



# **CORD-19 Application**

Developed by Kyriakos Tsaliagkos,  
for ECE NTUA

# CORD-19 Application

Η προτεινόμενη εφαρμογή είναι η παρουσίαση των top 5 πηγών άρθρων, σύμφωνα με τα στοιχεία του CORD-19 dataset, με βάση των αριθμό δημοσιευμένων άρθρων ανά πηγή, καθώς και το μέσο όρο και την τυπική απόκλιση του αριθμού των συγγραφέων που έχουν συμβάλει σε κάθε άρθρο, επίσης ανά πηγή.



# CORD-19 Application

Οι τεχνολογίες που θα χρησιμοποιηθούν είναι

- Node.js + Express.js για το back-end
- ReactJS για το front-end
- MongoDB για τη βάση δεδομένων
- Docker για το deployment όλων των παραπάνω

# CORD-19 Application

- Node.js

“A JavaScript runtime built on Chrome's V8 JavaScript engine”

- Express.js

“A fast, unopinionated, minimalist web framework for Node.js”

Ο συνδυασμός αυτών των δύο μας επιτρέπει να γράψουμε JavaScript εκτός του web browser, στο server-side, και να αναπτύξουμε πολύ γρήγορα high-performance εφαρμογές.

# CORD-19 Application

- ReactJS

“A JavaScript library for building user interfaces”

Βιβλιοθήκη για δημιουργία dynamic web/mobile application, με βασικό χαρακτηριστικό το rendering των components στο DOM-Document Object Model.

# CORD-19 Application

- MongoDB

“A general purpose, document-based, distributed database built for modern application developers and for the cloud era.”

Μια εύχρηστη, high-performance NoSQL βάση δεδομένων με πολύ καλό integration με τη Node.js, λόγω και της χρήσης JSON αρχείων.

# CORD-19 Application

- Docker

“Docker is a set of platform as a service (PaaS) products that use OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels. All containers are run by a single operating system kernel and therefore use fewer resources than virtual machines”

Διευκολύνει το deployment της εφαρμογής και ελαχιστοποιεί τα conflicts μεταξύ διαφορετικών version από software components και το “footprint” της εφαρμογής στο host μηχανήμα.



# **CORD-19 Application**

Οδηγίες εγκατάστασης:





# CORD-19 Application

Η χρήση Docker απλοποιεί πολύ τα πράγματα, καθώς δεν χρειάζεται να έχουμε εγκατεστημένο τίποτα άλλο στο host machine μας, παρά μόνο αυτό.

Install docker:

<https://docs.docker.com/get-docker/>

# CORD-19 Application

Αφού έχουμε εγκαταστήσει το docker, κατεβάζουμε το πιο πρόσφατο release\_file του CORD-19 dataset. Πρόκειται για ένα tar.gz αρχείο, το οποίο και πρέπει να κάνουμε extract, μιας και μας ενδιαφέρει το αρχείο metadata.csv που περιέχει.

Download CORD-19:

[https://ai2-semanticsscholar-cord-19.s3-us-west-2.amazonaws.com/historical\\_releases.html](https://ai2-semanticsscholar-cord-19.s3-us-west-2.amazonaws.com/historical_releases.html)

# CORD-19 Application

Στη συνέχεια, κατεβάζουμε το source code της εφαρμογής, είτε ως zip φάκελο που θα κάνουμε extract, ή κάνουμε clone το repository.

- Zip file:

<https://github.com/KyriakosTsalia/ntua-appathon-project/archive/master.zip>

- Git clone:

<https://github.com/KyriakosTsalia/ntua-appathon-project.git>



# CORD-19 Application

Μετά, απαραίτητο είναι να μεταφέρουμε το αρχείο `metadata.csv` στο directory `ntua-appathon-project/database`.

Από εκεί, τα δεδομένα θα εισαχθούν αυτόματα στη MongoDB βάση δεδομένων με το `deploy` της εφαρμογής.

# CORD-19 Application

Τέλος, και αφού είμαστε στο root directory του project, δηλαδή στο `ntua-appathon-project/database`, εκτελούμε την εντολή `docker-compose up`, για να κάνουμε deploy την εφαρμογή και να σηκώσουμε τους containers.

