλο βαθμό, λόγω του περιορισμού του πλάτους της σελίδας. Επιπλέον, το μενού δεν είναι πλέον ορατό κατά την κύλιση προς τα κάτω, διακυβεύοντας έτσι την ευκολία συσχέτισης μεταξύ των επιλογών και των επιστρεφόμενων αποτελεσμάτων. Τέλος, στο υβριδικό μοντέλο χρησιμοποιείται η κάθετη διάταξη στην αριστερή πλευρά για την πλειονότητα των όψεων, ενώ παράλληλα, επιλογές όψεων εμφανίζονται στο επάνω μέρος της οθόνης.

Σχετικό με το ζήτημα της διάταξης είναι η επιλογή της προεπιλεγμένης κατάστασης για καθεμία από τις όψεις. Οι επιλογές είναι τρεις: α) κλειστό από προεπιλογή (closed by default), β) ανοιχτό από προεπιλογή (open by default) ή γ) ένας συνδυασμός των δύο.

Στην πρώτη επιλογή όλες οι όψεις είναι κλειστές εξοικονομώντας χώρο οθόνης. Το μειονέκτημα αυτής της προσέγγισης είναι ότι η κάθε όψη υποδηλώνει σε μικρότερο βαθμό την πληροφορία από ότι αν εμφανιζόταν σε ανοιχτή κατάσταση.

Στη δεύτερη επιλογή οι όψεις εμφανίζονται ανοιχτές μεγιστοποιώντας την παρουσίαση της πληροφορίας και ενθαρρύνντας τη χρήση του πολύπλευρου μενού αναζήτησης. Συνήθως περιορίζεται ο αριθμός των εμφανών τιμών ανά όψη και στη συνέχεια ο χρήστης παροτρύνεται (call to action) για την εμφάνιση περισσότερων τιμών (π.χ. “Περισσότερα...”).

Η τρίτη επιλογή είναι ο συνδυασμός των δύο παραπάνω καταστάσεων η οποία γίνεται όλο και πιο δημοφιλής, καθώς εξυπηρετεί την αποτελεσματική χρήση του χώρου της οθόνης και παρέχει περισσότερες πληροφορίες για τις πρώτες όψεις, οι οποίες ιδανικά πρέπει να έχουν μεγαλύτερη προτεραιότητα [RT12].

2.3.3 Μορφές εμφάνισης (Display formats)

Όπως αναφέρθηκε προηγουμένως, με τη χρήση διαστάσεων μπορεί να ταξινομηθεί ένα αντικείμενο. Κάθε μία από τις διαστάσεις αυτές βασίζεται σε έναν τύπο δεδομένων, όπως οι ημερομηνίες ως ακέραιοι, τα ονόματα των κατασκευαστών ως κείμενο κ.λπ., κάτι που διαμορφώνει την αρχιτεκτονική των εφαρμογών πολύπλευρης αναζήτησης. Η εμφάνιση των όψεων στους τελικούς χρήστες και η αλληλεπίδραση με αυτές, πρέπει να γίνεται βάσει της αρχής της επικοινωνίας της φύσης των δεδομένων αντιστοιχίζοντας τη μορφή εμφάνισης με τη σημασιολογία των τιμών της όψης. Η ακατάλληλη χρήση των εκάστοτε επιλογών μπορεί να δημιουργήσει ασυνέπειες στην εμπειρία χρήστη. Παρακάτω, εξετάζονται οι κυριότερες κατηγορίες των επιλογών των όψεων [RT12].

Οι υπερσύνδεσμοι είναι αντιπροσωπεύουν τιμές κειμένου και παρέχουν απλή και άμεση αλληλεπίδραση μέσω της επιλογής τους (κλικ), ενώ συνδυαζόμενοι με τον αριθμό των επιστρεφόμενων αποτελεσμάτων (record counts / aggregations) παρέχουν αποτελεσματική περίληψη του πληροφοριακού χώρου. Από σύμβαση, οι υπερσύνδεσμοι χρησιμοποιούνται για την εμφάνιση όψεων μονής επιλογής.

Η χρήση πεδίων ελέγχου (checkbox) γίνεται κυρίως στην εμφάνιση όψεων πολλών επιλογών.

Στα παραπάνω, οι τιμές των όψεων θεωρούνται ποιοτικής φύσης. Σε περιπτώσεις ποσοτικών δεδομένων, π.χ. εύρος τιμών, χρησιμοποιούνται κλίμακες εύρους τιμών (range slider) όπου δίνεται βάση στη μέγιστη ή στην ελάχιστη τιμή της όψης ή και στον συνδυασμό των δύο. Μειονέκτημα είναι η απουσία ακριβούς ελέγχου των τιμών, αφού οι τιμές μπορούν να επεκταθούν σε μεγάλα διαστήματα, οπότε οι κλίμακες συνδυάζονται με πεδία κειμένου.

Οι ετικέτες συνάθροισης (tag clouds) οπτικοποιούν κατηγορίες κειμένου, ενώ η χρήση τους επεκτάθηκε για την συμπερίληψη αδόμητου περιεχομένου παρουσιάζοντας με τη μορφή ετικεών τους όρους που εξάγονται από τα έγγραφα.

Πολύ σημαντική είναι επίσης η κατανόηση μοτίβων σε ένα υψηλότερο επίπεδο με στόχο τη διευκόλυνση της ανάλυσης των δεδομένων και την παροχή διορατικότητας ως προς την πληροφορία, κάτι που επιτυγχάνεται με την οπτικοποίηση δεδομένων (data visualization), όπου οι όψεις παρέχουν άμεση επισκόπηση της κατανομής του συνόλου για κάθε διάσταση. Σημαντική υποκατηγορία είναι η οπτικοποίηση γεωχωρικών δεδομένων όπου παρουσιάζονται μοτίβα στην κατανομή των εγγράφων και παρέχονται δυνατότητες διερεύνησης των σχέσεων μεταξύ των όψεων και των κατανομών στον χάρτη.

Σημαντική λεπτομέρεια είναι η επικοινωνία της τρέχουσας τοποθεσίας και των εκάστοτε επιλογών πλοήγησης. Μία απλή τεχνική είναι η χρήση ίχνων πλοήγησης (breadcrumbs) που δείχνουν την τρέχουσα θέση του χρήστη σε μια ιεραρχία πληροφοριών ή ταξινομία, όπως π.χ. “Αρχική σελίδα > Κύρια κατηγορία > Δευτερεύουσα κατηγορία”. Η κλιμάκωση σε πολλές διαστάσεις μπορεί να γίνει με την εμφάνιση όλων των επιλογών όψης στο δικό τους τμήμα ίχνους (breadbox). Οι επιλογές ομαδοποιούνται σε ένα μέρος, το οποίο παραμένει ορατό ανεξάρτητα από τον αριθμό των όψεων. Μειονέκτημα είναι ότι η σχέση μεταξύ των όψεων και των επιλεγμένων τιμών είναι λιγότερο εμφανής και οι χρήστες πρέπει ή να θυμούνται ποια όψη συσχετίστηκε με κάθε επιλεγμένη τιμή ή να μπορούν να το συμπεραίνουν

Πολύ σημαντικό κομμάτι της πολύπλευρης αναζήτησης αποτελεί η διαδραστική συμπεριφορά των όψεων, αφού σε κάθε ανανέωση των επιλογών υπάρχει χρονική καθυστέρηση στην εμφάνιση των αποτελεσμάτων. Οι επιλογές σχεδιασμού στους τρόπους ανταπόκρισης και ανανέωσης μετά τις επιλογές των χρηστών έχουν σημαντική επιρροή στη συνολική εμπειρία χρήστη, αφού οι λανθασμένες επιλογές μπορούν να κάνουν την εφαρμογή να φαίνεται ασύνδετη και να αυξήσουν την πιθανότητα μηδενικών αποτελεσμάτων. Η πρώτη προσέγγιση στοχεύει στην άμεση ανταπόκριση του συστήματος κατά την ανανέωση των επιλογών (instant model), έτσι ώστε το σύνολο των αποτελεσμάτων και οι διαθέσιμες τιμές όψης να παραμένουν συνεπείς και να αντικατοπτρίζουν τα αποτελέσματα που είναι διαθέσιμα στον χρήστη ανά πάσα στιγμή. Το μειονέκτημα αυτής της προσέγγισης είναι ότι υπάρχει συνεχή ανανέωση της σελίδας, ενώ υπάρχουν περιπτώσεις που να υπάρχει απαίτηση για συνδυασμό δύο τιμών (π.χ. εύρος τιμής). Εναλλακτικά, στο μοντέλο δύο φάσεων (two-stage), οι χρήστες επιλέγουν τις τιμές και υποβάλουν τη φόρμα αναζήτησης στο σύνολό της. Το πλεονέκτημα είναι ότι οι χρήστες μπορούν να επιλέξουν όσες τιμές χρειάζεται χωρίς διακοπή και τα αποτελέσματα να εμφανιστούν όταν είναι έτοιμα, ενώ το μειονέκτημα είναι ότι οι χρήστες μπορούν να οδηγηθούν σε μηδενικά αποτελέσματα. Λύση στο παραπάνω πρόβλημα δίνει ο αμοιβαίος αποκλεισμός των υπόλοιπων όψεων όταν μία όψη έχει λάβει κάποια τιμή.

Επίσης, κατά την ανανέωση των αποτελεσμάτων είναι καλή πρακτική η ύπαρξη ενδιάμεσης κατάσταση που να είναι εμφανής διαδικασία ανάκτησης των αποτελεσμάτων, ώστε να είναι κατανοητή η αποσύνδεση μεταξύ των δύο συνόλων αποτελεσμάτων.

Η λειτουργία της προεπισκόπησης (preview) δίνει λύση στο πρόβλημα επιστροφής στη σελίδα αποτελεσμάτων εξαιτίας έλλιπούς ενημέρωσης (pogo sticking), κάτι που είναι χρονοβόρο και δημιουργεί όχληση.

Στον υπολογισμό της συνάφειας των παρεχόμενων αποτελεσμάτων βοηθάει η ανατροφοδότηση μέσω της βαθμολόγησης των αποτελεσμάτων ("+1 button"), ενώ πολύ σημαντική είναι η λειτουργία εξατομικευμένων αποτελεσμάτων βάσει προηγούμενων αναζητήσεων του χρήστη (π.χ. session, cookies).

Μετά τον καθορισμό των όρων και χαρακτηριστικών αναζήτησης, τα αποτελέσματα εμφανίζονται στις σελίδες αποτελεσμάτων (search engine results pages - SERP), οι οποίες παίζουν καθοριστικό ρόλο στην εμπειρία αναζήτησης.

Οι δύο πιο κλασσικοί τρόποι εμφάνισης είναι η μορφή λίστας (list) και πλέγματος (grid). Στη λίστα τα αποτελέσματα εμφανίζονται σε κατακόρυφη κατάταξη, ενώ στο πλέγμα κάθε αποτέλεσμα θεωρείται ένα κελί και το σύνολό τους παρουσιάζεται σε έναν διδιάστατο νοητό πίνακα. Η λίστα καταλαμβάνει λιγότερο χώρο, οπότε θεωρείται καλύτερη για αποτελέσματα που περιέχουν κείμενο επιτρέποντας την από πάνω προς τα κάτω αποτελεσματική σάρωση. Το πλέγμα προτείνεται για συνοπτική παρουσίαση αποτελεσμάτων που περιέχουν φωτογραφίες, επιτυγχάνοντας αποτελεσματική χρήση του χώρου προσφέροντας οπτική αρμονία. Σε κάθε περίπτωση, θεωρείται καλή πρακτική η παροχή και των δύο επιλογών. Μία ακόμα επιλογή παρουσίασης αποτελεσμάτων είναι η εμφάνιση λεπτομερειών ανά κάθετη στήλη, αφού με αυτόν τον τρόπο επιτυγχάνονται οι διαδικασίες της ανάλυσης και της σύγκρισης ομοειδών αποτελεσμάτων.

Σε περιπτώσεις γεωχωρικών δεδομένων προς εμφάνιση, θεωρείται αποδοτική μέθοδος η χρήση διδιάστατου χάρτη, κάτι που μπορεί να επαυξήσει την εμπειρία του χρήστη μέσω ομάδες αποτελεσμάτων (clusters).

Στις περιπτώσεις μηδενικών αποτελεσμάτων πρέπει να δίνονται επιλογές αναδιαμόρφωσης των κριτηρίων, όπως "did u mean" και αυτόματης διόρθωσης με παράλληλη εμφάνιση άλλων επιλογών πλοήγησης, όπως κορυφαίες αναζητήσεις, επιλεγμένα αποτελέσματα κ.ά.

Ο χειρισμός των αποτελεσμάτων επιτυγχάνεται με λειτουργίες όπως της σελιδοποίησης (pagination), της ταξινόμησης (sorting), του φιλτραρίσματος (filtering), της σύγκρισης (comparison) των αποτελεσμάτων, καθώς και της αποσαφήνισης του ερωτήματος.

Τέλος, τα πιο συναφή αποτελέσματα μπορούν να αποθηκευτούν για περαιτέρω εξέταση και μελλοντική διερεύνηση. Αυτή η λειτουργικότητα είναι βασική για το διαδικτυακό εμπόριο

όπου οι χρήστες είναι απαραίτητο να μπορούν εύκολα να προβούν στην ανάλυση και τη σύγκριση των προϊόντων για την εύρεση της καλύτερης επιλογής. Συνηθισμένη τεχνική είναι η εμφάνιση σε στήλες της πλήρης λεπτομέρειας όλων των αποθηκευμένων προς σύγκριση αντικειμένων, επιτρέποντας την εύκολη σύγκριση των αντικειμένων[RT12].

Εν κατακλείδι, η αναζήτηση πληροφορίας είναι μία επαναληπτική διαδικασία που απαιτεί τη δημιουργία, σύνθεση και ανασχηματισμό των ερωτημάτων, ενώ τα στοιχεία αναζήτησης πρέπει να ακολουθούν τις αρχές της προσιτότητας και της αλληλεπιδραστικότητας.

3

Περιγραφή δεδομένων, ανάκτηση και παρουσίαση της πληροφορίας

Σε αυτό το κεφάλαιο περιγράφεται σε ανώτερο επίπεδο το θέμα που πραγματεύεται η διπλωματική. Πιο συγκεκριμένα, περιγράφεται η φύση των δεδομένων, η προσέγγιση που ακολουθήθηκε για την ανάκτηση της πληροφορίας, καθώς και ο τρόπος παρουσίασής της στον τελικό χρήστη.

