



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟ-
ΓΙΣΤΩΝ**

**ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑ-
ΣΕΩΝ**

**Μελέτη και Σχεδιασμός Συστήματος Συλλογής, Εναρμόνισης και
Αποθήκευσης Δεδομένων Καταστροφών**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Ευγένι Δούφα

Επιβλέπων : Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2022



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ

ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑ-
ΤΩΝ ΑΠΟΦΑΣΕΩΝ

**Μελέτη και Σχεδιασμός Συστήματος Συλλογής, Εναρμόνισης και
Αποθήκευσης Δεδομένων Καταστροφών**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Ευγένι Δούφα

Επιβλέπων : Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 12^η Οκτωβρίου 2022.

.....
Δημήτριος Ασκούνης Ιωάννης Ψαρράς Χρυσόστομος Δούκας
Καθηγητής Ε.Μ.Π. Καθηγητής Ε.Μ.Π. Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2022

.....
Ευγκένη Δούφας

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

© 2022 – All rights reserved

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Αντικείμενο της παρούσας διπλωματικής αποτελεί μελέτη των δεδομένων που διατίθεται σε ποικίλες πηγές και βάσεις φυσικών καταστροφών και των προκληθέντων συνεπειών τους. Υπάρχουν εκατοντάδες βάσεις δεδομένων καταστροφών, οι οποίες όμως δεν ακολουθούν ενιαία πρότυπα μορφοποίησης των δεδομένων τους. Επομένως, καθίσταται εξαιρετικά δύσκολη η συνολική επισκόπηση και επεξεργασία των δεδομένων, η οποία είναι απαραίτητη για την μελέτη, πρόβλεψη και εξαγωγή συμπερασμάτων. Προκύπτει η ανάγκη συλλογής, εναρμόνισης και αποθήκευσης των δεδομένων φυσικών καταστροφών ακολουθώντας μια ενιαία μοντελοποίηση. Για τον σκοπό αυτόν σχεδιάζεται μοντέλο σχεσιακής βάσης, ικανό να φιλοξενήσει δεδομένα ποικίλων φυσικών καταστροφών, αποθηκεύοντας τα κυριότερα στοιχεία τους. Είναι απαραίτητη η εναρμόνιση δεδομένων από διαφορετικές πηγές, με διαφορετική διαμόρφωση των δεδομένων σε μια σχεσιακή βάση δεδομένων. Τα δεδομένα, λήφθηκαν από τις υπάρχουσες πηγές αποδομήθηκαν πλήρως και στη συνέχεια αναδομήθηκαν στη σχεσιακή βάση δεδομένων που σχεδιάστηκε. Για την διαδικασία αυτή υλοποιήθηκαν μετατροπές στην διαμόρφωση των δεδομένων. Επιπλέον, η σωστή επιλογή των πεδίων από τις υπάρχουσες βάσεις αλλά και εξαγωγή επιπλέον πληροφοριών από αυτά, συμβάλλει στη δημιουργία συνεκτικής και στοχευμένης βάσης. Για την απόδειξη της λειτουργικότητας του προτεινόμενου μοντέλου, υλοποιήθηκε εφαρμογή βάσης δεδομένων σε Python, η οποία επιτρέπει την αλληλεπίδραση του χρήστη με τα δεδομένα μέσω ενός συστήματος διεπαφών, RESTful API. Με τη συνένωσή τους υπό το ενιαίο αυτό μοντέλο, τα δεδομένα καταστροφών απαλλάσσονται από επιπλέον επίπεδα επεξεργασίας και καθίστανται έτοιμα για ερευνητική χρήση.

Λέξεις Κλειδιά: Φυσικές καταστροφές, Εναρμόνιση δεδομένων, Μοντελοποίηση δεδομένων, RESTful API.

Abstract

There are hundreds of disaster databases, but they do not follow uniform standards for formatting data. Therefore, it becomes extremely difficult to comprehensively review and process the data, which is necessary for study, prediction and drawing conclusions. There is a need to collect, harmonize and store natural disaster data following unified modeling. For this purpose, a relational database model is designed, capable of hosting data of various natural disasters, by storing their main elements. It is necessary to harmonize data from different sources, with different configurations. The data, taken from the existing sources were completely destructured and then restructured into the relational database designed. For this process, changes were implemented in the configuration of the data. In addition, the correct selection of the fields from the existing databases and the extraction of additional information from them contribute to the creation of a coherent and targeted database. To prove the functionality of the proposed model, a database application was implemented in Python, which allows the user to interact with the data through a system of interfaces, RESTful API. By bringing them together under this single model, disaster data are freed of additional layers of processing and made ready for research use.

Key Words: Natural disasters, data harmonization, data modeling, RESTful API.

Ευχαριστίες

Πρώτα από όλα θα ήθελα να ευχαριστήσω από καρδιάς τον κύριο Δημήτριο Ασκούνη, που με εμπιστεύτηκε για την εκπόνηση αυτού του ενδιαφέροντος θέματος για διπλωματική εργασία. Επιπλέον, ευχαριστώ θερμά τους Ηλιάνα Μάλλιου και Χριστόδουλο Σαντοριναίο για την πολύτιμη βοήθεια που μου προσέφεραν σε όλο αυτό διάστημα που μεσολάβησε από την ανάθεση της διπλωματικής έως και την τελειοποίηση της. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου για τη στήριξη της όλα αυτά τα χρόνια και τους συμφοιτητές μου Θεοδόσιο Παπαντωνίου, Δημήτριο Μυλωνάκη, Γιώργο Λιβιτσάνο και Ιάσωνα Ζαράνη για τα όμορφα φοιτητικά χρόνια που περάσαμε μαζί.

ΠΕΡΙΕΧΟΜΕΝΑ

1	Αναζήτηση και επιλογή ετερογενών βάσεων δεδομένων καταστροφών	13
1.1	Παρουσίαση πηγών/βάσεων δεδομένων	13
1.1.1	EMDAT	13
1.1.2	Η βάση Δεδομένων Ηφαιστειών του Κόσμου	14
1.1.3	Σύνολα Δεδομένων Πυροσβεστικού Σώματος Ελλάδος.	14
1.1.4	National Centers for Environmental Information (NCEI)	14
1.1.5	The Significant Earthquake Database	15
1.1.6	The NCEI/WDS Tsunami Runup Database.....	15
1.1.7	The Significant Volcanic Eruption Database.....	15
1.1.8	NatCatSERVICE.....	16
1.2	Καθορισμός κριτηρίων αξιολόγησης των πηγών/βάσεων δεδομένων.....	17
1.2.1	Εγκυρότητα	17
1.2.2	Πληρότητα	17
1.2.3	Ομογενοποίηση.....	18
1.2.4	Τρόποι εξαγωγής δεδομένων	18
1.2.5	Γλώσσα	18
1.3	Επιλογή των πηγών/βάσεων προς αξιοποίηση.....	18
1.3.1	EMDAT	19
1.3.2	NASA Open Data Portal- Global Landslide Catalog	20
1.3.3	Fema Disaster Declarations Summary.....	21
2	Σχεδιασμός της βάσης δεδομένων.....	23
2.1	Σχεσιακές βάσεις δεδομένων	23
2.1.1	Κλειδιά.....	23
2.1.2	Το πρότυπο SQL.....	24
2.1.3	Περιορισμοί πεδίου τιμών στην SQL	24
2.2	Η βάση που χρησιμοποιήθηκε για την εφαρμογή.....	25
2.3	Το σχήμα της βάσης.....	26
2.3.1	Πίνακας γενικών πληροφοριών καταστροφών	27
2.3.2	Πίνακας συμβάντος.....	28
2.3.3	Πίνακας πληροφοριών τοποθεσίας.....	29
2.3.4	Πίνακας χρονικών δεδομένων	30
2.3.5	Πίνακας συνέπειών	31

2.4	Κατασκευή της βάσης με γνώμονα το σχήμα της.....	31
3	Υλοποίηση της βάσης, κατασκευή της βάσης και εισαγωγή στοιχείων	39
3.1	Σύνδεση με βάση.....	39
3.2	Δημιουργία ερωτημάτων στην βάση.....	40
3.3	υποστηριζόμενα ερωτήματα.....	40
3.4	Εξαγωγή δεδομένων από τις βάσεις.....	42
4	Εναρμόνιση των δεδομένων	43
4.1.1	Επιλογή των κατάλληλων πεδίων	43
4.1.2	Εξομάλυνση της διαμόρφωση των πεδίων	44
5	Εργαλεία για την ανάπτυξη της βάσης και την υλοποίηση	47
5.1.1	Microsoft Visual Studio	47
5.1.2	Python	50
5.1.3	Εγκατάσταση βιβλιοθηκών.....	50
6	Διασύνδεση της βάσης με διεπαφές(RESTful APIs)	53
6.1	Διεπαφή προγραμματισμού εφαρμογών	53
6.2	Http πρωτόκολλο.....	53
6.2.1	Ορισμός.....	53
6.2.2	Μέθοδοι αίτησης του HTTP	55
6.2.3	Κωδικοί στις απαντήσεις	55
6.3	Restful διεπαφές.....	56
6.4	Υλοποίηση των διαπαφών.....	56
6.4.1	Εγκατάσταση	56
6.5	Μετατροπή των δεδομένων σε JSON	57
6.6	Διεπαφές που δημιουργήθηκαν	57
6.7	Αποτελέσματα.....	59
7	Συμπεράσματα και μελλοντικές προοπτικές	63
7.1	Συμπεράσματα	63
7.2	Μελλοντικές προοπτικές.....	64
7.2.1	Επικράτηση ενιαίου προτύπου στις φυσικές καταστροφές	64
7.2.2	Επέκταση	64
7.2.3	Επιπλέον διεπαφές	64
7.2.4	Ποιοτική αναβάθμιση των δεδομένων.....	64
7.2.5	Πρόβλεψη μελλοντικών φυσικών καταστροφών	65
8	Βιβλιογραφία.....	67

Περιεχόμενα πινάκων

Πίνακας 1: EM-DAT	19
Πίνακας 2: NASA Open Data Portal- Global Landslide Catalog.....	20
Πίνακας 3: Fema Disaster Declarations Summary	21
Πίνακας 4: Κατασκευή του σχήματος της βάσης.....	34
Πίνακας 5: Κατασκευή της βάσης με χρήση του σχήματος.....	37
Πίνακας 6: Εναρμόνιση των δεδομένων.....	45
Πίνακας 7: Κώδικας διεπαφής.....	59

Περιεχόμενα σχημάτων

Εικόνα 2.1 Το σχήμα της βάσης	26
Εικόνα 2.2 Πίνακας γενικών πληροφοριών καταστροφών.....	27
Εικόνα 2.3 Πίνακας συμβάντος	28
Εικόνα 2.4 Πίνακας πληροφοριών τοποθεσίας.....	29
Εικόνα 2.5 Πίνακας χρονικών δεδομένων	30
Εικόνα 2.6 Πίνακας συνεπειών.....	31
Εικόνα 2.7 Επιλογές εξαγωγής του κώδικα της βάσης.....	32
Εικόνα 3.1 Κώδικας σύνδεσης στην βάση.....	39
Εικόνα 3.2 Κώδικας εισαγωγής δεδομένων σε πίνακα της βάσης	40
Εικόνα 5.1 Μονοπάτι εγκατάστασης	48
Εικόνα 5.2 Επιπλέον επιλογές	49
Εικόνα 5.3 Τελικό περιβάλλον χρήστη.....	50
Εικόνα 5.4 Εγκατάσταση βιβλιοθηκών στο τερματικό	51
Εικόνα 6.1 Ανάλυση του πρωτοκόλλου HTTP	54
Εικόνα 6.2 Η δομή του μηνύματος.....	54
Εικόνα 6.3 Σύνολο διεπαφών που υλοποιήθηκαν	57
Εικόνα 6.4 Εκτέλεση του Fast API.....	59
Εικόνα 6.5 Τα δεδομένα που αποστέλλονται	60

1 ΑΝΑΖΗΤΗΣΗ ΚΑΙ ΕΠΙΛΟΓΗ ΕΤΕΡΟΓΕΝΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ ΚΑΤΑΣΤΡΟΦΩΝ

1.1 ΠΑΡΟΥΣΙΑΣΗ ΠΗΓΩΝ/ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Παρακάτω παρουσιάζονται οι βάσεις δεδομένων που εν δυνάμει μπορεί να επιλεγούν. Για κάθε βάση παρουσιάζονται βασικές πληροφορίες όπως τα δεδομένα που προσφέρονται, τρόποι εξαγωγής τους, πληρότητα κ.α. Όλες οι βάσεις διατίθενται από διακεκριμένους στον κλάδο οργανισμούς, έχουν κάλυψη σε επίπεδο χώρας, ηπείρου ή παγκόσμια και μπορεί να αφορούν ένα ή περισσότερα είδη καταστροφών.

1.1.1 EMDAT

Η Βάση Δεδομένων Γεγονότων Έκτακτης Ανάγκης (Emergency Events Database - EM-DAT) [1] [2] δημιουργήθηκε το 1988 και συντηρείται μέχρι και σήμερα από το Κέντρο Έρευνας για την Επιδημιολογία των Καταστροφών (Centre for Research on the Epidemiology of Disasters - CRED) της Σχολής Δημόσιας Υγείας του Καθολικού Πανεπιστημίου της Λουβαίν (UCLouvain). Η Βάση Δεδομένων Γεγονότων Έκτακτης δημιουργήθηκε με την αρχική υποστήριξη του Παγκόσμιου Οργανισμού Υγείας (ΠΟΥ) αλλά και της βελγικής κυβέρνησης.

Δεδομένα από οργανισμούς του ΟΗΕ, μη κυβερνητικές οργανώσεις, ασφαλιστικές εταιρείες, ερευνητικά ιδρύματα και πρακτορεία τύπου συνθέτουν τις πηγές των βάσεων δεδομένων. Δεδομένα από οργανισμούς του ΟΗΕ, κυβερνήσεις και τη Διεθνή Ομοσπονδία Ερυθρού Σταυρού και Ερυθράς Ημισελήνου εισάγονται με προτεραιότητα. Έτσι εξασφαλίζεται πως τα δεδομένα δεν επηρεάζονται από πολιτικές σκοπιμότητες και περιορισμούς.

Για αποφυγή ασυνέπειών, πλεονασμών και ελλείψεων ελέγχονται συνεχώς οι παρούσες εγγραφές. Ενώ το CRED συγχωνεύει και πληροφορεί τα δεδομένα σε καθημερινή βάση. Στο τέλος κάθε ημερολογιακού έτους γίνονται αναθεωρήσεις, επίσης ένας περαιτέρω έλεγχος γίνεται σε μηνιαία διαστήματα. Σε καθημερινή βάση συμπληρώνονται νέα δεδομένα εσωτερικά στη βάση EM-DAT. Η πρόσβαση στα δεδομένα της EM-DAT είναι ανοιχτή στην ιστοσελίδα της βάσης. [2]

Καταστροφικά γεγονότα εισάγονται στη βάση δεδομένων EM-DAT εφόσον πληρούν τουλάχιστον ένα από τα ακόλουθα κριτήρια συμπερίληψης: [1]

- το γεγονός οδήγησε στο θάνατο 10 ή περισσότερων ανθρώπων
- το γεγονός επηρέασε 100 ή περισσότερους ανθρώπους
- το γεγονός οδήγησε σε κήρυξη κατάστασης έκτακτης ανάγκης
- το γεγονός οδήγησε σε έκκληση για διεθνή βοήθεια.

1.1.2 Η βάση Δεδομένων Ηφαιστειών του Κόσμου

Η Βάση Δεδομένων Ηφαιστειών του Κόσμου (Volcanoes of the World Database) δημιουργήθηκε από το National Museum of Natural History – Smithsonian Institute ως μέρος του παγκοσμίου ηφαιστειακού προγράμματος το 1968. Ο κυριότερος στόχος της βάσης και του προγράμματος είναι η τεκμηρίωση και η κατανόηση των δεδομένων σχετικά με την παγκόσμια ηφαιστειακή δραστηριότητα καθώς και την διάδοσή τους . Μέσω αναφορών, αρχειοθέτησης και έρευνας των δεδομένων γύρω από τις ηφαιστειακές καταστροφές κατορθώνονται αυτοί οι στόχοι.. Η πρόσβαση στα δεδομένα είναι ανοιχτή μέσω της ιστοσελίδας του οργανισμού στην αγγλική γλώσσα.

Τα δεδομένα της βάσης δεδομένων Volcanoes of the World αναφέρονται σε περισσότερα από 11.000 γεγονότα ηφαιστειακής δραστηριότητας, για τα τελευταία 10.000 χρόνια ανά την οικουμένη. Πληροφορίες σχετικά με την τοποθεσία της ηφαιστειακής δραστηριότητας, του χρονικού πλαισίου της, καθώς και την έντασή της σε κλίμακα VEI περιέχονται στην βάση .

