



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

Πρόγραμμα Σπουδών

Τμήματος Μηχανικών Πληροφορικής Τ.Ε. Λάρισας

Διεπαφές Χάρτη στις

Web Εφαρμογές

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ευάγγελος Παραλής (ΑΜ: 4415173)

Επιβλέπων: Φ. Κόκκορας, Καθηγητής

ΛΑΡΙΣΑ 2022



«Εγώ ο Ευάγγελος Παραλής, δηλώνω υπεύθυνα ότι η παρούσα Πτυχιακή Εργασία με τίτλο «Διεπαφές χάρτη στις Web εφαρμογές» είναι δική μου και βεβαιώνω ότι:

- Σε όσες περιπτώσεις έχω συμβουλευτεί δημοσιευμένη εργασία τρίτων, αντό επισημαίνεται με σχετική αναφορά στα επίμαχα σημεία.
- Σε όσες περιπτώσεις μεταφέρω λόγια τρίτων, αντό επισημαίνεται με σχετική αναφορά στα επίμαχα σημεία. Με εξαίρεση τέτοιες περιπτώσεις, το υπόλοιπο κείμενο της πτυχιακής αποτελεί δική μου δουλειά.
- Αναφέρω ρητά όλες τις πηγές βοήθειας που χρησιμοποίησα.
- Σε περιπτώσεις που τμήματα της παρούσας πτυχιακής έγιναν από κοινού με τρίτους, αναφέρω ρητά ποια είναι η δική μου συνεισφορά και ποια των τρίτων.
- Γνωρίζω πως η λογοκλοπή αποτελεί σοβαρότατο παράπτωμα και είμαι ενήμερος(-η) για την επέλευση των νόμιμων συνεπειών»

  
Τιμωντις Συντόκος

Ο φοιτητής εντάχθηκε αυτοδίκαια στο Πανεπιστήμιο Θεσσαλίας, σύμφωνα με την παρ. 1 του άρθρου 6 του Ν.4589/2019 (ΦΕΚ 13/Α'29.01.2019). Η εκπαιδευτική λειτουργία του ανωτέρου προγράμματος σπουδών συνεχίζεται μεταβατικά σύμφωνα με την παρ. 2 του άρθρου 6 του Ν.4589/2019 (ΦΕΚ 13/Α'29.01.2019).

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

Τόπος: .....

Ημερομηνία: .....

### ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1. .....
2. .....
3. .....

# Περίληψη

Η παρούσα πτυχιακή εργασία αποτελεί μια λεπτομερή και εκτενή παρουσίαση των δημοφιλέστερων επιλογών που έχει στη διάθεση του ένας προγραμματιστής, για την εισαγωγή διαδραστικών χαρτών στην εφαρμογή του. Επίκεντρο αποτελούν οι επιλογές που προσφέρει η Google και το Open Street Map. Οι δύο αυτές πλατφόρμες, επιλέχθηκαν βάση συγκεκριμένων κριτηρίων, μεταξύ των οποίων είναι η ποικιλία των χαρακτηριστικών και των επιλογών που προσφέρουν στον προγραμματιστή, αλλά και ο υψηλός βαθμός της μεταξύ τους συγκρισμότητας.

Οι πλατφόρμες αυτές συγκρίνονται τόσο σε επίπεδο διεπαφής χρήστη, όσο και σε επίπεδο κώδικα. Για την αποτελεσματικότερη προβολή και αντιπαραβολή τους, έχει δημιουργηθεί μια demo εφαρμογή επίδειξης των κυριότερων κοινών τους χαρακτηριστικών, με παραδείγματα και επισήμανση του σχετικού κώδικα.

Σκοπός είναι η διευκόλυνση επιλογής μιας εκ των δύο πλατφορμών, η εξοικείωση του αναγνώστη με τις βιβλιοθήκες που παρέχουν οι πλατφόρμες, αλλά και η απόδειξη του πόσο εύκολη είναι η ενσωμάτωση χαρτών σε μια web εφαρμογή και η ανάδειξη της σημασίας τους για την εμπειρία του εκάστοτε χρήστη.

Υπάρχει πάντα χώρος για βελτιώσεις και προσθήκες, καθώς η εφαρμογή είναι κατασκευασμένη για να είναι εύκολα επεκτάσιμη με νέα χαρακτηριστικά, ενώ μπορεί να αναδεικνύει και περιπτώσεις υπεροχής, είτε λόγω έλλειψης λειτουργειών από τη μια πλευρά ή λόγω καλύτερης υλοποίησης ενός χαρακτηριστικού.



# **Ευχαριστίες**

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω τους γονείς μου, τους γονείς τους και τη θεία μου για όλη τη στήριξη που μου παρείχαν κατά την διάρκεια των σπουδών μου.

Παραλήγει Ευάγγελος

21/04/2022



# Περιεχόμενα

<b>ΠΕΡΙΛΗΨΗ .....</b>	<b>I</b>
<b>ΕΥΧΑΡΙΣΤΙΕΣ .....</b>	<b>III</b>
<b>ΠΕΡΙΕΧΟΜΕΝΑ .....</b>	<b>V</b>
<b>1 ΕΙΣΑΓΩΓΗ.....</b>	<b>1</b>
1.1 ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ .....	1
1.2 ΨΗΦΙΑΚΟΙ ΧΑΡΤΕΣ .....	2
1.3 ΑΠΕΙΚΟΝΙΣΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΧΑΡΤΩΝ ΜΕΣΩ ΛΟΓΙΣΜΙΚΟΥ.....	3
1.4 ΣΤΟΧΟΙ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ .....	5
1.5 ΔΙΑΡΘΡΩΣΗ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ .....	6
<b>2 ΓΕΩΓΡΑΦΙΚΟΙ ΧΑΡΤΕΣ ΣΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ.....</b>	<b>9</b>
2.1 WAZE.....	9
2.2 APPLE MAPS.....	11
2.3 MICROSOFT BING MAPS .....	13
2.4 MAPBox.....	14
2.5 GOOGLE MAPS .....	16
2.6 OPENSTREETMAP .....	18
<b>3 ΑΝΤΙΚΕΙΜΕΝΟ ΠΤΥΧΙΑΚΗΣ .....</b>	<b>21</b>
3.1 Επιλογή ΧΑΡΤΩΝ.....	21
3.2 ΑΝΤΙΠΑΡΑΒΟΛΗ ΧΑΡΤΩΝ GOOGLE MAPS KAI OPEN STREET MAPS .....	23
<b>4 ΑΝΤΙΠΑΡΑΒΟΛΗ GOOGLE MAPS KAI OPEN STREET MAPS.....</b>	<b>25</b>
4.1 ΑΡΧΙΚΟΠΟΙΗΣΗ ΧΑΡΤΗ .....	25
4.1.1 Google Maps.....	25
4.1.2 OpenStreetMap.....	26
4.2 ΣΤΟΙΧΕΙΑ ΕΛΕΓΧΟΥ ΧΑΡΤΗ.....	27
4.2.1 Google Maps.....	27
4.2.2 OpenStreetMap.....	29

4.3	MARKERS.....	30
4.3.1	<i>Google Maps</i> .....	30
4.3.2	<i>OpenStreetMap</i> .....	31
4.4	INFOWINDOW/POPUP .....	33
4.4.1	<i>Google Maps - Infowindow</i> .....	33
4.4.2	<i>OpenStreetMap - Popup</i> .....	34
4.5	ΣΧΗΜΑΤΑ .....	35
4.5.1	<i>Polylines – Google Maps</i> .....	36
4.5.2	<i>Polylines – OpenStreetMap</i> .....	36
4.5.3	<i>Polygons – Google Maps</i> .....	38
4.5.4	<i>Polygons – OpenStreetMap</i> .....	39
4.5.5	<i>Rectangles – Google Maps</i> .....	40
4.5.6	<i>Rectangles – OpenStreetMap</i> .....	41
4.5.7	<i>Circles – Google Maps</i> .....	42
4.5.8	<i>Circles - OpenStreetMap</i> .....	43
4.6	OVERLAYS/ LAYERS και LAYERS/ LAYERGROUPS.....	44
4.6.1	<i>Google Maps Overlays και Layers</i> .....	44
4.6.2	<i>OpenStreetMap Layers και LayerGroups</i> .....	46
<b>5</b>	<b>ΕΦΑΡΜΟΓΗ ΕΠΙΔΕΙΞΗΣ .....</b>	<b>47</b>
5.1	ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΕΦΑΡΜΟΓΗΣ.....	47
5.2	ΑΡΧΙΚΟΠΟΙΗΣΗ ΧΑΡΤΗ .....	49
5.2.1	<i>Αρχικοποίηση Χάρτη Google Maps</i> .....	50
5.2.2	<i>Αρχικοποίηση Χάρτη Open Street Maps</i> .....	50
5.3	CONTROLS ΧΑΡΤΗ .....	51
5.3.1	<i>Controls Google Maps Χάρτη</i> .....	51
5.3.2	<i>Controls Open Street Maps Χάρτη</i> .....	52
5.4	MARKERS.....	53
5.4.1	<i>Markers Google Maps Χάρτη</i> .....	54
5.4.2	<i>Markers Open Street Maps Χάρτη</i> .....	55
5.5	TOOLTIPS.....	56
5.5.1	<i>Tooltips Google Maps Χάρτη</i> .....	56
5.5.2	<i>Tooltips Open Street Maps Χάρτη</i> .....	57
5.6	INFO WINDOWS.....	57

5.6.1	<i>Info Windows Google Maps Χάρτη</i> .....	57
5.6.2	<i>Info Windows Open Street Maps Χάρτη</i> .....	59
5.7	POLYLINES.....	60
5.7.1	<i>Polylines Google Maps Χάρτη</i> .....	60
5.7.2	<i>Polylines Open Street Maps Χάρτη</i> .....	61
5.8	CIRCLES .....	62
5.8.1	<i>Circles Google Maps Χάρτη</i> .....	62
5.8.2	<i>Circles Open Street Maps Χάρτη</i> .....	64
5.9	RECTANGLES .....	65
5.9.1	<i>Rectangles Google Maps Χάρτη</i> .....	65
5.9.2	<i>Rectangles Open Street Maps Χάρτη</i> .....	66
5.10	POLYGONS.....	67
5.10.1	<i>Polygons Google Maps Χάρτη</i> .....	67
5.10.2	<i>Polygons Open Street Maps Χάρτη</i> .....	68
5.11	GROUND / IMAGE OVERLAYS .....	69
5.11.1	<i>Ground Overlays Google Maps Χάρτη</i> .....	70
5.11.2	<i>Ground Overlays Open Street Maps Χάρτη</i> .....	70
5.12	ΔΕΜΟ ΕΦΑΡΜΟΓΗΣ ΕΠΙΔΕΙΞΗΣ .....	71
<b>6</b>	<b>ΣΥΜΠΕΡΑΣΜΑΤΑ</b> .....	<b>75</b>
	<b>ΒΙΒΛΙΟΓΡΑΦΙΑ</b> .....	<b>77</b>



# 1 Εισαγωγή

Η χαρτογραφία, οι χάρτες, αλλά κυρίως η συλλογή γεωγραφικών δεδομένων, αποτελούν έννοιες που υπάρχουν εδώ και αιώνες. Οι χάρτες, ανέκαθεν αποτελούσαν, εργαλεία που βοηθούσαν τον άνθρωπο να κατανοήσει τον κόσμο και διευκολύναν την περιήγησή του σε αυτόν. Αυτοί, ακόμα και σήμερα αποτελούν ένα αναπόσπαστο κομμάτι της καθημερινότητας του και του επιτρέπουν να πετύχει τη σημερινή παγκόσμια διασύνδεση.

## 1.1 Ιστορική Αναδρομή

Οι πρώτες καταγραφές εμφάνισης χαρτών, έχουν σημειωθεί στην Αίγυπτο και στη Βαβυλωνία, με επιγραφές σφηνοειδούς γραφής. Και στις δύο περιπτώσεις, ο στόχος ήταν η καταγραφή του τότε γνωστού κόσμου. Η πρώτη επίσημη καταγραφή χαρτών, όμως, πραγματοποιήθηκε από τους αρχαίους Έλληνες και επισημοποιήθηκε από τους Ρωμαίους, οι οποίοι εισήγαγαν την έννοια της καταγραφής των ιδιοκτησιών τους (Ασβεστάς, 2008).

Στην Ευρώπη, η χαρτογραφία αναγεννήθηκε με τη πτώση της Βυζαντινής αυτοκρατορίας. Με τη μετάφραση του έργου *Geographia* του Κλαούντιου Πτολεμαίου, οι άνθρωποι του 15<sup>ου</sup> αιώνα, είχαν πλέον εικόνα του τότε γνωστού κόσμου. Μέχρι τον 19<sup>ο</sup> αιώνα, οι χάρτες χρησιμοποιούνταν κυρίως στο εμπόριο και στις εξερευνήσεις (Σουλακέλλης & Ρούσσου, 2019).

Με την εμφάνιση των αεροφωτογραφιών, η χαρτογράφηση της γης επιτεύχθηκε με ραγδαίους ρυθμούς και ολοκληρώθηκε σχεδόν πλήρως μετά τον 2<sup>ο</sup> Παγκόσμιο Πόλεμο τους (Ασβεστάς, 2008). Μέχρι σήμερα οι χάρτες έχουν βοηθήσει στην εξέλιξη των κοινωνιών, στην ανάπτυξη της πολυπλοκότητάς τους, καθώς και των υποδομών που τις προδιαγράφουν. Τέτοιες υποδομές είναι, για παράδειγμα, οι τηλεφωνικές γραμμές, το διαδίκτυο και οι σιδηρόδρομοι.

