Τμήμα Ηλεκτρολόγων Μηχανικών
& Μηχανικών Υπολογιστών
**ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΕΛΟΠΟΝΝΗΣΟΥ**

Τεχνολογίες και Υπηρεσίες Ευφυών Συστημάτων Πληροφορικής και Επικοινωνιών
Πρόγραμμα Μεταπτυχιακών Σπουδών

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ανάλυση, ταξινόμηση και αναπαράσταση στοχευμένων θεμάτων από τα κοινωνικά δίκτυα και τον παγκόσμιο ιστό

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΦΟΙΤΗΤΡΙΑΣ : Παναγιώτα Μποβιάτση

ΥΠΕΥΘΥΝΟΣ ΚΑΘΗΓΗΤΗΣ: Β. Ταμπακάς

ΠΑΤΡΑ, Σεπτέμβριος 2022

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

Πάτρα, Ημερομηνία

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ
1. Ονοματεπώνυμο, Υπογραφή
2. Ονοματεπώνυμο, Υπογραφή
3. Ονοματεπώνυμο, Υπογραφή

Υπεύθυνη Δήλωση Φοιτητή Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τη συγκεκριμένη εργασία. Η έγκριση της διπλωματικής εργασίας από το Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος. Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία της φοιτήτριας Μποβιάτσης Παναγιώτας που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης ο συγγραφέας/δημιουργός εκχωρεί στο Πανεπιστήμιο Πελοποννήσου, μη αποκλειστική άδεια χρήσης του δικαιώματος αναπαραγωγής, προσαρμογής, δημόσιου δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσής τους διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος και για όλο το χρόνο διάρκειας των δικαιωμάτων πνευματικής ιδιοκτησίας. Η ανοικτή πρόσβαση στο πλήρες κείμενο για μελέτη και ανάγνωση δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, αποθήκευση, πώληση, εμπορική χρήση, μετάδοση, διανομή, έκδοση, εκτέλεση, «μεταφόρτωση» (downloading), «ανάρτηση» (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού. Ο συγγραφέας/δημιουργός διατηρεί το σύνολο των ηθικών και περιουσιακών του δικαιωμάτων. Οι μορφή των εισαγωγικών σελίδων είναι με μικρούς χαρακτήρες ρωμαϊκής γραφής (i, ii, iii, iv, κλπ)

Περίληψη

Αντικείμενο της παρούσας εργασίας αποτελεί η μελέτη εργαλείων crawler. Πρόκειται για προγράμματα που επισκέπτονται web ιστοσελίδες και διαβάζουν τις σελίδες τους και άλλες πληροφορίες, δημιουργώντας καταχωρήσεις για ένα ευρετήριο μηχανής αναζήτησης.

Αρχικά θα γίνει αναφορά σε κάποιες εισαγωγικές έννοιες και γενικά σημεία του crawling. Έπειτα θα περιγραφεί και θα αναλυθεί το Apache Nutch και το Sparkler, θα περιγραφεί η αρχιτεκτονική τους και η διαδικασία εγκατάστασης τους. Τέλος θα παρουσιαστούν τα προβλήματα και τα συμπεράσματα που θα προκύψουν για το κάθε ένα. Επίσης θα γίνει αναφορά σε διάφορα εργαλεία crawler τα οποία κάνουν ανίχνευση ιστού σε σελίδες και είναι ανοιχτού κώδικα. Επίσης θα αναφερθούν κάποια βασικά πλεονεκτήματα και μειονεκτήματα τους.

Θα αναλυθούν οι τεχνολογίες που θα χρησιμοποιήθούν για την τελική εφαρμογή. Ποιές τεχνολογίες χρησιμοποιήθηκαν στον scraper, στο back-end με την παρουσίαση ενός swagger, και την front-end τελική εφαρμογή του χρήστη. Θα γίνει η ανάλυση της εφαρμογής, αναλύοντας τον τρόπο που θα γίνει η συλλογή των δεδομένων, περιεχόμενα του API / back-end, η αρχιτεκτονική της εφαρμογής και τέλος το interface της εφαρμογης. Στο τέλος θα αναφερθούν κάποια γενικά συμπεράσματα και παρατηρήσεις.

Abstract

The purpose of this work is the study of crawler tools. These are programs that visit web sites and read their pages and other information, creating entries for a search engine index.

First, reference will be made to some introductory concepts and general points of crawling. Apache Nutch and Sparkler will then be described and analyzed, their architecture and installation process described. Finally, the problems and the conclusions that will arise for each one will be presented. Reference will also be made to various crawler tools that crawl web pages and are open source. Some of their main advantages and disadvantages will also be mentioned.

The technologies to be used for the final application will be analyzed. What technologies were used in the scraper, swagger details, and front-end end user application. At the end, some general conclusions and observations will be mentioned.

Πίνακας Περιεχομένων

Πίνακας Εικονών

ΚΕΦΑΛΑΙΟ 1 - Εισαγωγή

1.1 Εισαγωγή

Στο ακόλουθο κεφάλαιο, αναφέρονται και περιγράφονται κάποιοι βασικοί ορισμοί οι οποίοι είναι βασικοί για την κατανόηση του περιεχομένου της παρούσας διπλωματικής.

1.2 Τι ειναι crawler

Ο ανιχνευτής (crawler) είναι ένα πρόγραμμα που επισκέπτεται web ιστοσελίδες και διαβάζει τις σελίδες τους και άλλες πληροφορίες μέσω ενός πρωτοκόλλου HTTP, προκειμένου να δημιουργήσει καταχωρήσεις για ένα ευρετήριο (index) μηχανής αναζήτησης. Το όνομα τους το απέκτησαν απο τον τρόπο που κάνουν ανιχνεύσεις, επειδή ανιχνεύουν σε έναν ιστότοπο μια σελίδα κάθε φορά, ακολουθώντας τους συνδέσμους προς άλλες σελίδες στον ιστότοπο μέχρι να διαβαστούν όλες οι σελίδες. Όλες οι μεγάλες μηχανές αναζήτησης έχουν ένα τέτοιο πρόγραμμα, το οποίο είναι γνωστό ως "spider" ή "bot". Εφαρμόζοντας έναν αλγόριθμο αναζήτησης σε όλα τα δεδομένα που έχουν συλλεχθεί από το πρόγραμμα, μπορούν να έχουν ως απαντήσεις διάφορους συνδέσμους σε κάθε ερώτηση των χρηστών. [3]

1.3 Τι είναι search indexing

Η ευρετηρίαση αναζήτησης είναι η διαδικασία που οι μηχανές αναζήτησης οργανώνουν την πληροφορία που έχουν συλλεχθεί, ώστε να έρχονται εξαιρετικά γρήγορες απαντήσεις σε κάθε ερώτημα των χρηστών. Έτσι η εμπειρία χρήστη γίνεται πολύ ευχάριστη. Χωρίς αυτή την διαδικασία, η εσωτερική αναζήτηση σε έναν ιστότοπο καταλαμβάνει πολλούς πόρους και κάνει τον ιστότοπο πιο αργό. Για αυτό είναι πολύ σημαντικό μέρος στην διαδικασία της αναζήτησης, η οποία χρησιμοποιείται για την άμεση απάντηση ερωτημάτων των χρηστών.

1.4 Web mining

Η εξόρυξη Ιστού είναι η διαδικασία των τεχνικών εξόρυξης δεδομένων για την αυτόματη ανακάλυψη και εξαγωγή πληροφοριών από έγγραφα και υπηρεσίες Ιστού. Αφορά την εφαρμογή τεχνικών, μεθοδολογιών και μοντέλων της εξόρυξης δεδομένων σε δεδομένα που προέρχονται από τα κοινωνικά μέσα. Ο κύριος σκοπός της εξόρυξης Ιστού είναι η ανακάλυψη χρήσιμων πληροφοριών από τον Παγκόσμιο Ιστό. Χρησιμοποιεί αυτοματοποιημένες μεθόδους για την εξαγωγή τόσο δομημένα όσο και μη δομημένα δεδομένα από ιστοσελίδες, αρχεία καταγραφής και συνδέσμους. Υπάρχουν τρείς κύριες υποκατηγορίες του web mining.

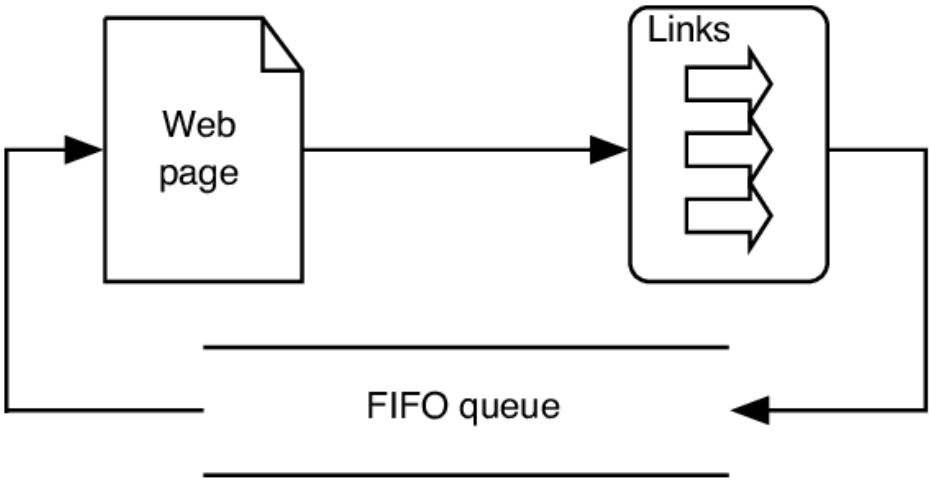
- Η εξόρυξη περιεχομένου Ιστού εξάγει πληροφορίες μέσα από μια σελίδα.
- Η εξόρυξη δομών Ιστού ανακαλύπτει τη δομή των υπερσυνδέσμων μεταξύ εγγράφων, κατηγοριοποιώντας πολλές ιστοσελίδες και βαθμολογώντας την ομοιότητα και τη σχέση μεταξύ διαφορετικών ιστοσελίδων.
- Η εξόρυξη χρήσης Ιστού βρίσκει τις περιπτώσεις χρήσης ιστοσελίδων.

1.5 Είδη crawlers

Υπάρχουν πολλές στρατηγικές ανίχνευσης (crawling), ανάλογα τον τρόπο ανίχνευσης και τον τρόπο ανάκτησης των ιστοσελίδων.

- **Breadth First Crawler:**

Χρησιμοποιώντας αυτό το είδος ανιχνευτή μπορεί να γίνει η εξερεύνηση σε μικρό σύνολο σελιδών και μετά να ακολουθήσει η εξερεύνηση σε άλλες σελίδες που περιέχονται οι σύνδεσμοι τους μεσα σε αυτήν. Δηλαδή ανίχνευση ανα πλάτος πρώτα, έτσι ώστε να ανιχνευτούν πρώτα οι πιο σημαντικές σελίδες.



Εικόνα 1: Breadth First Crawler

- **Hidden Web Crawlers:**

Αυτό το είδος ανιχνευτή βοηθάει στην εύρεση πληροφορίας που δεν είναι στην ‘επιφάνεια’ του ιστότοπου, για αυτό κιόλας ονομάζεται «deep web» ή «hidden web». Πολλά δεδομένα για να τα ανακτήσεις χρειάζεται να συμπληρώσεις κάποια στοιχεία, συνήθως σε φόρμα, τα οποία μπορεί να είναι αποθηκευμένα σε βάση δεδομένων. Έτσι αποκτιέται πρόσβαση σε πολλά δεδομένα τα οποία είναι κρυμμένα.

- **Incremental Web Crawlers:**

Ένας αυξητικός ανιχνευτής ουσιαστικά ανανεώνει σταδιακά την συλλογή που έχει, επισκέπτοντας τες συχνά, το πόσο συχνά θα γίνεται αυτή η ανανέωση εξαρτάται από το πόσο συχνά αλλάζουν αυτές οι σελίδες. Αντίθετα με έναν παραδοσιακό ανιχνευτή, ο οποίος ανανεώνει και αντικαθιστά περιοδικά τα παλιά έγγραφα με τα καινούργια ληφθέντα.

- **Focused Crawler:**

Αυτό το είδος crawler προσπαθεί να συλλέξει σελίδες ή/και έγγραφα που σχετίζονται με το συγκεκριμένο θέμα αναζήτησης. Ο ‘εστιασμένος’ ανιχνευτής καθορίζει πόσο η πληροφορία που έχει συλλέξει είναι σχετική με το θέμα αναζήτησης. Η λογική του είναι να εκτελέσει αναδρομικά μια εξαντλητική αναζήτηση μέχρι ένα δεδομένο βάθος, ξεκινώντας από σχετικές σελίδας με υψηλή ταξινόμηση. Μια τυπική χρήση ενός ‘εστιασμένου’ ανιχνευτή είναι η δημιουργία ψηφιακών βιβλιοθηκών σε μια συγκεκριμένη περιοχή γνώσης. Εφόσον ένας ‘εστιασμένος’ ανιχνευτής δεν προσπαθεί να ευρετηριάσει ολόκληρο τον ιστό, αλλά μόνο μια σχετικά μικρή υποπεριοχή, η απαιτούμενη υπολογιστική ισχύς είναι πολύ μικρότερη και

χρειάζονται λιγότεροι πόροι δικτύου. Ο ανιχνευτής μειώνει την αναλογία των άχρηστων πληροφοριών και συνδυάζει ταυτόχρονα θεματικά σχετική γνώση.

- **Parallel Crawlers:**

Ένας παράλληλος ανιχνευτής αποτελείται από πολλαπλές διεργασίες ανίχνευσης που ονομάζονται C-procs και μπορούν να εκτελεστούν σε δίκτυο σταθμών εργασίας. Ο παραλληλισμός αυτός βοηθάει πολύ στην άμεση λήψη εγγράφων σε πραγματικό χρόνο. Αυτό το είδος ανιχνευτή λύνει το πρόβλημα, το web μεγαλώνει συνεχώς και γίνεται πιο δύσκολη η γρήγορη ανάκτηση του κάθε ιστοτόπου.

- **Distributed Web Crawler:**

Το είδος της κατανεμημένης ανίχνευσης ιστού είναι μια κατανεμημένη υπολογιστική τεχνική. Το οποίο χρησιμοποιούν πολλοί ανιχνευτές γιατι εργάζονται για την διαδικασία την ανίχνευσης πολλοί ανιχνευτές. Αυτό φέρνει αυξημένη αποτελεσματικότητα και ποιοτικά αποτελέσματα αναζήτηση. Ένα απο τα πιο σημαντικά πλεονεκτήματα του είναι η ανθεκτικότητα του σε σφάλματα. Στο κατανεμημένο πρόγραμμα ανίχνευσης ιστού, ένας διακομιστής URL διανέμει μεμονωμένες διευθύνσεις URL σε πολλούς ανιχνευτές, οι οποίοι πραγματοποιούν λήψη ιστοσελίδων παράλληλα. Στη συνέχεια, στέλνουν τις ληφθείσες σελίδες σε ένα κεντρικό ευρετήριο, στο οποίο εξάγονται σύνδεσμοι και αποστέλλονται μέσω του διακομιστή URL στους ανιχνευτές. Αυτή η κατανεμημένη φύση της διαδικασίας ανίχνευσης μειώνει τις απαιτήσεις υλικού και αυξάνει τη συνολική ταχύτητα λήψης και αξιοπιστία.

1.6 Χρήσεις ανίχνευσης ιστού

Τα προγράμματα ανίχνευσης ιστού δεν περιορίζονται μόνο για χρήση στις μηχανές αναζήτησης. Γίνεται η χρήση τους και σε άλλες περιπτώσεις όπως αναλύονται παρακάτω.

- **Email crawling**

Η ανίχνευση email είναι από τις πιο χρήσιμες ενέργειες που μπορεί να κάνει μια εταιρεία. Με αυτόν τον τρόπο μπορεί να βρεί τα emails διαφορετικών πελατών που δεν είχαν τα στοιχεία τους μεχρι την δεδομένη στιγμή. Βέβαια αυτή η διαδικασία είναι παράνομη μιάς και παραβιάζει το προσωπικό απόρρητο των χρηστών και δεν μπορεί να χρησιμοποιηθεί χωρίς την άδεια του χρήστη.

- **News crawling**

Στο διαδίκτυο υπάρχει τεράστιος όγκος ειδήσεων, η εξαγωγή όλων αυτών των ειδήσεων απο διαφορετικούς ιστότοπους είναι αδύνατη να γίνει. Για αυτό τον λόγο, χρησιμοποιόντας κάποιο πρόγραμμα ανίχνευσης μπορεί να γίνει εξαγωγή σε απίστευτο πλήθος ιστοσελίδων και χρόνο.

- **Image crawling**

Το διαδύκτιο είναι γεμάτο εικόνες, με αυτόν τον τύπο ανίχνευσης γίνεται η εφαρμογή του σε αυτό το είδος δεδομένων. Αρχικά συλλέγει τις πληροφορίες που θα βασιστεί αυτή η αναζήτηση στο διαδίκτυο, οι λέξεις-κλειδιά ή οι φράσεις που δίνει ο χρήστης για να μπορέσει να γίνει ανάκτηση των επιθυμητών οπτικών αναπαραστάσεων. Χρησιμοποιώντας αυτά τα κριτήρια αναζήτηση μαζί με την βοήθεια εργαλείων αναζήτησης (Google, seekweb, DuckDuckGo κ.α.), γίνεται η συλλογή των δεδομένων. Μετα την συλλογή των δεδομένων, οι πληροφορίες αποθηκεύονται προσωρινά. Ώστε στην συνέχεια, να σαρωθεί και να εξαχθεί το url της κάθε εικόνας και να συγκριθεί με τα ήδη υπάρχοντα url που είναι στην βάση δεδομένων για να γίνει

αποφυγή διπλότυπων δεδομένων. Τέλος, μετα την επιλογή των δεδομένων που θα κρατηθούν, γίνεται η λήψη και η αποθήκευση τους στη βάση δεδομένων μαζί με τα μεταδεδομένα που περιέχουν, όπως είναι το url, το όνομα αρχείου, το μέγεθος κ.α. Έτσι, τέτοιοι ανιχνευτές βοηθούν τους χρήστες να βρίσκουν σχετικές εικόνες σε μια πληθώρα εικόνων στο διαδίκτυο.

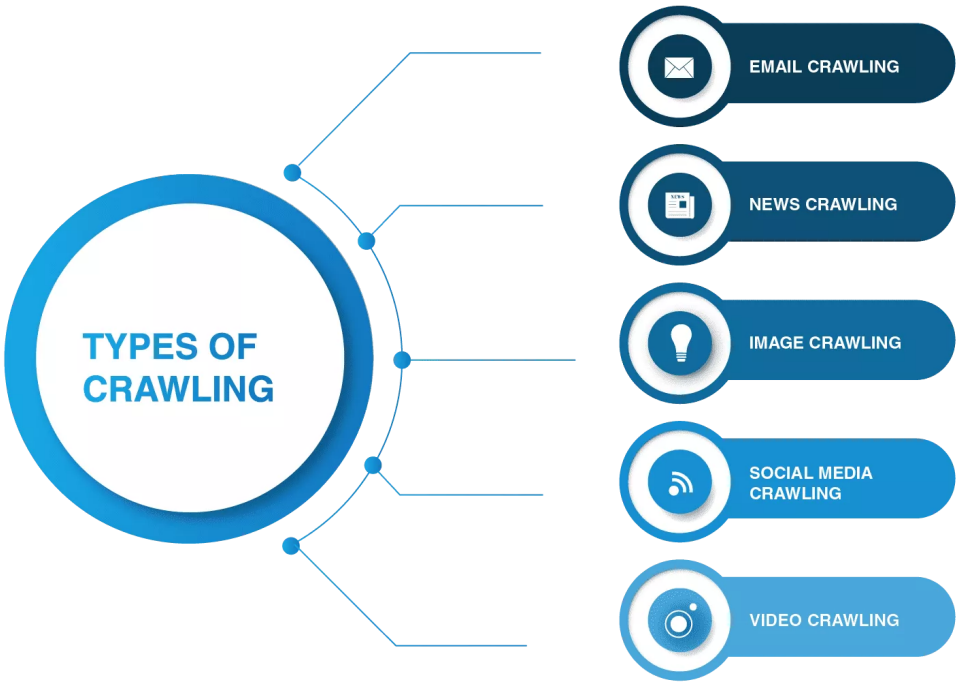
- **Social media crawling**

Μια αρκετά ενδιαφέρουσα ανίχνευση είναι αυτή των μέσων κοινωνικής δικτύωσης. Βέβαια δεν επιτρέπεται η ανίχνευση σε όλα τα μέσα κοινωνικής δικτύωσης, η διαδικασία ανίχνευσης σε αυτές τις περιπτώσεις είναι παράνομη. Όμως υπάρχουν και τα μέσα που αποδέχοντε αυτή την ενέργεια. Το Twitter ή το Pinterest επιτρέπουν στα bots να ανιχνεύσουν τις σελίδες τους, αν δεν περιέχονται προσωπικά δεδομένα μέσα σε αυτά, σε αντίθεση με το Facebook ή το LinkedIn.

Τα δεδομένα αυτά που δημιουργούνται από τους χρήστες, έχουν τη μορφή μη δομημένων δεδομένων. Εκτός από το τι κάνουν τα εργαλεία αυτόματης ανίχνευσης ιστού, πλέον πολλά κανάλια μέσων κοινωνικής δικτύωσης προσφέρουν επί πληρωμή API σε απλούς χρήστες, ακαδημαϊκούς, ερευνητές και ειδικούς οργανισμούς,

- **Video crawling**

Αρκετές φορές είναι πιο απλή και χρήσιμη η παρακολούθηση ενός βίντεο από το να γίνει ανάγνωση όλης της πληροφορίας που περιέχει αυτό. Στην περίπτωση που χρειαστεί να ενσωματωθεί κάποια πλατφόρμα με βίντεο όπως είναι το Youtube, μπορεί να ανιχνευθούν απο συγκεκριμένα προγράμματα ανίχνευσης.



Εικόνα 2: Τύποι Web Crawler

1.7 Συνεισφορά

Τελικός σκοπός της παρούσας διπλωματικής είναι η υλοποιήση μιας εφαρμογής που θα αντλεί πληροφορίες για συγκεκριμένο θέμα και απο συγκεκριμένες πηγές-ιστότοπους και οχι απο όλο το διαδίκτυο.

Για να γίνει αυτή η υλοποίηση πρέπει να μελετηθούν κάποια εργαλεία τα οποία ίσως βοηθήσουν στο τελικό αποτέλεσμα. Οπότε θα μελετηθούν διάφορα έτοιμα εργαλεια, όπως είναι το sparkler και το apache nutch, όμως και κάποια εργαλεία crawler, τα οποία για να γίνει η χρήση τους χρειάζεται και να τοποθετηθούν σε κάποια σημεία κώδικα.

Τέλος, όταν δημιουργείτε ένα εργαλείο ο δημιουργός του έχει κάποιο τελικό αποτέλεσμα σαν τελικό σκοπό της υλοποίησης, πολλές φορές όμως μπορεί να μην ταιριάζει στις υλοποιήσεις των υπόλοιπων εφαρμογών που θα χρησιμοποιηθούν στο μέλλον, με αποτέλεσμα να μην μπορει να γίνει η χρήση αυτών των εργαλείων και να πρέπει να δημιουργηθούν νέοι crawler απο την αρχή σχεδιασμένοι με βάση τις απαιτήσεις των χρηστών.

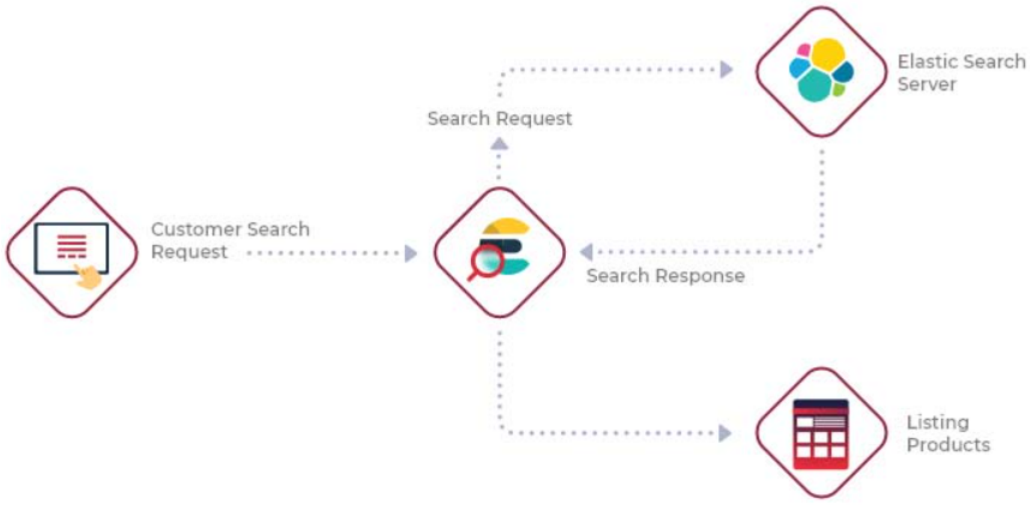
ΚΕΦΑΛΑΙΟ 2 - Εργαλεία και περιβάλλοντα

2.1 Εισαγωγή

Στο παρόν κεφάλαιο, γίνεται ανάλυση των εργαλείων και των περιβαλλόντων που χρησιμοποιούνται στην πορεία της διπλωματικής. Επεξηγώντας τα θα γίνει πιο εύκολη η κατανόηση τους, καθώς και η αναφορά τους στην συνέχεια θα είναι πιο οικεία.

2.2 Elasticsearch

Αρχικά, το ElasticSearch αναπτύχθηκε ως σύστημα αναζήτησης πλήρους κειμένου σε μεγάλου όγκου μη δομημένων δεδομένων. Μέχρι τώρα, το ElasticSearch είναι ένα πλήρες σύστημα με διάφορες δυνατότητες. Το ElasticSearch είναι μια κατανεμημένη μηχανή αναζήτησης και ανάλυσης.[6] Το ElasticSearch επίσης είναι μια βάση δεδομένων NoSQL, το οποίο σημαίνει ότι αποθηκεύει δεδομένα με μη δομημένο τρόπο για αυτό τον λόγο τα ερωτήματα που υποβάλλονται σε αυτό, δεν μπορούν να γίνουν με την χρήση της SQL. Τα δεδομένα στο ElasticSearch αποθηκεύονται σε ανεστραμμένη μορφή ευρετηρίου που βασίζεται στο Apache Lucene.[6]



Εικόνα 3: Κύρια χαρακτηριστικά του Elasticsearch

Το Elasticsearch (βασισμένο στο Apache Lucene) έχει ελαφρώς χαμηλότερη ταχύτητα ευρετηρίασης και αναζήτησης σε σύγκριση με το Sphinx, αλλά προσφέρει όχι μόνο αναζήτηση και αποθήκευση, αλλά περιέχει και άλλα εργαλεία (οπτικοποίηση, συλλογή αρχείων καταγραφής, σύστημα κρυπτογράφησης κ.λπ.) σε μεγάλα σύνολα ομαδοποιημένων δεδομένων που αντιστοιχούν στην αλληλεπίδραση των χρηστών με διάφορους πόρους πληροφοριών. Προτείνεται η χρήση των δυνατοτήτων του Elasticsearch για την οργάνωση της διεπαφής για εργασίες που σχετίζονται με τα Big Data (αναζήτηση και οπτικοποίηση), ενώ για την προκαταρκτική επεξεργασία και τις εργασίες τμηματοποίησης δεδομένων μπορεί να χρησιμοποιηθεί το μοντέλο MapReduce.[6]

2.3 Apache Lucene

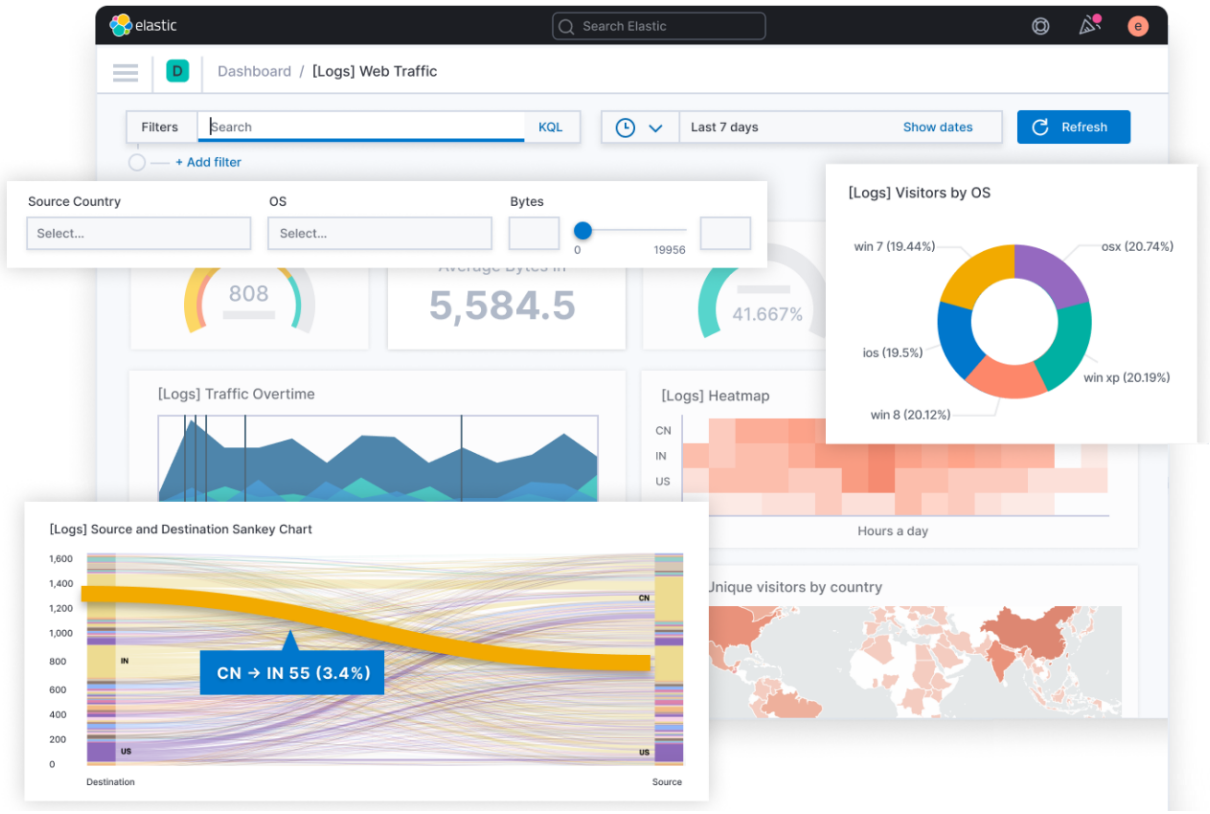
Το Apache Lucene είναι η πιο διάσημη μηχανή αναζήτησης. Το Lucene είναι μια βιβλιοθήκη για αναζήτηση πλήρους κειμένου υψηλής ταχύτητας, γραμμένη σε Java, που παρέχει προηγμένες δυνατότητες αναζήτησης. Το οποίο είναι ένα καλό σύστημα δημιουργίας ευρετηρίων και αποθήκευσης που μπορεί ταυτόχρονα να προσθέτει, να διαγράφει έγγραφα και να πραγματοποιεί βελτιστοποίηση μαζί με την αναζήτηση, καθώς και παράλληλη αναζήτηση σε ένα σύνολο ευρετηρίων που συνδυάζει τα αποτελέσματα. Όμως με σχετικά χαμηλή ταχύτητα ευρετηρίασης, καθώς και η έλλειψη API (για την οποία φροντίζει το Elasticsearch).[6]

Το Lucene έχει πολλά χαρακτηριστικά:

- Διαθέτει ισχυρούς, ακριβείς και αποτελεσματικούς αλγόριθμους αναζήτησης.
- Υπολογίζει μια βαθμολογία για κάθε έγγραφο σχετικά με το πόσο ταιριάζει με ένα δεδομένο ερώτημα (query) και επιστρέφει τα πιο σχετικά έγγραφα που ταξινομούνται με βάση τις βαθμολογίες.
- Υποστηρίζει πολλούς ισχυρούς τύπους ερωτημάτων (query), όπως PhraseQuery, WildcardQuery, RangeQuery, FuzzyQuery, BooleanQuery και άλλα.
- Υποστηρίζει την ανάλυση εμπλουτισμένων εκφράσεων ερωτημάτων (query) που εισάγονται από τον χρήστη.
- Επιτρέπει στους χρήστες να επεκτείνουν τη συμπεριφορά αναζήτησης χρησιμοποιώντας προσαρμοσμένη ταξινόμηση, φιλτράρισμα και ανάλυση εκφράσεων ερωτήματος.
- Χρησιμοποιεί μηχανισμό κλειδώματος που βασίζεται σε αρχεία για να αποτρέψει ταυτόχρονες τροποποιήσεις ευρετηρίου.
- Επιτρέπει την αναζήτηση και τη δημιουργία ευρετηρίου ταυτόχρονα.

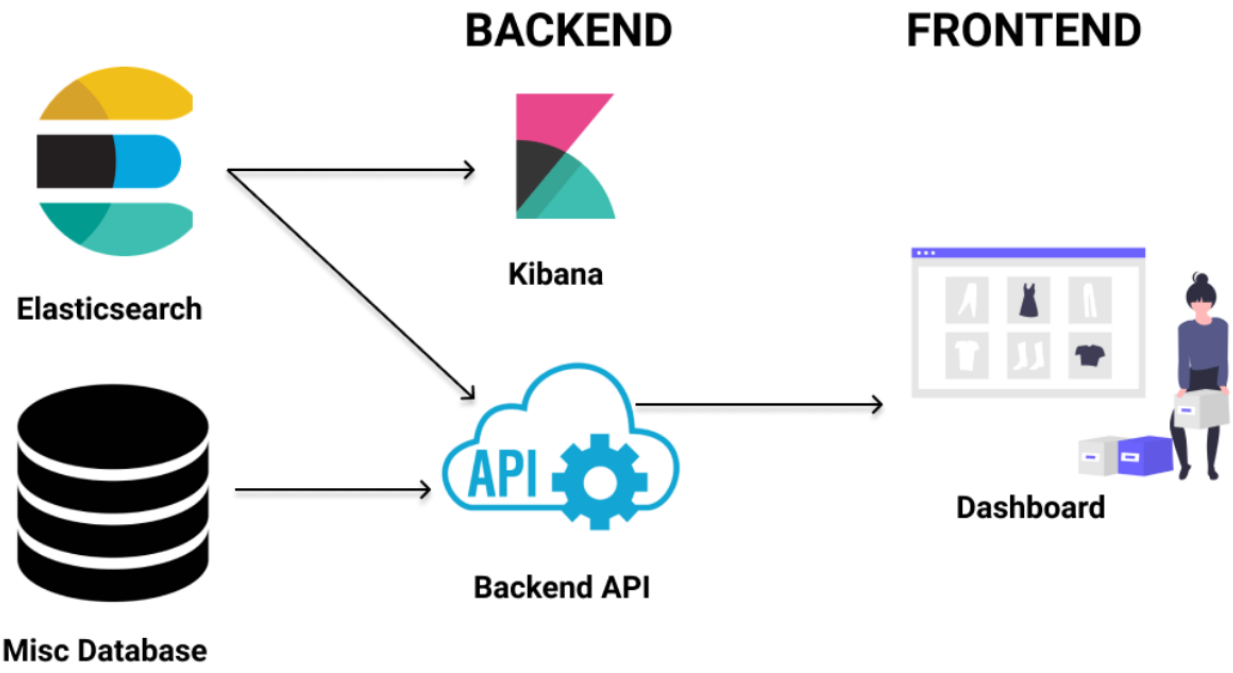
2.4 Kibana

Το Kibana είναι μια δωρεάν και ανοιχτού κώδικα εφαρμογή frontend που βρίσκεται στην κορυφή του Elastic Stack. Παρέχει δυνατότητες αναζήτησης και οπτικοποίησης δεδομένων για δεδομένα που έχουν αποθηκευτεί στο ElasticSearch. Προσφέρει εύχρηστες λειτουργίες όπως ιστογράμματα, γραφήματα γραμμής, γραφήματα πίτας, χάρτες θερμότητας και ενσωματωμένη γεωχωρική υποστήριξη.



Εικόνα 4: Dashboard του ElasticSearch

Το Kibana είναι ένα διαδικτυακό εργαλείο οπτικοποίησης που ενσωματώνεται με το Elasticsearch για να παρέχει εύκολους τρόπους πλοήγησης και οπτικοποίησης δεδομένων, χρησιμοποιώντας μια ποικιλία γραφημάτων, διαγραμμάτων και πινάκων. Έτσι, όταν οι χρήστες εκτελούν ερωτήματα στο Kibana δεν βλέπουν ζωντανά δεδομένα αλλά συγκεντρωτικά δεδομένα από τις προηγούμενες ημέρες.[7]

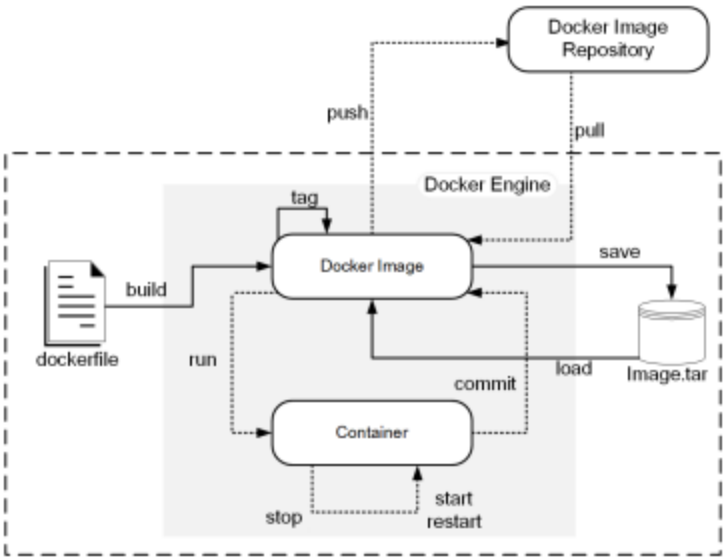


Εικόνα 5: Επικοινωνία ElasticSearch με άλλα εργαλεία

2.5 Docker

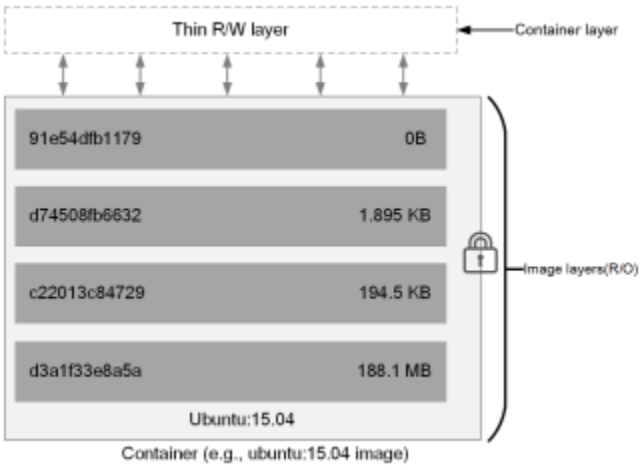
Το Docker είναι πλατφόρμα εφαρμογών ως υπηρεσία (platform as a service - PaaS) που χρησιμοποιούν εικονοποίηση σε επίπεδο λειτουργικού συστήματος για την παράδοση λογισμικού σε πακέτα-container. Έχει την δυνατότητα να κοινοποιεί με ταχύτητα περιβάλλοντα δημιουργίας εφαρμογών μεταξύ διαφόρων προγραμματιστών κάνοντας χρήση της τεχνολογίας Container. Το λογισμικό στο οποίο φιλοξενούνται τα containers ονομάζεται Docker Engine. Η ασφάλεια που προσφέρει περιέχει κάποια τρωτά σημεία τα οποία καθιστούν το Docker από τα εργαλεία με μικρή ασφάλεια. Αυτό βασίζεται στο ότι οι εικόνες του Docker μοιράζονται χωρίς κάποιο μέσο ασφάλειας. Η πλατφόρμα του Docker περιλαμβάνει UI, CLI, API και ασφάλεια που έχουν σχεδιαστεί για να συνεργάζονται σε ολόκληρο τον κύκλο ζωής της εφαρμογής.[8]

Το Docker κατασκευάζει ένα περιβάλλον container με βάση τις εικόνες του Docker. Το Docker μαζεύει όλα τα αρχεία που χρειάζονται για εφαρμογή όπως είναι οι βιβλιοθήκες, το ενδιάμεσο λογισμικό, το λειτουργικό σύστημα, η διαμόρφωση δικτύου κ.λπ. σε μια εικόνα Docker. Η εικόνα Docker μεταφέρονται σε ένα απομακρυσμένο χώρο αποθήκευσης που ονομάζεται αποθήκη εικόνων Docker και χρησιμοποιείται για την κοινή χρήση των εικόνων Docker μεταξύ των χρηστών. Στην περίπτωση του αποθετηρίου, μπορεί να διαμορφωθεί ως δημόσια ή ιδιωτικά αποθετήρια. Εάν ο προγραμματιστής θέλει να ανεβάσει μια εικόνα Docker, τότε αυτό μπορεί να γίνει σε ιδιωτικό ή δημόσιο επίπεδο. Αυτό καθορίζεται από τις παραμέτρους θα δοθούν στην εντολή δημιουργίας της εικόνας.[8]



Εικόνα 6: Docker image

Η πλατφόρμα Docker δίνει την δυνατότητα να μετατραπεί μια εικόνα Docker σε Docker-container εκτελώντας απλά την εντολή run που προσφέρει. Όταν η εντολή εκτελεστεί το Docker αντιγράφει αρχικά την εικόνα στα αρχεία του συστήματος που διαχειρίζεται το ίδιο, προσθέτοντας ένα στρώμα στην κορυφή αυτής προκειμένου να δημιουργηθεί ένα container. Στην συνέχεια το Docker εμφανίζει μια ολοκληρωμένη προβολή κάνοντας χρήση το σύστημα αρχείων κάνοντας τον χρήστη να αντιλαμβάνεται ότι είναι σε κάποιο ενιαίο σύστημα. Μέσα από αυτό το σύστημα (container) μπορεί ο προγραμματιστής να εκτελέσει όλες τις ενέργειες που μπορεί να κάνει και σε ένα κανονικό λειτουργικό σύστημα όπως είναι εγκατάσταση, ενημέρωση, αντιγραφή, διαγραφή κλπ. [8]



Εικόνα 7: Docker container

2.6 Scala

Η Scala είναι μια γλώσσα προγραμματισμου υψηλού επιπέδου που υποστηρίζει τόσο αντικειμενοστραφή προγραμματισμό όσο και λειτουργικό προγραμματισμό. Έχει σχεδιαστεί για

να μπορεί να αναπτύσεται παράλληλα με τις ανάγκες των χρηστών της. Επίσης έχει το ίδιο μοντέλο μεταγλώττισης με τη C# και την Java, δηλαδή ξεχωριστή μεταγλώττιση και δυναμική φόρτωση κλάσεων. Συνεπώς ο κώδικας που είναι υλοποιημένος σε Scala μπορεί να καλεί βιβλιοθήκες της Java ή βιβλιοθήκες της πλατφόρμας .NET.

