

ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ
ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΕΦΑΡΜΟΓΩΝ
ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ ΓΙΑ
ΤΟ ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2020-2021

ΟΜΑΔΑ 2672_2820

ΣΠΥΡΙΔΩΝΟΣ ΒΑΣΙΛΕΙΟΣ, 2820

ΓΚΟΤΖΑΓΙΑΝΝΗΣ ΒΑΣΙΛΕΙΟΣ, 2672

ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ

ΑΠΡΙΛΙΟΣ 2021

ΙΣΤΟΡΙΚΟ ΠΡΟΗΓΟΥΜΕΝΩΝ ΕΚΔΟΣΕΩΝ

Ημερομηνία	Έκδοση	Περιγραφή	Συγγραφέας
2021/04/14	1.0	Δημιουργία περιβάλλοντος αλληλεπίδρασης	Σπυρίδωνος Βασίλειος, Γκοτζαγιάννης Βασίλειος

Το κείμενο συμπληρώνεται προοδευτικά, όπως προχωρείτε στις φάσεις του Project.

1 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Πίνακας 1: **years**

Περιέχει ένα primary key id τύπου INT και το πεδίο year_name τύπου VARCHAR.

Πίνακας 2: **measures**

Περιέχει ένα primary key id τύπου INT και το πεδίο measure_name τύπου VARCHAR.

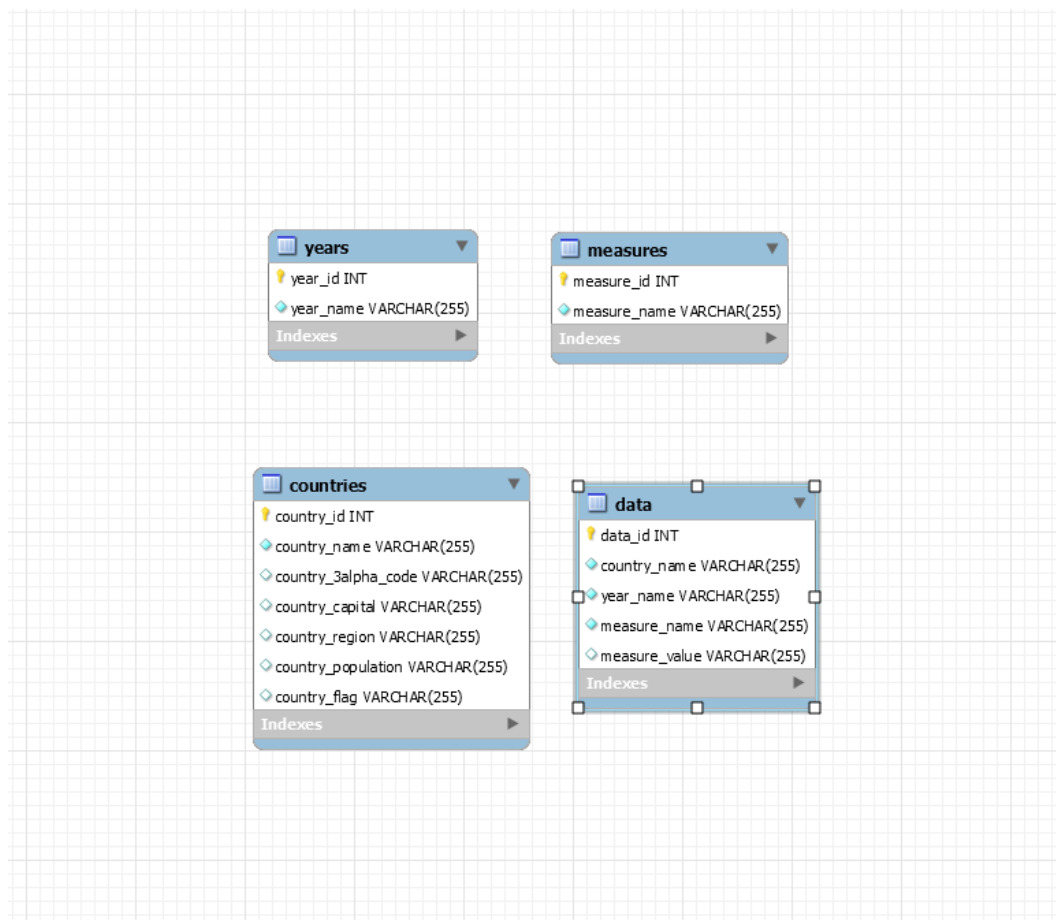
Πίνακας 3: **data**

Περιέχει ένα primary key id τύπου INT, το πεδίο year_name τύπου VARCHAR, το πεδίο country_name τύπου VARCHAR, το πεδίο measure_name τύπου VARCHAR και το πεδίο measure_value τύπου VARCHAR.