3.1 Ορισμός του προβλήματος

Τα διαθέσιμα δεδομένα της εφαρμογής, αφορούν αγγελίες ποικίλου περιεχομένου, κατηγοριών και ελεύθερου κειμένου. Στην τρέχουσα μορφή καθιστούσαν ελλιπή τον απλό τρόπο αναζήτησης και παρουσίασης πληροφορίας. Η χρήση παραδοσιακού συστήματος βάσης δεδομένων δεν καθιστούσε δυνατή τη σύνθετη αναζήτηση τόσο στο κείμενο όσο και κατά τον συνδυασμό με άλλες επιλογές φίλτρων. Ακόμα, δεν ήταν εύκολη η αναζήτηση με ορατά κριτήρια που να μπορούσαν να χρησιμοποιήσουν οι επισκέπτες για τη βελτίωση των ερωτημάτων αναζήτησης και των κατηγοριών τους, ενώ υπήρχε έλλειψη και στην κατανόηση των ανακτηθέντων αποτελεσμάτων.

Για τους παραπάνω λόγους, κρίθηκε απαραίτητη η χρήση ενός διαφορετικού συστήματος ανάκτησης πληροφορίας, το οποίο να έχει δυνατότητα ενσωμάτωσης πολλαπλών φίλτρων κατά την αναζήτηση και να υποστηρίζει παράλληλα την αναζήτηση πληροφορίας σε

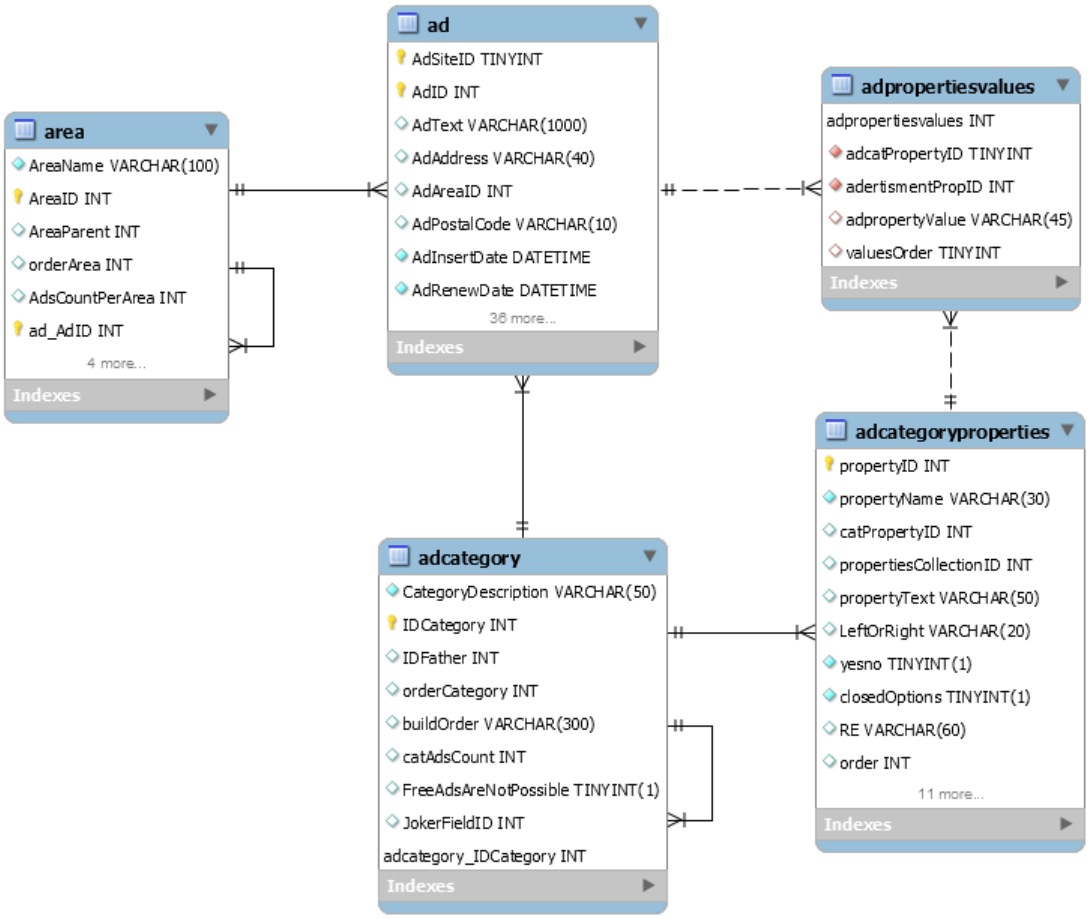
ημιδομημένο κείμενο με χρηση facets. Έτσι, υλοποιήθηκε web-based σύστημα αναζήτησης αγγελιών με φίλτρα και facets, με την ενσωμάτωση ενός μοντέρνου συστήματος ανάκτησης πληροφορίας που να παρέχει ικανή αποτελεσματικότητα στην αναζήτηση, και την παράλληλη εστίαση του συνόλου της εφαρμογής στην ευκολία χρήσης.

3.2 Περιγραφή βάσης δεδομένων

Το αρχικό σύνολο δεδομένων προήλθε από μία έτοιμη δοκιμαστική βάση δεδομένων αγγελιών, η οποία αποτελούνταν από 75.578 αγγελίες ποικίλου περιεχομένου. Το σχήμα της βάσης που δόθηκε, ήταν ορισμένο σε MySQL και αποτελούνταν από 11 βασικούς πίνακες, με τους κυριότερους από αυτούς να είναι οι παρακάτω:

- ad: Το σύνολο των αγγελιών με τα βασικά χαρακτηριστικά τους (κείμενο, περιοχή, κατηγορία κ.λπ.).
- area: Το σύνολο των περιοχών των αγγελιών με δυνατότητα ορισμού απεριόριστου βάθους υποπεριοχών.
- adcategory: Το σύνολο των κατηγοριών των αγγελιών με δυνατότητα ορισμού απεριόριστου βάθους υποκατηγοριών.
- adcategoryproperties: Το σύνολο των ιδιοτήτων οι οποίες μπορούν να υπάρχουν στις αγγελίες και που στην ουσία πρόκειται για τις διαστάσεις. Τα πιο βασικά χαρακτηριστικά αυτού του πίνακα είναι το πεδίο “yesno” που δείχνει αν είναι Boolean τύπου και το πεδίο “closedOptions” που δείχνει αν είναι κλειστού τύπου. Σημαντικό επίσης είναι το πεδίο “catPropertyID” που ως ξένο κλειδί συσχετίζει την κάθε ιδιότητα με την κατηγορία στην οποία ανήκει η αγγελία.
- adpropertiesvalues: Ο πίνακας που περιέχει την αντιστοίχιση των αγγελιών, των ιδιοτήτων και των τιμών τους.

Σο παρακάτω διάγραμμα Οντοτήτων-Συσχετίσεων (Entity Relationship – ER) παρουσιάζονται συνοπτικά οι κυριότεροι πίνακες της βάσης, καθώς και οι μεταξύ τους συσχετίσεις.



Εικόνα 11: Μοντέλο Οντοτήτων – Συσχετίσεων των κυριότερων πινάκων της βάσης δεδομένων.

Ενδεικτικά, μία αγγελία ανήκει σε μία περιοχή και μία κατηγορία, οι οποίες μπορεί να είναι αντικείμενα-παιδιά μίας άλλης περιοχής/κατηγορίας σε απροσδιόριστο βάθος. Η αγγελία μπορεί να έχει ορισμένες ιδιότητες που με τη σειρά τους ανήκουν σε μία κατηγορία και οι οποίες παίρνουν συγκεκριμένες τιμές.

3.3 Θέματα προς επίλυση

Κατά τη διάρκεια υλοποίησης της διπλωματικής έγινε στόχευση σε συγκεκριμένα θέματα.

Πρώτα από όλα, σε επίπεδο αρχιτεκτονικής, έπρεπε να γίνει έρευνα των συστημάτων ανάκτησης πληροφορίας που θα δέχονταν με σχετικά εύκολο τρόπο την ενσωμάτωση των δεδομένων της βάσης, αλλά και να επιδέχονται παραμετροποίηση μέσω κάποιας γλώσσας προγραμματισμού υψηλού επιπέδου. Σημαντικό σημείο ήταν ότι η εφαρμογή δεν θα έπρεπε να βασίζεται μόνο σε σύνολο στατικών δεδομένων αλλά θα έπρεπε να παρέχει δυνατότητες συνεχούς προσθήκης, διαγραφής και τροποποίησης των υπάρχοντων αλλά και νέων δεδομένων. Επίσης, καίριας σημασίας ήταν επίσης, η επιλογή των κατάλληλων πεδίων για τη σωστή αντιστοίχιση/χαρτογράφηση (mapping), καθώς και την εύκολη και γρήγορη αντιγραφή (migration) των υπάρχοντων δεδομένων στο σύστημα ανάκτησης πληροφορίας.

Σε επίπεδο εφαρμογής, υπήρχε πληθώρα λεπτομερειών που έπρεπε να ληφθούν υπόψη, όπως:

- Η ενοποίηση και ομογενοποίηση του τρόπου αναζήτησης πληροφορίας της βάσης δεδομένων και του συστήματος ανάκτησης πληροφορίας, με τρόπο διάφανο (transparent) που να παρέχει σε κάθε περίπτωση την κατ' ελάχιστον απαραίτητη πληροφορία.
- Ο τρόπος σύνθεσης του τελικού ερωτήματος για την αναζήτηση, ώστε κάθε φίλτρο και επιλογή να λειτουργεί με αγνωστικιστικό (agnostic) τρόπο και ανεξάρτητα από τις υπόλοιπες επιλογές και ρυθμίσεις, παράγοντας πάντα τα σωστά αποτελέσματα.
- Η αυτοματοποίηση της διαδικασίας χαρτογράφησης των facets και η εισαγωγή δεδομένων στα αντίστοιχα πεδία με σωστό και διαχειρίσιμο τρόπο.
- Η αυτοματοποίηση διαδικασιών παραγωγής συναθροίσεων (aggregetations) ανεξαρτήτως επιλογών.
- Η διασφάλιση της γρήγορης απόκρισης του συστήματος.
- Η αυτοματοποίηση της ανάκτησης των επιπέδων βάθους των στοιχείων γονέα-παιδί (parent-child), τόσο σε πρωτογενές επίπεδο όσο και ανά περίπτωση (ad-hoc).
- Ο διαχωρισμός του συστήματος διαχειριστή και απλού χρήστη (backend/frontend).

Σε επίπεδο παρουσίασης, ήταν σημαντικό να γίνει έρευνα σχετικά με το ποια από τα δεδομένα θα λειτουργούσαν ως φίλτρα και με τι επιλογές, ενώ επίσης έπρεπε να ληφθούν αποφάσεις σχετικά με τον τρόπο εμφάνισης στον χρήστη τόσο των φίλτρων όσο και των αποτελεσμάτων, ώστε η πληροφορία να είναι εύληπτη και κατανοητή.

4

Υλοποίηση συστήματος αναζήτησης αγγελιών

Στο κεφάλαιο αυτό περιγράφεται η υλοποίηση των σημαντικότερων σημείων της εφαρμογής, οι λόγοι που οδήγησαν στη λήψη της εκάστοτε απόφασης, καθώς και τα προβλήματα που ενέκυψαν κατά τη διάρκεια υλοποίησης και που έπρεπε να λυθούν για την επιτυχή ολοκλήρωση της εφαρμογής.

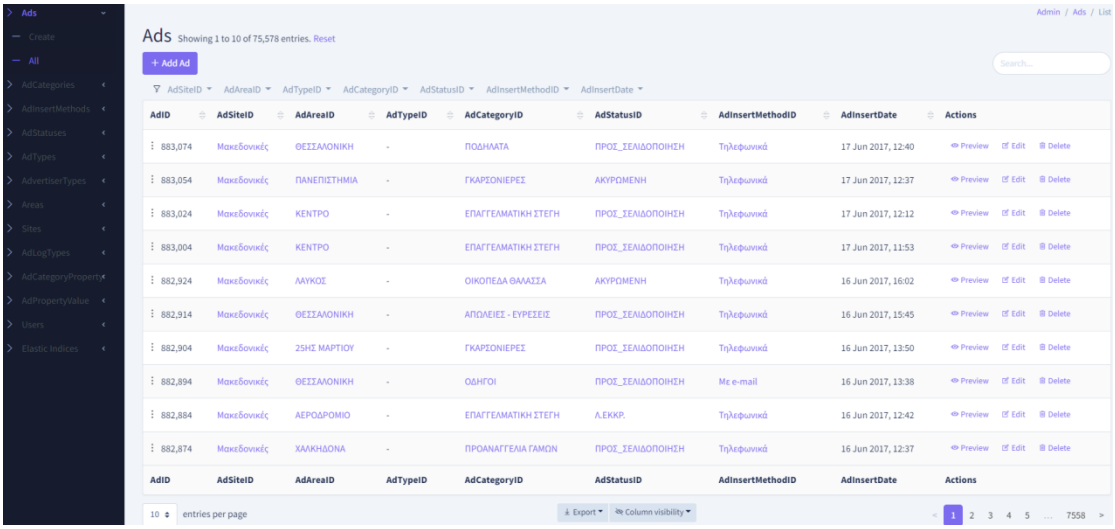
4.1 Προετοιμασία συστήματος

Μετά την επισκόπηση της διαθέσιμης βάσης δεδομένων, έπρεπε να αποφασιστεί ο τρόπος και ο μηχανισμός που να καθιστούσε δυνατή την ανάκτηση πληροφορίας με όσο το δυνατόν περισσότερες λειτουργίες περιγράφηκαν στη βιβλιογραφία (αναζήτηση ελεύθερου κειμένου, υποστήριξη δυαδικών τελεστών, ταξινόμηση βάσει βαθμολογίας συνάφειας κ.ά.), με συνδυασμό την υποστήριξη αναζήτησης με facets. Κατόπιν σχετικής έρευνας, επιλέχθηκε το σύστημα Elasticsearch, το οποίο και παρέχει όλες τις παραπάνω δυνατότητες. Ως γλώσσα προγραμματισμού επιλέχτηκε η PHP: Hypertext Preprocessor (PHP) για την οποία υπάρχει επίσημη βιβλιοθήκη ενσωμάτωσης από το Elasticsearch. Επίσης, χρησιμοποιήθηκε και το σύνολο συσχετιζόμενων βιβλιοθηκών (framework) Laravel. Η μεθολογία που ακολουθήθηκε ήταν αυτή του Μοντέλου – Όψης – Ελεγκτή (Model – View – Contoller, MVC).