Η τεχνολογία της χαρτογράφησης, έχει εξελιχθεί τόσο στις μέρες, όπου κάθε άτομο, μπορεί, ανά πάσα στιγμή, να δει σε ποιο σημείο της γης βρίσκεται, να δει τον χάρτη ολόκληρης της γης και όχι μόνο, αφού γνωστές εφαρμογές προσφέρουν εικόνα πληθώρας

πλανητών του ηλιακού μας συστήματος. Τη λύση σε όλα τα παραπάνω, έχουν δώσει οι ψηφιακοί χάρτες, η οποίοι θα αναλυθούν στα επόμενα κεφάλαια.

## 1.2 Ψηφιακοί Χάρτες

Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, μέχρι πρόσφατα, ο άνθρωπος χρησιμοποιούσε τους παραδοσιακούς χάρτες. Οι παραδοσιακοί χάρτες αναπαριστούν τον πραγματικό κόσμο υπό κλίμακα. Η κλίμακα ορίζεται ως ο λόγος της απόστασης στο χαρτί προς την απόσταση στην επιφάνεια της Γης. Ένας παραδοσιακός χάρτης, όπως απεικονίζεται στην Εικόνα 1, αναπαριστά πάντα ενός είδους πληροφορία και σε μια μόνο κλίμακα. Με την ανάπτυξη όμως της τεχνολογίας, εμφανίστηκαν οι ψηφιακοί χάρτες, οι οποίοι προσδίδουν πολλά παραπάνω χαρακτηριστικά και πληροφορίες στον εκάστοτε χρήστη.

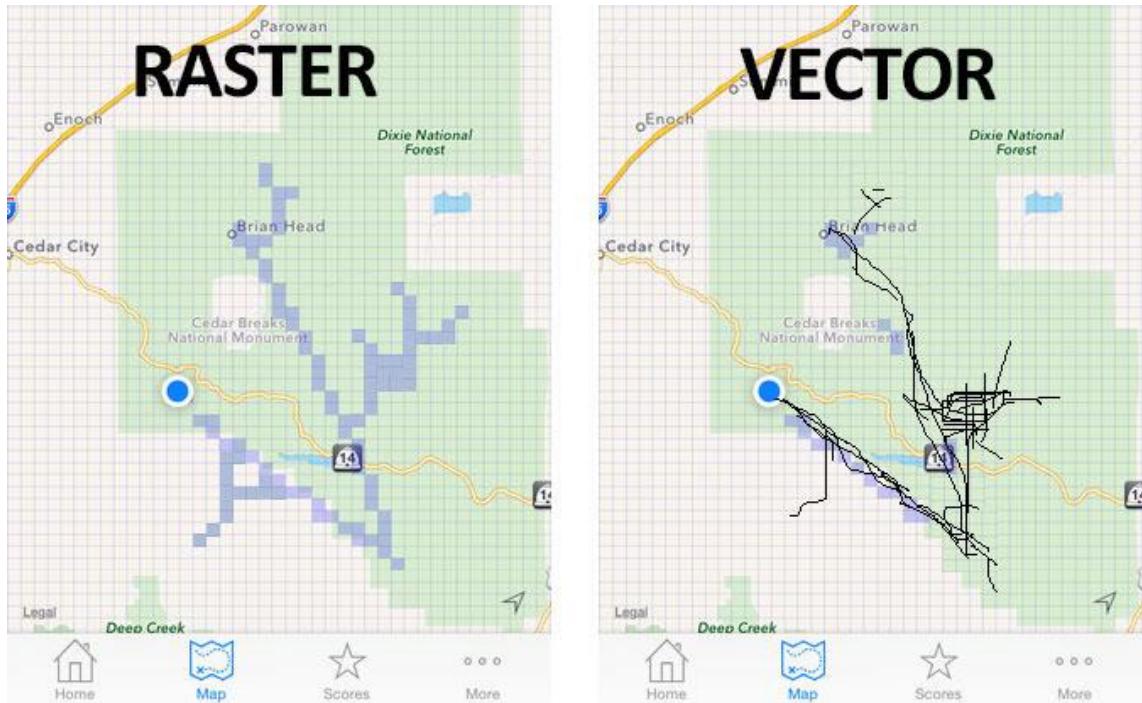


Εικόνα 1: Πολιτικός σχολικός χάρτης

Η αναπαράσταση τέτοιων χαρτών, μπορεί να γίνει με *Ψηφιδωτή* και *Διανυσματική* μορφή. Η Ψηφιδωτή μορφή (Raster) προκύπτει από την σάρωση του φυσικού χάρτη, κατά την οποία αυτός διαιρείται και παρουσιάζεται σε κελιά. Σε κάθε ένα από αυτά τα κελιά αναθέτεται ένα χαρακτηριστικό ή ιδιότητα. Αντίθετα, στη διανυσματική μορφή (Vector) μετά τη σάρωση, κάθε γεωγραφικό χαρακτηριστικό που εμφανίζεται στον χάρτη αναπαρίσταται στον υπολογιστή σαν σημείο. Τα σημεία που αποτυπώνονται στο χάρτη συνδέονται με ευθείες γραμμές η καμπύλες, ώστε να γίνουν κατανοητά τα εξαγόμενα

δεδομένα. Η πληροφορία των κελιών ή των γραμμών αυτών εντάσσεται στο σύστημα αναφοράς των υπολογίστων χαρτογραφικών δεδομένων που έχουν συλλεχθεί στο παρελθόν (Κυπριάδου, 2015).

Παρακάτω, στην Εικόνα 2, απεικονίζονται δύο παραδείγματα τέτοιων χαρτών, σε ψηφιακή και διανυσματική μορφή αντίστοιχα.



Εικόνα 2 : Raster vs Vector χάρτες

### 1.3 Απεικόνιση ηλεκτρονικών χαρτών μέσω λογισμικού

Οι ψηφιακοί χάρτες έχουν συμβάλει δραματικά στην καταγραφή και στη συλλογή πληροφοριών σε κεντρικές βάσεις δεδομένων. Με την βοήθεια αυτών των δεδομένων και του υλικού που έχει συγκεντρωθεί, η σημερινή τεχνολογία έχει καταφέρει να δημιουργήσει λογισμικά αναπαράστασης ηλεκτρονικών χαρτών, τα οποία ανά πάσα στιγμή μπορούν να απεικονίσουν έναν χάρτη, σε οποιαδήποτε μορφή και κλίμακα θελήσει κάποιος.

Η δημιουργία ενός τέτοιου λογισμικού, είναι προϊόν μεγάλων σύγχρονων εταιριών. Μπορεί να αποδοθεί σε Web, Mobile και Desktop εφαρμογές ανάλογα με το πόσο ευρεία είναι η παρούσα ανάπτυξή του. Ένα τέτοιο παράδειγμα λογισμικού, αποτελεί το πλέον ευρέως διαδεδομένο σε όλο το κόσμο, *Google Maps*. Το Google Maps, όχι μόνο απεικονίζει χάρτες σε διάφορες μορφές, αλλά προσφέρει ποικιλία διαφόρων χρήσιμων

λειτουργιών, όπως την επίδειξη της βέλτιστης διαδρομής από ένα σημείο A σε ένα σημείο B. Ένα παράδειγμα απεικονίζεται στην Εικόνα 3 παρακάτω:



Εικόνα 3 : Απεικόνιση βέλτιστης διαδρομής σε Google Maps

Όπως γίνεται κατανοητό, τα λογισμικά απεικόνισης χαρτών έχουν περισσότερα πλεονεκτήματα από τους απλούς, παραδοσιακούς, φυσικούς χάρτες. Μερικά από αυτά είναι τα εξής (Ασβεστάς, 2008):

- Το περιεχόμενο που απεικονίζει ο παραδοσιακός χάρτης είναι στατικό. Αντίθετα, ο ηλεκτρονικός χάρτης μπορεί να απεικονίζει περιεχόμενο δυναμικά καθώς και να εναλλάσσει την μορφή και τον τύπο του.
- Σε αντίθεση με του παραδοσιακούς χάρτες, οι ηλεκτρονικοί μπορούν να συνδυάσουν πληροφορίες από βάσεις δεδομένων και να αναπαραστήσουν δεδομένα, χρήσιμα για οποιουδήποτε τύπου μελέτη.
- Οι παραδοσιακοί χάρτες είναι πιθανό να μην μπορούν να αναπαραστήσουν ορισμένες πληροφορίες. Για παράδειγμα, η καμπυλότητα της γης είναι δύσκολο να αναπαρασταθεί στο χαρτί, σε αντίθεση με έναν ηλεκτρονικό χάρτη.
- Οι παραδοσιακοί χάρτες παρουσιάζουν μια επιφάνεια σε δύο διαστάσεις και από ένα συγκεκριμένο ύψος που καθορίζει την κλίμακα του χάρτη. Οι χάρτες λογισμικού μπορούν να απεικονίσουν τον χάρτη σε τρισδιάστατη μορφή και από μεταβλητό ύψος. Τα τελευταία χρόνια, ιδιαίτερα το λογισμικό των Google Maps,

έχει εισάγει τη δυνατότητα *Street View*, η οποία απεικονίζει ένα σημείο, όπως μοιάζει στη πραγματικότητα, σε επίπεδο δρόμου.

- Οι δημιουργοί των ηλεκτρονικών χαρτών, κάνουν εφικτή την αλληλεπίδραση με τα προϊόντα τους, σε επίπεδο κώδικα, μέσω APIs ή ακόμα σημαίνοντας τα Project τους ως Open Source. Αυτό δίνει τη δυνατότητα στους προγραμματιστές να χρησιμοποιήσουν τις εν λόγω λειτουργίες σε δικές τους εφαρμογές, εμπλουτίζοντας τες με την απεικόνιση χαρτών ή ακόμα και να επεκτείνουν το εκάστοτε Project.

Το τελευταίο πλεονέκτημα της παραπάνω λίστας, δηλαδή η παροχή δυνατοτήτων ηλεκτρονικών χαρτών σε εφαρμογές λογισμικού, αποτελεί το θέμα αυτής της πτυχιακής εργασίας.

## 1.4 Στόχοι της πτυχιακής εργασίας

Η χρήση των ηλεκτρονικών χαρτών σε σύγχρονες εφαρμογές λογισμικού, μπορεί να αυξήσει την απήχηση που θα έχουν αυτές στο ευρύ κοινό, καθώς θα την εμπλουτίζει με πληροφορίες και λειτουργίες, οι οποίες θα τις κάνουν πιο διαδραστικές και εύχρηστες.

Παίρνοντας ως παράδειγμα ένα ηλεκτρονικό κατάστημα ρούχων, που έχει ενσωματώσει έναν ηλεκτρονικό χάρτη, στη σελίδα «Επικοινωνία» του καταστήματός, τα πλεονεκτήματα που κερδίζει είναι τα εξής:

- Οι χρήστες του καταστήματος, μπορούν να βρουν πιο εύκολα το φυσικό κατάστημα. Πατώντας στο διαδραστικό χάρτη, μπορούν να ανακατευθυνθούν στη σελίδα των χαρτών και να παρατηρήσουν την φυσική τοποθεσία του καταστήματος.
- Οι ηλεκτρονικοί χάρτες μπορούν να συνδυάσουν πληροφορίες και να τις συσχετίσουν με τοποθεσίες, προσφέροντας, ενδεχομένως, περισσότερες πληροφορίες μιας υπηρεσίας από άλλες πηγές. Η τοποθεσία του καταστήματος, ίσως απεικόνιζε κάποιες επιπλέον πληροφορίες επικοινωνίας, γεγονός που θα βοηθούσε τους χρήστες να επικοινωνήσουν πιο γρήγορα και με περισσότερη ευκολία με το κατάστημα.
- Η γενική χρήση ενός ηλεκτρονικού χάρτη, έχει ως αποτέλεσμα τη πιθανή εμφάνιση του καταστήματος και να δώσει πληροφορίες γι' αυτό. Αυτό βοηθά το κατάστημα να ανακαλυφθεί από πιθανούς μελλοντικούς πελάτες.

- Τέλος οι πληροφορίες που μπορεί να δώσει ένας ηλεκτρονικός χάρτης, θα μπορούσαν να εμπλουτιστούν από παλιούς πελάτες του καταστήματος, οι οποίοι είτε θα μπορούσαν να αναρτήσουν κάποιες εικόνες ή να γράψουν κριτικές για την συνολική εμπειρία της μεταχείρισής τους.

Συνεπώς, ο εμπλουτισμός μια εφαρμογής λογισμικού με ηλεκτρονικούς χάρτες συνεισφέρει σε μεγάλο βαθμό.

Ο στόχος αυτής πτυχιακής εργασίας αφορά την παρουσίαση των δυο δημοφιλέστερων λογισμικών ηλεκτρονικών χαρτών, τα οποία μέσω της έκθεσης των API τους, δίνουν τη δυνατότητα σε προγραμματιστές να εμπλουτίσουν τις εφαρμογές τους με τους χάρτες τους. Τα δύο δημοφιλέστερα API που έχουν επιλεχθεί προς ανάλυση, θα παρουσιαστούν στο Κεφάλαιο 2.

Στο τέλος αυτής της πτυχιακής εργασίας, ο αναγνώστης θα πρέπει να έχει κατανοήσει τις βασικές έννοιες λειτουργίας αυτών των APIs και να μπορεί, μέσω των παραδειγμάτων κώδικα και της Demo εφαρμογής που θα παρουσιαστεί, να μπορεί να ενσωματώσει τις δυο δημοφιλέστερες πλατφόρμες χαρτών στην εφαρμογή του και να προσθέτει χαρακτηριστικά, που θα εμπλουτίζουν την εμπειρία των χρηστών.

