

# ΠΡΟΧΩΡΗΜΈΝΑ ΘΈΜΑΤΑ ΤΕΧΝΟΛΟΓΊΑΣ ΚΑΙ ΕΦΑΡΜΟΓΏΝ ΒΆΣΕΩΝ ΔΕΔΟΜΈΝΩΝ

---

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΕΡΓΑΣΙΑ ΓΙΑ ΤΟ  
ΑΚΑΔΗΜΑΪΚΟ ΈΤΟΣ 2019-2020

---

---

ΟΜΑΔΑ

ΕΥΤΥΧΙΑ ΚΙΑΦΑ, ΑΜ 3003

ΓΡΗΓΟΡΙΑ ΝΙΚΗΤΑ, ΑΜ 3048

ΕΙΡΗΝΗ ΜΟΥΣΕΛΛΗ, ΑΜ 3031

---

ΤΕΛΙΚΗ ΑΝΑΦΟΡΑ

ΜΑΪΟΣ 2020

ΙΣΤΟΡΙΚΟ ΠΡΟΗΓΟΥΜΕΝΩΝ ΕΚΔΟΣΕΩΝ

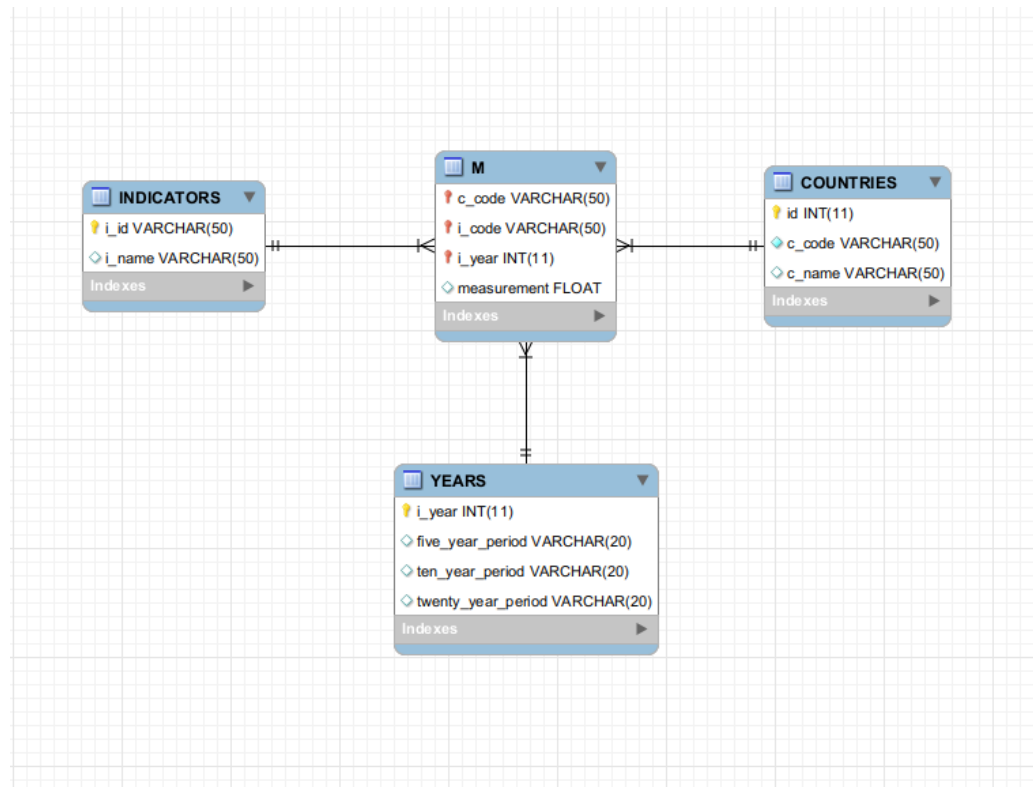
Ημερομηνία	Έκδοση	Περιγραφή	Συγγραφέας
2020/05/28	1.1	Συμπλήρωση 1ης ενότητας	Ειρήνη Μουσελλή
2020/05/29	1.2	Συμπλήρωση 2ης ενότητας	Ευτυχία Κιάφα
2020/06/03	1.3	Συμπλήρωση 3ης ενότητας	Γρηγορία Νικήτα

## 1 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Εδώ περιγράφονται τα σχήματα της βάσης δεδομένων που χρησιμοποιούνται στο project.