Μετά την εγκατάσταση των απαραίτητων προγραμμάτων και εξαρτήσεών τους, δημιουργήθηκαν όλες οι κλάσεις μοντέλων που αντιστοιχούν στους πίνακες της βάσης δεδομένων, καθώς και όλες οι συσχετίσεις τους. Η ενέργεια αυτή θεωρείται πολύ σημαντική,

καθώς στα μοντέλα ενθυλακώνεται όλη η λογική των απαραίτητων λειτουργιών που μπορούν να εφαρμοστούν στα δεδομένα.

Στο σημείο αυτό, δημιουργήθηκε ξεχωριστό διαχειριστικό backend τμήμα της εφαρμογής, στο οποίο παρέχονται όλες οι λειτουργίες Προσθήκης/Ανάγνωσης/Τροποποίησης/Διαγραφής (Create/Read/Update/Delete – CRUD) για όλα τα βασικά μοντέλα της εφαρμογής, οπότε με τον τρόπο αυτόν, η εφαρμογή έγινε πλήρως δυναμική στο σύνολό της.



Εικόνα 12: Το διαχειριστικό τμήμα της εφαρμογής (admin panel).

Επίσης, δημιουργήθηκε νέος πίνακας (elastic_indices) και το αντίστοιχο μοντέλο (ElasticIndex) στο οποία ενσωματώθηκαν γενικές λειτουργικότητες που παρέχει το Elasticsearch, βάσει των παραμέτρων που υπάρχουν σε κάθε κλάση του μοντέλου που υλοποιεί την αντίστοιχη διασύνδεση.

4.2 Ευρετηρίαση πληροφορίας

Το επόμενο βήμα ήταν ο καθορισμός της απαραίτητης για ευρετηρίαση πληροφορίας στο σύστημα αναζήτησης.

Για να καταστούν τα δεδομένα εφαρμόσιμα και συνεκτικά, τα έγγραφα και τα χαρακτηριστικά έπρεπε να αντιστοιχιστούν με τα ισοδύναμα της βάσης δεδομένων. Μετά από την ανάλυση των δεδομένων, έγινε η επιλογή των παρακάτω στοιχείων τα οποία χρησιμοποιούνται σε διάφορες λειτουργίες ανάκτησης πληροφορίας:

- Κείμενο αγγελίας (text)
- Περιοχή (keyword)
- Κατηγορία (keyword)
- Status (keyword)
- Πηγή προέλευσης (keyword)

- Τύπος καταγραφής (keyword)
- Τύπος κατηγορίας (keyword)
- Ημερομηνία ανάρτησης (date)
- Δείκτης φωτογραφιών (boolean)
- Αριθμός αστεριών (keyword)
- Επίπεδο 1 Περιοχής (keyword)
- Επίπεδο 2 Περιοχής (keyword)
- Επίπεδο 1 Κατηγορίας (keyword)
- Επίπεδο 2 Κατηγορίας (keyword)
- Προέλευση ανάρτησης (keyword)
- Facets (nested)

Ο ορισμός του τύπου κάθε πεδίου έγινε μετά από επισκόπηση της βάσης δεδομένων και τις προτεινόμενες επιλογές που υπάρχουν στην τεκμηρίωση του Elasticsearch. Πιο συγκεκριμένα, ο τύπος keyword χρησιμοποιείται για δομημένο περιεχόμενο όπως αναγνωριστικά, διευθύνσεις email, κωδικούς κατάστασης κ.λπ.

Η ευρετηριοποίηση του κειμένου της αγγελίας, έγινε με τη χρήση ελληνικού stemmer, τη μετατροπή όλων των χαρακτήρων σε πεζούς, τη χρήση των ελληνικών stopwords, ενώ έγινε και η απαλοιφή ετικετών HTML.

```
'analyzer' => [
    'rebuilt_greek' => [
        'type' => 'custom',
        'tokenizer' => 'standard',
        'char_filter' => [
            'html_strip',
        ],
        'filter' => [
            'greek_lowercase',
            'greek_stop',
            'greek_stemmer',
            'english_stemmer',
        ],
    ],
],
```

Για τον τύπο κατηγορίας, έγινε λεκτική ανάλυση του περιεχομένου συγκεκριμένου πεδίου της βάσης δεδομένων και λήφθηκαν μόνο οι πιο συχνές έγκυρες τιμές του (π.χ. πώληση, ενοικίαση, ζήτηση κ.λπ.). Έπειτα, δημιουργήθηκε μία γενική κατηγορία για παρεμφερείς τιμές και έγινε η αντιστοίχιση με τιμές της βάσης δεδομένων, με αποτέλεσμα τη χρήση διακριτών κατηγοριών.

Η εύρεση των τιμών των πεδίων επιπέδων περιοχής και κατηγορίας ήταν διαφορετική από την απλή ανάγνωση μίας τιμής. Ο λόγος ήταν ότι το αντίστοιχο πεδίο στη βάση δεδομένων περιείχε το προσδιοριστικό του τελικού βάθους και το προσδιοριστικό του γονέα του. Έτσι,

δεν ήταν δυνατόν να είναι γνωστή εκ των προτέρων η τιμή που θα έπρεπε να ευρετηριοποιηθεί και η οποία θα χρησιμοποιούνταν κατά τη διαδικασία ανάκτησης δεδομένων. Ο ίδιος περιορισμός υπήρχε τόσο κατά την παρουσίαση των φίλτρων όσο και των αποτελεσμάτων, οπότε η λύση έπρεπε να είναι ενιαία για όλες τις περιπτώσεις και για όλα τα μοντέλα. Για τον λόγο αυτόν, δημιουργήθηκε ένα PHP Trait (TopParentTrait) στο οποίο προστέθηκαν λειτουργίες αναδρομικής αναζήτησης βάσει προσδιοριστικού, ώστε να είναι δυνατή η εύρεση των στοιχείων-γονέα από ένα δεδομένο στοιχείο-παιδί. Με αυτόν τον τρόπο, υποστηρίζεται η ύπαρξη φίλτρων με δυναμικό τρόπο για οποιοδήποτε βάθος στοιχείων. Ταυτόχρονα, με τη λύση αυτή, αποφεύγεται η επανάληψη κώδικα σε κάθε κλάση που παρουσιάζει την ίδια δομή στοιχείων γονέων-παιδιών.

```
public function findLevelsFromId($id)
{
    $parent = optional(static::find($id))->parent;
    if (!empty($parent)
        && $parent->{static::PRIMARY_KEY} !=
        static::$topParentModelsConfig['topParentIdValue']
    ) {
        if (!in_array($id, $this->levels)) {
            $this->levels[] = $id;
        }
        $this->levels[] = $parent->{static::PRIMARY_KEY};
        static::findLevelsFromId($parent->{static::PRIMARY_KEY});
    } else {
            return $this->levels;
    }
}
```

Οι τιμές των πεδίων δείκτη φωτογραφιών και αριθμού αστεριών, έγινε με χρήση εικονικών ιδιοτήτων (virtual attributes), ώστε να φανεί η ύπαρξη δυνατοτήτων ευρετηριοποίησης τιμών οι οποίες δεν αντιστοιχούν σε αμιγώς δεδομένα της βάσης αλλά υπολογίζονται κατά τη διάρκεια εκτέλεσης της εφαρμογής.

Ο τύπος που χρησιμοποιήθηκε για τα facets (nested) παρουσιάζει ιδιαίτερο ενδιαφέρον. Όπως γράφτηκε και παραπάνω, οι τιμές των ιδιοτήτων/διαστάσεων δεν είναι προκαθορισμένες εκ των προτέρων με κάποιον συγκεκριμένο τρόπο. Βρίσκονται όλες στον πίνακα “adcategoryproperties”, ο οποίος δύναται να εμπλουτίζεται συνεχώς με νέες τιμές. Για παράδειγμα, δεν υπήρχαν προκαθορισμένα μόνο τρία facets (Χρώμα, Μάρκα, Μέγεθος) στα οποία θα ήταν εύκολο να γίνει η αντιστοίχιση των τιμών που λάμβανε κάθε αγγελία. Έτσι, έπρεπε να βρεθεί ένας τρόπος που να επιτρέπει την αντιστοίχιση οποιουδήποτε πλήθους ιδιοτήτων για κάθε αγγελία τόσο κατά την ευρετηρίασης όσο και κατά την ανάκτηση και παρουσίαση της πληροφορίας. Η λύση που επιλέχτηκε ήταν με τη χρήση του ένθετου (nested) τύπου δεδομένων, ως εξής:

```
'facets' => [
    'type' => 'nested',
```

```
    'properties' => [
        'facet_name' => [
            'type' => 'keyword',
        ],
        'facet_value' => [
            'type' => 'keyword',
        ],
    ],
],
```

Με τον παραπάνω τρόπο ορίζονται δύο ιδιότητες μέσα στην ιδιότητα “facets”, η “facet_name” και η “facet_value”, με κάθε μία να είναι τύπου keyword. Έτσι, αναλόγως των εκάστοτε διαθέσιμων δεδομένων, η εισαγωγή των στοιχείων γίνεται δυναμικά:

```
Αγγελία 1:
{
    "facets": [
        {"facet_name": "property_id_1", "facet_value": "true"},
        {"facet_name": "property_id_2", "facet_value": "true"},
        {"facet_name": "property_id_3", "facet_value": "true"},
        {"facet_name": "property_id_4", "facet_value": "true"}
    ]
}

Αγγελία 2:
{
    "facets": [
        {"facet_name": "property_id_1", "facet_value": "true"},
        {"facet_name": "property_2", "facet_value": "true"}
    ]
}
```

Η λύση αυτή επεκτείνεται και για τιμές κειμένου, καθώς και για αριθμούς.

Σημειώνεται ότι χωρίς τον ορισμό του τύπου ως ένθετο, τότε η εισαγωγή σε αντίστοιχες περιπτώσεις θα γίνονταν ως εξής:

```
{
    "facets_string.facet_name": [
        "χρώμα", "μέγεθος", "μάρκα"
    ],
    "facets_string.facet_value": [
        "μπλε", "xl", "όνομα_μάρκας_1"
    ]
}
```

Με τον παραπάνω τρόπο θα υπήρχε η απώλεια συσχέτισης της πληροφορίας για το κάθε facet με τα υπόλοιπα, οπότε και δεν θα ήταν δυνατή η χρήση φίλτρου για χρώμα "μπλε" και μέγεθος "xl", καθώς η αποθήκευση των δεδομένων θα γίνονταν με τη μορφή επίπεδου πίνακα. Σε αντίθεση, με τον ένθετο τύπο, διατηρείται η ανεξαρτησία των αντικειμένων και η συσχέτιση του συνόλου της πληροφορίας. Εδώ σημειώνεται, ότι η αντιστοίχιση των facets γίνεται κατά την εισαγωγή των αρχικών δεδομένων, καθώς η αναγνώριση νέων facets από

την ανάλυση του ελεύθερου κείμενου αποτελούσε αντικείμενο εκτός του πεδίου της συγκεκριμένης διπλωματικής εργασίας.

Μετά τον καθορισμό των Elasticsearch properties, έπρεπε να ληφθεί απόφαση για τον τρόπο δημιουργίας του ευρετηρίου στο Elasticsearch. Πιο συγκεκριμένα, υπήρχαν οι επιλογές να είναι σημειακής ενημέρωσης (one off) ή δυναμικού περιεχομένου. Η πρώτη επιλογή ήταν πιο εύκολη στην υλοποίηση, καθώς η ευρετηρίαση θα γίνονταν μόνο μία φορά αλλά θα ήταν στατικό, ενώ η δεύτερη επιλογή ήταν περισσότερο σύνθετη αλλά με περισσότερα πλεονεκτήματα, αφού θα υπήρχαν λειτουργίες για τη δυναμική ενημέρωσή του. Τελικά, επιλέχθηκε η δεύτερη προσέγγιση, για την οποία έπρεπε να υλοποιηθούν διάφορες λειτουργίες.

Η αρχική ευρετηριοποίηση για το σύνολο του migration από τη βάση δεδομένων της MySQL υλοποιήθηκε με τη λειτουργία bulk του Elasticsearch, όπου για κάθε property έγινε δυναμική αντιστοίχιση με την ανάλογη μέθοδο στο μοντέλο.

```
foreach (self::getInstance()->getFieldIndexMapping() as $key => $value) {
    if (!is_array($value)) {
        $bulkMapping[$key] = $model->{$value};
    } else {
        $bulkMapping[$key] = $model->{$value['function']}();
    }
}
static::$client->bulk($params);
```

Η δυναμική προσθήκη νέων στοιχείων στο ευρετήριο υλοποιήθηκε μέσω του backend του αντίστοιχου μοντέλου με τη λειτουργία προσθήκης νέων στοιχείων από τον διαχειριστή. Επίσης, για λόγους πληρότητας, αποφασίστηκε η ενσωμάτωση λειτουργίας με κάποια χαρακτηριστικά crawler, λειτουργικότητα που δύναται να διατρέχει ανά συγκεκριμένα χρονικά διαστήματα (cron job) τη σελίδα αποτελεσμάτων και μέσω ανάλυσης του HTML κώδικα (parsing) να αποθηκεύει τόσο στη βάση δεδομένων όσο και στο ευρετήριο τα νέα αποτελέσματα (data feeding). Αυτή η λειτουργία, αν και στην προκειμένη περίπτωση υλοποιήθηκε ενδεικτικά, θεωρείται πολύ σημαντική για την τροφοδότηση νέου περιεχομένου σε πλήθος εφαρμογών.