1.1.3 Σύνολα Δεδομένων Πυροσβεστικού Σώματος Ελλάδος.

Το Πυροσβεστικό Σώμα Ελλάδος παρέχει σε ανοιχτή διάθεση δεδομένα σχετικά με τις δραστηριότητες του σώματος, τόσο στα αστικά κέντρα όσο και στην επαρχία, με ιδιαίτερη έμφαση στις αστικές και αγροτοδασικές πυρκαγιές . Η ανοιχτή διάθεση των δεδομένων απαιτείται από την εθνική νομοθεσία, σύμφωνα με τις διατάξεις της οδηγίας 2013/37/ΕΕ του Ευρωπαϊκού Κοινοβουλίου, για την ενίσχυση της διαφάνειάς των δραστηριοτήτων του σώματος. Πυροσβεστικό Σώμα Ελλάδος προσφέρει δεδομένα που προέρχονται μέσα από τα αρχεία του ίδιου του σώματος.

Προσφέρει δύο ξεχωριστά σύνολα δεδομένων για συμβάντα για τα οποία δραστηριοποιήθηκαν δυνάμεις του πυροσβεστικού σώματος, μοναδικά στον ελλαδικό χώρο:

Το Σύνολο Δεδομένων Αστικών Συμβάντων, το οποίο περιλαμβάνει πληροφορίες σχετικά με το συμβάν καθαυτό, χρονικά και τοπικά δεδομένα, καθώς και το μέγεθος των πυροσβεστικών δυνάμεων που χρειάστηκαν για την επίλυση του συμβάντος. Το σύνολο αυτό καλύπτει το χρονικό πλαίσιο από το 2014 μέχρι και το 2021. Τα δεδομένα ανανεώνονται σε ετήσια βάση.

Το Σύνολο Δεδομένων Δασικών Συμβάντων ,οπού περιέχει πληροφορίες σχετικά με το συμβάν καθαυτό, χρονικά και τοπικά δεδομένα, το μέγεθος και το είδος των πυροσβεστικών δυνάμεων που κινητοποιήθηκαν για την επίλυσή του, καθώς και το μέγεθος της ζημιάς που προκάλεσε η φωτιά σε καμένη έκταση. Το σύνολο αυτό περιέχει περίπου 200.000 αναφορές δασικών συμβάντων, καλύπτοντας το χρονικό πλαίσιο από το 2000 μέχρι και το 2019.

1.1.4 National Centers for Environmental Information (NCEI)

Το Εθνικό Κέντρο Περιβαλλοντικών Πληροφοριών (National Centers for Environmental Information - NCEI)συνιστά την επίσημη αρχή συλλογής και καταγραφής περιβαλλοντικών

δεδομένων των Ηνωμένων Πολιτειών της Αμερικής. Το NCEI συντονίζει ένα από τα μεγαλύτερα αρχεία ατμοσφαιρικής, παράκτιας, γεωφυσικής και ωκεανικής έρευνας στον κόσμο και ανταποκρίνεται στην αυξανόμενη ανάγκη για δεδομένα υψηλής αξίας, συμβάλλοντας στην περιβαλλοντική έρευνα, και στην προσπάθεια μετατροπής των επιστημονικών γνώσεων σε δυναμικές, εύχρηστες πληροφορίες που θα ορίζουν την στρατηγική και τη λήψη αποφάσεων στον κυβερνητικό, ακαδημαϊκό και ιδιωτικό τομέα.

Το NCEI ταξινομεί τα στοιχεία σε αρχεία και φανερώνει την ύπαρξη σχέσεων μεταξύ δεδομένων τσουνάμι, σεισμών και ηφαιστειακής δραστηριότητας για την υποστήριξη της έρευνας, του σχεδιασμού και της απόκρισης σε αυτού του είδους τις φυσικές καταστροφές. Ειδικότερα, περιλαμβάνει, μεταξύ άλλων, τις ακόλουθες τρεις βάσεις δεδομένων.

1.1.5 The Significant Earthquake Database

Η Βάση Δεδομένων Αξιοσημείωτων Σεισμών (The Significant Earthquake Database) [3] περιλαμβάνει δεδομένα για καταστροφικούς σεισμούς από το 2150 π.Χ. έως σήμερα, που ικανοποιούν τουλάχιστον ένα από τα ακόλουθα κριτήρια:

- ο σεισμός προκάλεσε ζημιές τουλάχιστον ενός εκατομμυρίου αμερικάνικα δολάρια • ο σεισμός προκάλεσε 10 ή περισσότερους θανάτους
η ένταση του σεισμού ήταν τουλάχιστον 7.5 σε κλίμακα Richter ή 10 (X) σε κλίμακα Mercalli (Modified Mercalli Intensity - MMI)
- ο σεισμός προκάλεσε τσουνάμι. Τα δεδομένα συγκεντρώνονται και εισάγονται σε επίπεδο καταστροφής. Όταν η ίδια καταστροφή πλήττει πολλές χώρες, εισάγεται στη βάση μια εγγραφή, στην οποία αναφέρονται τόσο η τοποθεσία στην οποία συνέβη η καταστροφή, όσο και οι τοποθεσίες που επηρεάστηκαν.

1.1.6 The NCEI/WDS Tsunami Runup Database

Η Βάση Δεδομένων Τσουνάμι της NCEI/WD [4] (The NCEI/WDS Tsunami Runup Database), περιλαμβάνει περισσότερες από 28.300 εγγραφές σχετικές με θαλάσσια δραστηριότητα η οποία φέρει τα χαρακτηριστικά ενός παλιρροϊκού κύματος, παγκοσμίως, από το 2000 π.Χ. μέχρι σήμερα. Ειδικότερα Τα δεδομένα συλλέγονται σε επίπεδο χώρας. Όταν η ίδια καταστροφή προσκρούει πολλές χώρες, προστίθενται στη βάση ισάριθμες εγγραφές. Μας προσφέρει τη δυνατότητα για μελέτη επιπλέον κατηγοριών φυσικών καταστροφών καθώς προσφέρει δεδομένα για τσουνάμι. Οι καταστροφές αυτές έχουν ιδιαίτερο ενδιαφέρον στην ανάλυση και στην εξαγωγή αποτελεσμάτων καθώς υπάρχει συχνή φυσική σύνδεση τους με άλλες καταστροφές. Για παράδειγμα ένα τσουνάμι μπορεί να προκληθεί έπειτα από έναν ισχυρό σεισμό

1.1.7 The Significant Volcanic Eruption Database

Η Βάση Δεδομένων Αξιοσημείωτων Ηφαιστειακών Εκρήξεων (The Significant Volcanic Eruption Database) περιλαμβάνει παραπάνω από 800 εγγραφές σχετικές με ηφαιστειακή δραστηριότητα, και ειδικότερα, εκρήξεις ηφαιστειών παγκοσμίως, από το 4360 π.Χ. μέχρι σήμερα. Αποτελεί μία από τις πιο διαδεδωμένες βάσεις για άντληση και μελέτη δεδομένων ηφαιστειακών εκρήξεων. Τα δεδομένα συλλέγονται και εισάγονται σε επίπεδο καταστροφής και αναφέρεται μόνο η χώρα στην οποία συνέβη η έκρηξη, χωρίς πληροφορίες για άλλες χώρες που έφερε επιπτώσεις από αυτή. Το NCEI περιλαμβάνει στοιχεία σχετικά με την καταστροφή

καθαυτή, τις επιπτώσεις της στην ανθρώπινη ζωή, καθώς και τις προκληθείσες οικονομικές και υλικές απώλειες. Οι τιμές των εκτιμώμενων οικονομικών απωλειών δόθηκαν σε εκατομμύρια αμερικάνικα δολάρια. Οι βάσεις δεδομένων δείχνουν μια σχέση μεταξύ τους μέσω των ξεχωριστών αναγνωριστικών των εγγραφών των σεισμών, όπου υπάρχει συσχέτιση γεγονότων σεισμών, ηφαιστειακών εκρήξεων και θαλάσσιας δραστηριότητας. Τα σύνολα δεδομένων φυσικών καταστροφών προσφέρονται στους ενδιαφερομένους μέσω της διασυνδέσεως HazEL (Hazardous Event Lookup) , που δημιουργήθηκαν ,αναλύθηκαν και εξελίχθηκαν από την NCEI.

1.1.8 NatCatSERVICE

Η αντασφαλιστική εταιρεία Munich Re παρέχει και συντηρεί την βάση δεδομένων NatCatSERVICE από το 1980. Δεν παρέχει πρόσβαση στο κοινό απευθείας στα δεδομένα, μέσω αυτής διατίθενται επιλεγμένες αναλύσεις και πίνακες, ενώ δημοσιεύεται σε ετήσια βάση μια ανασκόπηση των φυσικών καταστροφών . Η βάση δεδομένων NatCatSERVICE περιλαμβάνει φυσικές καταστροφές κάθε είδους, με ιστορικές πληροφορίες που χρονολογούνται από το 79 μ.Χ. (έκρηξη του Βεζούβιου).Από το 1980 ,η βάση δεδομένων NatCatSERVICE, διατίθεται αντασφαλιστική επιχείρηση Munich Re .Το κοινό δεν είχε προσβασιμότητα στα δεδομένα της ,διέθεταν όμως επιλεγμένες αναλύσεις και πίνακες , ενώ δημοσιεύεται σε ετήσια βάση μια ανασκόπηση των φυσικών καταστροφών (Topics Geo Review of Natural Catastrophes).Στην βάση δεδομένων NatCatSERVICE περιέχονται φυσικές καταστροφές κάθε είδους ,με τις ιστορικές πληροφορίες να χρονολογούνται από το 79 μ.Χ.(έκρηξη του Βεζούβιου) αλλά και με δεδομένα αντίστοιχα για αναλυτική αξιολόγηση καταστροφών που καλύπτονται από το 1950 μέχρι σήμερα ,σε παγκόσμιο επίπεδο .Κάθε χρόνο ,η βάση δεδομένων NatCatSERVICE ,ανανεώνεται με περίπου 800 γεγονότα . Τα δεδομένα ενσωματώνονται με τον τύπο της καταστροφής, την περιγραφή της αλλά και ασφαλισμένες οικονομικές απώλειες. Τα δεδομένα οικονομικών απωλειών της NatCatSERVICE εξάγονται κυρίως από εσωτερικές εκθέσεις εταιρειών και εκτιμήσεις εμπειρογνομόνων αλλά πηγές περιλαμβάνουν και ασφαλιστικές ενώσεις, επιστημονικές πηγές, κυβερνητικούς και μη κυβερνητικούς οργανισμούς.

1.2 ΚΑΘΟΡΙΣΜΟΣ ΚΡΙΤΗΡΙΩΝ ΑΞΙΟΛΟΓΗΣΗΣ ΤΩΝ ΠΗΓΩΝ/ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Στην παρούσα ενότητα θα οριστούν ορισμένα κριτήρια αξιολόγησης των υπάρχοντων βάσεων δεδομένων που παρουσιάστηκαν στην παραπάνω Ενότητα. Σκοπός των κριτηρίων αυτών είναι η επιλογή ενός υποσυνόλου βάσεων δεδομένων καταστροφών, από τη μελέτη των οποίων θα προκύψει ένα μοντέλο για τη συνένωσή τους

Για την διασφάλιση της ποιότητας και της αξιοπιστίας των δεδομένων είναι σημαντικό να γίνει κατάλληλη επιλογή των δεδομένων που θα χρησιμοποιηθούν.

1.2.1 Εγκυρότητα

Η πηγή των δεδομένων πρέπει να παρέχει μια ικανοποιητική ασφάλεια ως προς την εγκυρότητά τους ώστε να μην αμφισβητείται η πραγματικότητα των δεδομένων. Η παραγωγή έγκυρων και ορθών συμπερασμάτων μετά την εξεργασία των δεδομένων απαιτεί να προέρχονται από έγκυρες πηγές.

1.2.2 Πληρότητα

Η καταγραφή των δεδομένων πρέπει να είναι πλήρης. Είναι σημαντικό να περιέχονται οι απαραίτητες πληροφορίες, όπως: Τόπος(συντεταγμένες ή διεύθυνση), Ένταση, Χρονολογία, Επιπτώσεις, ώστε να μπορεί να γίνει αξιοποίησή τους. Συγκεκριμένα θεωρούμε ότι μια βάση πληροί το κριτήριο της πληρότητας αν περιέχει (ή μπορούν να παραχθούν τα παρακάτω):

- **Συντεταγμένες:** Longitude και latitude πρέπει να περιλαμβάνονται. Επιπλέον, αυτή η πληροφορία μπορεί να παραχθεί αν παρέχεται η πόλη, αυτό όμως θα μειώσει την ακρίβεια των συντεταγμένων.
- **Ένταση:** Είναι σημαντικό να υπάρχει μέτρηση του μεγέθους της καταστροφής. Κάθε είδος καταστροφής έχει διαφορετικό τρόπο μέτρησης.
- **Χρονολογία:** Θα παράγουμε ή θα χρησιμοποιήσουμε την χρονολογία με Ευρωπαϊκή μορφοποίηση(η ώρα είναι προαιρετική)
- **Χώρα:** Η χώρα πρέπει να παρέχεται με αγγλικούς χαρακτήρες και με μοναδικό id ανά χώρα.
- **Είδος καταστροφής:** Απαραίτητο είναι να προσδιορίζεται και το είδος της καταστροφής καθώς επηρεάζει την επεξεργασία τους.

1.2.3 Ομογενοποίηση.

Για να είναι δυνατή η σύνθεση και αξιοποίηση δεδομένων από διαφορετικές πηγές, είδη καταστροφών και τρόπο μέτρησης και συλλογής, είναι απαραίτητη μια συσχέτιση μεταξύ τους. Ο τελικός σκοπός είναι παραγωγή ενός μοντέλου αποθήκευσης δεδομένων που θα περιέχει ομοιόμορφο δεδομένα αναλυμένες στην ίδια βάση. όσο πιο συνεκτικά είναι αυτά τα δεδομένα όταν εισαχθούν στο σύστημα επεξεργασίας τους τόσο αυξάνονται οι εν δυνάμει δυνατότητες στο μοντέλου.

1.2.4 Τρόποι εξαγωγής δεδομένων

Μεγάλη σημασία έχουν και οι τρόποι εξαγωγής και προσπέλασης των δεδομένων, είναι σημαντικό να γίνεται χρήση αξιόπιστων μεθόδων που προσφέρουν και ευελιξία προς την αξιοποίησή τους. Η ακρίβεια και η αξιοπιστία, καθώς και η ταχύτητα της εξαγωγής των δεδομένων από την υπάρχουσα βάση πρόσωπο δόμηση και επεξεργασία τους με τελικό σκοπό να εισαχθούν στο μοντέλο που δημιουργήθηκε, αποτελεί απαραίτητη προϋπόθεση.

1.2.5 Γλώσσα

Η γλώσσα στην οποία παρέχονται τα δεδομένα είναι επίσης σημαντικό κριτήριο επιλογής. Καθώς για ετερογενείς γλώσσες απαιτείται επιπλέον επεξεργασία των δεδομένων για να επιτευχθεί η ομογενοποίηση τους. Στα πλαίσια της διπλωματικής θα προτιμηθεί η επιλογή βάσεων που προσφέρουν τα δεδομένα τους στην αγγλική γλώσσα. η κοινή γλώσσα το τιμών των δεδομένων, προσφέρει μεγαλύτερη συνεκτικότητα στα αρχικά δεδομένα προσφέροντας έτσι καλύτερα αποτελέσματα στα τελικά δεδομένα που θα παραχθούν στο μοντέλο.

1.3 ΕΠΙΛΟΓΗ ΤΩΝ ΠΗΓΩΝ/ΒΑΣΕΩΝ ΠΡΟΣ ΑΞΙΟΠΟΙΗΣΗ

Δεν πληρούσαν όλες οι βάσεις που μελετήθηκαν τα απαραίτητα κριτήρια. Για την εκπόνηση της διπλωματικής έγινε επιλογή συγκεκριμένων βάσεων που αναλύονται παρακάτω. Για την ανάλυση και τη σύγκριση των βάσεων δημιουργήθηκε πίνακας, που περιλαμβάνει τις βασικές πληροφορίες της βάσης όπως για παράδειγμα τους τύπους των καταστροφών περιλαμβάνει τα κριτήρια εισαγωγής δεδομένων, τις διεπαφές προσφέρει κ.α.