## 1.5 Διάρθρωση της πτυχιακής εργασίας

Το 2<sup>ο</sup> κεφάλαιο της εργασίας θα εισάγει τον αναγνώστη στους γεωγραφικούς χάρτες στον παγκόσμιο ιστό. Θα γίνει αναφορά στις υπάρχουσες τεχνολογίες λογισμικού χαρτών, δηλαδή σε όλες τις επιλογές που παρέχονται σε έναν προγραμματιστή σήμερα. Τέλος, θα δοθεί έμφαση στις δύο δημοφιλέστερες πλατφόρμες που θα αναλυθούν σε αυτή την εργασία.

Στο 3<sup>ο</sup> κεφάλαιο θα παρουσιαστεί το αντικείμενο της πτυχιακής εκτενέστερα. Ουσιαστικά, θα γίνει η μελέτη των δύο λύσεων και η αντιπαραβολή τους. Θα γίνει αναφορά στον τύπο των λογισμικών που αποτελούν και σε ποιες περιπτώσεις θα μπορούσε να χρησιμοποιηθεί η κάθε λύση, εισάγοντας έτσι το αναγνώστη στο 4<sup>ο</sup> κεφάλαιο

Στο 4<sup>ο</sup> κεφάλαιο θα γίνει εκτενής μελέτη των βασικότερων λειτουργιών και χαρακτηριστικών των δύο λύσεων. Θα αναφερθεί ο τρόπος αρχικοποίησης αυτών σε μια εφαρμογή και έπειτα θα αναλυθούν ένα προς ένα οι κυριότερες, όμοιες βασικές λειτουργίες που δίνουν οι δύο λύσεις. Τέλος θα παρουσιαστούν οι κυριότερες διαφορές

αυτών, γεγονός που θα βοηθήσει τον αναγνώστη να τις διακρίνει και, εν τέλη, να επιλέξει αυτή που του αρμόζει.

Το 5<sup>ο</sup> κεφάλαιο θα παρουσιάσει μια demo εφαρμογή web, υλοποιημένη σε JAVASCRIPT, η οποία έχει συγκεντρώσει όλα τα όμοια χαρακτηριστικά που έχουν αναφερθεί στο 4<sup>ο</sup> κεφάλαιο και τα απεικονίζει το με τους δυο χάρτες τον έναν δίπλα στον άλλο. Ο αναγνώστης, έτσι, θα μπορεί να συγκρίνει τις χαρακτηριστικά των δύο λύσεων, απευθείας και ένα προς ένα. Επίσης θα μπορεί να αντιγράψει και να εφαρμόσει τον κώδικα που εμφανίζεται σε κάθε παράδειγμα, για κάθε περίπτωση, και να τον ενσωματώσει στη δική του εφαρμογή. Αυτό καθιστά, τη demo εφαρμογή, ως μια μορφή οδηγού για νέους προγραμματιστές.

Τέλος, στο 6<sup>ο</sup> κεφάλαιο αναγράφονται όλα τα συμπεράσματα που συγκεντρώθηκαν κατά τη συγγραφή αυτής της πτυχιακής εργασίας, καθώς και τρόποι ανάπτυξης και εξέλιξης της demo εφαρμογής που υλοποιήθηκε στα πλαίσια της.



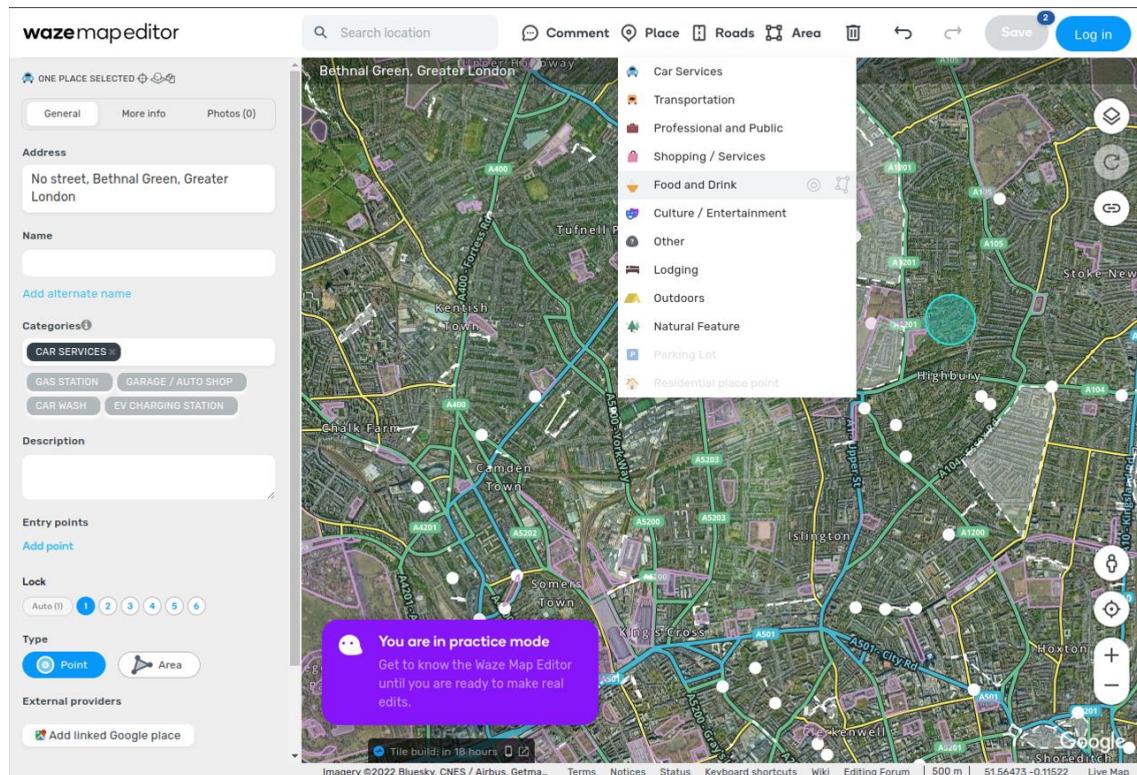
## **2 Γεωγραφικοί χάρτες στον παγκόσμιο ιστό**

Σε αυτό το κεφάλαιο θα γίνει αναφορά σε μερικές από τις μεγαλύτερες υπηρεσίες παροχής γεωγραφικών δεδομένων και χαρτών, με ιδιαίτερη έμφαση στο Google Maps και το OpenStreetMap, τα οποία αποτελούν και το βασικό αντικείμενο μελέτης της παρούσας εργασίας.

### **2.1 Waze**

Το Waze είναι, κατά βάση, μια εφαρμογή πλοήγησης, που προσφέρει ενημερώσεις για την κατάσταση των δρόμων σε πραγματικό χρόνο. Η λειτουργία της εξαρτάται από την κοινότητα των χρηστών της, οι οποίοι μπορούν να αναφέρουν και να μοιραστούν στοιχεία χρήσιμα για τους οδηγούς, όπως η κίνηση, ενδεχόμενα ατυχήματα, καιρικές συνθήκες. Επιπλέον, η εφαρμογή, από μόνη της, χωρίς δηλαδή να επέμβει ο χρήστης ή να χρησιμοποιήσει κάποια λειτουργία της, συλλέγει δεδομένα που την βοηθούν να υπολογίσει την μέση ταχύτητα κίνησης, να ελέγξει για τυχόν σφάλματα στους χάρτες, να βελτιώσει τη διάταξη των δρόμων καθώς και να μάθει τις κατευθύνεις των δρόμων και των στροφών (Waze Official Documentation, 2022). Επιπλέον, οι χρήστες έχουν τη δυνατότητα να επεξεργάζονται τους χάρτες, χειροκίνητα, μέσω του map editor, όπως

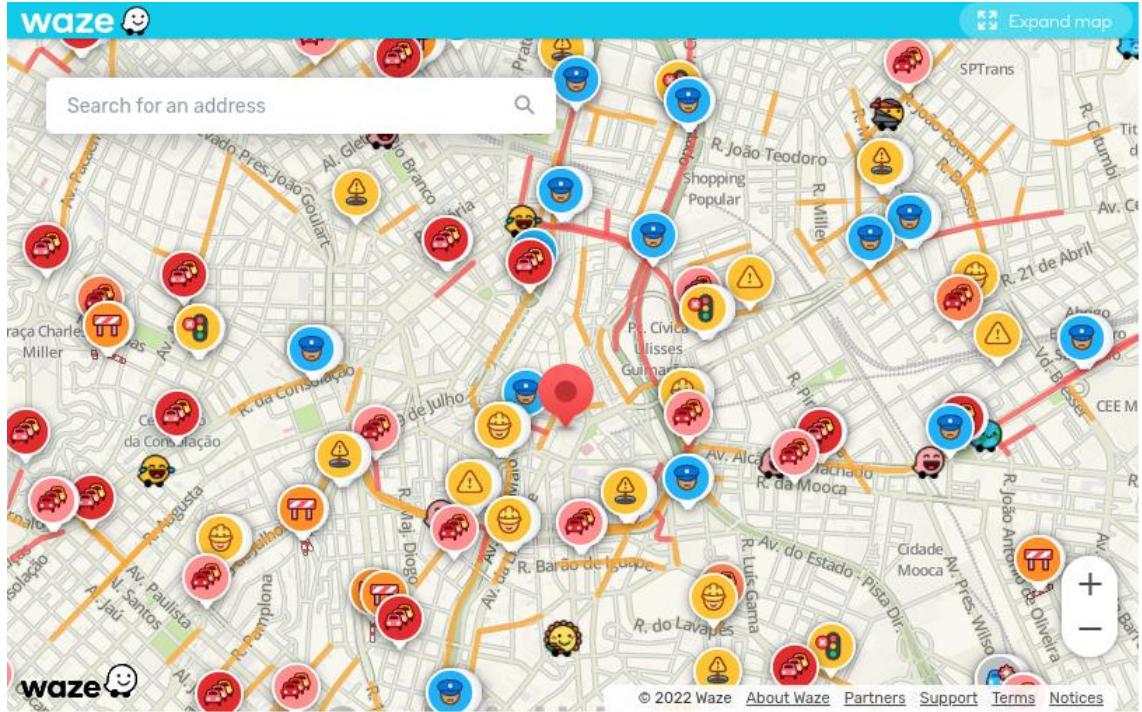
φαίνεται στην Εικόνα 4, μέσω του οποίου μπορούν να προσθέτουν μη καταγεγραμμένες τοποθεσίες καταστημάτων και υπηρεσιών, αλλά και νέους δρόμους.



Εικόνα 4 : Απεικόνιση του map editor

Οι υπηρεσίες του είναι διαθέσιμες τόσο ως web εφαρμογή όσο και σε κινητές συσκευές, android και ios. Διαθέτει SDKs για την ενσωμάτωση σε εφαρμογές και για τη χρήση των λειτουργιών πλοιήγησης. Επίσης, υποστηρίζει την ενσωμάτωση σε σελίδες ως iframe με hardcoded επιλογές για την εμφάνιση ενός marker στο κέντρο του χάρτη και την επιλογή εμφάνισης συμβόλων που περιγράφουν γεγονότα σχετικά με την κατάσταση

των δρόμων. Στην Εικόνα 5, φαίνεται, ενδεικτικά ένας embedded web map, με marker και σύμβολα.



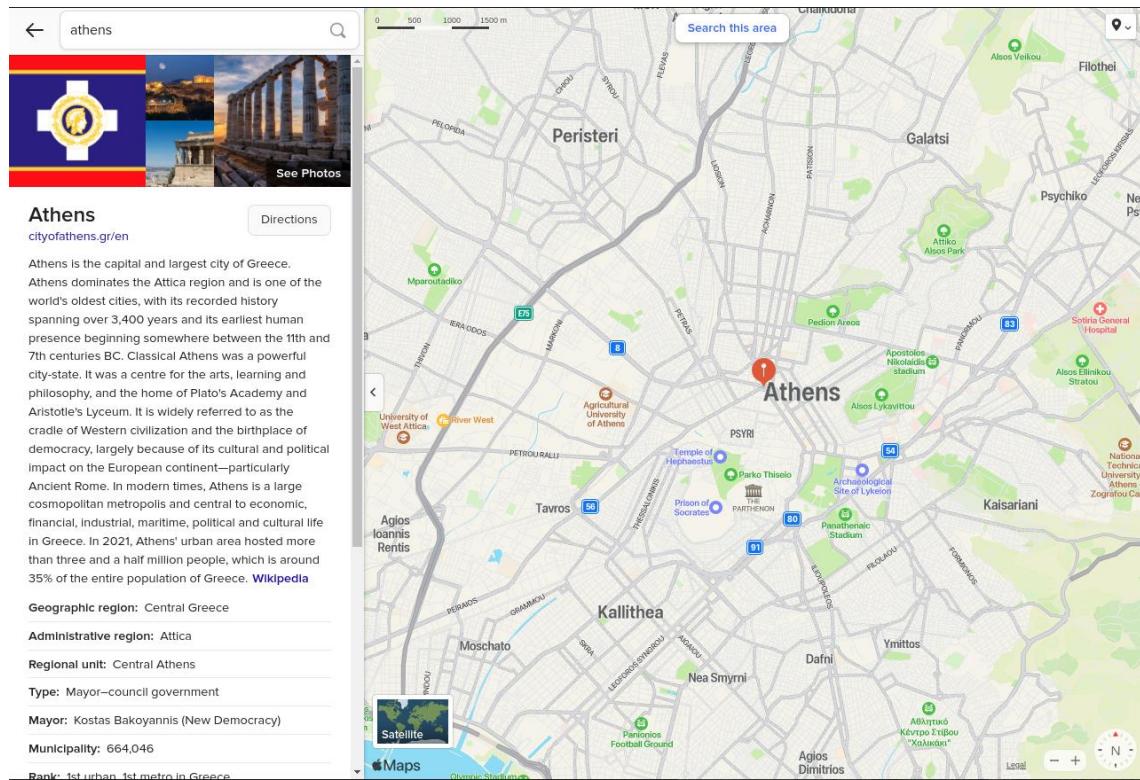
Εικόνα 5 : Παράδειγμα ενσωματωμένου χάρτη με marker και σύμβολα

Το Waze εξαγοράστηκε το 2013 από την Google (Ingrid Lunden, 2013), η οποία προσφέρει κατ' ανάγκη δεδομένα σε αυτό, όταν και αν κάποια περιοχή δεν έχει καταγραφεί από τους χρήστες. Τέλος, η χρήση της εφαρμογής, από οποιαδήποτε πλατφόρμα, είναι δωρεάν, ενώ δεν υπάρχει κάποια χρέωση για την ενσωμάτωση των χαρτών σε άλλες εφαρμογές.