Η Scala έχει σχεδιαστεί για να αλληλεπιδρά καλά με mainstream πλατφόρμες όπως η Java ή η C#. Μοιράζεται με αυτές τις γλώσσες τους περισσότερους βασικούς τελεστές, τύπους δεδομένων και δομές ελέγχου. Η Scala είναι μια λειτουργική γλώσσα με την έννοια ότι κάθε συνάρτηση είναι μια τιμή. Παρέχει μια ελαφριά σύνταξη για τον ορισμό ανώνυμων και καθορισμένων συναρτήσεων και υποστηρίζει επίσης ένθετες συναρτήσεις. [9]

Στη Scala, ο προγραμματιστής έχει την επιλογή μεταξύ της σύνθεσης αφαιρέσεων (abstractions) κατά το χρόνο εκτέλεσης χρησιμοποιώντας σύνθεση αντικειμένου ή κατά τη μεταγλώττιση χρησιμοποιώντας σύνθεση κλάσης. Το αν κάποιος χρησιμοποιεί σύνθεση αντικειμένου ή κλάσης εξαρτάται κυρίως από τις ιδιαίτερες απαιτήσεις ευελιξίας και ασφάλειας. Έχει πλούσια σύνταξη και σύστημα τύπων, που συνδυάζει έννοιες από αντικειμενοστρεφή προγραμματισμό και λειτουργικό προγραμματισμό. Η Scala έχει ορισμούς αντικειμένων (ξεκινώντας με αντικείμενο) εκτός από ορισμούς κλάσεων. Ένας ορισμός αντικειμένου ορίζει μια κλάση με ένα μόνο στιγμιότυπο - αυτό μερικές φορές ονομάζεται αντικείμενο singleton. [9]

2.7 Apache Maven

Το Apache Maven είναι ένα εργαλείο κατασκευής και διαχείρισης λογισμικού. Βασισμένο στην ιδέα ενός μοντέλου αντικειμένου έργου (POM), μπορεί να διαχειριστεί την δημιουργία, την αναφορά και την τεκμηρίωση ενός project από μια κεντρική πληροφορία. Χρησιμοποιείται κυρίως σε Java project. Ο πρωταρχικός στόχος του Maven είναι να βοηθήσει τους προγραμματιστές java να δημιουργήσουν και να διαχειριστούν οποιαδήποτε έργα που βασίζονται σε Java και να κάνουν την καθημερινή τους εργασία ευκολότερη και κατανοητή. Επίσης, υποστηρίζει plug-in και είναι αποδεκτές οι αναβαθμίσεις σε νέες εκδόσεις και δυνατότητες. [10]

Η Maven βοηθά επίσης έργα για διαχείριση εκδόσεων και παρακολούθηση προβλημάτων. Προωθεί επίσης μια συγκεκριμένη διάταξη για το έργο Java, έτσι ώστε μόλις ο προγραμματιστής εξοικειωθεί με τη διάταξη, να μπορεί εύκολα να κατανοήσει οποιοδήποτε άλλο έργο χρησιμοποιεί Maven. Η Maven έχει δύο σημαντικούς ρόλους στη διαχείριση εκδόσεων και την CI:

- **Διαχείριση διανομής**: Κάθε έργο που πρόκειται να υποστηρίξει τη συνεχή ενοποίηση θα πρέπει να έχει διαμορφώσει τις ρυθμίσεις διαχείρισης διανομής στο αρχείο POM του. Με αυτές τις ρυθμίσεις η Maven θα καταλάβει ότι μετά τη διαδικασία κατασκευής σε ποιό σημείο θα πρέπει να αποθηκευτεί το δυαδικό αρχείο, το οποίο μπορεί να είναι οποιοδήποτε απομακρυσμένο ή τοπικό αποθετήριο.[10]
- **Ρυθμίσεις αποθετηρίου στιγμιότυπου**: Υπάρχουν μερικές ακόμη ρυθμίσεις που χρησιμοποιεί η Maven για να επικοινωνεί με τα αποθετήρια. Το ένα είναι η πολιτική ενημέρωση, η οποία θα προσδιορίζει πόσο συχνά πρέπει να ελέγχει η Maven για την πιο πρόσφατη έκδοση των εξαρτήσεων. Δεδομένου ότι τα API προορίζονται να αλλάζουν

συχνά, αυτή η ρύθμιση πρέπει να ρυθμιστεί ανάλογα. Μια άλλη σημαντική ρύθμιση είναι τα στοιχεία διακομιστή, η οποία διατηρεί τα στοιχεία για την πρόσβαση του Maven στο απομακρυσμένο ή τοπικό αποθετήριο. Ο κωδικός πρόσβασης μπορεί να κρυπτογραφηθεί με βάση τον οδηγό κρυπτογράφησης κωδικού πρόσβασης του Maven.[10]

```
<project>
 <modelVersion>4.0.0</modelVersion>

 <groupId>com.mycompany.app</groupId>
 <artifactId>my-app</artifactId>
 <version>1</version>
</project>
```

2.8 Apache Ant

Το Apache Ant είναι μια βιβλιοθήκη Java και ένα εργαλείο γραμμής εντολών ανοιχτού κώδικα και είναι γραμμένο σε Java. Μπορεί να χρησιμοποιηθεί από γλώσσες προγραμματισμού που βασίζονται στο JVM. Χρησιμοποιείται για την αυτοματοποίηση των διαδικασιών κατασκευής λογισμικού όπως η μεταγλώττιση, εκτέλεση, δοκιμή και συναρμολόγηση εφαρμογής Java

Το Ant είναι ένα εργαλείο αυτοματισμού κατασκευής που χρησιμοποιείται συχνά για την υποστήριξη συνεχούς ενοποίησης (CI). Η CI είναι μια πρακτική ανάπτυξης λογισμικού στην οποία τα μέλη μιας ομάδας ενσωματώνουν συχνά την εργασία τους. Η ενοποίηση περιλαμβάνει την εκτέλεση εργασιών που στοχεύουν στην κατασκευή και τον έλεγχο του λογισμικού. Το Ant επινοήθηκε ως ένα πλαίσιο για την οργάνωση αφηρημένων εργασιών CI με τέτοιο τρόπο ώστε η προσθήκη νέων τύπων εργασιών να είναι εύκολη. Χρησιμοποιεί αρχεία XML για να περιγράψει ποιες εργασίες πρέπει να εκτελούνται και με ποια σειρά, κατά την ενσωμάτωση. Η αποδοχή του Apache Ant από τους προγραμματιστές δείχνει ότι μπορεί να αντιμετωπίσει ένα πραγματικό πρόβλημα και είναι μια αποδεκτή λύση.[11]

Το Ant είναι ένα πλαίσιο για συνεχή ενοποίηση και η δυνατότητα ενσωμάτωσης νέων αυτοματοποιημένων εργασιών ανάπτυξης λογισμικού αποτελεί απαραίτητο χαρακτηριστικό για να παραμείνει το σύστημα χρήσιμο. Ως εργαλείο αυτοματισμού κατασκευής, συνήθως εκτελείται χωρίς γραφικό περιβάλλον χρήστη, λαμβάνοντας παραμέτρους απευθείας από τη γραμμή εντολών. Αυτό το χαρακτηριστικό του επιτρέπει να ενσωματώνεται εύκολα σε διαφορετικά περιβάλλοντα. Έτσι, πολλά έργα ανοιχτού κώδικα και βιομηχανικά έργα έχουν υιοθετήσει το Ant, δημιουργώντας την ανάγκη να εξελίσεται συνεχώς το λογισμικό και να διορθώνονται τα ελαττώματα που εντόπισαν οι χρήστες.[11]

ΚΕΦΑΛΑΙΟ 3 - Apache Nutch

3.1 Εισαγωγή

Σε αυτό το κεφάλαιο, γίνεται η περιγραφή του Apache Nutch και η ανάλυση του. Στην συνέχεια περιγράφονται κάποια βασικά σημεία του, τα οποία είναι η αρχιτεκτονική του και η τεχνική των ερωτημάτων - quering. Στα παράρτημα 1 υπάρχει και η διαδικασία εγκατάστασης του Apache Nutch. Τέλος παρουσιάζονται τα προβλήματα και τα συμπεράσματα που προέκυψαν απο αυτή την έρευνα-εγκατάσταση.

3.2 Περιγραφή

Το Apache Nutch είναι ένας επεκτάσιμος και υψηλής απόδοσης web crawler. Το Apache Nutch είναι κωδικοποιημένο εξ ολοκλήρου στη γλώσσα προγραμματισμού Java, όμως τα δεδομένα δεν έχουν κάποιον περιορισμό στην γλώσσα που θα είναι. Η τελευταία έκδοση που είναι διαθέσιμη είναι πολύ ώριμη σαν αντικείμενο και είναι έτοιμη για παραγωγικούς web crawlers. Το Apache Nutch μπορεί επίσης να εξάγει περιεχόμενο κειμένου από διάφορες μορφές εγγράφων όπως HTML, RSS, ATOM, PDF, μορφές ms (doc, excel, ppt). Υποστηρίζει, επίσης, ανάκτηση περιεχομένου με διάφορα πρωτόκολλα όπως HTTP, HTTPS, FTP.

Το Apache Nutch είναι μια από τις μηχανές αναζήτησης ανοιχτού κώδικα, οπότε ο κάθε χρήστης μπορεί να το επεκτείνει. Οι τεχνολογίες ευρετηρίασης και κατάταξης σελίδας είναι όλες ανοιχτές και ορατές. Το Apache Nutch βασίζεται σε plug-in και πολλά ερευνητικά έργα που βασίζονται στο Apache Nutch . Ειναι μια ολοκληρωμένη μηχανή αναζήτησης, παρέχει όλα τα εργαλεία που είναι απαραίτητα για αυτο. [1]

1. **Διαφάνεια**: Το Apache Nutch είναι ανοιχτού κώδικα, οπότε ο καθένας μπορεί να δει πώς λειτουργούν οι αλγόριθμοι κατάταξης. Με τις μηχανές αναζήτησης που βρίσκονται στην αγορά, οι ακριβείς λεπτομέρειες των αλγορίθμων είναι μυστικές, επομένως δεν είναι δυνατόν ποτέ να μάθετε πως και γιατί ένα συγκεκριμένο αποτέλεσμα αναζήτησης κατατάσσεται.
2. **Κατανόηση**: Δεν υπάρχει ο πηγαίος κώδικας της Google δημόσια, οπότε το Apache Nutch είναι ίσως το καλύτερο εργαλείο. Το Apache Nutch έχει κατασκευαστεί χρησιμοποιώντας ιδέες από τον ακαδημαϊκό χώρο και τη βιομηχανία, το οποίο προέκυψε από τα Εργαστήρια Google. Το Apache Nutch είναι ελκυστικό για ερευνητές που θέλουν να δοκιμάσουν νέους αλγόριθμους αναζήτησης, καθώς είναι τόσο εύκολο να επεκταθεί.
3. **Επεκτασιμότητα**: Το Apache Nutch είναι πολύ ευέλικτο: μπορεί να προσαρμοστεί και να ενσωματωθεί σε οποιαδήποτε εφαρμογή. [1]

3.3 Αρχιτεκτονική

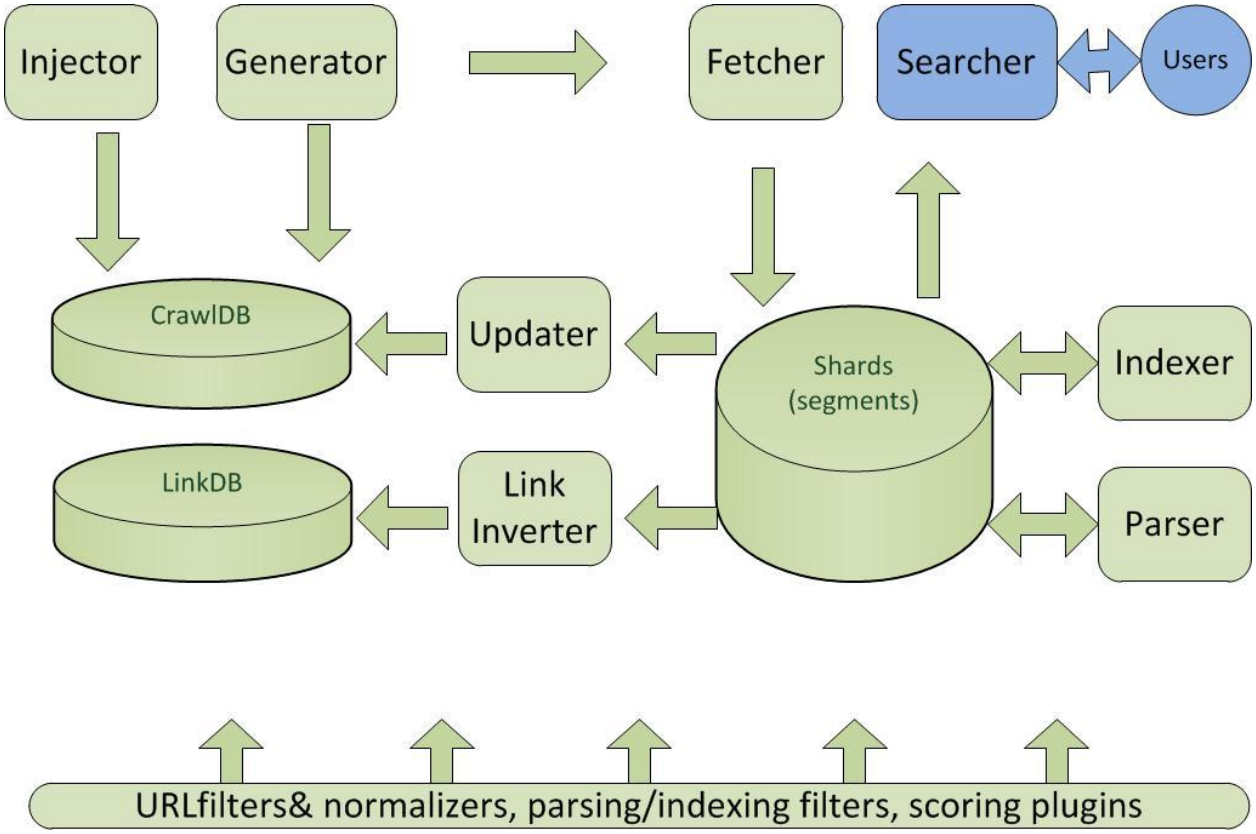
Το Apache Nutch έχει μια εξαιρετικά αρθρωτή αρχιτεκτονική που χρησιμοποιεί plug-in APIs για ανάλυση τύπου πολυμέσων, ανάλυση HTML, πρωτόκολλα ανάκτησης δεδομένων και ερωτήματα. Έχει τέσσερα κύρια συστατικά: [2]

- Αναζήτηση: Με δεδομένο ένα ερώτημα που γίνεται, πρέπει να βρει γρήγορα ένα σχετικό υποσύνολο ενός σώματος εγγράφων και στη συνέχεια να παρουσιαστούν. Η εύρεση ενός μεγάλου σχετικού υποσυνόλου γίνεται συνήθως με έναν ανεστραμμένο δείκτη του σώματος, κατάταξη σε αυτό το σύνολο για την παραγωγή των πιο σχετικών εγγράφων, τα οποία στη συνέχεια πρέπει να συνοψιστούν για εμφάνιση.
- Indexer: Δημιουργεί το ευρετήριο από το οποίο εξάγει τα αποτελέσματα ο ερευνητής (searcher). Χρησιμοποιεί ευρετήρια αποθήκευσης Lucene.
- Βάση δεδομένων: Αποθηκεύει τα περιεχόμενα του εγγράφου για ευρετηρίαση(indexing) και στην συνεχεια σύνοψη από τον ερευνητή(searcher), μαζί με πληροφορίες όπως η δομή συνδέσμων του εγγράφου και η ώρα λήψης κάθε εγγράφου για τελευταία φορά.
- Fetcher: Ζητάει ιστοσελίδες, τις αναλύει και εξάγει συνδέσμους από αυτές. Το ρομπότ του Apache Nutch έχει γραφτεί εξ ολοκλήρου από την αρχή.

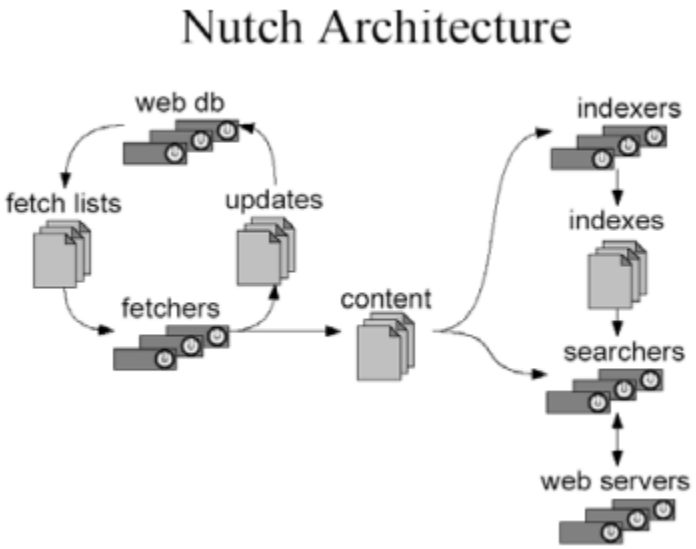
Βήματα στη μηχανή αναζήτησης Apache Nutch (Ανίχνευση και ευρετήριο) [1]

1. Δημιουργία νέου CrawlDB (admin db -create).
2. Εισαγωγή των διευθύνσεων URL στο CrawlDB (inject).
3. Αρχικοποίηση λίστας ανάκτησης από το CrawlDB σε ένα νέο τμήμα (generate).
4. Λήψη περιεχομένου από διευθύνσεις URL στη λίστα ανάκτησης (fetch).
5. Ενημέρωση της CrawlDB με συνδέσμους από σελίδες που ανακτήθηκαν (updatedb).
6. Ενημέρωση τμημάτων με βαθμολογίες και συνδέσμους από το CrawlDB (updatesegs).
7. Δημιουργία ευρετηρίου των σελίδων που ανακτήθηκαν (index).
8. Αφαιρεση διπλοεγγραφών (και τις διπλές διευθύνσεις URL) από τα ευρετήρια (dedup).
9. Συγχώνευση των ευρετηρίων σε ένα ενιαίο ευρετήριο για αναζήτηση (merge).

Το παρακάτω σχήμα περιγράφει τις σχέσεις μεταξύ των στοιχείων που αναφέρονται το ένα στο άλλο, τοποθετώντας τα στο ίδιο πλαίσιο, και εκείνων από τα οποία εξαρτώνται σε ένα χαμηλότερο επίπεδο. Για παράδειγμα, το πρωτόκολλο δεν εξαρτάται από το δίκτυο, επειδή το πρωτόκολλο είναι μόνο ένα σημείο διασύνδεσης για πρόσθετα που παρέχουν στην πραγματικότητα μεγάλο μέρος της λειτουργικότητας του Apache Nutch. [2]



Εικόνα 8: Σχέσεις μεταξύ των στοιχείων του Nutch

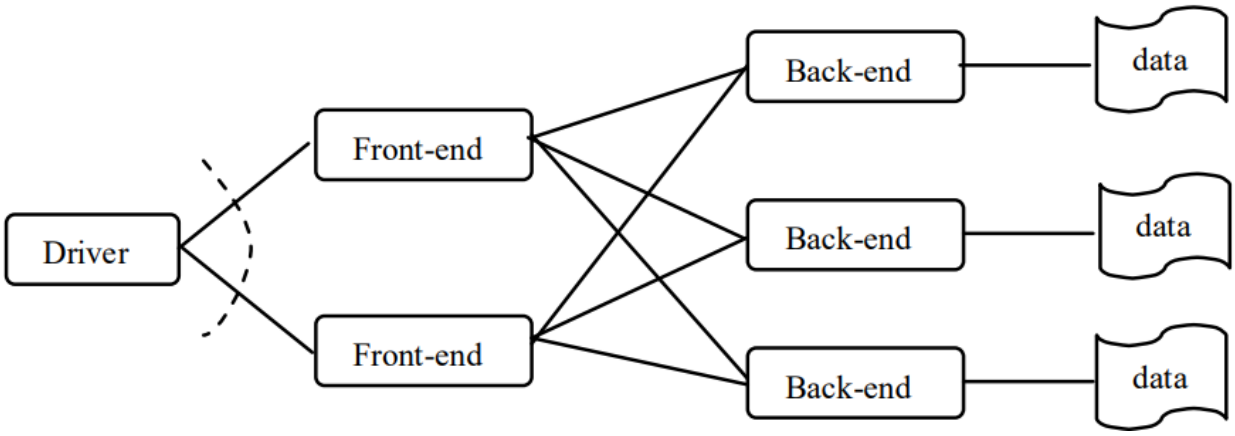


Εικόνα 9: Αρχιτεκτονική Nutch

3.4 Ερωτήματα - Quering

Στις περισσότερες εφαρμογές αναζήτησης, το ερώτημα θεωρείται ένα από τα πιο σημαντικά κομμάτια της διαδικασία και της συνολικής προσπάθειας. Κατά την εκτέλεση

ενός ερωτήματος, ένα σύνολο όρων ευρετηρίασης παρουσιάζεται σε μια μηχανή ερωτημάτων, η οποία στη συνέχεια ανακτά τα έγγραφα που ταιριάζουν καλύτερα με αυτό το σύνολο όρων. Η συνολική αρχιτεκτονική της μηχανής παράλληλων ερωτημάτων Apache Nutch φαίνεται στην παρακάτω εικόνα. Το τμήμα της μηχανής ερωτημάτων αποτελείται από από πολλά back end και front end κομμάτια. Κάθε back-end έχει άμεση επαφή με ένα τμήμα του πλήρους συνόλου δεδομένων. Το πρόγραμμα οδήγησης αντιπροσωπεύει εξωτερικούς χρήστες και είναι το σημείο στο οποίο μετριέται η απόδοση του ερωτήματος, ως προς τα ερωτήματα ανά δευτερόλεπτο (qps). [2]



Εικόνα 10: Ερωτήματα Nutch

3.5 Συμπεράσματα

Μετα την μελέτη των αποτελεσμάτων που αποθηκεύονται στο elasticsearch, διαπιστώθηκε πως το content περιλάμβανε όλο το περιεχόμενο της σελίδας σαν κείμενο. Αυτο σημαίνει πως δεν υπήρχε καμία ετικέτα - tag, για αυτό τον λόγο αναζητήθηκε τρόπος να αποθηκεύεται το html περιεχόμενο της κάθε σελίδας. Συμπερασματικά, η αλλαγή που πραγματοποιεί αυτή την τροποποίηση του περιεχομενου ηταν να αλλαχτεί ένα plugin του html parser. Η μοναδική αλλαγή αυτή έγινε στο αρχείο `nutch/src/plugins/parse-html/src/java/org/apache/nutch/parse/html/HtmlParser.java` στην γραμμή 255. Η μια παράμετρο του ParseImpl, η οποία είναι το `text` σε `new String(content.getContent())`.

Όμως πέρα απο την αλλαγή του τύπου της τιμής του content, υπήρξε και άλλο πρόβλημα στην συνέχεια. Το Apache Nutch λειτουργεί σύμφωνα με το robot.txt του κάθε site, οπότε μέσω αυτού δίνεται πρόσβαση σε κάποιες σελίδες και σε κάποιες άλλες οχι. Αυτό σημαίνει ότι δεν επιτρέπεται να γίνει λήψη των πληροφοριών που χρειάζονται για την τελική εφαρμογή. Έγινε η προσπάθεια να απενεργοποιηθεί αλλά μάταια.

ΚΕΦΑΛΑΙΟ 4 - Sparkler

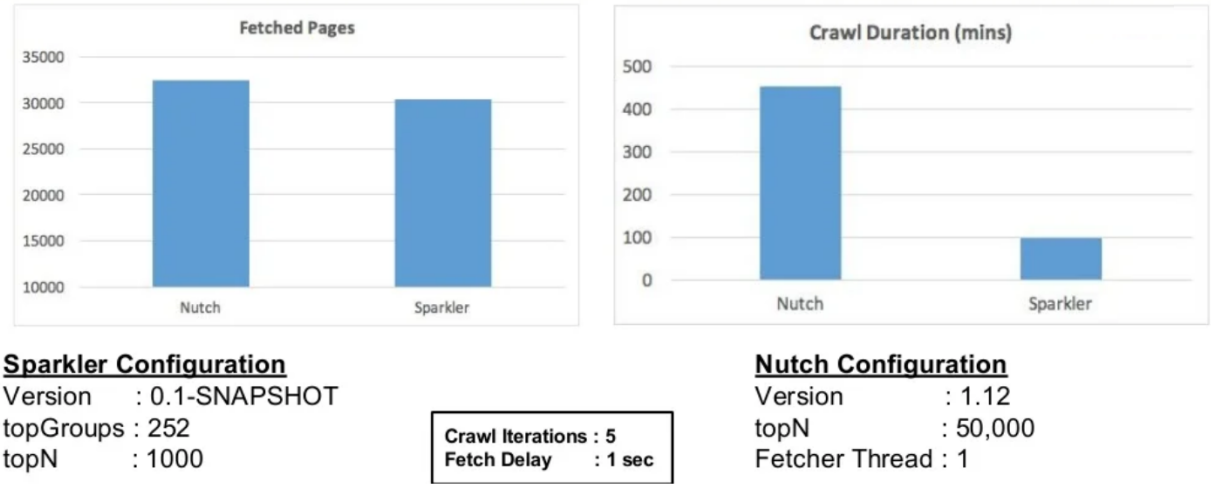
4.1 Εισαγωγή

Σε αυτό το κεφάλαιο, γίνεται η περιγραφή του Sparkler και η ανάλυση του. Στην συνέχεια περιγράφεται η αρχιτεκτονική του. Στα παράρτημα 2 υπάρχει η προσπάθεια εγκατάστασης του Sparkler. Τέλος παρουσιάζονται τα προβλήματα και τα συμπεράσματα που προέκυψαν απο αυτή την έρευνα-εγκατάσταση.

4.2 Περιγραφή

Το Sparkler είναι ένας επεκτάσιμος, εξαιρετικά scalable και υψηλής απόδοσης web crawler ο οποίος είναι εξέλιξη του Apache Nutch. Αξιοποιεί τις εξελίξεις στους τομείς κατανεμημένων υπολογιστών (όπως είναι το Spark) και την ανάκτηση πληροφοριών (όπως είναι το elasticsearch ή το Solr). Επεκτείνει τη λειτουργικότητα του Apache Spark, παράλληλα με άλλα έργα Apache, όπως τα Kafka, Lucene/Solr, Elasticsearch, Tika και pf4j. Ένα από τα πλεονεκτήματα της χρήσης του Sparkler είναι οτι έχει υψηλή απόδοση, σε συνδυασμό με την ανάλυση σε πραγματικό χρόνο, η οποία επιτρέπει ελεγχόμενες ανιχνεύσεις μεγάλης κλίμακας. Το Sparkler διαθέτει επίσης ένα επεκτάσιμο πλαίσιο προσθήκης και διατίθεται προσυσκευασμένο με πολλά χρήσιμα πρόσθετα, συμπεριλαμβανομένου ενός πρόσθετου για απόδοση JavaScript, το οποίο επιτρέπει την αναζήτηση ιστοσελίδων στην τελική τους κατάσταση απόδοσης. [4]

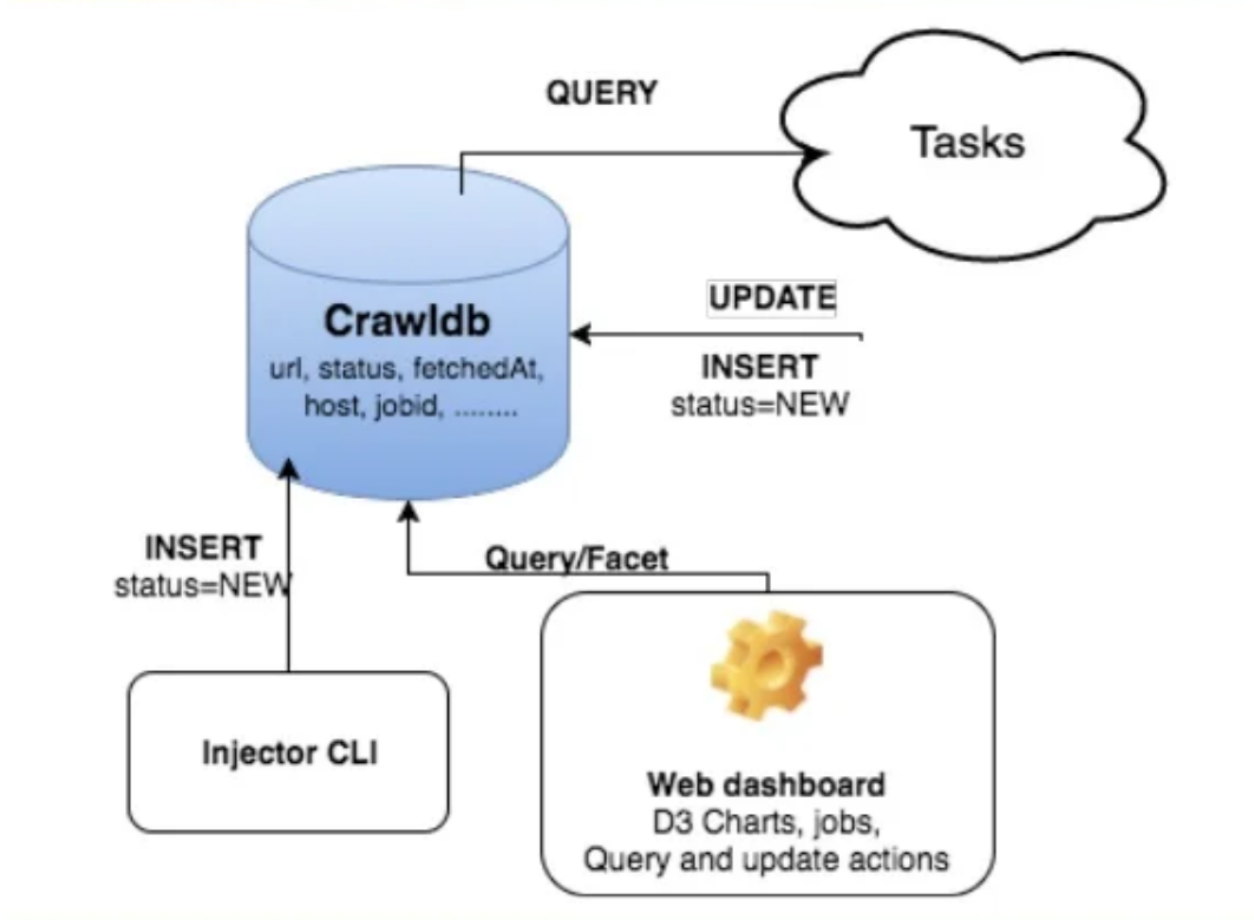
Το Sparkler Crawl Environment (SCE) είναι ένα σύνολο εργαλείων που έχουν δημιουργηθεί πάνω από το Sparkler που κατέχει μια αποτελεσματική αρχιτεκτονική λογισμικού που χρησιμοποιείται για τον εμπλουτισμό ενός τομέα επεκτείνοντας συλλογή αντικειμένων. Παρέχει ένα περιβάλλον γραμμής εντολών με βάση το Docker (CLI) για την κατασκευή και εκτέλεση εργασιών. Λόγω του ότι η επεκτασιμότητα ήταν χαρακτηριστικό και των δύο, χρησιμοποιήθηκε το Sparkler και το SCE για πειραματικές ανιχνεύσεις PDF, χωρίς να υπάρχει πάντα ένα επιτυχημένο αποτέλεσμα. Το Sparkler χρησιμεύει ως λύση για ανίχνευση κατ' απαίτηση και συλλογή αρχείων PDF. Δεδομένου ότι δεν προϋποθέτει προηγούμενη γνώση των τοποθεσιών που ανιχνεύει, είναι πολύ λιγότερο αποτελεσματικό από μεμονωμένους scrapers ιστοτόπων και δεν μπορεί εύκολα να διασχίσει ιστότοπους που έχουν δημιουργηθεί γύρω από τα API αναζήτησης. [4] Η ταχύτητα που έχει το Sparkler σε σύγκριση με το Apache Nutch είναι σαφώς μεγαλύτερη.[5]



Εικόνα 11: Σύγκριση Apache Nutch and Sparkler

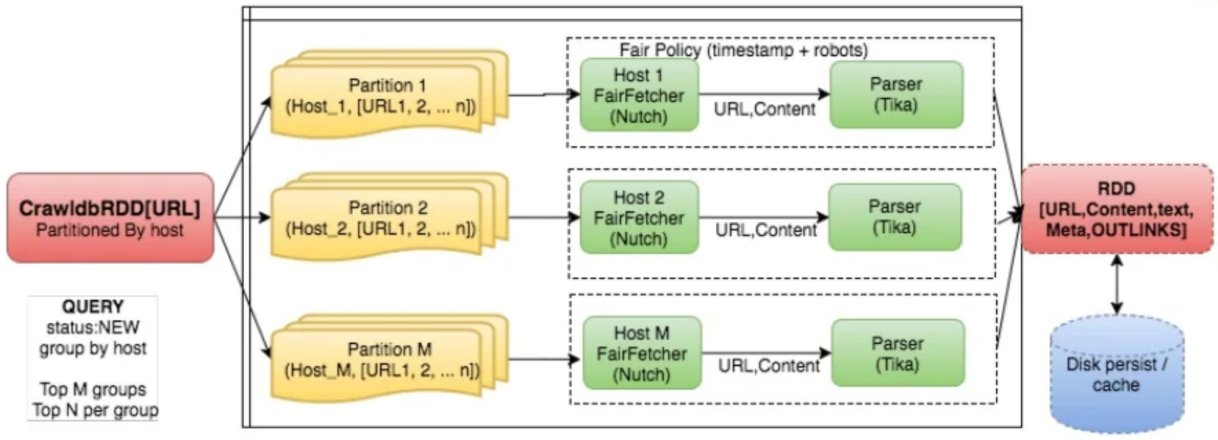
4.3 Αρχιτεκτονική

Αρχικά, χρειάζεται μια βάση στην οποία θα αποθηκεύονται τα δεδομενα. Αυτη η βάση μπορεί να είναι είτε elasticsearch είτε sorl. Στην οποία ο διαχειριστής θα πρεπει να φτιάξει τα jobs και να κανει τις κατάλληλες εισαγωγές/τροποποιήσεις στην βάση μέσω του dashboard. Ταυτόχρονα γίνονται οι εισαγωγές/τροποποιήσεις στην βάση μέσω των ενεργειών που εκτελούνται. Και τέλος κάνοντας κάποια ερωτήματα στην βάση επιστρέφονται κάποιες εργασίες που πρέπει να εκτελεστούν στην πορεία.[5]



Εικόνα 12: Sparkler - Crawldb

Οι παραπάνω εργασίες που είναι απαραίτητο να εκτελεστούν στην συνέχεια, χωρίζονται κατανεμημένα και γίνεται η ανίχνευση(crawling) στο κάθε ένα. Τέλος αποθηκεύονται προσωρινά στην cache.[5]

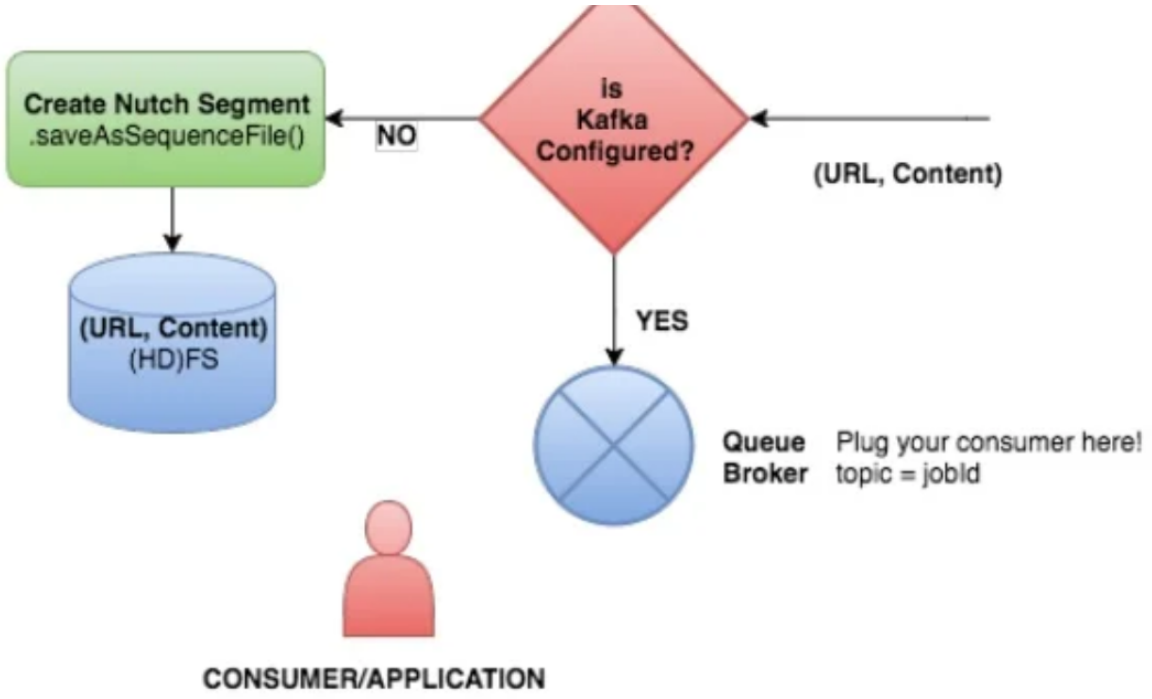


Εικόνα 13: Sparkler - RDD

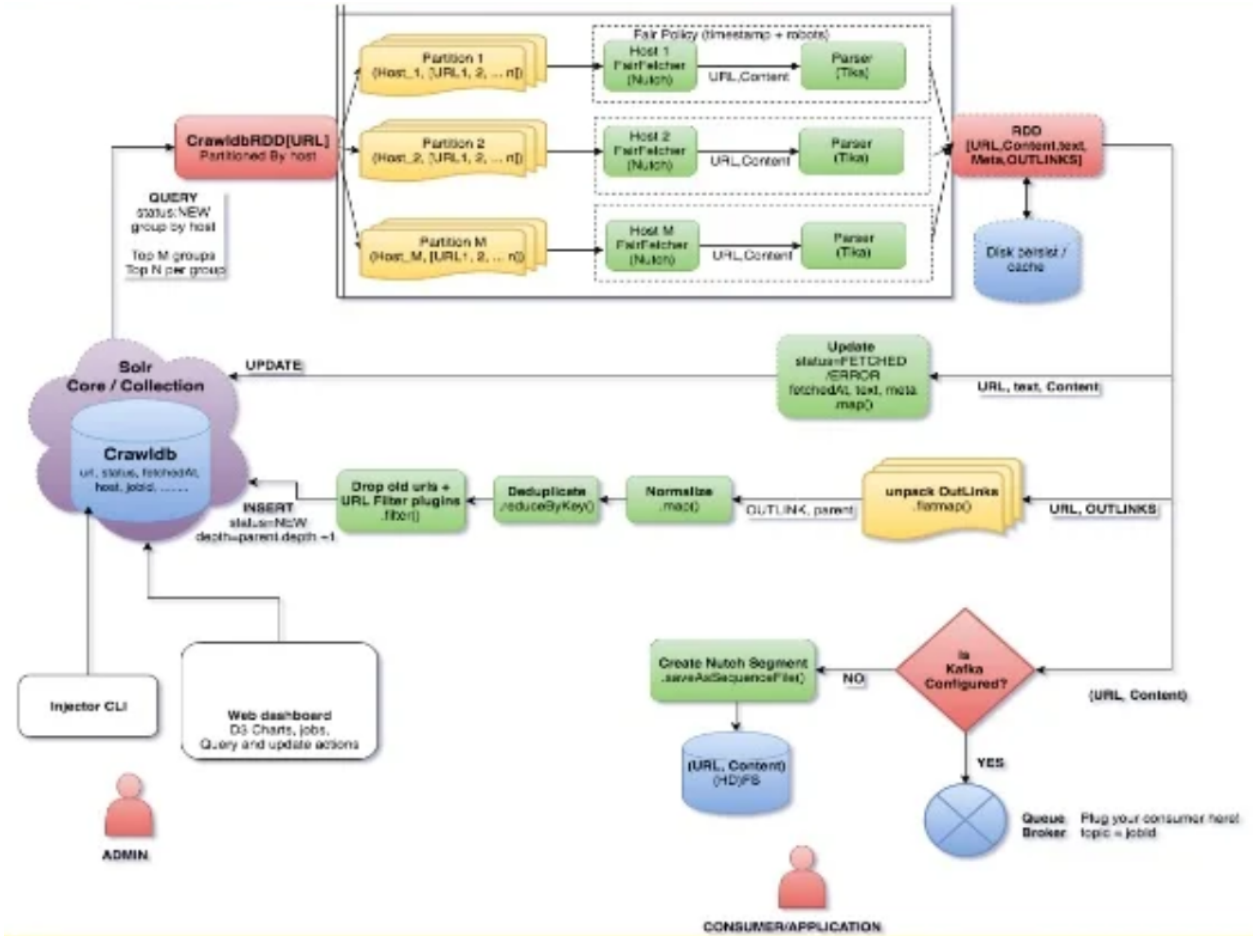
Στην πορεία γίνεται ένα ξεσκαρτάρισμα των νέων διευθύνσεων (urls) και διαλεγει ποια υπάρχουν ήδη ώστε να τα διαγράψει, για να μην ξαναπεράσουν την διαδικασία του rdd και ποιά να κρατήσει και να συνεχίσει σε ανίχνευση τους.[5]

Εικόνα 14: Sparkler - Links pipeline

Στο τέλος της διαδικασίας ελέγχει αν είναι ρυθμισμένο το kafka, στην περίπτωση που είναι, ο “πελάτης” λαμβάνει την ουρά με τα αποτελέσματα, αλλιώς δημιουργεί ένα τμήμα με τα αποτελέσματα αυτα και τα αποθηκεύει σε ενα αρχείο. [5]



Εικόνα 15: Sparkler - Output consumption



Εικόνα 16: Sparkler - Workflow

4.4 Συμπεράσματα

Το tika και το kafka είναι έτοιμα να λειτουργήσουν, αρκεί να γίνει η ενεργοποίηση τους απο το κατάλληλο αρχείο. Το kafka επίσης δημιουργεί ένα topic ανα job με το όνομα sparkler_<job_id>. Το banana είναι μονο για sorl και το αντίστοιχο του elasticseach είναι το kibana - το οποίο εφαρμόστηκε κιολας στην εγκατάσταση. Για να βρούμε χρήσιμη πληροφορία, σε κατάλληλο format, χρησιμοποιούν beautiful soup, scrapy και άλλα παρόμοια εργαλεία.

Το repository του sparkler στο git δεν τρέχει σωστά το crawling της σελίδας που δίνεται, αυτό διαπιστώνεται και απο το μήνυμα `build failed` κατα την διάρκεια του builld οτι δεν ολοκληρώνεται επιτυχώς. Μελετώντας το git repository του sparkler, είναι προφανές πως δεν είναι έτοιμο για παραγωγικότητα. Αρχικά, έχει παρα πολλά ανοιχτά issues, δείχνοντας έτσι ότι δεν έχει τελείωσει ακόμα η μετάβαση από maven σε sbt https://github.com/USCDataScience/sparkler/issues/184. Στην συνέχεια, μιας και δεν εχει τελειώσει η μετάβαση από maven σε sbt, δοκιμάστηκε να γίνει δοκιμή με branches πριν ξεκινήσουν οι αλλαγές σε sbt, αλλά κατά το build τους δεν βρίσκει κάποια plugin από κάποια url και είναι προφανές μιας και έχουν περάσει τοσα χρόνια.

ΚΕΦΑΛΑΙΟ 5 - Crawler εργαλεία

5.1 Εισαγωγή

Στο ακόλουθο κεφάλαιο, γίνεται αναφορά σε διάφορα εργαλεία τα οποία κάνουν παρόμοια εργασία. Κάνουν ανίχνευση ιστού σε σελίδες και είναι ανοιχτού κώδικα. Επίσης αναφέρονται κάποια βασικά πλεονεκτήματα και μειονεκτήματα τους.

5.2 Heritrix

5.2.1. Περιγραφή

Το Heritrix είναι ένα πρόγραμμα ανίχνευσης ιστού (web crawler) ανοιχτού κώδικα και επεκτάσιμο. Έχει δημιουργήθει απο το Internet Archive, η οποία είναι μια μη κερδοσκοπική βιβλιοθήκη με εκατομμύρια δωρεάν βιβλία, ταινίες, λογισμικό, μουσική, ιστότοπους και πολλά άλλα. Είναι διαθέσιμο με άδεια ελεύθερου λογισμικού και γραμμένο σε Java. Το Heritrix εκτελείται σε κατανεμημένο περιβάλλον. Είναι επεκτάσιμο, αλλά όχι δυναμικά επεκτάσιμο. Αυτό σημαίνει ότι πρέπει να αποφασίσετε για τον αριθμό των μηχανημάτων προτού ξεκινήσετε την ανίχνευση ιστού.