4.3 Διαχωρισμός αρχιτεκτονικής

Σημαντική απόφαση που έπρεπε να ληθφεί, ήταν ο τρόπος υλοποίησης και ενσωμάτωσης των λειτουργιών του Elasticsearch στο μοντέλο των αγγελιών. Η πρώτη προσέγγιση ήταν όλες οι αντίστοιχοι μέθοδοι να υλοποιηθούν απευθείας στο συγκεκριμένο μοντέλο. Αν και για τα δεδομένα της διπλωματικής εργασίας αυτό θα ήταν αρκετό, παρόλα αυτά κάτι τέτοιο θα ήταν περιοριστικό, καθώς δεν θα υπήρχε η δυνατότητα προσθήκης νέων ευρετηρίων για κανένα άλλο μοντέλο σε περίπτωση που κάτι τέτοιο θεωρούνταν απαραίτητο. Για τον λόγο αυτόν,

μελετήθηκε μια δεύτερη προσέγγιση, αυτή της κληρονομικότητας λειτουργιών. Σημαντικός περιορισμός στην PHP θεωρείται η δυνατότητα κληρονομικότητας μόνο από μία κλάση, όμως κάτι τέτοιο ξεπεράστηκε με τη δημιουργία ενός PHP Trait (ElasticSearchTrait), στο οποίο ενθυλακώθηκαν όλες τις βασικές λειτουργίες του Elasticsearch και με το οποίο πλέον είναι δυνατή η "προσκόλληση" της συγκεκριμένης λειτουργικότητας σε όποιο μοντέλο έχει τα αντίστοιχα απαιτούμενα χαρακτηριστικά. Με αυτόν τον τρόπο, το εκάστοτε μοντέλο διατηρεί όλα τα πλεονεκτήματα του προτύπου Ενεργής Εγγραφής (Active Record) και επιπλέον ενσωματώνει την απαραίτητη λειτουργικότητα του Elasticsearch μέσω του αντίστοιχου client που παρέχεται από την επίσημη βιβλιοθήκη.

```
public static function getElasticSearchClient()
{
    if (empty(static::$client)) {
        static::$client = ClientBuilder::create()
            ->setHosts(config('elasticsearch.hosts'))
            ->build();
    }
    return static::$client;
}
```

Με τον παραπάνω τρόπο, ελέγχεται αν η Singleton attribute έχει οριστεί κατά τη διάρκεια της εκτέλεσης. Αν κάτι τέτοιο ισχύει, τότε επιστρέφεται το στιγμιότυπο που υπάρχει, ενώ σε διαφορετική περίπτωση πρώτα γίνεται κλήση του αντίστοιχου client, δημιουργείται το στιγμιότυπο και έπειτα αυτό επιστρέφεται. Μέσω του συγκεκριμένου στιγμιοτύπου γίνονται όλες οι λειτουργίες ανάκτησης δεδομένων από το Elasticsearch.

Στη συνέχεια, έπρεπε να σχεδιαστεί η αρχιτεκτονική της επικοινωνίας μεταξύ της διεπαφής χρήστη και του συστήματος ανάκτησης δεδομένων. Αρχικά, θεωρήθηκε αρκετή η απευθείας επικοινωνία με το Elasticsearch κατά την αναζήτηση πληροφορίας. Μετά όμως από σκέψη, η προηγούμενη άποψη αναθεωρήθηκε, καθώς το σύστημα έπρεπε να είναι επεκτάσιμο και με ανοχή σε σφάλματα. Αυτό σημαίνει ότι η σχεδίαση της αρχιτεκτονικής έπρεπε να γίνει κατά τέτοιον τρόπο που να παρέχεται πρόσβαση στα δεδομένα ακόμα και αν το ευρετήριο του Elasticsearch δεν ήταν διαθέσιμο για κάποιο χρονικό διάστημα ή αν στο μέλλον έπρεπε να προστεθούν και άλλα συστήματα και τρόποι ανάκτησης της πληροφορίας αναλόγως των παραμέτρων ανά περίπτωση.

Εξετάστηκαν δύο επιλογές, εκ των οποίων η πρώτη ήταν η υλοποίηση ενός MicroService και η δεύτερη ήταν η υλοποίηση ενός Interface. Το πρώτο κρίθηκε αρκετά πολύπλοκο για τον σκοπό της εργασίας αυτής, οπότε επιλέχτηκε η δεύτερη επιλογή. Σύμφωνα με αυτόν τον τρόπο, δημιουργήθηκε ένα Interface (IRInterface), το οποίο και έπρεπε να είναι υλοποιήσιμο από κάθε τρόπο ανάκτησης πληροφορίας (π.χ. Elasticsearch, MySQL κ.λπ.). Σε αυτό, καθορίστηκαν οι μέθοδοι που πρέπει κατ' ελάχιστον να υποστηρίζονται από οποιοδήποτε σύστημα πρόκειται να αλληλεπιδράσει με τη βάση δεδομένων / ευρετήριο. Για παράδειγμα,

το μοντέλο των αγγελιών (Ad), σε περίπτωση που δεν υπάρχει πρόσβαση στο ευρετήριο του Elasticsearch, θέτει ως ελάχιστη υπηρεσία (fallback) την ανάκτηση πληροφορίας από τη βάση δεδομένων. Στη συνέχεια, δημιουργήθηκε η κλάση ElasticSearch, στην οποία σημαντικό σημείο της αρχιτεκτονικής είναι η στιγμή προσπέλασής της:

```
public function __construct($model){
    $this->model = $model;
    static::getElasticSearchClient();
}
```

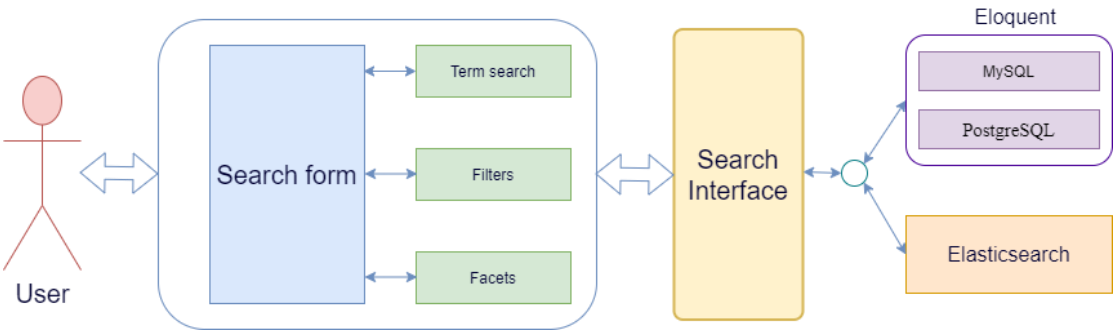
Με τον τρόπο αυτόν γίνεται η σύνθεση όλων των λειτουργιών του μοντέλου ActiveRecord με τις λειτουργίες του Elasticsearch, του οποίου πλέον η κλάση έχει πρόσβαση σε όλες τις ιδιότητες του μοντέλου από το οποίο έγινε η κλήση.

```
Ad::getSearchEngineInstance($request->input('searchMode'));
```

Η κλήση αυτή, θα καλέσει αυτόματα την μέθοδο του Trait, ώστε η απόφαση για την υλοποίηση που θα ακολουθηθεί θα γίνει σε πραγματικό χρόνο, σύμφωνα με τις επιλογές του χρήστη και το status διαθεσιμότητας του ευρετηρίου.

```
public static function getSearchEngineInstance(?string $searchMode = null)
{
    self::setSearchMode($searchMode);
    if (self::getSearchMode() != static::SEARCH_MODE_ELASTIC) {
        return self::getInstance();
    } else {
        return new ElasticSearch(self::getInstance());
    }
}
```

Έτσι, μπορεί να γίνει η κλήση οποιασδήποτε υπηρεσίας με αγνωστικιστικό τρόπο αφού κάθε φορά γίνεται η κλήση όποιας έχει οριστεί κατά τη διάρκεια της εκτέλεσης αναλόγως διαθεσιμότητας και χωρίς να είναι γνωστή εκ των προτέρων ποια υλοποίηση θα ακολουθηθεί.



Εικόνα 13: Η αρχιτεκτονική του συστήματος αναζήτησης.

Η παραπάνω λειτουργικότητα επαυξάνεται ακόμα περισσότερο από την διάφανη παρασκηνιακή υλοποίηση που υλοποιείται από το Laravel για τη διασύνδεση με τη βάση δεδομένων (Eloquent) και λειτουργεί επίσης με αγνωστικιστική προσέγγιση για τους τύπους των βάσεων δεδομένων (MySQL, MariaDB, PostgreSQL κ.ά.).

4.4 Διαδικασία αναζήτησης

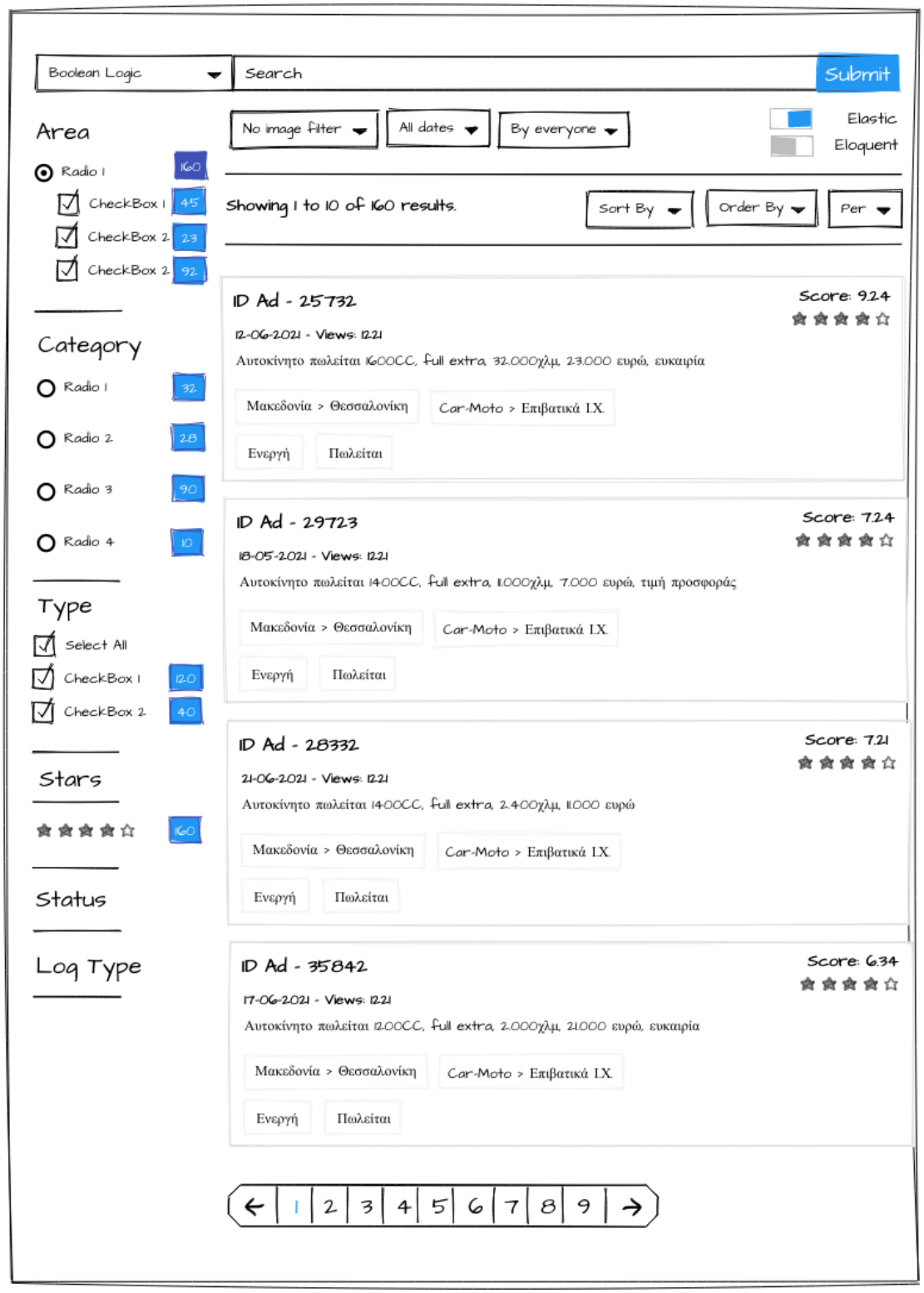
Με την εισαγωγή των παραμέτρων στη φόρμα αναζήτησης, γίνεται η αντίστοιχη κλήση προς την κατάλληλη υλοποιημένη διεπαφή, βρίσκονται τα αποτελέσματα που ταιριάζουν στις πληροφοριακές ανάγκες του χρήστη και επιστρέφονται βάσει βαθμολογίας συνάφειας (score). Η βαθμολογία συνάφειας βασίζεται στη διαδικασία αναζήτησης ελεύθερου κειμένου βάσει εγγύτητας περιεχομένου και επιλεγμένων τελεστών αναζήτησης (AND, OR) και είναι ένας θετικός αριθμός κυμαινόμενου σημείου (float) που επιστρέφεται ως μεταδεδομένο στο πεδίο "_score" του Elasticsearch API. Όσο υψηλότερη είναι η βαθμολογία αυτή τόσο πιο σχετικό είναι το έγγραφο [Elastic]. Τα υπόλοιπα φίλτρα και facets μειώνουν το σύνολο των τελικών αποτελεσμάτων.

Προϋπόθεση για την εμφάνιση των facets είναι η επιλογή περιοχής, κατηγορίας και υποκατηγορίας. Έπειτα, ο χρήστης επιλέγει τις επιθυμητές τιμές, γίνεται αναζήτηση στο Elasticsearch και για τον συνδυσμό των επιλογών που έχουν γίνει εμφανίζονται τα facets που αντιστοιχούν. Με κάθε νέα υποβολή της φόρμας γίνεται η αντίστοιχη τροποποίηση των αποτελεσμάτων και των νέων διαθέσιμων επιλογών που παρέχονται.

Για κάθε επιλογή, δημιουργούνται οι κατάλληλες συναθροίσεις, ώστε να παρέχεται απευθείας στον τελικό χρήστη το πλήθος που παίρνουν οι διακριτές τιμές ανά πεδίο αποτελεσμάτων. Αυτό επιτυγχάνεται με τις συναθροίσεις δοχείων (bucket aggregations), οι οποίες, σε αντίθεση με τις συναθροίσεις μετρήσεων που απλά υπολογίζουν μετρικές στα πεδία, δημιουργούν σύνολα (buckets) εγγράφων. Κάθε bucket συσχετίζεται με ένα κριτήριο από το οποίο καθορίζεται αν κάποιο έγγραφο εμπίπτει στα σωστά αποτελέσματα ή όχι. Επίσης, με αυτόν τον τρόπο υποστηρίζονται και οι υποσυναθροίσεις βάσει των συνόλων εγγράφων που υπάρχουν σε υπερσυναθροίσεις.

Η ανάκτηση των εγγράφων γίνεται από το Elasticsearch και για κάθε αποτέλεσμα επιστρέφεται η τιμή του κύριου κλειδιού που έχει οριστεί στο αντίστοιχο ευρετήριο μαζί με το score. Έπειτα, οι υπόλοιπες τιμές της κάθε εγγραφής αναζητούνται από τη σχεσιακή βάση δεδομένων. Για τη βελτιστοποίηση της απόκρισης του συστήματος, τα αποτελέσματα σελιδοποιούνται και γίνεται η αναζήτηση μόνο εκείνων των αποτελεσμάτων που βρίσκονται στο εύρος της κάθε σελίδας.