## 2.2 Apple Maps

Το Apple Maps αποτελεί τη λύση της Apple για εφαρμογές χαρτών. Διατίθεται μέσω εφαρμογής σε όλες τις iOS συσκευές, ενώ για το web, αποκλειστικά μέσω της μηχανής αναζήτησης DuckDuckGo, όπου και είναι η προκαθορισμένη επιλογή για προβολή χαρτών. Μέσω των χαρτών, ο χρήστης μπορεί να αναζητήσει καταστήματα, υπηρεσίες, μνημεία και αξιοθέατα, να βρει τη βέλτιστη διαδρομή για τον προορισμό του καθώς και να πλοηγηθεί προς αυτόν, αλλά και να δει την επιλεγμένη τοποθεσία σε επίπεδο δρόμου. Επιπλέον λειτουργίες είναι η χρήση επαυξημένης πραγματικότητας για προσανατολισμό και καθοδήγηση όταν ο χρήστης είναι πεζός, τα Indoor Maps για την πλοήγηση σε

εσωτερικούς χώρους, όπου αυτή υποστηρίζεται. Η Εικόνα 6 δείχνει την πόλη της Αθήνας, όπως φαίνεται στην web έκδοση του Apple Maps.



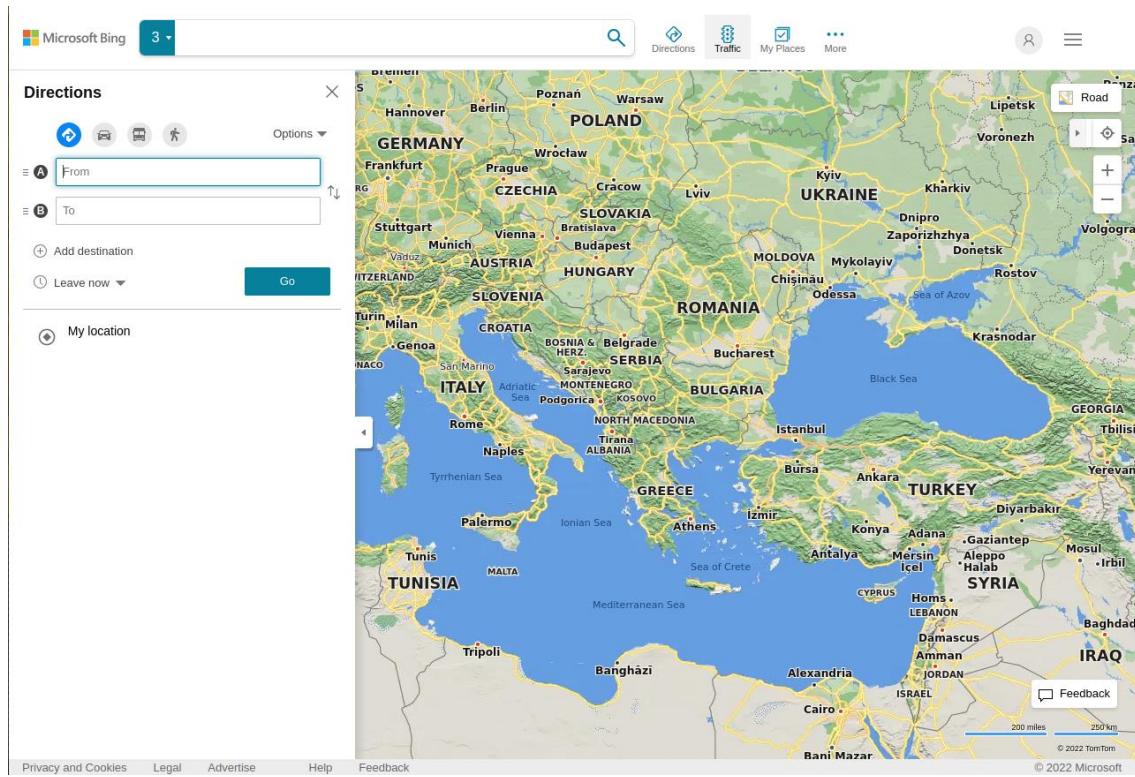
Εικόνα 6: Απεικόνιση Web έκδοσης του Apple Maps

Η Apple διαθέτει το MapKit, ένα SDK για την ενσωμάτωση των χαρτών σε iOS εφαρμογές, καθώς και το MapKitJs για την ενσωματώση τους σε σελίδες. Και τα δύο προσφέρουν χάρτες με δυνατότητα αλληλεπίδρασης από τον χρήστη, ενώ είναι δυνατή η τοποθέτηση overlay για προβολή markers, annotations και points of interest. Επίσης, είναι δυνατή και η τοποθέτηση στατικών χαρτών με τη χρήση των Snapshot υπηρεσιών.

Η χρήση των υπηρεσιών είναι δωρεάν για όλους τους χρήστες των υποστηριζόμενων πλατφόρμων, ωστόσο, η χρήση των APIs υπόκειται σε χρεώσεις. Πιο συγκεκριμένα, για την χρήση των υπηρεσιών για developers, ο προγραμματιστής πρέπει να διαθέτει συνδρομή στο Apple Developer Program, κόστους 99\$ ετησίως. Η συνδρομή αυτή επιτρέπει την απεριόριστη χρήση του MapKit SDK, ενώ για το MapKitJs έχει οριστεί το ημερήσιο όριο των 250,000 θεάσεων χαρτών και των 25,000 service calls στις υπηρεσίες του API.

## 2.3 Microsoft Bing Maps

To Bing Maps είναι η web εφαρμογή χαρτών της Microsoft. Προσφέρει όλες τις βασικές λειτουργίες ενός online χάρτη, όπως την αναζήτηση τοποθεσιών με θεματικά και ονομαστικά κριτήρια, την προβολή στοιχείων σχετικά με την κατάσταση των δρόμων και την πλοήγηση μεταξύ τοποθεσιών. Διατίθεται δωρεάν στο web ενώ δε διαθέτει κάποια αφιερωμένη εφαρμογή για κινητές συσκευές. Ωστόσο, αυτό δεν περιορίζει τις δυνατότητες των προγραμματιστών που θέλουν να εντάξουν τους χάρτες στις εφαρμογές τους. Η Microsoft εκθέτει τους χάρτες της και τις λειτουργίες τους μέσω APIs για web development και SDKs, για ενσωμάτωση σε android και iOS εφαρμογές, καθώς και σε windows εφαρμογές. Ο προγραμματιστής έχει την ευχέρεια να επεξεργαστεί τους χάρτες, προσθέτοντας markers και icons, custom και μη, pop-up παράθυρα για προβολή πληροφοριών, σχήματα και γραμμές καθώς και overlays για την επίστρωση εικόνων και άλλων πληροφοριών. Για τη χρήση αυτών των υπηρεσιών, απαιτείται εγγραφή στο Bing Maps Dev Center με χρήση ενός Microsoft λογαριασμού, ενώ το είδος του API key που θα χρησιμοποιηθεί, καθορίζεται από την άδεια που θα επιλέξει ο χρήστης. Οι developers, με τη χρήση του Basic Key, δικαιούνται, αποκλειστικά για εκπαιδευτικούς ή μη κερδοσκοπικούς σκοπούς, 125,000 συναλλαγές, (οποιαδήποτε πράξη του χρήστη προς τον χάρτη μπορεί να τιμολογηθεί) για web και mobile εφαρμογές, 50,000 για windows εφαρμογές εντός 24 ωρών και 125,000 συναλλαγές για development και testing. Η Εικόνα 7 παρουσιάζει το Bing Map, με ανοιχτό το παράθυρο για τον υπολογισμό διαδρομής μεταξύ τοποθεσιών.

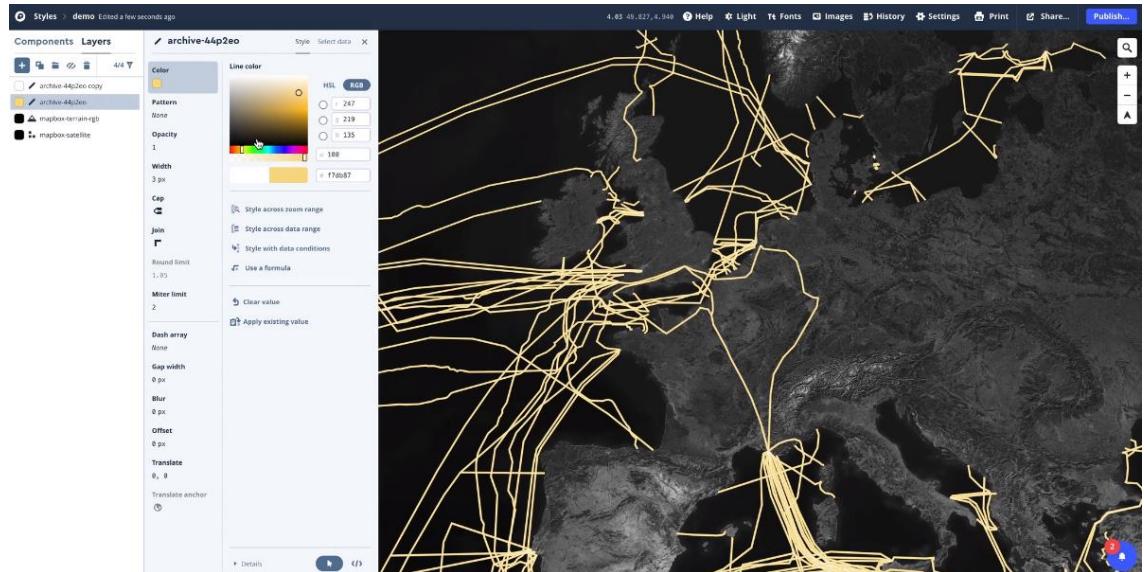


Εικόνα 7: Απεικόνιση Web έκδοσης του Microsoft Bing Map

## 2.4 MapBox

Το MapBox αποτελεί μια cloud πλατφόρμα παροχής υπηρεσιών χαρτών και τοποθεσίας. Παρέχει APIs και SDKs για την ενσωμάτωση των υπηρεσιών σε σελίδες και εφαρμογές, αλλά και μια αφιερωμένη εφαρμογή, το Mapbox Studio για την κατασκευή προσαρμοσμένων χαρτών, ανάλογα με τις εκάστοτε ανάγκες, χωρίς τη χρήση κώδικα. Τα δεδομένα των χαρτών συλλέγονται από ανοιχτές πηγές δεδομένων, όπως το Open Street Map, το Wikidata, το Microsoft Open Maps, αλλά και από ιδιωτικές πηγές και ιδρύματα (Tomasz Bąk, 2020). Η Εικόνα 8 δείχνει ένα παράδειγμα χρήσης του Mapbox Studio. Οι

κίτρινες γραμμές αντιπροσωπεύουν δρομολόγια πλοίων και έχουν εισαχθεί μέσω αντίστοιχου dataset.