### 1.1 ΣΧΕΣΙΑΚΟ ΣΧΗΜΑ ΣΕ ΛΟΓΙΚΟ ΕΠΙΠΕΔΟ

#### WORKBENCH SCREENSHOT



#### ΕΝΤΟΛΕΣ ΚΑΤΑΣΚΕΥΗΣ ΠΙΝΑΚΩΝ

##### COUNTRIES:

```
CREATE TABLE IF NOT EXISTS COUNTRIES (  
    ID INT,  
    C_CODE VARCHAR(50) UNIQUE NOT NULL,  
    C_NAME VARCHAR(50),  
    KEY (C_CODE),  
    PRIMARY KEY (ID)  
);
```

INDICATORS:

```
CREATE TABLE IF NOT EXISTS INDICATORS (  
    I_ID VARCHAR(50),  
    I_NAME VARCHAR(50),  
    PRIMARY KEY (I_ID)  
);
```

YEARS:

```
CREATE TABLE IF NOT EXISTS YEARS (  
    I_YEAR INT,  
    FIVE_YEAR_PERIOD VARCHAR(20),  
    TEN_YEAR_PERIOD VARCHAR(20),  
    TWENTY_YEAR_PERIOD VARCHAR(20),  
    PRIMARY KEY (I_YEAR)  
);
```

M:

```
CREATE TABLE IF NOT EXISTS M (  
    C_CODE VARCHAR(50),  
    I_CODE VARCHAR(50),  
    I_YEAR INT,  
    MEASUREMENT FLOAT,  
    PRIMARY KEY (C_CODE , I_CODE , I_YEAR),  
    FOREIGN KEY (C_CODE)  
        REFERENCES COUNTRIES (C_CODE)  
        ON UPDATE CASCADE ON DELETE CASCADE,  
    FOREIGN KEY (I_CODE)  
        REFERENCES INDICATORS (I_ID)  
        ON UPDATE CASCADE ON DELETE CASCADE,  
    FOREIGN KEY (I_YEAR)  
        REFERENCES YEARS (I_YEAR)  
        ON UPDATE CASCADE ON DELETE CASCADE  
);
```

ΣΗΜΕΙΩΣΗ:

Τα παραπάνω βρίσκονται στο αρχείο load\_tables.sql.

## 1.2 ΣΧΕΣΙΑΚΟ ΣΧΗΜΑ ΣΕ ΦΥΣΙΚΟ ΕΠΙΠΕΔΟ

---

Εδώ καταγράφονται και οι ρυθμίσεις σε φυσικό επίπεδο.

### 1.2.1 ΡΥΘΜΙΣΗ ΤΩΝ ΠΑΡΑΜΕΤΡΩΝ ΤΟΥ DBMS

---

Storage engine: InnoDB engine

memory allocation:

`innodb_buffer_pool_chunk_size = 128M`

`innodb_buffer_pool_size = 2432M`

## 2 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΛΟΓΙΣΜΙΚΟΎ

---

### 2.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΚΑΙ ΔΟΜΗ ETL

---

Τα αρχεία με τις απαραίτητες πληροφορίες όπως δείκτες, χώρες, έτη κ.α. τα οποία προορίζονταν να προωθηθούν στην βάση προέκυψαν χειροκίνητα αλλά και με script. Τα αρχεία Countries, Years και Indicators λόγω της μικρής δυσκολίας δημιουργήθηκαν χειροκίνητα και περιείχαν τα εξής:

- Countries(C\_Code, C\_name): όπου C\_CODE είναι ο κωδικός χώρας και C\_NAME είναι το όνομα της.

