Diagram

Description automatically generated

Μηχανή αναζήτησης άρθρων σχετικά με τον Covid-19

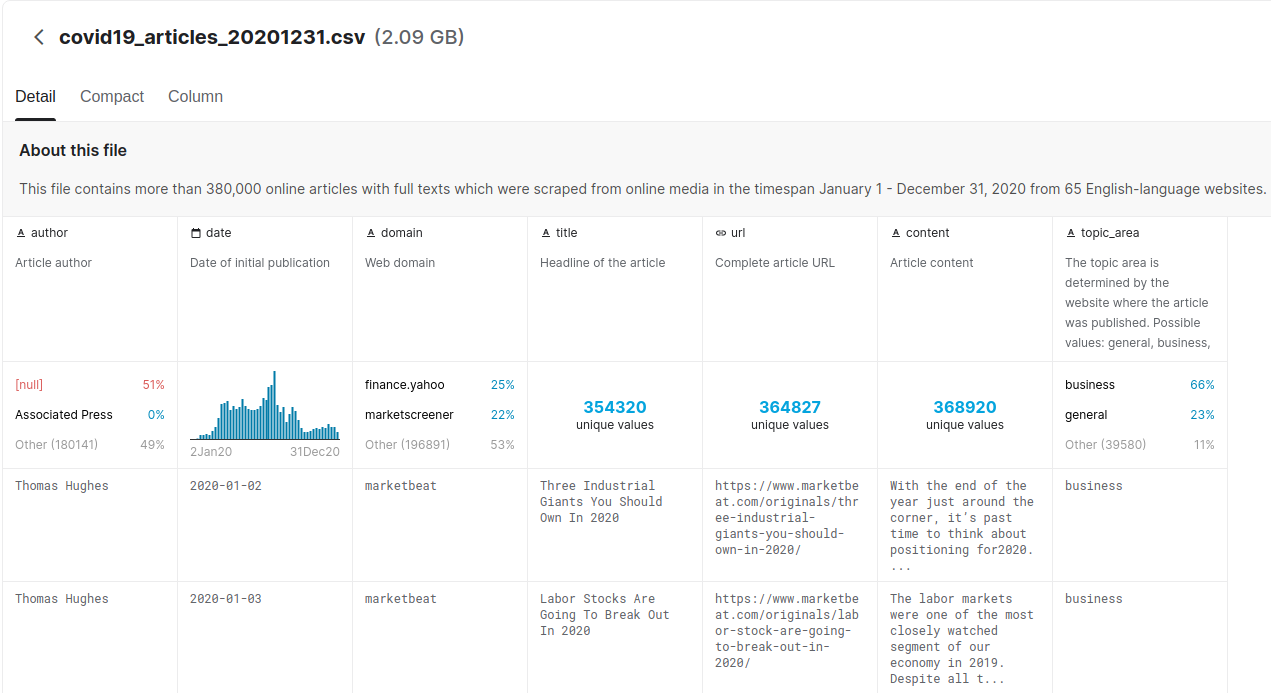
Μπαρμπαλιάς Ευάγγελος 2766

Ζέρβας Δημήτριος 2784

Ανάκτηση Πληροφορίας | 28-5-21

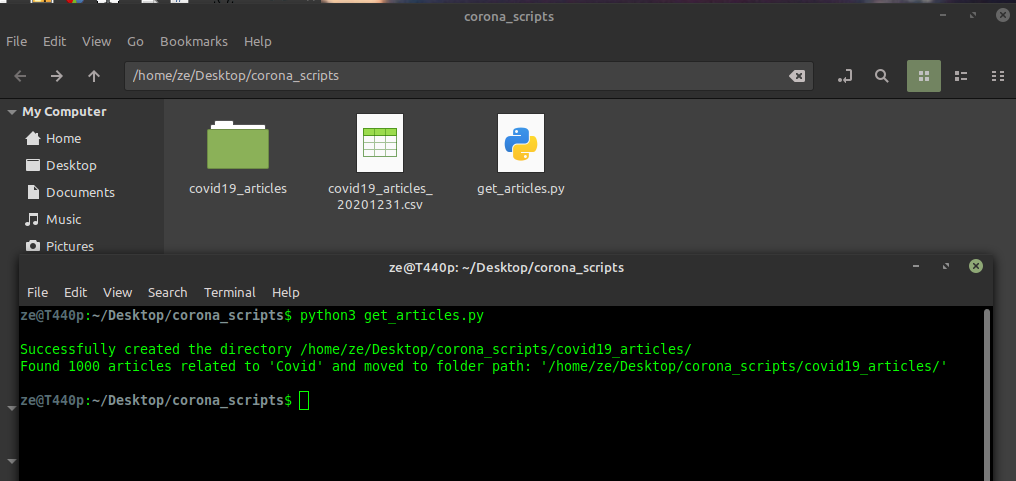
## Σχεδιασμός και υλοποίηση τμημάτων της μηχανής αναζήτησης