Πίνακας 4: **countries**

Περιέχει ένα primary key id τύπου INT, το πεδίο country_name, country_3alpha_code, country_capital, country_region, country_population, country_flag.

1.1 ΣΧΕΣΙΑΚΟ ΣΧΗΜΑ ΣΕ ΛΟΓΙΚΟ ΕΠΙΠΕΔΟ



1.2 ΣΧΕΣΙΑΚΟ ΣΧΗΜΑ ΣΕ ΦΥΣΙΚΟ ΕΠΙΠΕΔΟ

Όταν θα έχετε στήσει και ρυθμίσει τη βάση δεδομένων σας, εδώ καταγράφονται και οι ρυθμίσεις σε φυσικό επίπεδο. Ενδεικτικά:

1.2.1 ΡΥΘΜΙΣΗ ΤΩΝ ΠΑΡΑΜΕΤΡΩΝ ΤΟΥ DBMS

(στο 2^ο version)

1.2.2 ΡΥΘΜΙΣΗ ΤΟΥ ΦΥΣΙΚΟΥ ΣΧΗΜΑΤΟΣ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

1.2.3 ΡΥΘΜΙΣΗ ΑΣΦΑΛΕΙΑΣ

2 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΛΟΓΙΣΜΙΚΟΥ

2.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΔΟΜΗ ETL

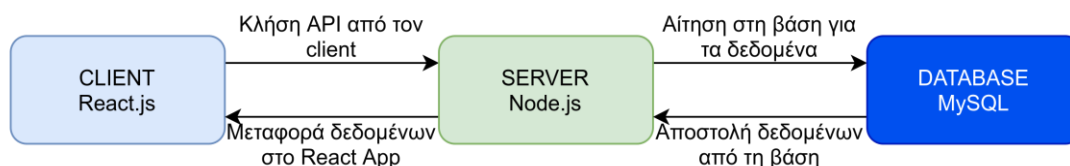
Υπάρχουν 3 αρχεία: `export_data.py`, `transform_data.py`, `load_data.py`. Τα αρχεία που τρέχουμε είναι τα `transform_data` και `load_data`.

Τα modules που χρειάζονται είναι: `csv`, `requests`:

- `pip install csv`
- `pip install requests`

Πρώτα, τρέχουμε το `transform_data` (`python transform_data.py`). Πριν το τρέξουμε, θέτουμε στη global μεταβλητή, `output_path`, το path στο φάκελο της βάσης που θα φορτωθούν τα δεδομένα, που βρίσκεται στο φάκελο Data του Mysql server. Όταν τελειώσει το script (κάνει κάποια calls σε api με χώρες για να μαζέψει κάποιες πληροφορίες), τρέχουμε το script `load_data` (`python load_data.py`). Επίσης, πριν το τρέξουμε αλλάζουμε την global μεταβλητή `db` στο όνομα της βάσης που έχουμε φτιάξει. Το script είναι υπεύθυνο, να φτιάξει και τα tables, οπότε αν το τρέξουμε 2^η φορά πρέπει να βγάλουμε τον κώδικα που φτιάχνει τα tables.

2.2 ΔΙΑΓΡΑΜΜΑΤΑ ΠΑΚΕΤΩΝ / ΥΠΟΣΥΣΤΗΜΑΤΩΝ ΚΕΝΤΡΙΚΗΣ



ΕΦΑΡΜΟΓΗΣ

Το σύστημα χωρίζεται σε δύο κύρια τμήματα, τον διακομιστή server (backend API) και τον client (app).

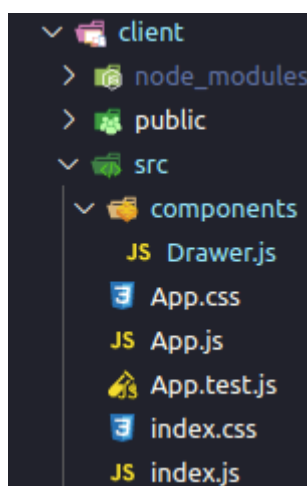
Η δουλειά του server, ο οποίος είναι συνδεδεμένος με τη βάση δεδομένων, είναι να δέχεται αιτήσεις από τον client και σαν απάντηση να παρέχει τα δεδομένα που ζητήθηκαν.