-Years(YEAR, 5YRPERIOD, 10YRPERIOD, 20YRPERIOD):

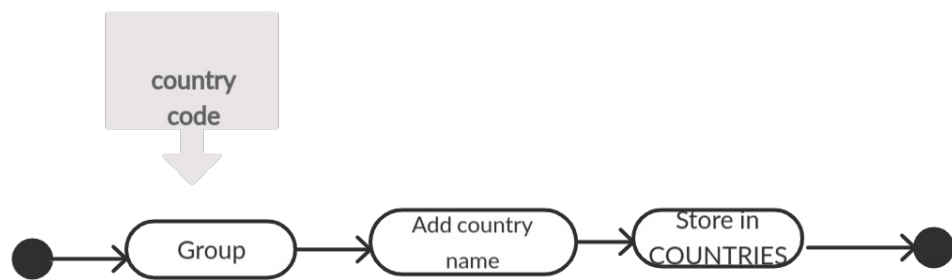
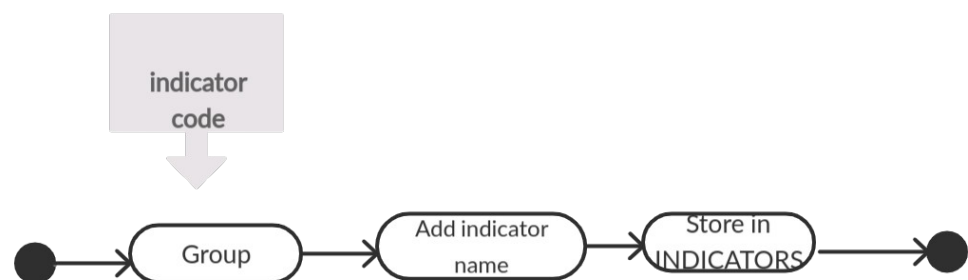
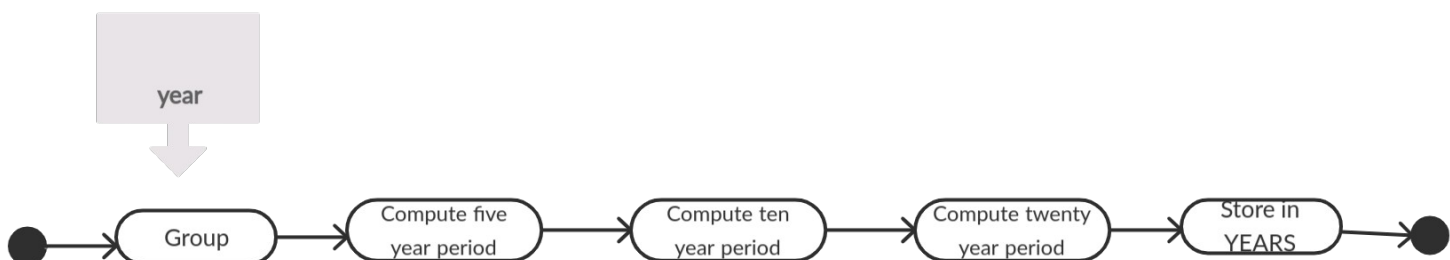
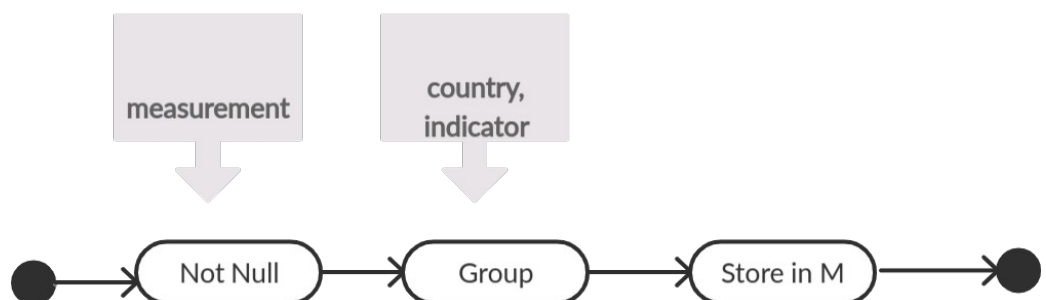
όπου YEAR είναι η χρονιά, και μετα αντίστοιχα 5YRPERIOD, 10YRPERIOD, 20YRPERIOD είναι χρονικές περίοδοι που αργότερα θα χρησιμοποιήσει ο χρήστης στα γραφήματα του.

-INDICATORS(I\_ID, I\_NAME):

Όπου I\_ID είναι ο κωδικός δείκτη και I\_NAME είναι το όνομα του.