Εικόνα 8: Απεικόνιση δρομολόγιων πλοίων μέσω του Mapbox Studio

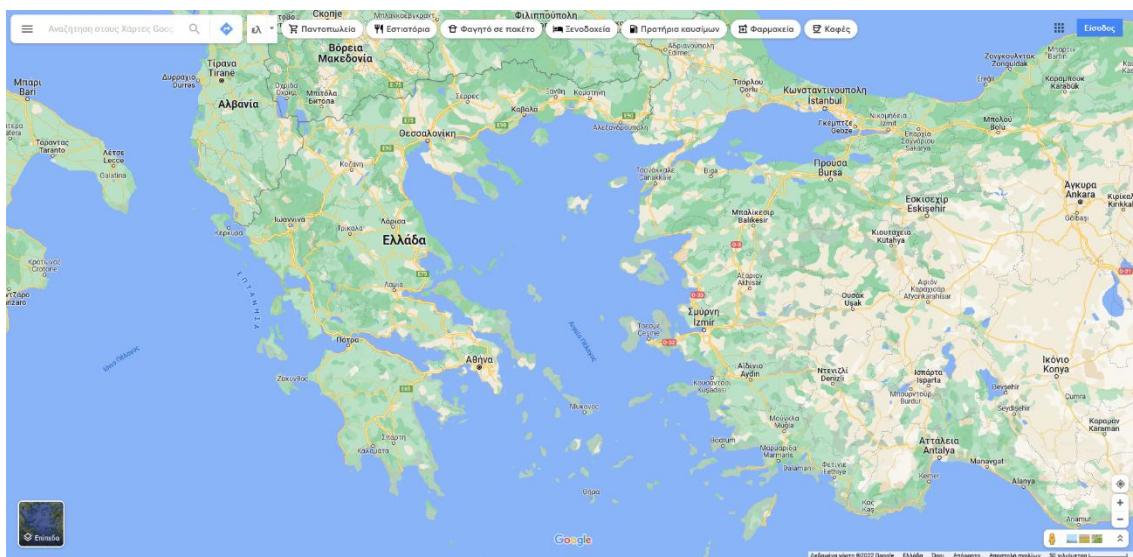
Οι χάρτες στο web εξυπηρετούνται μέσω του Mapbox GLJS, μιας javascript βιβλιοθήκης για vector maps, η οποία προσφέρει τη δυνατότητα για πλήρως διαδραστικούς χάρτες, 3D χάρτες και data visualization της επιθυμίας του προγραμματιστή, ενώ υπάρχει και η δυνατότητα χρήσης στατικών χαρτών. Τα παραπάνω είναι διαθέσιμα και για εφαρμογές android και iOS μέσω του Mobile Maps SDK. Επιπλέον, μέσω αυτών, γίνεται δυνατή η χρήση των υπηρεσιών πλοϊγησης με λειτουργίες όπως η ζωντανή εικόνα κίνησης και ατυχημάτων στους δρόμους, με την προτεινόμενη διαδρομή να ανανεώνεται σε πραγματικό χρόνο με βάση τα τρέχοντα δεδομένα, την turn-by-turn πλοϊγηση για ενημέρωση για κάθε αλλαγή στην πορεία της κατεύθυνσης, την ειδοποίηση υπέρβασης ορίων ταχύτητας αλλά και την ειδοποίηση άφιξης στον προορισμό. Επίσης, το MapBox προσφέρει προς χρήση και πληθώρα από ιδιόκτητα datasets όπως το Movement, που διαθέτει δεδομένα μετακίνησης πληθυσμών σε παγκόσμια κλίμακα, το Traffic Data, για δεδομένα της κίνησης στους δρόμους αλλά και το Boundaries, για την απεικόνιση συνόρων, συνοδευόμενα από πληροφορίες νομικής, διοικητικής και ταχυδρομικής φύσης. Τα δεδομένα αυτά μπορούν να χρησιμοποιηθούν για εμπλουτισμό εφαρμογών αλλά και για στατιστικούς και ερευνητικούς σκοπούς.

Η τιμολόγηση ακολουθεί το πλάνο του pay-as-you-go, με εκπτώσεις σε μεγάλους όγκους χρηστών. Κάθε υπηρεσία, μέσω των αντίστοιχων APIs και SDKs, διαθέτει ένα

όριο δωρεάν μηνιαίας χρήσης, όπως για παράδειγμα, 50,000 φορτώματα χάρτη στο web ή 25,000 ενεργοί χρήστες σε mobile εφαρμογές.

## 2.5 Google Maps

Το Google Maps αποτελεί μια web εφαρμογή χαρτών η οποία αναπτύσσεται και διατίθεται από την Google. Αποτελεί τη δημοφιλέστερη εφαρμογή για αυτή τη χρήση, συγκεντρώνοντας 154,5 εκατομμύρια χρήστες μηνιαίως, τριπλάσιους από ότι καταφέρνουν οι δύο μεγαλύτεροι ανταγωνιστές του μαζί, το Waze και το Apple Maps (Microsoft Voice Report, 2019). Πέρα από την web εφαρμογή, διατίθεται και στις android συσκευές, όπου και σε αυτήν την πλατφόρμα, αποτελεί την πιο διαδεδομένη εφαρμογή χαρτών, με 2,9 εκατομμύρια λήψεις από το Play Store στις Ηνωμένες Πολιτείες της Αμερικής, για το 2021 (L. Ceci, 2022). Το Maps, πέρα από την εμπορική του χρήση, προσφέρεται και ως υπηρεσία, υπό τη στέγη του Google Maps Platform. Σε αυτό συμπεριλαμβάνονται τα Google Routes, για τη δρομολόγηση και πλοήγηση, και το Google Places για την παρουσίαση πληροφοριών και αλληλεπίδραση με σημεία αναφοράς, όπως δημοφιλείς τοποθεσίες, αξιοθέατα καθώς και επιχειρήσεις και καταστήματα. Και τα τρία συστατικά κομμάτια, τα οποία συνδυαστικά αποτελούν αυτό που ένας μέσος χρήστης αντιλαμβάνεται ως Google Maps, διατίθενται δωρεάν για το κοινό. Στην Εικόνα 9, φαίνονται οι χάρτες της Google στην web εκδοχή τους.

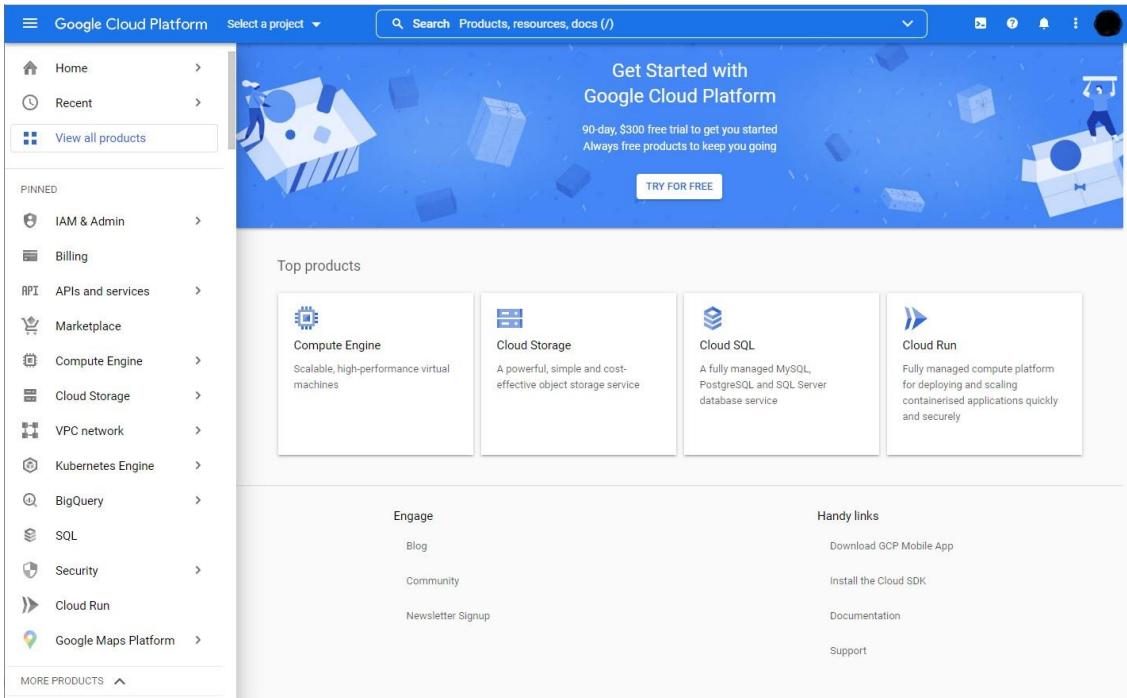


Εικόνα 9: Απεικόνιση του Google Maps, όπως φαίνεται μέσω browser

Ωστόσο, το ίδιο δεν ισχύει για τους developers που επιδιώκουν να αξιοποιήσουν και να ενσωματώσουν τις υπηρεσίες της google στις εφαρμογές τους. Οι δυνατότητες κάθε

υπηρεσίας εκτίθενται στον προγραμματιστή μέσω APIs για web ενσωμάτωση και SDKs για κινητές συσκευές και tablets, τόσο Android όσο και iOS. Προαπαιτούμενο της χρήσης τους, όμως, αποτελεί, προφανώς, ένας google λογαριασμός, μέσω του οποίο αποκτά κανείς πρόσβαση στο Cloud Platform Console. Το Google Cloud Platform Console αποτελεί έναν διαχειριστικό πίνακα για όλες τις υπηρεσίες και τα προϊόντα που προσφέρει και διαθέτει, αντίστοιχα, η Google. Μέσω του Console, όπως φαίνεται στην Εικόνα 10, μπορεί ο πελάτης να δημιουργήσει ένα νέο project κα αφού ενεργοποιήσει την τιμολόγηση, είναι σε θέση να προσθέσει στο project του τα APIs και τα SDKs που επιθυμεί. Το τελευταίο βήμα της διαδικασίας είναι η έκδοση ενός API key, ενός μοναδικού αναγνωριστικού το οποίο αυθεντικοποιεί τα αιτήματα που σχετίζονται με τα project του χρήστη, για λόγους παρακολούθησης χρήσης και τιμολόγησης (Google Developers, 2022). Όλες οι σχετιζόμενες με τους χάρτες υπηρεσίες της Google διατίθενται επί πληρωμή με ένα μηνιαία συνδρομητικό πλάνο κλιμακωτής χρέωσης ανάλογα με το πλήθος των αιτημάτων. Η χρέωση προσδιορίζεται με βάση συγκεκριμένων αιτήματα που γίνονται στα APIs. Παράδειγμα αποτελεί η χρήση των Dynamic Maps όπου η χρέωση γίνεται για κάθε φόρτωση του χάρτη, αλλά όχι για οποιαδήποτε αλληλεπίδραση με αυτόν, όπως το zoom, η αλλαγή layer η μετατόπιση του. Ωστόσο, υπάρχει μια ισχύουσα για όλους προσφορά, κατά την οποία προσφέρονται 300\$ για χρήση των υπηρεσιών, με ισχύ 90 ημερών ή μέχρι να καταναλωθεί το ποσό - οποιοδήποτε από τα δύο συμβεί πρώτο. Επιπλέον, προσφέρεται σταθερά, σε μηνιαία βάση, το ποσό των 200\$ για αξιοποίηση στις υπηρεσίες, το οποίο αντιστοιχεί σε 28,500 χιλιάδες φορτώματα χάρτη τον μήνα. Αν το ποσό αυτό ξεπεραστεί, ισχύει η κανονική χρέωση, όπως αυτή έχει οριστεί για την

εκάστοτε υπηρεσία. Εν ολίγοις, η Google κάνει δυνατή τη δωρεάν χρήση των υπηρεσιών της για μικρής κλίμακας εφαρμογές με αντίστοιχα μικρό user base.



Εικόνα 10: Google Cloud Platform Console

## 2.6 OpenStreetMap

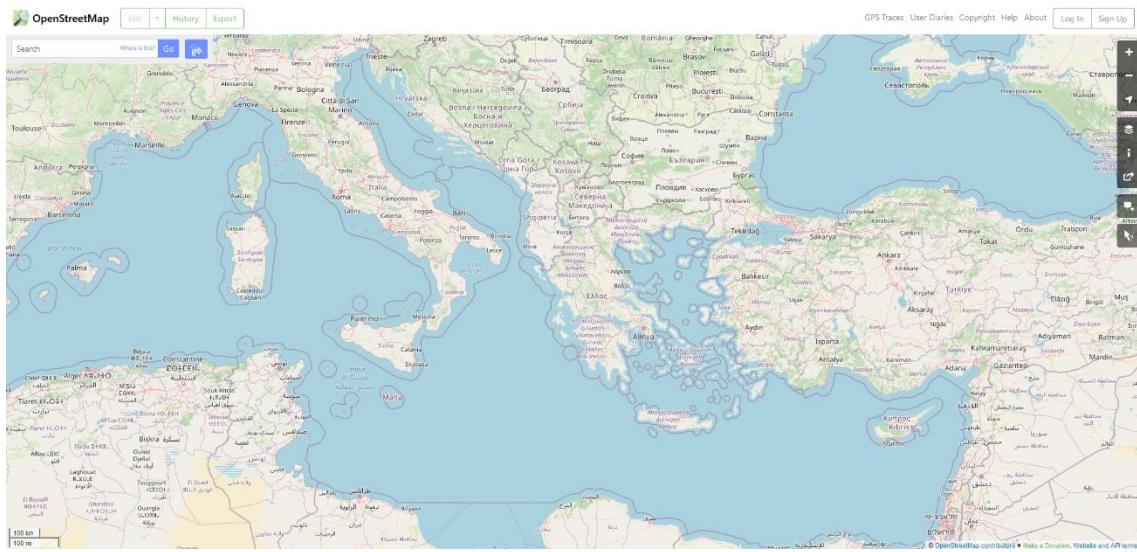
Το OpenStreetMap είναι ένας δωρεάν, επεξεργάσιμος παγκόσμιος χάρτης, κατασκευασμένος από εθελοντές υπό την άδεια ανοιχτού περιεχομένου (OpenStreetMap Official Wiki, 2022) που μετράει πάνω από 8 εκατομμύρια ενεργούς χρήστες (OpenStreetMap stats, 2022). Η ιδέα πίσω από την δημιουργία και διάθεση δωρεάν χαρτών βασίζεται στη αρχή δωρεάν διάθεσης δημοσίων αγαθών. Συνήθως, τα γεωγραφικά δεδομένα παρέχονται από κυβερνητικά πρακτορεία ή τρίτους, οι οποίοι πωλούν τα δεδομένα στους πολίτες, με αποτέλεσμα, ο πολίτης να πληρώνει για αυτά τα μέσω φόρων καθώς και κατ’ απαίτηση, όταν χρησιμοποιεί τις εκάστοτε υπηρεσίες. Επιπλέον, ένας χρήστης δεν μπορεί να επέμβει σε αυτά τα δεδομένα ούτε να τα χρησιμοποιήσει ελεύθερα για προσωπικούς σκοπούς. Η πρωτοβουλία αυτή στηρίζεται σχεδόν εξ’ ολοκλήρου σε άτομα που επεκτείνουν το project εθελοντικά, ενώ έχουν πραγματοποιηθεί και σποραδικές εισαγωγές δεδομένων από πηγές με άδειες διάθεσης ανοιχτού τύπου (OpenStreetMap Official Wiki, 2022). Οι χάρτες του OpenStreetMap υποστηρίζονται από τους χρήστες και προορίζονται για αυτούς, οπότε δεν υπάρχει κάποιο ουσιαστικό κίνητρο για εξαπάτηση η παροχή ψευδών πληροφοριών. Ακόμα και

σε αυτήν την περίπτωση, πραγματοποιείται έλεγχος από τους ίδιους, καθώς είναι προς το κοινό όφελος να υπάρχει ακρίβεια στα δεδομένα. Η ακρίβεια αυτή, ωστόσο, είναι τόσο μεγάλη όσο η κάλυψη των περιοχών από τους εθελοντές και η ενασχόλησή τους. Οι χάρτες βασίζονται σε μια εν γενώς απροσδιόριστου βαθμού ακρίβειας διαδικασία, καθώς μπορούν να συμβούν λάθη από την κοινότητα λόγω πειραματισμού ή απροσεξίας, ωστόσο, αυτή η διαδικασία εύκολης και άμεσης παρέμβασης στα δεδομένα, τόσο για εισαγωγή νέων όσο και για την εξακρίβωση τους, καθιστά το OpenStreetMap γρήγορα προσαρμόσιμο σε αλλαγές. Η προστασία, προαγωγή και υποστήριξη του project γίνεται από το OpenStreetMap Foundation, το οποίο σε καμιά περίπτωση δεν οικειοποιείται των δεδομένων, αφού αυτά ανήκουν εξολοκλήρου στους χρήστες.