1.3.1 EMDAT

Βάση	EMDAT
Url	https://public.emdat.be/data
Περιγραφή	
Είδος δεδομένων	
Τύποι καταστροφών	Σεισμοί, Πλυμήρες, Υφέστια, Καταιγίδες, Ακραίες θερμοκρασίες, Πυρκαγιές
Γλώσσα	Αγγλικά
Προσβασιμότητα	Δημόσια αλλά απαιτείται εγγραφή
Συχνότητα ανανέωσης δεδομένων	-
Τρόποι εξαγωγής δεδομένων	Κατέυασμα αρχείου
Μορφοποίηση δεδομένων	xlsx
Απευθύνεται σε	
Σχετικότητα	Περιέχει δεδομένα τα οποία ανήκουν στο πλαίσιο των φυσικών καταστροφών που εξετάζουμε
Κριτήριο εισαγωγής δεδομένων στην βάση	<ul style="list-style-type: none"> • Περισσότεροι από 10 θάνατοι ή • περισσότεροι από 100 άνθρωποι επηρεάστηκαν από την καταστροφή ή • Ανακήρυξη κατάστασης εκτάκτου ανάγκης • Βοήθεια από άλλες χώρες
Πληρότητα	Πληροί πλήρως τις προϋποθέσεις που θέσαμε

Πίνακας 1: EM-DAT

1.3.2 NASA Open Data Portal- Global Landslide Catalog

Βάση	NASA Open Data Portal- Global Landslide Catalog
Url	https://data.nasa.gov/Earth-Science/Global-Landslide-Catalog-Export/dd9e-wu2v/data
Περιγραφή	
Είδος δεδομένων	
Τύποι καταστροφών	Υποχώρηση εδάφους
Γλώσσα	Αγγλική
Προσβασιμότητα	Δημόσια
Συχνότητα ανανέωσης δεδομένων	Τελευταία Ενημέρωση: 3 Απριλίου 2018
Τρόποι εξαγωγής δεδομένων	Κατέβασμα αρχείου και διεπαφές/Api endpoint
Μορφοποίηση δεδομένων	<ul style="list-style-type: none"> • Xlsx • Csv • Rdf • Tsv • Json
Απευθύνεται σε	
Σχετικότητα	Περιέχει δεδομένα τα οποία ανήκουν στο πλαίσιο των φυσικών καταστροφών που εξετάζουμε
Κριτήριο εισαγωγής δεδομένων στην βάση	<ul style="list-style-type: none"> • Περισσότεροι από 10 θάνατοι ή • περισσότεροι από 100 άνθρωποι επηρεάστηκαν από την καταστροφή ή • Ανακήρυξη κατάστασης εκτάκτου ανάγκης • Βοήθεια από άλλες χώρες
Πληρότητα	Πληροί πλήρως τις προϋποθέσεις που θέσαμε

Πίνακας 2: NASA Open Data Portal- Global Landslide Catalog

1.3.3 Fema Disaster Declarations Summary

Βάση	Fema Disaster Declarations Summary
Url	https://www.fema.gov/openfema-data-page/disaster-declarations-summaries-v1
Περιγραφή	
Είδος δεδομένων	
Τύποι καταστροφών	Πλημύρες, πυρκαγιές, ανεμοστρόφυλους, καταιγίδες
Γλώσσα	Αγγλική
Προσβασιμότητα	Δημόσια
Συχνότητα ανανέωσης δεδομένων	Τελευταία Ενημέρωση: 3 Ιουνίου 2022
Τρόποι εξαγωγής δεδομένων	Κατέβασμα αρχείου και διεπαφών/ApI endpoint
Μορφοποίηση δεδομένων	<ul style="list-style-type: none">• Csv• Json• Jsona
Απευθύνεται σε	
Σχετικότητα	Περιέχει δεδομένα τα οποία ανήκουν στο πλαίσιο των φυσικών καταστροφών που εξετάζουμε
Κριτήριο εισαγωγής δεδομένων στην βάση	
Πληρότητα	Πληροί πλήρως τις προϋποθέσεις που θέσαμε, όμως περιέχει μεγαλύτερη πληθώρα καταστροφών.

Πίνακας 3: Fema Disaster Declarations Summary

2 ΣΧΕΔΙΑΣΜΟΣ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

2.1 ΣΧΕΣΙΑΚΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Η βάση δεδομένων συνιστάται από ένα σύνολο πινάκων δεδομένων. Οι ακολουθίες τιμών ουσιαστικά είναι οι σειρές του πίνακα όπου λέγονται εγγραφές. Ιδιότητες ονομάζονται οι εγγραφές όπου ερμηνεύονται προς μια σχέση αναμεταξύ των στηλών του πίνακα. Πιο αναλυτικά, οι ιδιότητες του πίνακα συσχετίζονται μέσω συγκεκριμένων τιμών στο οποίο ερμηνεύεται η εγγραφή [5]. Από μια εγγραφή, συνδέονται βασικές τιμές από ιδιότητες του πίνακα και παίρνει το όνομα του από το σχεσιακό πρότυπο των βάσεων δεδομένων. Το σχεσιακό πρότυπο έχει τον όρο να μεταχειρίζεται για έναν πίνακα.

Όταν αναφερόμαστε για μια βάση δεδομένων, επιβάλλεται να διαφοροποιούμε το σχήμα της με βάση των δεδομένων όπου είναι η αιτιολογία σχεδιασμού των βάσεων δεδομένων από ένα στιγμιότυπο βάσης δεδομένων όπου κάποια χρονική στιγμή είναι τα δεδομένα της βάσης. Αν αναφερθούμε στο στιγμιότυπο μιας σχέσης μπορεί να προκαλέσει μια σύγχυση διότι κάνουμε χρήση του ίδιου ονόματος. [6]

2.1.1 Κλειδιά

Αφού εισαχθούν τα δεδομένα στη βάση είναι απαραίτητο να υπάρχει μια διαφοροποίηση ανάμεσα στις εγγραφές για το λόγο αυτό χρειάζεται κάποιου είδους αναγνωριστικό το οποίο ονομάζεται κλειδί και είναι μοναδικό να εγγραφεί και ανά πίνακα. Αυτό διατυπώνεται σε σχέση με τις ιδιότητες τους. Με άλλα λόγια, οι ιδιότητες μια εγγραφής απαιτούνται να είναι τέτοιες όπου να μπορούν να καθορίζουν μόνο την εγγραφή. Δηλαδή, δεν πρέπει όλες οι ιδιότητες να έχουν την ίδια τιμή σε μια σχέση δύο εγγράφων.

Ένα υπέρ-κλειδί είναι το σύνολο ιδιοτήτων από μια ή περισσότερες που αν ληφθούν ομαδικά, μας αφήνουν να ορίζουμε μοναδικά μια εγγραφή της σχέσης. Υπάρχουν πολλές ιδιότητες που μπορεί να περιέχει ένα υπέρ-κλειδί. [6] Ονομάζονται υποψήφια κλειδιά αυτά τα ελάχιστα υπέρ-κλειδιά. Θα αξιοποιήσουμε τον όρο πρωτεύον κλειδί για να εκδηλώσουμε ένα υποψήφιο κλειδί όπου έχει επιλεγθεί από τον σχεδιαστή της βάσης των δεδομένων ως το βασικό μέσο προσδιορισμού της εγγραφής μιας σχέσης. Το κλειδί έχει την ιδιότητα όλης της σχέσης και δυο τυχαίες εγγραφές στο σύνολο οντοτήτων μας απαγορεύουν να έχουμε ταυτόχρονα την ίδια τιμή της

ιδιότητας του κλειδιού. Μοντελοποιείται η κατάσταση ενός περιορισμού όπου αντιπροσωπεύει ένα κλειδί.

2.1.2 Το πρότυπο SQL

Η πιο κοινά γνωστή γλώσσα είναι η γλώσσα SQL [5] (Structured Query Language) για την διαχείριση δεδομένων σε σχεσιακά σύστημα βάσεων δεδομένων. Μια σύνοψη των λειτουργιών της SQL διαχωρίζουν τις εντολές της σε ετερογενείς ομάδες: [6]

- Εντολές ορισμού δεδομένων: Η SQL έχει τις εντολές για τον ορισμό σχεσιακών σχημάτων, διότι έχει την ευκαιρία να αναπτυχθεί μέσω της εξάλειψης ή της μετατροπής των σχέσεων όπου έχει καθορίσει.
- Εντολές χειρισμού δεδομένων: Οι εντολές που καταστούν μια ισχυρή πρόσβαση και διαχείριση των δεδομένων της βάσης. Για την ακρίβεια, παρέχουν την ανάκληση, την εισαγωγή, της διαγραφή και την επεξεργασία των πληροφοριών που τοποθετούνται στη βάση. Επίσης όπως η καρδιά της SQL είναι queries, τους οι εντολές αυτές που παλινωδούν τις πληροφορίες που πληρούν κάποια συνθήκη.
- Εντολές ακεραιότητας παραπομπών: Σε αυτήν την κατηγορία υπάρχουν οι εντολές που μας βεβαιώνουν από την αποτροπή εισαγωγής δεδομένων στην βάση των δεδομένων όσο τα δεδομένα προσκρούουν κανόνες όπου έχουν θεσπίσει

2.1.3 Περιορισμοί πεδίου τιμών στην SQL

Οι περιορισμοί διατυπώνονται με τον τύπο της συγκεκριμένης ιδιότητας. Οι πιο διαδεδομένοι τύποι που προτείνει το SQL [5] είναι οι ακόλουθοι:

- char(n): σταθερού μήκους συμβολοσειρά
- varchar(n): μεταβλητού μήκους συμβολοσειρά
- int: ακέραιος αριθμητικός αριθμός.
- numeric(p,d): αποτελεί έναν αριθμό σταθερής υποδιαστολής για τον οποίο ο χρήστης επιλέγει την ακρίβεια. Ο αριθμός αποτελείται από p ψηφία και τα d από τα p ψηφία είναι στα δεξιά του δεκαδικού συμβόλου.
- real, double precision: Με ακρίβεια εξαρτώμενη από τον υπολογιστή, με αριθμούς κινητής υποδιαστολής και διπλής ακριβείας.
- float(n): Η ακρίβεια των ψηφίων και είναι ο αριθμός κινητής υποδιαστολής

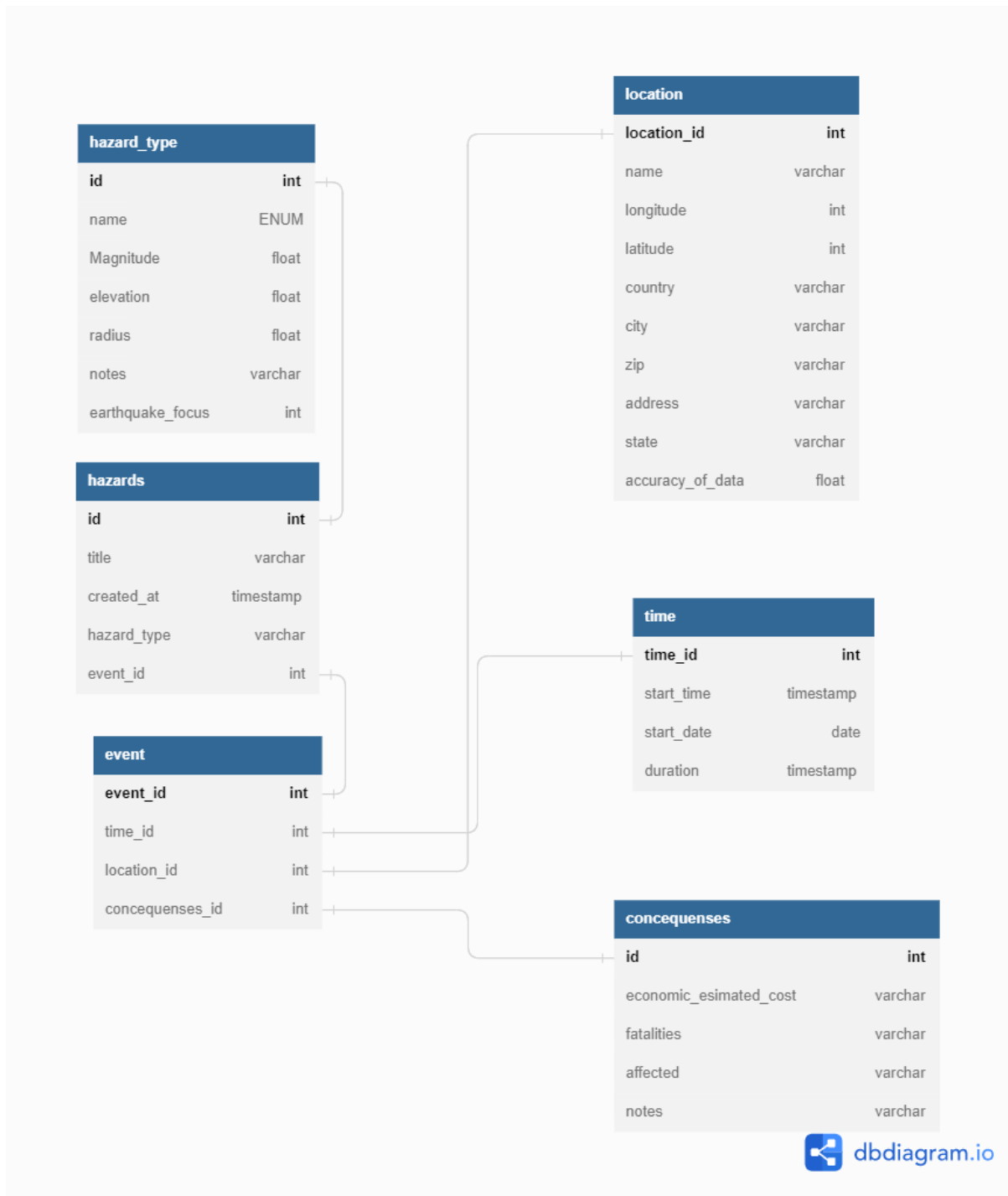
2.2 Η ΒΑΣΗ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΕ ΓΙΑ ΤΗΝ ΕΦΑΡΜΟΓΗ

Παρακάτω θα αναλυθεί η βάση που σχεδιάστηκε και υλοποιήθηκε για την συγκέντρωση των δεδομένων καταστροφών. Όλες οι συνδέσεις είναι 1 to 1 και το id-αναγνωριστικό παραμένει ίδια για κάθε καταστροφή.

Στην εικόνα 2.1 περιγράφεται την ανάλυση της βάσης όπου σχεδιάστηκε και υλοποιήθηκε για να συγκεντρωθούν τα δεδομένα καταστροφών. Οι συνδέσεις είναι 1 to 1 και το id-αναγνωριστικό μένει ίδιο για κάθε καταστροφή.

Για εξαγωγή των δεδομένων πρέπει να γίνει inner join ανάμεσα σε όλους τους πίνακες. Δηλαδή παράχθηκαν δεδομένα από διαφορετικούς πίνακες που περιείχαν ένα ίδιο χαρακτηριστικό. Για διασφάλιση της συνεκτικότητας των δεδομένων και της ποιότητας αυτών των συνδέσεων των πινάκων, επιλέχθηκε η χρήση μοναδικό κλειδιού σε κάθε εγγραφή του των πινάκων και με βάση αυτό τον αριθμό συνενωθήκαν οι πίνακες για την παραγωγή του τελικού αποτελέσματος.
[8]

2.3 ΤΟ ΣΧΗΜΑ ΤΗΣ ΒΑΣΗΣ



Εικόνα 2.1 Το σχήμα της βάσης

Για την αποθήκευση και την ενοποίηση των δεδομένων είναι απαραίτητο να σχεδιαστεί και να κατασκευαστεί η κατάλληλη βάση. Το διάγραμμα του σχήματος της βάσης απεικονίζει τους πίνακες από τους οποίους αποτελείτε η βάση, τα πεδία τους και τις σχέσεις μεταξύ των πινάκων. [9]

Όπως φαίνεται στο παραπάνω σχήμα της βάσης χρησιμοποιήθηκαν οι πίνακες:

2.3.1 Πίνακας γενικών πληροφοριών καταστροφών

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχ
<input type="checkbox"/>	1 id	int(11)			Όχι	Καμία	
<input type="checkbox"/>	2 title	varchar(255)	utf8mb4_general_ci		Ναι	NULL	
<input type="checkbox"/>	3 created_at	timestamp			Όχι	current_timestamp()	
<input type="checkbox"/>	4 hazard_type	varchar(255)	utf8mb4_general_ci		Ναι	NULL	
<input type="checkbox"/>	5 event_id	int(11)			Ναι	NULL	

Εικόνα 2.2 Πίνακας γενικών πληροφοριών καταστροφών

Κώδικας κατασκευής του:

```
CREATE TABLE `hazards`(  
  `id` INT PRIMARY KEY AUTO_INCREMENT,  
  `title` VARCHAR(255),  
  `created_at` TIMESTAMP,  
  `hazard_type` VARCHAR(255),  
);
```

Σκοπός του πίνακα είναι να αποθηκεύσει τις βασικότερες πληροφορίες για κάθε καταστροφή και να συνδέσει τους υπόλοιπους πίνακες.