Το Heritrix χρησιμοποιείται κυρίως σε Linux. Μπορεί να χρησιμοποιηθεί και σε άλλες πλατφόρμες αλλά δεν έχει δοκιμαστεί ούτε υπάρχει η κατάλληλη υποστήριξη σε αυτές. Το Heritrix απαιτεί Java 8 ή 11 - ιδανικά είναι τα πακέτα της OpenJDK 11. Το Heritrix εκτελείται σε ένα κατανεμημένο περιβάλλον κατακερματίζοντας τα URL τους σε κατάλληλα μηχανήματα. Η μορφή εξόδου του Heritrix είναι αρχεία WARC, μια αποτελεσματική μορφή εγγράφων για την εγγραφή πολλαπλών πόρων (όπως HTML) και των μεταδεδομένων τους σε ένα αρχείο αρχειοθέτησης. Το Heritrix έχει συντηρηθεί καλά από την έκδοση που κυκλοφόρησε το 2004 και χρησιμοποιείται στην παραγωγή από διάφορους ιστότοπους.

5.2.2. Πλεονεκτήματα

- Πολύ καλή τεκμηρίωση εγγράφων (documentation) και εύκολη εγκατάσταση
- Επεκτάσιμη, καλή απόδοση και πολύ καλή υποστήριξη για κατανεμημένες ανιχνεύσεις (clawls)
- Ώριμη πλατφόρμα. Βρίσκεται σε παραγωγική χρήση για πάνω από μια δεκαετία
- Σέβεται το robot.txt

5.2.3. Μειονεκτήματα

- Δεν υποστηρίζει η συνεχή ανίχνευση (clawling)
- Δεν είναι δυναμικά επεκτάσιμο
- Εξάγει αρχεία ARC/WARC. Η προσθήκη υποστήριξης άλλων εξόδων θα απαιτούσε μεγάλη αλλαγή του κώδικα.

5.3 StormCrawler

5.3.1. Περιγραφή

Η StormCrawler είναι μια βιβλιοθήκη ανοιχτού κώδικα για τη δημιουργία κατανεμημένων ανιχνευτών ιστού που βασίζονται στο Apache Storm. Το έργο είναι υπό την άδεια της Apache και αποτελείται από μια συλλογή επαναχρησιμοποιήσιμων πόρων και στοιχείων και γραμμένο κυρίως σε Java. Το πλαίσιο (framework) αυτό βασίζεται στο πλαίσιο επεξεργασίας ροής της Apache Storm και όλες οι λειτουργίες πραγματοποιούνται ταυτόχρονα, όπως – ανάκτηση, ανάλυση και αναζήτηση(indexing) των διευθύνσεων URL συνεχώς – γεγονός που καθιστά ολόκληρη τη διαδικασία ανίχνευσης δεδομένων πιο αποτελεσματική.

Οι προγραμματιστές μπορούν να την αξιοποιήσουν για να δημιουργήσουν τα δικά τους προγράμματα ανίχνευσης. Ο στόχος του StormCrawler είναι να βοηθήσει στη δημιουργία ανιχνευτών ιστού που είναι επεκτάστιμο, με μικρή καθυστέρηση και ανεκτικό σε αλλαγές. Χρησιμοποιείται στην παραγωγή από πολλές εταιρείες και η συντήρηση του είναι ενεργή.

5.3.2. Πλεονεκτήματα

- Κατάλληλο για μεγάλης κλίμακας αναδρομικές ανιχνεύσεις
- Κατάλληλο για ανίχνευση ιστού χαμηλής καθυστέρησης

5.3.3. Μειονεκτήματα

- Δεν υποστηρίζει τη διαγραφή εγγράφων

5.4 Scrapy

5.4.1. Περιγραφή

Το Scrapy είναι μία υψηλού επιπέδου βιβλιοθήκη ανοιχτού κώδικα, που χρησιμοποιείται για την ανίχνευση ιστοτόπων και την εξαγωγή δομημένων δεδομένων από τις σελίδες τους. Είναι βιβλιοθήκη για την Python που χρησιμοποιείται για τη δημιουργία web scrapers. Η έκδοση της Python πρέπει να είναι μεγαύτερη απο την 3.6. Παρέχει όλα τα εργαλεία που χρειάζονται για να γίνει η εξαγωγή δεδομένων από ιστότοπους,η επεξεργασία και η αποθήκευση σε ποικίλες μορφές και δομές. Το Scrapy έχει μερικές ενσωματωμένες κάποιες εύχρηστες μορφές εξαγωγής όπως JSON, XML και CSV. Λειτουργεί σε συστήματα Linux, Mac OS και Windows.

Αυτό που ξεχωρίζει στο Scrapy είναι η ευκολία χρήσης και η καλή τεκμηρίωση που παρέχει. Το Scrapy κατασκευάστηκε για την εξαγωγή συγκεκριμένων πληροφοριών από ιστότοπους, για αυτόν τον λογο δεν έχει ενσωματωμένη λειτουργικότητα για εκτέλεση σε κατανεμημένο περιβάλλον. Αυτό δεν σημαίνει ότι το Scrapy δεν μπορεί να χρησιμοποιηθεί για ευρεία ανίχνευση, όμως κάποια άλλα εργαλεία μπορεί να είναι καλύτερα κατάλληλα για αυτόν τον σκοπό. Συντηρείται κυρίως απο την Zyte (πρώην Scrapinghub) και πολλούς άλλους συνεργάτες.

5.4.2. Πλεονεκτήματα

- Εύκολο στη ρύθμιση και χρήση (προυπόθεση γνώση γλώσσας Python)
- Λεπτομερής τεκμηρίωση
- Ενεργή Κοινότητα - μεγάλη κοινότητα προγραμματιστών
- Ενσωματωμένες μορφές εξαγωγής JSON, JSON lines, XML και CSV

5.4.3. Μειονεκτήματα

- Δεν υπάρχει υποστήριξη για λειτουργία σε κατανεμημένο περιβάλλον
- Δεν υπάρχει υποστήριξη για συνεχείς ανιχνεύσεις (crawls)
- Η εξαγωγή μεγάλων ποσοτήτων δεδομένων είναι δύσκολη
- Δεν χειρίζεται JavaScript

5.5 Apify SDK

5.5.1. Περιγραφή

Το Apify SDK είναι μια βιβλιοθήκη Node.js που μοιάζει πολύ με το Scrapy το οποίο είναι μια βιβλιοθήκη σε Javascript. Υπάρχει υποστήριξη σε Puppeteer, σε Cheerio και άλλα. Υποστηρίζει ένα απλό πακέτο για παράλληλη ανίχνευση (crawling). Είναι διαθέσιμο ως το npm πακέτο apify.

Το Apify SDK απλοποιεί την ανάπτυξη προγραμμάτων ανίχνευσης ιστού (crawlers), scrapers, εξαγωγέων δεδομένων και αυτόματα jobs. Το Apify SDK καλύπτει την διαδικασία απο την ανίχνευση του ιστότοπου (crawling) μέχρι και την αποθήκευση των δεδομένων, τα οποία είναι εύκολα επεξεργάσιμα. Μπορεί να χρησιμοποιηθεί τόσο τοπικά σε οποιονδήποτε υπολογιστή όσο και στην πλατφόρμα του Apify. Διαθέτει ένα εργαλείο,το Basic Crawler, που απαιτεί από τον χρήστη να υλοποιήσει τη λήψη της σελίδας και την εξαγωγή δεδομένων.

5.5.2. Πλεονεκτήματα

- Υποστηρίζει κάθε τύπο ιστότοπου
- Η καλύτερη βιβλιοθήκη για ανίχνευση ιστού σε Javascript που έχει δοκιμάσει μέχρι στιγμής.
- Ενσωματωμένη υποστήριξη του Puppeteer

5.6 NodeCrawler

5.6.1. Περιγραφή

Το Nodecrawler είναι το πιο ισχυρό, δημοφιλές και παραγωγικό πακέτο ανίχνευσης για Node. Είναι ένα ελαφρύ εργαλείο ανίχνευσης node.js που λαμβάνει υπόψη την αποτελεσματικότητα και την ευκολία, υποστηρίζει κατανεμημένα συστήματα παρακολούθησης, υποστηρίζει hard coding και επίσης υποστηρίζει πράκτορες front-level HTTP. Υποστηρίζει Node.js 4 και άνω. Εάν ο χρήστης προτιμάει την κωδικοποίηση JavaScript ή το έργο που θέλει να το χρησιμοποιήσει είναι ένα έργο Javascript, το Nodecrawler θα είναι το πιο κατάλληλο

πρόγραμμα ανίχνευσης ιστού για χρήση. Η εγκατάσταση του είναι επίσης αρκετά απλή. Υλοποιεί γρήγορη ανάλυση δεδομένων DOM και selector λειτουργία που τηρεί την σύνταξη του jQuery, το Cheerio χρησιμοποιείται από προεπιλογή και μπορεί να αντικαταστήσει το JSDOM με άλλους αναλυτές DOM.

5.6.2. Πλεονεκτήματα

- Εύκολη εγκατάσταση
- Διαθέτει DOM στην πλευρά του διακομιστή και αυτόματη εισαγωγή jQuery με Cheerio (προεπιλογή) ή JSDOM.
- Παρέχει προτεραιότητα ουράς αιτημάτων υποστήριξης (δηλαδή τα αιτήματα για διαφορετικές διευθύνσεις URL μπορεί να έχουν διαφορετικές προτεραιότητες)
- Λειτουργία καθυστέρησης υποστήριξης (delay function), ορισμένοι διακομιστές έχουν όριο στον αριθμό των συνδέσεων ανά λεπτό.

5.6.3. Μειονεκτήματα

- Δεν έχει υποστήριξη Promise

5.7 MechanicalSoup

5.7.1. Περιγραφή

Το MechanicalSoup είναι μια βιβλιοθήκη Python για την αυτοματοποίηση της αλληλεπίδρασης με ιστότοπους. Το MechanicalSoup αποθηκεύει και στέλνει αυτόματα cookies, ακολουθεί ανακατευθύνσεις και μπορεί να ακολουθεί συνδέσμους και να υποβάλλει φόρμες. Το MechanicalSoup έχει σχεδιαστεί για να μιμείται τη συμπεριφορά του τρόπου με τον οποίο οι άνθρωποι αλληλεπιδρούν με προγράμματα περιήγησης ιστού.

Το MechanicalSoup δημιουργήθηκε από τον Hickford, ο οποίος ήταν πάντα ενθουσιασμένος με την βιβλιοθήκη Mechanize. Όμως το Mechanize δεν είναι συμβατό με την Python 3 και η ανάπτυξή του έχει σταματήσει για αρκετά χρόνια. Το MechanicalSoup παρέχει ένα παρόμοιο API, βασισμένο σε αιτήματα της Python (για συνεδρίες http) και BeautifulSoup (για πλοήγηση σε έγγραφα). Από το 2017 είναι ένα έργο που συντηρείται ενεργά από μια μικρή ομάδα προγραμματιστών.

5.7.2. Πλεονεκτήματα

- Αυτοματοποίηση της αλληλεπίδρασης με ιστότοπους

5.7.3. Μειονεκτήματα

- Ο ιστότοπος πρέπει να μην βασίζεται σε JavaScript, γιατί τότε χρειάζεστε ένα πλήρες πρόγραμμα περιήγησης και οχι MechanicalSoup.
- Ο ιστότοπος πρέπει να μην περιέχει σελίδες HTML, γιατί το MechanicalSoup δεν θα φέρει τίποτα σε σύγκριση με τα αιτήματα(requests)
- Ο ιστότοπος πρέπει να μην παρέχει υπηρεσία (webservice) API, όπως είναι το REST

ΚΕΦΑΛΑΙΟ 6 - Εφαρμογή

6.1 Εισαγωγή

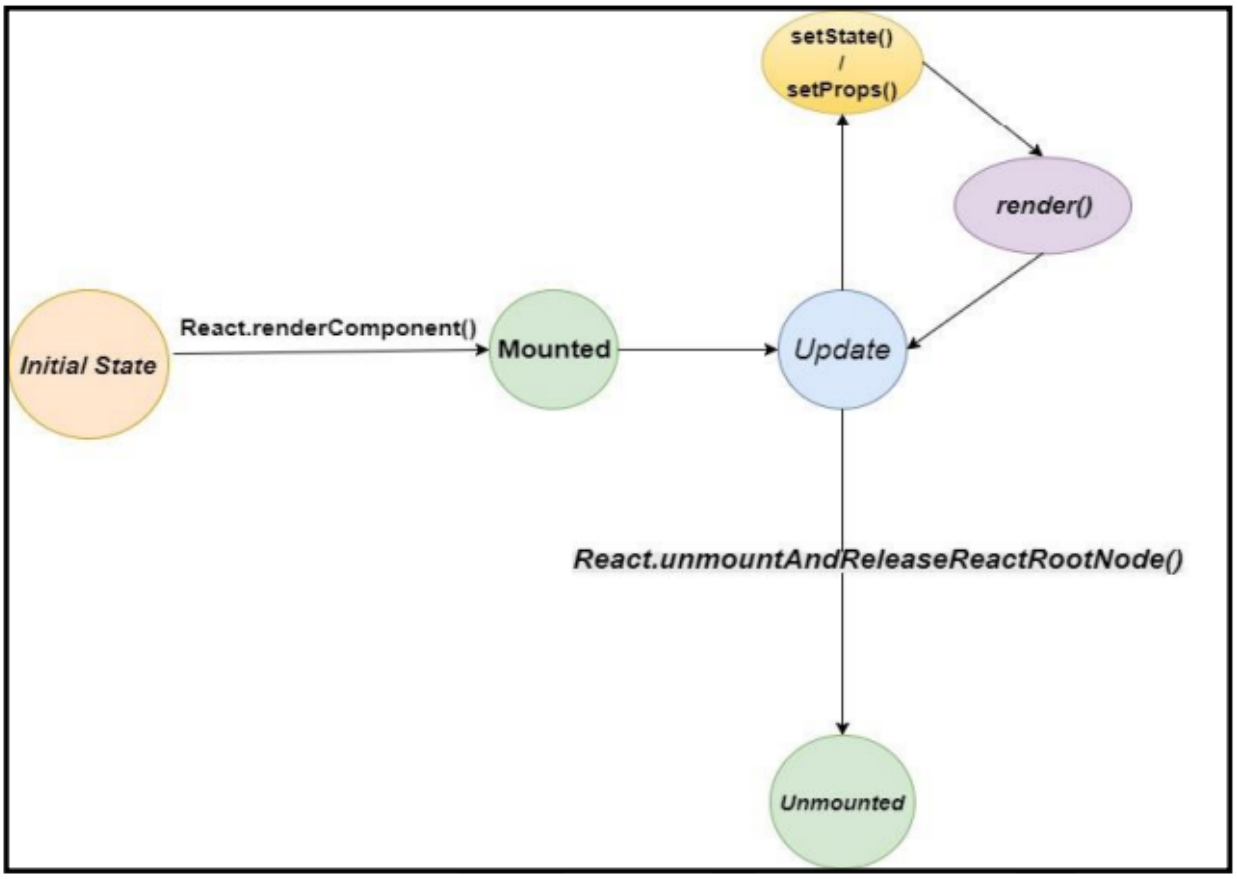
Στο παρακάτω κεφάλαιο, αναλύονται οι τεχνολογίες που χρησιμοποιήθηκαν για την τελική εφαρμογή. Οι παρακάτω τεχνολογίες χρησιμοποιήθηκαν στον scraper που δημιουργήθηκε, στο back-end με την παρουσίαση ενός swagger, και την front-end εφαρμογή του χρήστη. Στην συνέχεια, γίνεται η ανάλυση της εφαρμογής. Αναλύεται ο τρόπος που έγινε η συλλογή των δεδομένων, τι περιλαμβάνει το API / back-end, ποια είναι η αρχιτεκτονική της εφαρμογής και τέλος παρουσιάζεται το interface της εφαρμογης. Τέλος στο παράρτημα 3 παρουσιάζεται ο κώδικας για το κάθε κομμάτι που αναφέρθηκε προηγουμένως.

6.2 Τεχνολογίες

6.2.1 React

Το ReactJS είναι βιβλιοθήκη JavaScript που αναπτύσσεται για την ανάπτυξη στοιχείων διεπαφής χρήστη (UI) επαναχρησιμοποιήσιμων. Το ReactJS είναι μια βιβλιοθήκη βασισμένη σε στοιχεία που αναπτύσσεται για την ανάπτυξη διαδραστικών διεπαφών χρήστη. Αυτήν τη στιγμή είναι η πιο δημοφιλής βιβλιοθήκη JS front-end. Υποστηρίζεται από το Facebook, το Instagram και μια κοινότητα μεμονωμένων προγραμματιστών και οργανισμών. Το React ουσιαστικά επιτρέπει την ανάπτυξη μεγάλων και πολύπλοκων εφαρμογών που βασίζονται στον ιστό που μπορούν να αλλάξουν τα δεδομένα του χωρίς συνεχόμενες ανανεώσεις σελίδας. Στοχεύει στην παροχή καλύτερες εμπειρίες στους χρήστες και με εκπληκτική γρήγορη και ισχυρή ανάπτυξη εφαρμογών ιστού. [13]

Το React ουσιαστικά επιτρέπει την ανάπτυξη μεγάλων και πολύπλοκων εφαρμογών που βασίζονται στον ιστό που μπορούν να αλλάξουν τα δεδομένα του χωρίς επακόλουθες ανανεώσεις σελίδας. Χρησιμοποιείται ως προβολή (V) στο Model-View-Controller (MVC). Το React αποδίδει κυρίως στην πλευρά του διακομιστή χρησιμοποιώντας το NodeJS και η υποστήριξη για εγγενείς εφαρμογές για κινητά προσφέρεται χρησιμοποιώντας το React Native.[13]



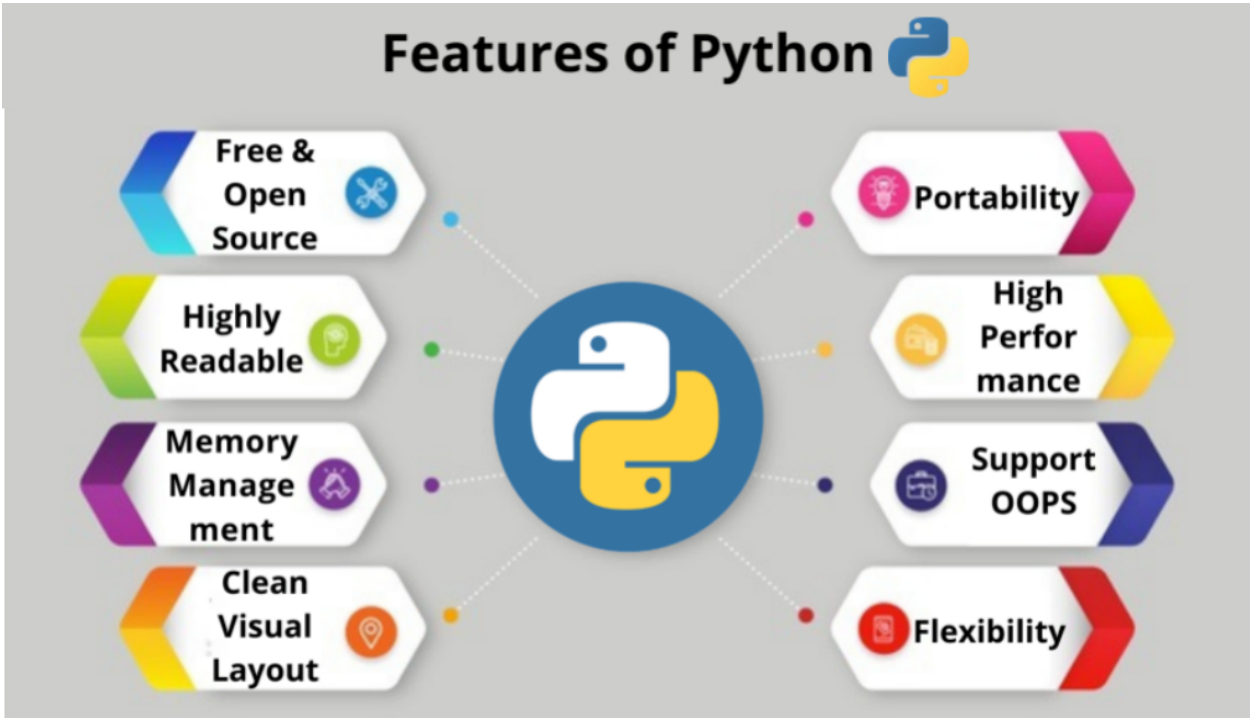
Εικόνα 17: Ο κύκλος ζωής του React Component

6.2.2 Python

Η Python είναι μια ερμηνευμένη, διαδραστική, αντικειμενοστραφής γλώσσα προγραμματισμού. Έχει μια εξαιρετικά απλή και κομψή σύνταξη και ωστόσο είναι μια ισχυρή και γλώσσα προγραμματισμού γενικού σκοπού. Είναι δωρεάν, ακόμη και για εμπορικούς σκοπούς, και μπορεί να εκτελεστεί σχεδόν σε οποιονδήποτε σύγχρονο υπολογιστή. Ένα πρόγραμμα python μεταγλωττίζεται αυτόματα από τον διερμηνέα σε κώδικα byte ανεξάρτητου πλατφόρμας, ο οποίος στη συνέχεια ερμηνεύεται. Εκτελούμε μη τροποποιημένα στοιχεία γραμμένα σε Python σε Linux, Windows NT, 98, 95, IRIX, SunOS, OSF. Η Python είναι αρθρωτή από τη φύση της.[14]

Παρέχει δομές δεδομένων υψηλού επιπέδου, όπως λίστα και συσχετιστικούς πίνακες (που ονομάζονται λεξικά), δυναμική πληκτρολόγηση και δυναμική σύνδεση, ενότητες, κλάσεις, εξαιρέσεις, αυτόματη διαχείριση μνήμης, κ.λπ. Η διανομή Python περιλαμβάνει μια ποικιλόμορφη βιβλιοθήκη τυπικών επεκτάσεων (μερικές γραμμένες σε Python, άλλες σε C ή C++) για λειτουργίες που κυμαίνονται από χειρισμούς συμβολοσειρών και κανονικές εκφράσεις τύπου Perl, έως γεννήτριες γραφικών διεπαφής χρήστη (GUI) και συμπεριλαμβανομένων βοηθητικών προγραμμάτων που σχετίζονται με υπηρεσίες λειτουργικού συστήματος του ιστού, εργαλεία εντοπισμού σφαλμάτων και δημιουργίας προφίλ, κ.λπ. Υπάρχει ένας σημαντικός

αριθμός μονάδων επέκτασης που έχουν αναπτυχθεί και διανέμονται από μέλη της κοινότητας χρηστών Python. Αυτές οι μονάδες επέκτασης, που μερικές φορές αναφέρονται ως "πακέτα" ή συστατικά περιλαμβάνουν ως GADFLY, έναν διαχειριστή βάσης δεδομένων SQL γραμμένο σε Python, η βιβλιοθήκη απεικόνισης Python(PIL-Python imaging library).[14]



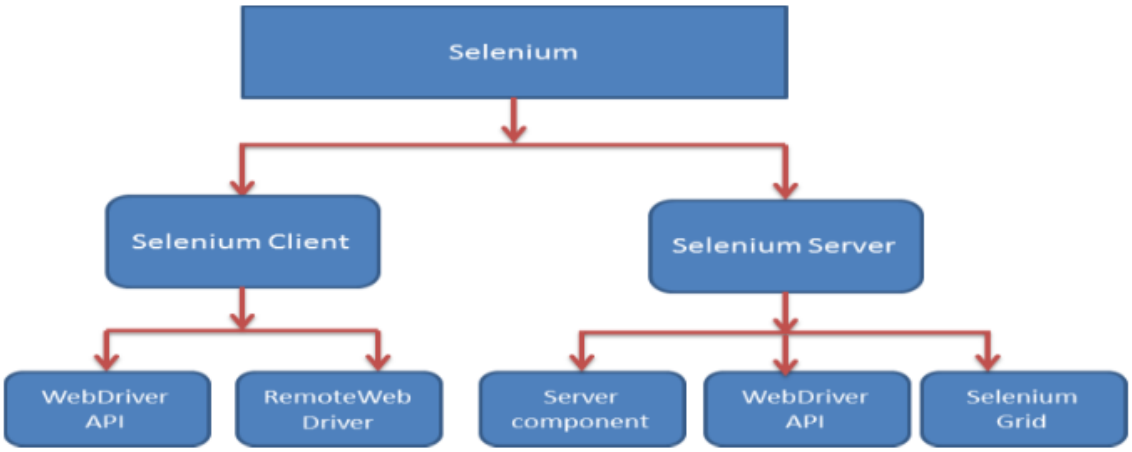
Εικόνα 18: Χαρακτηριστικά της Python

6.2.3 Selenium

Το Selenium είναι μια δημοφιλής βιβλιοθήκη ανοιχτού κώδικα που χρησιμοποιείται για τη δοκιμή εφαρμογών ιστού λειτουργικών στοιχείων. Το Selenium είναι μια αυτοματοποιημένη δοκιμαστική ακολουθία ανοιχτού κώδικα που βοηθά στην αυτοματοποιημένη δοκιμή εφαρμογών Ιστού. Το Selenium είναι μια δωρεάν εφαρμογή που υποστηρίζει διαφορετικές πλατφόρμες, όπως Linux, Windows, Mac κ.λπ. Το Selenium μπορεί να χρησιμοποιηθεί για προγράμματα περιήγησης όπως το Google-Chrome, το Firefox και το Internet-Explorer. Η Java, C#, Python, Ruby μπορούν να χρησιμοποιηθούν για τη σύνταξη δοκιμαστικών περιπτώσεων. Το Selenium είναι ένα εργαλείο ανοιχτού κώδικα που εστιάζει στη συμβατότητα που υποστηρίζει επίσης διαφορετικά προγράμματα περιήγησης, μορφές, γλώσσες και πλαίσια. Όλες οι δημοφιλείς πλατφόρμες και γλώσσες προγραμματισμού υποστηρίζονται από το Selenium. Η Java ήταν η πρώτη επιλογή των δοκιμαστών για χρήση με το Selenium πριν από την Python. Λόγω των πολλών πλεονεκτημάτων της Python, το Selenium έχει γίνει επίσης μια πολύ δημοφιλής γλώσσα.[11]

Η αρχιτεκτονική του Selenium περιλαμβάνει δύο θεμελιώδη στοιχεία: Client και Selenium Server. Στο μέρος του πελάτη περιλαμβάνεται ένα WebDriver API για αλληλεπιδράσεις ιστοσελίδων και άλλες δυνατότητες εφαρμογής. Παρέχει επίσης το πρόγραμμα Remote Web

Driver που επικοινωνεί με τον απομακρυσμένο διακομιστή Selenium. Ο server Selenium αποτελείται από ένα τμήμα διακομιστή που χρησιμοποιείται για τη λήψη αιτημάτων από την κλάση Remote Web Driver του client Selenium. Περιλαμβάνει επίσης το API του προγράμματος, το οποίο μπορεί να χρησιμοποιηθεί για τη δοκιμή του προγράμματος περιήγησης ιστού σε μια μηχανή διακομιστή. Το τέταρτο μέρος είναι το Selenium Grid, το οποίο χρησιμοποιεί ο Selenium Server μέσω παραμέτρων από την γραμμή εντολών για χαρακτηριστικά πλέγματος, έχοντας ένα κεντρικό hub και διάφορους κόμβους και αγαπημένες ικανότητες προγράμματος περιήγησης. Το Grid είναι μια μέθοδος που επιτρέπει τη διεξαγωγή παράλληλων πειραμάτων σε πολλά μηχανήματα και διάφορα προγράμματα περιήγησης, γεγονός που επηρεάζει τον μειωμένο χρόνο εκτέλεσης. [11]



Εικόνα 19: Αρχιτεκτονική Selenium

6.3 Ανάλυση εφαρμογής

6.3.1 Συλλογή δεδομένων - scraper

Αρχικά γίνεται η λήψη των url/site που πρέπει να ελεγχθούν και να αποθηκευτεί η πληροφορία τους. Τα url/site που χρησιμοποιούνται στην συγκεκριμένη εφαρμογή περιέχουν πολλά άρθρα για το θέλα που έχουμε επιλέξει, κάπως σαν κατηγορία δηλαδή. Στην συνέχεια, υπάρχει ένα json object το οποίο περιέχει για κάθε site τα tag και τους selectors ανάλογα το html code της σελίδας και ανάλογα τι πληροφορία χρειάζεται να αντληθεί απο την σελίδα. Για κάθε url/site που υπάρχει στο ElasticSearch μαζεύονται όλα τα links/άρθρα σε έναν πίνακα που περιέχονται στην κάθε σελίδα για να γίνει η συλλογή των πληροφοριών. Μετα την συλλογή όλων των links/άρθρων που πρέπει να γίνει άντληση του περιεχομένου τους, για κάθε σελίδα γίνεται η εξής διαδικασία. Οπότε για κάθε link/άρθρο γίνεται ο διαχωρισμός των πληροφοριών που χρειάζονται για την εφαρμογή. Οι πληροφορίες που έχουν συλλεχθεί είναι οι εξής:

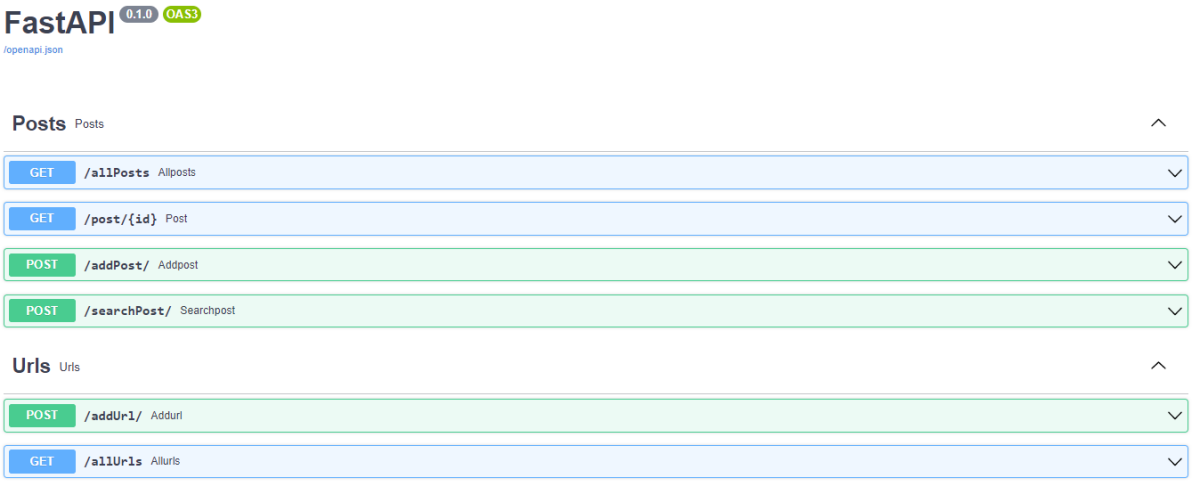
- το src της κεντρικής εικόνας του άρθρου
- τον τίτλο του άρθρου
- την ημερομηνία δημοσίευσης του άρθρου - για να γίνει η σωστή αποθήκευση της ημερομηνίας των άρθων στην ίδια μορφή έγινε η χρήση της datetime βιβλιοθήκης
- την περιγραφή του άρθρου

- το περιεχόμενο του άρθου
- το id του άρθου που έχει το ίδιο το site

Τέλος, γίνεται η ένωση των παραπάνω πληροφοριών σε json object και μέσω μιας κλήσης στο API γίνεται η αποθήκευση των παραπάνω πληροφοριών.

6.3.2 API

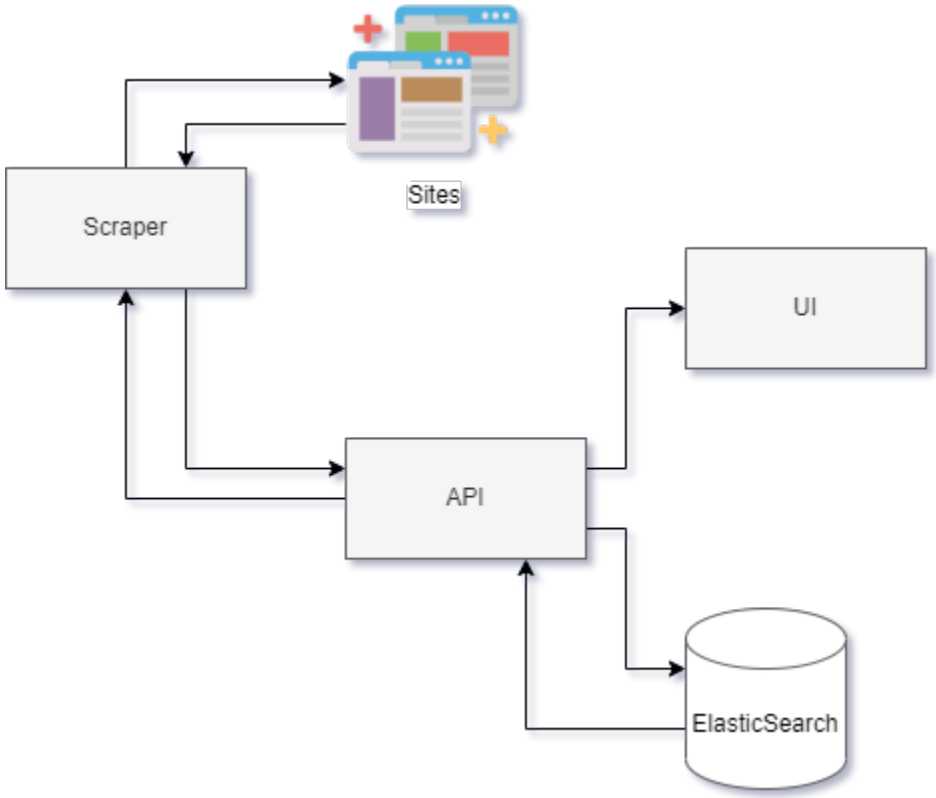
Αρχικά, το API περιέχει δυο κατηγορίες πληροφοριών, για τα urls και τα άρθρα.
Στα urls, υπάρχει ένα request για να επιστρέφει όλα τα url που πρέπει να “χτυπήσει” το προγραμμα του scraper για να αντλήσει την πληροφορία που θέλουμε. Το άλλο request είναι για να προστίθονται και άλλα url στην λίστα.
Στα άρθρα, υπάρχει ένα request για να επιστρέφει όλα τα άρθρα που έχουν μαζευτεί απο τα url που έχουν scraping. Το άλλο request είναι για να προστίθονται και άλλα άρθρα απο το scraping στην λίστα. Ακόμη άλλο ένα request είναι για να μπορεί να γίνει αναζήτηση στα άρθρα με κάποια λέξη κλειδί. Και το τελευταίο request επιστρέφει τα στοιχεία συγκεκριμένου άρθρου, ανάλογα το id του.



Εικόνα 20: Μορφή API

6.3.3 Αρχιτεκτονική εφαρμογής

Το Elasticsearch στέλνει στο API τα sites/urls που πρέπει να στείλει στον scraper. Αυτά τα sites/urls τα παίρνει σαν πληροφορία ο scraper και ψαχνει μέσα στο περιεχόμενο τους την πληροφορία που χρειάζεται. Στην συνέχεια ο scraper μαζεύει τα δεδομένα που χρειάζονται απο την κάθε σελίδα και μέσω μιας κλήσης του API τα αποθηκεύει στο ElasticSearch. Αφού έχουν μαζευτεί όλα τα δεδομένα, μια εφαρμογή κάνοντας κάποια requests στο API επιστρέφονται τα δεδομένα και γίνεται η οπτικοποίηση τους.



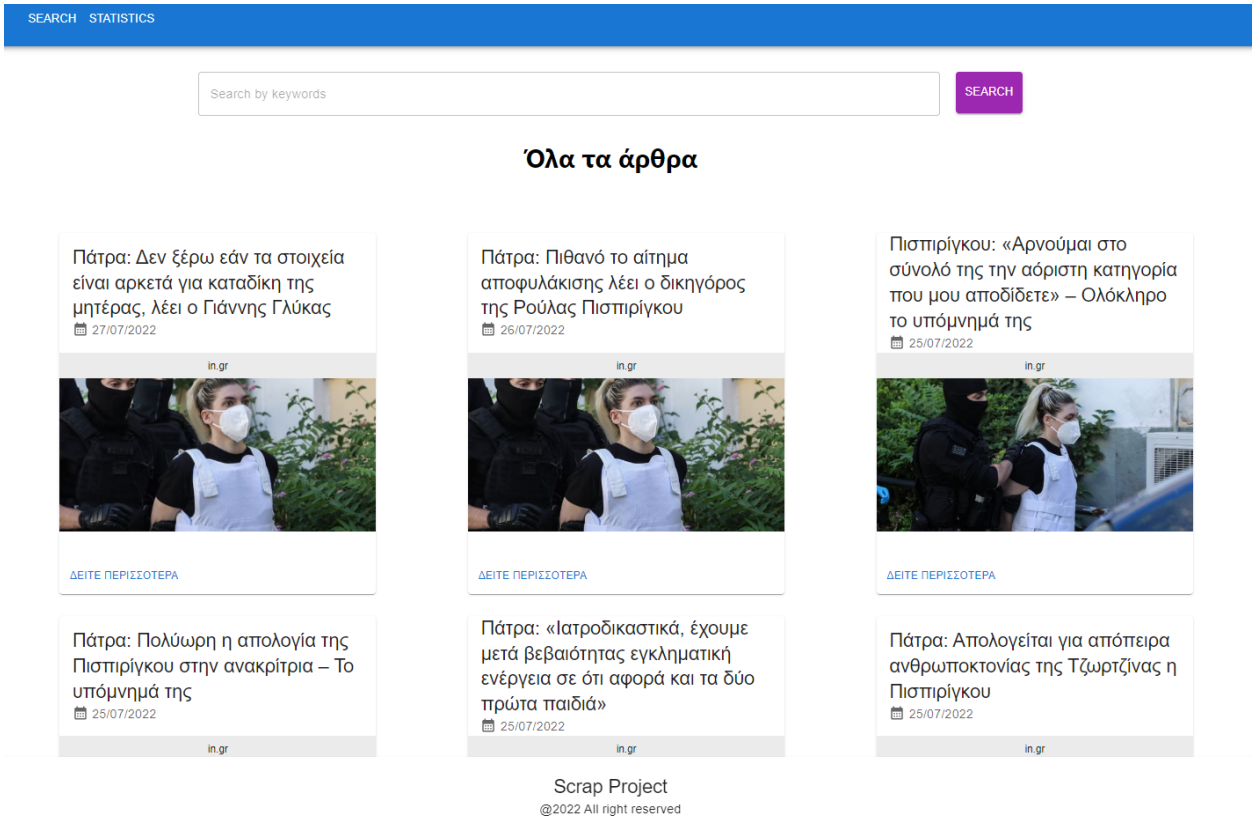
Εικόνα 21: Αρχιτεκτονική της εφαρμογής

6.3.4 Παρουσίαση εφαρμογής

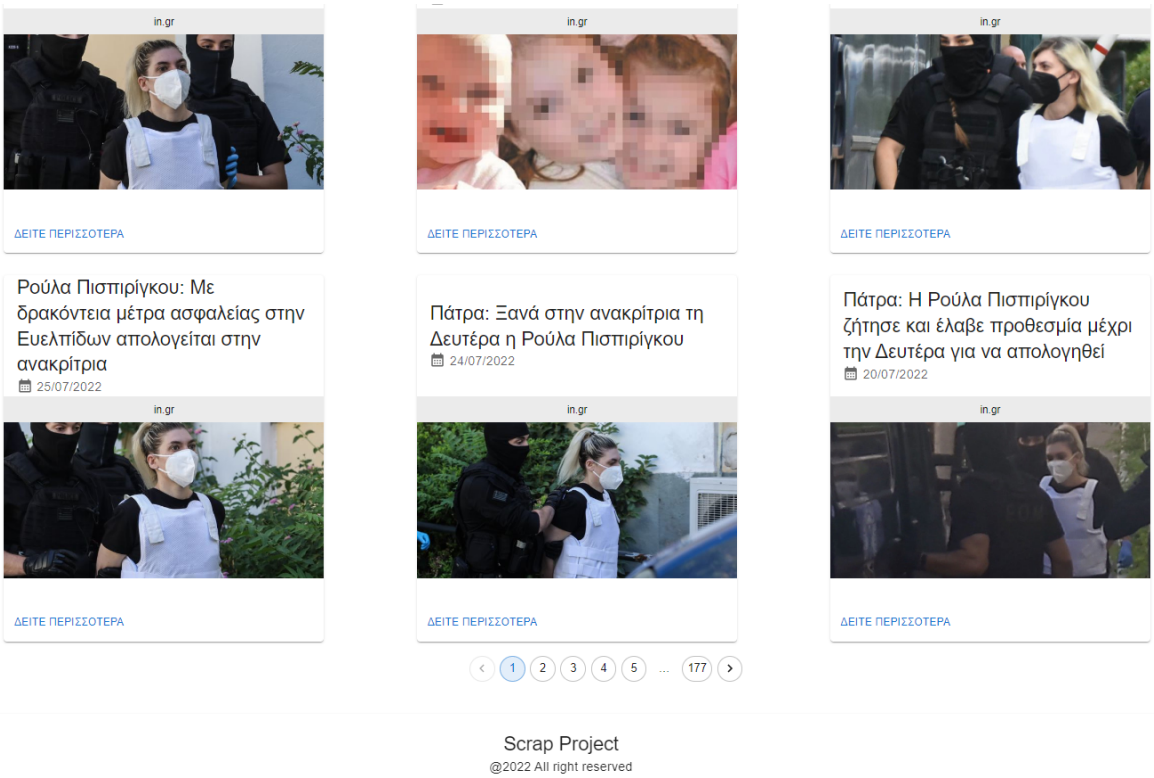
Αρχική οθόνη

Η αρχική οθόνη περιλαμβάνει τα άρθρα που έχουν συλλεχθεί, αυτή η οθόνη περιέχει και σελιδοποίηση. Περιέχει μια μπάρα αναζήτησης, στην οποία εισάγοντας ένα κλειδί μπορεί και επιστρέφει όλα τα σχετικά αποτελέσματα άρθρων μαζί με ένα διαγραμμα για να συγκρίνεται το

πλήθος των αποτελεσμάτων με το συνολικό πλήθος των άρθρων που έχουν συλλεχθεί.