Για την επεξεργασία των χαρτών, χρησιμοποιείται πληθώρα από editors όπως o iD, μέσω web browser, οι Potlach και Merkaartor ως desktop εφαρμογές και ο Vespucci, για τις android συσκευές. Για την χρήση των δεδομένων του OpenStreetMap και την ενσωμάτωση τους σε web εφαρμογές, η πιο δημοφιλής βιβλιοθήκη είναι το leafletjs, ενώ για android και iOS, υπάρχει πληθώρα από SDKs κατηγοριοποιημένα με βάση της λειτουργίες που προσφέρουν, όπως ποροβολή και αλληλεπίδραση με τους χάρτες, δρομολόγηση και geocoding. Τα APIs και τα SDKs αποτελούν κατά πλειοψηφία προϊόντα παραγόμενα από την κοινότητα του OpenStreetMap και υπάγονται στις αντίστοιχες άδειες. Φυσικά, υπάρχουν και τα ιδιόκτητα, ωστόσο ο προγραμματιστής έχει ποικιλία επιλογών.

Ο χάρτης είναι διαθέσιμος σαν web εφαρμογή, όπως φαίνεται στην Εικόνα 11, και διαθέτει όλες τις βασικές λειτουργίες ενός σύγχρονου χάρτη. Διαθέτει αναζήτηση τόσο για γεωγραφικές περιοχές όσο και για καταστήματα, υπηρεσίες και σημαντικά σημεία, επιλογές για δρομολόγηση και βέλτιστη διαδρομή μεταξύ σημείων, σημάνσεις για δημόσια κτήρια, μνημεία και τοποθεσίες δημοσίου ενδιαφέροντος, θεματικά διαχωριζόμενη πληροφορία μέσω των layers, καθώς και επιλογή για δημιουργία δημόσιας σήμανσης/επισήμανσης/σχολίου στην επιλεγόμενη από τον χρήστη τοποθεσία. Οι σημάνσεις αυτές μπορούν να χρησιμοποιηθούν για την επέκταση του χάρτη, καθώς ενδεχομένως, περιέχουν σχετική με την τοποθεσία πληροφορία, που έχει παραληφθεί μέχρι την εκάστοτε χρονική στιγμή. Όσον αφορά τις εφαρμογές των κινητών συσκευών, υπάρχουν εφαρμογές που χρησιμοποιούν τους χάρτες μέσω των SDKs, όπως το MapsMe, τόσο ως offline εναλλακτική όσο και ως online. Ωστόσο, η web εφαρμογή είναι προσβάσιμη από οποιονδήποτε browser, ανεξαρτήτου πλατφόρμας.

Τέλος, η χρήση των υπηρεσιών και των γεωγραφικών δεδομένων είναι δωρεάν και δεν υπάρχουν περιορισμοί στη χρήση των APIs και SDKs.



Εικόνα 11: Απεικόνιση Web εκδοχής του OpenStreetMap

# 3 Αντικείμενο πτυχιακής

Στο κεφάλαιο 2, προηγήθηκε η ανάλυση μερικών από τους πιο διαδεδομένους ηλεκτρονικούς χάρτες που παρέχονται στο κοινό, μέσω του διαδικτύου. Η πτυχιακή αυτή, ωστόσο, καλείται να εισάγει τον αναγνώστη στις προγραμματιστικές δυνατότητες και στα στοιχεία που παρέχουν αυτοί οι χάρτες, καθώς και να του δώσει μια γενική ιδέα, όχι μόνο για το πως θα μπορούσε να τους χρησιμοποιήσει σε άλλες εφαρμογές, αλλά και για τον τρόπο με τον οποίο θα μπορεί να τους μελετήσει περαιτέρω, μέσω των πηγών που παρέχουν αυτοί.

Για την επίτευξη του παραπάνω σκοπού, έχουν επιλεχθεί δύο χάρτες από τη λίστα που παρουσιάστηκε στο δεύτερο κεφάλαιο. Αυτοί είναι το Google Maps και το OpenStreetMap. Μέσω των δύο αυτών χαρτών, θα αναλυθούν οι βασικότερες κοινές προγραμματιστικές λειτουργίες που παρέχουν, αφού πρώτα γίνει επεξήγηση για τον τρόπο εγκατάστασής τους σε ένα πρόγραμμα. Επιπλέον, η παραπάνω σύγκριση δεν θα ήταν ολοκληρωμένη, εάν δεν υπήρχε και η αναφορά των διαφορών τους.

Στις επόμενες ενότητες αυτού του κεφαλαίου, γίνεται αναφορά στους λόγους τους οποίους επιλέχθηκαν αυτοί οι χάρτες και στην αντιπαραβολή αυτών. Ο αναγνώστης θα είναι σε θέση να κατανοήσει την επιλογή των χαρτών, οι οποίοι αν και τόσο διαφορετικοί, είναι ταυτόχρονα και τόσο όμοιοι μεταξύ τους.

## 3.1 Επιλογή Χαρτών

Η επιλογή των δύο χαρτών πραγματοποιήθηκε από την λίστα των επιλογών που παρουσιάστηκαν στο κεφάλαιο 2. Όπως προαναφέρθηκε, η τελική επιλογή αφορά το Google Maps και το OpenStreetMap. Η τελική επιλογή πραγματοποιήθηκε με βάση τα παρακάτω κριτήρια:

- *Ποικιλία επιλογών και χαρακτηριστικών.* Οι συγκρινόμενοι χάρτες είναι απαραίτητο να έχουν μια γκάμα από επιλογές προς σύγκριση, ώστε ο αναγνώστης να αποκομίσει τις περισσότερες δυνατές γνώσεις.
- *Δημοτικότητα.* Το πόσο διαδεδομένος είναι ένας χάρτης είναι σημαντικό κριτήριο, καθώς όσο πιο δημοφιλής, τόσο περισσότερη υποστήριξη και περιεχόμενο

αναμένεται να παρέχεται σε αυτόν, από την κοινότητα των προγραμματιστών και από τους ίδιους τους δημιουργούς του.

- **Συγκρισιμότητα.** Οι δυο χάρτες θα πρέπει να περιέχουν αρκετά όμοια χαρακτηριστικά και λειτουργίες, ώστε να είναι συγκρίσιμοι και να μπορεί ο αναγνώστης να επιλέξει ανάμεσα σε δύο διαφορετικά εργαλεία με παρόμοιες δυνατότητες.
- **Ποικιλομορφία λογισμικού.** Αναφέρεται στη διαφορετικότητα των δυο επιλογών, ως προς το είδος του λογισμικού που αποτελούν. Για παράδειγμα, λογισμικό ανοιχτού κώδικα ή όχι.

Λαμβάνοντας λοιπόν, ως πρώτο κριτήριο την ποικιλία επιλογών και χαρακτηριστικών, διακρίνονται αυτόματα οι χάρτες OpenStreetMap, Google Maps, Microsoft Bing Maps και Apple maps. Οι υπόλοιπες επιλογές δεν μπορούν να συγκριθούν με τις παραπάνω στη ποικιλία χαρακτηριστικών ή τη δημοσιότητα. Ειδικά οι χάρτες Waze και Mapbox, όπως αναφέρθηκε στο 2<sup>o</sup> κεφάλαιο, αποτελούν χάρτες που εξυπηρετούν κάποιες συγκεκριμένες λειτουργίες, όπως ενημερώσεις για την κατάσταση των δρόμων σε πραγματικό χρόνο ή την παροχή εφαρμογής χάρτη μέσω datasets, εξαγόμενα από άλλους χάρτες, αντίστοιχα.

Καθώς γίνεται αναφορά στη δημοσιότητα, ως πρώτος χάρτης επιλέγεται το Google Maps, καθώς, από τους τέσσερεις που ξεχώρισαν, αυτός είναι αυτός με τη μεγαλύτερη δημοτικότητα, αλλά και την μεγαλύτερη υποστήριξη από πλευράς λογισμικού.

Τέλος, η δεύτερη επιλογή έγινε με βάση το τέταρτο κριτήριο, δηλαδή αυτό της ποικιλομορφίας του λογισμικού. Ουσιαστικά, οι χάρτες Google Maps, Microsoft Bing Maps και Apple Map, έχουν δημιουργηθεί από κερδοσκοπικές εταιρίες, δηλαδή από την Google, τη Microsoft και την Apple αντίστοιχα. Συνεπώς, οι υπηρεσίες που παρέχουν, έρχονται με κάποιο αντίτιμο, ακολουθώντας κάποιο συνδρομητικό πλάνο. Όπως για παράδειγμα, το Google Maps, το οποίο απαιτεί εγγραφή και παροχή τρόπου πληρωμής, ώστε να χρησιμοποιήσει κανείς το API που παρέχει. Σε περίπτωση που κάποιος ξεπεράσει το όριο των ελάχιστων υπηρεσιών που διαθέτει το δωρεάν συνδρομητικό πλάνο της Google για τους χάρτες της, ο εγγεγραμμένος χρήστης χρεώνεται και καλείται να πληρώσει για τις επιπλέον υπηρεσίες, μέσω του τρόπου πληρωμής που εισήγαγε κατά την εγγραφή του. Αντιθέτως, ο χάρτης που παρέχεται από το Open Street Maps, όπως αναφέρθηκε στο 2<sup>o</sup> κεφάλαιο, είναι ανοιχτού κώδικα και είναι μη κερδοσκοπικός. Δε χρειάζεται κάποιο συνδρομητικό πλάνο ώστε να έχει κάποιος πρόσβαση στο API τους

και είναι πολύ εύκολος στην αρχική εγκατάσταση. Αναπτύσσεται συνεχόμενα από μια ενεργή κοινότητα προγραμματισμών και διατίθεται δωρεάν. Σε αυτό το σημείο είναι ενδιαφέρον να αναφερθεί πως, οποιοσδήποτε προγραμματιστής μπορεί να συμβάλει στην ανάπτυξη του λογισμικού του και να γίνει επίσημος contributor του project OpenStreetMap.

Εν κατακλείδι, η δεύτερη επιλογή αποτελεί τον χάρτη του Open Street Maps, εξαιτίας της διαφορετικότητας της φύσης του λογισμικού του αλλά και για τον αριθμό όμοιων χαρακτηριστικών και δυνατοτήτων που έχει με τον χάρτη GoogleMaps. Στην επόμενη ενότητα, αναφέρονται κάποια παραδείγματα αυτών των δυνατοτήτων μέσω των διαδεδομένων APIs που παρέχουν το GoogleMaps και το OpenStreetMap.

## 3.2 Αντιπαραβολή χαρτών Google Maps και Open Street Maps

Όπως προειπώθηκε, οι χάρτες αν και τόσο διαφορετικοί ως προς τη φύση του λογισμικού τους έχουν πολλές όμοιες δυνατότητες. Η σύγκριση αυτών των χαρτών θα γίνει μέσω του *Maps Javascript API* της Google και του API της βιβλιοθήκης *Leaflet.js*, που αξιοποιεί τον ανοικτό κώδικα του OpenStreetMap.

Και οι δύο χάρτες δίνουν στο προγραμματιστή επιλογές για προσαρμογή, αλλά και δυνατότητες τυπώματος διαδραστικών ενδείξεων πάνω στον χάρτη, που κάνουν την εμπειρία του χρήστη πλουσιότερη. Αναφορικά και δύο χάρτες δίνουν δυνατότητες για τα παρακάτω:

- Αρχικοποίηση ( initialization ) Χάρτη, που αποτελεί την αρχική εμφάνιση ενός προεπιλεγμένου χάρτη πάνω στον καμβά.
- Παραμετροποίηση χάρτη, ως προς των είδος των κουμπιών ελέγχου (zooming) που εμφανίζονται πάνω σε αυτόν, αλλά και για το είδος του χάρτη πχ. πολιτικός, γεωγραφικός κ.λπ.
- Εισαγωγή σημείων Markers πάνω στο χάρτη ή αλλιώς τις λεγόμενες πινέζες.
- Εισαγωγή Info Windows ή αλλιώς παράθυρα πληροφοριών, που μπορούν να εμφανίζουν πληροφορίες για γεωγραφικές τοποθεσίες.
- Εισαγωγή Shapes ή αλλιώς σχημάτων, τα οποία καλύπτουν γεωγραφικές περιοχές
- Εισαγωγή Overlays ή αλλιώς επικαλύψεων πάνω στο χάρτη με τη προσθήκη εικόνας, βίντεο ή κάποιας άλλης μορφής media.