Ο πίνακας περιέχει:

- Το αναγνωριστικό κλειδί του πίνακα (id): μοναδικός αύξον αριθμός για τον προσδιορισμό της καταστροφής. Το κλειδί είναι ίδιο και στους άλλους πίνακες για σύνδεση των δεδομένων μιας καταστροφής με inner join.
- Τίτλος (title): Σύντομη περιγραφή της καταστροφής
- Χρόνος εισαγωγής στην βάση (created_at): Πεδίο ημέρας-ώρας με την στιγμή εισαγωγής της εγγραφής.
- Είδος καταστροφής: Πεδίο ελεύθερου κειμένου που περιέχει την κατηγορία στην οποία ανήκει η καταστροφή.

- Event_id: Αναφορικό κλειδί(reference key) που συνδέει την καταχώρηση του πίνακα με τους υπόλοιπους.

2.3.2 Πίνακας συμβάντος

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό
<input type="checkbox"/>	1 id	int(11)			Όχι
<input type="checkbox"/>	2 name	varchar(255)	utf8mb4_general_ci		Ναι
<input type="checkbox"/>	3 Magnitude	float			Ναι
<input type="checkbox"/>	4 elevation	float			Ναι
<input type="checkbox"/>	5 radius	float			Ναι
<input type="checkbox"/>	6 notes	varchar(512)	utf8mb4_general_ci		Ναι
<input type="checkbox"/>	7 earthquake_focus	int(11)			Ναι

Εικόνα 2.3 Πίνακας συμβάντος

Περιέχει όλες τις πληροφορίες για τα φυσικά μεγέθη των καταστροφών.

- Name: Όνομα της καταστροφής
- Magnitude: Ρίχτερ για σεισμούς
- Radius/Ακτίνα επίδρασης: Μέτρηση της ακτίνας επίδρασης σε χιλιόμετρα. Περιέχεται σε κάθε είδος καταστροφής. Πραγματικό αριθμητικό πεδίο όπου αναφέρεται η μέγιστη απόσταση στην οποία έγινε αισθητή η καταστροφή (π.χ. για σεισμό, για απόσταση που ταξίδεψαν κύματα τσουνάμι από το σημείο παραγωγής τους, για φωτιά, κλπ.).
- Αναγνωριστικό id: Ακέραιο αριθμητικό πεδίο που ορίζεται ως αύξων αριθμός των εγγραφών, με τη σειρά που εισάγονται στη βάση.
- Elevation: Πραγματικό αριθμητικό πεδίο όπου αναφέρεται τυχόν σχετικό με την καταστροφή ύψος ή βάθος (π.χ. εστιακό βάθος επίκεντρου σεισμού, μέγιστο ύψος κυμάτων τσουνάμι, υψόμετρο ηφαιστείου). Εισάγεται σε μέτρα (m).

- Ακτίνα: Πραγματικό αριθμητικό πεδίο όπου αναφέρεται η μέγιστη απόσταση στην οποία έγινε αισθητή η καταστροφή (π.χ. για σεισμό, για απόσταση που ταξίδεψαν κύματα τσουνάμι από το σημείο παραγωγής τους, για φωτιά, κλπ.). Εισάγεται σε χιλιόμετρα (km).
- Notes/Σημειώσεις: Πεδίο ελεύθερου κειμένου

2.3.3 Πίνακας πληροφοριών τοποθεσίας

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά
<input type="checkbox"/>	1 location_id	int(11)		
<input type="checkbox"/>	2 name	varchar(255)	utf8mb4_general_ci	
<input type="checkbox"/>	3 longitude	float(11,9)		
<input type="checkbox"/>	4 latitude	float(11,9)		
<input type="checkbox"/>	5 country	varchar(255)	utf8mb4_general_ci	
<input type="checkbox"/>	6 city	varchar(255)	utf8mb4_general_ci	
<input type="checkbox"/>	7 zip	varchar(255)	utf8mb4_general_ci	
<input type="checkbox"/>	8 address	varchar(255)	utf8mb4_general_ci	
<input type="checkbox"/>	9 state	varchar(255)	utf8mb4_general_ci	
<input type="checkbox"/>	10 accuracy_of_data	varchar(256)	utf8mb4_general_ci	

Εικόνα 2.4 Πίνακας πληροφοριών τοποθεσίας

Για την τοποθεσία το πιο σημαντικό δεδομένο το οποίο θα παράγεται-παρέχεται πριν εισαχθεί μια καταστροφή στην βάση είναι οι συντεταγμένες. Οι συντεταγμένες δίνουν την μεγαλύτερη

ακρίβεια και από αυτές μπορούν να παραχθούν τα υπόλοιπα πεδία. Παρόλα αυτά δεν θα οριστούν ως απαραίτητα στον πίνακα καθώς δεν παρέχονται από όλες τις βάσεις.

- Γεωγραφικό μήκος (longitude): Πραγματικός αριθμός, που λαμβάνει τιμές στο κλειστό διάστημα [-180, 180].
- Γεωγραφικό πλάτος (latitude): Πραγματικός αριθμός, που λαμβάνει τιμές στο κλειστό διάστημα [-90, 90].
- Χώρα : Πεδίο ελεύθερου κειμένου όπου αναφέρεται το όνομα της χώρας στην οποία συνέβη ή την οποία επηρέασε το γεγονός.
- Πόλη: Πεδίο ελεύθερου κειμένου όπου αναφέρεται το όνομα της πόλης στην οποία συνέβη ή την οποία επηρέασε το γεγονός. Μπορούν να εισαχθούν περισσότερα από ένα ονόματα, χωρισμένα με κόμμα.
- Ακρίβεια τοποθεσίας: Πεδίο ελεύθερου κειμένου, όπου αναφέρεται η ακτίνα μέσα στην οποία βρίσκεται το επίκεντρο της καταστροφής, όταν αυτό δεν είναι επακριβώς γνωστό.
- Ταχυδρομικός Κώδικας: Ο ταχυδρομικός κώδικας της περιοχής στην οποία συνέβη η καταστροφή, αν φυσικά παρέχεται από την βάση.

2.3.4 Πίνακας χρονικών δεδομένων

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα	Ενέργεια
1	time_id	int(11)			Όχι	Καμία		AUTO_INCREMENT	Αλλαγή Διαγραφή Περισσότερα
2	start_time	timestamp			Όχι	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()	Αλλαγή Διαγραφή Περισσότερα
3	start_date	date			Ναι	NULL			Αλλαγή Διαγραφή Περισσότερα
4	duration	int(11)			Ναι	NULL			Αλλαγή Διαγραφή Περισσότερα

Εικόνα 2.5 Πίνακας χρονικών δεδομένων

Για τα χρονικά δεδομένα μια τιμή δεν είναι αρκετή για αυτό θα χρησιμοποιηθούν τα παρακάτω:

- Ώρα έναρξης/start time: Ο χρόνος του συμβάντος. Είναι απαραίτητο δεδομένο για όλες τις εγγραφές και αποτελεί πεδίο τύπου timestamp.
- Ημέρα έναρξης/start date: Ημέρα έναρξης του συμβάντος
- Διάρκεια: Η διάρκεια του συμβάντος, μετρίεται σε λεπτά και είναι προαιρετικό στην βάση.

2.3.5 Πίνακας συνέπειών

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα	Ενέργεια
<input type="checkbox"/>	1 id	int(11)			Όχι	Καμία			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	2 economic_esimated_cost	varchar(255)	utf8mb4_general_ci		Ναι	NULL			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	3 fatalities	varchar(255)	utf8mb4_general_ci		Ναι	NULL			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	4 affected	varchar(255)	utf8mb4_general_ci		Ναι	NULL			Αλλαγή Διαγραφή Περισσότερα
<input type="checkbox"/>	5 notes	varchar(255)	utf8mb4_general_ci		Ναι	NULL			Αλλαγή Διαγραφή Περισσότερα

Επιλογή όλων *Με τους επιλεγμένους:* Περίγηση Αλλαγή Διαγραφή Πρωτεύον Μοναδικό Ευρετήριο Χωρική Πλήρες
 Απομάκρυνση από κεντρικές στήλες

Εικόνα 2.6 Πίνακας συνεπειών

Σκοπός του συγκεκριμένου πίνακα είναι η καταγραφή δεδομένων για τις συνέπειες της καταστροφής. Είναι απαραίτητες πληροφορίες για την περαιτέρω μελέτη των φυσικών καταστροφών. Ο πίνακας αποτελείται από τα κάτωθι πεδία:

- Id: μοναδικό αναγνωριστικό
- Οικονομικές συνέπειες: Το συνολικό οικονομικό κόστος που επέφερε η φυσική καταστροφή.
- Θάνατοι: Ο συνολικός αριθμός θανάτων.
- Επηρεασμένοι: Ο αριθμός των ανθρώπων που επηρεάστηκαν.
- Σημειώσεις: Για αποθήκευση επιπλέον πληροφοριών.

2.4 ΚΑΤΑΣΚΕΥΗ ΤΗΣ ΒΑΣΗΣ ΜΕ ΓΝΩΜΟΝΑ ΤΟ ΣΧΗΜΑ ΤΗΣ

Για την κατασκευή του μοντέλου της βάσης ή σχήματος, είναι απαραίτητο να χρησιμοποιηθεί πλήρης οριοθέτηση και ορισμός των πεδίων της βάσης, καθώς και των συνδέσεων που θα υπάρξουν μεταξύ των πινάκων. Το [5] χρησιμοποιεί sql για την κατασκευή του δίνει την δυνατότητα παραγωγής κώδικα απευθείας από το σχήμα της βάσης. Αφού οριστεί πλήρως το σχήμα της βάσης δίνεται δυνατότητα εξαγωγής τις βάσεις είτε ως εικόνα-pdf (η οποία χρησιμοποιήθηκε για την αναπαράσταση της στην εικόνα), είτε κώδικα που μπορεί να χρησιμοποιηθεί για την υλοποίηση της βάσης που σχεδιάστηκε. Δίνεται πληθώρα επιλογών για το πλαίσιο των εργαλείων που αποφασίστηκε να χρησιμοποιηθούν στη διπλωματική θα γίνει εξαγωγή σε mysql. [8]

The screenshot shows a web-based database schema editor interface. The main area contains SQL code for creating several tables: hazards, hazard_type, location, event, and time. The code is as follows:

```

1  //// -- LEVEL 1
2  //// -- Tables and References
3
4  // Creating tables
5  Table hazards {
6    id int [pk, increment] // auto-increme
7    title varchar
8    created_at timestamp
9    hazard_type varchar
10   event_id int
11 }
12
13 Table hazard_type{
14   id int [pk, increment]
15   name ENUM ('earthquake', 'flood','tsu
16   Magnitude float
17   elevation float
18   radius float
19   notes varchar
20   earthquake_focus int
21 }
22
23
24 Table location {
25   location_id int [pk, not null]
26   name varchar [not null]
27   longitude int [not null]
28   latitude int [not null]
29   country varchar [not null]//maybe coun
30   city varchar
31   zip varchar
32   address varchar //street-number( or op
33   state varchar
34   accuracy_of_data float
35 }
36
37 Table event {
38   event_id int [pk]
39   time_id int [ref: - time.time_id]
40   location_id int
41   concequences_id int
42 }
43
44 Table time {
45   time_id int [pk, increment]
46   start_time timestamp //we have to def
47   start_date date
48   duration timestamp //optional
49
50
51

```

At the top right, there is a navigation bar with buttons for 'Share', 'History', and 'Export'. The 'Export' dropdown menu is open, showing the following options:

- Export to PDF
- Export to PostgreSQL
- Export to MySQL
- Export to SQL Server
- Export to PNG

On the right side, there is a preview window for the 'hazards' table, showing its columns:

hazards	
id	
title	
created_at	ti
hazard_type	
event_id	

Εικόνα 2.7 Επιλογές εξαγωγής του κώδικα της βάσης

Στο πλαίσιο της διπλωματικής επιλέχθηκε η MySQL για την επικοινωνία με την βάση. Ο παραχθείσας κώδικας είναι:

```

CREATE TABLE `hazards` (
  `id` int PRIMARY KEY AUTO_INCREMENT,
  `title` varchar(255),
  `created_at` timestamp,
  `hazard_type` varchar(255),
  `event_id` int
);

CREATE TABLE `hazard_type` (
  `id` int PRIMARY KEY AUTO_INCREMENT,
  `name` ENUM,
  `Magnitude` float,
  `elevation` float,
  `radius` float,
  `notes` varchar(255),
  `earthquake_focus` int
);

CREATE TABLE `location` (
  `location_id` int PRIMARY KEY NOT NULL,
  `name` varchar(255) NOT NULL,
  `longitude` int NOT NULL,
  `latitude` int NOT NULL,
  `country` varchar(255) NOT NULL,
  `city` varchar(255),
  `zip` varchar(255),
  `address` varchar(255),
  `state` varchar(255),
  `accuracy_of_data` float
);

CREATE TABLE `event` (
  `event_id` int PRIMARY KEY,
  `time_id` int,
  `location_id` int,
  `concequenses_id` int
);

CREATE TABLE `time` (
  `time_id` int PRIMARY KEY AUTO_INCREMENT,
  `start_time` timestamp,
  `start_date` date,
  `duration` timestamp
);

CREATE TABLE `concequenses` (
  `id` int PRIMARY KEY,
  `economic_esimated_cost` varchar(255),

```

```

`fatalities` varchar(255),
`affected` varchar(255),
`notes` varchar(255)
);

ALTER TABLE `event` ADD FOREIGN KEY (`time_id`) REFERENCES `time`
(`time_id`);

ALTER TABLE `location` ADD FOREIGN KEY (`location_id`) REFERENCES `event`
(`location_id`);

ALTER TABLE `event` ADD FOREIGN KEY (`event_id`) REFERENCES `hazards`
(`event_id`);

ALTER TABLE `hazard_type` ADD FOREIGN KEY (`id`) REFERENCES `hazards`
(`id`);

ALTER TABLE `consequences` ADD FOREIGN KEY (`id`) REFERENCES `event`
(`consequences_id`);

```

Πίνακας 4: Κατασκευή του σχήματος της βάσης

Όπως αναμενόταν παράχθηκαν και οι εντολές για την κατασκευή των πινάκων και των πεδίων τους αλλά και οι σχέσεις μεταξύ των πινάκων.

Ο κώδικας για την παραγωγή του σχήματος της βάσης:

```

//// -- LEVEL 1
//// -- Tables and References

// Creating tables

Table hazards {

    id int [pk, increment] // auto-increment

    title varchar

    created_at timestamp

    hazard_type varchar
    event_id int
}

```

Table hazard_type{

id int [pk, increment]

name ENUM ('earthquake', 'flood','tsunami','volcano')

Magnitude float

elevation float

radius float

notes varchar

earthquake_focus int

}

Table location {

location_id int [pk, not null]

name varchar [not null]

longitude int [not null]

latitude int [not null]

country varchar [not null]//maybe country_Id and a separated table for countries

city varchar

zip varchar

address varchar //street-number(or opposite)

state varchar

accuracy_of_data float

}

Table event {

event_id int [pk]

time_id int [ref: - time.time_id]

location_id int

concequences_id int

}

```

Table time {

time_id int [pk, increment]
start_time timestamp //we have to define the timezone GMT/UTC etc
start_date date
duration timestamp //optional

}
// Creating references
// You can also define relationship separately
// > many-to-one; < one-to-many; - one-to-one

Ref: event.location_id - location.location_id

Ref: hazards.event_id - event.event_id

Ref: hazards.id - hazard_type.id //assumption: flood+earthquake from same event would be
in separated rows

//-----//

//// -- LEVEL 2
//// -- Adding column settings

//-----//

//// -- Level 3
//// -- Enum, Indexes

Table concequences {

id int [pk, ref: - event.concequences_id]
economic_esimated_cost varchar

```

```
fatalities varchar  
affected varchar  
notes varchar  
  
}
```

Πίνακας 5: Κατασκευή της βάσης με χρήση του σχήματος.

3 ΥΛΟΠΟΙΗΣΗ ΤΗΣ ΒΑΣΗΣ, ΚΑΤΑΣΚΕΥΗ ΤΗΣ ΒΑΣΗΣ ΚΑΙ ΕΙΣΑΓΩΓΗ ΣΤΟΙΧΕΙΩΝ

3.1 ΣΥΝΔΕΣΗ ΜΕ ΒΑΣΗ.

Για την σύνδεση με την βάση χρησιμοποιήθηκε η βιβλιοθήκη Mysql.connector [6]. Η παραπάνω βιβλιοθήκη επιτρέπει στην Python να προσπελάσει Mysql βάσεις. Στην περίπτωσή μας έχουμε μια τοπική βάση οπότε θα συνδεθούμε σε localhost.

```
import mysql.connector

conn = mysql.connector.connect(
    user='root', host='127.0.0.1', database='hazards')
```

Εικόνα 3.1 Κώδικας σύνδεσης στην βάση

- User: Το όνομα του χρήστη που έχει σεταριστεί η βάση.
- Password(προαιρετικό): Σε περίπτωση που ο χρήστης έχει δημιουργηθεί με κωδικό πρέπει να περαστεί ως παράμετρος στον connector. Στην δικιά μου υλοποίηση επέλεξα να μην σετάρω τον user με κωδικό.
- Host: Το που βρίσκεται η βάση.
- Database: Το όνομα της βάσης.