Χρησιμοποιήσαμε δεδομένα από το **kaggle** της μορφής

(<https://www.kaggle.com/jannalipenkova/covid19-public-media-dataset>)

Το αρχείο **covid19\_articles\_20201231.csv**, έχει τα εξής πεδία

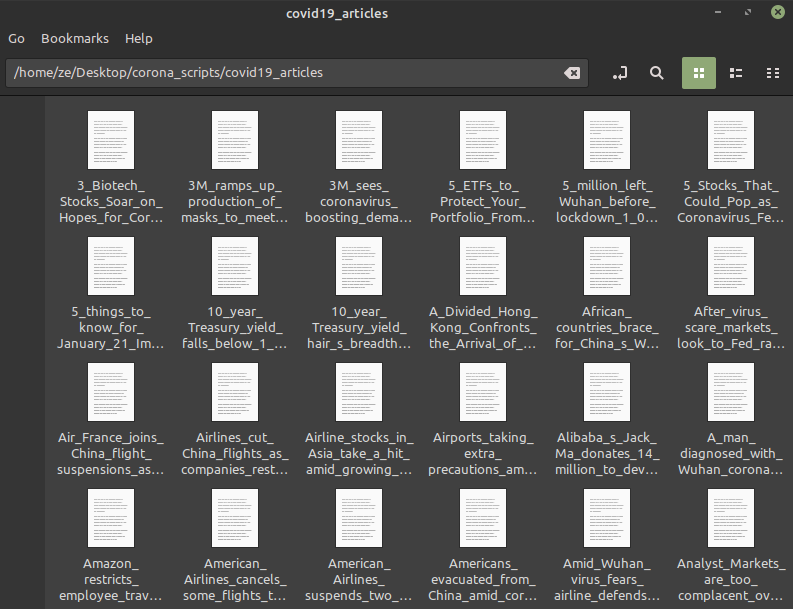
* Author
* Date
* Domain
* Title
* Url
* Content
* Topic\_area



Με το αρχείο αυτό /corona\_scripts/**get\_articles.py** διαβάζουμε κάθε γραμμή του csv και ελέγχουμε αν κάποια από τις λέξεις covid, sars, coronavirus , virus , pneumonia , flu , epidemic εμφανίζεται στον τίτλο(title). Αν εμφανίζεται, τότε αποθηκεύει όλα τα πεδία με την παρακάτω σειρά author, date, domain, title, url, topic\_area, content με new line σε ένα νέο φάκελο που δημιουργείται.

Ο φάκελος δημιουργείται αυτόματα (εάν δεν υπάρχει ήδη) και εκεί αποθηκεύονται αρχεία txt, με βάση τον τίτλο του κάθενος

πχ 3 Biotech Stocks Soar on Hopes for Coronavirus Vaccine.txt



Έχουμε συλλέξει 1000 τέτοια άρθρα, οπότε ο φάκελος covid19\_articles περιέχει:

Το κάθε αρχείο είναι χωρισμένο με βάση τα πεδία του και new line χαρακτήρες, δηλαδή