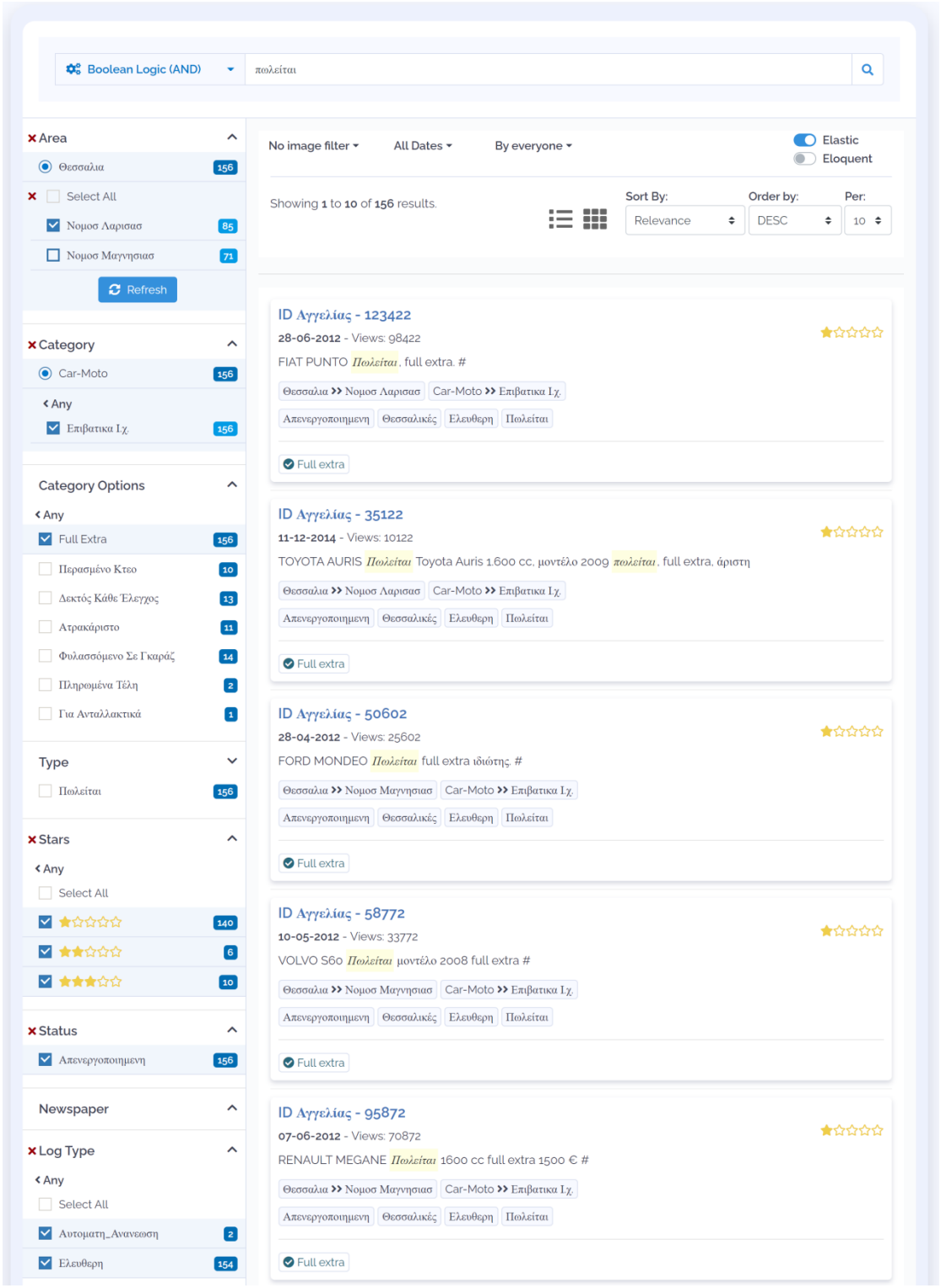
Για τη σχεδίαση της διεπαφής χρήστη, χρησιμοποιήθηκε εργαλείο προτυποποίησης όψεων (wireframing), ώστε να δοκιμαστούν διάφορες επιλογές έως ότου βρεθεί αυτή η οποία κρίθηκε ότι καλύπτει καλύτερα τις απαιτήσεις χρήστη, αλλά και τις προτεινόμενες από τη βιβλιογραφία πρακτικές. Επίσης, με τη χρήση wireframing καθίσταται πιο εύκολη η μετάβαση της σχεδίασης στο προγραμματιστικό περιβάλλον.



Εικόνα 14: Το wireframe του συστήματος αναζήτησης.

Η επιλογή των πεδίων αναζήτησης και του τρόπου εμφάνισης κάθε μίας από αυτές, έγινε σύμφωνα με τις καλές πρακτικές που παρουσιάστηκαν σε προηγούμενο κεφάλαιο της βιβλιογραφικής επισκόπησης. Η διεπαφή χρήστη έγινε με προσαρμοστική προσέγγιση, προσαρμόζοντας δυναμικά τα στοιχεία ελέγχου στις εκάστοτε επιλογές του χρήστη, χρησιμοποιώντας την προσέγγιση πλέγματος (grid). Σε κάθε επανάληψη της αναζήτησης

εμφανίζονται οι υποκατηγορίες και οι συναθροίσεις των αποτελεσμάτων παρέχοντας καθοδήγηση στον χρήστη για τις επιλογές εμβάθυνσης με σκοπό την επίτευξη καλύτερων αποτελεσμάτων. Ιδιαίτερη προσοχή λήφθηκε στην αποφυγή συνδυασμών παραγωγής μηδενικών αποτελεσμάτων. Έτσι, κατά τις διάφορες επιλογές του χρήστη εμφανίζονται μόνο αυτές που είναι ενεργές και οδηγούν στην παραγωγή νέων αποτελεσμάτων. Στο σημείο αυτό σημειώνονται οι δύο επιλογές που υπήρχαν για την επιλογή των facets. Στην περίπτωση της προσέγγισης υποστήριξης λειτουργίας πολλαπλών επιλογών με συνδυαστική λειτουργία μέσω μίας μόνο αλληλεπίδρασης, τότε υπήρχε η πιθανότητα παραγωγής μηδενικών αποτελεσμάτων. Στις περιπτώσεις που η προσέγγιση είναι η αυτόματη ανανέωση της ιστοσελίδας μετά από κάθε επιλογή ή η υποστήριξη μόνο μίας επιλογής τιμής facets, τότε δεν θα υπήρχε τέτοια περίπτωση. Σημειώνεται ότι δοκιμάστηκαν και οι δύο επιλογές.



Εικόνα 15: Η διαδικασία αναζήτησης με χρήση facets

Στην παραπάνω εικόνα γίνεται η τελική επισκόπηση του συνόλου των επιλογών της σελίδας αναζήτησης, όπως τα δεδομένα προέρχονται από το σύστημα ανάκτησης πληροφορίας.

5

Τεχνολογίες που χρησιμοποιήθηκαν

Στην ενότητα αυτήν κρίνεται σκόπιμο να παρουσιαστούν περισσότερες λεπτομέρειες σχετικά με το σύστημα ανάκτησης δεδομένων που χρησιμοποιήθηκε, καθώς και με το επιλεγμένο framework PHP. Επίσης, περιγράφεται η εγκατάσταση των απαραίτητων προγραμμάτων και οι απαραίτητες ρυθμίσεις.

5.1 Elasticsearch

Ως βασικό σύστημα ανάκτησης πληροφορίας, χρησιμοποιήθηκε το Elasticsearch, το οποίο είναι βασισμένο στο Apache Lucene [Elastic]. Πρόκειται για μία κατανεμημένη, RESTful μηχανή αναζήτησης και ανάλυσης κειμένου η οποία είναι ικανή για πάρα πολύ γρήγορη αναζήτηση βασισμένη στη συνάφεια των αποτελεσμάτων που συσχετίζονται με τους όρους αναζήτησης βάσει λεκτικής ανάλυσης. Υποστηρίζει δομημένο και αδόμητο κείμενο, αριθμητικά και γεωχωρικά δεδομένα, ένθετους τύπους δεδομένων κ.ά., ενώ η αποθήκευση και ευρετηρίαση των εγγράφων γίνεται με τέτοιον τρόπο ώστε να είναι πολύ γρήγορη, ενώ η κατανεμημένη αρχιτεκτονική του συστήματος το καθιστά πλήρως κλιμακώσιμο. Πέρα από την απλή αναζήτηση, υποστηρίζονται διαδικασίες συνάθροισης πληροφορίας, μηχανικής μάθησης για ανακάλυψη τάσεων και μοτίβων σε δεδομένα, ανάλυση μετρικών κ.ά.

Η αποθήκευση της πληροφορίας γίνεται με τη χρήση σύνθετων δομών δεδομένων που έχουν πρώτα σειριοποιηθεί ως JSON. Αυτό έρχεται σε αντίθεση με τα συστήματα διαχείρισης βάσεων δεδομένων όπου η πληροφορία αποθηκεύεται ως εγγραφές με στήλες.

Το Elasticsearch έχει σχήμα (schema), το οποίο ονομάζεται χαρτογράφηση (mapping) σύμφωνα με το οποίο περιγράφεται ένα ευρετήριο το οποίο θεωρείται ως μια βελτιστοποιημένη συλλογή εγγράφων. Μέσω της χαρτογράφησης περιγράφονται τα πεδία και οι τύποι τους, ώστε να είναι δυνατή η διαχείριση των εγγράφων. Κάθε έγγραφο έχει ξεχωριστό αναγνωριστικό και αποθηκεύεται στο ευρετήρια που έχει προηγούμενως δημιουργηθεί και συσχετίζεται με έναν τύπο δεδομένων. Τα δεδομένα των εγγράφων είναι της μορφής κλειδί-τιμή (key: value). Για παράδειγμα, {“institute” : “International Hellenic University”}. Ένας τύπος δεδομένων αντιπροσωπεύει μία κλάση παρόμοιων εγγράφων, όπως ένα άρθρο, μία αγγελία κ.λπ., η οποία έχει τη χαρτογράφηση όλων των πεδίων του εγγράφου και τους τύπους δεδομένων τους.

Κατά την αποθήκευση ενός εγγράφου, το έγγραφο καταχωρείται στο ευρετήριο και είναι πλήρως προσπελάσιμο σχεδόν σε πραγματικό χρόνο (περίπου 1 δευτερόλεπτο) με τη χρήση της τεχνικής του ανεστραμμένου ευρετηρίου. Η ανάλυση του κειμένου εφαρμόζεται τόσο κατά τη διάρκεια της ευρετηριοποίησης όσο και κατά την αναζήτηση του εγγράφου, πράγμα που σημαίνει ότι όλοι οι όροι που τίθενται προς αναζήτηση στις διεπαφές χρήστη περνάνε από όλα τα στάδια λεκτικής ανάλυσης που είχαν περάσει και κατά τη στιγμή της ευρετηρίασης. Ο μετασχηματισός των δεδομένων γίνεται μέσω της διαδικασίας ανάλυσης (analyzer) εφαρμόζοντας τις γνωστές τεχνικές που έχουν περιγραφεί στο κεφάλαιο της ανάκτησης πληροφορίας (character filtering, tokenization, stemming κ.λπ).

Οι κύριες λειτουργίες που εφαρμόζονται σε ένα ευρετήριο είναι η προσθήκη, η διαγραφή, η ανάκτηση και η τροποποίηση ενός εγγράφου, όπως και η αναζήτηση στο σύνολο του ευρετηρίου. Η αναζήτηση αποτελείται από ένα ή περισσότερα ερωτήματα που συνδυάζονται και αποστέλλονται στο Elasticsearch και τα έγγραφα που αντιστοιχούν στα ερωτήματα επιστρέφονται ως αποτελέσματα (hits). Υποστηρίζονται δομημένα ερωτήματα, ερωτήματα ελεύθερου κειμένου, καθώς και σύνθετα ερωτήματα. Τα δομημένα ερωτήματα είναι παρόμοια με αυτά της SQL. Τα ερωτήματα ελεύθερου κειμένου επιστρέφουν ταξινομημένα βάσει σχετικότητας όλα τα έγγραφα που ταιριάζουν στους όρους αναζήτησης. Πέρα από την αναζήτηση διακριτών όρων, παρέχεται η δυνατότητα για χρήση ερωτημάτων φράσης, εγγύτητας, μπαλαντέρ (wildcard), ασαφών όρων κ.ά. Επιτρέπεται η χρήση των τελεστών δυαδικής λογικής, καθώς και μερικού ή πλήρους ταιριάσματος όρων.

Εκτός από τα αποτελέσματα που ταιριάζουν, το Elasticsearch μπορεί να υπολογίζει και να επιστρέφει τις συναθροίσεις αποτελεσμάτων, καθιστώντας έτσι δυνατή τη δημιουργία σύνθετων περιλήψεων των δεδομένων, οπτικοποιώντας μετρικές, μοτίβα και τάσεις.

Η γλώσσα ερωτημάτων που χρησιμοποιείται είναι η Γλώσσα Καθορισμένου Τομέα (Domain Specific Language – Query DSL) και βασίζεται στην Apache Lucene Query Language.

5.2 Laravel

Η εφαρμογή έχει αναπτυχθεί με τη χρήση του Laravel, το οποίο είναι ένα σύνολο συσχετιζόμενων βιβλιοθηκών (framework) ανοιχτού κώδικα για τη γλώσσα προγραμματισμού PHP. Πρόκειται για ένα στιβαρό, επεκτάσιμο και εύκολο στην κατανόηση framework, το οποίο ακολουθεί το πρότυπο σχεδίασης Μοντέλο – Όψη – Ελεγκτής (Model – View – Controller, MVC) και προσφέρει ένα πλούσιο σύνολο λειτουργιών και χαρακτηριστικών που επιταχύνουν την ανάπτυξη λογισμικού με εύκολο και κλιμακωτό τρόπο. Η λογική του βασίζεται στους περιέκτες υπηρεσίας (Service Containers) στους οποίους γίνεται η διαχείριση των εξαρτήσεων κλάσεων (class dependencies, dependency injection), στους παρόχους υπηρεσιών (Service Providers) στους οποίους γίνεται η παροχή των προηγούμενων εξαρτήσεων στα σημεία που χρειάζεται και τις προσόψεις (Facades), οι οποίες είναι στατικές διεπαφές για τη διασύνδεση με τα Service Containers. Τέλος, ο πυρήνας εφαρμόζει ένα μοτίβο Αντιστροφής της Λογικής (Inversion of Control – IoC) που επιτρέπει την προσαρμογή και επανεγγραφή οποιουδήποτε μέρους του πλαισίου (αίτημα, καταγραφή, έλεγχος ταυτότητας κ.λπ.).