Στο τέλος του κώδικα πρέπει να κλείνει η σύνδεση με την μέθοδο close(), ώστε να κλείσει η σύνδεση που δημιουργήθηκε ανάμεσα στην python στην βάση.

3.2 ΔΗΜΙΟΥΡΓΙΑ ΕΡΩΤΗΜΑΤΩΝ ΣΤΗΝ ΒΑΣΗ

Αφού έχει δημιουργηθεί η σύνδεση με την βάση μπορούμε να δημιουργήσουμε ερωτήματα (queries) στην βάση:

```
conn = mysql.connector.connect(
    user='root', host='127.0.0.1', database='test')

def insert_into_hazards(conn,data):
    #Creating a cursor object using the cursor() method
    cursor = conn.cursor()
    # Preparing SQL query to INSERT a record into the database.
    insert_stmt = (
        "INSERT INTO hazards(id,title, hazard_type)"
        "VALUES (%s, %s,%s)"
    )
    try:
        # Executing the SQL command
        cursor.execute(insert_stmt, data)
        # Commit your changes in the database
        conn.commit()
    except:
        # Rolling back in case of error
        conn.rollback()
    #print("Data inserted")
```

Εικόνα 3.2 Κώδικας εισαγωγής δεδομένων σε πίνακα της βάσης

Δημιουργούμε με την μέθοδο cursor πάνω στην σύνδεση που έχουμε φτιάξει έναν κέρσορα πάνω στην βάση που έχουμε ανοίξει. Στην συνέχεια δημιουργούμε ένα query ως αλφαριθμητικό.

3.3 ΥΠΟΣΤΗΡΙΖΟΜΕΝΑ ΕΡΩΤΗΜΑΤΑ

Μπορούμε να δημιουργήσουμε οποιοδήποτε query υποστηρίζεται από την MySQL, τα πιο βασικά από τα οποία είναι [5]:

1. Εισαγωγή

INSERT INTO

ΣΥΝΤΑΞΗ:

```
INSERT INTO TABLE NAME (Column1, Column2,..) VALUES (val1, val2..);
```

ΠΑΡΑΔΕΙΓΜΑ:

```
INSERT INTO EMPLOYEE (EMP_NAME, EMP_SAL) Values ('TOM', '3000');
```

Για την εισαγωγή στοιχείων στην βάση χρησιμοποιείτε το INSERT TO. Παράλληλα είναι απαραίτητο να δοθούν το όνομα του πίνακα, σε ποιες στήλες θέλουμε να εισάγουμε δεδομένα και τις τιμές τους.

2. Ανανέωση

Update [5]

ΣΥΝΤΑΞΗ:

```
UPDATE [TABLE NAME] SET COLUMN1 =' ' WHERE COLUMN2 =' ';
```

ΠΑΡΑΔΕΙΓΜΑ:

```
UPDATE EMPLOYEE SET EMP_SAL=6000 WHERE EMP_ID=200;
```

Αν δεν θέλουμε να δημιουργήσουμε καινούργια εισαγωγή αλλά θέλουμε να αλλάξουμε μια υπάρχουσα, μπορούμε να χρησιμοποιήσουμε την εντολή update της MySQL. Είναι απαραίτητο πάλι να εισάγουμε το όνομα του πίνακα, πληροφορίες για το ποιες εισαγωγές θέλουμε να ανανεώσουμε και τις επιθυμητές τιμές.

3. Διαγραφή

Delete

ΣΥΝΤΑΞΗ:

```
DELETE FROM [TABLE NAME] WHERE CONDITION;
```

ΠΑΡΑΔΕΙΓΜΑ:

```
DELETE FROM EMPLOYEE WHERE EMP_ID=154;
```

Χρησιμοποιείτε για διαγραφή μιας εγγραφής, είναι απαραίτητη μια συνθήκη για την επιλογή των στοιχείων που θα διαγραφούν. [5]

4. Επιλογή

Select [5]

ΣΥΝΤΑΞΗ:

```
SELECT DISTINCT [COLUMN NAME] FROM [TABLE NAME];
```

ΠΑΡΑΔΕΙΓΜΑ:

```
SELECT * FROM EMPLOYEE;
```

3.4 ΕΞΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΙΣ ΒΑΣΕΙΣ

Για την άντληση των δεδομένων χρησιμοποιήθηκαν οι βάσεις που αναλύθηκαν στο πρώτο κεφάλαιο . Αυτές οι βάσεις παρείχαν ποικιλία διεπαφών συγκεκριμένα

- RESTful APIs
- αρχεία τύπου csv
- αρχεία τύπου excel
- αρχεία τύπου rdf

Για όλες τις προαναφερθείσες διεπαφές έχει γίνει ξεχωριστή υλοποίηση στο περιβάλλον της Python με χρήση των κατάλληλων βιβλιοθηκών. Αφού εισαχθούν τα δεδομένα αποθηκεύονται σε λίστα τουπλών, δομή δεδομένων της Python. Έπειτα για κάθε βάση ξεχωριστά αναλύονται τα δεδομένα, γίνεται επιλογή των κατάλληλων πεδίων. Με τον τρόπο αυτό καθαρίζεται η πληροφορία από παραπλανητικά δεδομένα και συντηρούνται τα απαραίτητα στοιχεία των καταστροφών.

Αφού εισαχθούν τα δεδομένα των βάσεων και φιλτραριστούν τα πεδία τους, ακολουθεί η διαδικασία της εναρμόνισής τους. Καθώς τα δεδομένα προέρχονται από ετερογενείς πηγές και διαφορετικούς τρόπους άντλησης παρουσιάζουν διαφορές τόσο στο περιεχόμενο όσο και στον τρόπο αποθήκευσης . Για παράδειγμα υπάρχουν βάσεις που προσδιορίζουν την τοποθεσία με συντεταγμένες που προσδιορίζουν επακριβώς το σημείο του συμβάντος, αλλά υπάρχουν και βάσεις που προσδιορίζουν το σημείο με ένα πεδία ελεύθερου κειμένου που καθιστά μη προσδιορισμένο το ακριβές σημείο. Επίσης χαρακτηριστικό παράδειγμα διαφοράς στην μορφοποίηση είναι οι ημερομηνίες που μπορεί να ακολουθούν είτε ευρωπαϊκό είτε αμερικανικό πρότυπο.

Όλες οι παραπάνω διαφοροποιήσεις καθιστούν απαραίτητη την επεξεργασία και εναρμόνιση των δεδομένων πριν την εισαγωγή τους στην σχεσιακή βάση που δημιουργήθηκε.

4 ΕΝΑΡΜΟΝΙΣΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ

Για να εισαχθούν δεδομένα από διαφορετικές πηγές και μορφοποιήσεις σε μια ομοιογενή βάση είναι απαραίτητο αφού πραγματοποιηθεί η εξαγωγή των δεδομένων να γίνει εναρμόνιση τους και προσαρμογή τους στις προδιαγραφές της βάσης που δημιουργήσαμε. [7] Η εξομάλυνση των δεδομένων γίνεται σε δυο επίπεδα, αρχικά στην επιλογή των κατάλληλων πεδίων από τις υπάρχουσες βάσεις και έπειτα διαμόρφωση των πεδίων. [13]

4.1.1 Επιλογή των κατάλληλων πεδίων

Η πληρότητα των δεδομένων ήταν σημαντικός παράγοντας για την επιλογή άντλησης δεδομένων από μια βάση. Παρόλα αυτά σε αρκετές από τις βάσεις προσφέρονταν περισσότερα δεδομένα από αυτά που είναι αναγκαία. Για το λόγο αυτό έγινε επιλογή συγκεκριμένων πεδίων πριν εισαχθούν στην βάση. Σκοπός είναι να αποθηκεύουν τα δεδομένα σε μια κοινή βάση ώστε να είναι εφικτή η συνολική τους ανάλυση και η εξαγωγή έγκυρων συμπερασμάτων.

Τα βασικά στοιχεία στα οποία βασιστήκαμε για να αντλήσουμε δεδομένα για τις καταστροφές είναι:

Είδος καταστροφής: Η κατηγορία της φυσικής καταστροφής είναι απαραίτητη τόσο στην αποθήκευση των δεδομένων στην βάση, καθώς μετριέται και αποθηκεύεται διαφορετικά το μέγεθος και η δυναμική της καταστροφής ανάλογα την κατηγορία της, τόσο και στα συμπεράσματα που θα εξαχθούν.

Μέγεθος καταστροφής: Είναι απαραίτητη η μέτρηση της δυναμικής της καταστροφής, καθώς δίνει τη δυνατότητα φιλτραρίσματος των δεδομένων και καλύτερη εξαγωγή συμπερασμάτων σε αυτά. Αποθηκεύεται και μετριέται διαφορετικά για κάθε είδος καταστροφής.

Γεωγραφικά δεδομένα: Είναι απαραίτητο να έχουμε πλήρη γεωγραφικά δεδομένα για κάθε καταστροφή, είτε μέσω συντεταγμένων του σημείου, είτε με προσδιορισμό μέσω πεδίων ελεύθερου κειμένου της περιοχής, με πεδία όπως η χώρα, η πόλη και το σημείο της καταστροφής. Προφανώς οι συντεταγμένες αποτελούν την καλύτερη και πιο έγκυρη πηγή για προσδιορισμό του σημείου.

Χρονολογικά δεδομένα: Επίσης αποτελεί απαραίτητο στοιχείο της καταστροφής ο προσδιορισμός της χρονικής στιγμής που έγινε. Τα χρονικά πεδία των βάσεων ήταν και τα πεδία που χρειάστηκαν μετέπειτα την μεγαλύτερη εξομάλυνση στην διαμόρφωση των πεδίων καθώς προέρχονται από ποικίλες πηγές, με χρήση διαφορετικών προτύπων.

Δεδομένα συνεπειών: Τα δεδομένα συνεπειών αποτελούν προαιρετικό στοιχείο για τις εγγραφές καθώς δεν παρέχονται από όλες τις πηγές. Μας ενδιαφέρουν συνέπειες τόσο στον άνθρωπο με θανάτους και επηρεασμένους ανθρώπους όσο και στις οικονομικές επιπτώσεις που επέφερε η καταστροφή.

4.1.2 Εξομάλυνση της διαμόρφωση των πεδίων

Γίνανε πολλές μετατροπές των δεδομένων πριν την εισαγωγή τους στην σχεσιακή βάση δεδομένων που δημιουργήθηκε. Μερικές από τις πιο βασικές παρουσιάζονται παρακάτω

Δεδομένα ημερομηνιών

Σύνταξη ημερομηνιών: Θα ακολουθηθεί το ευρωπαϊκό πρότυπο στην σύνταξη [7]. Δηλαδή είναι απαραίτητο να παρέχονται με την σειρά ημέρα/μήνας/έτος. Επίσης, αποδεκτό διαχωριστικό πρέπει να αποτελεί το σύμβολο “-”.

Δεδομένα χρόνου

Τα δεδομένα χρόνου είναι επίσης ένα πεδίο που θα πρέπει να τηρηθούν αυστηρά κάποια κριτήρια μορφοποίησης. Στα δεδομένα χρόνου είναι σημαντικό να ακολουθούνται οι παρακάτω προδιαγραφές πριν αυτά εισαχθούν στην βάση.

- Ακρίβεια: Η ακρίβεια μέτρησης της τιμής. Για τους σκοπούς της διπλωματικής επιλέχθηκε να εισάγονται δεδομένα στην βάση με ακρίβεια δευτερολέπτου.
- Αναπαράσταση σε σύστημα 12 ή 24 ωρών: Επιλέχθηκε το σύστημα 24 ωρών με ορισμό ως 0 τα μεσάνυχτα.
- Ζώνη ώρας: Όλα τα δεδομένα μετατρέπονται και αποθηκεύονται σε UTC ζώνη ώρας.
- Διαχωρισμός: Ο διαχωρισμός γίνεται με το σύμβολο ‘:’.

Nan τιμές

Η μορφοποίηση των κενών/nan τιμών επηρεάζει την εκτέλεση των queries. Πολλές φορές ενώ παρέχονταν κενές τιμές ως είσοδο από άλλη βάση αυτές δεν μπορούσαν να μεταφραστούν σωστά για εισαγωγή. Για τον λόγο αυτό δημιουργήσαμε ένα εξτρά επίπεδο ελέγχου που φιλτράρει τα δεδομένα και αν είναι απαραίτητο μετατρέπει το float nan σε πεδίο τύπου str. Έπειτα είναι δυνατή η εισαγωγή των δεδομένων στην βάση.

Παρακάτω παρουσιάζονται μερικές από τις συναρτήσεις που χρησιμοποιήθηκαν για τις μετατροπές.

```
import pandas as pd
import numpy as np
import csv
```

```

def timestamp_conversion(data_time):
    a=data_time.split(' ')
    b=a[0].split('/')
    date=b[2] + '-' + b[0] + '-' + b[1]
    #print(date)
    if (a[2]=='AM'):
        time=a[1]
    if (a[2]=='PM'):
        b=a[1].split(':')
        b[0]=str(int(b[0])+12)
        if(b[0]=='24'):
            b[0]='00'
        time=b[0]+'-'+b[1]+'-'+b[2]
    return time, date

```

```

def data_check(data):
    data1=list(data)
    j=0
    for i in data1:
        if (data1[j] != data1[j]): #einai float nan
            data1[j]=str(data1[j])
        j=j+1
    return data1

```

Πίνακας 6: Εναρμόνιση των δεδομένων

5 ΕΡΓΑΛΕΙΑ ΓΙΑ ΤΗΝ ΑΝΑΠΤΥΞΗ ΤΗΣ ΒΑΣΗΣ ΚΑΙ ΤΗΝ ΥΛΟΠΟΙΗΣΗ

Για την υλοποίηση της εφαρμογής επιλέχθηκε ως κύριο εργαλείο η γλώσσα προγραμματισμού Python 3.10. Κύριο κριτήριο ήταν η ευκολία επεξεργασίας δεδομένων που προσφέρει η γλώσσα καθώς και η ικανότητα της σε μεγάλο όγκο δεδομένων.

Ως βασικός ολοκληρωμένο περιβάλλον ανάπτυξης χρησιμοποιήθηκε το Visual studio code. Είναι ίσως το πιο διαδεδομένο ολοκληρωμένο περιβάλλον ανάπτυξης αυτήν την στιγμή και υποστηρίζει την μεγαλύτερη πληθώρα γλωσσών προγραμματισμού.

Επιπλέον, χρησιμοποιήθηκε SQL για το χτίσιμο του σχήματος της βάσης, MySQL για την κατασκευή της βάσης με βάση το σχήμα της. Επίσης, MySQL χρησιμοποιήθηκε για όλα τα queries που γίνανε στην βάση μέσω της Python με την χρήση της βιβλιοθήκης Mysql.connector . Επίσης το FastAPI αποτελεί και αυτό ένα framework βασισμένο στην Python.