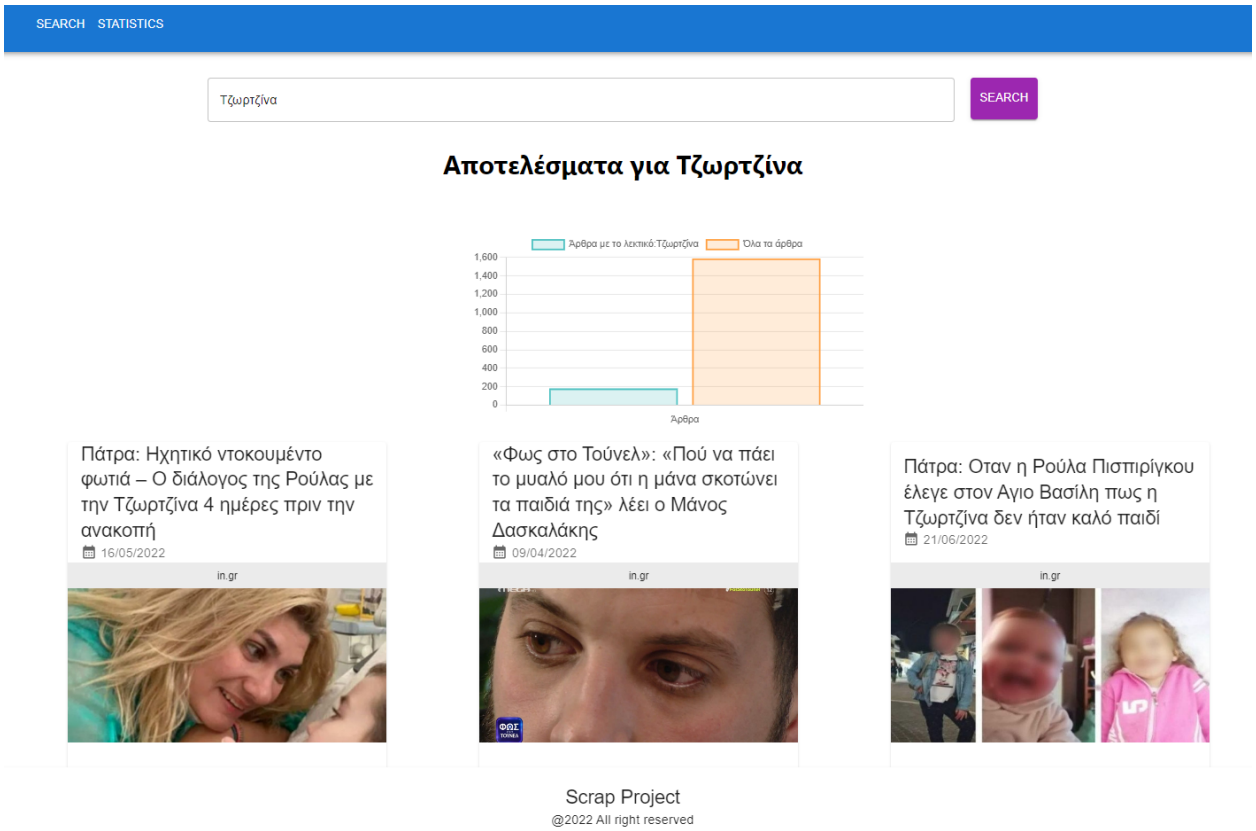


Εικόνα 22: Τελική εφαρμογή - Άρθρα



Εικόνα 23: Τελική εφαρμογή - Άρθρα με σελιδοποίηση

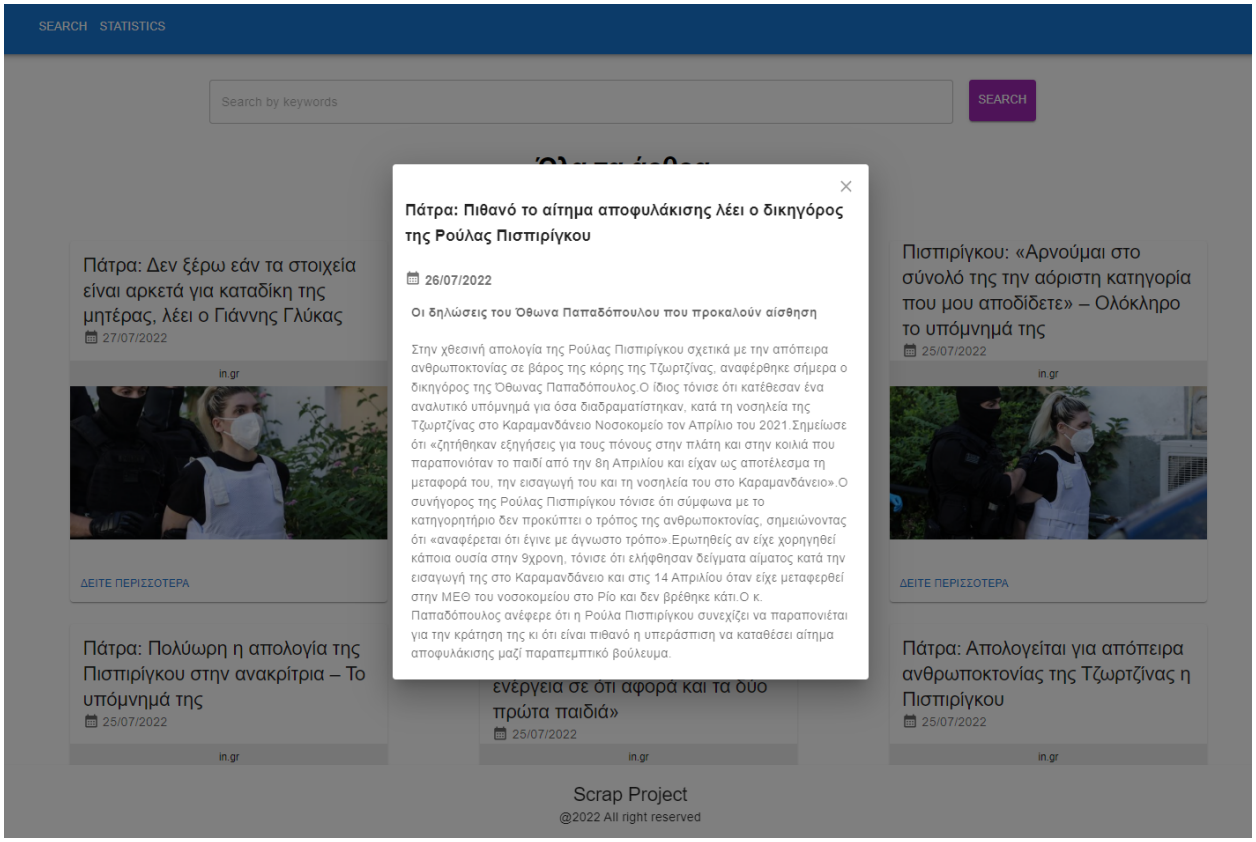
Εισάγοντας ένα κλειδί μπορεί και επιστρέφει όλα τα σχετικά αποτελέσματα άρθρων μαζί με ένα διαγραμμα για να συγκρίνεται το πλήθος των αποτελεσμάτων με το συνολικό πλήθος των άρθρων που έχουν συλλεχθεί.



Εικόνα 24: Τελική εφαρμογή - Στατιστικά για συγκεκριμένο όρο

Εικόνα 25: Τελική εφαρμογή - Στατιστικά για συγκεκριμένο όρο

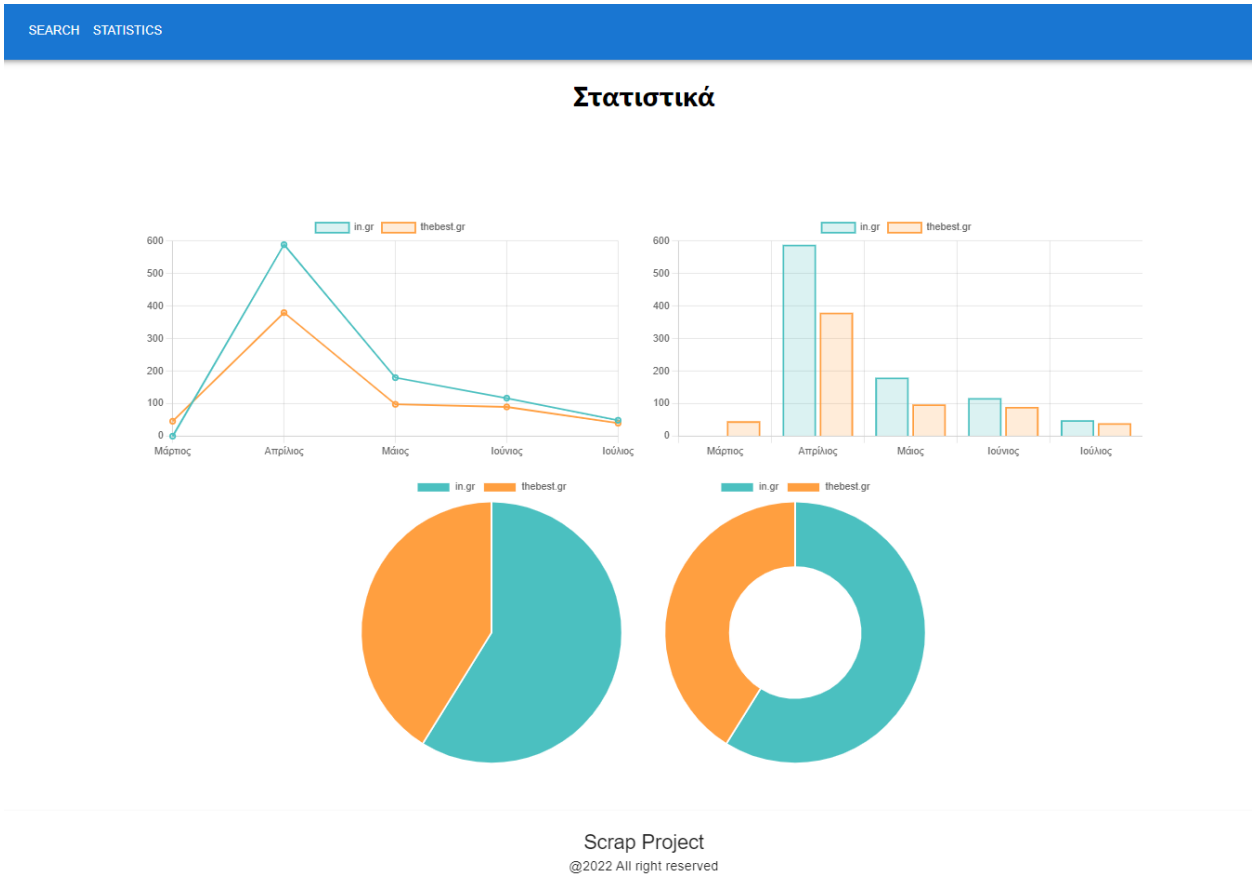
Πατώντας το "Δείτε Περισσότερα", απο κάποιο άρθρο, εμφανίζει όλες τις πληροφορίες του κάθε άρθρου.



Εικόνα 26: Τελική εφαρμογή - Λεπτομέρειες άρθρου

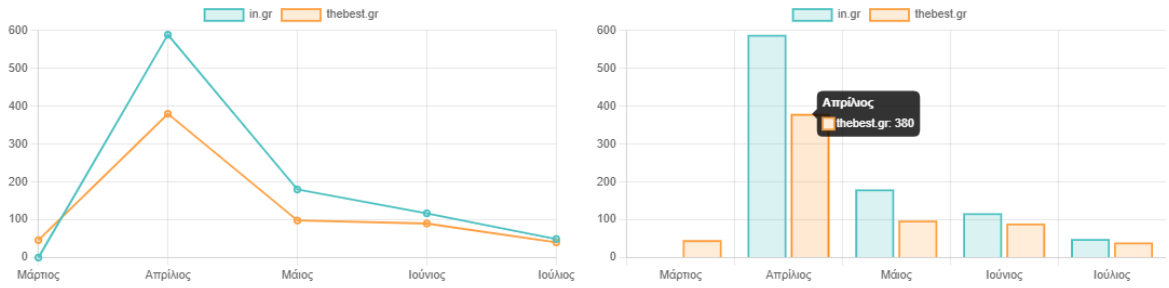
Στατιστικά

Η δεύτερη σελίδα περιέχει τα στατιστικά των άρθρων.



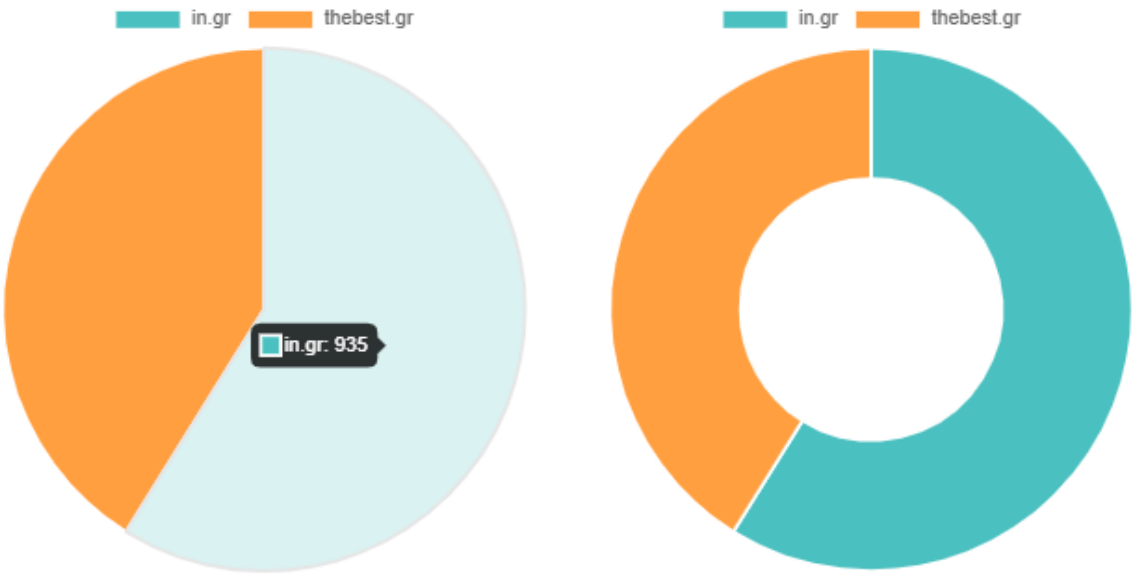
Εικόνα 27: Τελική εφαρμογή - Στατιστικά για συνολικά δεδομένα

Στα εξής διαγράμματα παρουσιάζεται ποσα αρθρα ανα site υπάρχουν ανα μηνα.



Εικόνα 28: Τελική εφαρμογή - Στατιστικά για συνολικά δεδομένα σε μορφή ραβδογράμμα και ιστόγραμμα

Και τέλος στις πίτες φαίνονται ποσα αρθρα συνολικά ανέβασε το καθε site για το συγκεκριμένο θέμα.



Εικόνα 29: Τελική εφαρμογή - Στατιστικά για συνολικά δεδομένα σε μορφή πίτας

ΚΕΦΑΛΑΙΟ 7 - ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ ΑΝΑΠΤΥΞΗΣ

7.1 Εισαγωγή

Στο κεφάλαιο αυτό, αναλύονται τα συμπεράσματα που προέκυψαν απο την έρευνα που έγινε σε όλα τα παραπάνω κεφάλαια. Επίσης παρουσιάζονται κάποιες προτάσεις ανάπτυξης που θα μπορούσαν να υλοποιηθούν βασιζόμενες στην προεργασία που έχει ήδη υλοποιηθεί στην παρούσα διπλωματική.

7.2 Συμπέρασματα

Σε αυτή την διπλωματική, ερευνήθηκαν σε θεωρητικό κομμάτι αρκετά περιβάλλοντα που μπορεί να πραγματοποιήθει crawler σε σελίδες, βέβαια όλα είχαν και αρνητικά και θετικά, τα οποία με εμπόδισαν στον να τα χρησιμοποιήσω στην τελική μου εφαρμογή. Επίσης έγινε επικέντρωση σε δύο περιβάλλοντα που είναι στο επίκεντρο για το crawling.

Το ένα περιβάλλον ήταν το sparkler, στο οποίο έγινε μια μεγάλη έρευνα στο διαδίκτυο και πολλές δοκιμές μέχρι να γίνει απλά η εγκατάσταση του. Δυστυχώς, το συγκεκριμένο περιβάλλον δεν είναι έτοιμο για χρήση μιας και όπως ηταν γνωστό από την αρχή, δεν είναι μέρος του Apache. Άλλο ένα εμφανές πρόβλημα είναι ότι είχε πολλά ανοιχτά issues στο git, το οποίο σημαίνει ότι προσπαθούν ακόμα να λυθούν. Η μετατροπή του sparkler από Apache Maven σε Apache Ant έχει παίξει πολύ σημαντικό ρόλο στον μεγάλο αριθμό ανοιχτών issue μιας και ακόμα δεν μπορεί να γίνει build 100% επιτυχώς. Αγνοώντας τα παραπάνω, έγιναν αρκετές προσπάθειες να γίνει κατανόηση των προβλημάτων λυνοντας τα και να πραγματοποιηθεί η εγκατάσταση του, όμως χωρίς κάποιο ικανοποιητικό αποτέλεσμα. Μιας και ούτε ολοκληρωμένες οδηγίες εγκατάστασης υπήρχαν στο διαδίκτυο έγινε ακόμα πιο δυσκολη η έρευνα του Sparkler. Πέρα από την εγκατάσταση, άλλο ένα πρόβλημα ήταν η μορφή της πληροφορίας που θα αποθήκευε. Για την συγκεκριμένη εφαρμογή χρειαζοταν να λαμβάνεται ο html κώδικας της κάθε σελίδας ώστε να μπορεί να γίνει η αναζήτηση του κατάλληλου selector για την κάθε πληροφορία, και αυτό δεν ήταν εφικτό χωρίς να προστεθεί άλλος κώδικας. Οπότε για να μπορέσει να υπάρχει η σωστή μορφή της πληροφορίας που χρειαζόταν θα έπρεπε να προστεθούν κάποια εργαλεία όπως είναι το beautiful soup.

Το επόμενο περιβάλλον που εξετάστηκε ήταν το Apache Nutch. Σε αυτό το περιβάλλον ήταν πιο εύκολη η εγκατάσταση του, όμως υπηρχαν αλλά προβλήματα. Ένα από αυτά ήταν ότι το περιεχόμενο που αποθηκεύοταν ήταν το κείμενο του άρθρου όλο χωρίς κάποιο html tag. Όμως βρέθηκε λύση και έγινε η επίλυση αυτου του προσωρινού προβλήματος. Στην συνέχεια όμως, εντοπίστηκε το εξής πρόβλημα, το ίδιο το Nutch δεν μπορούσε να κάνει αναζήτηση σε url που περιλάμβαναν κάποια λεκτικά και κάποια σύμβολα, όπως ήταν η λέξη *search.* Αυτό που το εμπόδιζε ήταν το robot.txt του κάθε site το οποίο δεν μπορούσε να αλλαχτεί/απενεργοποιηθεί, οπότε από την στιγμή που υπήρχαν τέτοιοι περιορισμοί που κοίταξε το Nutch δεν γινόταν να συνεχιστεί η συνέχιση της εφαρμογης σε αυτό το περιβάλλον.

Οπότε, μετα απο τις παραπάνω προσπάθειες αρκετών μηνών, αποφασίστηκε να δημιουργηθεί ένα πρόγραμμα, απο την αρχή γραμμένο σε Python και με την βοήθεια του Selenium, το οποίο θα έψαχνε όποια σελίδα χρειαζόταν αντλώντας όλες τις πληροφορίες και οι οποίες υπήρχαν μέσα στον κώδικα html κάθε σελίδας.

7.3 Προτάσεις ανάπτυξης

7.3.1 Επεκτασιμότητα θέματος έρευνας

Για να γίνει πιο επεκτάσιμο και σε άλλα site αυτή η εφαρμογή θα πρέπει αρχικά να γίνει έλεγχος στα tag που περιέχεται η πληροφορία που χρειάζεται να συλλεχθεί. Οπότε θα πρέπει να βρεθεί σε ποιο div, για παράδειγμα, βρίσκεται ο τίτλος ή το περιεχόμενο του άρθρου. Έτσι ανάλογα τους selector που θα βρεθούν θα πρέπει να προστεθούν στο json object (allSelectors), τι οποίο περιέχει για κάθε site που χρησιμοποιείται στην εφαρμογή ποιοι selector ισχύουν για το κάθε ενα. Μεγάλη προσοχή θα πρέπει να δοθεί στην ημερομηνία του άρθρου και την δομή που θα υπάρχει ήδη στον κώδικα του άρθρου ώστε να μπορέσει να τροποποιηθεί με την βιβλιοθήκη που χρησιμοποιείται σε κοινό μοτίβο. Επίσης το id του άρθρου μπορεί να είναι είτε σε κάποιο div είτε στο ίδιο το url του άρθρου, αυτό διαφέρει ανάλογα το site. Όλες αυτές οι αλλαγές πρέπει να γίνουν στο αρχείου του scraper. Έτσι θα υπάρχει η αντίστοιχη δομή των πληροφοριών και η παρουσίαση τους στην εφαρμογή που υπάρχει ήδη θα γίνει αυτόματα, μιας και το μόνο που θα αλλαχτεί είναι το περιεχόμενο των δεδομένων αλλά όχι η δομή.

Για να γίνει επεκτάσιμο και σε άλλα θέματα αρκεί να αλλαχτούν τα url που είναι αποθηκευμένα στο Elasticsearch και να αντικατασταθούν με νέα url που περιέχουν την νέα κατηγορία θέματος. Στην συνέχεια, να διαγραφούν όλα τα δεδομένα από το ElasticSearch για τα άρθρα, ώστε να προστεθεί η νέα πληροφορία για το νέο θέμα έρευνας.

7.3.2 Δημιουργία βιβλιοθήκης

Θα ήταν πολύ ενδιαφέρον, αν φτιαχνόταν μια βιβλιοθήκη η οποία θα λάμβανε σαν παραμέτρους της κάποια σημαντικά και κομβικά στοιχεία. Αυτή η βιβλιοθήκη θα μπορούσε να χρησιμοποιηθεί και για γρήγορη επεκτασιμότητα.

Οι παράμετροι που θα χρειαζόντουσαν για να ολοκληρωθεί πλήρως το scraping για κάποιο θέμα έρευνας είναι οι εξής:

Για να οριστεί το περιβάλλον που θα γίνει το scraping χρειάζονται

- Τα url που θα πραγματοποιηθεί η αναζήτηση του scraper σε αυτά
- Και το domain του site

Επειδή χρειαζονται όμως συγκεκριμενες πληροφορίες για κάθε άρθρο θα χρειαστούν κάποιοι selector.

- **nextPage**: Όπως αναφέρθηκε και προηγουμένως το κάθε url που γίνεται scraping, είναι σελίδες που περιέχουν πολλά άρθρα για συγκεκριμένο άρθρο, κάπως σαν κατηγορία για

συγκεκριμένο θεμα. Οπότε χρειάζεται ο selector του κουμπιού για να μεταβεί στην επόμενη σελίδα των αποτελεσμάτων.

- **search**: Ο selector αυτός περιέχει όλα τα link άρθρων που περιεχονται σαν αποτέλεσμα, αλλά όχι αυτά που είναι γενικά στην σελίδα αυτή.
- **selectorId**: Αυτό το πεδίο θα μπορούσε να ήταν κενό αν το id του άρθρου είναι πάνω στο ίδιο το url του άρθρο. Αν δεν ισχύει αυτή η περίπτωση θα πρέπει να περιέχει τον selector του σημείου που βρίσκεται το id του άρθρου, συνήθως είναι σαν πατέρας του ίδιου του περιεχομένου του άρθρου.
- **urlImg**: Αυτός ο selector περιέχει το σημείο που βρίσκεται η κεντρική εικόνα του άρθρου.
- **title**: Αυτός ο selector περιέχει το σημείο που βρίσκεται ο τίτλος του άρθρου.
- **date**: Αυτός ο selector περιέχει το σημείο που βρίσκεται η ημερομηνία δημοσίευσης - και μερικές φορές ημερομηνία τροποποίησης- του άρθρου.
- **bodyDescription**: Αυτός ο selector περιέχει το σημείο που βρίσκεται η σύντομη περιγραφή του άρθρου.
- **body**: Αυτός ο selector περιέχει το σημείο που βρίσκεται όλο το περιεχόμενο του άρθρου.

Εκτός από αυτές τις παραμέτρους που αναφέρθηκαν παραπάνω θα ήταν απαραίτητο και πριν δημιουργηθεί η βιβλιοθήκη αυτή, να γίνουν κάποιοι έλεγχοι και δημιουργία περιπτώσεων σε ένα πλήθος διαφορετικών site. Για παράδειγμα, στο thebest.gr εντόπισα πρόβλημα στην ημερομηνία δημοσίευσης, οπότε έπρεπε να χειριστώ διαφορετικά αυτή την πληροφορία σε σύγκριση με το in.gr που είχα. Για αυτό τον λόγο θα ήταν δόκιμο να ελεγχθούν κάποια βασικά σημεία που ενδέχεται να διαφέρουν ανά site - εκτός φυσικά τους selectors που θα υπάρχουν ήδη σαν παράμετροι.

7.3.3 Προσθήκη έξυπνων ενεργειών

Θα ήταν ενδιαφέρον αν υπήρχε η δυνατότητα η εφαρμογή να καταλαβαίνει τον χρήστη. Μια ενέργεια που θα μπορούσε να πραγματοποιείται για να παρέχεται αυτή η δυνατότητα είναι, η εφαρμογή να καταλαβαίνει αν ο χρήστης που περιηγήθηκε στην εφαρμογή κάποια στιγμή επέλεξε άρθρα από συγκεκριμένο site, στην επόμενη είσοδο του στην εφαρμογή να του εμφανίζονται πρώτα τα άρθρα από το site που συνήθιζε να διαβάζει.

Επίσης άλλη μια ενδιαφέρουσα δυνατότητα θα ήταν να υπάρχουν πολλαπλά θέματα στην εφαρμογή, και όχι μόνο ένα όπως υπάρχει τώρα, από διαφορετικά site. Και στην συνέχεια η εφαρμογή να παρουσίαζε στον χρήστη ποια άρθρα διάβασε την τελευταία φορά είσοδος του στην εφαρμογή ή πόσα άρθρα διάβασε για την κάθε θεματολογία και στο μέλλον να εμφανίζει τις προτιμήσεις του σε μια επιπλέον σελίδα στην οποία θα είχε μόνο τις κατηγορίες που διαβάζει συνήθως.

Τέλος, άλλη μια δυνατότητα θα μπορούσε να ήταν η συλλογή των δεδομένων αναζήτησης των χρηστών. Δηλαδή να κρατούσε σε μια λίστα τι λέξεις-κλειδιά έκαναν αναζήτηση οι χρήστες τις εφαρμογής. Παρουσιάζοντας τα σε διάφορα διαγράμματα, ώστε να φαίνεται τις προτιμήσεις των χρηστών τις εφαρμογης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Άρθρα

[1] Amruta Mantri , Priyanka Nawale , Trupti Pardeshi , Rajeshwary Shisode , Reena Pagare - Profile Based Search Engine - 2013
[2] N. KOWSALYA - An Approach of Web Crawling and Indexing of Nutch - 2014
[3] Mini Singh Ahuja, Dr Jatinder Singh Bal, Varnica - Web Crawler: Extracting the Web Data - 2014
[4] Tim Allison, Wayne Burke, Valentino Constantinou, Edwin Goh, Chris Mattmann, Anastasija Mensikova, Philip Southam, Ryan Stonebraker, Virisha Timmaraju Jet Propulsion Laboratory, California Institute of Technology Pasadena, California - Research Report: Building a Wide Reach Corpus for Secure Parser Development - 2020
[5] Sparkler—Crawler on Apache Spark: Spark Summit East talk by Karanjeet Singh and Thamme Gowda Narayanaswamy
[6] Aleksei Voit, Aleksei Stankus, Shamil Magomedov, Irina Ivanova - Big Data Processing for Full-Text Search and Visualization with Elasticsearch - International Journal of Advanced Computer Science and Applications -2017
[7] James Hamilton, Brad Schofield, Manuel Gonzalez Berges, Jean-Charles Tournier - SCADA STATISTICS MONITORING USING THE Elastic Stack (Elasticsearch, Logstash, Kibana) - 16th Int. Conf. on Accelerator and Large Experimental Control Systems - 2017
[8] SOONHONG KWON AND JONG-HYOUK LEE - DIVDS: Docker Image Vulnerability Diagnostic System - SPECIAL SECTION ON SECURE COMMUNICATION FOR THE NEXT GENERATION 5G AND IOT NETWORKS - 2020
[9] Odersky, Martin ; Altherr, Philippe ; Cremet, Vincent ; Emir, Burak ; Maneth, Sebastian ; Micheloud, Stéphane ; Mihaylov, Nikolay ; Schinz, Michel ; Stenman, Erik ; Zenger, Matthias - An Overview of the Scala Programming Language - 2004
[10] Shahbaz Ali Syed, Tariq Rahim Soomro - Achieving Software Release Management and Continuous Integration using Maven, Jenkins and Artifactory - International Journal of Experiential Learning & Case Studies - 2018
[11] Márcio de OliveiraBarros, Fábio de AlmeidaFarzat,Guilherme HortaTravassos - Learning from optimization: A case study with Apache Ant - Information and Software Technology - 2015
[12] S. Nyamathulla, Dr. P. Ratnababu, Nazma Sultana Shaik,Bhagya Lakshmi. N - A Review on Selenium Web Driver with Python - 2021
[13] Sanchit Aggarwal - Modern Web-Development using ReactJS - 2018
[14] M. F. SANNER - PYTHON: A PROGRAMMING LANGUAGE FOR SOFTWARE INTEGRATION AND DEVELOPMENT

Ιστοσελίδες

https://www.iosrjournals.org/iosr-jce/papers/Vol16-issue1/Version-6/A016160105.pdf
https://www.cloudflare.com/learning/bots/what-is-a-web-crawler/
https://litslink.com/blog/what-is-a-web-crawler-and-how-does-it-work

https://www.researchgate.net/publication/265652647_AN_IMAGE_CRAWLER_FOR_CONTENT_BASED_IMAGE_RETRIEVAL_SYSTEM
https://www.octoparse.com/blog/top-5-social-media-scraping-tools-for-2021
https://logz.io/blog/elasticsearch-tutorial/
https://www.elastic.co/kibana/
https://aws.amazon.com/opensearch-service/the-elk-stack/kibana/
https://www.docker.com/
https://en.wikipedia.org/wiki/Docker_(software)
https://en.wikipedia.org/wiki/Scala_(programming_language)
https://www.scala-lang.org/
https://el.wikipedia.org/wiki/Scala
https://maven.apache.org/
https://ant.apache.org/
https://www.vogella.com/tutorials/ApacheAnt/article.html
https://www.scrapehero.com/best-web-crawling-tools-and-frameworks/
https://outsourceit.today/comparison-open-source-web-crawlers/
https://github.com/internetarchive/heritrix3
https://heritrix.readthedocs.io/en/latest/getting-started.html
http://stormcrawler.net/
https://github.com/scrapy/scrapy
https://pypi.org/project/Scrapy/
https://docs.zyte.com/
https://github.com/apify/apify-js
https://sdk.apify.com/
https://node-crawler.readthedocs.io/zh_CN/latest/
https://github.com/bda-research/node-crawler
https://analyticsindiamag.com/mechanicalsoup-web-scraping-custom-dataset-tutorial/
https://mechanicalsoup.readthedocs.io/en/stable/introduction.html
https://cwiki.apache.org/confluence/display/nutch/#Home-WhatisApacheNutch?
https://en.wikipedia.org/wiki/Apache_Nutch
https://www.linkedin.com/pulse/crawling-apache-nutch-maq-webster/
https://quicktomaster.com/web-crawling-with-nutch-and-elasticsearch/#
https://stackoverflow.com/questions/5123757/how-to-get-the-html-content-from-nutch
https://docs.google.com/document/d/1SU0YESlY5JViA9ezCSPr_SSF9e9VuvyFRICupGlfUKs/edit#
https://java.libhunt.com/sparkler-alternatives
https://github.com/ravindrabajpai/ana/blob/main/ground_zero
https://ntechcomputereducation.com/python-programming-language-for-beginners/
https://www.knstek.com/full-text-search-using-apache-lucene-part-i/

ΠΑΡΑΡΤΗΜΑΤΑ

Παράρτημα 1

Οδηγίες εγκατάστασης Apache Nutch

Εγκατάσταση java 8 και 11

```
sudo apt-get install openjdk-8-jdk
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get install openjdk-8-jdk
[sudo] password for pboviatsi:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni libice-dev
  libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-8-jdk-headless
  openjdk-8-jre openjdk-8-jre-headless x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  default-jre libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-8-demo openjdk-8-source visualvm
  icedtea-8-plugin fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei fonts-wqy-zenhei
The following NEW packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni libice-dev
  libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-8-jdk
  openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
0 upgraded, 21 newly installed, 0 to remove and 137 not upgraded.
Need to get 43.5 MB of archives.
After this operation, 162 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 java-common all 0.72 [6,816 B]
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates/universe amd64 openjdk-8-jre-headless amd64 8u312-b07-0ubuntu1~20.04 [28.2 MB]
```

```
java -version
```

```
pboviatsi@ubuntu:~/Desktop$ java -version
openjdk version "1.8.0_312"
OpenJDK Runtime Environment (build 1.8.0_312-8u312-b07-0ubuntu1~20.04-b07)
OpenJDK 64-Bit Server VM (build 25.312-b07, mixed mode)
pboviatsi@ubuntu:~/Desktop$
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$ java -version
openjdk version "11.0.15" 2022-04-19
OpenJDK Runtime Environment (build 11.0.15+10-Ubuntu-0ubuntu0.20.04.1)
OpenJDK 64-Bit Server VM (build 11.0.15+10-Ubuntu-0ubuntu0.20.04.1, mixed mode, sharing)
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$
```

Εγκατάσταση git

```
sudo apt install git
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 137 not upgraded.
Need to get 5,471 kB of archives.
After this operation, 38.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 liberror-perl all 0.17029-1 [26.5 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 git-man all 1:2.25.1-1ubuntu3.4 [885 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 git amd64 1:2.25.1-1ubuntu3.4 [4,560 kB]
Fetched 5,471 kB in 14s (381 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 156705 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17029-1_all.deb ...
Unpacking liberror-perl (0.17029-1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.25.1-1ubuntu3.4_all.deb ...
Unpacking git-man (1:2.25.1-1ubuntu3.4) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.25.1-1ubuntu3.4_amd64.deb ...
Unpacking git (1:2.25.1-1ubuntu3.4) ...
Setting up liberror-perl (0.17029-1) ...
Setting up git-man (1:2.25.1-1ubuntu3.4) ...
Setting up git (1:2.25.1-1ubuntu3.4) ...
Processing triggers for man-db (2.9.1-1) ...
pboviatsi@ubuntu:~/Desktop$
```

```
git --version
```

```
pboviatsi@ubuntu:~/Desktop$ git --version
git version 2.25.1
pboviatsi@ubuntu:~/Desktop$
```

Δημιουργία φακελου που θα περιέχει το project

```
mkdir nutch-crawler
cd nutch-crawler
```

```
pboviatsi@ubuntu:~/Desktop$ mkdir nutch-crawler
pboviatsi@ubuntu:~/Desktop$ cd nutch-crawler
pboviatsi@ubuntu:~/Desktop/nutch-crawler$
```

Εγκατάσταση ant

```
sudo apt install ant
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler$ sudo apt install ant
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ant-optional
Suggested packages:
  ant-doc antlr javacc junit junit4 jython libactivation-java libbcel-java libbsf-java libcommons-logging-java libcommons-net-java libmail-java libjaxp1.3-java
  libjdepend-java libjsch-java liblog4j1.2-java liboro-java libregexp-java libxalan2-java libxml-commons-resolver1.1-java libxz-java
The following NEW packages will be installed:
  ant ant-optional
0 upgraded, 2 newly installed, 0 to remove and 137 not upgraded.
Need to get 2,468 kB of archives.
After this operation, 3,415 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 ant all 1.10.7-1 [2,100 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 ant-optional all 1.10.7-1 [368 kB]
Fetched 2,468 kB in 6s (389 kB/s)
Selecting previously unselected package ant.
(Reading database ... 157640 files and directories currently installed.)
Preparing to unpack .../archives/ant_1.10.7-1_all.deb ...
Unpacking ant (1.10.7-1) ...
Selecting previously unselected package ant-optional.
Preparing to unpack .../ant-optional_1.10.7-1_all.deb ...
Unpacking ant-optional (1.10.7-1) ...
Setting up ant (1.10.7-1) ...
Setting up ant-optional (1.10.7-1) ...
Processing triggers for man-db (2.9.1-1) ...
pboviatsi@ubuntu:~/Desktop/nutch-crawler$
```

Εγκατάσταση του project

```
git clone https://github.com/apache/nutch.git
cd nutch
```

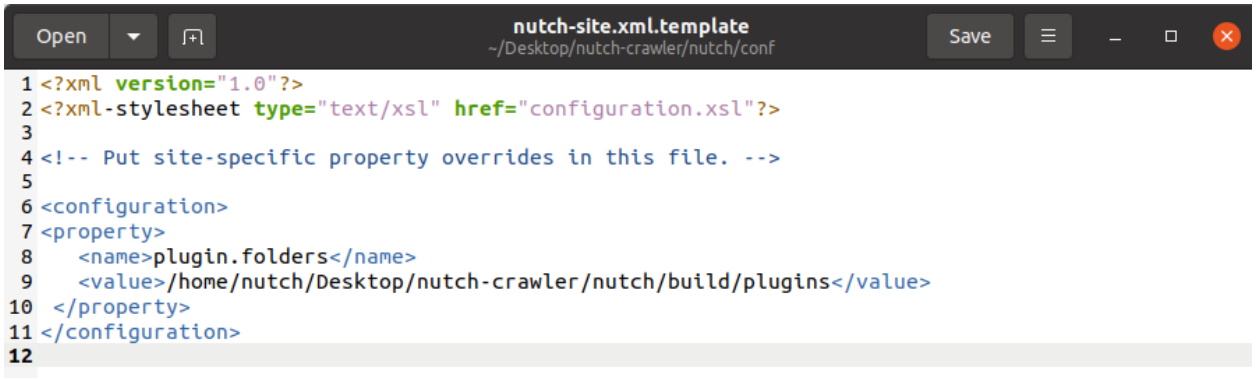
```
pboviatsi@ubuntu:~/Desktop/nutch-crawler$ git clone https://github.com/apache/nutch.git
Cloning into 'nutch'...
remote: Enumerating objects: 67498, done.
remote: Counting objects: 100% (1331/1331), done.
remote: Compressing objects: 100% (536/536), done.
remote: Total 67498 (delta 386), reused 1192 (delta 330), pack-reused 66167
Receiving objects: 100% (67498/67498), 133.27 MiB | 3.22 MiB/s, done.
Resolving deltas: 100% (32310/32310), done.
pboviatsi@ubuntu:~/Desktop/nutch-crawler$ cd nutch
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$
```

Επειδή δεν κάνει build το project, δοκιμάστηκε μια προσθήκη ενός property στο conf/nutch-site.xml αρχείο

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler$ ant clean runtime
Buildfile: build.xml does not exist!
Build failed
pboviatsi@ubuntu:~/Desktop/nutch-crawler$
```

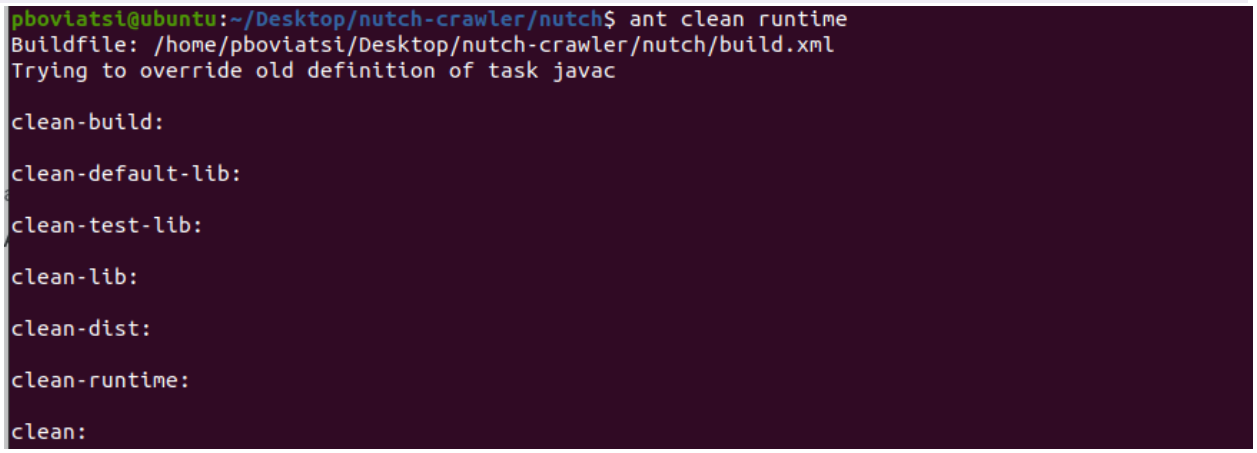
```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
   <name>plugin.folders</name>
   <value>/home/nutch/Desktop/nutch-crawler/nutch/build/plugins</value>
```

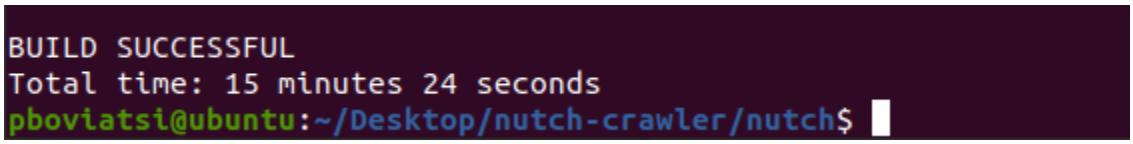
```
 </property>
</configuration>
```



Build του project

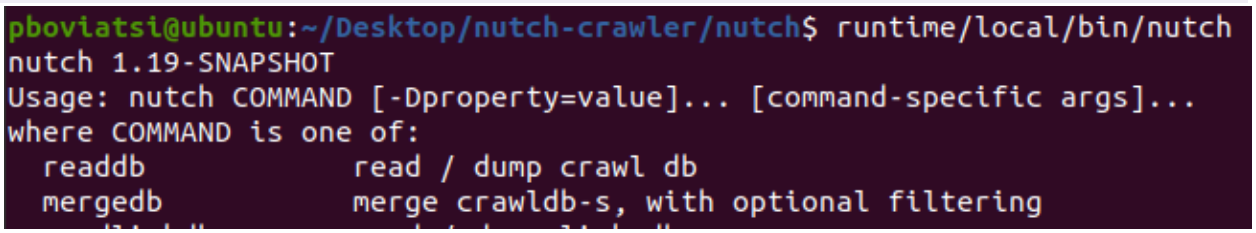
```
ant clean runtime
```





Για να γίνει επαλήθευση ότι παίζει το project

```
runtime/local/bin/nutch
```



Για να ξεκινήσει ο crawler πρέπει να ρυθμίσουμε κάποιες παραμέτρους του nutch, οπότε προσθέτουμε στο conf/nutch-site.xml καποια property

```
<property>
<name>http.agent.name</name>
<value>SICrawler</value>
<description>HTTP 'User-Agent' request header. MUST NOT be empty -
please set this to a single word uniquely related to your organization.
NOTE: You should also check other related properties:
http.robots.agents
http.agent.description
http.agent.url
http.agent.email
http.agent.version
and set their values appropriately.
</description>
</property>
<property>
<name>plugin.includes</name>
<value>protocol-http|urlfilter-regex|parse-(html|tika)|index-(basic|anchor)
|urlnormalizer-(pass|regex|basic)|scoring-opic|indexer-elastic</value>
</property>
<property>
<name>db.ignore.external.links</name>
<value>false</value>
<description>If true, outlinks leading from a page to external hosts or
domain
will be ignored. This is an effective way to limit the crawl to include
only initially injected hosts or domains, without creating complex
URLFilters.
See 'db.ignore.external.links.mode'.
</description>
</property>
<property>
<name>elastic.host</name>
<value>localhost</value>
<description>The hostname to send documents to using TransportClient.
Either host and port must be defined or cluster.
</description>
</property>
<property>
<name>elastic.port</name>
<value>9300</value>
<description>
```

```
The port to connect to using TransportClient.
</description>
</property>
<property>
<name>elastic.cluster</name>
<value>elasticsearch</value>
<description>The cluster name to discover. Either host and port must
be defined.
</description>
</property>
<property>
<name>elastic.index</name>
<value>nutch</value>
<description>
The name of the elasticsearch index. Will normally be autocreated if it
doesn't exist.
</description>
</property>
```

Στην συνέχεια, χρειάζεται η δημιουργία ενός νέου φακέλου που θα περιέχει τους ιστότοπου που θέλουμε να αντλήσουμε πληροφορίες.