Βεβαίως, η εφαρμογή και η εισαγωγή των παραπάνω χαρακτηριστικών πραγματοποιείται με διαφορετικούς τρόπους μέσω κώδικα και με διαφορετική ανάγνωση του documentation που παρέχουν τα δύο APIs. Ωστόσο, εκτός από αυτές τις ομοιότητες, οι δύο λύσεις, έχουν και πολλές διαφορές, τόσο στις λειτουργίες που έχει ο ένας και όχι ο άλλος, όσο και στον τρόπο που υλοποιούνται και παρέχονται κάποια όμοια χαρακτηριστικά, είτε μέσω τρίτων APIs ή βιβλιοθηκών.

Το 4<sup>o</sup> κεφάλαιο, καλείται να καλύψει αναλυτικά και με παραδείγματα κώδικα, όλες τις παραπάνω ομοιότητες, αλλά και να τονίσει κάποιες από τις σημαντικότερες διαφορές των δυνατοτήτων των API τους.

# 4 Αντιπαραβολή Google Maps και Open Street Maps

Στο κεφάλαιο αυτό θα γίνει αντιπαραβολή των δύο εν λόγω εφαρμογών χαρτών. Θα αναφερθούν, με παραδείγματα, τα σημαντικότερα χαρακτηριστικά που από κοινού μοιράζονται και πως υλοποιούνται σε κάθε περίπτωση, καθώς και οι διαφορές και οι ανεπάρκειες από τις δύο πλευρές. Τα χαρακτηριστικά που θα αναλυθούν, είναι τα στοιχεία ελέγχου/αλληλεπίδρασης του χρήστη με τον χάρτη, τα *markers*, τα *symbols*, τα *pop-up* παράθυρα, τα σχήματα (γραμμές, πολύγωνα, τετράγωνα, κύκλοι), τα *overlays* και τα *layers*. Σκοπός δεν είναι η ανάλυση κάθε λεπτομέρειας των χαρακτηριστικών, καθώς η παρούσα εργασία δεν αποτελεί το documentation της κάθε πλατφόρμας, αλλά η παρουσίαση των κυριότερων εξ αυτών.

## 4.1 Αρχικοποίηση Χάρτη

Η αρχικοποίηση του χάρτη είναι σχεδόν πανομοιότυπη και στις δύο περιπτώσεις. Προϋπόθεση είναι η ύπαρξη ενός DIV HTML tag, στο οποίο θα γίνει render ο χάρτης, με τα options που έχει αρχικοποιηθεί.

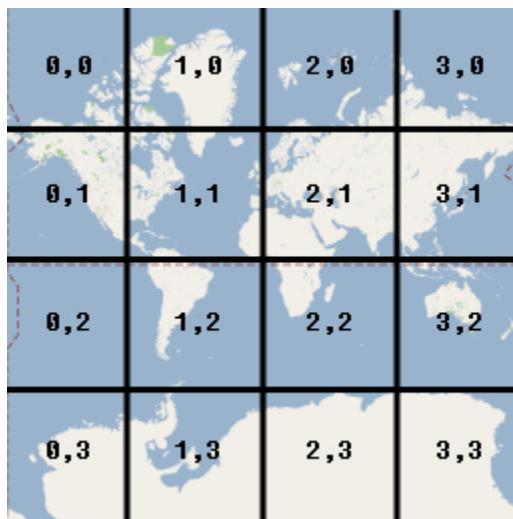
### 4.1.1 Google Maps

Για να αρχικοποιηθεί ο χάρτης, απαιτούνται δύο βασικά συστατικά. Το πρώτο είναι ένα options αντικείμενο, που περιλαμβάνει τις παραμετροποιήσεις του χάρτη. Οι βασικές παράμετροι είναι:

- *Zoom*. Αναπαριστά το επίπεδο του zoom του χάρτη κατά την αρχικοποίηση και διαθέτει συνολικά 22 επίπεδα.
- *Center*. Προσδιορίζει το κέντρο του χάρτη κατά την φόρτωση του. Αποτελεί ένα geopoint, δηλαδή ένα αντικείμενο συντεταγμένων.
- *MapTypeId*. Προσδιορίζει τον τύπο του χάρτη. Μπορεί να πάρει τέσσερις προκαθορισμένες τιμές, ωστόσο ο προγραμματιστής μπορεί να σχεδιάσει τον δικό του τύπο. Ο τύπος καθορίζει τα tiles που θα χρησιμοποιηθούν από τον χάρτη. Τα tiles αποτελούν λογικές υποδιαιρέσεις του χάρτη, οι οποίες συμβάλλουν στην αποδοτική φόρτωση του, ανεξαρτήτως του επιπέδου zoom. Οι τέσσερις βασικοί

τύποι είναι το roadmap, το οποίο αποτελεί την default επιλογή, το satellite, το οποίο χρησιμοποιεί εικόνες του Google Earth, το hybrid το οποίο προσφέρει μια μίξη των δύο προηγούμενων και το terrain, που προβάλλει έναν φυσικό χάρτη βασισμένο σε δεδομένων του εδάφους. Στην Εικόνα 12 φαίνεται ο διαμερισμός του χάρτη σε tiles με καθώς και η οργάνωση τους με βάση τις συντεταγμένες.

Το δεύτερο συστατικό είναι ένα DOM element, συνήθως ένα DIV, για την υποδοχή του χάρτη και την προβολή του.



Εικόνα 12: Παράδειγμα των tile coordinates

#### 4.1.2 OpenStreetMap

Για τη δημιουργία ενός barebone χάρτη, όπως και στο Google Maps, απαιτείται ένα DIV HTML tag, ενώ το options αντικείμενο είναι προαιρετικό. Τα βασικά ορίσματα του options αντικειμένου είναι τα:

- *Center*. Το αρχικό κέντρο του χάρτη, κατά τη φόρτωσή του. Αποτελεί ένα LatLng αντικείμενο, ένα geopoint δηλαδή. Ωστόσο, η παροχή των συντεταγμένων σε μορφή πίνακα είναι επίσης αποδεκτή.
- *Zoom*. Το αρχικό επίπεδο zoom του χάρτη.

Σε περίπτωση που δεν δοθεί κάποιο options αντικείμενο, για να προβληθεί ο χάρτης, πρέπει να κληθεί η μεθόδος setView() του map αντικειμένου, η οποία παίρνει σαν παράμετρο τις συντεταγμένες για το κέντρο του χάρτη και το επίπεδο του zoom.

## 4.2 Στοιχεία Ελέγχου Χάρτη

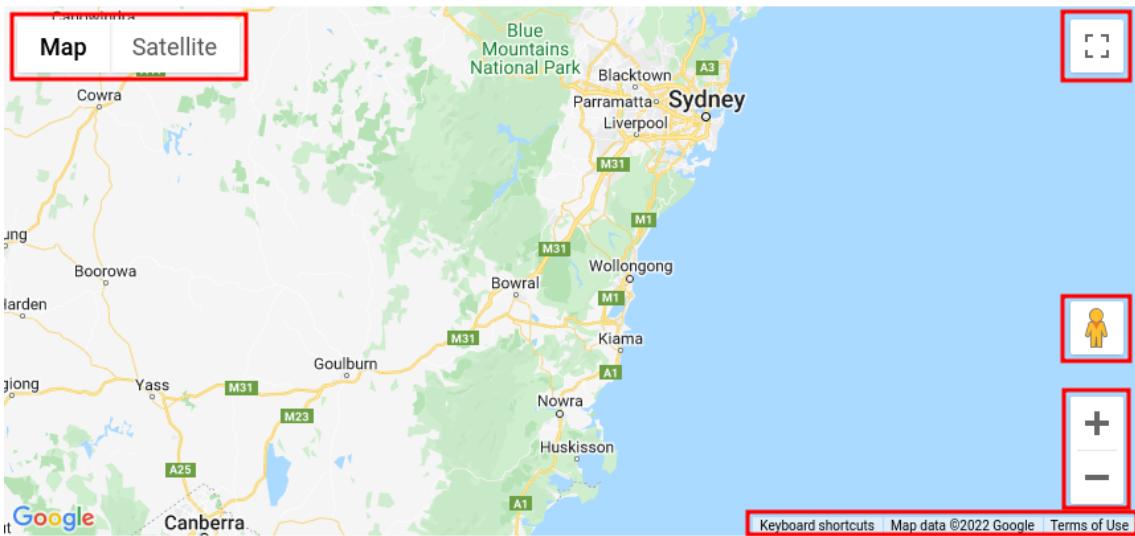
Τα στοιχεία ελέγχου αναφέρονται στα UI στοιχεία, τα οποία επιτρέπουν την αλληλεπίδραση του χρήστη με τον χάρτη. Μέσω αυτών των στοιχείων, ο χρήστης μπορεί να επιδράσει στην εμφάνισή του και να επιλέξει τι είδους πληροφορία θα προβάλλεται, αλλάζοντας layers, μεταβάλλοντας το zoom, τη φορά του χάρτη και πολλά ακόμα που θα αναλυθούν παρακάτω.

### 4.2.1 Google Maps

Ο Google Maps χάρτης, αρχικοποιείται με συγκεκριμένα προκαθορισμένα στοιχεία ελέγχου. Για την απενεργοποίησή τους, ο χρήστης πρέπει απλά να δώσει την τιμή *true* στην παράμετρο *disableDefaultUI* του *options* αντικειμένου του χάρτη. Τα προκαθορισμένα στοιχεία είναι:

- *Map Type*. Επιτρέπει στον χρήστη να επιλέξει μεταξύ τεσσάρων προκαθορισμένων, βασικών τύπων χάρτη. Αυτοί είναι οι Roadmap, Satellite, Hybrid και Terrain.
- *Zoom*. Κάνει δυνατή τη ρύθμιση του επιπέδου zoom.
- *Fullscreen*. Δίνει τη δυνατότητα μετάβασης του χάρτη σε προβολή πλήρους οθόνης.
- *Street View*. Ο χρήστης μπορεί να σύρει ένα κίτρινο ανθρωπάκι, λεγόμενο Pegman, σε οποιοδήποτε σημείο του χάρτη και αμέσως θα προβληθεί το Street View του σημείου αυτού, αν είναι διαθέσιμο.
- *Attribution Bar*. Αποτελεί μια μπάρα στο κάτω μέρος του χάρτη η οποία περιλαμβάνει:
  - (1) *Keyboard Shortcuts*. Πατώντας το κουμπί αυτό, εμφανίζεται ένα dialog box με όλες τις διαθέσιμες πληροφορίες για την αλληλεπίδραση του χάρτη μέσω συντομεύσεων του πληκτρολογίου.
  - (2) *Terms of Use*. Προβάλει τους όρους χρήσης της εφαρμογής.
  - (3) *Report a map error*. Χρησιμοποιείται για την αναφορά σφάλμα των σχετικά με τον χάρτη.
  - (4) *Copyright*. Το Copyright σχετικά με τα δεδομένα του χάρτη.

Στην Εικόνα 13 φαίνονται τα στοιχεία αυτά, επισημασμένα.



Εικόνα 13: Default controls του Google Maps

Λοιπά στοιχεία αποτελούν τα:

- *Rotate*. Προσφέρει επιλογές για την περιστροφή του χάρτη και την κλίση του
- *Scale*. Εμφανίζει την κλίμακα του χάρτη, με βάση το τρέχον επίπεδο zoom, στο Attribution Bar. Ο χρήστης μπορεί να εναλλάσσει τη μονάδα μέτρησης μεταξύ χιλιομέτρων και μιλίων, κάνοντας κλικ πάνω του.