5.1.1 Microsoft Visual Studio

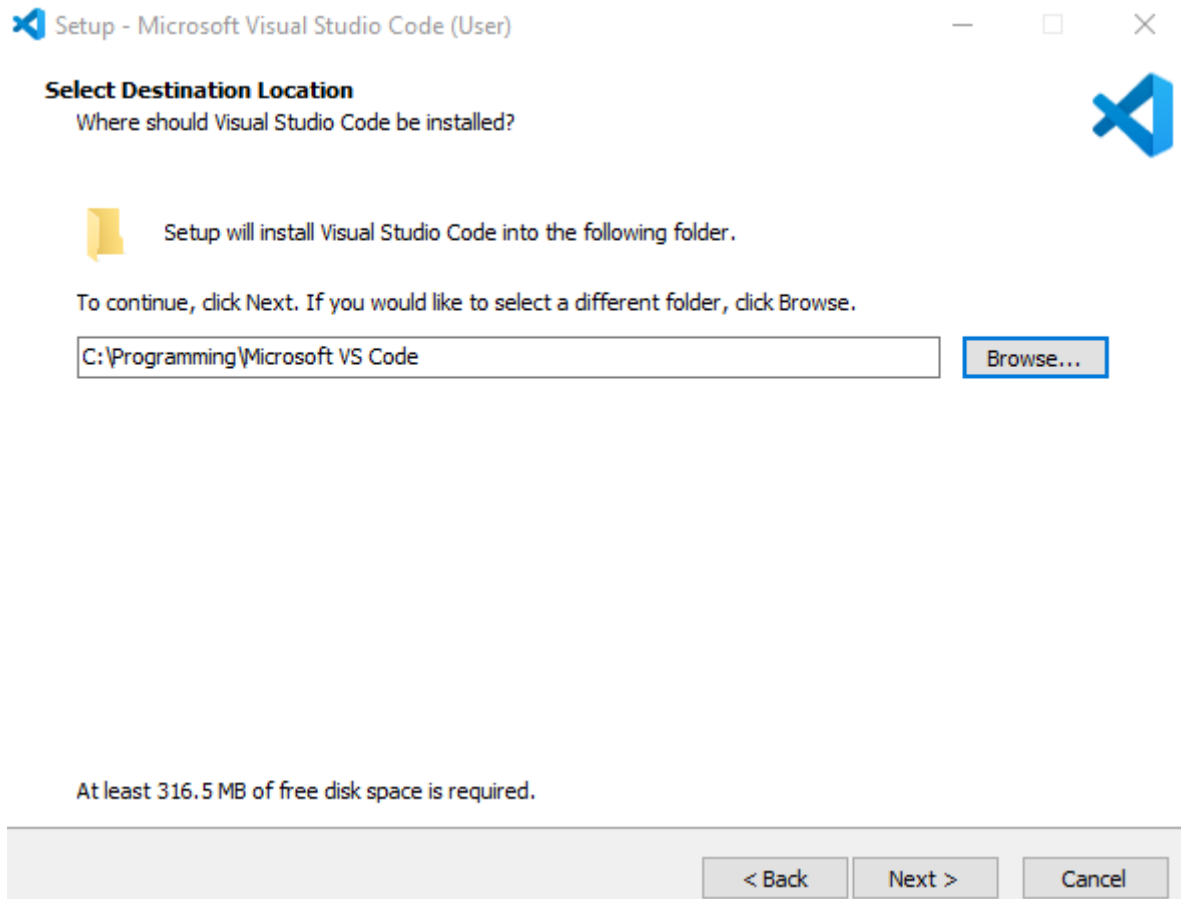
Το Microsoft Visual Studio [7]είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης από τη Microsoft που χρησιμοποιείται για την δημιουργία προγραμμάτων υπολογιστών, καθώς και ιστοσελίδων, εφαρμογών ιστού, υπηρεσιών ιστού και εφαρμογών για κινητά. Το Visual Studio χρησιμοποιεί πλατφόρμες ανάπτυξης λογισμικού της Microsoft όπως το Windows API, τα Windows Forms, το Windows Presentation Foundation, το Windows Store και το Microsoft Silverlight. Μπορεί να παραχωρήσει τόσο εγγενή όσο και διαχειριζόμενο κώδικα. [8]

Είναι λογισμικό που διατίθεται ελεύθερα και υποστηρίζει 36 διαφορετικές γλώσσες προγραμματισμού. Οι ενσωματωμένες γλώσσες περιλαμβάνουν C,^[5] C++, Visual Basic, .NET, C#, F#, JavaScript, TypeScript, XML, XSLT, HTML και CSS. Η υποστήριξη για άλλες γλώσσες όπως Python, Ruby, Node.js και M μεταξύ άλλων είναι διαθέσιμη μέσω . Η Java (και το J#) υποστηρίζονταν στο παρελθόν.

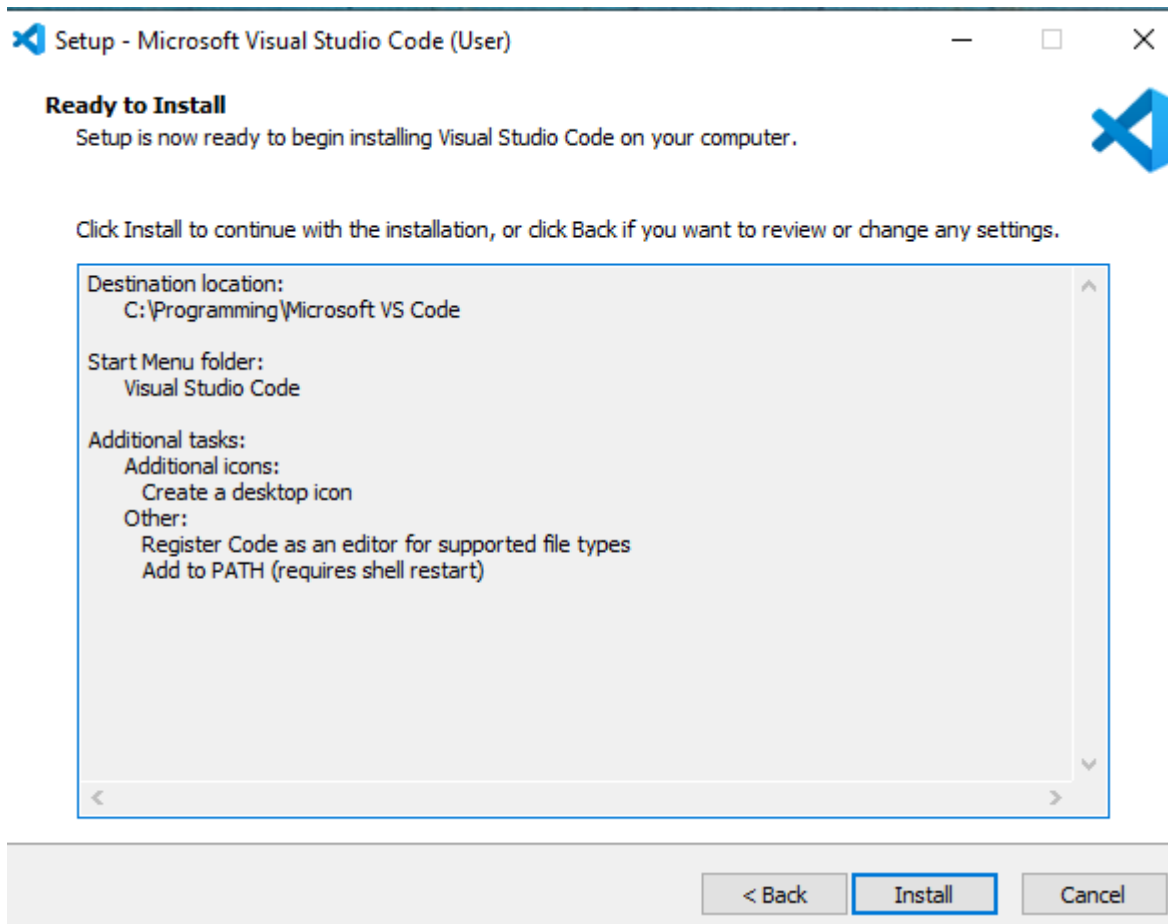
Επιπλέον το Visual Studio περιέχει ένα πρόγραμμα επεξεργασίας κώδικα που ενσωματώνει το IntelliSense όπως και την αναδιαμόρφωση κώδικα. Το ενσωματωμένο πρόγραμμα εντοπισμού σφαλμάτων ενεργεί τόσο σε επίπεδο πηγής όσο και σε επίπεδο μηχανής. Επίσης, ενσωματωμένα εργαλεία περιέχουν έναν προγραμματιστή κώδικα, έναν σχεδιαστή για τη δημιουργία εφαρμογών GUI, τον σχεδιαστή ιστοσελίδων, τον σχεδιαστή κλάσεων και τον σχεδιαστή σχημάτων βάσης δεδομένων.

Διαδικασία εγκατάστασης

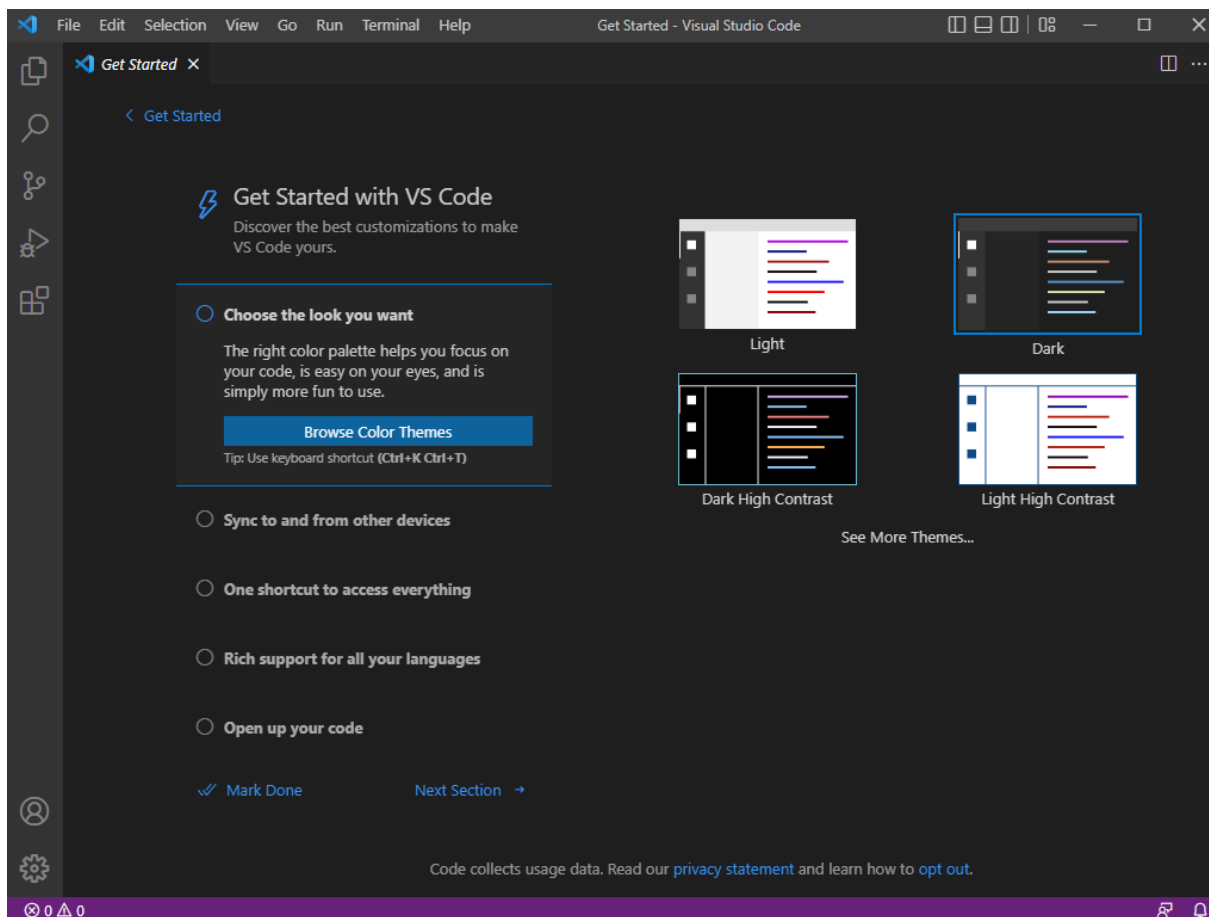
Για να γίνει εγκατάσταση του λογισμικού είναι απαραίτητο να επισκεφτούμε την επίσημη ιστοσελίδα διάθεσης του λογισμικού [8], να κατευνάσουμε το εκτελέσιμο και να ακολουθήσουμε τα απαραίτητα βήματα.



Εικόνα 5.1 Μονοπάτι εγκατάστασης



Εικόνα 5.2 Επιπλέον επιλογές



Εικόνα 5.3 Τελικό περιβάλλον χρήστη

Το περιβάλλον της εφαρμογής είναι αρκετά επεξηγηματικό, προσφέροντας στο χρήστη ποικίλες επιλογές για εκτέλεση και επεξεργασία των αρχείων. Σημαντικό πλεονέκτημα αποτελεί και η ενσωμάτωση τερματικού στο περιβάλλον χρήστη.

5.1.2 Python

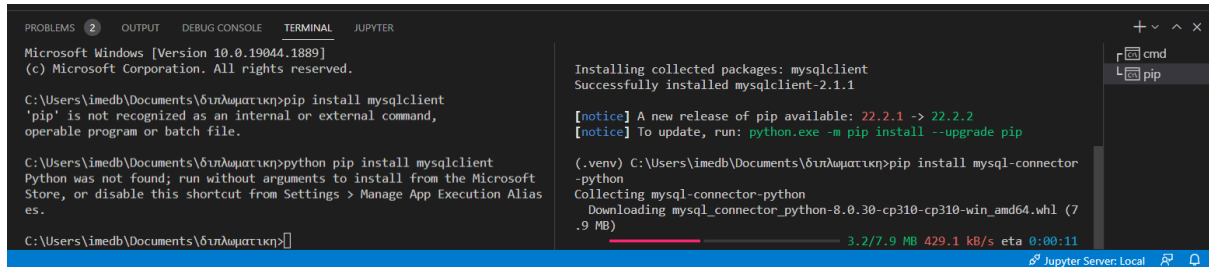
Η Python είναι μια γλώσσα προγραμματισμού που αναπτύχθηκε το 1991 από τον Ολλανδό Γκίντο βαν Ρόσσουμ (Guido van Rossum) στο ερευνητικό κέντρο Centrum Wiskunde & Informatica (CWI), αποτελεί μια γλώσσα υψηλού επιπέδου και ένα από τα βασικά χαρακτηριστικά της είναι πως εκτελείται με διερμηνευτή. Για να γίνει εγκατάσταση της Python, επισκεπτόμαστε την επίσημη ιστοσελίδα [9]. Επιλέχθηκε η τελευταία έκδοση της, Python 3.10.6.

5.1.3 Εγκατάσταση βιβλιοθηκών

Για την εκτέλεση της διπλωματικής χρησιμοποιήθηκε ικανός αριθμός βιβλιοθηκών. Ως χαρακτηριστικό παράδειγμα για την διαδικασία της εγκατάστασης παρακάτω αναλύεται η εγκατάσταση της βιβλιοθήκης Pandas [10].

Pandas

1. Άνοιγμα του τερματικού/cmd.
2. Πληκτρολόγηση και εκτέλεση `python -m pip install pandas`.
3. Επανεκκίνηση του visual studio code. [11]



```
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\imedb\Documents\δουλωματικη>pip install mysqlclient
'pip' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\imedb\Documents\δουλωματικη>python pip install mysqlclient
Python was not found; run without arguments to install from the Microsoft
Store, or disable this shortcut from Settings > Manage App Execution Alias
es.

C:\Users\imedb\Documents\δουλωματικη>

Installing collected packages: mysqlclient
Successfully installed mysqlclient-2.1.1

[notice] A new release of pip available: 22.2.1 -> 22.2.2
[notice] To update, run: python.exe -m pip install --upgrade pip

(.venv) C:\Users\imedb\Documents\δουλωματικη>pip install mysql-connector
-python
Collecting mysql-connector-python
  Downloading mysql_connector_python-8.0.30-cp310-cp310-win_amd64.whl (7
  .9 MB)
3.2/7.9 MB 429.1 kB/s eta 0:00:11
```

Εικόνα 5.4 Εγκατάσταση βιβλιοθηκών στο τερματικό

6 ΔΙΑΣΥΝΔΕΣΗ ΤΗΣ ΒΑΣΗΣ ΜΕ ΔΙΕΠΑΦΕΣ (RESTFUL APIS)

6.1 ΔΙΕΠΑΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΕΦΑΡΜΟΓΩΝ

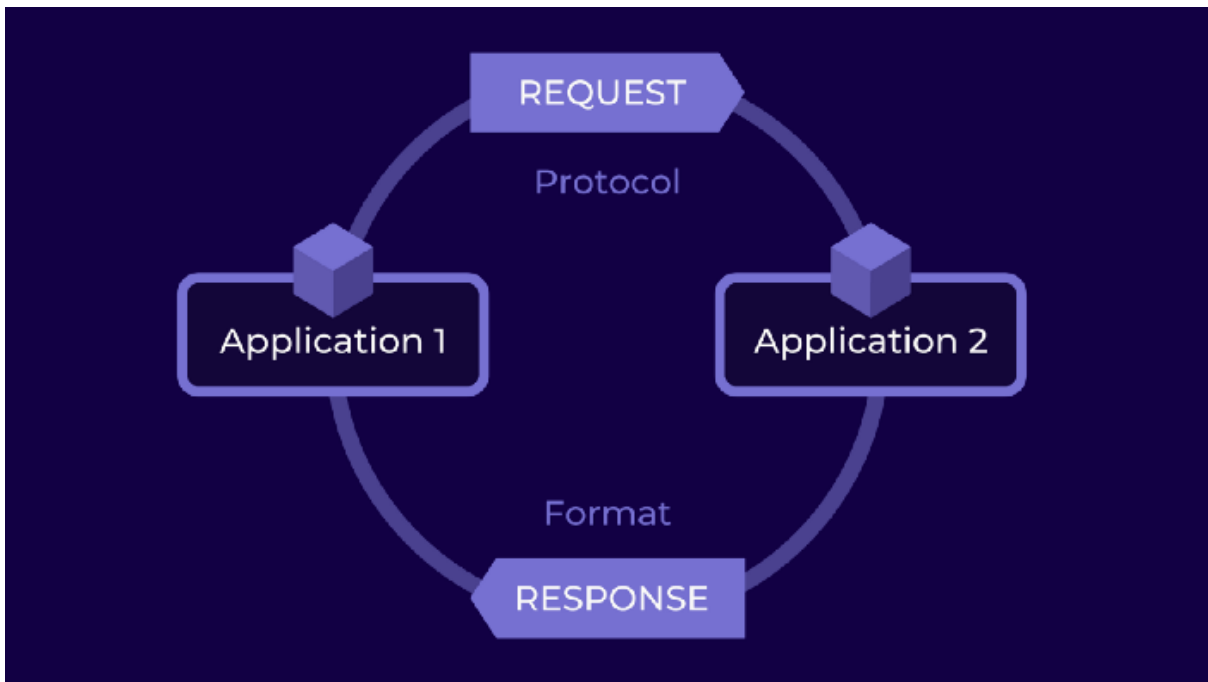
Η Διασύνδεση Προγραμματισμού Εφαρμογών (ή Διεπαφή Προγραμματισμού Εφαρμογών) είναι η διεπαφή των προγραμματιστικών διαδικασιών που προσφέρει ένα αποτελεσματικό σύστημα, βιβλιοθήκη ή εφαρμογή προκειμένου να επιτρέψει να γίνονται προς αυτά αίτηση από άλλα προγράμματα ή και εναλλαγή δεδομένων.