Σημαντικό πλεονέκτημα είναι ότι μπορεί να αναπτυχθεί με το Docker, το οποίο είναι ένα εργαλείο που βασίζεται σε containers και επιτρέπει τη δημιουργία και την ανάπτυξη των εφαρμογών μέσω του ορισμού των εξαρτήσεων και την ανάπτυξη όλης της εφαρμογής ως ένα ενιαίο πακέτο. Επίσης, παρέχεται η δυνατότητα αρθρωτής σχεδίασης μέσω της χρήσης ενοτήτων (modules) που φορτώνονται μέσω του προγράμματος διαχείρισης κλάσεων Composer για την εισαγωγή όλων των απαραίτητων εξαρτήσεων και βιβλιοθηκών, ενώ είναι εύκολα τροποποιήσιμο μέσω εκτεταμένων ρυθμίσεων και επεκτάσιμο μέσω εύκολης ενσωμάτωσης με βιβλιοθήκες τρίτων οντοτήτων.

Το Laravel ενσωματώνει λειτουργείες δημιουργίας ερωτημάτων (Eloquent) που ενθυλακώνουν τη διάφανη λειτουργικότητα, ανεξαρτήτως βάσης δεδομένων, μέσω της χαρτογράφησης αντικειμένων και συσχετίσεων (Object Relational Mapping – ORM) και της λειτουργίας ενεργής εγγραφής (ActiveRecord), ενώ οι δυνατότητες του καθορισμού σχήματος βάσης (Schema Builder) παρέχουν τη λειτουργικότητα για τον χειρισμό του ορισμού βάσης δεδομένων. Ο χειρισμός της δρομολόγησης (Routing) των κλήσεων γίνεται μέσω μιας ευέλικτης προσέγγισης που επιτρέπει στον χρήστη τον καθορισμό των διαδρομών στην εφαρμογή, αυξάνοντας την αποδοτικότητά της. Μέσα από τη μηχανή δημιουργίας προτύπων όψεων (Blade Engine) δίνεται η δυνατότητα για τον σχεδιασμό ιεραρχικών μπλοκ και παραγωγής περιεχομένου με δυναμικό τρόπο και αποσύνδεση της επιχειρησιακής λογικής από τις όψεις, με αποτέλεσμα η βάση κώδικα να είναι διατηρήσιμη πιο εύκολα. Εκτός των άλλων, το Laravel περιλαμβάνει δυνατότητες που βοηθούν στη δοκιμή μέσω διαφόρων

δοκιμαστικών περιπτώσεων (unit testing), δυνατότητα που εξυπηρετεί στη διατήρηση του κώδικα σύμφωνα με τις προκαθορισμένες απαιτήσεις.

Περιλαμβάνονται μια σειρά χαρακτηριστικών ασφάλειας, όπως έλεγχος ταυτότητας χρήστη, εξουσιοδοτήσεις ρόλου χρήστη, επαλήθευση email, υπηρεσίες κρυπτογράφησης, κατακερματισμός κωδικού πρόσβασης, δυνατότητες επαναφοράς κωδικού πρόσβασης, ελέγχου ροής κλήσεων (throttling) κ.ά. Ταυτόχρονα, παρέχεται πλήθος λειτουργιών για την επικύρωση δεδομένων (data validation) μέσω των οριζόμενων κανόνων που διαθέτει.

Παρέχεται ένα εύκολο σύστημα ελέγχου εκδόσεων (migrations) για πλήθος βάσεων δεδομένων, δυνατότητα μέσω της οποίας παρακολουθούνται οι αλλαγές στη βάση δεδομένων κατά τη διάρκεια του χρόνου, καθιστώντας έτσι ευκολότερη την καταστροφή, την αναδημιουργία και τη μεταφορά της βάσης δεδομένων στις περιπτώσεις που κάτι τέτοιο κριθεί απαραίτητο.

Επίσης, υποστηρίζονται λειτουργίες δοκιμών ελέγχου (unit testing), με τις οποίες γίνεται έλεγχος μικρών απομονωμένων ενοτήτων κώδικα εφαρμογής και χαρακτηριστικών, δυνατότητα που εξυπηρετεί στη διατήρηση του κώδικα σύμφωνα με τις προκαθορισμένες απαιτήσεις.

Τέλος, χρησιμοποιείται το πακέτο Flysystem PHP για την παροχή προγραμμάτων οδήγησης για εργασία με μια ποικιλία συστημάτων αρχείων, από τοπικά συστήματα αρχείων έως αποθήκευση στο cloud όπως το Amazon S3, ενώ παρέχεται επίσης δυνατότητα μεταφοράς αρχείων με SSH File Transfer Protocol (SFTP), διευκολύνοντας έτσι τη διαχείριση αρχείων που προέρχονται από ή που προορίζονται για διάφορους προορισμούς [Laravel].

5.3 Εγκατάσταση προγραμμάτων

Το Elastic Stack είναι ένα πλούσιο οικοσύστημα στοιχείων που λειτουργούν ως μια πλήρη στοίβα (stack) αναζήτησης και ανάλυσης στοιχείων. Τα κύρια συστατικά που απαρτίζουν το συγκεκριμένο stack είναι τα Elasticsearch, Logstash, Kibana, Beats και X-Pack.

Το Elasticsearch βρίσκεται στην καρδιά του Elastic Stack, παρέχοντας δυνατότητες αποθήκευσης, αναζήτησης και ανάλυσης. Το Kibana, επίσης αναφέρεται ως η διεπαφή αλληλεπίδρασης στο Elastic Stack, το οποίο παρέχει εξαιρετικές δυνατότητες απεικόνισης των διαφόρων λειτουργιών και αποτελεσμάτων. Το Logstash και το Beats βοηθούν στη λήψη των δεδομένων στο Elastic Stack. Το X-Pack παρέχει ισχυρές δυνατότητες όπως παρακολούθηση, ειδοποιήσεις, ασφάλεια, παροχή γραφημάτων και μηχανική μάθηση [SK19].

Η εγκατάσταση του Elastic Stack μπορεί να γίνει σε εικονική μηχανή (Virtual Machine, VM) ή απευθείας στον υπολογιστή. Πριν την εγκατάσταση είναι απαραίτητο να καλύπτονται οι

ελάχιστες απαιτήσεις για τα λειτουργικά συστήματα [https://www.elastic.co/support/matrix] και την Εικονική Μηχανή της Java (JVM) [https://www.elastic.co/support/matrix#matrix_jvm]. Ακολουθούν τα βήματα για την εγκατάσταση των στοιχείων του Elastic Stack που χρησιμοποιήθηκαν κατά τη διάρκεια εκπόνησης της συγκεκριμένης διπλωματικής εργασίας.

5.3.1 Εγκατάσταση Elasticsearch

Το Elasticsearch μπορεί να χρησιμοποιηθεί σε προσωπικό υπολογιστή ή μέσω της την φιλοξενούμενης υπηρεσία Elasticsearch στο Elastic Cloud. Η υπηρεσία Elasticsearch είναι διαθέσιμη τόσο σε AWS όσο και σε GCP. Για την εγκατάσταση σε Windows 10, ακολουθούνται τα παρακάτω βήματα:

1. Κατέβασμα του αρχείου zip Elasticsearch 7.12.1 για Windows από την επίσημη ιστοσελίδα του Elasticsearch.
2. Αποσυμπίεση του περιεχομένου του αρχείου σε έναν φάκελο του υπολογιστή (π.χ. C:\Program Files).
3. Εκτέλεση ως “Διαχειριστής” μίας γραμμής τερματικού και πλοήγηση στον φάκελο που περιέχει τα αποσυμπιεσμένα αρχεία.
4. Εκκίνηση του Elasticsearch με την εντολή:

```
bin\elasticsearch.bat
```

5. Έλεγχος λειτουργίας του Elasticsearch αποστέλλοντας ένα HTTP GET request στο port 9200 ή απλά αναγράφοντας την παρακάτω διεύθυνση URL σε έναν browser:

```
curl http://127.0.0.1:9200
```

Η απόκριση πρέπει να είναι παρόμοια με την παρακάτω:

```
{
  "name" : "your_host_name",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "CE9RdlV-SLaMlTFYK7JSDQ",
  "version" : {
    "number" : "7.12.1",
    "build_flavor" : "default",
    "build_type" : "tar",
    "build_hash" : "6837139b9c6b6d23c3200870651f10d334",
    "build_date" : "2021-04-20T20:56:39.040728659Z",
    "build_snapshot" : false,
    "lucene_version" : "8.8.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

5.3.2 Εγκατάσταση Kibana

Η εγκατάσταση του Kibana προτείνεται να γίνει στον ίδιο server με το Elasticsearch, αλλά κάτι τέτοιο δεν είναι απαραίτητο. Σε περίπτωση που η εγκατάσταση γίνει σε διαφορετικούς server, τότε πριν την εκκίνηση του Kibana θα πρέπει να τροποποιηθεί η URL (IP:PORT) του Elasticsearch server στο αρχείο παραμετροποίησης του Kibana (kibana.yml). Για την εγκατάσταση σε Windows 10, ακολουθούνται τα παρακάτω βήματα:

1. Κατέβασμα του αρχείου zip Kibana 7.12.1 για Windows από την επίσημη ιστοσελίδα του Elasticsearch.
2. Αποσυμπίεση του περιεχομένου του αρχείου σε έναν φάκελο του υπολογιστή (π.χ. C:\Program Files).
3. Εκτέλεση ως "Διαχειριστής" μίας γραμμής τερματικού και πλοήγηση στον φάκελο που περιέχει τα αποσυμπιεσμένα αρχεία.
4. Εκκίνηση του Kibana με την εντολή:

```
bin\kibana.bat
```

5. Εναλλακτικά, εκκίνηση του Kibana μέσω web interface:

```
http://127.0.0.1:5601
```

5.3.3 Εγκατάσταση Beats

Το Beats είναι λογισμικό ανοικτού κώδικα που εγκαθιστάται ως πράκτορας στον server και χρησιμοποιείται για την αποστολή επιχειρησιακών δεδομένων απευθείας στο Elasticsearch ή μέσω του Logstash, όπου έπειτα τα δεδομένα αυτά μπορούν να υποστούν περαιτέρω επεξεργασία. Κάθε στοιχείο του Beats είναι ένα ξεχωριστό προϊόν. Για τη συγκεκριμένη εφαρμογή θα εγκατασταθεί το Metricbeat για τη συλλογή μετρικών συστήματος.

1. Κατέβασμα του αρχείου zip "metricbeat-7.12.1-windows" για Windows από την επίσημη ιστοσελίδα του Elasticsearch.
2. Αποσυμπίεση του περιεχομένου του αρχείου σε έναν φάκελο του υπολογιστή (π.χ. C:\Program Files).
3. Μετονομασία του φακέλου "metricbeat-7.12.1-windows" σε "Metricbeat"
4. Εκτέλεση ως "Διαχειριστής" μίας γραμμής τερματικού PowerShell και πλοήγηση στον φάκελο που περιέχει τα αποσυμπιεσμένα αρχεία.
5. Από το τερματικό, εκτέλεση των παρακάτω εντολών, ώστε να ολοκληρωθεί η εγκατάσταση του Metricbeat ως Windows service:

```
PS > cd 'C:\Program Files\Metricbeat'
PS C:\Program Files\Metricbeat> .\install-service-metricbeat.ps1
```

Το Metricbeat περιέχει το module συλλογής μετρήσεων σε επίπεδο συστήματος, όπως χρήση CPU, μνήμη, σύστημα αρχείων, IO δίσκου και στατιστικά IO δικτύου, καθώς και στατιστικά για κάθε διαδικασία που εκτελείται στο σύστημα. Πριν την παραμετροποίηση, είναι απαραίτητη η επαλήθευση λειτουργίας των Elasticsearch και Kibana και ότι το Elasticsearch είναι έτοιμο για να δεχτεί δεδομένα από το Metricbeat. Η παραμετροποίηση του συγκεκριμένου "system" module γίνεται ως εξής:

1. Ενεργοποίηση του "system" modulea από τον φάκελο εγκατάστασης του Metricbeat.

```
PS C:\Program Files\Metricbeat> .\metricbeat.exe modules enable system
```

2. Αρχικοποίηση περιβάλλοντος:

```
PS C:\Program Files\Metricbeat> metricbeat.exe setup -e
```

   Η εντολή "setup" φορτώνει τους πίνακες ελέγχου Kibana. Εάν οι πίνακες ελέγχου έχουν ήδη ρυθμιστεί, η εντολή αυτή παραλείπεται. Η σημαία "-e" είναι προαιρετική και στέλνει έξοδο σε τυπικό σφάλμα αντί για το syslog.

3. Εκκίνηση του Metricbeat:

```
PS C:\Program Files\Metricbeat> Start-Service metricbeat
```

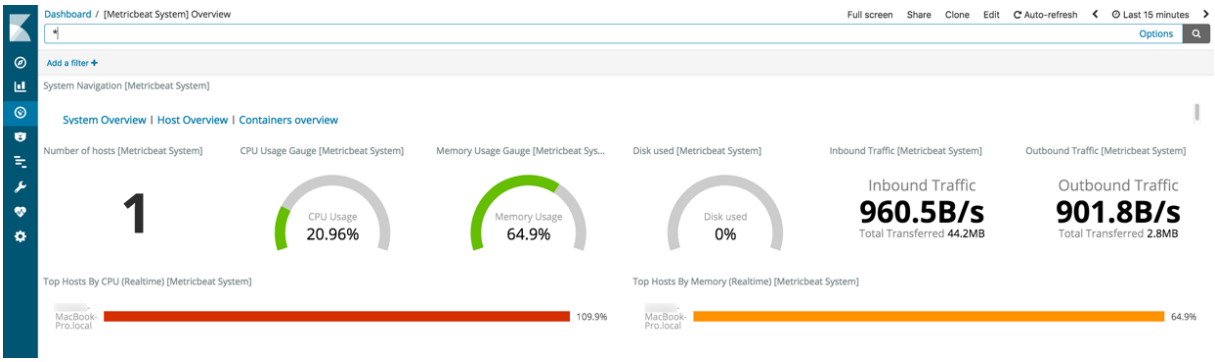
   Πλέον, το Metricbeat έχει ξεκινήσει την αποστολή δεδομένων προς το Elasticsearch.

5.3.4 Οπτικοποίηση μετρικών συστήματος στο Kibana

Η οπτικοποίηση των μετρικών συστήματος είναι δυνατή μέσω του browser από το URL:

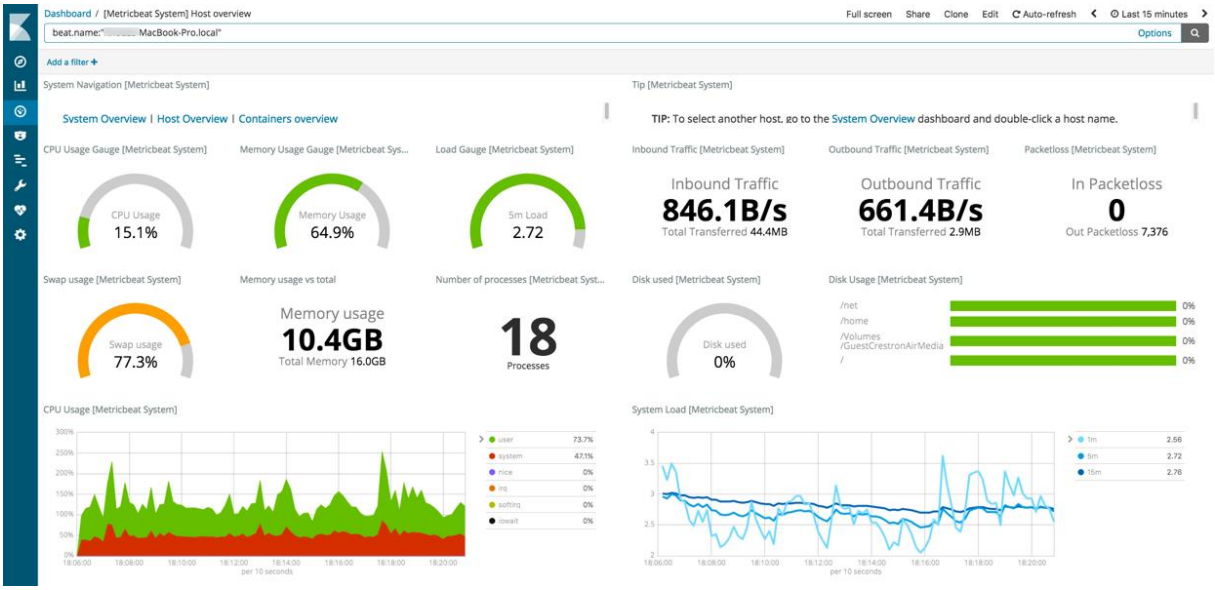
```
http://localhost:5601/app/kibana#/dashboard/Metricbeat-system-overview-ecs
```

Τα δεδομένα θα εμφανιστούν ως ακολούθως:



Εικόνα 16: Δεδομένα μετρικών συστήματος στο Kibana.

Από την επιλογή "Host Overview" υπάρχουν αναλυτικά δεδομένα μετρικών.



Εικόνα 17: Αναλυτικά δεδομένα μετρικών συστήματος στο Kibana.

Τα δεδομένα αυτά παρέχουν πολύτιμες επισκοπήσεις για τη χρήση των εφαρμογών στο Elasticseach.

5.3.5 Εγκατάσταση Logstash

Το Logstash έχει ως προαπαιτούμενο μία εκ των εκδόσεων Java 8, 11 ή 15. Ο έλεγχος της έκδοσης γίνεται από την κονσόλα του τερματικού ως εξής:

```
java -version
```

Εφόσον η Java είναι εγκατεστημένη στο σύστημα, η παραπάνω εντολή θα έχει ως αποτέλεσμα κάτι όπως το παρακάτω:

```
java version "1.8.0_211"
Java(TM) SE Runtime Environment (build 1.8.0_211-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.211-b12, mixed mode)
```

Το Logstash χρησιμοποιεί την έκδοση Java που έχει οριστεί στην παράμετρο “JAVA_HOME”. Η αντίστοιχη μεταβλητή περιβάλλοντος πρέπει να είναι ορισμένη ώστε το Logstash να λειτουργεί σωστά.

5.3.6 Εγκατάσταση Docker

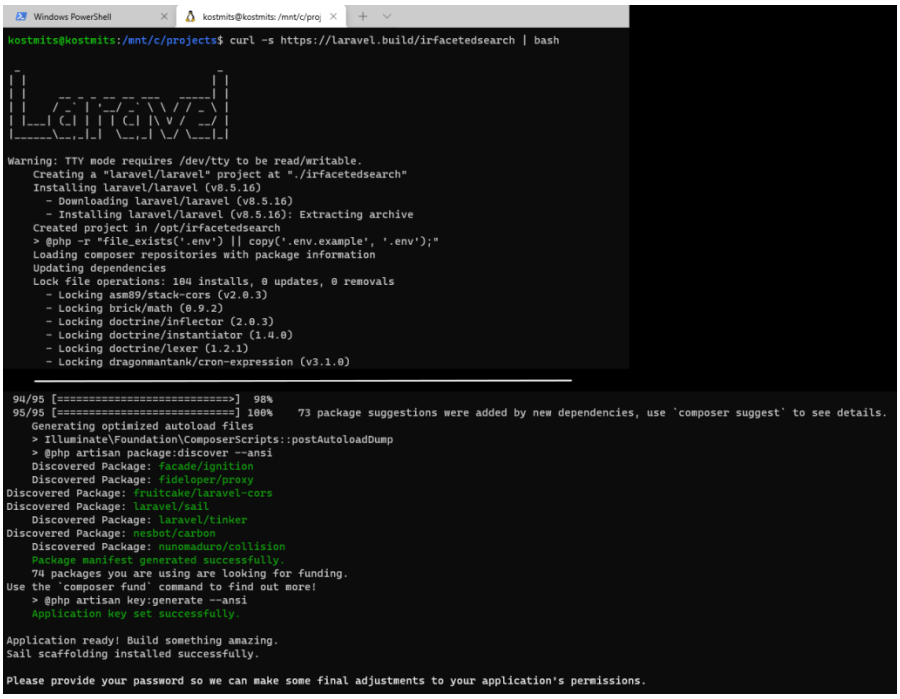
Για τη δημιουργία μίας εφαρμογής Laravel στα Windows 10, θα πρέπει να είναι εγκαταστημένο το Docker Desktop [https://www.docker.com/products/docker-desktop] και να είναι εγκατεστημένο και ενεργοποιημένο το Υποσύστημα Windows για Linux 2 (WSL2), το οποίο επιτρέπει την εκτέλεση Linux binary αρχείων από τα Windows [https://docs.microsoft.com/en-us/windows/wsl/install-win10].

5.3.7 Εγκατάσταση Laravel

Η δημιουργία μίας νέας εφαρμογής Laravel γίνεται μέσω μίας νέας συνόδου WSL2 Linux και εκτέλεσης της εντολής στον επιθυμητό φάκελο:

```
curl -s https://laravel.build/irfacetedsearch | bash
```

Το σύστημα θα δημιουργήσει και θα φορτώσει όλα τα απαραίτητα πακέτα λογισμικού, όπως αυτά έχουν οριστεί στα προαπαιτούμενα του Laravel.



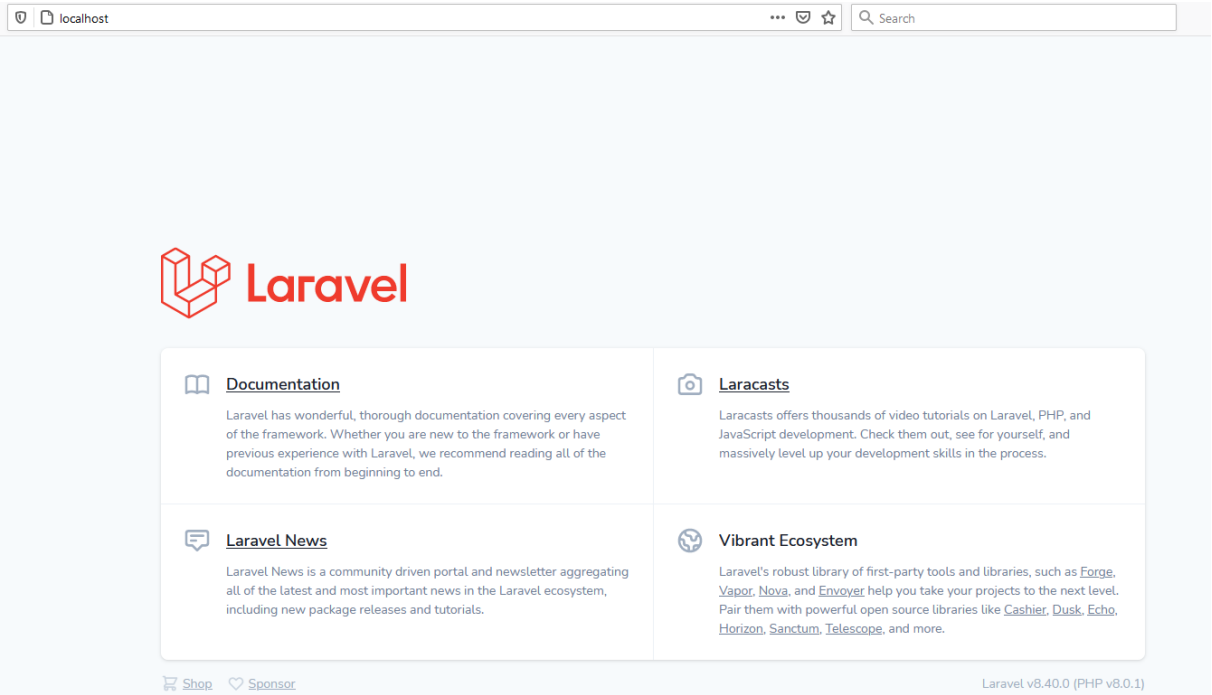
Εικόνα 18: Εγκατάσταση Laravel.

Μόλις ολοκληρωθεί η εγκατάσταση, θα πρέπει να γίνει η εκκίνηση του Laravel Sail, το οποίο είναι μία απλή διεπαφή μεταξύ του Laravel και του Docker.

```
cd irfacetedsearch
./vendor/bin/sail up
```

Την πρώτη φορά που θα εκτελεστεί η εντολή “sail up”, θα δημιουργηθούν στο σύστημα οι περιέκτες εφαρμογών (container) του Sail. Μόλις ξεκινήσουν τα container της εφαρμογής Docker, η εφαρμογή είναι προσβάσιμη μέσω της διεύθυνσης:

```
http://localhost
```



Εικόνα 19: Αρχική οθόνη της εφαρμογής Laravel.

Αναλυτικές λεπτομέρειες για την εγκατάσταση της εφαρμογής μπορούν να βρεθούν στην επίσημη ιστοσελίδα του Laravel [https://laravel.com/docs/8.x/installation].

Σε περίπτωση που γίνει χρήση του Elasticsearch μέσω Docker, τότε πρέπει να γίνει επιπλέον η παρακάτω ρύθμιση, όπως περιγράφεται στην επίσημη ιστοσελίδα [https://www.elastic.co/guide/en/elasticsearch/reference/7.12/docker.html]:

```
sudo sysctl -w vm.max_map_count=262144
```

5.3.8 Εγκατάσταση Node, NPM και Laravel Mix

Για την πλήρη λειτουργικότητα του Laravel θα πρέπει να εγκατασταθούν τα Node.js και NPM στο σύστημα. Αναλόγως συστήματος και παραμετροποίησης θα πρέπει να ακολουθηθούν οι επίσημες οδηγίες του αντίστοιχου προγράμματος [https://nodejs.org/en/download/].

Το μόνο βήμα που απομένει είναι η εγκατάσταση του Laravel Mix. Στη νέα εγκατάσταση του Laravel, υπάρχει ένα αρχείο “package.json” στη ρίζα της δομής του καταλόγου. Το προεπιλεγμένο αρχείο “package.json” περιλαμβάνει ήδη όλα όσα χρειάζονται για τη χρήση του Laravel Mix.

```
sail npm install
```

Παραμετροποίηση εφαρμογής και χρήση Bootstrap στην εφαρμογή:

```
sail composer require laravel/ui
sail artisan ui bootstrap
```

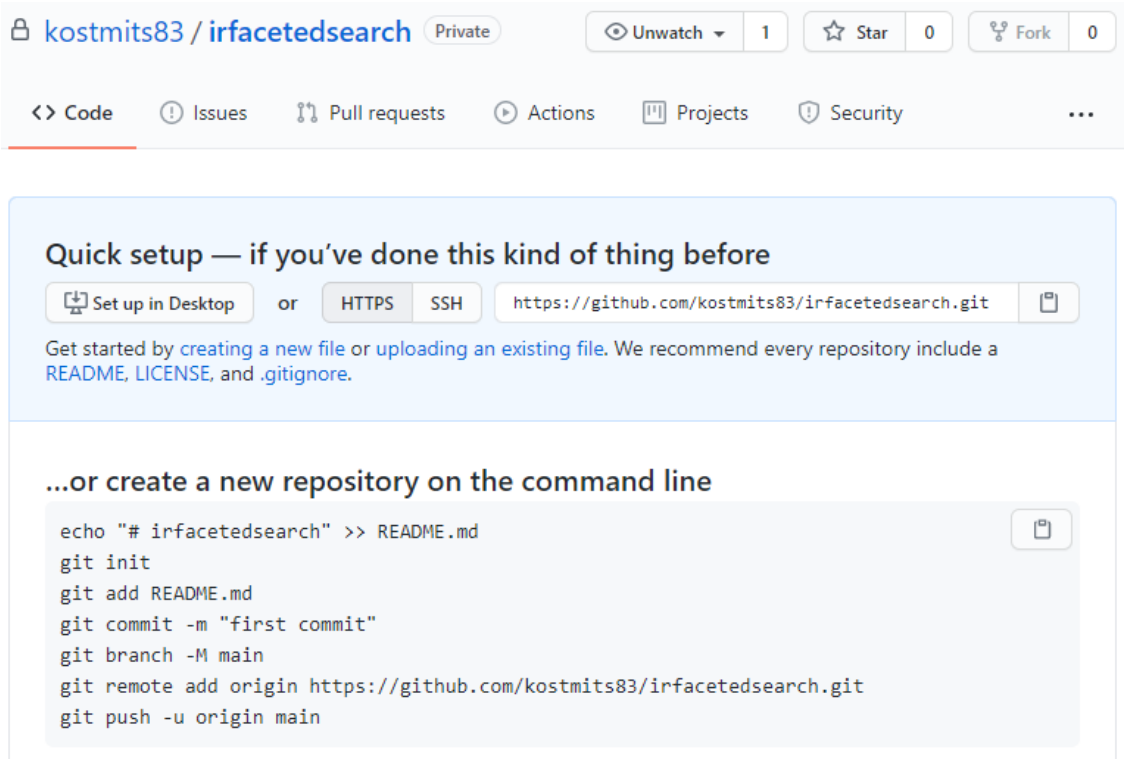
```
npm install
npm run dev
```

5.3.9 Αρχικοποίηση git

Για την καλύτερη υλοποίηση και επισκόπηση του project θα γίνει χρήση του git και της πλατφόρμας github.com. Το git είναι ένα κατανεμημένο σύστημα ελέγχου εκδόσεων στο οποίο οι κόμβοι δεν ενημερώνουν μόνο το τελευταίο στιγμιότυπο των αρχείων τους αλλά αναπαράγουν εξ ολοκλήρου το αποθετήριο, λειτουργώντας έτσι ως αποκεντρωμένα αντίγραφα όλων των δεδομένων. Επιπλέον, πολλά από αυτά τα συστήματα περιλαμβάνουν πλήθος απομακρυσμένων αποθετηρίων, έτσι ώστε να είναι δυνατή η συνεργασία με διαφορετικές ομάδες ανθρώπων και με διαφορετικούς τρόπους ταυτόχρονα στο ίδιο έργο, γεγονός το οποίο επιτρέπει στους χρήστες ενός κατανεμημένου συστήματος ελέγχου εκδόσεων να δημιουργούν διάφορους τύπους ροής εργασιών (π.χ. ιεραρχικά μοντέλα).

Η εγκατάσταση του git γίνεται ακολουθώντας τις οδηγίες στην επίσημη ιστοσελίδα [http://git-scm.com/download/win].

Ως πλατφόρμα αποθετηριών θα χρησιμοποιηθεί το github.com, μέσα από το οποίο διευκολύνεται ο έλεγχος των εκδόσεων μέσα από ετικές, ορόσημα, εκδόσεις κ.ά. Απαραίτητη προϋπόθεση είναι η δημιουργία λογαριασμού στη συγκεκριμένη πλατφόρμα.



Εικόνα 20: Οδηγίες αρχικοποίησης στην πλατφόρμα github.com.

Μετά από αυτό η εφαρμογή μπορεί να παραμετροποιηθεί στο τοπικό σύστημα.

6

Επίλογος

6.1 Σύνοψη και συμπεράσματα

Ο κλάδος της αναζήτησης και ανάκτησης πληροφορίας παρουσιάζει ιδιαίτερο ενδιαφέρον και πλήθος εφαρμογών. Η χρήση νέων τρόπων αναζήτησης τόσο από την πλευρά αποθήκευσης της πληροφορίας όσο και από την πλευρά του τρόπου αλληλεπίδρασης των συστημάτων με τους τελικούς χρήστες, πρόκειται για μία συνεχή και εξελισσόμενη διεργασία. Στην παρούσα διπλωματική εργασία, παρουσιάστηκαν οι βασικές έννοιες όλων των στοιχείων που κρίθηκαν απαραίτητα για την ολιστική προσέγγιση του θέματος. Βασική δομική μονάδα αποτέλεσε η έννοια των τρόπων ανάκτησης πληροφορίας, καθώς βάσει αυτών εξαρτώνται και οι μέθοδοι αναζήτησης που μπορούν να εφαρμοστούν.

Για την επίτευξη του στόχου έγινε χρήση των πιο γνωστών μεθόδων αναζήτησης και εφαρμόστηκε η τεχνική της διαστασιοποίησης των δεδομένων (facets) για τα πεδία που ήταν εφικτό κάτι τέτοιο και σε συνδυασμό με τις υπόλοιπες λειτουργίες αναζήτησης που υλοποιήθηκαν, δημιουργήθηκε ένα ολοκληρωμένο σύστημα αναζήτησης πληροφορίας το οποίο παρέχει πληθώρα επιλογών αναζήτησης.

Όπως είναι εύκολα κατανοητό, η εφαρμογή πολλών από τις λειτουργίες που εξετάστηκαν δεν θα ήταν δυνατή σε ένα παραδοσιακό σύστημα αναζήτησης, είτε λόγω υπολογιστικού φόρτου (π.χ. facet aggregations), είτε λόγω μη εγγενής υποστήριξης των εκάστοτε δυνατοτήτων (score, partial matching κ.ά.), είτε λόγω της πολυπλοκότητας που θα είχε μία τέτοια

προσέγγιση. Εν κατακλείδι, η εφαρμογή αυτή μπορεί να βοηθήσει στην αποτελεσματική διαχείριση και αναζήτηση πληροφορίας σε αγγελίες που διατίθεται σε μορφή ελεύθερου κειμένου συνδυαζόμενου με επιπλέον πεδία και μεταδεδομένα.

6.2 Μελλοντικές επεκτάσεις

Πέρα από την ad hoc αναζήτηση πληροφορίας μέσω των αντίστοιχων φορμών αναζήτησης, είναι αρκετά διαδεδομένη η χρήση προγραμματιστικών διασυνδέσεων (Application Programming Interface – API) για την παροχή on demand RESTful λειτουργιών αναζήτησης. Ως μελλοντική επέκταση του συστήματος θα μπορούσε να υλοποιηθεί ένα τέτοιο υποσύστημα για την καλύτερη και μαζικότερη εξυπηρέτηση των τελικών χρηστών.

Περαιτέρω βελτίωση του συστήματος θα μπορούσε να είναι η καλύτερη αντιστοίχιση των facets με βελτιωμένα HTML στοιχεία, αναλόγως των δεδομένων που αντικατοπτρίζουν (π.χ. ημερομηνίες ή εύρος τιμών).

Επιπλέον, οι ιδιότητες που απαρτίζουν τα facets δύναται να είναι υποκείμενα λεκτικής ανάλυσης για την εύρεση περιπτώσεων συνωνυμίας, πολυσημίας και ομωνυμίας, έτσι ώστε με την κατάλληλη ομαδοποίηση και απαλοιφή να επιτευχθεί κανονικοποίηση των επιλογών και περιορισμός μόνο στις απολύτως απαραίτητες, οδηγώντας έτσι σε καλύτερη αποθήκευση πληροφορίας και εν τέλει σε ακόμη μεγαλύτερη ικανοποίηση των τελικών χρηστών.

7

Βιβλιογραφία

[AKLP11]	S. Attfield, G. Kazai, M. Lalmas, B. Piwowarski. Towards a science of user engagement (Position Paper), 2011.
[BBE+13]	A. Bozzon, M. Brambilla, D. V. Emanuele, P. Fraternali, S. Quarteroni. The Information Retrieval Process, Web Information Retrieval, Springer Berlin Heidelberg, pp.13-26, 2013.
[BBK16]	A. Bosch, T. Bogers, M. Kunder. Estimating search engine index size variability: a 9-year longitudinal study, Scientometrics, 107, pp 839-856, 2016, DOI: 10.1007/s11192-016-1863-z
[BCC10]	S. Büttcher, C. L. Clarke, G. V. Cormack. Information Retrieval Implementing and Evaluating Search Engines, The MIT Press, 2010, ISBN-13: 978-0262528870
[Bew95]	E. W. Brown. Fast Evaluation of Structured Queries for Information Retrieval, SIGIR '95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, pp 30-38, 1995, DOI: 10.1145/215206.215329
[Bj12]	J. Baren. Relating content to the user, Academic and professional publishing, Oxford: Chandos Publishing, Chapter 11, 2012.
[BLLW17]	A. Black. P. Luna, O. Lund, S. Walker. Information design research and practice, Routledge, 2017, ISBN 9780415786324
[BS08]	R. Blanco, F. Silvestri. Efficiency Issues in Information Retrieval Workshop, Efficiency Issues in Information Retrieval Workshop, 4956, p 711, 2008, DOI: 10.1007/978-3-540-78646-7_84
[CJB14]	J. Champ, A. Joly, P. Bonnet. Fine-grained Visual Faceted Search, MM: Conference on Multimedia, Orland, FL, United States, pp 721-722, 2014, DOI: 10.1145/2647868.2654875

[CMS15]	W. B. Croft, D. Metzler, T. Strohman. Search Engines, Information Retrieval in Practice, Pearson Education Inc., 2015, ISBN 978-0-12-407171-1
[Elastic]	https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html
[Elasticquent]	https://github.com/elasticquent/Elasticquent
[FGG06]	C. Flavian, M. Guinaliu, R. Gurrea. The role played by perceived usability, satisfaction and consumer trust on website loyalty, Information & Management, 43, Issue 1, pp 1-14, 2006, DOI: 10.1016/j.im.2005.01.002
[Fp13]	P. Fredelius. Faceted search with a large amount of properties, Chalmers University of Technology, Sweden, 2013, ISSN: 1651-4769
[Google]	https://support.google.com/analytics/answer/1009409?hl=en
[GRG18]	V. N. Gudivada, D. L. Rao, A. R. Gudivada. Information Retrieval: Concepts, Models, and Systems, Handbook of Statistics, 38, pp 331-401, 2018, DOI: 10.1016/bs.host.2018.07.009
[He01]	E. Herrera-Viedma. Modeling the Retrieval Process for an Information Retrieval System Using an Ordinal Fuzzy Linguistic Approach, Journal of the American Society for Information Science and Technology, 52, pp 460-475, 2001, DOI: 10.1002/1532-2890(2001)9999:9999<::AID-ASI1087>3.0.CO;2-Q
[HMT17]	D. Hawking, A. Moffat, A. Trotman: Efficiency in information retrieval: introduction to special issue, Information Retrieval Journal, 20, pp 169-171, 2017, DOI: 10.1007/s10791-017-9309-7
[HS12]	J. He, T. Suel. Optimizing Positional Index Structures for Versioned Document Collections, SIGIR'12 - Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 245-254, 2012, DOI: 10.1145/2348283.2348319
[ISO 9241-210:2010]	ISO 9241-210:2010, Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems, 2010.
[JBS08]	B.J. Jansen, D.L. Booth, A. Spink. Determining the informational, navigational, and transactional intent of Web queries, Information Processing and Management 44, pp 1251-1266, 2008, DOI: 10.1016/j.ipm.2007.07.015
[Ka11]	A. Kopliku. "Approaches to implement and evaluate aggregated search", University of Toulouse, 2011, available at https://www.researchgate.net/publication/277063050_Approaches_to_implement_and_evaluate_aggregated_search
[Laravel]	https://laravel.com/
[LC14]	B. Long, Y. Chang. Entity Ranking, Relevance Ranking for Vertical Search Engines, pp 107-125, 2014.
[LM03]	Z. Lu, K. S. McKinley. Partial Collection Replication versus Caching for Information Retrieval Systems, Inf. Retr., 6. pp 159-198, 2000, DOI: 10.1145/345508.345591
[MAI18]	M. Mahdi, A. A. Ahmad, R. Ismail. Paradigm Extension of Faceted Search Techniques: A Review, 2018.
[MIAT+18]	M. Mahdi, R. Ismail, A. Ahmad, K. Thambiratnam, M. Mohammed. A Design of Faceted Search Engine – a Review, International Journal of Engineering and Technology (UAE), 7, pp 489-493, 2018, DOI: 10.14419/ijet.v7i3.20.20595

[MRS09] C. D. Manning, P. Raghavan, H. Schütze. An Introduction to Information Retrieval, Cambridge University Press, 2009, ISBN: 9780521865715

[MS10] A. A. Maskari, M. Sanderson. A Review of Factors Influencing User Satisfaction in Information Retrieval, Journal of the American Society for Information Science and Technology 61, pp 859-868, 2010, DOI: 10.1002/asi.21300

[NH14] X. Niu, B. Hemminger. Analyzing the Interaction Patterns in a Faceted Search Interface, Journal of the Association for Information Science and Technology, 66, 2014, DOI: 10.1002/asi.23227

[ON10] C. Olston, M. Najork. Web Crawling, Foundations and Trends in Information Retrieval, 4, No. 3, 2010, DOI: 0.1561/1500000017

[Pr07] R. Pettersson. Information Design It Depends, International Institute for Information Design, 2007.

[Pr10] R. Pettersson. Information Design–Principles and Guidelines, Journal of Visual Literacy, 29, Number 2, pp. 167-182, 2010, DOI: 10.1080/23796529.2010.11674679

[Project-A] https://project-a.github.io/on-site-search-design-patterns-for-e-commerce/

[RAIS13] A. Rashid, N. Anwer, M. Iqbal, M. Sher. A Survey Paper: Areas, Techniques and Challenges of Opinion Mining, International Journal of Computer Science Issues, 10, Issue 6, No 2, pp 18-29, 2013.

[RHG+13] O. Rusu, I. Halcu, O. Grigoriu, G. Neculoiu, V. Sandulescu, M. Marinescu, V. Marinescu. Converting unstructured and semi-structured data into knowledge, 11th RoEduNet International Conference, pp 1-4, 2013, DOI: 10.1109/RoEduNet.2013.6511736

[RT12] T. Russel-Rose, T. Tate. Designing the Search Experience: The Information Architecture of Discovery, Elsevier Inc., 2012, ISBN: 9780123969811

[SH15] A. Stolz, M. Hepp. Adaptive Faceted Search for Product Comparison on the Web of Data, Engineering the Web in the Big Data Era, ICWE 2015, Lecture Notes in Computer Science, Springer, 9114, 2015, DOI: 10.1007/978-3-319-19890-3_27

[SK19] P. Shukla, S. Kumar. Learning Elastic Stack 7.0 Second Edition, Packt Publishing, 2019, ISBN 9781789954395

[ST09] G. M. Sacco, Y. Tzitzikas. Dynamic Taxonomies and Faceted Search Theory, Practice, and Experience, Springer, 2009, ISBN 978-3-642-02359-0

[Uj11] J. Urbano. Information Retrieval Meta-Evaluation: Challenges and Opportunities in the Music Domain, Proceedings of the 12th International Society for Music Information Retrieval Conference, pp 609-614, Ismir, Turkey, 2011.

[VJ15] G. Veena, G. Jalaja. Levenshtein Distance based Information Retrieval, International Journal of Scientific & Engineering Research, 6, Issue 5, pp 112-116, 2015, ISSN 2229-5518

[Vp09] P. Vora. Web Application Design Patterns, Elsevier Inc., 2009, ISBN: 9780123742650

[WI12] W. Wisam, P. Ipeirotis. Automatic Extraction of Useful Facet Hierarchies from Text Databases, IEEE 24th International Conference on Data Engineering, pp. 466-475, 2008, DOI: 10.1109/ICDE.2008.4497455

[WLL12] X. Wei, X. Luo, Q. Li. Automatic Facet Extraction based on Multidimensional Semantic Index, Eighth International Conference on Semantics, Knowledge and Grids, pp 64-71, 2012, DOI: 10.1109/SKG.2012.22

[WLZZ+13] B. Wei, J. Liu, Q. Zheng, W. Zhang, X. Fu, B. Feng. A survey of faceted search. Journal of Web Engineering, 12, pp 41-64, 2013.

[YGN+08] O. Yitzhak, N. Golbandi, N. Harel, R. Lempel, A. Neumann, S. Ofek-Koifman, D. Sheinwald, E. Shekita, B. Sznajder, S. Yogev. Beyond basic faceted search. WSDM'08 - Proceedings of the 2008 International Conference on Web Search and Data Mining, pp 33-44, 2008, DOI: 10.1145/1341531.1341539