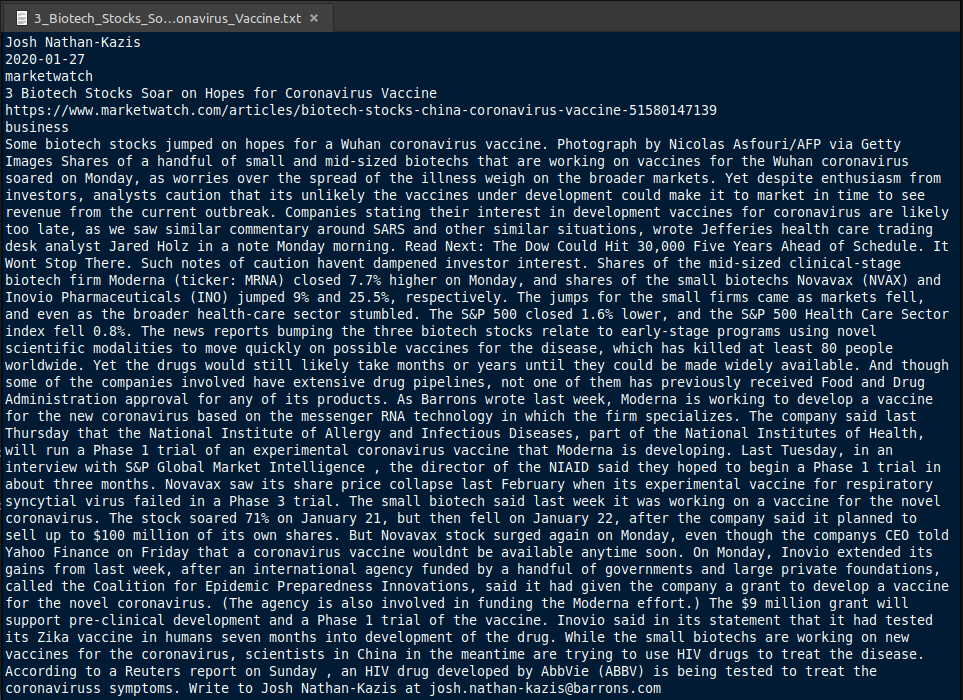
- όπου η 1η γραμμή είναι -> author

- η 2η είναι -> date

- η 3η είναι -> domain

- η 4η είναι -> title

- η 5η είναι -> topic\_area

- η 6η είναι -> content

Κατά το pre-processing αφαιρούμε από το κυρίως κείμενο (content)

special characters, new lines,

javascript code ( document.write(…); )

…

Τα υπόλοιπα πεδία δεν χρειάζονται pre-processing.

## ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΛΟΓΙΣΜΙΚΟΥ

Το μοντέλο αρχιτεκτονικής που χρησιμοποιήθηκε για τη δημιουργία της εφαρμογής είναι το Model-View-Controller (MVC). Στο μοντέλο αυτό η εφαρμογή διαιρείται σε τρία διασυνδεδεμένα μέρη ώστε να διαχωριστεί η παρουσίαση της πληροφορίας στον χρήστη από την μορφή που έχει αποθηκευτεί στο σύστημα.

Diagram

Description automatically generated

* view package : Περιέχει τις boundary classes, υπεύθυνες για την αναπαράσταση της  
  πληροφορίας που περιέχει το model, δημιουργώντας γραφική παρουσίαση στον χρήστη
* controllers package : Το controllers πακέτο περιέχει κλάσεις-controllers για τον χειρισμό των διάφορων  
  components που απαρτίζουν το UI, καθώς και των διάφορων ενεργειών που υλοποιούνται  
  κατά την αλληλεπίδραση του χρήστη με το UI.
* mainegnine package : Κεντρική business logic engine, με interface προς υλοποίηση
* searchengine package : Υποσύστημα για τη λειτουργία της μηχανής αναζήτησης.
* model package : Domain classes του συστήματος.

## Διάγραμμα UML

Diagram

Description automatically generated

# Λειτουργίες Εφαρμογής

Κατά την έναρξη της εφαρμογή, τα .txt αρχεία που βρίσκονται στον φάκελο covid19\_articles επεξεργάζονται ώστε να μετατραπούν σε αντικείμενα τύπου Article, τη θεμελιώδη κλάση που αναπαριστά ένα άρθρο με τα σχετικά πεδία του. Έπειτα η λίστα με αυτά τα αντικείμενα ‘μεταφέρεται’ στη Lucene, προκειμένου να δημιουργηθούν indexes για αυτά. Για το παραπάνω απαιτείται κάποιος χρόνος, για αυτό και καθυστερεί η εφαρμογή κατά την εκκίνηση της.

Μόλις φορτωθεί το γραφικό περιβάλλον και το βασικό παράθυρο της εφαρμογής, ο χρήστης μπορεί να πληκρολογήσει τον όρο που θα αναζητηθεί. Μέσω της mainengine που συνδέει back με front -end καλείται η μέθοδος searchForDocument με όρισμα τον παραπάνω όρο, αναζητώντας στα πεδία «TEXT» και «TITLE» για τον όρο αυτό. Αν βρεθεί, επιστρέφεται μια λίστα με τα αντικείμενα τύπου Article που τον περιέχουν είτε στον τίτλο είτε στο κείμενο.

Για την παραπάνω λειτουργία, γίνεται η χρήση του StandardAnalyzer ο οποίος μετατρέπει σε lowercase τα token και αφαιρεί κοινές και σημεία στίξης, αν αυτά υπάρχουν.

Τα αποτελέσματα εμφανίζονται στο 2ο παράθυρο της εφαρμογής ανα 10, με τον χρήστη να έχει τη δυνατότητα να δει τα 10 επόμενα ή/και 10 προηγούμενα με τη χρήση των πλήκτρων *Next page* και *Previous Page* αντίστοιχα. Το κάθε αποτέλεσμα εμφανίζει πληροφορίες για τον τίτλο, συγγραφέα, ημερομηνία και κατηγορία άρθρου ενώ επιλέγοντας το, εμφανίζεται και το κείμενο του. Τέλος ο χρήστης μπορεί να διατάξει τα άρθρα που βρέθηκαν με βάση την ημερομηνία συγγραφής τους, είτε κατά αύξουσα είτε φθίνουσα σειρά.

## Screenshots Εφαρμογής