-M(C\_CODE, I\_CODE, YEAR, MEASUREMENTS):

Το αρχείο M δημιουργήθηκε ώστε να περιέχει συγκεντρωτικά πληροφορίες από όλες τις χώρες, τους δείκτες τους και τις χρονιές. Χρησιμοποιήθηκε script για την παραγωγή του, σε γλώσσα Python. Στο script δημιουργείται ένας πίνακας M, με στοιχεία που αντλεί από τα csv files των χωρών που έχουμε επιλέξει.

UML BASED ETL DIAGRAMSCOUNTRIES:INDICATORS:YEARS:M:



Στο διάγραμμα απεικονίζεται το εσωτερικό της εφαρμογής όπως διαμορφώνεται από την παρουσία βιβλιοθηκών και των αρχείων που την απαρτίζουν. Από το διαγράμμα λείπουν οι φάκελοι `node_modules` που εκτίνασσουν την πολυπλοκότητα του γράφου σε ό,τι αφορά την κατανόηση.

### **Ανάλυση ροής λειτουργίας τους συστήματος:**

Αρχικά γίνεται η ενεργοποίηση του server, με την εντολή: `node server.js` (<http://localhost:5000/>), ο οποίος ακούει τα αιτήματα του client (<http://localhost:3000/>), που ενεργοποιείται αμέσως μετά με την εντολή: `npm start`, και τα προωθεί στον client με την βοήθεια της `mysql`, για την οποία έχουν διαμορφωθεί κατάλληλα δυναμικά queries ώστε να εξασφαλιστεί η αρτιότητα επικοινωνίας με την βάση. Η `node.js` διαχειρίζεται το back-end. Είναι υπεύθυνη με ένα thread να μεταφέρει μηνύματα στην βάση, να δρά παράλληλα με την αναμονή της για την απάντηση από την βάση, προωθεί μέσω του framework `express` τα αποτελέσματα, ώστε να μορφοποιηθούν κατάλληλα και να αποσταλούν στην διεπαφή. Διαμορφώνεται δηλαδή η λειτουργία ενός API που σερβίρει την ζητούμενη πληροφορία.

Η επικοινωνία μεταξύ client και server επιτυγχάνεται με την χρήση `cors` (Cross-Origin Resource Sharing). Ο μηχανισμός αυτός, χρησιμοποιεί πρόσθετες κεφαλίδες HTTP για να πει στους περιηγητές, να δώσουν σε μια εφαρμογή ιστού που λειτουργεί σε μία προέλευση, πρόσβαση σε επιλεγμένους πόρους από διαφορετική προέλευση. Για αυτό το σκοπό χρησιμοποιούμε και `proxy server` ή αλλιώς διακομιστή μεσολάβησης, ο οποίος δρά σαν μεσάζων και βρίσκεται στην διεύθυνση <http://localhost:5000/>.