```
mkdir runtime/local/urls
```

Ρυθμισμα Java Home

Χρειάζεται ορισμός του Java Home γιατι το elasticsearch χρειάζεται το jdk 11

```
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
echo $JAVA_HOME
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$ mkdir runtime/local/urls
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$ export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$ echo $JAVA_HOME
/usr/lib/jvm/java-11-openjdk-amd64
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$
```

Εγκατάσταση elasticsearch

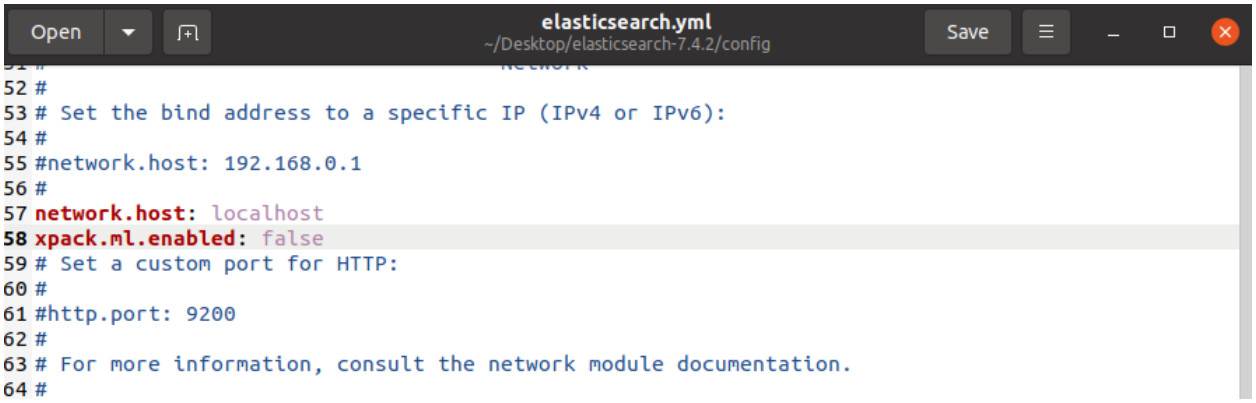
```
wget -c
https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.4.2-da
rwin-x86_64.tar.gz -O - | tar -xz
```

```
pboviatsi@ubuntu:~/Desktop$ wget -c https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.4.2-darwin-x86_64.tar.gz -O - | tar -xz
--2022-05-22 02:03:41--  https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.4.2-darwin-x86_64.tar.gz
Resolving artifacts.elastic.co (artifacts.elastic.co)... 34.120.127.130, 2600:1901:0:1d7::
Connecting to artifacts.elastic.co (artifacts.elastic.co)|34.120.127.130|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 279956874 (267M) [application/x-gzip]
Saving to: 'STDOUT'

-                           100%[=================================================>] 266.99M  5.10MB/s    in 48s

2022-05-22 02:04:30 (5.51 MB/s) - written to stdout [279956874/279956874]

pboviatsi@ubuntu:~/Desktop$
```

Ρύθμιση του elasticsearch

Προθήκη στο config/elasticsearch.yml

```
network.host: localhost
xpack.ml.enabled: false
```



Εκκίνηση του elasticsearch

```
bin/elasticsearch
```

Εγκατάσταση curl

```
pboviatsi@ubuntu:~/Desktop/elasticsearch-7.4.2$ sudo apt install curl
[sudo] password for pboviatsi:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libcurl4
The following NEW packages will be installed:
  curl libcurl4
0 upgraded, 2 newly installed, 0 to remove and 137 not upgraded.
Need to get 397 kB of archives.
After this operation, 1,121 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libcurl4 amd64 7.68.0-1ubuntu2.11 [235 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.11 [162 kB]
Fetched 397 kB in 1s (269 kB/s)
Selecting previously unselected package libcurl4:amd64.
(Reading database ... 158404 files and directories currently installed.)
```

Για να γίνει επαλήθευση ότι παίζει το elasticsearch

```
curl http://localhost:9200
```

```
pboviatsi@ubuntu:~/Desktop/elasticsearch-7.4.2$ curl http://localhost:9200
{
  "name" : "ubuntu",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "Mk2weHOtRKOJ8qEszZcrpA",
  "version" : {
    "number" : "7.4.2",
    "build_flavor" : "default",
    "build_type" : "tar",
    "build_hash" : "2f90bbf7b93631e52bafb59b3b049cb44ec25e96",
    "build_date" : "2019-10-28T20:40:44.881551Z",
    "build_snapshot" : false,
    "lucene_version" : "8.2.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
pboviatsi@ubuntu:~/Desktop/elasticsearch-7.4.2$
```

Εγκατάσταση Kibana

```
wget -c
https://artifacts.elastic.co/downloads/kibana/kibana-7.4.2-linux-x86_64.tar
.gz -O - | tar -xz
```

```
pboviatsi@ubuntu:~/Desktop$ wget -c https://artifacts.elastic.co/downloads/kibana/kibana-7.4.2-linux-x86_64.tar.g
z -O - | tar -xz
--2022-05-22 02:14:51--  https://artifacts.elastic.co/downloads/kibana/kibana-7.4.2-linux-x86_64.tar.gz
Resolving artifacts.elastic.co (artifacts.elastic.co)... 34.120.127.130, 2600:1901:0:1d7::
Connecting to artifacts.elastic.co (artifacts.elastic.co)|34.120.127.130|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 252554263 (241M) [application/x-gzip]
Saving to: 'STDOUT'

-                           100%[=========================================>] 240.85M  10.2MB/s    in 40s

2022-05-22 02:15:32 (5.95 MB/s) - written to stdout [252554263/252554263]

pboviatsi@ubuntu:~/Desktop$
```

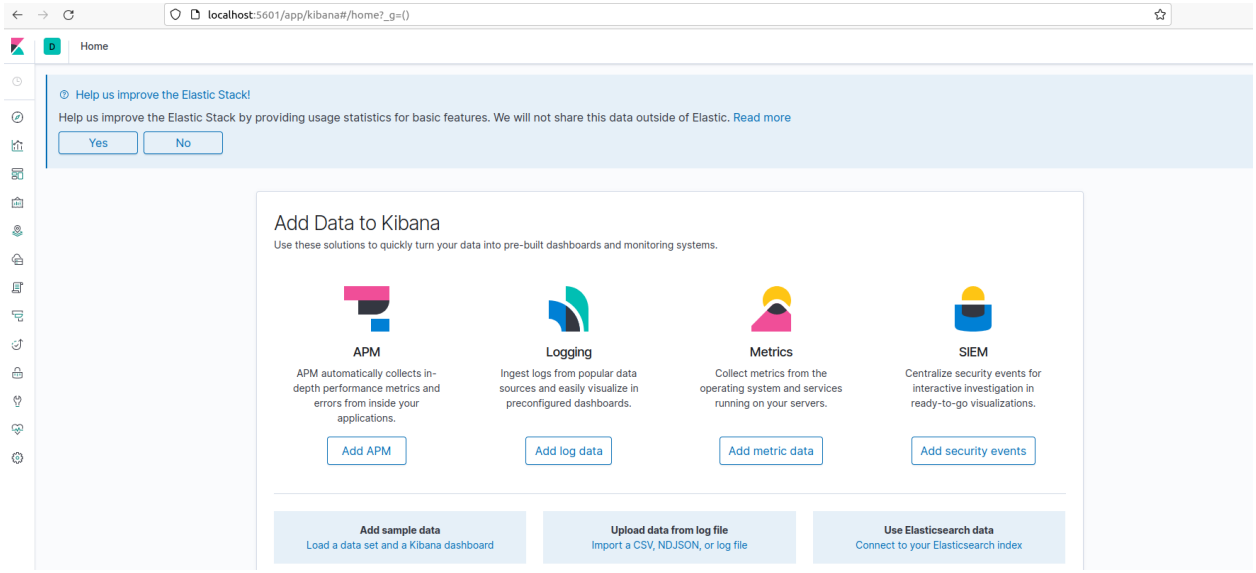
```
cd /crawler/kibana-7.4.2
```

Εκκίνηση του kibana

```
bin/kibana
```

```
pboviatsi@ubuntu:~/Desktop$ cd kibana-7.4.2-linux-x86_64/
pboviatsi@ubuntu:~/Desktop/kibana-7.4.2-linux-x86_64$ bin/kibana
  log   [09:16:53.719] [info][plugins-system] Setting up [4] plugins: [security,translations,inspector,data]
  log   [09:16:53.745] [info][plugins][security] Setting up plugin
  log   [09:16:53.751] [warning][config][plugins][security] Generating a random key for xpack.security.encryption
Key. To prevent sessions from being invalidated on restart, please set xpack.security.encryptionKey in kibana.yml
  log   [09:16:53.754] [warning][config][plugins][security] Session cookies will be transmitted over insecure con
nections. This is not recommended.
  log   [09:16:54.083] [info][plugins][translations] Setting up plugin
  log   [09:16:54.085] [info][data][plugins] Setting up plugin
  log   [09:16:54.098] [info][plugins-system] Starting [3] plugins: [security,translations,data]
```

Για να γίνει επαλήθευση ότι παίζει το kibana στο http://localhost:5601/



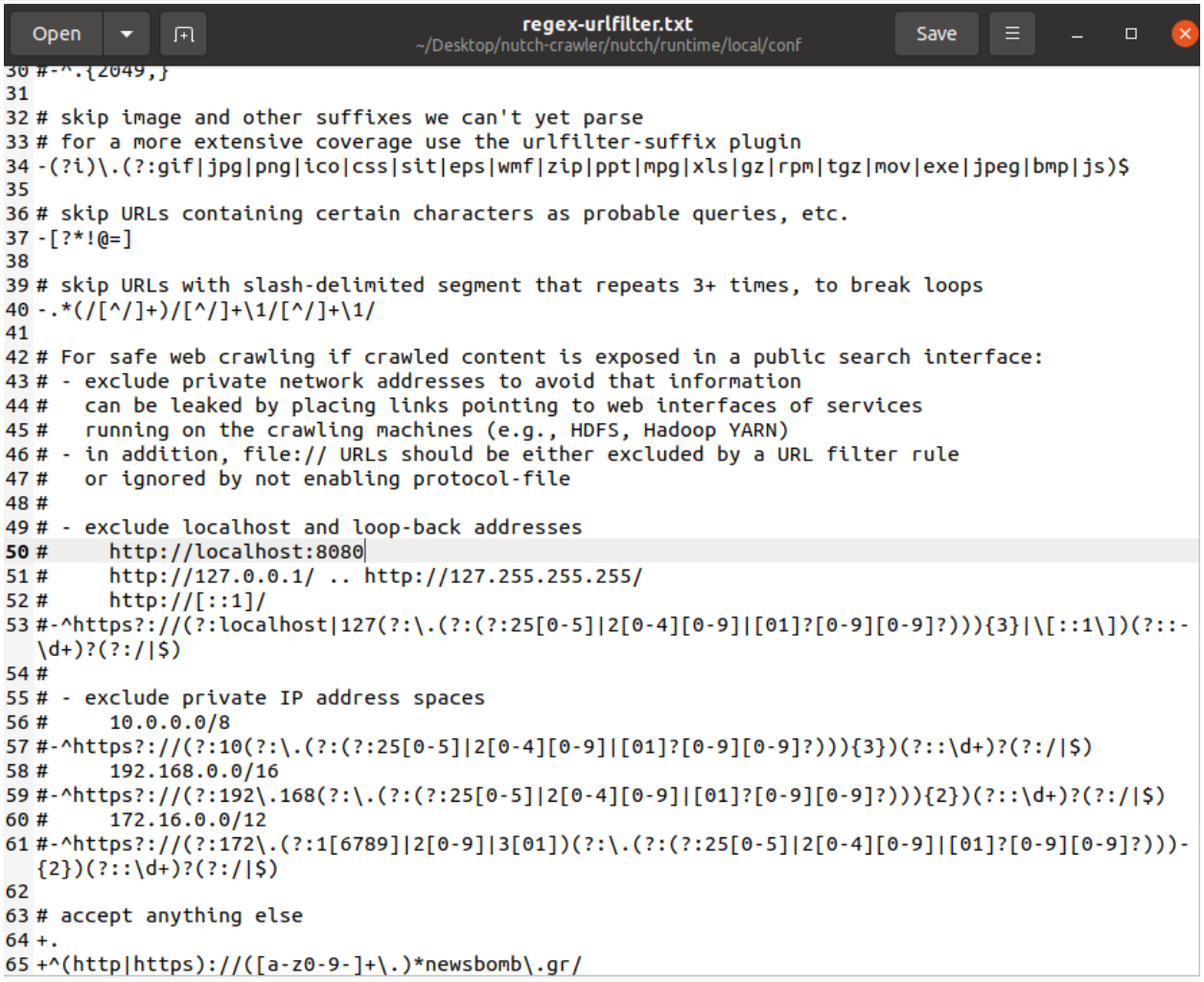
Αλλαγές στα αρχεία του φακέλου του nutch

Δημιουργία του αρχείου seed.txt στον φάκελο runtime/local/urls και του βάζουμε τα url απο τα οποία θα αντλήσουμε όλη την πληροφορία

```
https://www.newsbomb.gr/
```

Στην συνέχεια χρειάζεται η επεξεργασία του runtime/local/conf/regex-urlfilter.txt για να μπορεί να διαλέγει όλα τα url που περιέχουν ένα κομμάτι του url.

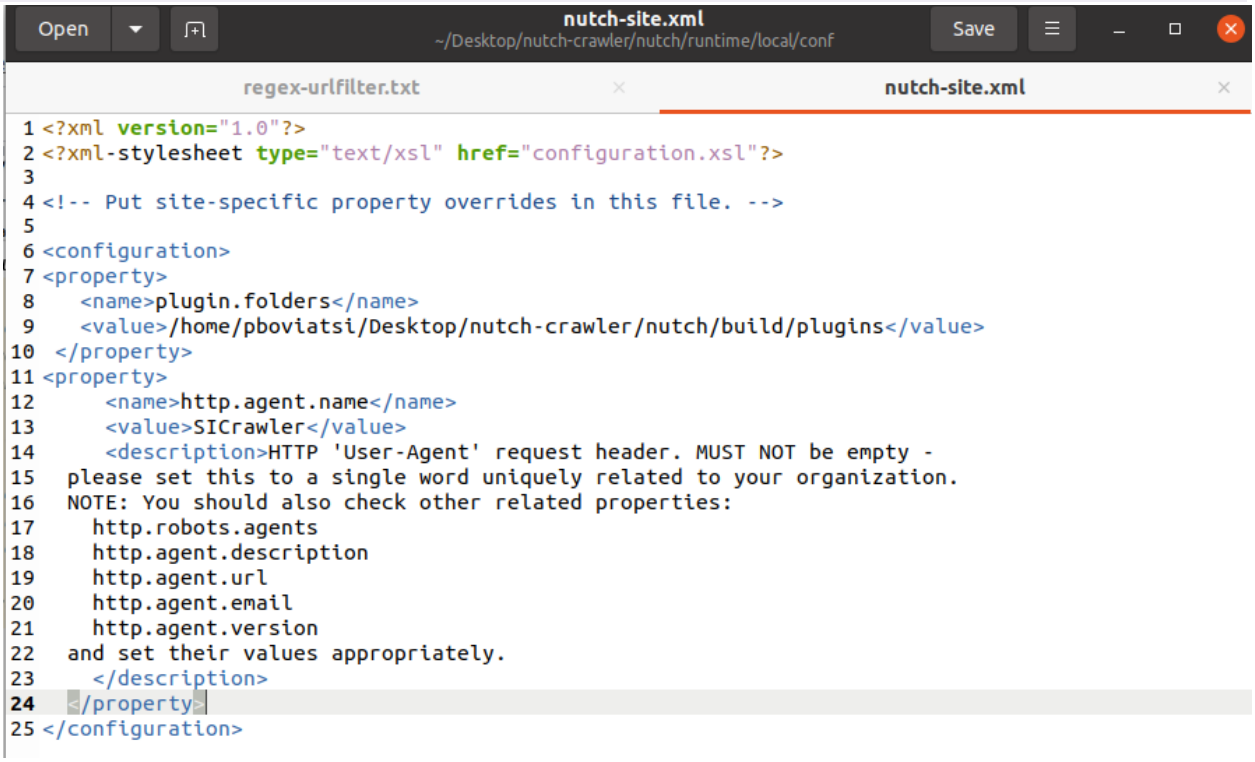
```
+^(http|https)://([a-z0-9-]+\.)*newsbomb\.gr/
```



Για να ξεκινήσει ο crawler πρέπει να ρυθμίσουμε κάποιες ακόμα παραμέτρους του nutch, οπότε προσθέτουμε στο conf/nutch-site.xml καποια property

```
<property>
      <name>http.agent.name</name>
```

```
    <value>SICrawler</value>
    <description>HTTP 'User-Agent' request header. MUST NOT be empty -
  please set this to a single word uniquely related to your organization.
  NOTE: You should also check other related properties:
    http.robots.agents
    http.agent.description
    http.agent.url
    http.agent.email
    http.agent.version
  and set their values appropriately.
    </description>
  </property>
```



Διαβασμα του αρχικό αρχείου και δημιουργία ή ενημέρωση του κατάλογου crawldb

```
cd runtime/local
bin/nutch inject crawl/crawldb urls
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch$ cd runtime/local
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch inject crawl/crawldb urls
2022-05-22 02:32:22,505 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:32:23,545 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:32:23,548 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:32:23,549 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:32:23,550 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:32:23,550 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:32:23,551 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
```

Στην συνέχεια διαβάζονται όλα τα url απο το crawldb για να δημιουργηθεί ένα τμήμα (segment), ώστε να ξεκινήσει η διαδικασία

```
bin/nutch generate crawl/crawldb crawl/segments
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch generate crawl/crawldb crawl/segments
2022-05-22 02:34:18,232 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:34:19,254 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:34:19,272 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:34:19,280 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:34:19,280 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:34:19,281 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:34:19,281 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:34:19,282 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:34:19,282 INFO o.a.n.p.PluginRepository [main]    Anchor Indexing Filter (index-anchor)
2022-05-22 02:34:19,282 INFO o.a.n.p.PluginRepository [main]    Tika Parser Plug-in (parse-tika)
2022-05-22 02:34:19,283 INFO o.a.n.p.PluginRepository [main]    Basic URL Normalizer (urlnormalizer-basic)
```

Εκχωρείται το τμήμα (segment) σε μία μεταβλητή για να μπορεί να γίνει η χρήση του στην συνέχεια

```
s1=`ls -d crawl/segments/2* | tail -1`
echo $s1
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ s1=`ls -d crawl/segments/2* | tail -1`
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ echo $s1
crawl/segments/20220522023427
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Ανάκτηση του πραγματικού περιεχομένου καθεμιάς από τις διευθύνσεις URL και αποθήκευση τους στον αντίστοιχο φάκελο τμημάτων τους.

```
bin/nutch fetch $s1
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch fetch $s1
2022-05-22 02:36:08,055 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:36:08,771 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:36:08,774 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:36:08,775 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:36:08,776 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:36:08,779 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:36:08,779 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:36:08,780 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:36:08,780 INFO o.a.n.p.PluginRepository [main]    Anchor Indexing Filter (index-anchor)
2022-05-22 02:36:08,781 INFO o.a.n.p.PluginRepository [main]    Tika Parser Plug-in (parse-tika)
2022-05-22 02:36:08,782 INFO o.a.n.p.PluginRepository [main]    Basic URL Normalizer (urlnormalizer-basic)
2022-05-22 02:36:08,782 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter Framework (lib-regex-filter)
2022-05-22 02:36:08,783 INFO o.a.n.p.PluginRepository [main]    Regex URL Normalizer (urlnormalizer-regex)
```

```
2022-05-22 02:36:15,890 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=1, spinWaiting=0, fetchQueues.totalSize=0, fetchQueues.getQueueCount=1
2022-05-22 02:36:16,899 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=1, spinWaiting=0, fetchQueues.totalSize=0, fetchQueues.getQueueCount=1
2022-05-22 02:36:17,170 INFO o.a.n.f.FetcherThread [FetcherThread] FetcherThread 52 has no more work available
2022-05-22 02:36:17,171 INFO o.a.n.f.FetcherThread [FetcherThread] FetcherThread 52 -finishing thread FetcherThread, activeThreads=0
2022-05-22 02:36:17,902 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=0, spinWaiting=0, fetchQueues.totalSize=0, fetchQueues.getQueueCount=0
2022-05-22 02:36:17,903 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=0
2022-05-22 02:36:19,393 INFO o.a.n.f.Fetcher [main] Fetcher: finished at 2022-05-22 02:36:19, elapsed: 00:00:10
```

Μετατροπή του περιεχομένου σε json

```
bin/nutch parse $s1
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch parse $s1
2022-05-22 02:36:42,663 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:36:44,215 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:36:44,221 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:36:44,221 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:36:44,222 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:36:44,222 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:36:44,223 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:36:44,223 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:36:44,224 INFO o.a.n.p.PluginRepository [main]    Anchor Indexing Filter (index-anchor)
2022-05-22 02:36:44,225 INFO o.a.n.p.PluginRepository [main]    Tika Parser Plug-in (parse-tika)
2022-05-22 02:36:44,225 INFO o.a.n.p.PluginRepository [main]    Basic URL Normalizer (urlnormalizer-basic)
2022-05-22 02:36:44,226 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter Framework (lib-regex-filter)
2022-05-22 02:36:44,226 INFO o.a.n.p.PluginRepository [main]    Regex URL Normalizer (urlnormalizer-regex)
```

```
2022-05-22 02:36:51,316 INFO o.a.n.c.SignatureFactory [LocalJobRunner Map Task Executor #0] Using Signature impl: org.apache.nutch.crawl.MD5Signatu
re
2022-05-22 02:36:51,351 INFO o.a.n.p.ParseSegment [LocalJobRunner Map Task Executor #0] Parsed (1429ms): https://www.newsbomb.gr/
2022-05-22 02:36:51,903 INFO o.a.n.n.URLExemptionFilters [pool-6-thread-1] Found 0 extensions at point:'org.apache.nutch.net.URLExemptionFilter'
2022-05-22 02:36:52,293 INFO o.a.n.n.u.r.RegexURLNormalizer [pool-6-thread-1] can't find rules for scope 'outlink', using default
2022-05-22 02:36:53,358 INFO o.a.n.p.ParseSegment [main] ParseSegment: finished at 2022-05-22 02:36:53, elapsed: 00:00:07
```

Ενημέρωση του crawldb με την παραπάνω πληροφορία που συλλέχτηκε

```
bin/nutch updatedb crawl/crawldb $s1
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch updatedb crawl/crawldb $s1
2022-05-22 02:37:11,177 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:37:11,982 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:37:11,986 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:37:11,989 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:37:11,990 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:37:11,991 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:37:11,992 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:37:11,999 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:37:12,000 INFO o.a.n.p.PluginRepository [main]    Anchor Indexing Filter (index-anchor)
2022-05-22 02:37:12,001 INFO o.a.n.p.PluginRepository [main]    Tika Parser Plug-in (parse-tika)
2022-05-22 02:37:12,002 INFO o.a.n.p.PluginRepository [main]    Basic URL Normalizer (urlnormalizer-basic)
2022-05-22 02:37:12,003 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter Framework (lib-regex-filter)
```

```
2022-05-22 02:37:14,973 INFO o.a.n.p.h.Http [main] http.enable.cookie.header = true
2022-05-22 02:37:19,841 INFO o.a.n.c.FetchScheduleFactory [pool-5-thread-1] Using FetchSchedule impl: org.apache.nutch.crawl.DefaultFetchSchedule
2022-05-22 02:37:19,844 INFO o.a.n.c.AbstractFetchSchedule [pool-5-thread-1] defaultInterval=2592000
2022-05-22 02:37:19,846 INFO o.a.n.c.AbstractFetchSchedule [pool-5-thread-1] maxInterval=7776000
2022-05-22 02:37:20,281 INFO o.a.n.c.CrawlDb [main] CrawlDb update: finished at 2022-05-22 02:37:20, elapsed: 00:00:06
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Επανάληψη της διαδικασίας για τα πρώτα 1000 url που βρέθηκαν στο url που δώθηκε στην αρχή

```
bin/nutch generate crawl/crawldb crawl/segments -topN 1000
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch generate crawl/crawldb crawl/segments -topN 1000
2022-05-22 02:39:09,028 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:39:10,440 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:39:10,443 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:39:10,444 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:39:10,444 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:39:10,445 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:39:10,445 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:39:10,445 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:39:10,446 INFO o.a.n.p.PluginRepository [main]    Anchor Indexing Filter (index-anchor)
```

```
2022-05-22 02:39:15,977 INFO o.a.n.n.u.r.RegexURLNormalizer [pool-5-thread-1] can't find rules for scope 'generate_host_count', using default
2022-05-22 02:39:16,857 INFO o.a.n.c.Generator [main] Generator: number of items rejected during selection:
2022-05-22 02:39:16,870 INFO o.a.n.c.Generator [main] Generator:      1  SCHEDULE_REJECTED
2022-05-22 02:39:16,885 INFO o.a.n.c.Generator [main] Generator: Partitioning selected urls for politeness.
2022-05-22 02:39:17,894 INFO o.a.n.c.Generator [main] Generator: segment: crawl/segments/20220522023917
2022-05-22 02:39:19,578 INFO o.a.n.c.Generator [main] Generator: finished at 2022-05-22 02:39:19, elapsed: 00:00:07
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Εκχωρείται ένα νέο τμήμα (segment) σε μία μεταβλητή για να μπορεί να γίνει η χρήση του στην συνέχεια

```
s2=`ls -d crawl/segments/2* | tail -1`
echo $s2
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ s2=`ls -d crawl/segments/2* | tail -1`
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ echo $s2
crawl/segments/20220522023917
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Ανάκτηση του πραγματικού περιεχομένου καθεμιάς από τις διευθύνσεις URL και αποθήκευση τους στον αντίστοιχο φάκελο τμημάτων τους.

```
bin/nutch fetch $s2
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch fetch $s2
2022-05-22 02:42:09,443 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:42:10,098 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:42:10,099 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:42:10,100 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:42:10,101 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:42:10,101 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:42:10,101 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:42:10,101 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:50:13,487 INFO o.a.n.f.FetcherThread [FetcherThread] FetcherThread 54 has no more work available
2022-05-22 02:50:13,487 INFO o.a.n.f.FetcherThread [FetcherThread] FetcherThread 54 -finishing thread FetcherThread, activeThreads=0
2022-05-22 02:50:14,182 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=0, spinWaiting=0, fetchQueues.totalSize=0, fetch
Queues.getQueueCount=0
2022-05-22 02:50:14,183 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=0
2022-05-22 02:50:17,109 INFO o.a.n.f.Fetcher [main] Fetcher: finished at 2022-05-22 02:50:17, elapsed: 00:08:06
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Μετατροπή του περιεχομένου σε json

```
bin/nutch parse $s2
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch parse $s2
2022-05-22 02:51:55,462 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:51:56,057 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:51:56,058 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:51:56,059 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:51:56,059 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:51:56,059 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:51:56,060 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:51:56,060 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:51:56,060 INFO o.a.n.p.PluginRepository [main]    Anchor Indexing Filter (index-anchor)
2022-05-22 02:52:10,558 INFO o.a.n.p.ParseSegment [LocalJobRunner Map Task Executor #0] Parsed (53ms): https://www.newsbomb.gr/ygeia/story/1309584
/pos-ginetai-i-skoni-proteinis-kai-poia-einai-i-katallili-gia-esas
2022-05-22 02:52:11,201 INFO o.a.n.n.URLExemptionFilters [pool-6-thread-1] Found 0 extensions at point:'org.apache.nutch.net.URLExemptionFilter'
2022-05-22 02:52:11,577 INFO o.a.n.n.u.r.RegexURLNormalizer [pool-6-thread-1] can't find rules for scope 'outlink', using default
2022-05-22 02:52:16,805 INFO o.a.n.p.ParseSegment [main] ParseSegment: finished at 2022-05-22 02:52:16, elapsed: 00:00:20
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Ενημέρωση του crawldb με την παραπάνω πληροφορία που συλλέχτηκε

```
bin/nutch updatedb crawl/crawldb $s2
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch updatedb crawl/crawldb $s2
2022-05-22 02:52:38,990 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:52:39,625 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:52:39,626 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:52:39,627 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:52:39,628 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:52:39,628 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:52:39,628 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:52:39,629 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:52:39,629 INFO o.a.n.p.PluginRepository [main]    Anchor Indexing Filter (index-anchor)
2022-05-22 02:52:41,100 INFO o.a.n.p.h.Http [main] http.enable.cookie.header = true
2022-05-22 02:52:44,346 INFO o.a.n.c.FetchScheduleFactory [pool-5-thread-1] Using FetchSchedule impl: org.apache.nutch.crawl.DefaultFetchSchedule
2022-05-22 02:52:44,349 INFO o.a.n.c.AbstractFetchSchedule [pool-5-thread-1] defaultInterval=2592000
2022-05-22 02:52:44,350 INFO o.a.n.c.AbstractFetchSchedule [pool-5-thread-1] maxInterval=7776000
2022-05-22 02:52:45,291 INFO o.a.n.c.CrawlDb [main] CrawlDb update: finished at 2022-05-22 02:52:45, elapsed: 00:00:04
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Επανάληψη της διαδικασίας για τα επόμενα 1000 url που βρέθηκαν στο url

```
bin/nutch generate crawl/crawldb crawl/segments -topN 1000
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch generate crawl/crawldb crawl/segments -topN 1000
2022-05-22 02:53:14,241 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:53:14,762 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:53:14,764 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:53:14,765 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:53:14,765 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:53:14,766 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:53:14,766 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:53:14,767 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:53:19,718 INFO o.a.n.c.Generator [main] Generator:     96  SCHEDULE_REJECTED
2022-05-22 02:53:19,722 INFO o.a.n.c.Generator [main] Generator: Partitioning selected urls for politeness.
2022-05-22 02:53:20,724 INFO o.a.n.c.Generator [main] Generator: segment: crawl/segments/20220522025320
2022-05-22 02:53:22,166 INFO o.a.n.c.Generator [main] Generator: finished at 2022-05-22 02:53:22, elapsed: 00:00:06
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Εκχωρείται ένα νέο τμήμα (segment) σε μία μεταβλητή για να μπορεί να γίνει η χρήση του στην συνέχεια

```
s3=`ls -d crawl/segments/2* | tail -1`
echo $s3
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ s3=`ls -d crawl/segments/2* | tail -1`
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ echo $s3
crawl/segments/20220522025320
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Ανάκτηση του πραγματικού περιεχομένου καθεμιάς από τις διευθύνσεις URL και αποθήκευση τους στον αντίστοιχο φάκελο τμημάτων τους.

```
bin/nutch fetch $s3
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch fetch $s3
2022-05-22 02:54:11,594 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 02:54:12,164 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 02:54:12,166 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 02:54:12,166 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 02:54:12,167 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 02:54:12,167 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 02:54:12,168 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 02:54:12,169 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 02:59:53,923 INFO o.a.n.f.FetcherThread [FetcherThread] FetcherThread 61 -finishing thread FetcherThread, activeThreads=0
2022-05-22 02:59:54,315 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=0, spinWaiting=0, fetchQueues.totalSize=0, fetch
Queues.getQueueCount=0
2022-05-22 02:59:54,315 INFO o.a.n.f.Fetcher [LocalJobRunner Map Task Executor #0] -activeThreads=0
2022-05-22 02:59:56,173 INFO o.a.n.f.Fetcher [main] Fetcher: finished at 2022-05-22 02:59:56, elapsed: 00:05:43
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Μετατροπή του περιεχομένου σε json

```
bin/nutch parse $s3
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch parse $s3
2022-05-22 03:07:58,320 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 03:07:58,914 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 03:07:58,915 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 03:07:58,916 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 03:07:58,916 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 03:07:58,916 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 03:07:58,917 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 03:08:09,994 INFO o.a.n.p.ParseSegment [LocalJobRunner Map Task Executor #0] Parsed (21ms): https://www.newsbomb.gr/ygeia/story/1309584
/pos-ginetai-i-skoni-proteinis-kai-poia-einai-i-katallili-gia-esas/amp
2022-05-22 03:08:10,544 INFO o.a.n.n.URLExemptionFilters [pool-6-thread-1] Found 0 extensions at point:'org.apache.nutch.net.URLExemptionFilter'
2022-05-22 03:08:10,949 INFO o.a.n.n.u.r.RegexURLNormalizer [pool-6-thread-1] can't find rules for scope 'outlink', using default
2022-05-22 03:08:14,596 INFO o.a.n.p.ParseSegment [main] ParseSegment: finished at 2022-05-22 03:08:14, elapsed: 00:00:15
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Ενημέρωση του crawldb με την παραπάνω πληροφορία που συλλέχτηκε

```
bin/nutch updatedb crawl/crawldb $s3
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch updatedb crawl/crawldb $s3
2022-05-22 03:10:21,792 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 03:10:22,456 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 03:10:22,462 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 03:10:22,462 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 03:10:22,463 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 03:10:22,463 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 03:10:22,463 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 03:10:22,464 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
2022-05-22 03:10:26,866 INFO o.a.n.c.FetchScheduleFactory [pool-5-thread-1] Using FetchSchedule impl: org.apache.nutch.crawl.DefaultFetchSchedule
2022-05-22 03:10:26,870 INFO o.a.n.c.AbstractFetchSchedule [pool-5-thread-1] defaultInterval=2592000
2022-05-22 03:10:26,871 INFO o.a.n.c.AbstractFetchSchedule [pool-5-thread-1] maxInterval=7776000
2022-05-22 03:10:27,378 INFO o.a.n.c.CrawlDb [main] CrawlDb update: finished at 2022-05-22 03:10:27, elapsed: 00:00:04
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Ενημέρωση του linkdb με όσα στοιχεία έχουμε συλλέξει

```
bin/nutch invertlinks crawl/linkdb -dir crawl/segments
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch invertlinks crawl/linkdb -dir crawl/segments
2022-05-22 03:11:20,946 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 03:11:21,770 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 03:11:21,773 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 03:11:21,773 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 03:11:21,774 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 03:11:21,775 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 03:11:21,776 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 03:11:26,864 INFO o.a.n.p.h.Http [LocalJobRunner Map Task Executor #0] http.accept = text/html,application/xhtml+xml,application/xml;q=0.9,*
/*;q=0.8
2022-05-22 03:11:26,864 INFO o.a.n.p.h.Http [LocalJobRunner Map Task Executor #0] http.enable.cookie.header = true
2022-05-22 03:11:26,869 INFO o.a.n.n.u.r.RegexURLNormalizer [LocalJobRunner Map Task Executor #0] can't find rules for scope 'linkdb', using default
2022-05-22 03:11:27,966 INFO o.a.n.n.u.r.RegexURLNormalizer [LocalJobRunner Map Task Executor #0] can't find rules for scope 'linkdb', using default
2022-05-22 03:11:29,008 INFO o.a.n.n.u.r.RegexURLNormalizer [LocalJobRunner Map Task Executor #0] can't find rules for scope 'linkdb', using default
2022-05-22 03:11:30,322 INFO o.a.n.c.LinkDb [main] LinkDb: finished at 2022-05-22 03:11:30, elapsed: 00:00:07
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

Για να ξεκινήσει ο crawler πρέπει να ρυθμίσουμε κάποιες ακόμα παραμέτρους του nutch, οπότε προσθέτουμε στο conf/nutch-site.xml καποια property

```
<property>
      <name>plugin.includes</name>

<value>protocol-http|urlfilter-regex|parse-(html|tika)|index-(basic|anchor)
|urlnormalizer-(pass|regex|basic)|scoring-opic|indexer-elastic</value>
  </property>
  <property>
      <name>db.ignore.external.links</name>
      <value>false</value>
      <description>If true, outlinks leading from a page to external hosts
or domain
      will be ignored. This is an effective way to limit the crawl to
include
      only initially injected hosts or domains, without creating complex
URLFilters.
      See 'db.ignore.external.links.mode'.
      </description>
  </property>
  <property>
      <name>elastic.host</name>
      <value>localhost</value>
```

```
      <description>The hostname to send documents to using TransportClient.
      Either host and port must be defined or cluster.
      </description>
  </property>
  <property>
      <name>elastic.port</name>
      <value>9300</value>
      <description>
      The port to connect to using TransportClient.
      </description>
  </property>
  <property>
      <name>elastic.cluster</name>
      <value>elasticsearch</value>
      <description>The cluster name to discover. Either host and port must
      be defined.
      </description>
  </property>
  <property>
      <name>elastic.index</name>
      <value>nutch</value>
      <description>
      The name of the elasticsearch index. Will normally be autocreated if
it
      doesn't exist.
      </description>
  </property>
```

Open | nutch-site.xml ~/Desktop/nutch-crawler/nutch/runtime/local/conf | Save

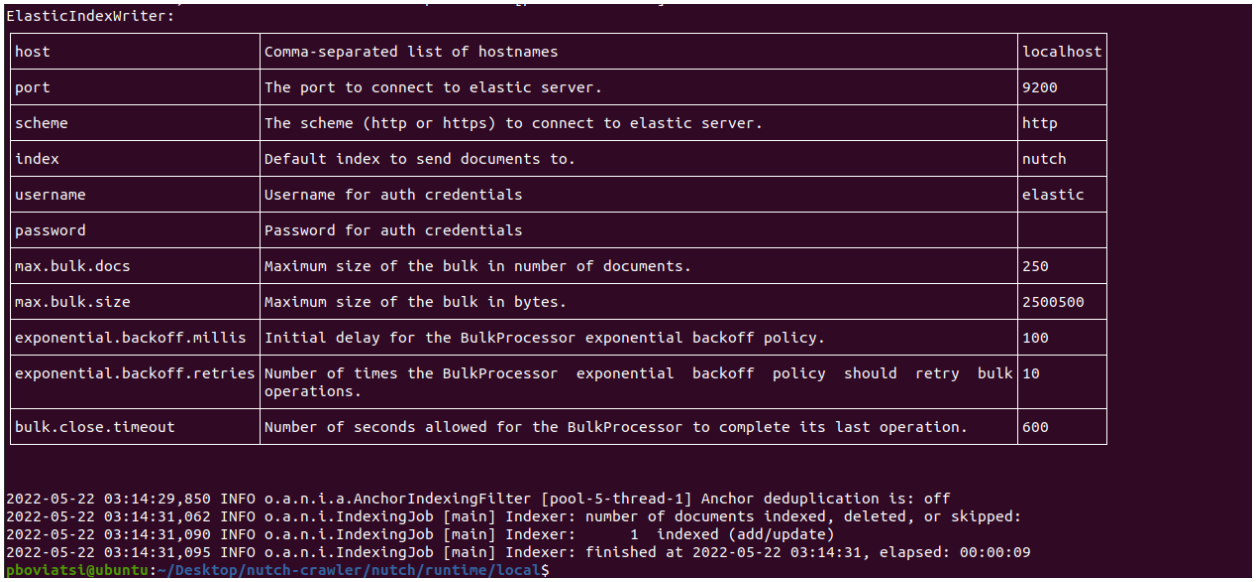
regex-urlfilter.txt | nutch-site.xml

```
  </property>
  <property>
     <name>plugin.includes</name>
     <value>protocol-http|urlfilter-regex|parse-(html|tika)|index-(basic|anchor)|-
urlnormalizer-(pass|regex|basic)|scoring-opic|indexer-elastic</value>
  </property>
  <property>
      <name>db.ignore.external.links</name>
      <value>false</value>
      <description>If true, outlinks leading from a page to external hosts or domain
      will be ignored. This is an effective way to limit the crawl to include
      only initially injected hosts or domains, without creating complex URLFilters.
      See 'db.ignore.external.links.mode'.
      </description>
  </property>
  <property>
      <name>elastic.host</name>
      <value>localhost</value>
      <description>The hostname to send documents to using TransportClient.
      Either host and port must be defined or cluster.
      </description>
  </property>
  <property>
      <name>elastic.port</name>
      <value>9300</value>
      <description>
      The port to connect to using TransportClient.
      </description>
  </property>
  <property>
      <name>elastic.cluster</name>
      <value>elasticsearch</value>
      <description>The cluster name to discover. Either host and port must
      be defined.
      </description>
  </property>
  <property>
      <name>elastic.index</name>
      <value>nutch</value>
      <description>
      The name of the elasticsearch index. Will normally be autocreated if it
      doesn't exist.
      </description>
  </property>
</configuration>
```

Αποθήκευση όλων των δεδομένων στο elasticSearch

```
bin/nutch index crawl/crawldb/ -linkdb crawl/linkdb/ $s1 -filter -normalize
-deleteGone
```

```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch index crawl/crawldb/ -linkdb crawl/linkdb/ $s1 -filter -normalize -deleteGone
2022-05-22 03:14:19,887 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 03:14:20,679 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 03:14:20,681 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 03:14:20,682 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 03:14:20,683 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 03:14:20,684 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 03:14:20,684 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
2022-05-22 03:14:20,685 INFO o.a.n.p.PluginRepository [main]    Basic Indexing Filter (index-basic)
```

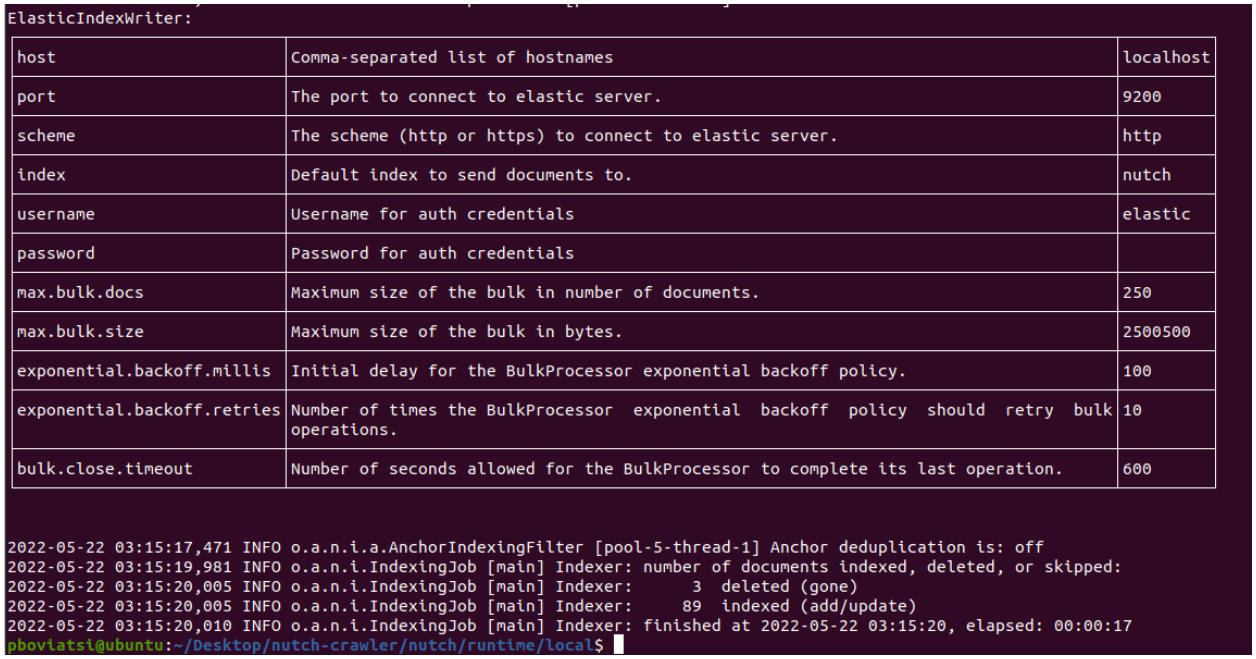


ElasticIndexWriter:

host	Comma-separated list of hostnames	localhost
port	The port to connect to elastic server.	9200
scheme	The scheme (http or https) to connect to elastic server.	http
index	Default index to send documents to.	nutch
username	Username for auth credentials	elastic
password	Password for auth credentials	
max.bulk.docs	Maximum size of the bulk in number of documents.	250
max.bulk.size	Maximum size of the bulk in bytes.	2500500
exponential.backoff.millis	Initial delay for the BulkProcessor exponential backoff policy.	100
exponential.backoff.retries	Number of times the BulkProcessor exponential backoff policy should retry bulk operations.	10
bulk.close.timeout	Number of seconds allowed for the BulkProcessor to complete its last operation.	600

```
2022-05-22 03:14:29,850 INFO o.a.n.i.a.AnchorIndexingFilter [pool-5-thread-1] Anchor deduplication is: off
2022-05-22 03:14:31,062 INFO o.a.n.i.IndexingJob [main] Indexer: number of documents indexed, deleted, or skipped:
2022-05-22 03:14:31,090 INFO o.a.n.i.IndexingJob [main] Indexer:      1  indexed (add/update)
2022-05-22 03:14:31,095 INFO o.a.n.i.IndexingJob [main] Indexer: finished at 2022-05-22 03:14:31, elapsed: 00:00:09
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