Η δουλειά του πελάτη, είναι να παρέχει το περιβάλλον αλληλεπίδρασης στον χρήστη ώστε να μπορεί να επιλέξει τα δεδομένα για την οπτικοποίηση.

2.3 ΔΙΑΓΡΑΜΜΑ(ΤΑ) ΚΛΑΣΕΩΝ ΚΕΝΤΡΙΚΗΣ ΕΦΑΡΜΟΓΗΣ

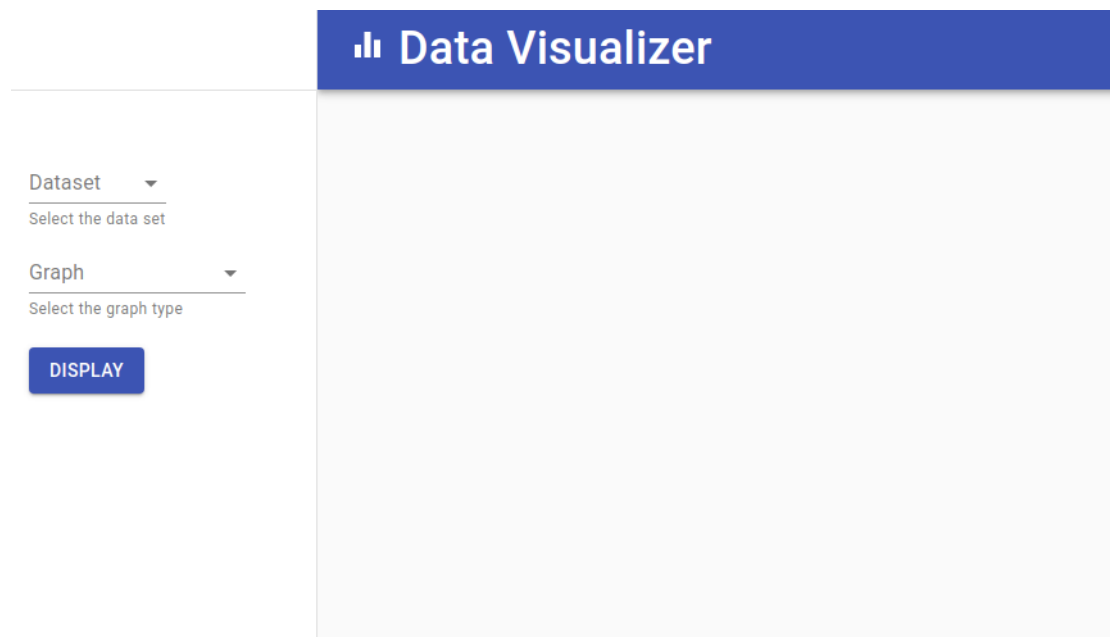
Η ανάπτυξη του server έγινε με Node.js και τη βιβλιοθήκη Express.js. Το αρχείο index.js περιέχει κώδικα για την κατασκευή του server και για τη σύνδεση με τη βάση δεδομένων, καθώς και συναρτήσεις που εκτελούνται ανάλογα με την κλήση του API. Οι συναρτήσεις κατασκευάζουν MySQL queries για τη βάση, λαμβάνουν τα αποτελέσματα και τα στέλνουν στον client.

Η ανάπτυξη του client έγινε με το πλαίσιο React.js χρησιμοποιώντας τη βιβλιοθήκη d3.js.