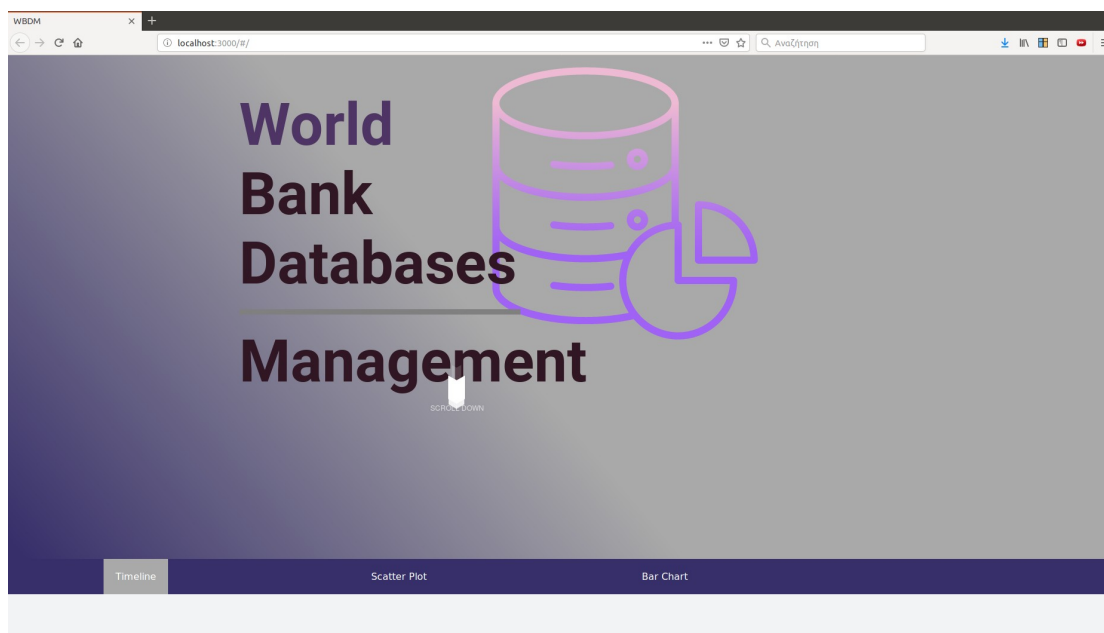
Αφού ξεκινήσει η αλληλεπίδραση με τον χρήστη η διαδικασία είναι η εξής: Ο χρήστης επιλέγει από το μενού τι χρειάζεται να δει, με το `Submit`, δημιουργεί ο client ένα αίτημα στον server, με πέρασμα πληροφοριών μέσω των `options`, με `fetch`. Τα `options` περιέχουν την μέθοδο προώθησης της απάντησης, ορισμό `headers` και μετατροπή των `data` σε `string`. Ο server παίρνει το αίτημα, δημιουργεί `queries`, συνδέεται με την βάση και παίρνει στην απάντηση. Η απάντηση προωθείται στον client μέσω `fetch`, ο οποίος γεμίζει με την απάντηση τον πίνακα `data` και μορφοποιεί τα δεδομένα ώστε να αποδοθούν σαν γραφήματα. Αμέσως μετά γίνεται `render` και οι πληροφορίες είναι πια εμφανείς στον χρήστη διαγραμματικά.