```
bin/nutch index crawl/crawldb/ -linkdb crawl/linkdb/ $s2 -filter -normalize
-deleteGone
```

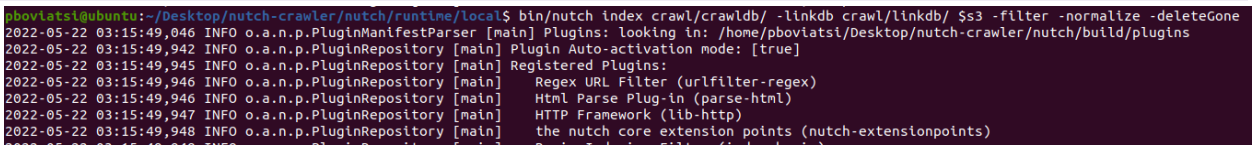


ElasticIndexWriter:

host	Comma-separated list of hostnames	localhost
port	The port to connect to elastic server.	9200
scheme	The scheme (http or https) to connect to elastic server.	http
index	Default index to send documents to.	nutch
username	Username for auth credentials	elastic
password	Password for auth credentials	
max.bulk.docs	Maximum size of the bulk in number of documents.	250
max.bulk.size	Maximum size of the bulk in bytes.	2500500
exponential.backoff.millis	Initial delay for the BulkProcessor exponential backoff policy.	100
exponential.backoff.retries	Number of times the BulkProcessor exponential backoff policy should retry bulk operations.	10
bulk.close.timeout	Number of seconds allowed for the BulkProcessor to complete its last operation.	600

```
2022-05-22 03:15:17,471 INFO o.a.n.i.a.AnchorIndexingFilter [pool-5-thread-1] Anchor deduplication is: off
2022-05-22 03:15:19,981 INFO o.a.n.i.IndexingJob [main] Indexer: number of documents indexed, deleted, or skipped:
2022-05-22 03:15:20,005 INFO o.a.n.i.IndexingJob [main] Indexer:      3  deleted (gone)
2022-05-22 03:15:20,005 INFO o.a.n.i.IndexingJob [main] Indexer:     89  indexed (add/update)
2022-05-22 03:15:20,010 INFO o.a.n.i.IndexingJob [main] Indexer: finished at 2022-05-22 03:15:20, elapsed: 00:00:17
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

```
bin/nutch index crawl/crawldb/ -linkdb crawl/linkdb/ $s3 -filter -normalize
-deleteGone
```



```
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$ bin/nutch index crawl/crawldb/ -linkdb crawl/linkdb/ $s3 -filter -normalize -deleteGone
2022-05-22 03:15:49,046 INFO o.a.n.p.PluginManifestParser [main] Plugins: looking in: /home/pboviatsi/Desktop/nutch-crawler/nutch/build/plugins
2022-05-22 03:15:49,942 INFO o.a.n.p.PluginRepository [main] Plugin Auto-activation mode: [true]
2022-05-22 03:15:49,945 INFO o.a.n.p.PluginRepository [main] Registered Plugins:
2022-05-22 03:15:49,946 INFO o.a.n.p.PluginRepository [main]    Regex URL Filter (urlfilter-regex)
2022-05-22 03:15:49,946 INFO o.a.n.p.PluginRepository [main]    Html Parse Plug-in (parse-html)
2022-05-22 03:15:49,947 INFO o.a.n.p.PluginRepository [main]    HTTP Framework (lib-http)
2022-05-22 03:15:49,948 INFO o.a.n.p.PluginRepository [main]    the nutch core extension points (nutch-extensionpoints)
```

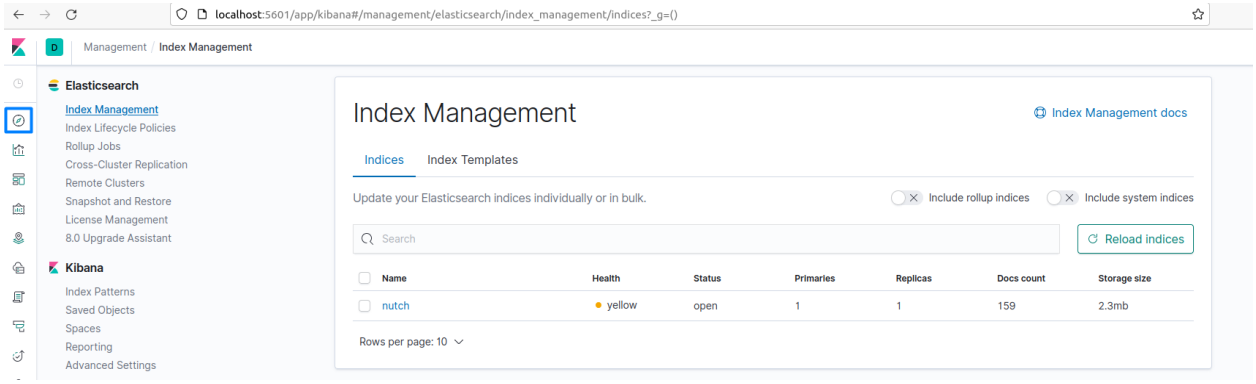
ElasticIndexWriter:

host	Comma-separated list of hostnames	localhost
port	The port to connect to elastic server.	9200
scheme	The scheme (http or https) to connect to elastic server.	http
index	Default index to send documents to.	nutch
username	Username for auth credentials	elastic
password	Password for auth credentials	
max.bulk.docs	Maximum size of the bulk in number of documents.	250
max.bulk.size	Maximum size of the bulk in bytes.	2500500
exponential.backoff.millis	Initial delay for the BulkProcessor exponential backoff policy.	100
exponential.backoff.retries	Number of times the BulkProcessor exponential backoff policy should retry bulk operations.	10
bulk.close.timeout	Number of seconds allowed for the BulkProcessor to complete its last operation.	600

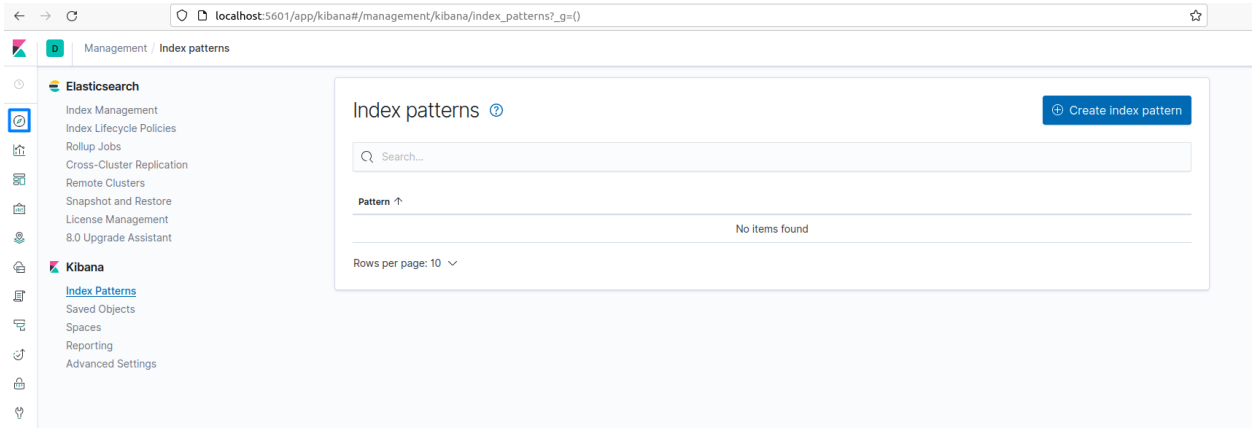
```
2022-05-22 03:16:06,478 INFO o.a.n.i.a.AnchorIndexingFilter [pool-5-thread-1] Anchor deduplication is: off
2022-05-22 03:16:09,578 INFO o.a.n.i.IndexingJob [main] Indexer: number of documents indexed, deleted, or skipped:
2022-05-22 03:16:09,608 INFO o.a.n.i.IndexingJob [main] Indexer:      2  deleted (gone)
2022-05-22 03:16:09,609 INFO o.a.n.i.IndexingJob [main] Indexer:      8  deleted (redirects)
2022-05-22 03:16:09,612 INFO o.a.n.i.IndexingJob [main] Indexer:     69  indexed (add/update)
2022-05-22 03:16:09,638 INFO o.a.n.i.IndexingJob [main] Indexer: finished at 2022-05-22 03:16:09, elapsed: 00:00:18
pboviatsi@ubuntu:~/Desktop/nutch-crawler/nutch/runtime/local$
```

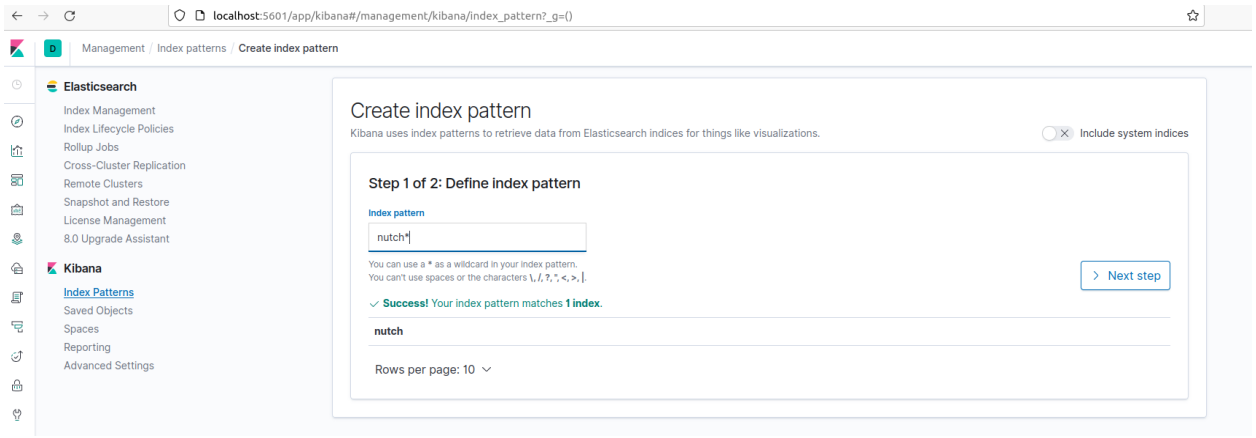
Περιβάλλον του Kibana

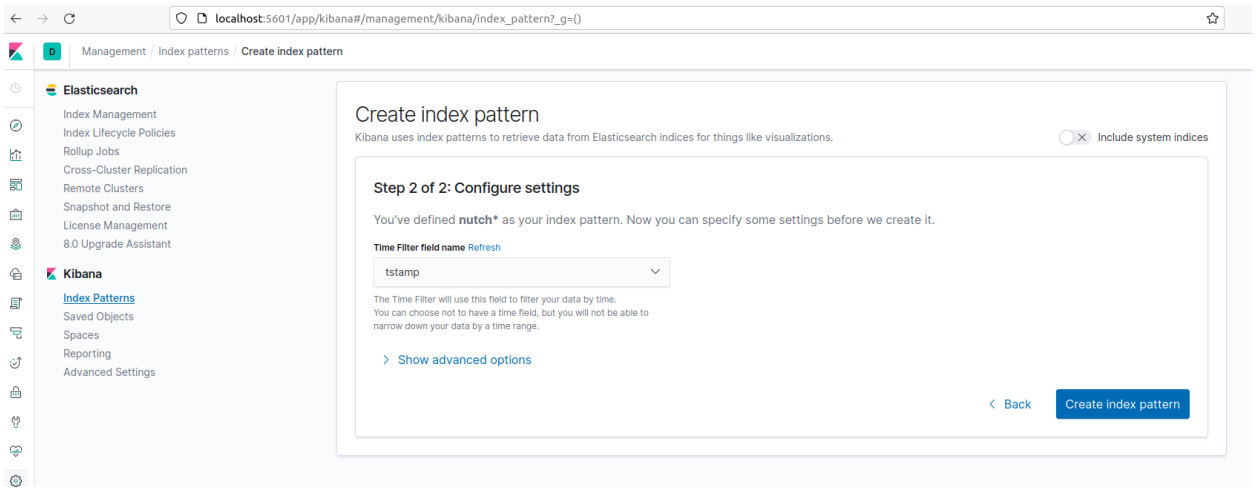
Μπαίνοντας στο http://localhost:5601 και επιλέξοντας “Elasticsearch/Index Management” φαίνονται όλοι οι index που υπάρχουν.

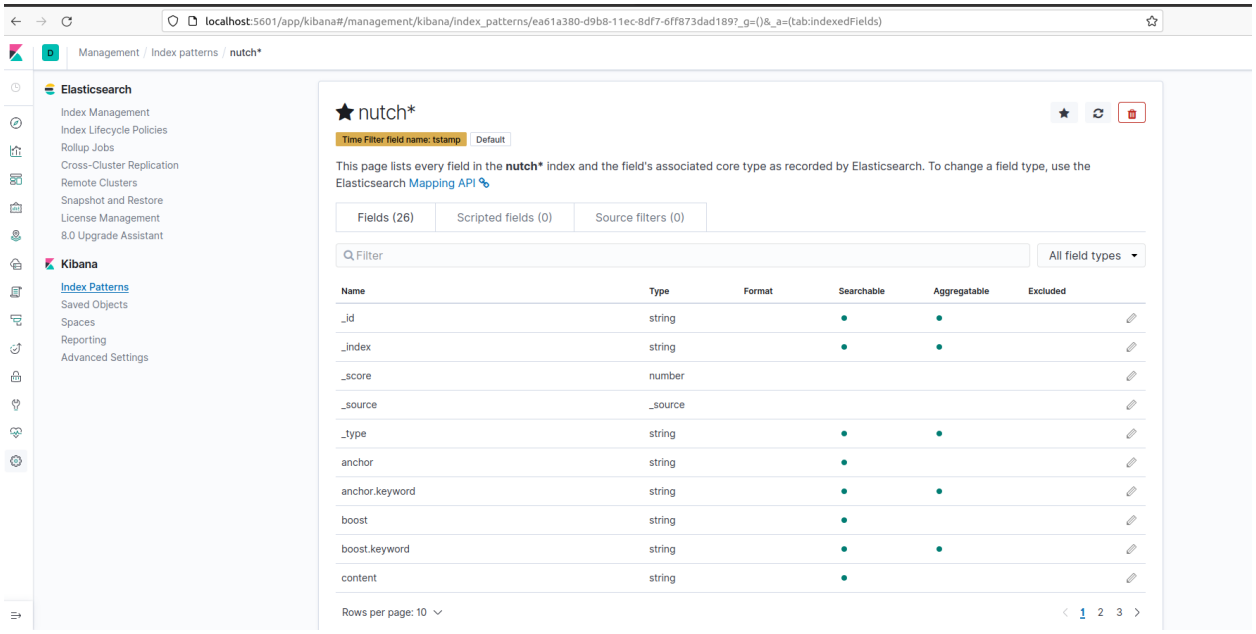


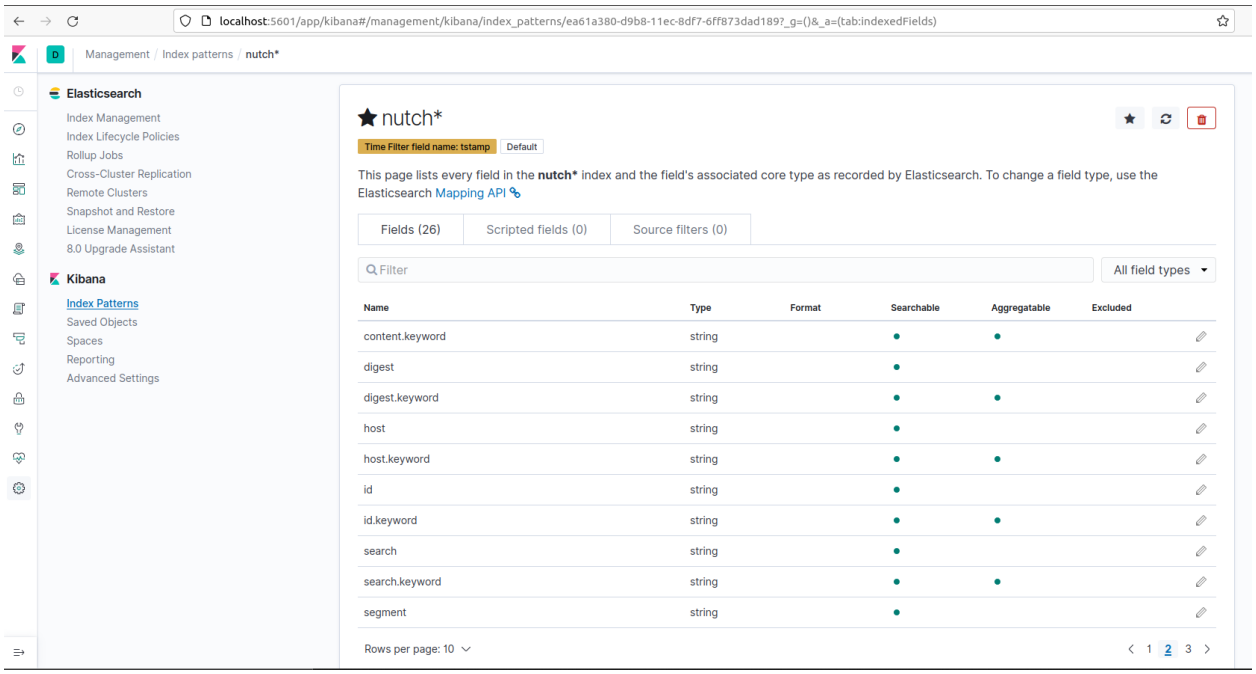
Για να γίνει η αλληλεπίδραση με τον νέο index μέσα στο kibana, πρέπει πρώτα να δημιουργηθεί ένα μοτίβο(pattern) ευρετηρίου. Επιλέγοντας το κουμπί “Create index pattern”, γίνεται μετάβαση του χρήστη στην οθόνη που χρειάζεται να γίνει εισαγωγή στοιχείων. Συμπληρώνοντας όλα τα απαραίτητα πεδία γίνεται η επιτυχής δημιουργία του index pattern.

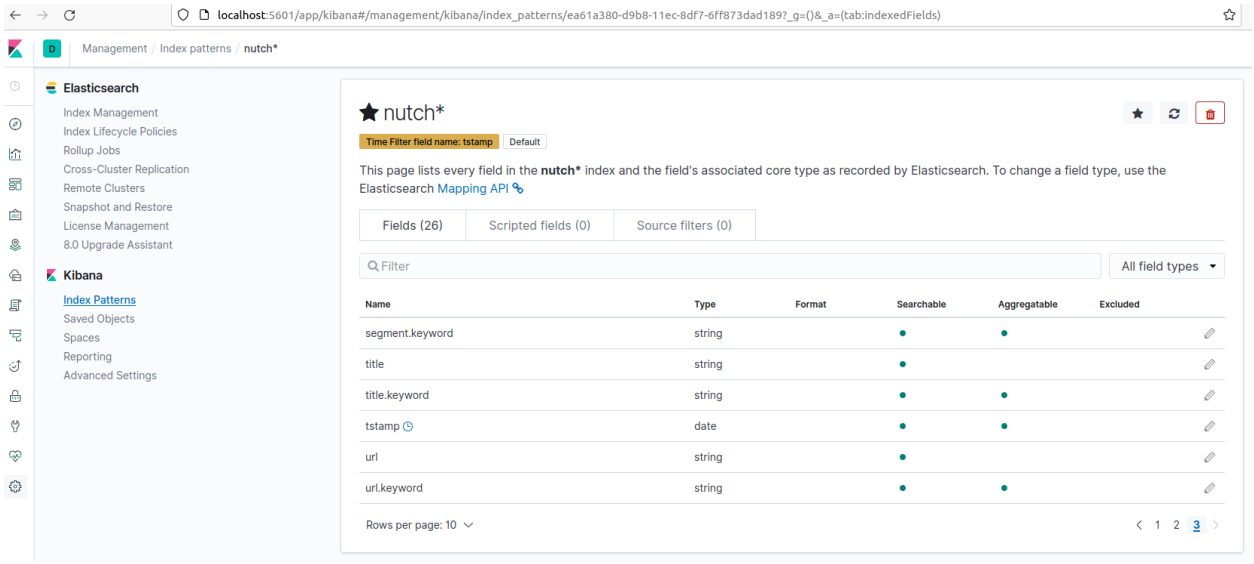

localhost:5601/app/kibana#/management/kibana/index_patterns?_g=()
Management / Index patterns
Elasticsearch
Index Management
Index Lifecycle Policies
Rollup Jobs
Cross-Cluster Replication
Remote Clusters
Snapshot and Restore
License Management
8.0 Upgrade Assistant
Kibana
Index Patterns
Saved Objects
Spaces
Reporting
Advanced Settings
Index patterns
Create index pattern
Search...
Pattern
No items found
Rows per page: 10


localhost:5601/app/kibana#/management/kibana/index_pattern?_g=()
Management / Index patterns / Create index pattern
Elasticsearch
Index Management
Index Lifecycle Policies
Rollup Jobs
Cross-Cluster Replication
Remote Clusters
Snapshot and Restore
License Management
8.0 Upgrade Assistant
Kibana
Index Patterns
Saved Objects
Spaces
Reporting
Advanced Settings
Create index pattern
Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.
Include system indices
Step 1 of 2: Define index pattern
Index pattern
nutch*
You can use a * as a wildcard in your index pattern.
You can't use spaces or the characters \, /, ?, ", <, >, |.
Next step
Success! Your index pattern matches 1 index.
nutch
Rows per page: 10

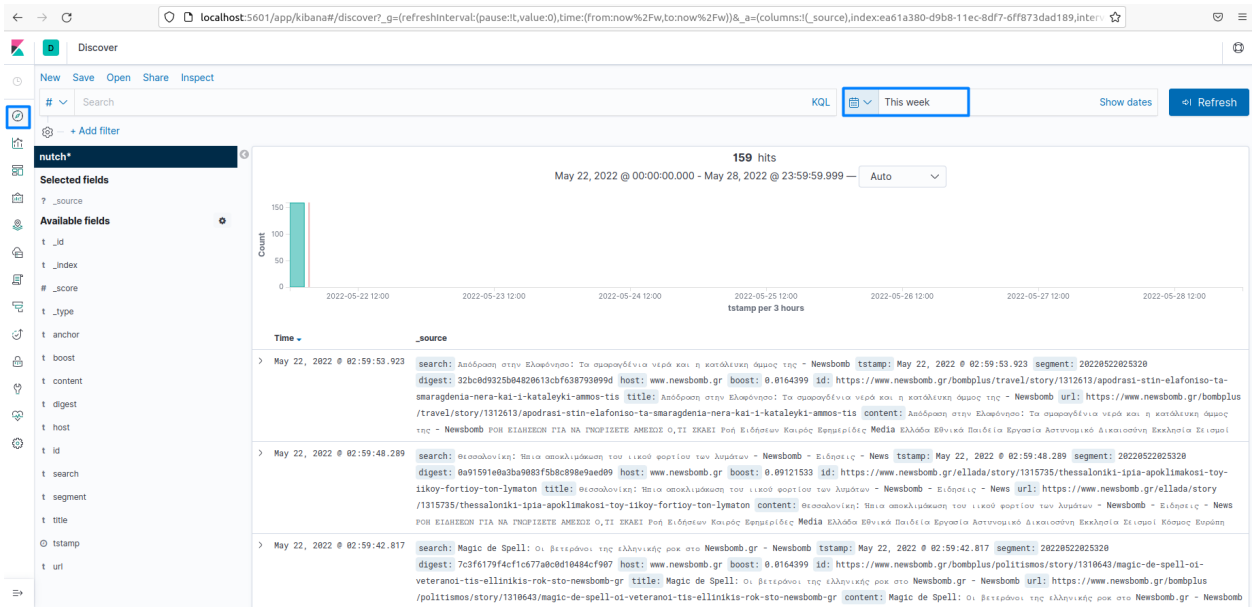

localhost:5601/app/kibana#/management/kibana/index_pattern?_g=()
Management / Index patterns / Create index pattern
Elasticsearch
Index Management
Index Lifecycle Policies
Rollup Jobs
Cross-Cluster Replication
Remote Clusters
Snapshot and Restore
License Management
8.0 Upgrade Assistant
Kibana
Index Patterns
Saved Objects
Spaces
Reporting
Advanced Settings
Create index pattern
Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.
Include system indices
Step 2 of 2: Configure settings
You've defined nutch* as your index pattern. Now you can specify some settings before we create it.
Time Filter field name Refresh
tstamp
The Time Filter will use this field to filter your data by time.
You can choose not to have a time field, but you will not be able to narrow down your data by a time range.
Show advanced options
Back
Create index pattern


localhost:5601/app/kibana#/management/kibana/index_patterns/ea61a380-d9b8-11ec-8df7-6ff873dad189?_g=()&_a=(tab:indexedFields)
Management / Index patterns / nutch*
Elasticsearch
Index Management
Index Lifecycle Policies
Rollup Jobs
Cross-Cluster Replication
Remote Clusters
Snapshot and Restore
License Management
8.0 Upgrade Assistant
Kibana
Index Patterns
Saved Objects
Spaces
Reporting
Advanced Settings
nutch*
Time Filter field name: tstamp
Default
This page lists every field in the nutch* index and the field's associated core type as recorded by Elasticsearch. To change a field type, use the Elasticsearch Mapping API
Fields (26)
Scripted fields (0)
Source filters (0)
Filter
All field types
Name
Type
Format
Searchable
Aggregatable
Excluded
_id string
_index string
_score number
_source _source
_type string
anchor string
anchor.keyword string
boost string
boost.keyword string
content string
Rows per page: 10
1 2 3


localhost:5601/app/kibana#/management/kibana/index_patterns/ea61a380-d9b8-11ec-8df7-6ff873dad189?_g=()&_a=(tab:indexedFields)
Management / Index patterns / nutch*
Elasticsearch
Index Management
Index Lifecycle Policies
Rollup Jobs
Cross-Cluster Replication
Remote Clusters
Snapshot and Restore
License Management
8.0 Upgrade Assistant
Kibana
Index Patterns
Saved Objects
Spaces
Reporting
Advanced Settings
nutch*
Time Filter field name: tstamp
Default
This page lists every field in the nutch* index and the field's associated core type as recorded by Elasticsearch. To change a field type, use the Elasticsearch Mapping API
Fields (26)
Scripted fields (0)
Source filters (0)
Filter
All field types
Name
Type
Format
Searchable
Aggregatable
Excluded
content.keyword string
digest string
digest.keyword string
host string
host.keyword string
id string
id.keyword string
search string
search.keyword string
segment string
Rows per page: 10
1 2 3



Οπτικοποίηση των δεδομένων που συλλέχθηκαν.



Παράρτημα 2

Οδηγίες εγκατάστασης Sparkler

Net Tools

```
sudo apt install net-tools
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt install net-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 36 not upgraded.
Need to get 196 kB of archives.
After this operation, 864 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 net-tools amd64 1.60+git20180626.aebd88e-1ubuntu1 [196 kB]
Fetched 196 kB in 2s (103 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 160063 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20180626.aebd88e-1ubuntu1_amd64.deb ...
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Setting up net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
pboviatsi@ubuntu:~/Desktop$
```

Java

```
sudo apt-get update
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get update
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:3 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:5 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:6 https://scala.jfrog.io/artifactory/debian all InRelease
Ign:7 https://scala.jfrog.io/artifactory/debian  InRelease
Hit:8 https://scala.jfrog.io/artifactory/debian  Release
Reading package lists... Done
pboviatsi@ubuntu:~/Desktop$
```

```
sudo apt-get install openjdk-8-jdk
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get install openjdk-8-jdk
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java
  libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev
  libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev
  openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
  default-jre libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc
  openjdk-8-demo openjdk-8-source visualvm icedtea-8-plugin
  fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
  fonts-wqy-zenhei
The following NEW packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java
  libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev
  libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-8-jdk
  openjdk-8-jdk-headless openjdk-8-jre openjdk-8-jre-headless
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
0 upgraded, 21 newly installed, 0 to remove and 104 not upgraded.
Need to get 43.5 MB of archives.
After this operation, 162 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 java-common all 0.72 [6,816 B]
```

```
Processing triggers for fontconfig (2.13.1-2ubuntu3) ...
Processing triggers for desktop-file-utils (0.24-1ubuntu3) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for ca-certificates (20210119~20.04.2) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...

done.
done.
Processing triggers for sgml-base (1.29.1) ...
Setting up x11proto-dev (2019.2-1ubuntu1) ...
Setting up libxau-dev:amd64 (1:1.0.9-0ubuntu1) ...
Setting up libice-dev:amd64 (2:1.0.10-0ubuntu1) ...
Setting up libsm-dev:amd64 (2:1.2.3-1) ...
Setting up libxdmcp-dev:amd64 (1:1.1.3-0ubuntu1) ...
Setting up x11proto-core-dev (2019.2-1ubuntu1) ...
Setting up libxcb1-dev:amd64 (1.14-2) ...
Setting up libx11-dev:amd64 (2:1.6.9-2ubuntu1.2) ...
Setting up libxt-dev:amd64 (1:1.1.5-1) ...
pboviatsi@ubuntu:~/Desktop$
```

```
java -version
```

```
pboviatsi@ubuntu:~/Desktop$ java -version
openjdk version "1.8.0_312"
OpenJDK Runtime Environment (build 1.8.0_312-8u312-b07-0ubuntu1~20.04-b07)
OpenJDK 64-Bit Server VM (build 25.312-b07, mixed mode)
pboviatsi@ubuntu:~/Desktop$
```

Docker

```
sudo apt-get install ca-certificates curl gnupg lsb-release
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get install ca-certificates curl gnupg lsb-release
Reading package lists... Done
Building dependency tree
Reading state information... Done
lsb-release is already the newest version (11.1.0ubuntu2).
lsb-release set to manually installed.
ca-certificates is already the newest version (20210119~20.04.2).
ca-certificates set to manually installed.
gnupg is already the newest version (2.2.19-3ubuntu2.1).
gnupg set to manually installed.
The following additional packages will be installed:
  libcurl4
The following NEW packages will be installed:
  curl libcurl4
0 upgraded, 2 newly installed, 0 to remove and 104 not upgraded.
Need to get 396 kB of archives.
After this operation, 1,121 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 libcurl4 amd64 7.68.0-1ubunt
u2.10 [235 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.1
0 [161 kB]
Fetched 396 kB in 2s (260 kB/s)
Selecting previously unselected package libcurl4:amd64.
(Reading database ... 156754 files and directories currently installed.)
Preparing to unpack .../libcurl4_7.68.0-1ubuntu2.10_amd64.deb ...
Unpacking libcurl4:amd64 (7.68.0-1ubuntu2.10) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.68.0-1ubuntu2.10_amd64.deb ...
Unpacking curl (7.68.0-1ubuntu2.10) ...
Setting up libcurl4:amd64 (7.68.0-1ubuntu2.10) ...
Setting up curl (7.68.0-1ubuntu2.10) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
pboviatsi@ubuntu:~/Desktop$
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg
--dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

```
pboviatsi@ubuntu:~/Desktop$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
pboviatsi@ubuntu:~/Desktop$
```

```
echo   "deb [arch=$(dpkg --print-architecture)
signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]
```

```
https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list
> /dev/null
```

```
pboviatsi@ubuntu:~/Desktop$ echo   "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyr
ings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \
>   $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
sudo apt-get update
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get update
Get:1 https://download.docker.com/linux/ubuntu focal InRelease [57.7 kB]
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease
Get:5 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [15.5 kB]
Hit:6 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Fetched 73.1 kB in 2s (38.2 kB/s)
Reading package lists... Done
```

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get install docker-ce docker-ce-cli containerd.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras docker-scan-plugin git git-man liberror-perl pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit git-doc git-el
  git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras docker-scan-plugin git git-man
  liberror-perl pigz slirp4netns
0 upgraded, 10 newly installed, 0 to remove and 104 not upgraded.
Need to get 102 MB of archives.
After this operation, 444 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 https://download.docker.com/linux/ubuntu focal/stable amd64 containerd.io amd64 1.5.11-1 [22.
```

```
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system
/containerd.service.
Setting up docker-ce-cli (5:20.10.14~3-0~ubuntu-focal) ...
Setting up pigz (2.4-1) ...
Setting up git-man (1:2.25.1-1ubuntu3.4) ...
Setting up docker-ce-rootless-extras (5:20.10.14~3-0~ubuntu-focal) ...
Setting up docker-ce (5:20.10.14~3-0~ubuntu-focal) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/doc
ker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.
socket.
Setting up git (1:2.25.1-1ubuntu3.4) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for systemd (245.4-4ubuntu3.15) ...
pboviatsi@ubuntu:~/Desktop$
```

```
sudo docker run hello-world
```

```
pboviatsi@ubuntu:~/Desktop$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:10d7d58d5ebd2a652f4d93fdd86da8f265f5318c6a73cc5b6a9798ff6d2b2e67
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

pboviatsi@ubuntu:~/Desktop$
```

Scala

```
wget www.scala-lang.org/files/archive/scala-2.13.0.deb
```

```
pboviatsi@ubuntu:~/Desktop$ wget www.scala-lang.org/files/archive/scala-2.13.0.deb
--2022-05-01 07:04:37--  http://www.scala-lang.org/files/archive/scala-2.13.0.deb
Resolving www.scala-lang.org (www.scala-lang.org)... 128.178.218.78
Connecting to www.scala-lang.org (www.scala-lang.org)|128.178.218.78|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://www.scala-lang.org/files/archive/scala-2.13.0.deb [following]
--2022-05-01 07:04:37--  https://www.scala-lang.org/files/archive/scala-2.13.0.deb
Connecting to www.scala-lang.org (www.scala-lang.org)|128.178.218.78|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 609688786 (581M) [application/x-debian-package]
Saving to: ‘scala-2.13.0.deb’

scala-2.13.0.deb                    100%[=============================================================================>] 581.44M  3.51MB/s    in 3m 0s

2022-05-01 07:07:37 (3.24 MB/s) - ‘scala-2.13.0.deb’ saved [609688786/609688786]

pboviatsi@ubuntu:~/Desktop$
```

```
sudo dpkg -i scala*.deb
```

```
pboviatsi@ubuntu:~/Desktop$ sudo dpkg -i scala*.deb
Selecting previously unselected package scala.
(Reading database ... 160112 files and directories currently installed.)
Preparing to unpack scala-2.13.0.deb ...
Unpacking scala (2.13.0-400) ...
Setting up scala (2.13.0-400) ...
Creating system group: scala
Creating system user: scala in scala with scala daemon-user and shell /bin/false
Processing triggers for man-db (2.9.1-1) ...
pboviatsi@ubuntu:~/Desktop$
```

Sbt

```
sudo apt-get update
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get update
Hit:1 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:2 https://download.docker.com/linux/ubuntu focal InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:5 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Reading package lists... Done
pboviatsi@ubuntu:~/Desktop$
```

```
sudo apt-get install apt-transport-https curl gnupg -yqq
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get install apt-transport-https curl gnupg -yqq
Selecting previously unselected package apt-transport-https.
(Reading database ... 161799 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.0.6_all.deb ...
Unpacking apt-transport-https (2.0.6) ...
Setting up apt-transport-https (2.0.6) ...
pboviatsi@ubuntu:~/Desktop$
```

```
echo "deb https://repo.scala-sbt.org/scalasbt/debian all main" | sudo tee /etc/apt/sources.list.d/sbt.list
```

```
pboviatsi@ubuntu:~/Desktop$ echo "deb https://repo.scala-sbt.org/scalasbt/debian all main" | sudo tee /etc/apt/sources.list.d/sbt.list
deb https://repo.scala-sbt.org/scalasbt/debian all main
pboviatsi@ubuntu:~/Desktop$
```

```
echo "deb https://repo.scala-sbt.org/scalasbt/debian /" | sudo tee /etc/apt/sources.list.d/sbt_old.list
```

```
pboviatsi@ubuntu:~/Desktop$ echo "deb https://repo.scala-sbt.org/scalasbt/debian /" | sudo tee /etc/apt/sources.list.d/sbt_old.list
deb https://repo.scala-sbt.org/scalasbt/debian /
pboviatsi@ubuntu:~/Desktop$
```

```
curl -sL
"https://keyserver.ubuntu.com/pks/lookup?op=get&search=0x2EE0EA64E40A89B84B
```

```
2DF73499E82A75642AC823" | sudo -H gpg --no-default-keyring --keyring gnupg-ring:/etc/apt/trusted.gpg.d/scalasbt-release.gpg --import
```

```
pboviatsi@ubuntu:~/Desktop$ curl -sL "https://keyserver.ubuntu.com/pks/lookup?op=get&search=0x2EE0EA64E40A89B84B2DF73499E82A75642AC823" | sudo -H gpg --no-default-keyring --keyring gnupg-ring:/etc/apt/trusted.gpg.d/scalasbt-release.gpg --import
gpg: keyring '/etc/apt/trusted.gpg.d/scalasbt-release.gpg' created
gpg: directory '/root/.gnupg' created
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key 99E82A75642AC823: public key "sbt build tool <scalasbt@gmail.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1
pboviatsi@ubuntu:~/Desktop$
```

```
sudo chmod 644 /etc/apt/trusted.gpg.d/scalasbt-release.gpg
```

```
pboviatsi@ubuntu:~/Desktop$ sudo chmod 644 /etc/apt/trusted.gpg.d/scalasbt-release.gpg
pboviatsi@ubuntu:~/Desktop$
```

```
sudo apt-get update
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get update
Hit:1 https://download.docker.com/linux/ubuntu focal InRelease
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:5 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:6 https://scala.jfrog.io/artifactory/debian all InRelease [3,558 B]
Ign:7 https://scala.jfrog.io/artifactory/debian  InRelease
Get:9 https://scala.jfrog.io/artifactory/debian all/main amd64 Packages [1,423 B]
Get:8 https://scala.jfrog.io/artifactory/debian  Release [815 B]
Get:10 https://scala.jfrog.io/artifactory/debian all/main i386 Packages [1,423 B]
Get:11 https://scala.jfrog.io/artifactory/debian  Release.gpg [821 B]
Get:12 https://scala.jfrog.io/artifactory/debian  Packages [5,122 B]
Fetched 13.2 kB in 14s (961 B/s)
Reading package lists... Done
pboviatsi@ubuntu:~/Desktop$
```

```
sudo apt-get install sbt
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt-get install sbt
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  sbt
0 upgraded, 1 newly installed, 0 to remove and 104 not upgraded.
Need to get 19.9 kB of archives.
After this operation, 49.2 kB of additional disk space will be used.
Get:1 https://scala.jfrog.io/artifactory/debian all/main amd64 sbt all 1.6.2 [19.9 kB]
Fetched 19.9 kB in 12s (1,666 B/s)
Selecting previously unselected package sbt.
(Reading database ... 161803 files and directories currently installed.)
Preparing to unpack .../apt/archives/sbt_1.6.2_all.deb ...
Unpacking sbt (1.6.2) ...
Setting up sbt (1.6.2) ...
Creating system group: sbt
Creating system user: sbt in sbt with sbt daemon-user and shell /bin/false
Processing triggers for man-db (2.9.1-1) ...
pboviatsi@ubuntu:~/Desktop$
```

Apache Maven

```
sudo apt install maven
```

```
pboviatsi@ubuntu:~/Desktop$ sudo apt install maven
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libaopalliance-java libapache-pom-java libatinject-jsr330-api-java libcdi-api-java
  libcommons-cli-java libcommons-io-java libcommons-lang3-java libcommons-parent-java
  libgeronimo-annotation-1.3-spec-java libgeronimo-interceptor-3.0-spec-java libguava-java
  libguice-java libhawtjni-runtime-java libjansi-java libjansi-native-java libjsr305-java
  libmaven-parent-java libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java
  libplexus-cipher-java libplexus-classworlds-java libplexus-component-annotations-java
  libplexus-interpolation-java libplexus-sec-dispatcher-java libplexus-utils2-java
  libsisu-inject-java libsisu-plexus-java libslf4j-java libwagon-file-java
  libwagon-http-shaded-java libwagon-provider-api-java
Suggested packages:
  libaopalliance-java-doc libatinject-jsr330-api-java-doc libservlet3.1-java
  libcommons-io-java-doc libcommons-lang3-java-doc libasm-java libcglib-java libjsr305-java-doc
  libmaven-shared-utils-java-doc liblogback-java libplexus-cipher-java-doc
  libplexus-classworlds-java-doc libplexus-sec-dispatcher-java-doc libplexus-utils2-java-doc
  junit4 testng libcommons-logging-java liblog4j1.2-java
The following NEW packages will be installed:
```

```
Setting up libwagon-file-java (3.3.4-1) ...
Setting up libcommons-io-java (2.6-2ubuntu0.20.04.1) ...
Setting up libguice-java (4.2.1-1) ...
Setting up libjansi-java (1.18-1) ...
Setting up libmaven-shared-utils-java (3.3.0-1) ...
Setting up libsisu-inject-java (0.3.3-1) ...
Setting up libsisu-plexus-java (0.3.3-3) ...
Setting up libmaven3-core-java (3.6.3-1) ...
Setting up maven (3.6.3-1) ...
update-alternatives: using /usr/share/maven/bin/mvn to provide /usr/bin/mvn (mvn) in auto mode
pboviatsi@ubuntu:~/Desktop$
```

Git Clone

#Clone and build Sparkler

```
git clone https://github.com/USCDataScience/sparkler
```

```
pboviatsi@ubuntu:~/Desktop$ git clone https://github.com/USCDataScience/sparkler
Cloning into 'sparkler'...
remote: Enumerating objects: 13840, done.
remote: Counting objects: 100% (4866/4866), done.
remote: Compressing objects: 100% (1275/1275), done.
remote: Total 13840 (delta 2109), reused 4783 (delta 2067), pack-reused 8974
Receiving objects: 100% (13840/13840), 22.45 MiB | 5.50 MiB/s, done.
Resolving deltas: 100% (5509/5509), done.
pboviatsi@ubuntu:~/Desktop$
```

Build και τροποποιήσεις repository

```
cd sparkler/
sbt package
```

```
pboviatsi@ubuntu:~/Desktop$ cd sparkler/
pboviatsi@ubuntu:~/Desktop/sparkler$ sbt package
downloading sbt launcher 1.6.2
[info] [launcher] getting org.scala-sbt sbt 1.5.0  (this may take some time)...
[info] [launcher] getting Scala 2.12.13 (for sbt)...
[info] welcome to sbt 1.5.0 (Private Build Java 1.8.0_312)
[info] loading settings for project sparkler-build-build from metals.sbt ...
[info] loading project definition from /home/pboviatsi/Desktop/sparkler/project/project
[info] Updating
https://repo1.maven.org/maven2/ch/epfl/scala/sbt-bloop_2.12_1.0/1.4.10-8-8d1cbc4f/sbt-bloop-1.4.10-…
  100.0% [##########] 2.8 KiB (5.7 KiB / s)
https://repo1.maven.org/maven2/net/java/dev/jna/jna-platform/4.5.0/jna-platform-4.5.0.pom
  100.0% [##########] 1.8 KiB (9.8 KiB / s)
https://repo1.maven.org/maven2/com/google/code/findbugs/jsr305/3.0.2/jsr305-3.0.2.pom
  100.0% [##########] 4.2 KiB (11.8 KiB / s)
https://repo1.maven.org/maven2/com/google/code/gson/gson/2.7/gson-2.7.pom
  100.0% [##########] 1.4 KiB (3.5 KiB / s)
https://repo1.maven.org/maven2/net/java/dev/jna/jna/4.5.0/jna-4.5.0.pom
  100.0% [##########] 1.5 KiB (4.0 KiB / s)
https://repo1.maven.org/maven2/com/google/code/gson/gson-parent/2.7/gson-parent-2.7.pom
  100.0% [##########] 3.5 KiB (29.9 KiB / s)
[info] Resolved  dependencies
```

```
[info] Strategy 'deduplicate' was applied to 2 files (Run the task at debug level to see details)
[info] Strategy 'discard' was applied to 9604 files (Run the task at debug level to see details)
[info] Strategy 'filterDistinctLines' was applied to 13 files (Run the task at debug level to see details)
[info] Strategy 'first' was applied to 115 files (Run the task at debug level to see details)
[info] Strategy 'rename' was applied to 17 files (Run the task at debug level to see details)
[warn] Ignored unknown package option FixedTimestamp(Some(1262304000000))
[warn] four warnings found
[info] Main Scala API documentation successful.
[warn] 69 warnings found
[success] All package validations passed
[success] All package validations passed
[success] Total time: 287 s (04:47), completed May 1, 2022 7:29:26 AM
pboviatsi@ubuntu:~/Desktop/sparkler$
```