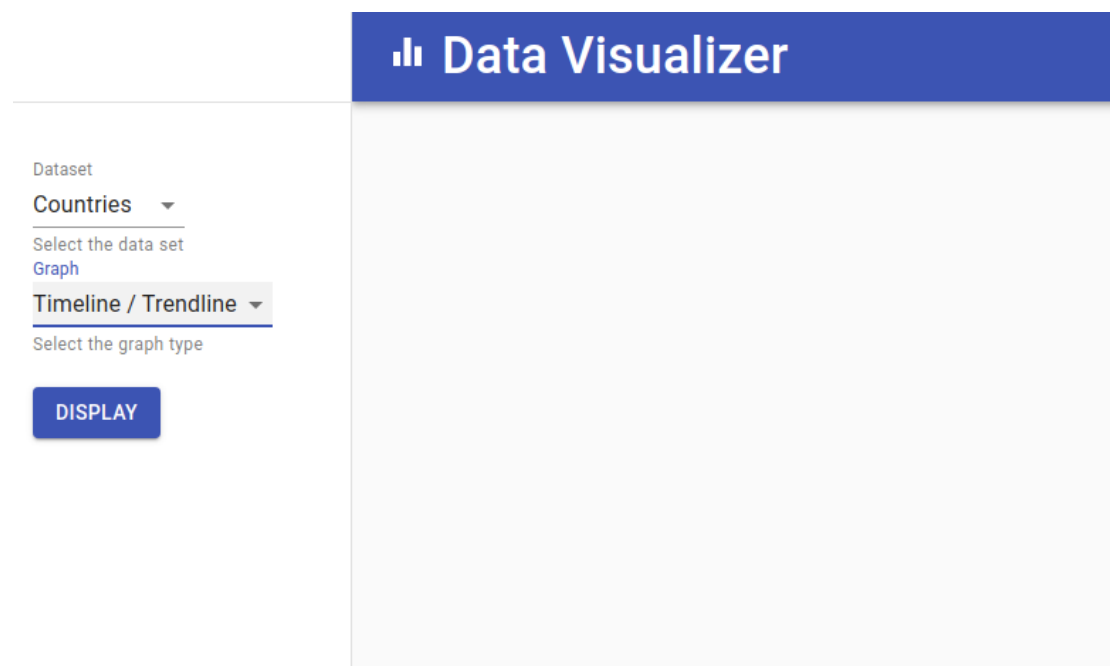
Το βασικό component είναι το Drawer.js το οποίο υλοποιεί όλο το περιβάλλον αλληλεπίδρασης μέχρι στιγμής (αργότερα θα “σπάσει” σε μικρότερα components).

Περιέχει μία φόρμα στην οποία ο χρήστης μπορεί να επιλέξει το dataset που θέλει και τον τύπο του γραφήματος. Μόλις ο χρήστης επιλέξει “DISPLAY”, το app κάνει αίτηση στον server και μόλις λάβει την απάντηση με τα δεδομένα, τα οπτικοποιεί χρησιμοποιώντας τη βιβλιοθήκη d3.

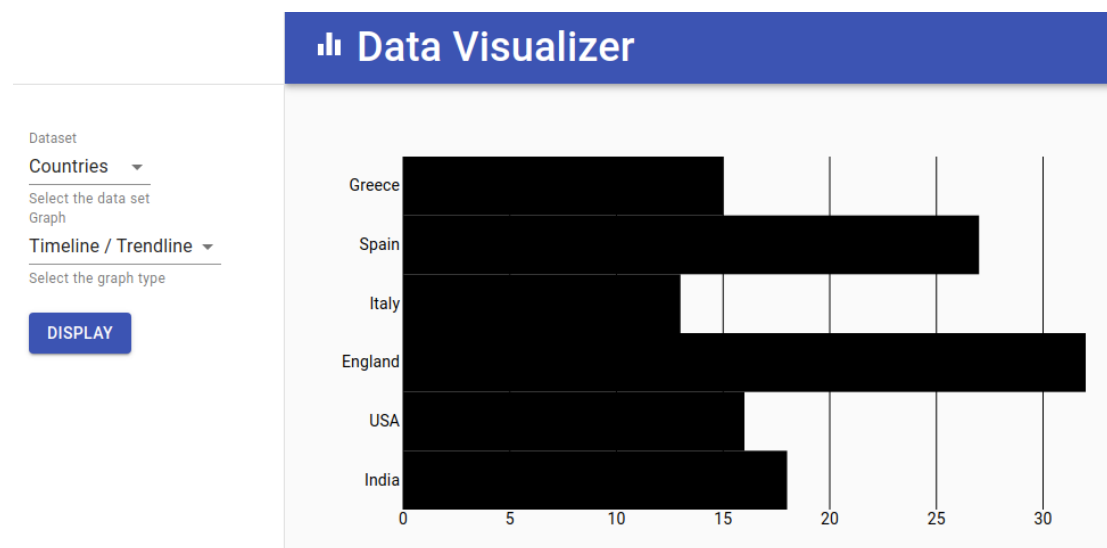


3 ΥΠΟΔΕΙΓΜΑΤΑ ΕΡΩΤΗΣΕΩΝ ΚΑΙ ΑΠΑΝΤΗΣΕΩΝ

Εικόνα 1: Αρχική εμφάνιση



Εικόνα 2: Επιλογή dataset και τύπου γραφήματος



Εικόνα 3: Εμφάνιση γραφήματος

4 ΤΕΚΜΗΡΙΩΣΗ ΚΑΙ ΛΟΙΠΑ ΣΧΟΛΙΑ

Ότι άλλα σχόλια υπάρχουν