Ένας από τους βαθύτερους σκοπούς μίας διεπαφής είναι να εκθέτει το σύνολο των λειτουργιών (υπηρεσιών) που μπορεί να παρέχει μια βιβλιοθήκη ή ένα λειτουργικό σύστημα σε άλλα προγράμματα, χωρίς να παραχωρεί πρόσβαση στον κώδικα που υλοποιεί αυτές τις υπηρεσίες. Η Διεπαφή, ένα συμβόλαιο κλήσης μεταξύ καλούντος και καλούμενου, διαχωρίζει την προγραμματιστική υλοποίηση κάποιων υπηρεσιών από τη χρήση τους

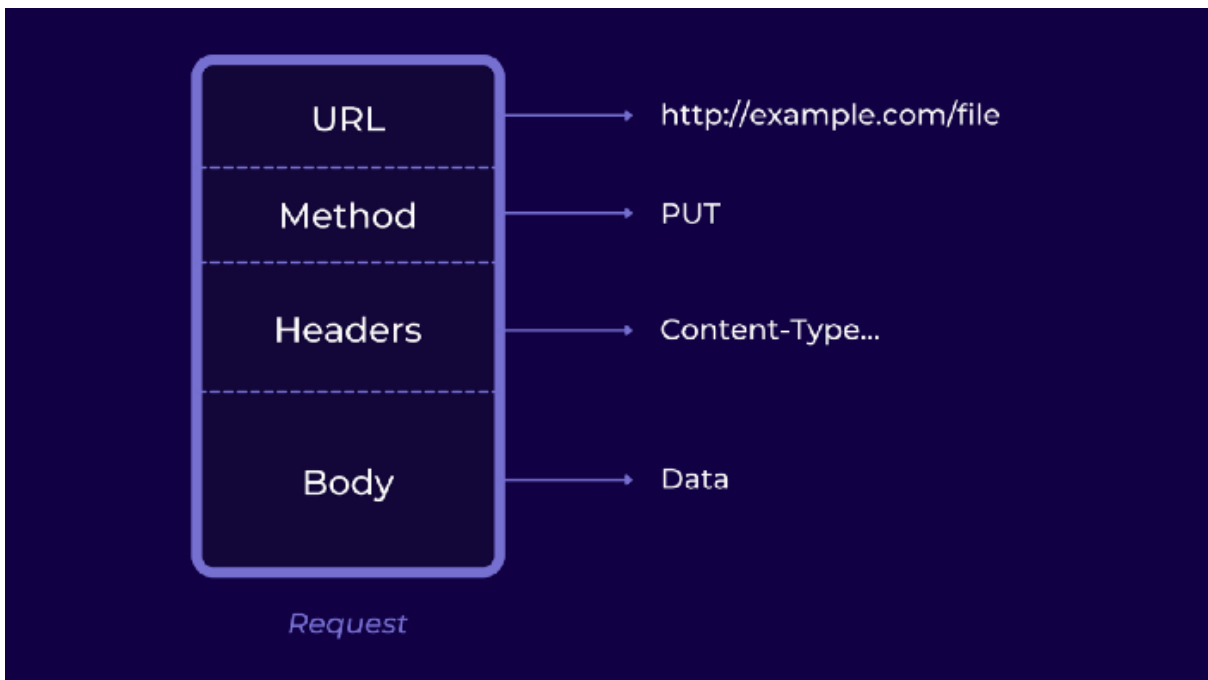
6.2 ΗΤΤΡ ΠΡΩΤΟΚΟΛΛΟ

6.2.1 Ορισμός

Το **Πρωτόκολλο Μεταφοράς Υπερκειμένου** [12] αποτελεί το πιο συνηθισμένο πρωτόκολλο για την επικοινωνία στις υπηρεσίες λογισμικού. Χρησιμοποιήθηκε αρχικά το 1965 από τον Τεντ Νέλσον και η γενική ιδέα του αναπτύχθηκε μαζί με την γλώσσα προγραμματισμού HTML. Πάνω στις βασικές του αρχές θα στηριχθούν και οι διεπαφές που θα υλοποιήσουμε, καθώς και όλες οι επικοινωνίες του χρήστη από και προς την βάση. [15]



Εικόνα 6.1 Ανάλυση του πρωτοκόλλου HTTP



Εικόνα 6.2 Η δομή του μηνύματος

6.2.2 Μέθοδοι αίτησης του HTTP

Παρακάτω παρουσιάζονται οι μέθοδοι αίτησης του πρωτοκόλλου HTTP [16]:

GET

Η μέθοδος GET ζητά από το διακομιστή να στείλει τη σελίδα.

HEAD

Η μέθοδος HEAD ζητά μόνο την κεφαλίδα του μηνύματος, χωρίς τη πραγματική σελίδα.

POST

Η μέθοδος POST απασχολείται κατά την υποβολή φορμών. Ύστερα ο διακομιστής μπορεί να κάνει κάτι με αυτά τα δεδομένα ανάλογα με το URL. Ακολούθως η μέθοδος επιστρέφει μια σελίδα που εμφανίζει το αποτέλεσμα.

PUT

Η μέθοδος PUT απασχολείται με τις αντίθετες λειτουργίες της GET, δημιουργεί τη σελίδα αντί να την διαβάσει. Βοηθάει θετικά την κατασκευή μιας συλλογής ιστοσελίδων σε έναν μακρινό διακομιστή καθώς το σώμα της αίτησης παρέχει τη σελίδα

DELETE

Η μέθοδος DELETE καταργεί τη σελίδα.

TRACE

Η μέθοδος TRACE απασχολείται για την αποσφαλμάτωση και απαιτεί από τον διακομιστή να επαναφέρει την αίτηση.

CONNECT

Η μέθοδος CONNECT παραχωρεί στον χρήστη την άδεια να πραγματοποιήσει σύνδεση με έναν διακομιστή Ιστού μέσω μιας ενδιάμεσης συσκευής.

OPTIONS

Η μέθοδος OPTIONS επιστρέφει τις μεθόδους και τις κεφαλίδες που μπορούν να χρησιμοποιηθούν με αυτή τη σελίδα.

6.2.3 Κωδικοί στις απαντήσεις

Μετά την αποστολή του αιτήματος λαμβάνεται απάντηση από τον παραλήπτη. Οι απαντήσεις αυτές περιέχουν κωδικούς από 100 έως 599. Ανάλογα τον κωδικό της απάντησης παίρνουμε πληροφορίες για την πορεία του αιτήματος.

- **100–199:** Απάντηση
- **200–299:** Επιτυχία
- **300–399:** Υποδήλωση πως το αίτημα πρέπει να εκτελεστεί αλλού
- **400–499:** Σφάλμα του χρήστη
- **500–599:** Σφάλμα στον σέρβερ

6.3 RESTFUL ΔΙΕΠΑΦΕΣ

Αφού τα δεδομένα ομογενοποιηθούν, μορφοποιηθούν και εισαχθούν στην βάση, είναι απαραίτητο να δημιουργηθούν μερικές διεπαφές για την επικοινωνία της βάσης με τον υπόλοιπο κόσμο. Για την επικοινωνία αυτή υλοποιούνται μερικά RESTful APIs[17]με χρήση του FastAPI framework που θα αναλυθεί παρακάτω.

Τα RESTful APIs υποστηρίζουν τις παρακάτω μεθόδους του πρωτοκόλλου http:

- GET
- POST
- PUT
- PATCH
- DELETE

Στα πλαίσια της εκπόνησης της διπλωματικής θεωρήθηκε απαραίτητη μόνο η δυνατότητα άντλησης δεδομένων και όχι δυνατότητες για εισαγωγή ή ανανέωση δεδομένων. Για τον λόγο αυτό δημιουργήθηκαν διεπαφές μόνο για την χρήση της μεθόδου get. Μελλοντικά υπάρχει δυνατότητα επέκτασης του μοντέλου για χρήση περισσότερων διεπαφών.

Οι διεπαφές που δημιουργήθηκαν παρέχουν δυνατότητα είτε άντλησης όλων των δεδομένων της βάσης, είτε με συγκεκριμένα κριτήρια. Παρακάτω αναλύονται όλες οι διεπαφές. Επίσης, αυτές επεξηγούνται και στο swagger file που δημιουργήθηκε από το FastAPI και είναι προσπελάσιμο τοπικά στην διεύθυνση **http://127.0.0.1:8000/docs**.

Για την προσπέλαση τους δεν απαιτείται τοκεν ταυτοποίησης, είναι απλά απαραίτητη η χρήση του κατάλληλου url/συνδέσμου. Ανάλογα την διεπαφή ο σύνδεσμος περιέχει και τις παραμέτρους που χρειάζονται για την φιλτραρισμένη παροχή των δεδομένων.

6.4 ΥΛΟΠΟΙΗΣΗ ΤΩΝ ΔΙΑΠΑΦΩΝ

6.4.1 Εγκατάσταση

Για την εγκατάσταση του FastAPI [14]εκτελούνται οι παρακάτω εντολές στο τερματικό:

```
pip install fastapi
pip install "uvicorn[standard]"
```

Με τις παραπάνω εντολές εγκαθιστάμε επίσης και το uvicorn.

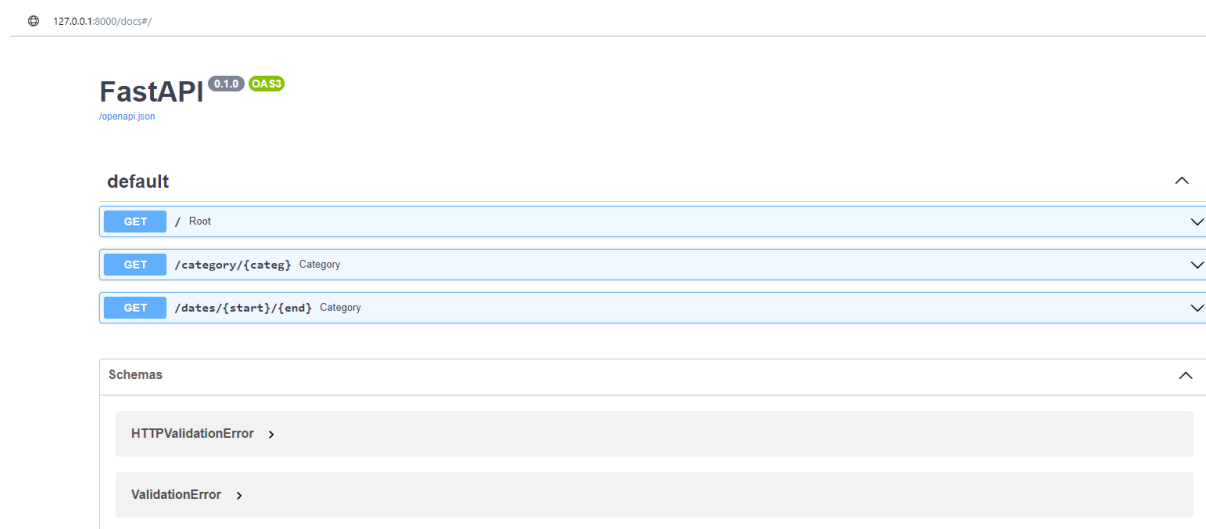
6.5 ΜΕΤΑΤΡΟΠΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΣΕ JSON

Η κλήση στην διεπαφή πρέπει να μεταφραστεί σε κλήση στην βάση. Έπειτα είναι απαραίτητο η μετάφραση των δεδομένων σε JSON [15]. **JavaScript Object Notation** ή **JSON** είναι ένας τρόπος κωδικοποίησης πληροφορίας για να χρησιμοποιηθεί σε κλήσεις του πρωτοκόλλου HTTP.

6.6 ΔΙΕΠΑΦΕΣ ΠΟΥ ΔΗΜΙΟΥΡΓΗΘΗΚΑΝ

Για τα πλαίσια της διπλωματικής δημιουργήθηκαν 3 διεπαφές για την άντληση των δεδομένων από την βάση. Πληροφορίες για τις διεπαφές δημιουργούνται αυτόματα από FastAPI και είναι προσπελάσιμες στην διεύθυνση:

<http://127.0.0.1:8000/docs/>



Εικόνα 6.3 Σύνολο διεπαφών που υλοποιήθηκαν

Ως ρίζα ορίζεται η διεύθυνση <http://127.0.0.1:8000/>, φυσικά η διεύθυνση αυτή είναι δυνατόν να τροποποιηθεί. Όλες οι επιπλέον διεπαφές που δημιουργούνται είναι προέκταση της διεύθυνσης αυτή της ρίζας.

Get μέθοδος στο Root

Δεν δέχεται παραμέτρους και επιστρέφει όλα τα δεδομένα της βάσης ως αντικείμενα JSON. Σκοπός της μέθοδο είναι η συνολική άντληση των δεδομένων από τη βάση ανεξάρτητα από τις τιμές των χαρακτηριστικών το δεδομένο.

Get μέθοδος στο Root/category/{categ}

Δέχεται ως παράμετρο το είδος της καταστροφής και επιστρέφει όλα τα δεδομένα του συγκεκριμένου είδους. προσφέρεται καλύτερη εποπτεία των δεδομένων καθώς αυτά παρέχονται ομοιόμορφα και αφορούν μόνο το επιλεγμένο είδος καταστροφής. είναι σημαντικό να αναδειχθεί ότι τυχόν λάθη στον προσδιορισμό το αλφαριθμητικού της κατηγορίας δεν γίνονται αντιληπτά και δεν διορθώνονται αυτόματα. Δηλαδή σε περίπτωση που χρήστης καλέσει στη διεπαφή γράφοντας λάθος το όνομα της κατηγορίας δεν θα λάβει δεδομένα,

Get μέθοδος στο Root/dates/{start}/{end}

Φιλτράρει τα δεδομένα επιστροφής με βάση τις ημερομηνίες διεξαγωγής της καταστροφής. Προσφέροντας έτσι στόμα χρήστη δυνατότητα φιλτραρίσματος τους με βάση τη χρονική περίοδο που θέλει να μελετήσει τέλειο

```
from fastapi import FastAPI
import json

import mysql.connector

def get_all(conn):
    cursor10 = conn.cursor(buffered=True)
    query = ("SELECT * FROM hazards\
INNER JOIN hazard_type ON hazards.id = hazard_type.id\
INNER JOIN location ON hazards.id = location.location_id\
INNER JOIN time ON hazards.id = time.time_id\
INNER JOIN concequences ON hazards.id = concequences.id\
#LIMIT 10\

")
    cursor10.execute(query)
    objects_list = []
    result1=cursor10.fetchall()
    for i in result1:
        id=i[0]
        name=i[1]
        address=i[19]
        category=i[3]
        latitude=i[15]
        longtitude=i[14]
        country=i[16]
```

```

    city=i[17]
    zip=i[18]
    deaths=i[28]
    affected=i[29]
    date=i[23]
    json_data={"id":id,"name":name,"category":category,"date":date,"address":ad-
dress,"longtitude":longtitude,"latitude":latitude,"country":coun-
try,"city":city,"zip":zip,"deaths":deaths,"affected":affected}

    objects_list.append(json_data)
return objects_list

app = FastAPI()

@app.get("/")
async def root():
    conn = mysql.connector.connect(
        user='root', host='127.0.0.1', database='hazards')
    data=get_all(conn)
    conn.close()
    return data

```

Πίνακας 7: Κώδικας διεπαφής

6.7 ΑΠΟΤΕΛΕΣΜΑΤΑ

Πέρα από την κατασκευή των δια παθών είναι σημαντικό να αναλύσουμε και τα αποτελέσματα, δηλαδή τι θα επιστραφεί στο χρήστη όταν αυτές καλεστούν. αρχικά είναι σημαντικό να θέσουμε σε λειτουργία τις διεπαφές, διαδικασία αυτή φαίνεται παρακάτω. Για να ενεργοποιήσουμε τις διεπαφές, εκτελούμε την εντολή στο τερματικό:

```

Uvicorn <όνομα αρχείου> :app --reload