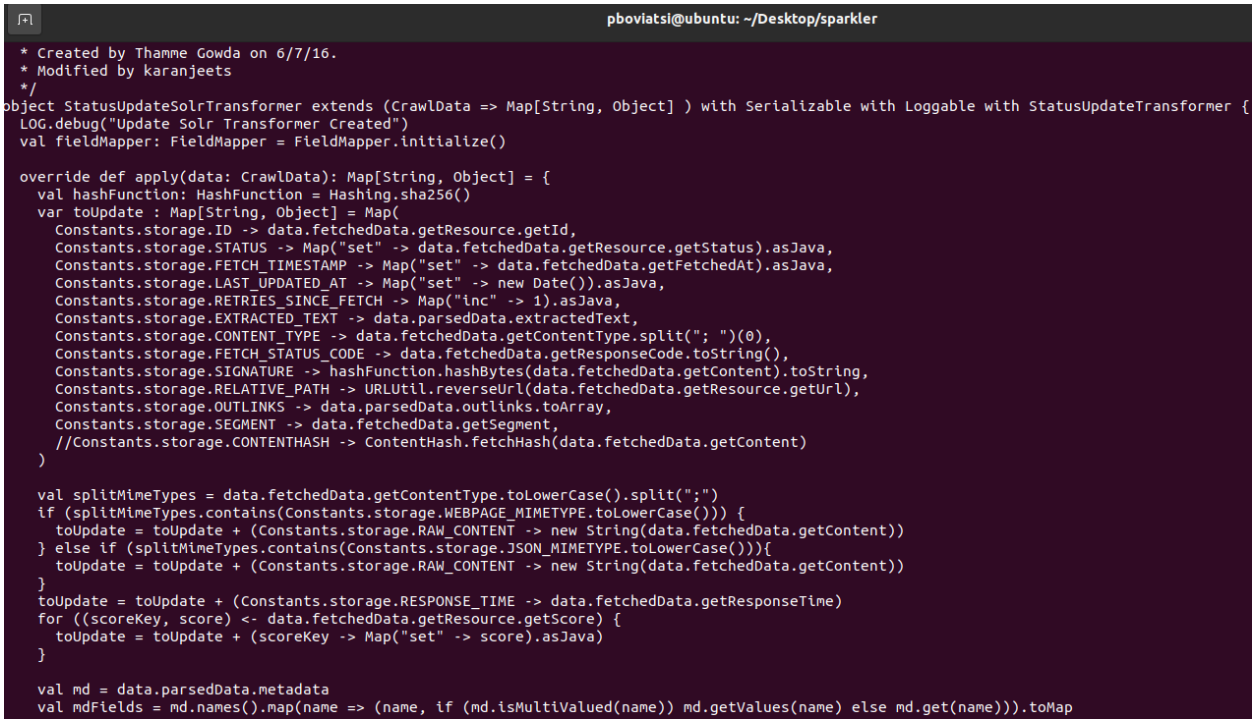
Αφαίρεση του ‘test’ απο το αρχείο ./release.sh

```
vi ./release.sh
```

```
pboviatsi@ubuntu: ~/Desktop/sparkler
#!/bin/bash
# Licensed to the Apache Software Foundation (ASF) under one or more
# contributor license agreements.  See the NOTICE file distributed with
# this work for additional information regarding copyright ownership.
# The ASF licenses this file to You under the Apache License, Version 2.0
# (the "License"); you may not use this file except in compliance with
# the License.  You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

# Script     : release.sh
# Usage      : ./release.sh
# Description: Release Sparkler Silently - Create tag with version in version.sbt and bump it

sbt clean package && sbt releaseSilent
~
```

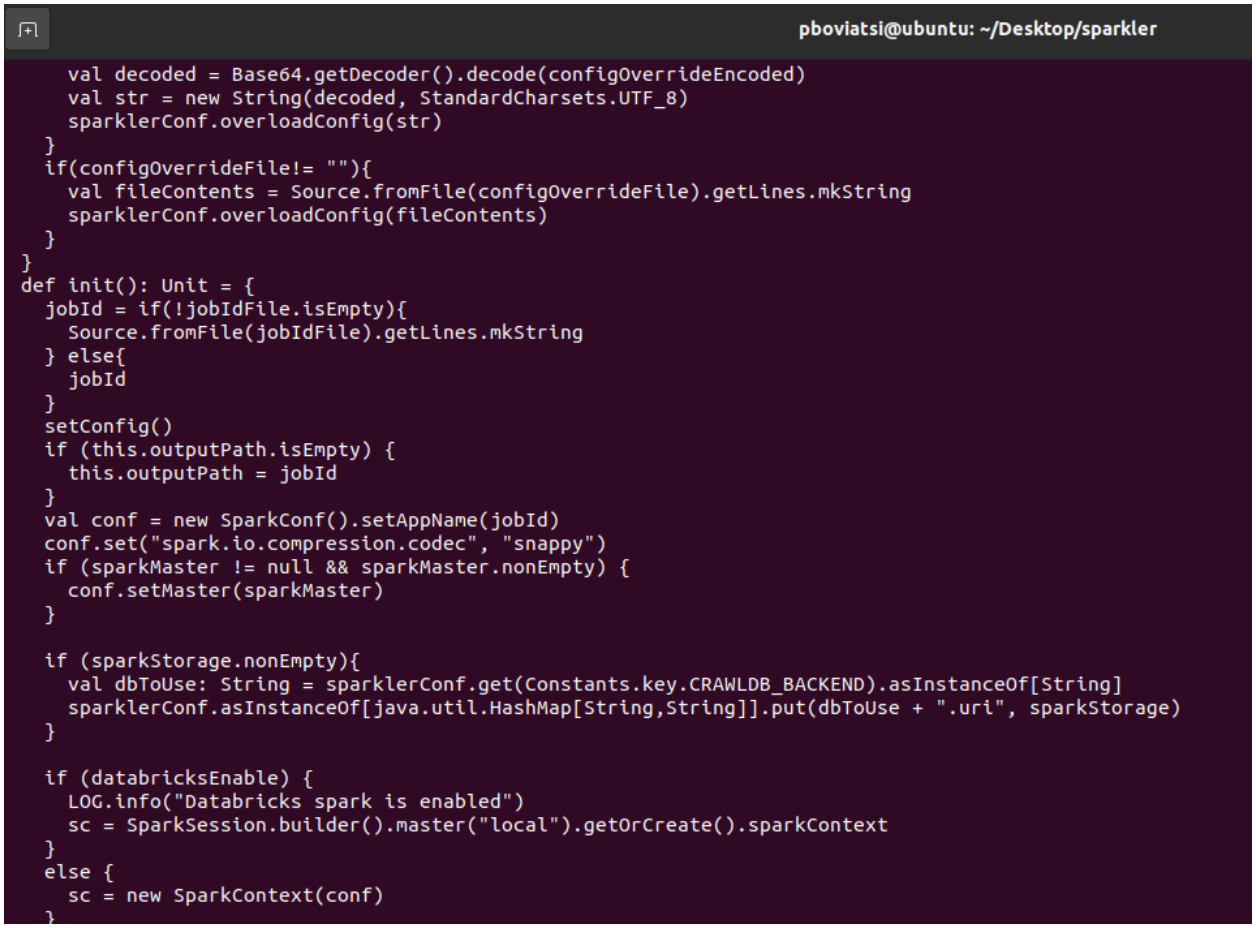
Αφαίρεση ή σχολιασμός της γραμμής που περιέχει το εξής περιεχόμενο `Constants.storage.CONTENTHASH -> ContentHash.fetchHash(data.fetchedData.getContent)` απο το αρχείο ./sparkler-app/src/main/scala/edu/usc/irds/sparkler/storage/solr/StatusUpdateSolrTransformer.scala

```
vi ./sparkler-app/src/main/scala/edu/usc/irds/sparkler/storage/solr/StatusUpdateSolrTransformer.scala
```

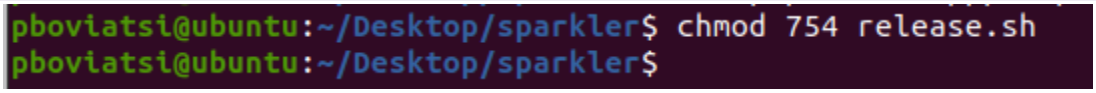


Προσθήκη της εξής γραμμής `conf.set("spark.io.compression.codec", "snappy")` στο αρχείο ./sparkler-app/src/main/scala/edu/usc/irds/sparkler/pipeline/Crawler.scala και σειρά 171.

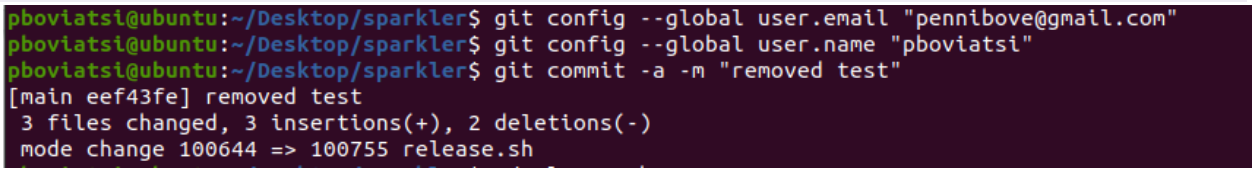
```
vi ./sparkler-app/src/main/scala/edu/usc/irds/sparkler/pipeline/Crawler.scala
```



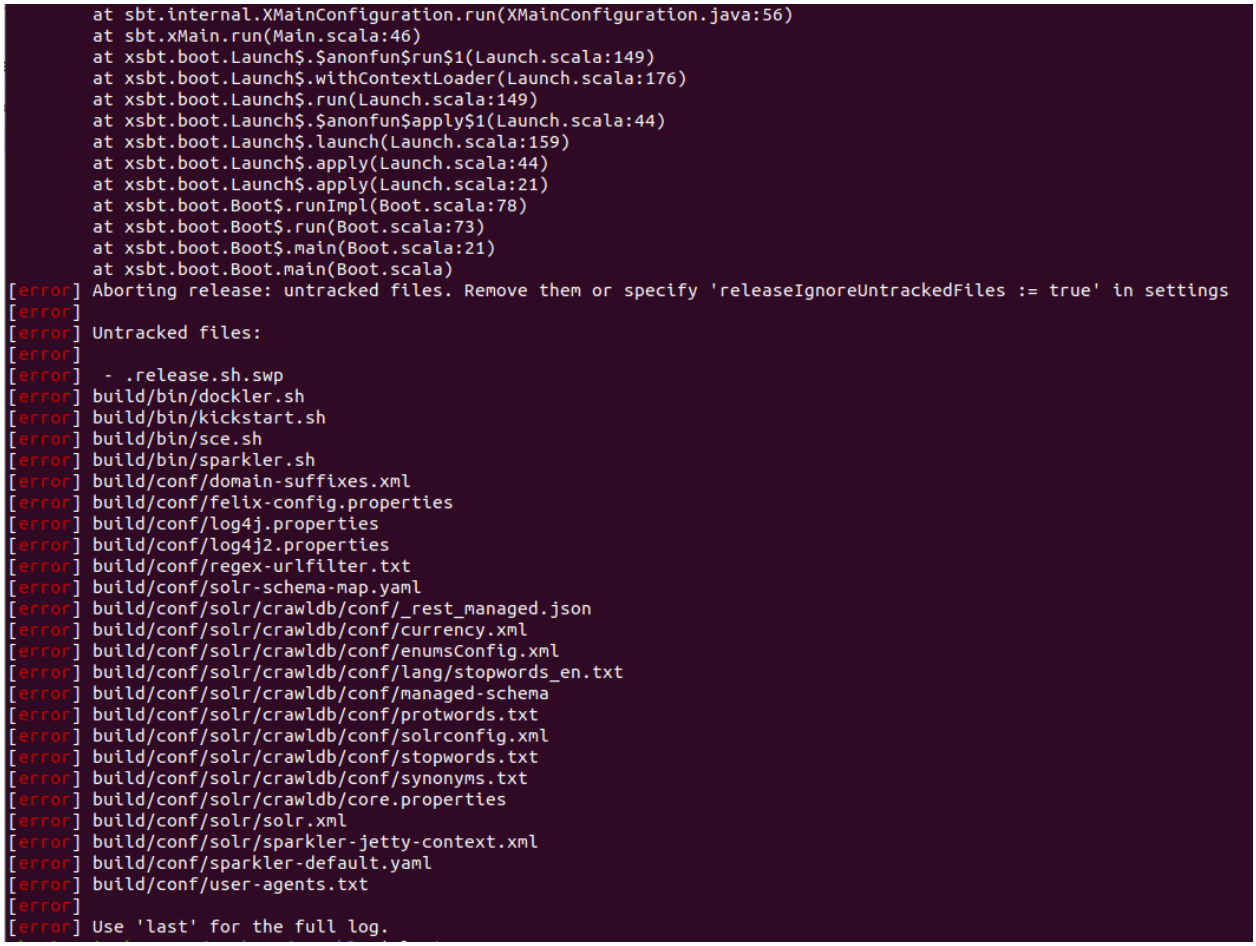

```
chmod 754 release.sh
```




```
git commit -a -m "removed test"
```




```
./release.sh
```



Προσπαθώντα να λυθεί το error προστέθηκε το releaseIgnoreUntrackedFiles := true στο build.sbt αρχείο. Στην συνέχεια εμφανίστηκε άλλο error.

```
java.lang.RuntimeException: Aborting release due to snapshot dependencies.
        at scala.sys.package$.error(package.scala:30)
        at sbtrelease.ReleaseStateTransformations$.$anonfun$checkSnapshotDependencies$1(ReleaseExtra.scala:27)
        at sbtrelease.ReleasePlugin$autoImport$ReleaseKeys$.filterFailure$1(ReleasePlugin.scala:178)
        at sbtrelease.ReleasePlugin$autoImport$ReleaseKeys$.$anonfun$releaseCommand$9(ReleasePlugin.scala:196)
        at scala.Function$.$anonfun$chain$2(Function.scala:26)
        at scala.collection.LinearSeqOptimized.foldLeft(LinearSeqOptimized.scala:126)
        at scala.collection.LinearSeqOptimized.foldLeft$(LinearSeqOptimized.scala:122)
        at scala.collection.immutable.List.foldLeft(List.scala:91)
        at scala.collection.TraversableOnce.$div$colon(TraversableOnce.scala:187)
        at scala.collection.TraversableOnce.$div$colon$(TraversableOnce.scala:187)
        at scala.collection.AbstractTraversable.$div$colon(Traversable.scala:108)
        at scala.Function$.$anonfun$chain$1(Function.scala:26)
        at sbtrelease.ReleasePlugin$autoImport$ReleaseKeys$.$anonfun$releaseCommand$2(ReleasePlugin.scala:202)
        at sbt.Command$.$anonfun$applyEffect$4(Command.scala:150)
        at sbt.Command$.$anonfun$applyEffect$2(Command.scala:145)
        at sbt.Command$.process(Command.scala:189)
        at sbt.MainLoop$.$anonfun$processCommand$5(MainLoop.scala:245)
        at scala.Option.getOrElse(Option.scala:189)
        at sbt.MainLoop$.process$1(MainLoop.scala:245)
        at sbt.MainLoop$.processCommand(MainLoop.scala:276)
        at sbt.MainLoop$.$anonfun$next$5(MainLoop.scala:163)
        at sbt.State$StateOpsImpl$.runCmd$1(State.scala:289)
        at sbt.State$StateOpsImpl$.process$extension(State.scala:325)
        at sbt.MainLoop$.$anonfun$next$4(MainLoop.scala:163)
        at sbt.internal.util.ErrorHandling$.wideConvert(ErrorHandling.scala:23)
        at sbt.MainLoop$.next(MainLoop.scala:163)
        at sbt.MainLoop$.run(MainLoop.scala:144)
        at sbt.MainLoop$.$anonfun$runWithNewLog$1(MainLoop.scala:119)
        at sbt.io.Using.apply(Using.scala:27)
        at sbt.MainLoop$.runWithNewLog(MainLoop.scala:112)
        at sbt.MainLoop$.runAndClearLast(MainLoop.scala:66)
        at sbt.MainLoop$.runLoggedLoop(MainLoop.scala:51)
        at sbt.MainLoop$.runLogged(MainLoop.scala:42)
        at sbt.StandardMain$.runManaged(Main.scala:192)
        at sbt.xMain$.$anonfun$run$8(Main.scala:101)
        at scala.util.DynamicVariable.withValue(DynamicVariable.scala:62)
        at scala.Console$.withIn(Console.scala:230)
        at sbt.internal.util.Terminal$.withIn(Terminal.scala:560)
        at sbt.internal.util.Terminal$.$anonfun$withStreams$1(Terminal.scala:350)
        at scala.util.DynamicVariable.withValue(DynamicVariable.scala:62)
        at scala.Console$.withOut(Console.scala:167)
```

```
        at xsbt.boot.Boot$.main(Boot.scala:21)
        at xsbt.boot.Boot.main(Boot.scala)
[error] Aborting release due to snapshot dependencies.
[error] Use 'last' for the full log.
pboviatsi@ubuntu:~/Desktop/sparkler$
```

ls ./build/sparkler-app-0.5.26-SNAPSHOT/lib/com.kythera.sparkler-app-0.5.26-SNAPSHOT.jar

```
pboviatsi@ubuntu:~/Desktop/sparkler$ ls ./build/sparkler-app-0.5.26-SNAPSHOT/lib/com.kythera.sparkler-app-0.5.26-SNAPSHOT.jar
./build/sparkler-app-0.5.26-SNAPSHOT/lib/com.kythera.sparkler-app-0.5.26-SNAPSHOT.jar
pboviatsi@ubuntu:~/Desktop/sparkler$
```

Spark

```
cd
wget
https://downloads.apache.org/spark/spark-3.0.3/spark-3.0.3-bin-hadoop2.7.tg
z
```

```
pboviatsi@ubuntu:~$ wget https://downloads.apache.org/spark/spark-3.0.3/spark-3.0.3-bin-hadoop2.7.tgz
--2022-05-01 09:22:31--  https://downloads.apache.org/spark/spark-3.0.3/spark-3.0.3-bin-hadoop2.7.tgz
Resolving downloads.apache.org (downloads.apache.org)... 88.99.95.219, 135.181.214.104, 2a01:4f8:10a:201a::2, ...
Connecting to downloads.apache.org (downloads.apache.org)|88.99.95.219|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 220400553 (210M) [application/x-gzip]
Saving to: ‘spark-3.0.3-bin-hadoop2.7.tgz’

spark-3.0.3-bin-hadoop2.7.tgz            100%[=====================================================================================>] 210.19M  1.45MB/s    in 58s

2022-05-01 09:23:29 (3.65 MB/s) - ‘spark-3.0.3-bin-hadoop2.7.tgz’ saved [220400553/220400553]

pboviatsi@ubuntu:~$
```

```
tar xvf spark-*
```

```
pboviatsi@ubuntu:~$ tar xvf spark-*
spark-3.0.3-bin-hadoop2.7/
spark-3.0.3-bin-hadoop2.7/NOTICE
spark-3.0.3-bin-hadoop2.7/kubernetes/
spark-3.0.3-bin-hadoop2.7/kubernetes/tests/
spark-3.0.3-bin-hadoop2.7/kubernetes/tests/worker_memory_check.py
spark-3.0.3-bin-hadoop2.7/kubernetes/tests/py_container_checks.py
spark-3.0.3-bin-hadoop2.7/kubernetes/tests/pyfiles.py
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/entrypoint.sh
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/bindings/
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/bindings/R/
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/bindings/R/Dockerfile
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/bindings/python/
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/bindings/python/Dockerfile
spark-3.0.3-bin-hadoop2.7/kubernetes/dockerfiles/spark/Dockerfile
spark-3.0.3-bin-hadoop2.7/jars/
spark-3.0.3-bin-hadoop2.7/jars/jackson-xc-1.9.13.jar
spark-3.0.3-bin-hadoop2.7/jars/commons-digester-1.8.jar
spark-3.0.3-bin-hadoop2.7/jars/api-util-1.0.0-M20.jar
spark-3.0.3-bin-hadoop2.7/jars/hive-vector-code-gen-2.3.7.jar
spark-3.0.3-bin-hadoop2.7/jars/derby-10.12.1.1.jar
spark-3.0.3-bin-hadoop2.7/jars/commons-beanutils-1.9.4.jar
spark-3.0.3-bin-hadoop2.7/jars/httpcore-4.4.12.jar
spark-3.0.3-bin-hadoop2.7/jars/hadoop-yarn-api-2.7.4.jar
spark-3.0.3-bin-hadoop2.7/jars/scala-library-2.12.10.jar
spark-3.0.3-bin-hadoop2.7/jars/parquet-format-2.4.0.jar
spark-3.0.3-bin-hadoop2.7/jars/kryo-shaded-4.0.2.jar
spark-3.0.3-bin-hadoop2.7/jars/xercesImpl-2.12.0.jar
spark-3.0.3-bin-hadoop2.7/jars/commons-logging-1.1.3.jar
spark-3.0.3-bin-hadoop2.7/jars/okio-1.15.0.jar
spark-3.0.3-bin-hadoop2.7/jars/commons-compiler-3.0.16.jar
spark-3.0.3-bin-hadoop2.7/jars/jdo-api-3.0.1.jar
spark-3.0.3-bin-hadoop2.7/jars/spire-macros_2.12-0.17.0-M1.jar
spark-3.0.3-bin-hadoop2.7/jars/arrow-memory-0.15.1.jar
spark-3.0.3-bin-hadoop2.7/jars/JLargeArrays-1.5.jar
spark-3.0.3-bin-hadoop2.7/jars/jsp-api-2.1.jar
spark-3.0.3-bin-hadoop2.7/jars/logging-interceptor-3.12.6.jar
spark-3.0.3-bin-hadoop2.7/jars/javax.servlet-api-3.1.0.jar
spark-3.0.3-bin-hadoop2.7/jars/jcl-over-slf4j-1.7.30.jar
spark-3.0.3-bin-hadoop2.7/jars/hive-cli-2.3.7.jar
spark-3.0.3-bin-hadoop2.7/jars/apacheds-i18n-2.0.0-M15.jar
```

```
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-modernizr.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-spire.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-leveldbjni.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-join.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-zstd-jni.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-slf4j.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-arpack.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-jsp-api.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-JTransforms.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-JLargeArrays.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-bootstrap.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-reflectasm.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-javassist.html
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-zstd.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-json-formatter.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-matchMedia-polyfill.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-scala.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-jakarta.activation-api.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-automaton.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-javax-transaction-transaction-api.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-jaxb-runtime.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-minlog.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-mustache.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-xmlenc.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-jline.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-istack-commons-runtime.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-py4j.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-vis-timeline.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-re2j.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-kryo.txt
spark-3.0.3-bin-hadoop2.7/licenses/LICENSE-cloudpickle.txt
pboviatsi@ubuntu:~$
```

```
sudo mv spark-3.0.3-bin-hadoop2.7 /opt/spark
```

```
pboviatsi@ubuntu:~$ sudo mv spark-3.0.3-bin-hadoop2.7 /opt/spark
[sudo] password for pboviatsi:
pboviatsi@ubuntu:~$
```

Προσθήκη του εξής κώδικα

```
export SPARK_HOME=/opt/spark
export PATH=$PATH:$SPARK_HOME/bin:$SPARK_HOME/sbin
export PYSPARK_PYTHON=/usr/bin/python3
```

στο /etc/environment αρχείο.

```
sudo vi /etc/environment
```


```
source /etc/environment
start-master.sh
```

```
pboviatsi@ubuntu:~$ source /etc/environment
pboviatsi@ubuntu:~$ start-master.sh
starting org.apache.spark.deploy.master.Master, logging to /opt/spark/logs/spark-pboviatsi-org.apache.spark.deploy.master.Master-1-ubuntu.out
pboviatsi@ubuntu:~$
```

```
start-slave.sh spark://localhost:7077
```

```
pboviatsi@ubuntu:~$ start-slave.sh spark://localhost:7077
starting org.apache.spark.deploy.worker.Worker, logging to /opt/spark/logs/spark-pboviatsi-org.apache.spark.deploy.worker.Worker-1-ubuntu.out
pboviatsi@ubuntu:~$
```

```
spark-shell
```

```
pboviatsi@ubuntu:~$ spark-shell
22/05/01 09:32:07 WARN Utils: Your hostname, ubuntu resolves to a loopback address: 127.0.1.1; using 192.168.88.129 instead (on interface ens33)
22/05/01 09:32:07 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
22/05/01 09:32:08 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
Spark context Web UI available at http://192.168.88.129:4040
Spark context available as 'sc' (master = local[*], app id = local-1651422743134).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.0.3
      /_/

Using Scala version 2.12.10 (OpenJDK 64-Bit Server VM, Java 1.8.0_312)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

Δημιουγούμε εικονικό data frame για να δούμε οτι λειτουργεί σωστά το spark.

```
val df = spark.sql("select 1 i, 2 j, 3 k")
```

```
scala> val df = spark.sql("select 1 i, 2 j, 3 k")
df: org.apache.spark.sql.DataFrame = [i: int, j: int ... 1 more field]

scala>
```

```
df.show()
```

```
scala> df.show()
+---+---+---+
|  i|  j|  k|
+---+---+---+
|  1|  2|  3|
+---+---+---+


scala>
```

Elasticsearch

```
curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
```

```
pboviatsi@ubuntu:~$ curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
OK
pboviatsi@ubuntu:~$
```

```
echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list
```

```
pboviatsi@ubuntu:~$ echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list
deb https://artifacts.elastic.co/packages/7.x/apt stable main
pboviatsi@ubuntu:~$
```

```
sudo apt update
```

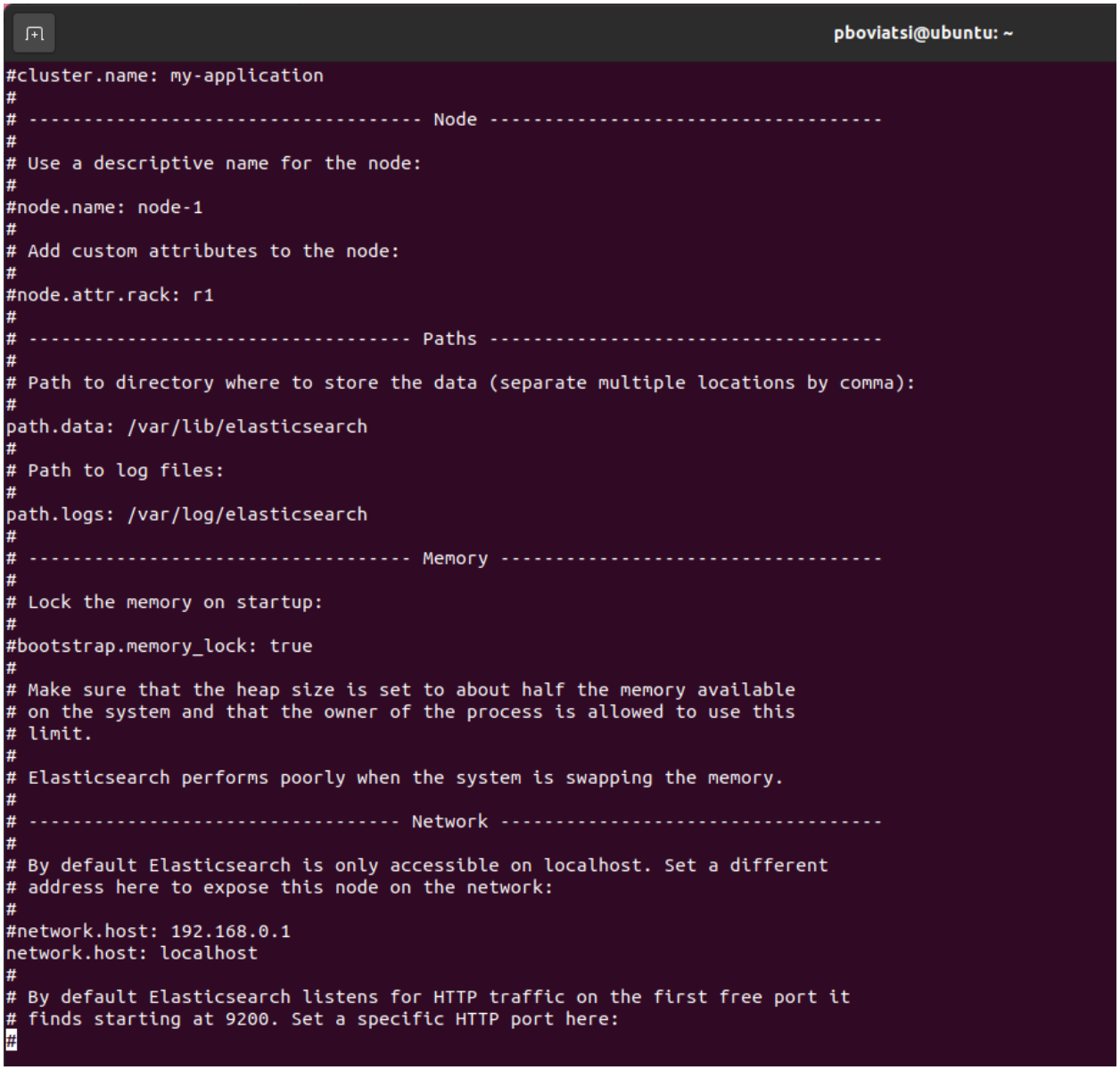
```
pboviatsi@ubuntu:~$ sudo apt update
Hit:1 https://download.docker.com/linux/ubuntu focal InRelease
Get:2 https://artifacts.elastic.co/packages/7.x/apt stable InRelease [13.7 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:4 http://us.archive.ubuntu.com/ubuntu focal InRelease
Get:5 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:8 https://artifacts.elastic.co/packages/7.x/apt stable/main i386 Packages [73.8 kB]
Get:9 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:10 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 Packages [98.8 kB]
Hit:6 https://scala.jfrog.io/artifactory/debian all InRelease
Ign:7 https://scala.jfrog.io/artifactory/debian  InRelease
Hit:11 https://scala.jfrog.io/artifactory/debian  Release
Fetched 522 kB in 13s (41.4 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
104 packages can be upgraded. Run 'apt list --upgradable' to see them.
pboviatsi@ubuntu:~$
```

```
sudo apt install elasticsearch
```

```
pboviatsi@ubuntu:~$ sudo apt install elasticsearch
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  elasticsearch
0 upgraded, 1 newly installed, 0 to remove and 104 not upgraded.
Need to get 312 MB of archives.
After this operation, 518 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 elasticsearch amd64 7.17.3 [31
2 MB]
Fetched 312 MB in 52s (5,954 kB/s)
Selecting previously unselected package elasticsearch.
(Reading database ... 162753 files and directories currently installed.)
Preparing to unpack .../elasticsearch_7.17.3_amd64.deb ...
Creating elasticsearch group... OK
Creating elasticsearch user... OK
Unpacking elasticsearch (7.17.3) ...
Setting up elasticsearch (7.17.3) ...
### NOT starting on installation, please execute the following statements to configure elasticsearch
 service to start automatically using systemd
 sudo systemctl daemon-reload
 sudo systemctl enable elasticsearch.service
### You can start elasticsearch service by executing
 sudo systemctl start elasticsearch.service
Created elasticsearch keystore in /etc/elasticsearch/elasticsearch.keystore
Processing triggers for systemd (245.4-4ubuntu3.15) ...
pboviatsi@ubuntu:~$
```

Δηλωση του network.host στο /etc/elasticsearch/elasticsearch.yml αρχείο.

```
sudo vi /etc/elasticsearch/elasticsearch.yml
```



Εκκίνηση και ενεργοποίηση του elasticsearch.
Επίσης εκτυπώνουμε την κατασταση του (status) για να δούμε αν έγινε επιτυχώς η ενεργοποίηση.

```
sudo systemctl start elasticsearch
sudo systemctl enable elasticsearch
sudo systemctl status elasticsearch
```

```
pboviatsi@ubuntu:~$ sudo systemctl start elasticsearch
pboviatsi@ubuntu:~$ sudo systemctl enable elasticsearch
Synchronizing state of elasticsearch.service with SysV service script with /lib/systemd/systemd-sysv
-install.
Executing: /lib/systemd/systemd-sysv-install enable elasticsearch
Created symlink /etc/systemd/system/multi-user.target.wants/elasticsearch.service → /lib/systemd/sys
tem/elasticsearch.service.
pboviatsi@ubuntu:~$ sudo systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
     Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2022-05-01 23:58:15 PDT; 26s ago
       Docs: https://www.elastic.co
   Main PID: 5844 (java)
      Tasks: 84 (limit: 6886)
     Memory: 3.0G
     CGroup: /system.slice/elasticsearch.service
             ├─5844 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networkaddress.cache.tt>
             └─6040 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64/bin/controller
```

#securing es

```
sudo ufw allow from 198.51.100.0 to any port 9200
```

```
pboviatsi@ubuntu:~$ sudo ufw allow from 198.51.100.0 to any port 9200
Rules updated
pboviatsi@ubuntu:~$
```

```
sudo ufw enable
```

```
pboviatsi@ubuntu:~$ sudo ufw enable
Firewall is active and enabled on system startup
pboviatsi@ubuntu:~$
```

Δοκιμή get και put για να δούμε αν μπορούν να εκτελεστούν queries.

```
curl -X GET 'http://localhost:9200'
```

```
pboviatsi@ubuntu:~$ curl -X GET 'http://localhost:9200'
{
  "name" : "ubuntu",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "t-K6x8nxRoyxTR4uzJkaFw",
  "version" : {
    "number" : "7.17.3",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "5ad023604c8d7416c9eb6c0eadb62b14e766caff",
    "build_date" : "2022-04-19T08:11:19.070913226Z",
    "build_snapshot" : false,
    "lucene_version" : "8.11.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
pboviatsi@ubuntu:~$
```

```
curl -X PUT 'http://localhost:9200/crawldb'
```

```
pboviatsi@ubuntu:~$ curl -X PUT 'http://localhost:9200/crawldb'
{"acknowledged":true,"shards_acknowledged":true,"index":"crawldb"}pboviatsi@ubuntu:~$
```

Run sparkler

Τρέχουμε το sparkler για ένα τυχαίο site, για να δούμε οτι μπορούμε να πάρουμε τον κώδικα του.

```
spark-submit --class edu.usc.irds.sparkler.Main --master
spark://localhost:7077 --driver-java-options
'-Dpf4j.pluginsDir=/home/pboviatsi/Desktop/sparkler/build/plugins' --jars
$(echo
/home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHOT/lib/*.j
ar | tr ' ' ',')
/home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHOT/lib/com
.kythera.sparkler-app-0.5.26-SNAPSHOT.jar inject -su https://news.bbc.co.uk
```

```
pboviatsi@ubuntu:/home$ spark-submit --class edu.usc.irds.sparkler.Main --master spark://localhost:7
077 --driver-java-options '-Dpf4j.pluginsDir=/home/pboviatsi/Desktop/sparkler/build/plugins' --jars
$(echo /home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHOT/lib/*.jar | tr ' ' ',') /
home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHOT/lib/com.kythera.sparkler-app-0.5.
26-SNAPSHOT.jar inject -su https://news.bbc.co.uk
22/05/02 00:17:17 WARN Utils: Your hostname, ubuntu resolves to a loopback address: 127.0.1.1; using
 192.168.88.130 instead (on interface ens33)
22/05/02 00:17:17 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
22/05/02 00:17:19 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... u
sing builtin-java classes where applicable
log4j:WARN No appenders could be found for logger (org.pf4j.DefaultPluginStatusProvider).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
>>jobId = sjob-1651475841833
pboviatsi@ubuntu:/home$
```

```
java -Xms1g -cp /home/pboviatsi/Desktop/sparkler/build/conf:$(echo
/home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHOT/lib/*.j
ar | tr ' ' ':')
-Dpf4j.pluginsDir=/home/pboviatsi/Desktop/sparkler/build/plugins
edu.usc.irds.sparkler.Main inject -id sjob-1 -su https://news.bbc.co.uk
```

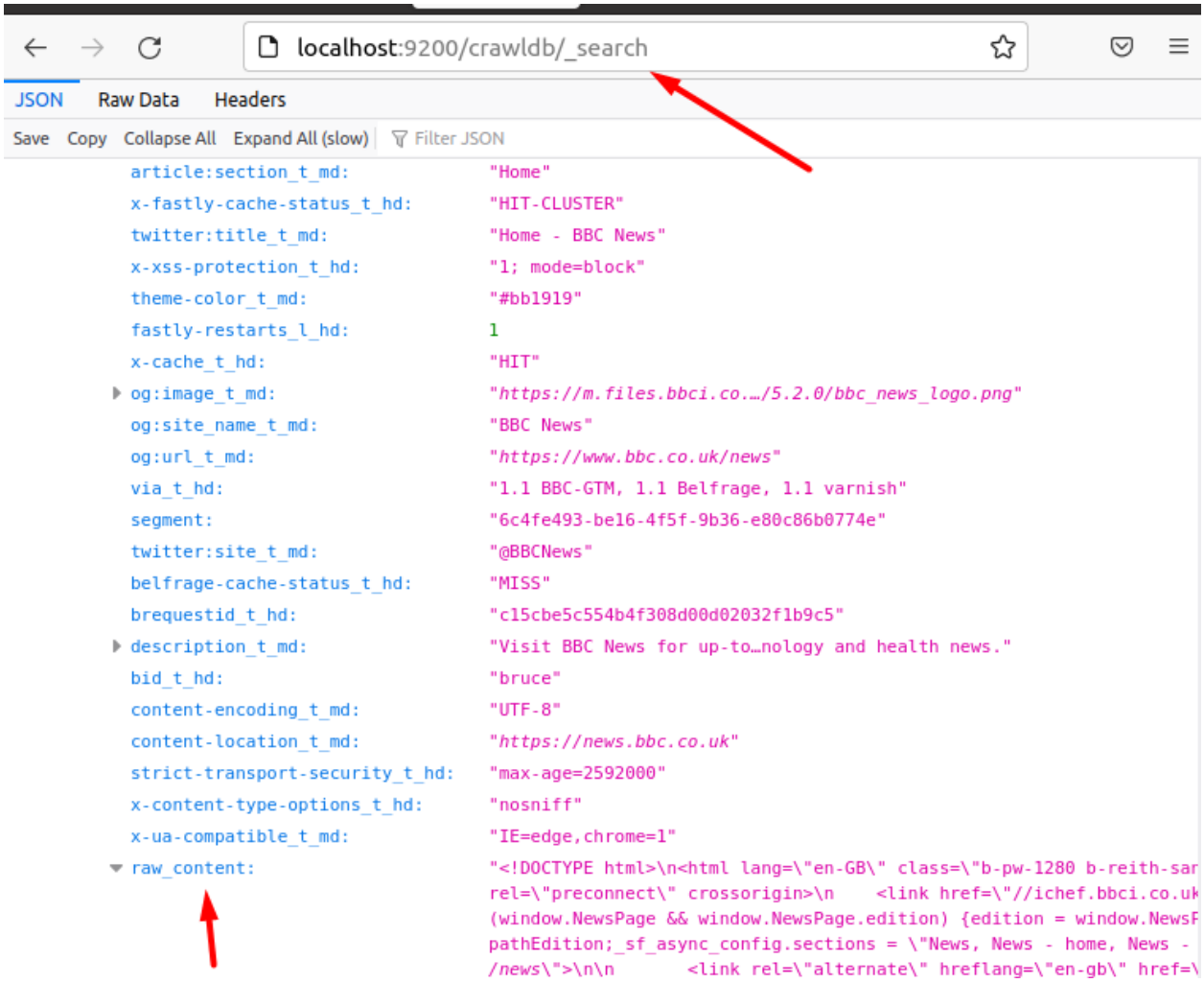
```
pboviatsi@ubuntu:/home$ java -Xms1g -cp /home/pboviatsi/Desktop/sparkler/build/conf:$(echo /home/pbo
viatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHOT/lib/*.jar | tr ' ' ':') -Dpf4j.pluginsDir
=/home/pboviatsi/Desktop/sparkler/build/plugins edu.usc.irds.sparkler.Main inject -id sjob-1 -su htt
ps://news.bbc.co.uk

SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHO
T/lib/ch.qos.logback.logback-classic-1.2.6.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHO
T/lib/org.apache.logging.log4j.log4j-slf4j-impl-2.11.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHO
T/lib/org.slf4j.slf4j-log4j12-1.7.30.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [ch.qos.logback.classic.util.ContextSelectorStaticBinder]
00:19:01.323 [main] INFO org.pf4j.DefaultPluginStatusProvider - Enabled plugins: []
00:19:01.329 [main] INFO org.pf4j.DefaultPluginStatusProvider - Disabled plugins: []
00:19:01.340 [main] INFO org.pf4j.DefaultPluginManager - PF4J version 3.6.0 in 'deployment' mode
00:19:01.342 [main] INFO edu.usc.irds.sparkler.service.PluginService$ - Loading plugins...
00:19:01.342 [main] DEBUG org.pf4j.AbstractPluginManager - Lookup plugins in '[/home/pboviatsi/Deskt
op/sparkler/build/plugins]'
```

```
00:19:07.013 [main] WARN org.elasticsearch.client.RestClient - request [PUT http://localhost:9200/crawldb/_doc/3979B75A8CDBE0558DD426BD1A08C2ED8CF3D2CCA218FE8D71B952B115874D0D?timeout=1m] returned 1 warnings: [299 Elasticsearch-7.17.3-5ad023604c8d7416c9eb6c0eadb62b14e766caff "Elasticsearch built-in security features are not enabled. Without authentication, your cluster could be accessible to anyone. See https://www.elastic.co/guide/en/elasticsearch/reference/7.17/security-minimal-setup.html to enable security."]
00:19:07.031 [main] DEBUG org.apache.http.impl.nio.conn.PoolingNHttpClientConnectionManager - Connection manager is shutting down
00:19:07.032 [main] DEBUG org.apache.http.impl.nio.conn.ManagedNHttpClientConnectionImpl - http-outgoing-0 127.0.0.1:59006<->127.0.0.1:9200[ACTIVE][r:r]: Close
00:19:07.033 [I/O dispatcher 1] DEBUG org.apache.http.impl.nio.client.InternalIODispatch - http-outgoing-0 [CLOSED]: Disconnected
00:19:07.047 [main] DEBUG org.apache.http.impl.nio.conn.PoolingNHttpClientConnectionManager - Connection manager shut down
>>jobId = sjob-1
00:19:07.062 [Thread-1] WARN edu.usc.irds.sparkler.service.PluginService$ - Stopping all plugins...
Runtime is about to exit.
00:19:07.063 [Thread-1] INFO org.pf4j.AbstractPluginManager - Stop plugin 'urlfilter-samehost@0.5.26-SNAPSHOT'
00:19:07.063 [Thread-1] INFO org.pf4j.AbstractPluginManager - Stop plugin 'urlfilter-regex@0.5.26-SNAPSHOT'
pboviatsi@ubuntu:/home$
```

```
java -Xms1g -cp /home/pboviatsi/Desktop/sparkler/build/conf:$(echo
/home/pboviatsi/Desktop/sparkler/build/sparkler-app-0.5.26-SNAPSHOT/lib/*.j
ar | tr ' ' ':')
-Dpf4j.pluginsDir=/home/pboviatsi/Desktop/sparkler/build/plugins
edu.usc.irds.sparkler.Main crawl -id sjob-1 -tn 10 -i 1
```