### 3 ΥΠΟΔΕΙΓΜΑΤΑ ΕΡΩΤΗΣΕΩΝ ΚΑΙ ΑΠΑΝΤΗΣΕΩΝ

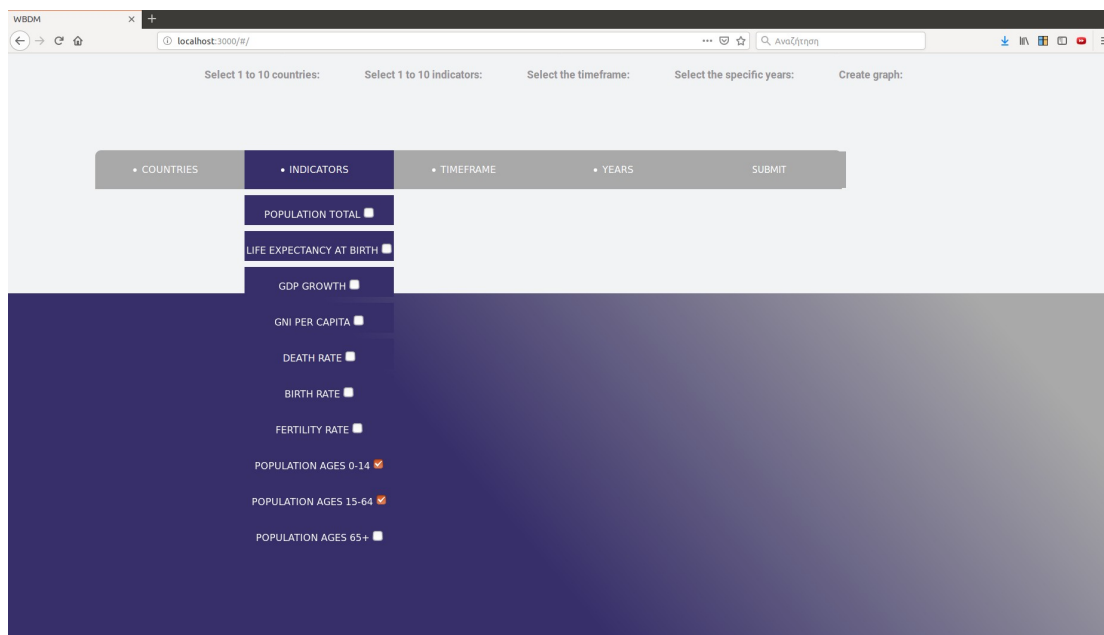
**Παράδειγμα χρήσης για την ερώτηση:**

**Πληθυσμός της Ελλάδας ηλικίας 0-14 και 15-64 από το 1970 μέχρι το 2000 ανά χρονιά.**



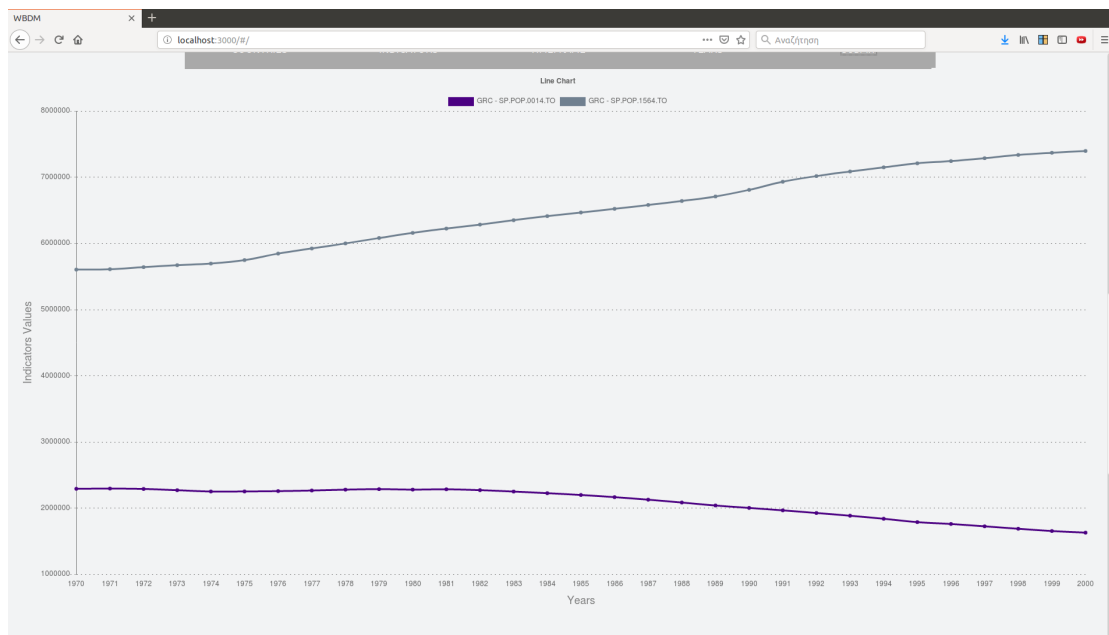
#### 3.1 Αρχική σελίδα.

Στην Αρχική σελίδα γίνεται επιλογή τύπου γραφήματος: Timeline, ScatterPlot ή BarCart.