Πλέον η εφαρμογή μας είναι online και μπορούμε να δούμε το UI της στο `localhost:3000/`

# CORD-19 Application

Συγκεκριμένα, για host machine με Linux OS και πιο πρόσφατο release\_file αυτό της 18-8-2020:

```
mkdir cord19_app && cd cord19_app/  
  
wget -c  
https://ai2-semantic scholar-cord-19.s3-us-west-2.amazonaws.com/historical_releases  
/cord-19_2020-08-18.tar.gz -P .  
  
tar -zxvf cord-19_2020-08-18.tar.gz  
  
cp ./2020-08-18/metadata.csv .  
  
git clone https://github.com/KyriakosTsalia/ntua-appathon-project.git  
cd ./ntua-appathon-project  
  
mv ../metadata.csv ./database  
  
sudo docker-compose up --build
```



# CORD-19 Application

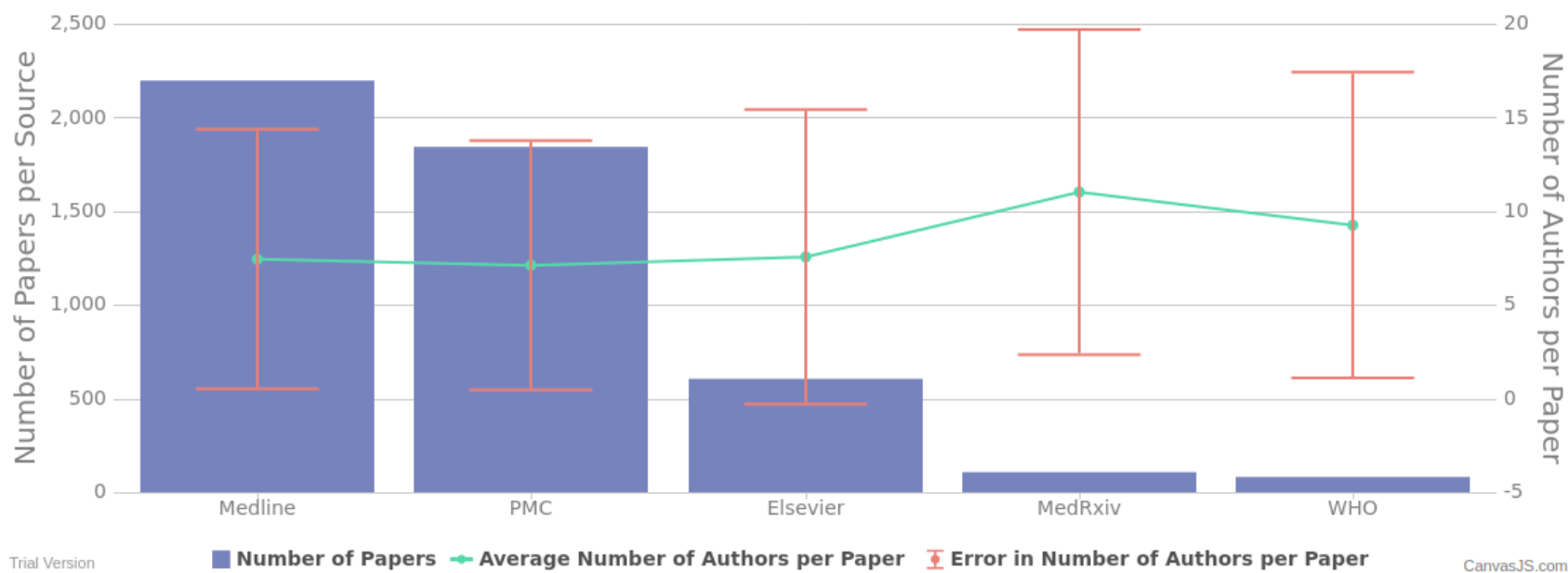
Μερικά test cases:

# CORD-19 Application

Disease: cancer

Min\_publish\_time: 22-2-2018

These are the top scientific sources regarding "cancer" since 2018-02-22



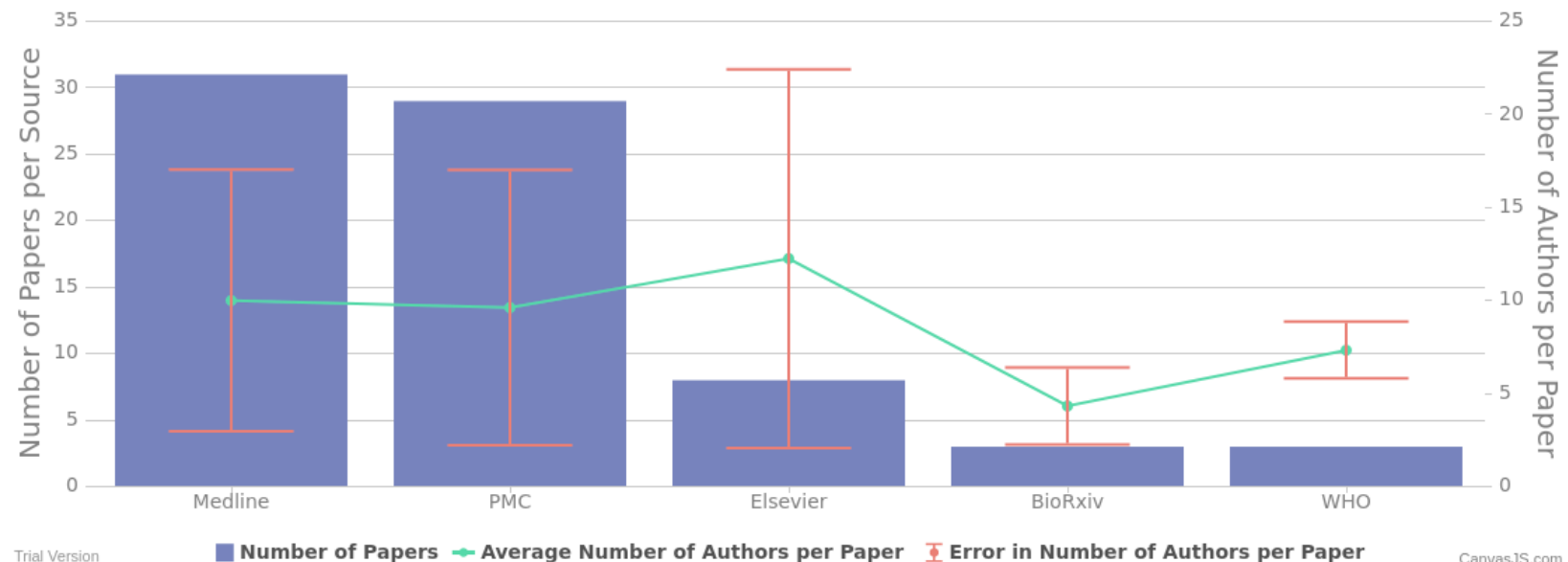


# CORD-19 Application

Disease: kidney failure

Min\_publish\_time: 19-6-2019

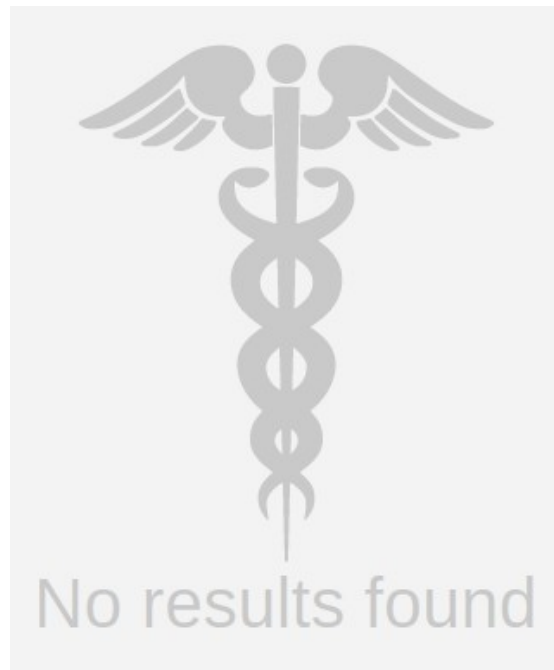
These are the top scientific sources regarding "kidney failure" since 2019-06-19



# CORD-19 Application

Disease: someStuff / Empty

Min\_publish\_time: tooLateDate / Empty



# CORD-19 Application

GET request σε ένα invalid route,  
π.χ. localhost:3000/error Empty



# CORD-19 Application

Για να κατεβάσουμε την εφαρμογή, αρκεί και πάλι στο root directory του project μας να εκτελέσουμε την εντολή `docker-compose down`.

Συγκεκριμένα, για host machine με Linux OS:

```
sudo docker-compose down
```

Οι containers σταματούν και διαγράφονται και λόγω του docker, το host machine μας είναι στην ίδια κατάσταση με πριν, δίχως περιττά libraries/packages/frameworks/databases.



# **CORD-19 Application**

Ευχαριστώ για την  
προσοχή σας!