```

```

INFO:      Started server process [26032]
INFO:      Waiting for application startup.
INFO:      Application startup complete.
INFO:      127.0.0.1:57613 - "GET / HTTP/1.1" 200 OK
INFO:      127.0.0.1:58432 - "GET /docs HTTP/1.1" 200 OK
INFO:      127.0.0.1:58432 - "GET /openapi.json HTTP/1.1" 200 OK
INFO:      127.0.0.1:53599 - "GET / HTTP/1.1" 200 OK

```

Εικόνα 6.4 Εκτέλεση του Fast API

Αφού εκτελεστεί η εντολή, λαμβάνουμε πληροφορίες στο τερματικό αν ενεργοποιήθηκε επιτυχώς. Το «200 OK» σύμφωνα με τους κωδικούς του HTTP πρωτοκόλλου που αναλύθηκε σε προηγούμενο κεφάλαιο δηλώνει την επιτυχή δημιουργία των διασαφών. Φυσικά αυτό δηλώνει πως η διεπαφή ενεργοποιήθηκε επιτυχώς και δεν διασφαλίζει την ποιότητα των δεδομένων που θα επιστραφούν. Ένα παράδειγμα των δεδομένων που αποστέλλονται είναι:

```
"id": 57,  
"name": "Mudslide crushed the Sankay Inn",  
"category": "landslide",  
"date": "2010-01-01T12:00:00",  
"address": "Rio de Janeiro state",  
"longtitude": -44.322498322,  
"latitude": -23.013744354,  
"country": "Brazil",  
"city": null,  
"zip": null,  
"deaths": "22.0",  
"affected": "50.0"  
},
```

Εικόνα 6.5 Τα δεδομένα που αποστέλλονται

Το παραπάνω αντικείμενο, αποτελεί χαρακτηριστικό παράδειγμα των δεδομένων που εξάγονται από τη βάση και της μορφής που αυτά αναπαριστούνται. το πρώτο πεδίο, id αποτελεί μοναδικό χαρακτηριστικό για τον προσδιορισμό της εγγραφής στη βάση διαχωρίζει τις εγγραφές μεταξύ τους και προσφέρει μια αναφορά σε κάθε εγγραφή. Το δεύτερο πεδίο, προσδιορίζει το όνομα που έχει δοθεί στην φυσική καταστροφή. Σκοπός του ονόματος είναι επίσης ο προσδιορισμός καταστροφής δίνοντας μερικά επιπλέον χαρακτηριστικά στοιχεία αυτής. Δεν θα μπορούσε να παραληφθεί η κατηγορία της καταστροφής η οποία στο παράδειγμα μας είναι η κατολίσθηση. στη συνέχεια παρατηρούμε πληροφορίες για την ημερομηνία και το σημείο πραγματοποίησης καταστροφής για την ημερομηνία παίρνουμε την ημέρα και την ώρα. για το τόπο της καταστροφής προσφέρουν περισσότερα δεδομένα όπως η διεύθυνση το οποίο είναι πεδίο ελεύθερο κείμενο που προσδιορίζει τη διεύθυνση του σημείου. Επίσης συντεταγμένες γεωγραφικό πλάτος και γεωγραφικό μήκος για τον ακριβή προσδιορισμό του σημείου. Επιπλέον είναι απαραίτητο να προσδιοριστεί η χώρα της καταστροφής. Στη συνέχεια παρατηρούμε το όνομα της πόλης καθώς και το ταχυδρομικό κώδικα, είναι δεδομένα που δεν υπάρχουν σε όλες τις καταστροφές και όπως φαίνεται χαρακτηριστικά στο παράδειγμά μας μπορεί να είναι κενά. αυτό φυσικά δεν επηρεάζει στην εξαγωγή των αποτελεσμά-

των καθώς μπορεί να προσδιορισθεί ο τόπος του συμβάντος από τα υπόλοιπα χαρακτηριστικά της εγγραφής. Τέλος παρατηρούμε τα πεδία θανάτων και επηρεασμένων ανθρώπων δίνοντάς μας μια καλή εποπτεία για τις συνέπειες καταστροφής στον άνθρωπο.

7 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΟΠΤΙΚΕΣ

7.1 ΣΥΜΠΕΡΑΣΜΑΤΑ

Δεδομένα φυσικών καταστροφών διατίθενται από εκατοντάδες πηγές, με διαφορετικά πρότυπα συλλογής και αποθήκευσης. Είναι εμφανής η απουσία μιας ενιαίας βάσης που θα συγκεντρώνει όλους τους τύπους των καταστροφών με τις βασικές τους πληροφορίες. Θεμελιώδη δεδομένα, όπως αναλύθηκε στα αρχικά κεφάλαια, θεωρούμε δεδομένα για το μέγεθος της καταστροφής, τον τόπο διεξαγωγής της, τον ακριβή χρόνο και την διάρκεια(αν δεν είναι ακαριαίο το φαινόμενο) και της συνέπειες που προέκυψαν, είτε σε ανθρώπινο κόστος είτε σε οικονομικό.

Ένα από τα βασικότερα συμπεράσματα έπειτα από εκτενή μελέτη των δεδομένων φυσικών καταστροφών είναι οι ομοιότητες που εντοπίστηκαν σε δεδομένα διαφορετικών καταστροφών. Έχουμε την δυνατότητα να αναλύσουμε τις καταστροφές σε κοινή βάση ανεξάρτητα από τη φυσική προέλευσή τους και τη μορφή του φυσικού φαινομένου. Τα βασικά χαρακτηριστικά των καταστροφών παραμένουν ίδια για όλες τις κατηγορίες. Εξασφαλίζεται έτσι η μορφολογική συνένωσή τους και η περαιτέρω συγκεντρωτική μελέτη τους.

Επίσης αναδεικνύεται η τεράστια σημασία της εναρμόνισης των δεδομένων. Πραγματοποιείται μία πλήρης αποδόμηση των δεδομένων που παρέχουν οι υπάρχοντες βάσεις δεδομένων και αναδόμηση τους πάλι στο μοντέλο το σχεσιακό που επιλέξαμε να υλοποιήσουμε. Διασφαλίζεται πως κάθε εγγραφή στη βάση εξασφαλίζει συγκεκριμένα κριτήρια στην ποιότητα των δεδομένων. Απαλείφεται η πλεονάζουσα πληροφορία, καθαρίζονται τα δεδομένα, αφαιρούνται λάθη και μη συνεκτικότητες στα δεδομένα φυσικών καταστροφών.

Επίσης η μελέτη των υπάρχοντων βάσεων, ανέδειξε της ελλείψεις των υπαρχουσών πηγών. Οι υπάρχουσες βάσεις δεδομένων αποθηκεύουν τα δεδομένα χωρίς εκτενή ανάλυση και επεξεργασία τους, αφήνοντάς το στο πεδίο του εκάστοτε χρήστη. Συνέπεια αυτού είναι η εξαγωγή μη συνεπών και συνεκτικών συμπερασμάτων με βάση την επεξεργασία που επέλεξε να εκτελέσει ο ερευνητής. Επίσης προσθέτει επιπλέον επίπεδο πολυπλοκότητας για στην εξαγωγή συμπερασμάτων, καθώς προστίθεται ένα επιπλέον επίπεδο επεξεργασίας. Επίσης παρατηρήθηκε καθυστέρηση στην καταγραφή των συνεπειών των φυσικών καταστροφών με αποτέλεσμα πια πολλές εγγραφές να έχουν κενές τιμές σε αυτά τα πεδία.

Τέλος, έχουν αναπτυχθεί RESTful APIs για την άντληση των παραπάνω δεδομένων, οι οποίες προσφέρονται είτε άντληση όλων των δεδομένων, είτε με βάση συγκεκριμένα κριτήρια. Τα κριτήρια που αναπτύχθηκαν είναι το είδος της καταστροφής και η ημερομηνία του συμβάντος.

7.2 ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΟΠΤΙΚΕΣ

7.2.1 Επικράτηση ενιαίου προτύπου στις φυσικές καταστροφές

Η παρούσα διπλωματική εργασία, μπορεί να αποτελέσει σημαντικό προπομπό για μελλοντικές ερευνητικές δραστηριότητες. Η μελέτη δεδομένων διαφορετικών καταστροφών και με διαφορετικό τρόπο συλλογής, δίνει την δυνατότητα για ανάπτυξη ενός ενιαίου προτύπου συλλογής και αποθήκευσης δεδομένων φυσικών καταστροφών. Ανεξάρτητα από το είδος της φυσικής καταστροφής συμπεραίνουμε σημαντικά κοινά στοιχεία στα δεδομένα και την εν δυνάμει ενοποιημένη μελέτη τους. Αυτό θα μπορούσα μελλοντικά να οδηγήσει στην κατασκευή ενός προτύπου για όλα τα δεδομένα των φυσικών καταστροφών.

7.2.2 Επέκταση

Είναι δυνατή η επέκταση του μοντέλου για συλλογή περισσότερων δεδομένων από μεγαλύτερη πληθώρα βάσεων. Η επεκτασιμότητα αυτή δίνει στην βάση που κατασκευάσαμε νέες προοπτικές για αποθήκευση δεδομένων, προφέροντας την δυνατότητα συγκέντρωσης όλων των δεδομένων φυσικών καταστροφών σε μια πηγή. Μεγαλύτερους όγκους δεδομένων μπορεί να προσφέρει μεγαλύτερες δυνατότητες εποπτείας και ανάλυσής τους. Είναι σημαντικό να επισημανθεί, πως δεν χρειάζονται να γίνουν αλλαγές στο μοντέλο και μπορεί απευθείας να χρησιμοποιηθεί για νέες βάσεις.

7.2.3 Επιπλέον διεπαφές

Οι διεπαφές RESTful APIs που έχουν δημιουργηθεί μπορούν να επεκταθούν για υποστήριξη περισσότερων επιλογών. Θα μπορούσαν να αναπτυχθούν διεπαφές για εισαγωγή, ή ακόμη και διεργαφία των δεδομένων. οι διεπαφές αυτές είναι απαραίτητες για τη σωστή επικοινωνία του μοντέλου με τον υπόλοιπο κόσμο. Πέρα από άντληση δεδομένων είναι πάρα πολύ σημαντικό να υπάρχει δυνατότητα εισαγωγής τους καθώς και επεξεργασίας τους. Επιπλέον διεπαφές θα δώσουν στο χρήστη και δυνατότητα να αλληλοεπιδρά αποκλειστικά με αυτές με τη βάση, θα μπορεί να εισάγει, να διαγραφεί κι ακόμα και να τροποποιεί δεδομένα, τηρώντας όλους τους περιορισμούς του μοντέλου.

7.2.4 Ποιοτική αναβάθμιση των δεδομένων

Το κατασκευασμένο μοντέλο προσφέρει μια αξιολογη εοπτεία των δεδομένων φυσικών καταστροφών που προσφέρονται από διαφορετικές πηγές. Παρ όλα αυτά υπάρχουν εν δυνάμει μεγάλες δυνατότητες βελτιώσεις της ποιότητας των δεδομένων. Μια σημαντική μελλοντική προοπτική, ειδικά αν συγκεντρώσουμε δεδομένα από μεγάλο αριθμό πηγών είναι η δυνατό-

τητα αφαίρεσης των διπλότυπων εγγράφων. Καθιστώντας έτσι, μια μοναδικότητα σε κάθε εγγραφή στη βάση. Επιπλέον θα μπορούσε να πραγματοποιηθεί πληρότερη εναρμόνιση στα δεδομένα, ώστε αυτά να περιέχουν μεγαλύτερη πληροφορία με στοχευμένο τρόπο. Για παράδειγμα για τον προσδιορισμό του τόπου της καταστροφής, θα μπορούσαν να χρησιμοποιούνται αποκλειστικά συντεταγμένες για τον τοπικό προσδιορισμό των εγγράφων. Ακόμη και αν αυτές δεν υπάρχουν στο σύστημα θα μπορούσε να γίνει εξαγωγή τους από τα υπόλοιπα δεδομένα, όπως η πόλη, ο δρόμος, ο ταχυδρομικός κώδικας.

7.2.5 Πρόβλεψη μελλοντικών φυσικών καταστροφών

Με βάση το μοντέλο που σχεδιάστηκε αναπτύχθηκε στην παρούσα διπλωματική, είναι δυνατόν να αναπτυχθεί και να εκτελεστεί ένα μοντέλο μηχανικής μάθησης ή ένα νευρωνικό δίκτυο που με είσοδο τα δεδομένα της βάσης που παράξαμε, τα οποία αφού επεξεργαστούν, να πραγματοποιεί προβλέψεις για μελλοντικές καταστροφές. Η σημασία μιας τέτοιας εφαρμογής θα είχε τεράστιο αντίκτυπο καθώς η πρόβλεψη μελλοντικών φυσικών καταστροφών καθώς δυναμικά θα μπορούσε να εξοικονομήσει τεράστιο ανθρώπινο και υλικό κόστος.

8 ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] «EMDAT,» [Ηλεκτρονικό]. Available: <https://public.emdat.be>. [Πρόσβαση 2022].
- [2] G. & H. S. N. Shen, «Spatial–Temporal snapshots of global natural disaster impacts Revealed from EM-DAT for 1900-2015.,» *Geomatics, Natural Hazards and Risk*, 2019.
- [3] «The Significant Earthquake Database,» [Ηλεκτρονικό]. Available: <https://www.ncei.noaa.gov/access/metadata/landing-page/bin/iso?id=gov.noaa.ngdc.mgg.hazards:G012153>. [Πρόσβαση 2022].
- [4] «The NCEI/WDS Tsunami Runup Database,» [Ηλεκτρονικό]. Available: <https://www.ngdc.noaa.gov/hazel/view/hazards/tsunami/runup-search>. [Πρόσβαση 2022].
- [5] J. Melton, «Database language sql.,» *Handbook on Architectures of Information Systems*, 1998.
- [6] A. H. F. K. a. S. S. Silberschatz, «Database system concepts.,» *New York: McGraw-Hill*, 2002.
- [7] «Primary key - Wikipedia,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Primary_key. [Πρόσβαση 2022].
- [8] V. Vaswani, «MySQL database usage & administration,» *McGraw Hill Professional*, 2009.
- [9] «dbdiagram.io,» [Ηλεκτρονικό]. Available: dbdiagram.io/. [Πρόσβαση 2022].
- [10] «mysql.com/connector-python,» [Ηλεκτρονικό]. Available: <https://dev.mysql.com/doc/connector-python/en/> . [Πρόσβαση 2022].
- [11] «educba,» [Ηλεκτρονικό]. Available: <https://www.educba.com/mysql-query-commands/>. [Πρόσβαση 2022].
- [12] A. F. a. H. H. H. Dutka, «Fundamentals of data normalization,» *Addison-Wesley Longman Publishing Co., Inc.*, 1991.
- [13] C. P. F. J. G. F. P. M. & B. M. T. Pardo, «A process for driving the harmonization of models.,» *Proceedings of the 11th International Conference on Product Focused Software.*, 6 2010.
- [14] M. & W. C. Wolf, «Date and time formats.,» *W3C NOTE-datetime-19980827*, August 1998.

- [15] «Visual Studio Code - Wikipedia,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Visual_Studio_Code. [Πρόσβαση 2022].
- [16] «Documentation for Visual Studio Code,» [Ηλεκτρονικό]. Available: <https://code.visualstudio.com/docs/?dv=win>. [Πρόσβαση 2022].
- [17] «Python,» [Ηλεκτρονικό]. Available: <https://www.python.org/downloads/>. [Πρόσβαση 2022].
- [18] «pandas: a Foundational Python Library for Data Analysis and Statistics».
- [19] «pandas - Python Data Analysis Library,» [Ηλεκτρονικό]. Available: <https://pandas.pydata.org>. [Πρόσβαση 2022].
- [20] «API Architecture: The HTTP Protocol and Its Importance | by Dan Suciu | API World,» [Ηλεκτρονικό]. Available: <https://medium.com/api-world/api-architecture-the-http-protocol-and-its-importance-aeba0fe46f91>. [Πρόσβαση 2022].
- [21] R. e. a. FIELDING, «RFC2616: Hypertext Transfer Protocol--HTTP/1.1,» 1999.
- [22] R. e. a. Fielding, «RFC2068: Hypertext Transfer Protocol--HTTP/1.1.,» 1997.
- [23] A. Rodriguez, «Restful web services: The basics.,» *IBM developerWorks* 33, 2008.
- [24] P. & O. A. Bansal, «Study on Integration of FastAPI and Machine Learning for Continuous Authentication of Behavioral Biometrics.,» *2022 International Symposium on Networks, Computers and Communications (ISNCC)*, 2022.
- [25] D. PETKOVIĆ, «JSON integration in relational database systems,» *Int J Comput Appl*, 2017.