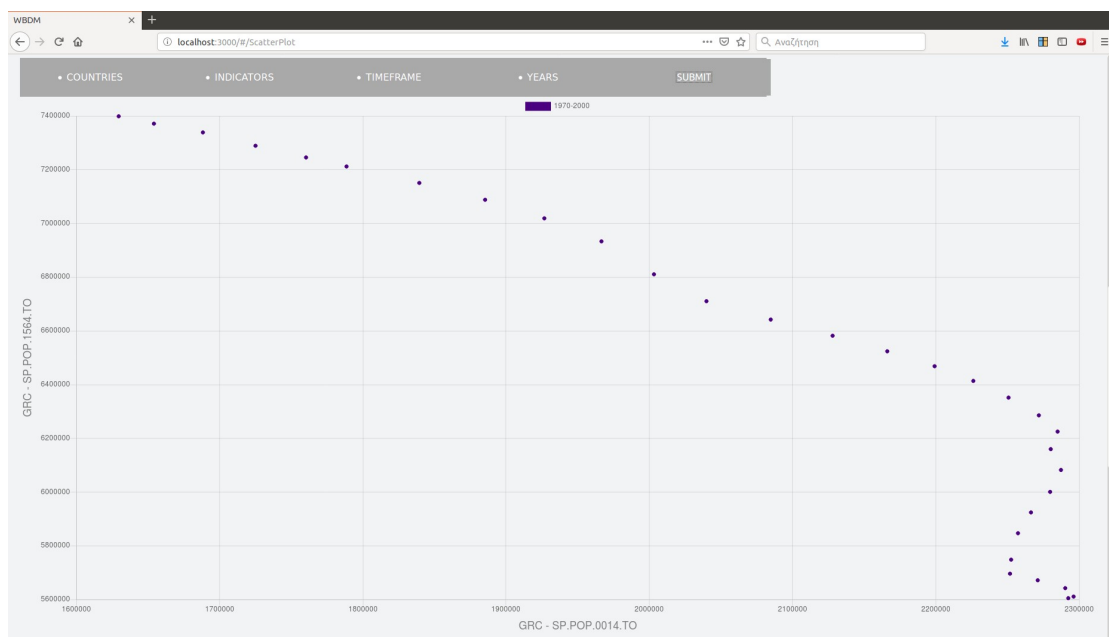


#### 3.2 Επιλογή δεικτών για τη δημιουργία γραφημάτων.

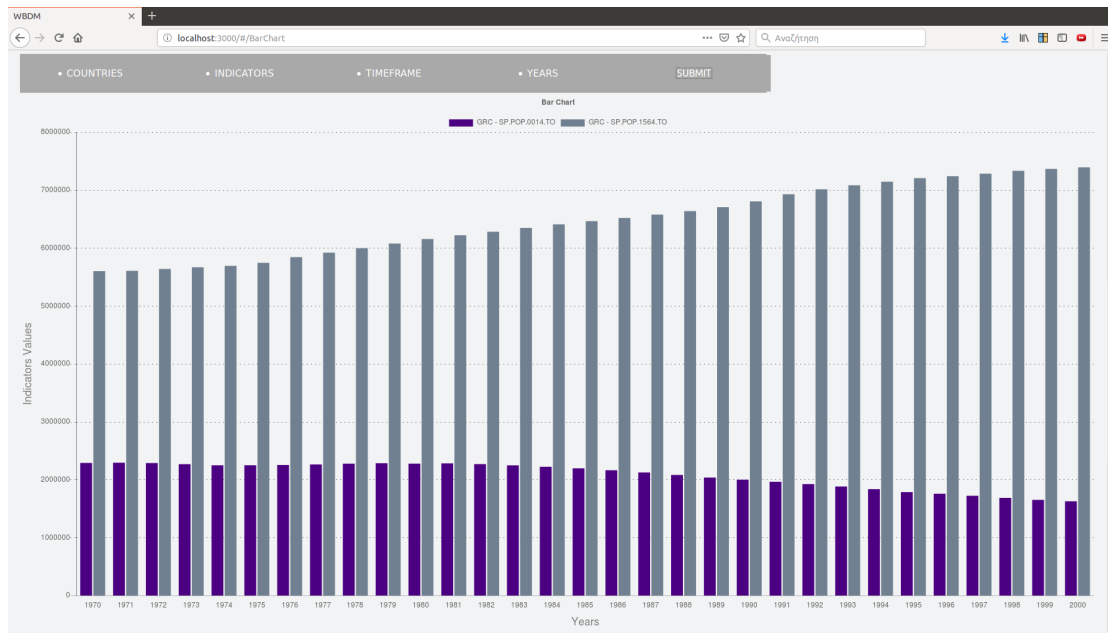
Επιλέγονται χώρες, δείκτες, timeframe και εύρος χρονολογίας ανάλογα με τις οδηγίες που βρίσκονται πάνω από τις επιλογές για κάθε τύπο γραφήματος και επιλέγεται submit για την εμφάνιση του γραφήματος.



3.3. Timeline: Greece - Population ages 0-14 & Population ages 15-64, 1970-2000.



3.4. ScatterPlot: Greece - Population ages 0-14 & Population ages 15-64, 1970-2000.



3.5. BarChart: Greece – Population ages 0-14 & Population ages 15-64, 1970-2000.