```
00:50:37.021 [SparkUI-30] DEBUG org.sparkproject.jetty.util.thread.QueuedThreadPool - ran org.sparkproject.jetty.util.thread.QueuedThreadPool$$Lambda$383/1970779713@252945b1 in QueuedThreadPool[SparkUI]@60afd40d{STOPPING,8<=0<=200,i=8,r=-1,q=0}[NO_TRY]
00:50:37.024 [SparkUI-32] DEBUG org.sparkproject.jetty.util.thread.QueuedThreadPool - Thread[SparkUI-32,5,main] exited for QueuedThreadPool[SparkUI]@60afd40d{STOPPING,8<=0<=200,i=8,r=-1,q=0}[NO_TRY]
00:50:37.024 [SparkUI-30] DEBUG org.sparkproject.jetty.util.thread.QueuedThreadPool - Thread[SparkUI-30,5,main] exited for QueuedThreadPool[SparkUI]@60afd40d{STOPPING,8<=0<=200,i=8,r=-1,q=0}[NO_TRY]
00:50:37.025 [main] DEBUG org.sparkproject.jetty.util.thread.QueuedThreadPool - Waiting for Thread[SparkUI-30,5,] for 14988
00:50:37.026 [main] DEBUG org.sparkproject.jetty.util.component.AbstractLifeCycle - STOPPED QueuedThreadPool[SparkUI]@60afd40d{STOPPED,8<=0<=200,i=8,r=-1,q=0}[NO_TRY]
00:50:37.026 [main] DEBUG org.sparkproject.jetty.util.component.AbstractLifeCycle - STOPPED Server@6831d8fd{STOPPED}[9.4.34.v20201102]
00:50:37.029 [main] INFO org.apache.spark.ui.SparkUI - Stopped Spark web UI at http://192.168.88.130:4040
00:50:37.107 [dispatcher-event-loop-3] INFO org.apache.spark.MapOutputTrackerMasterEndpoint - MapOutputTrackerMasterEndpoint stopped!
00:50:37.140 [main] INFO org.apache.spark.storage.memory.MemoryStore - MemoryStore cleared
00:50:37.141 [main] INFO org.apache.spark.storage.BlockManager - BlockManager stopped
00:50:37.153 [main] INFO org.apache.spark.storage.BlockManagerMaster - BlockManagerMaster stopped
00:50:37.169 [dispatcher-event-loop-0] INFO org.apache.spark.scheduler.OutputCommitCoordinator$OutputCommitCoordinatorEndpoint - OutputCommitCoordinator stopped!
00:50:37.184 [main] INFO org.apache.spark.SparkContext - Successfully stopped SparkContext
00:50:37.185 [Thread-17] WARN edu.usc.irds.sparkler.service.PluginService$ - Stopping all plugins... Runtime is about to exit.
00:50:37.186 [Thread-17] INFO org.pf4j.AbstractPluginManager - Stop plugin 'urlfilter-samehost@0.5.26-SNAPSHOT'
00:50:37.190 [Thread-17] INFO org.pf4j.AbstractPluginManager - Stop plugin 'urlfilter-regex@0.5.26-SNAPSHOT'
00:50:37.200 [shutdown-hook-0] INFO org.apache.spark.util.ShutdownHookManager - Shutdown hook called
00:50:37.201 [shutdown-hook-0] INFO org.apache.spark.util.ShutdownHookManager - Deleting directory /tmp/spark-ae1c7344-5b40-4993-9ee4-2b384565b00e
00:50:37.208 [Thread-2] DEBUG org.apache.hadoop.util.ShutdownHookManager - Completed shutdown in 0.020 seconds; Timeouts: 0
00:50:37.231 [Thread-2] DEBUG org.apache.hadoop.util.ShutdownHookManager - ShutdownHookManger completed shutdown.
pboviatsi@ubuntu:~$
```


localhost:9200/crawldb/_search
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All (slow)
Filter JSON
took: 17
timed_out: false
_shards:
total: 1
successful: 1
skipped: 0
failed: 0
hits:
total:
value: 119
relation: "eq"
max_score: 1
hits:
0:
_index: "crawldb"
_type: "_doc"
_id: "47DAEA42BE8DD45DBFE913E2…A893DB59198AEC8A1FE632A"
_score: 1
_ignored: […]
_source:
fetch_timestamp: "2022-05-02T00:50:23.236Z"
discover_depth: 0
parent: "seed"
seed: 1
retries_since_fetch: 0
version: "1.0"
url: "https://news.bbc.co.uk"
dedupe_id: "FB201A1A292AB4B44C82C984…66D60F6322A988B3FF14128"
hostname: "news.bbc.co.uk"
crawl_id: "sjob-1"
http_method: "GET"
modified_time: "2022-05-02T07:49:51.137Z"
jobmeta: null
indexed_at: "2022-05-02T07:49:51.142Z"
crawler: "sparkler"
generate_score: 0
last_updated_at: "2022-05-02T00:50:27.098Z"
fetch_status_code: "200"
fetch_depth: 0
response_time: 1615


localhost:9200/crawldb/_search
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All (slow)
Filter JSON
article:section_t_md: "Home"
x-fastly-cache-status_t_hd: "HIT-CLUSTER"
twitter:title_t_md: "Home - BBC News"
x-xss-protection_t_hd: "1; mode=block"
theme-color_t_md: "#bb1919"
fastly-restarts_l_hd: 1
x-cache_t_hd: "HIT"
og:image_t_md: "https://m.files.bbci.co.…/5.2.0/bbc_news_logo.png"
og:site_name_t_md: "BBC News"
og:url_t_md: "https://www.bbc.co.uk/news"
via_t_hd: "1.1 BBC-GTM, 1.1 Belfrage, 1.1 varnish"
segment: "6c4fe493-be16-4f5f-9b36-e80c86b0774e"
twitter:site_t_md: "@BBCNews"
belfrage-cache-status_t_hd: "MISS"
brequestid_t_hd: "c15cbe5c554b4f308d00d02032f1b9c5"
description_t_md: "Visit BBC News for up-to…nology and health news."
bid_t_hd: "bruce"
content-encoding_t_md: "UTF-8"
content-location_t_md: "https://news.bbc.co.uk"
strict-transport-security_t_hd: "max-age=2592000"
x-content-type-options_t_hd: "nosniff"
x-ua-compatible_t_md: "IE=edge,chrome=1"
raw_content:
"<!DOCTYPE html>\n<html lang=\"en-GB\" class=\"b-pw-1280 b-reith-sar
rel=\"preconnect\" crossorigin>\n <link href=\"//ichef.bbci.co.uk
(window.NewsPage && window.NewsPage.edition) {edition = window.NewsP
pathEdition;_sf_async_config.sections = \"News, News - home, News -
/news\">\n\n <link rel=\"alternate\" hreflang=\"en-gb\" href=\

Εγκατάσταση Kibana

```
wget -c
https://artifacts.elastic.co/downloads/kibana/kibana-7.4.2-linux-x86_64.tar
.gz -O - | tar -xz
```

```
sparkler@ubuntu:~/Desktop/sparkler$ cd ../
sparkler@ubuntu:~/Desktop$ wget -c https://artifacts.elastic.co/downloads/kibana
/kibana-7.4.2-linux-x86_64.tar.gz -O - | tar -xz
--2022-06-12 04:07:21--  https://artifacts.elastic.co/downloads/kibana/kibana-7.
4.2-linux-x86_64.tar.gz
Resolving artifacts.elastic.co (artifacts.elastic.co)... 34.120.127.130, 2600:19
01:0:1d7::
Connecting to artifacts.elastic.co (artifacts.elastic.co)|34.120.127.130|:443...
 connected.
HTTP request sent, awaiting response... 200 OK
Length: 252554263 (241M) [application/x-gzip]
Saving to: ‘STDOUT’

-                   100%[===================>] 240.85M  4.05MB/s    in 56s

2022-06-12 04:08:17 (4.31 MB/s) - written to stdout [252554263/252554263]
```

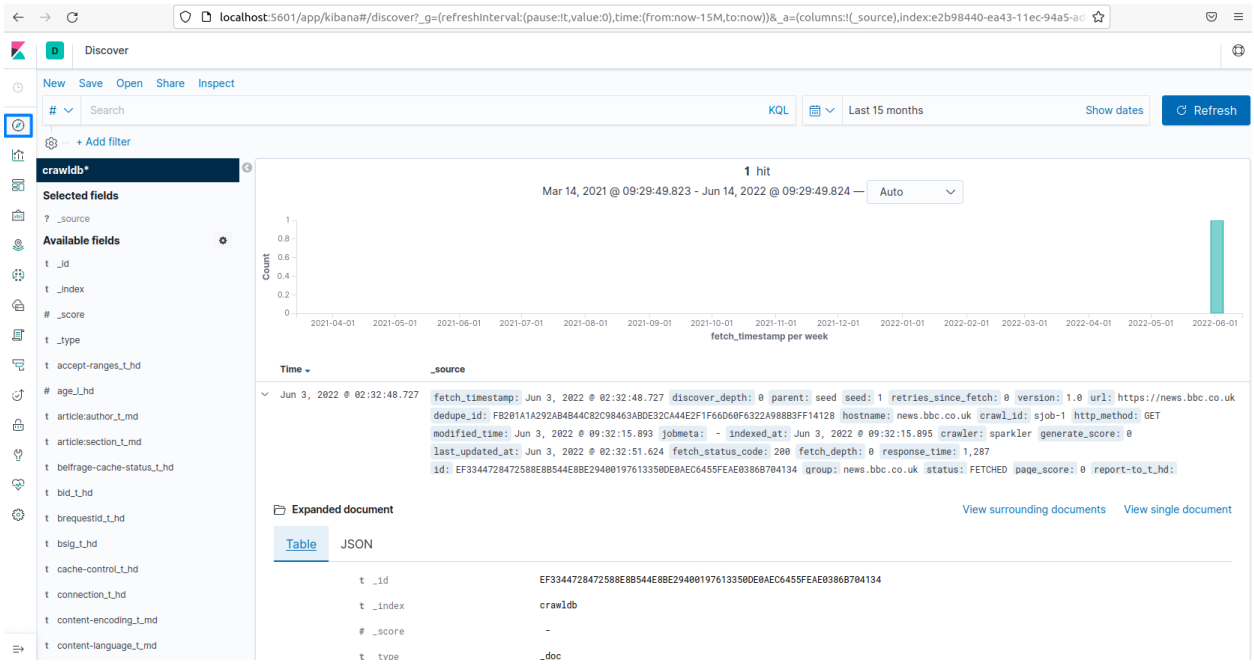
```
cd /crawler/kibana-7.4.2
```

Εκκίνηση του kibana

```
bin/kibana
```

```
sparkler@ubuntu:~/Desktop/kibana-7.4.2-linux-x86_64$ bin/kibana
  log   [11:25:54.899] [info][plugins-system] Setting up [4] plugins: [security,inspector,translations,data]
  log   [11:25:54.908] [info][plugins][security] Setting up plugin
  log   [11:25:54.909] [warning][config][plugins][security] Generating a random key for xpack.security.encryptionKey. To prevent
 sessions from being invalidated on restart, please set xpack.security.encryptionKey in kibana.yml
  log   [11:25:54.910] [warning][config][plugins][security] Session cookies will be transmitted over insecure connections. This
is not recommended.
  log   [11:25:54.992] [info][plugins][translations] Setting up plugin
  log   [11:25:54.992] [info][data][plugins] Setting up plugin
  log   [11:25:54.995] [info][plugins-system] Starting [3] plugins: [security,translations,data]
  log   [11:26:43.194] [info][status][plugin:kibana@7.4.2] Status changed from uninitialized to green - Ready
  log   [11:26:43.205] [info][status][plugin:elasticsearch@7.4.2] Status changed from uninitialized to yellow - Waiting for Elas
ticsearch
  log   [11:26:43.209] [info][status][plugin:xpack_main@7.4.2] Status changed from uninitialized to yellow - Waiting for Elastic
search
  log   [11:26:43.230] [info][status][plugin:telemetry@7.4.2] Status changed from uninitialized to green - Ready
  log   [11:26:43.235] [info][status][plugin:graph@7.4.2] Status changed from uninitialized to yellow - Waiting for Elasticsearc
h
  log   [11:26:43.259] [info][status][plugin:monitoring@7.4.2] Status changed from uninitialized to green - Ready
  log   [11:26:43.270] [info][status][plugin:spaces@7.4.2] Status changed from uninitialized to yellow - Waiting for Elasticsear
ch
  log   [11:26:43.356] [info][status][plugin:security@7.4.2] Status changed from uninitialized to green - Ready
  log   [11:26:43.359] [info][status][plugin:searchprofiler@7.4.2] Status changed from uninitialized to yellow - Waiting for Ela
```

Για να γίνει επαλήθευση ότι emfanizei ta dedomena pou exw apothikeusei sto elasticsearch, στο http://localhost:5601/



Script για να γίνεται εύκολα/γρήγορα η εκκίνηση του nutch

Για να μην χάνεται χρόνος με την εκκίνηση του nutch, φτιάχτηκαν τρια διαφορετικά script. Στο πρώτο γίνεται εκκίνηση του ElasticSearch. Στο δεύτερο γίνεται εκκίνηση του Kibana. Στο τρίτο, είναι μαζεμένες όλες οι εντολές οι οποίες χρειάζονται για να εκτελεστούν και να ξεκινήσει να μαζεύει δεδομένα.

Script es

```
#!/bin/bash

export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
/home/pboviatsi/Desktop/elasticsearch-7.4.2/bin/elasticsearch
```

Script kibana

```
#!/bin/bash

export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
/home/pboviatsi/Desktop/kibana-7.4.2-linux-x86_64/bin/kibana
```

Script start nutch

```
import os
import datetime

def getDateWithTime():
    timeInfo = datetime.datetime.now()
    return str(timeInfo.day)+'/' +str(timeInfo.month)+ '/'
```

```
+str(timeInfo.year)+ ' ' +str(timeInfo.hour)+ ':' + str(timeInfo.minute)+
':' +str(timeInfo.second)

if __name__ == '__main__':

    print("Begin!")

    timeInfo = datetime.datetime.now()
    dateFormat = str(timeInfo.day)+'-' +str(timeInfo.month)+ '-'
+str(timeInfo.year)
    javaHomePath = "/usr/lib/jvm/java-11-openjdk-amd64"
    nutchPath =
"/home/pboviatsi/Desktop/nutch-crawler/nutch/runtime/local/bin/nutch"
    clawldbPath =
"/home/pboviatsi/Desktop/nutch-crawler/nutch/runtime/local/crawl/crawldb"
    linkdbPath =
"/home/pboviatsi/Desktop/nutch-crawler/nutch/runtime/local/crawl/linkdb"
    urlPath =
"/home/pboviatsi/Desktop/nutch-crawler/nutch/runtime/local/urls"
    segmentsPath =
"/home/pboviatsi/Desktop/nutch-crawler/nutch/runtime/local/crawl/segments"
    logFilePath = "/home/pboviatsi/Desktop/"+dateFormat+".txt"
    depth = 2
    countUrl = 1000
    arrayWithSegments = []

    os.environ["JAVA_HOME"] = javaHomePath
    print("setenv JAVA_HOME", os.environ["JAVA_HOME"])

    os.system('rm -rf ' + segmentsPath)
    os.system('rm -rf ' + linkdbPath)
    os.system('rm -rf ' + clawldbPath)
    for x in range(depth):
       print("depth : ",x)

       if x == 0:
            os.system(nutchPath + ' inject ' + clawldbPath + ' ' + urlPath
+ ' >> ' + logFilePath)
            print(getDateWithTime()+"---> end of url inject")

            os.system(nutchPath + ' generate ' + clawldbPath + ' ' +
segmentsPath + ' >> ' + logFilePath)
            print(getDateWithTime()+"---> end of crawldb generate")
```

```
        else:
                os.system(nutchPath + ' generate ' + clawldbPath + ' ' +
segmentsPath+ ' -topN ' + str(countUrl) + ' >> ' + logFilePath)
                print(getDateWithTime()+"---> end of crawldb generate")

        newFile = os.listdir(segmentsPath+'/')
        newFile.sort()
        lengthOfFiles = len(newFile)

        if lengthOfFiles == 0:
                print("lengthOfFiles is 0")

        if lengthOfFiles > 1:
                segment = segmentsPath + '/' + str(newFile[-1])
        else:
                segment = segmentsPath + '/' + str(newFile[0])
        print(getDateWithTime()+"---> segment path is",segment)
        arrayWithSegments.append(segment)

        os.system(nutchPath + ' fetch ' + segment + ' >> ' + logFilePath)
        print(getDateWithTime()+"---> end of fetch")

        os.system(nutchPath + ' parse ' + segment + ' >> ' + logFilePath)
        print(getDateWithTime()+"---> end of parse")

        os.system(nutchPath + ' updatedb ' + clawldbPath + ' ' + segment + '
>> ' + logFilePath)
        print(getDateWithTime()+"---> end of updatedb")

    print(getDateWithTime()+"---> array with segments :
",arrayWithSegments)

    os.system(nutchPath + ' invertlinks ' + linkdbPath + ' -dir ' +
segmentsPath + ' >> ' + logFilePath)
    print(getDateWithTime()+"---> end of invertlinks")


    for x in arrayWithSegments:
        os.system(nutchPath + ' index ' + clawldbPath + ' -linkdb ' +
linkdbPath + ' ' + x + ' -filter -normalize -deleteGone' + ' >> ' +
logFilePath)
        print(getDateWithTime()+"---> end of save data")
    else:
```

```
        print(getDateWithTime()+"---> Finally finished!")
```

Παράρτημα 3

Κώδικας scraper

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import WebDriverWait
from time import sleep
from bs4 import BeautifulSoup
import unicodedata
from ast import literal_eval
import requests
import json
from datetime import datetime

driver = webdriver.Chrome("C:\chromedriver\chromedriver.exe")

months = {
   "Ιανουαρίου": "January",
   "Φεβρουαρίου": "February",
   "Μαρτίου": "March",
   "Απριλίου": "April",
   "Μαΐου": "May",
   "Ιουνίου": "June",
   "Ιουλίου": "July",
   "Αυγούστου": "August",
   "Σεπτεμβρίου": "September",
   "Οκτωβρίου": "October",
   "Νοεμβρίου": "November",
   "Δεκεμβρίου": "December"
}
headers = {"Accept": "application/json", "Content-type":
"application/json"}
sites = requests.get('http://localhost:8000/allUrls', headers=headers,
verify=False)
dataSites = sites.json()

for site in dataSites['response']['hits']['hits']:
```

```
urlSearch = site['_source']['url']
site = site['_source']['site']

allSelectors = {
    'in.gr': {
        'nextPage': '.nxtnav > a',
        'search': '.flexgrid > a',
        'selectorId': '#single-post > article',
        'urlImg': '.post-main-image img',
        'title': '.art-start > header > h1',
        'date': '.ingr-postmeta > .meta-author > time',
        'bodyDescription': '.main-content > .description',
        'body': '.main-content > div > p',
    },
    'thebest.gr': {
        'nextPage': '.col.text-left > a',
        'search': '.flex-column > header > a',
        'selectorId': '',
        'urlImg': '#view_article > figure > picture > img',
        'title': 'header > .article-header',
        'date': '#view_article > div > time',
        'bodyDescription': '.article-content > p > em',
        'body': '.bodypart-text > p',
    }
}

driver.get(urlSearch)

urls = []

while 1:
    soupAllResults = BeautifulSoup(driver.page_source, "html.parser")
    nextPage = soupAllResults.select(allSelectors[site]['nextPage'])
    if nextPage:
        # print(nextPage[0])
        search = soupAllResults.select(allSelectors[site]['search'])

        for h in search:
            # print(h.get('href'))
            urls.append(h.get('href'))

        driver.get(nextPage[0].get('href'))
    else:
```

```
            break
    i = 1
    articleResults = {}

    print("All articles url have been saved", len(urls))

    for url in urls:
        driver.get(url)
        sleep(10)
        soup = BeautifulSoup(driver.page_source, "html.parser")

        urlImg = soup.select_one(allSelectors[site]['urlImg'])

        if urlImg:
            urlImg = driver.find_element(By.CSS_SELECTOR,
allSelectors[site]['urlImg']).get_attribute("src")

        title = soup.select_one(allSelectors[site]['title'])

        if title:
            title = title.get_text().replace("\n", "").strip()

        date = soup.select_one(allSelectors[site]['date'])

        if date:
            if site == 'in.gr':
                date = driver.find_element(By.CSS_SELECTOR,
allSelectors[site]['date']).get_attribute("title")
                date = date.split(" ")
                date[1] = months[date[1]]
                date = ' '.join(date)
                dateWithFormat = datetime.strptime(date, '%d %B %Y, %H:%M')
            else:
                date = date.get_text().replace("\n", "").strip()
                dateWithFormat = datetime.strptime(date, '%d/%m/%Y, %H:%M')

        bodyDescription =
soup.select_one(allSelectors[site]['bodyDescription'])

        if bodyDescription:
            bodyDescription = bodyDescription.get_text().replace("\n",
"").strip()
```

```
        articleBody = soup.select(allSelectors[site]['body'])
        finalArticleBody = ''

        if articleBody:
            for articleParagraph in articleBody:
                # remove \xa0 from body use
unicodedata.normalize("NFKD",...)
                finalArticleBody += unicodedata.normalize("NFKD",
articleParagraph.get_text())

        if allSelectors[site]['selectorId']:
            articleId = driver.find_element(By.CSS_SELECTOR,
allSelectors[site]['selectorId']).get_attribute("id")
        else:
            articleIndexId = url.find("article/") + 8
            articleIndex = url.find("-", articleIndexId)
            articleId = url[articleIndexId:articleIndex]

        articleResults = {
            'id': site + '_' + articleId,
            'urlImg': str(urlImg),
            'title': title,
            'date': dateWithFormat.strftime('%d/%m/%Y'),
            'bodyDescription': bodyDescription,
            'finalArticleBody': finalArticleBody
        }

        print("Article : ", articleResults)
        headers = {"Accept": "application/json", "Content-type":
"application/json"}
        requests.post('http://localhost:8000/addPost', json=articleResults,
headers=headers, verify=False)
        print("Saved :", i)

        i = i + 1

print("End crawling")
driver.close()
```

Κώδικας front-end εφαρμογής

App.js

```
import './App.css';
import * as React from 'react';
import {BrowserRouter as Router, Routes, Route} from 'react-router-dom';
import Typography from "@mui/material/Typography";
import {Paper} from "@mui/material";
import NavBar from './components/navbar'
import Statistics from './pages/statistics'
import Search from './pages/search'

export default function App() {
   return (
       <Router>
           <div style={{marginBottom: "140px"}}>
               <NavBar/>
               <Routes>
                   <Route exact path='/' element={<Search/>}/>
                   <Route exact path='/statistics'
element={<Statistics/>}/>
                   <Route exact path='/search' element={<Search/>}/>
               </Routes>
           </div>
           <footer style={{
               color: "gray",
               position: "fixed",
               bottom: 0,
               width: "100%",
               height: "100px",
               marginBottom: "0px"
           }}>
               <Paper style={{
                   width: "100%",
                   height: "100px",
                   display: "flex",
                   flexDirection: "column",
                   alignItems: "center",
                   justifyContent: "center"
               }} elevation={1}>
                   <Typography variant="h5" component="h3">
                       Scrap Project
```

```
                    </Typography>
                    <Typography component="p">
                        @2022 All right reserved
                    </Typography>
                </Paper>
            </footer>
        </Router>
    );
}
```

Components

card.js

```
import * as React from 'react';
import Card from '@mui/material/Card';
import CardHeader from '@mui/material/CardHeader';
import CardMedia from '@mui/material/CardMedia';
import CardContent from '@mui/material/CardContent';
import CardActions from '@mui/material/CardActions';
import IconButton from '@mui/material/IconButton';
import Typography from '@mui/material/Typography';
import Button from '@mui/material/Button';
import Dialog from '@mui/material/Dialog';
import DialogTitle from '@mui/material/DialogTitle';
import DialogContent from '@mui/material/DialogContent';
import CloseIcon from '@mui/icons-material/Close';
import DialogContentText from '@mui/material/DialogContentText';
import {Chip, Grid} from "@mui/material";
import CalendarMonthIcon from '@mui/icons-material/CalendarMonth';
import {useEffect} from "react";
import axios from "axios";
import Pagination from '@mui/material/Pagination';
import Stack from '@mui/material/Stack';
import LinearProgress from '@mui/material/LinearProgress';
import {Bar} from "react-chartjs-2";

export interface DialogTitleProps {
   id: string;
   children?: React.ReactNode;
   onClose: () => void;
}

const BootstrapDialogTitle = (props: DialogTitleProps) => {
```

```
    const {children, onClose, ...other} = props;

    return (
        <DialogTitle sx={{m: 0, p: 2}} {...other}>
            {children}
            {onClose ? (
                <IconButton
                    aria-label="close"
                    onClick={onClose}
                    sx={{
                        position: 'absolute',
                        right: 8,
                        top: 8,
                        color: (theme) => theme.palette.grey[500],
                    }}
                >
                    <CloseIcon/>
                </IconButton>
            ) : null}
        </DialogTitle>
    );
};

export default function RecipeReviewCard({typePage, finalValue}) {
    const [open, setOpen] = React.useState(false);
    const [posts, setPosts] = React.useState([]);
    const [allPosts, setAllPosts] = React.useState([]);
    const descriptionElementRef = React.useRef(null);
    const [page, setPage] = React.useState(1);
    const [progress, setProgress] = React.useState(false);
    const [dialogData, setDialogData] = React.useState({
        bodyDescription: "",
        finalArticleBody: "",
        date: "",
        title: "",
        open: false
    });
    const [numberOfPosts, setNumberOfPosts] = React.useState(0);
    const [numberOfAllPosts, setNumberOfAllPosts] = React.useState(0);

    const handleClickOpen = (postDialog) => {
        setProgress(true)
        setOpen(true);
```

```
        setDialogData({...postDialog, open: true})
        setProgress(false)
    };

    const handleClose = () => {
        setProgress(true);
        setOpen(false);
        setDialogData({bodyDescription: "", finalArticleBody: "", date: "",
title: "", open: false})
        setProgress(false)
    };

    const getPosts = async () => {
        if (typePage === "Article" || !finalValue) {
            setProgress(true)
            const {data} = await axios.get(`http://localhost:8000/allPosts`)
            setPosts(data.response.hits.hits.slice(0, 9));
            setAllPosts(data.response.hits.hits);
            setProgress(false)
        } else if (typePage === "Search") {
            setProgress(true)
            const {data} = await
axios.post(`http://localhost:8000/searchPost?key=${finalValue}`)
            setPosts(data.response.hits.hits.slice(0, 9));
            setAllPosts(data.response.hits.hits);
            setNumberOfPosts(data.response.hits.hits.length)
            const allData = await
axios.get(`http://localhost:8000/allPosts`)
            setNumberOfAllPosts(allData.data.response.hits.hits.length)
            setProgress(false)
        }
    };

    const setPagination = (event, value) => {
        setPage(value)
        setPosts(allPosts.slice((value - 1) * 9, (value * 10) - 1))
    }

    const stateBar = {
        labels: ['Άρθρα'],
        datasets: [
            {
                label: "Άρθρα με το λεκτικό:" + finalValue,
```

```
                backgroundColor: [
                    'rgba(75, 192, 192, 0.2)'
                ],
                borderColor: [
                    'rgb(75, 192, 192)'
                ],
                borderWidth: 2,
                data: [numberOfPosts]
            },
            {
                label: 'Όλα τα άρθρα',
                backgroundColor: [
                    'rgba(255, 159, 64, 0.2)'
                ],
                borderColor: [
                    'rgb(255, 159, 64)'
                ],
                borderWidth: 2,
                data: [numberOfAllPosts]
            }
        ]
    }

    useEffect(() => {
        if (open) {
            const {current: descriptionElement} = descriptionElementRef;
            if (descriptionElement !== null) {
                descriptionElement.focus();
            }
        }
    }, [open]);

    useEffect(() => {
        getPosts();
    }, [typePage])

    useEffect(() => {
        getPosts();
    }, [finalValue])

    return (
        <> {progress ? <Stack id="stackLoader" sx={{height: '100%', width:
'100%'}}><LinearProgress/> </Stack> :
```

```
            <Grid container rowSpacing={1} columnSpacing={{xs: 1, sm: 2, md:
3}}>
                {numberOfPosts && finalValue ?
                    <>
                        <Grid item xs={0} sm={4} md={4}>
                        </Grid>
                        <Grid item xs={12} sm={4} md={4} style={{display:
'flex', alignItems: 'center'}}>
                            <Bar
                                data={stateBar}
                                options={{
                                    title: {
                                        display: true,
                                        text: 'Average Rainfall per month',
                                        fontSize: 20
                                    },
                                    legend: {
                                        display: true,
                                        position: 'right'
                                    },
                                }}/>
                        </Grid>
                        <Grid item xs={0} sm={4} md={4}>
                        </Grid>
                    </> : ''}
                <Grid item xs={12}>
                    <Grid container rowSpacing={1} columnSpacing={{xs: 1,
sm: 2, md: 3}}>

                        {posts.map((post) => {
                            return (
                                <Grid item xs={4}>
                                    <Card sx={{maxWidth: 400}}
style={{margin: 10}}>
                                        <CardHeader
                                            style={{minHeight: 119,
maxHeight: 119}}
                                            title={post._source.title.length
> 110 ? post._source.title.substring(0, 110) + '...' : post._source.title}
                                            subheader={
                                                <div style={{display:
'flex'}}>
                                                    <CalendarMonthIcon
```

```
fontSize={"small"}

style={{marginRight: '5px'}}/>{post._source.date}
                                                </div>
                                            }
                                        />
                                        <Chip label="in.gr"
                                              style={{
                                                  borderRadius: 0,
                                                  display: 'flex',
                                                  justifyContent: 'center',
                                                  flexDirection: 'row',
                                                  alignItems: 'center'
                                              }}/>
                                        <CardMedia
                                            component="img"
                                            height="194"
                                            image={post._source.urlImg}
                                            alt={post._source.title}
                                        />
                                        <CardContent>
                                            <Typography variant="body2"
color="text.secondary">

{post._source.descriptionBody}
                                            </Typography>
                                        </CardContent>
                                        <CardActions>
                                            <Button onClick={() =>
handleClickOpen(post._source)} size="small">Δειτε
                                                Περισσοτερα</Button>
                                        </CardActions>
                                    </Card>
                                </Grid>
                            )
                        })}
                    </Grid>
                </Grid>
                <Grid item xs={12}>
                    <Stack spacing={2}>
                        <Pagination page={page}
count={Math.ceil(allPosts.length / 9)} onChange={setPagination}
                                    variant="outlined" color="primary"
```

```
                                    style={{display: 'flex', justifyContent:
'center'}}/>
                    </Stack>
                </Grid>
                <Dialog
                    open={dialogData.open}
                    onClose={handleClose}
                    fullWidth={true}
                    scroll='paper'
                    aria-labelledby="scroll-dialog-title"
                    aria-describedby="scroll-dialog-description"
                >
                    <BootstrapDialogTitle id="customized-dialog-title"
onClose={handleClose}>
                        <h4>{dialogData.title}</h4>
                        <h5 style={{margin: 0, color: '#4c4c4cc7', display:
'flex'}}>
                            <CalendarMonthIcon fontSize={"small"}
style={{marginRight: '5px'}}/>
                            {dialogData.date}
                        </h5>
                    </BootstrapDialogTitle>
                    <DialogContent>
                        <DialogContentText
                            id="scroll-dialog-description"
                            ref={descriptionElementRef}
                            tabIndex={-1}
                        >
                            <b>
                                {dialogData.bodyDescription}
                            </b>
                        </DialogContentText>
                        <br/>
                        <DialogContentText
                            id="scroll-dialog-description"
                            ref={descriptionElementRef}
                            tabIndex={-1}
                        >
                            {dialogData.finalArticleBody}
                        </DialogContentText>
                    </DialogContent>
                </Dialog>
            </Grid>
```

```
        }</>
    );
}
```

navBar.js

```
import * as React from 'react';
import AppBar from '@mui/material/AppBar';
import Box from '@mui/material/Box';
import Toolbar from '@mui/material/Toolbar';
import IconButton from '@mui/material/IconButton';
import Typography from '@mui/material/Typography';
import Menu from '@mui/material/Menu';
import MenuIcon from '@mui/icons-material/Menu';
import Container from '@mui/material/Container';
import Button from '@mui/material/Button';
import MenuItem from '@mui/material/MenuItem';
import AdbIcon from '@mui/icons-material/Adb';
import {Link} from "react-router-dom";

const pages = ['Search', 'Statistics'];

export default function NavBar() {
    const [anchorElNav, setAnchorElNav] = React.useState(null);
    const [anchorElUser, setAnchorElUser] = React.useState(null);

    const handleOpenNavMenu = (event) => {
        setAnchorElNav(event.currentTarget);
    };
    const handleOpenUserMenu = (event) => {
        setAnchorElUser(event.currentTarget);
    };

    const handleCloseNavMenu = (page) => {
        setAnchorElNav(null);
    };

    const handleCloseUserMenu = () => {
        setAnchorElUser(null);
    };

    return (
        <AppBar position="static">
            <Container maxWidth="xl">
```

```
                <Toolbar disableGutters>
                    <Box sx={{flexGrow: 1, display: {xs: 'flex', md:
'none'}}}>
                        <IconButton
                            size="large"
                            aria-label="account of current user"
                            aria-controls="menu-appbar"
                            aria-haspopup="true"
                            onClick={handleOpenNavMenu}
                            color="inherit"
                        >
                            <MenuIcon/>
                        </IconButton>
                        <Menu
                            id="menu-appbar"
                            anchorEl={anchorElNav}
                            anchorOrigin={{
                                vertical: 'bottom',
                                horizontal: 'left',
                            }}
                            keepMounted
                            transformOrigin={{
                                vertical: 'top',
                                horizontal: 'left',
                            }}
                            open={Boolean(anchorElNav)}
                            onClose={handleCloseNavMenu}
                            sx={{
                                display: {xs: 'block', md: 'none'},
                            }}
                        >
                            {pages.map((page) => (
                                <MenuItem key={page}
onClick={handleCloseNavMenu}>
                                    <Typography
textAlign="center">{page}</Typography>
                                </MenuItem>
                            ))}
                        </Menu>
                    </Box>
                    <AdbIcon sx={{display: {xs: 'flex', md: 'none'}, mr:
1}}/>
```

```
                    <Box sx={{flexGrow: 1, display: {xs: 'none', md:
'flex'}}}>
                        {pages.map((page) => (
                            <Button
                                key={page}
                                component={Link}
                                to={{pathname: '/' + page.toLowerCase()}}
                                sx={{my: 2, color: 'white', display:
'block'}}
                            >
                                {page}
                            </Button>
                        ))}
                    </Box>

                </Toolbar>
            </Container>
        </AppBar>
    );
};
```

Pages

search.js

```
import * as React from 'react';
import {Grid, Box, TextField} from "@mui/material";
import Card from '../components/card';
import Autocomplete from '@mui/material/Autocomplete';
import Button from "@mui/material/Button";

export default function Search() {
    const [inputValueKey, setInputValueKey] = React.useState("");
    const [finalValue, setFinalValue] = React.useState("");
    const searchByKeyword = async () => {
        setFinalValue(inputValueKey)
    }

    return (
        <div>
            <br/>
            <Grid container>
                <Grid item xs={0} sm={2} md={2}>
                </Grid><Grid item xs={12} sm={8} md={8} style={{display:
```

```
"flex"}}>

                <Autocomplete
                    freeSolo={true}
                    options={[]}
                    inputValue={inputValueKey}
                    onInputChange={(event, newInputValue) => {
                        setInputValueKey(newInputValue);
                    }}
                    id="controllable-states-demo"
                    style={{margin: "10px", marginLeft: "25px"}}
                    fullWidth
                    renderInput={(params) => <TextField
                        fullWidth
                        id="standard-bare"
                        variant="outlined"
                        placeholder="Search by keywords"
                        {...params}
                    />}
                />
                <Button color="secondary" variant="contained"
onClick={searchByKeyword}
                        style={{height: "53px", margin: "10px", marginRight:
"25px"}}>Search</Button>
            </Grid>
                <Grid item xs={0} sm={2} md={2}>
                </Grid>
            </Grid>
            <h1 style={{
                display: 'flex',
                justifyContent: 'center'
            }}>{finalValue ? "Αποτελέσματα για " + finalValue : "Όλα τα
άρθρα"}</h1>
            <Box>
                <Grid container>
                    <Grid item xs={0} sm={1} md={1}>
                    </Grid>
                    <Grid item xs={12} sm={10} md={11}>
                        <div>
                            <br/><br/><br/>
                            <Grid container spacing={{xs: 2, md: 3}}
columns={{xs: 4, sm: 8, md: 12}}>
                                <Grid item xs={12} key={1} style={{display:
```

```
'contents'}}>


                                    <Card typePage="Search"
finalValue={finalValue}/>
                                </Grid>
                            </Grid>
                        </div>
                    </Grid>
                </Grid>
            </Box>

        </div>
    );
}
```

statistics.js

```
import React, {useEffect} from 'react';
import {Bar, Line, Pie, Doughnut} from 'react-chartjs-2';
import {registerables, Chart} from 'chart.js';
import axios from "axios";
import moment from "moment";
import {Grid} from "@mui/material";

Chart.register(...registerables)

export default function Statistics() {
    const [posts, setPosts] = React.useState([]);
    const [postsInMonth, setPostsInMonth] = React.useState([0, 0]);
    const [postsInMonthsIn, setPostsInMonthsIn] = React.useState([0, 0, 0,
0, 0]);
    const [postsInMonthsTheBest, setPostsInMonthsTheBest] =
React.useState([0, 0, 0, 0, 0]);

    const getPosts = async () => {
        const {data} = await axios.get(`http://localhost:8000/allPosts`)
        let posts_in_month = [0, 0]
        let posts_in_months_in = [0, 0, 0, 0, 0]
        let posts_in_months_theBest = [0, 0, 0, 0, 0]


        data.response.hits.hits.map((dt) => {
            if (dt._id.indexOf("thebest.gr") !== -1) {
```

```
                posts_in_month[1]++
                moment(dt._source.date, "DD/MM/YYYY").month()
                const dateMoment = moment(dt._source.date,
"DD/MM/YYYY").month() + 1
                posts_in_months_theBest[dateMoment - 3]++
            } else if (dt._id.indexOf("in.gr") !== -1) {
                posts_in_month[0]++
                try {
                    const dateMoment = moment(dt._source.date,
"DD/MM/YYYY").month() + 1
                    posts_in_months_in[dateMoment - 3]++
                } catch (e) {
                    console.log(e)
                }
            }
        })
        setPostsInMonth(posts_in_month)
        setPostsInMonthsIn(posts_in_months_in)
        setPostsInMonthsTheBest(posts_in_months_theBest)
        setPosts(data.response.hits.hits);
    };

    const statePie = {
        labels: ['in.gr',
            'thebest.gr'],
        datasets: [
            {
                label: 'Rainfall',
                backgroundColor: [
                    'rgb(75, 192, 192)',
                    'rgb(255, 159, 64)'
                ],
                hoverBackgroundColor: [
                    'rgba(75, 192, 192, 0.2)',
                    'rgba(255, 159, 64, 0.2)'
                ],
                data: postsInMonth
            }
        ]
    }

    const stateBar = {
        labels: ['Μάρτιος',
```

```
            'Απρίλιος',
            'Μάιος',
            'Ιούνιος',
            'Ιούλιος'],
        datasets: [
            {
                label: 'in.gr',
                backgroundColor: [
                    'rgba(75, 192, 192, 0.2)'
                ],
                borderColor: [
                    'rgb(75, 192, 192)'
                ],
                borderWidth: 2,
                data: postsInMonthsIn
            },
            {
                label: 'thebest.gr',
                backgroundColor: [
                    'rgba(255, 159, 64, 0.2)'
                ],
                borderColor: [
                    'rgb(255, 159, 64)'
                ],
                borderWidth: 2,
                data: postsInMonthsTheBest
            }
        ]
    }

    useEffect(() => {
        getPosts();
    }, [])

    return (
        <>
            <h1 style={{display: 'flex', justifyContent:
'center'}}>Στατιστικά</h1>
            <Grid container spacing={2} style={{padding: 100, display:
'flex', justifyContent: 'center'}}>
                <Grid item xs={5} style={{display: 'flex', alignItems:
'center'}}>
                    <Line
```

```
                    data={stateBar}
                    options={{
                        title: {
                            display: true,
                            text: 'Average Rainfall per month',
                            fontSize: 20
                        },
                        legend: {
                            display: true,
                            position: 'right'
                        },
                    }}/>
            </Grid>
            <Grid item xs={5} style={{display: 'flex', alignItems:
'center'}}>
                <Bar
                    data={stateBar}
                    options={{
                        title: {
                            display: true,
                            text: 'Average Rainfall per month',
                            fontSize: 20
                        },
                        legend: {
                            display: true,
                            position: 'right'
                        },
                    }}/>
            </Grid>
            <Grid item xs={3} style={{display: 'flex', alignItems:
'center'}}>
                <Pie
                    data={statePie}
                    options={{
                        title: {
                            display: true,
                            text: 'Average Rainfall per month',
                            fontSize: 20
                        },
                        legend: {
                            display: true,
                            position: 'right'
                        }
```

```
                    }}
                />
            </Grid>
            <Grid item xs={3} style={{display: 'flex', alignItems:
'center'}}>
                <Doughnut
                    data={statePie}
                    options={{
                        title: {
                            display: true,
                            text: 'Average Rainfall per month',
                            fontSize: 20
                        },
                        legend: {
                            display: true,
                            position: 'right'
                        }
                    }}
                />
            </Grid>
        </Grid>
    </>
  );
}
```

Κώδικας backend-end εφαρμογής / swagger

```
from fastapi import FastAPI
from opensearchpy import OpenSearch
from typing import Union
from pydantic import BaseModel
from fastapi.encoders import jsonable_encoder
from fastapi.middleware.cors import CORSMiddleware
import requests

tags_metadata = [
   {
       "name": "Posts",
       "description": "Posts"
   },
   {
       "name": "Urls",
       "description": "Urls"
```

```
    },

]

app = FastAPI(openapi_tags=tags_metadata)
app.add_middleware(CORSMiddleware, allow_credentials=True,
allow_methods=["*"], allow_origins=["*"],
                   allow_headers=["*"])
index_name = 'posts'
index_url_name = 'urls'
hosts = ["localhost"]
port = 9200
client = OpenSearch(
    hosts=hosts,
    port=port
)

class Item(BaseModel):
    id: str
    urlImg: str
    title: str
    date: str
    bodyDescription: str
    finalArticleBody: str

class Url(BaseModel):
    url: str
    site: str

@app.get("/allPosts", tags=["Posts"])
async def allPosts():
    # Search for the document.
    q = {'match_all': {}}
    query = {
        'size': 10000,
        'query': q
    }

    response = client.search(
        body=query,
        index=index_name
    )
    print('Search results:')
```

```
    return {'response': response}  # return data with 200 OK

@app.get("/post/{id}", tags=["Posts"])
async def post(id):
    # Search for the document.
    q = {
        "bool": {
            "must": [
                {"term": {"_id": id}}
            ]
        }
    }
    query = {
        'size': 50,
        'query': q
    }

    response = client.search(
        body=query,
        index=index_name
    )
    print('Search results:')
    return {'response': response}  # return data with 200 OK

@app.post("/addPost/", tags=["Posts"])
async def addPost(item: Item):
    jsonData = jsonable_encoder(item)
    headers = {"Accept": "application/json", "Content-type":
"application/json"}
    x = requests.put('http://localhost:9200/posts/_doc/' +
str(jsonData["id"]), json=jsonData, headers=headers,
                     verify=False)
    print('Insert status:', x.status_code)
    return {'response': jsonData}, 200  # return data with 200 OK


@app.post("/searchPost/", tags=["Posts"])
async def searchPost(key):
    query = {
        'size': 10000,
        "query": {
            "bool": {
                "must": {
```

```
                    "multi_match": {
                        "query": key,
                        "fields": ["title", "bodyDescription",
"finalArticleBody"]
                    }
                }
            }
        }
    }

    response = client.search(
        body=query,
        index=index_name
    )
    print('Search results:')
    return {'response': response}  # return data with 200 OK

@app.post("/addUrl/", tags=["Urls"])
async def addUrl(url: Url):
    jsonData = jsonable_encoder(url)
    headers = {"Accept": "application/json", "Content-type":
"application/json"}
    x = requests.post('http://localhost:9200/urls/_doc', json=jsonData,
headers=headers, verify=False)
    print('Insert status:', x.status_code)
    return {'response': jsonData}, 200  # return data with 200 OK

@app.get("/allUrls", tags=["Urls"])
async def allUrls():
    # Search for the document.
    q = {'match_all': {}}
    query = {
        'size': 50,
        'query': q
    }

    response = client.search(
        body=query,
        index=index_url_name
    )
    print('Search results:')
    return {'response': response}  # return data with 200 OK
```