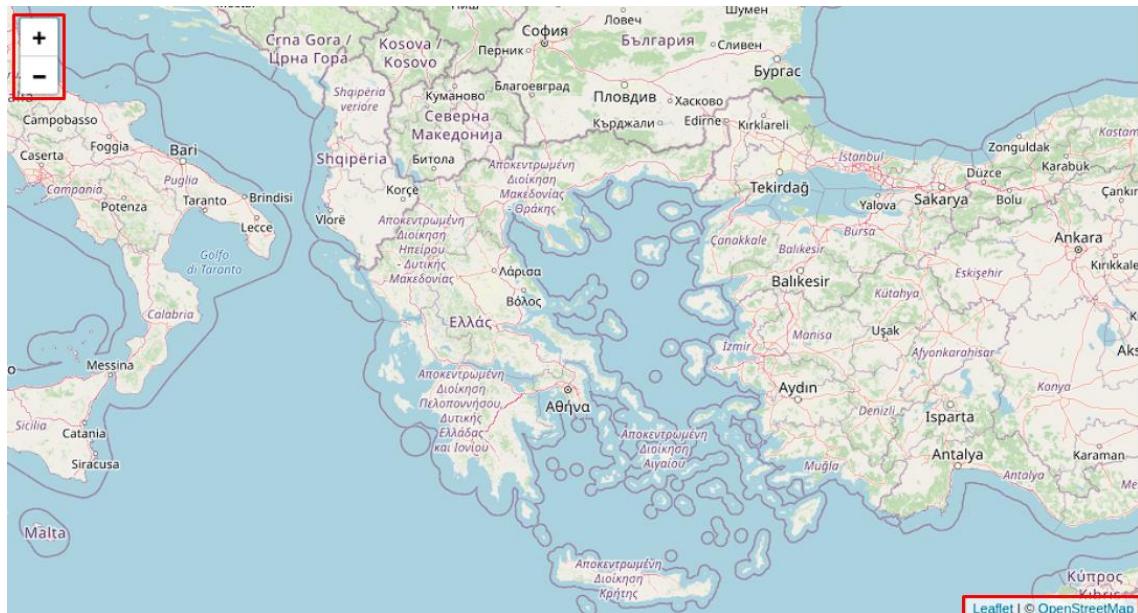
Ο προγραμματιστής δεν μπορεί να επιδράσει άμεσα σε αυτά τα στοιχεία. Μπορεί να ελέγξει όμως ποια από αυτά θα εμφανίζονται ανά πάσα στιγμή, που θα τοποθετούνται και κατά περίπτωση, πως θα φαίνονται. Αυτό γίνεται θέτοντας την κατάλληλη boolean τιμή στις αντίστοιχες παραμέτρους του options αντικειμένου του χάρτη, ενώ η μορφοποίηση κάθε στοιχείου πραγματοποιείται μέσω του δικού του options αντικειμένου. Τέλος, ο προγραμματιστής μπορεί να δημιουργήσει τα δικά του στοιχεία ελέγχου. Τα custom στοιχεία δεν είναι παρά DIV tags, τα οποία έχουν absolute (απόλυτες) θέσεις στον χάρτη. Διαχειρίζονται την αλληλεπίδραση με τον χρήστη μέσω event handlers, ενώ η μορφοποίηση τους γίνεται μέσω CSS. Η Εικόνα 14 παρουσιάζει ένα custom κουμπί, το οποίο επιστρέφει τον χάρτη στην κεντρική του θέση.



Εικόνα 14: Custom στοιχείο ελέγχου

#### 4.2.2 OpenStreetMap

Ένας OpenStreetMap χάρτης έρχεται με δύο στοιχεία ελέγχου. το *Zoom*, για την αυξομείωση του επιπέδου του zoom και το *Attribution Bar*, για την αναγνώριση των παρόχων του εκάστοτε layer, με το κείμενο να τίθεται αυτόμata κατά περίπτωση, μέσω της μεθόδου `getAttribution`. Η Εικόνα 15 δείχνει τα default controls.



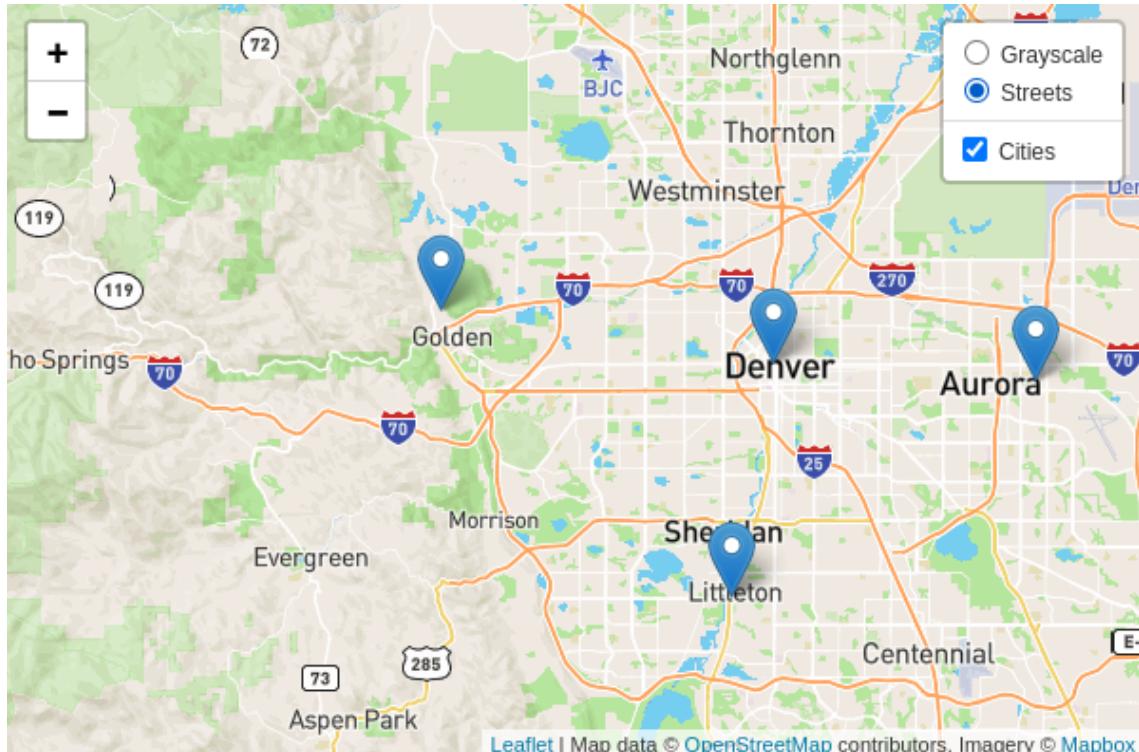
Εικόνα 15: Default controls του OpenStreetMap

Λοιπά στοιχεία αποτελούν τα:

- *Layers*. Δίνουν τη δυνατότητα στον χρήστη να εναλλάσσει τα layers και να ενεργοποιεί/απενεργοποιεί τα διαθέσιμα overlays, όπως φαίνεται στην Εικόνα 16.

- *Scale*. Εμφανίζει την τρέχουσα κλίμακα του χάρτη, με βάση το επίπεδο zoom, στο μετρικό σύστημα (m/km) και στο αυτοκρατορικό (mi/ft).

Για κάθε επιλογή, προσφέρονται options για την παραμετροποίηση των στοιχείων, καθώς και οι μέθοδοι για τη διαχείριση τους.



Εικόνα 16: Εναλλαγή μεταξύ map layers

## 4.3 Markers

Οι markers είναι στοιχεία που προσδιορίζουν ένα σημείο στον χάρτη. Μπορούν να έχουν μια προκαθορισμένη εικόνα, καθώς και κάποια που θα ορίσει ο χρήστης. Είναι, εν γενώς, διαδραστικά στοιχεία, καθώς διαθέτουν onClick events για την παρουσίαση, σχετικής με την τοποθεσία, πληροφορίας.

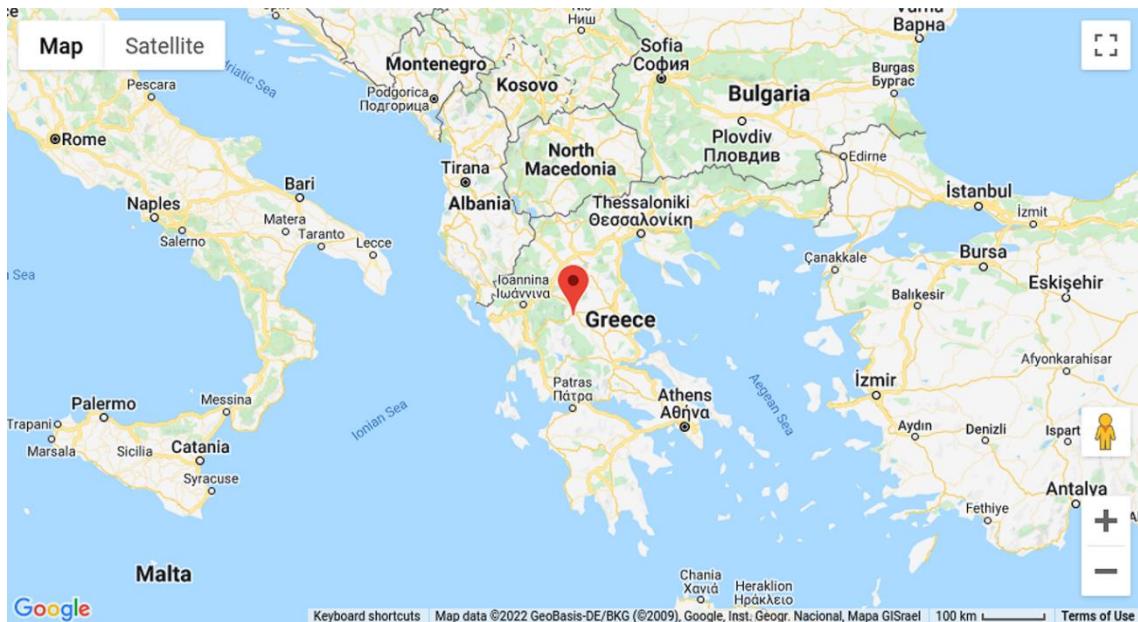
### 4.3.1 Google Maps

Ένας marker, κατά την αρχικοποίηση του, δέχεται ένα σύνολο από παραμέτρους. Οι πιο βασικές και συνηθισμένες είναι το position, το οποίο ορίζει τις συντεταγμένες για τη θέση του στον χάρτη, το title, που θέτει τον τίτλο του, καθώς και το map, το οποίο δεν είναι παρά ο χάρτης στον οποίο θα προσκολληθεί ο marker. Το map μπορεί να παραληφθεί, στην περίπτωση αυτή, όμως, ο marker απλά θα αρχικοποιηθεί, χωρίς να συσχετιστεί με κάποιο χάρτη. Για την μετέπειτα ανάθεση του σε χάρτη, πρέπει να κληθεί η setMap()

συνάρτηση του αντικειμένου marker, με παράμετρο τον εν λόγω χάρτη. Λοιπές παράμετροι είναι τα εξής:

- *Icon*. Ορίζει ένα εναλλακτικό εικονίδιο για τον marker.
- *Label*. Ορίζει ένα γράμμα ή αριθμό που θα φαίνεται μέσα στον marker.
- *Optimized*. Αποτελεί μια boolean παράμετρο που βελτιώνει την απόδοση του χάρτη, κάνοντας render πολλούς markers ως στατικά στοιχεία, χρήσιμο σε περιπτώσεις όπου χρησιμοποιείται μεγάλο πλήθος από αυτούς.
- *Draggable*. Δίνει τη δυνατότητα στον χρήστη να σύρει και να αλλάξει θέση τον marker κατά βούληση.
- *Animation*. Μέσω του μπορεί να οριστεί ένα animation από τη συλλογή google.maps Animation, τόσο για τη είσοδο του όσο και για άλλα events, όπως το click.

Για την αφαίρεση ενός marker από τον χάρτη, καλείται η μέθοδος setMap(), με παράμετρο null. Ωστόσο, αυτό δεν καταστρέφει τον marker εντελώς, απλώς τον αφαιρεί από τον χάρτη. Για να γίνει αυτό, πρέπει να τεθεί η αναφορά του marker, αυτού καθεαυτού, σε null. Η Εικόνα 17 δείχνει έναν marker στον χάρτη.



Εικόνα 17: Google Maps Marker

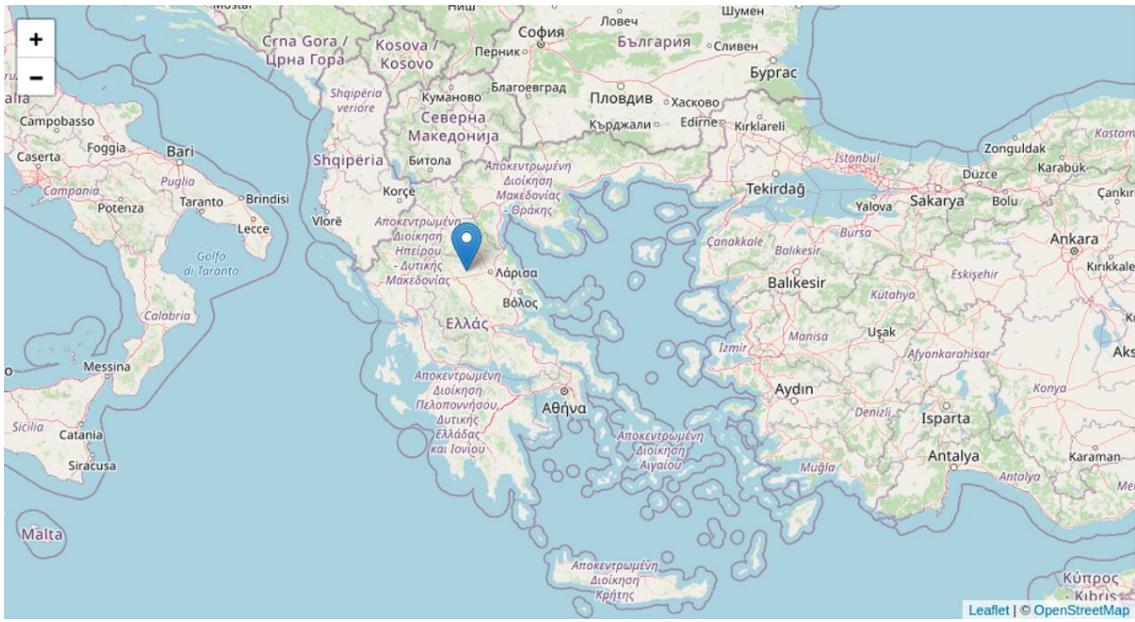
### 4.3.2 OpenStreetMap

Η αρχικοποίηση ενός marker απαιτεί μόνο τις συντεταγμένες που θα τοποθετηθεί, σε μορφή πίνακα ή LatLang, geopoint δηλαδή, αντικειμένου. Προαιρετικά, μπορεί να

χορηγηθεί ένα αντικείμενο options, για την περαιτέρω μορφοποίηση του. Το αντικείμενο αυτό δέχεται παραμέτρους όπως:

- *Icon*. Ορίζει ένα εναλλακτικό εικονίδιο για τον marker.
- *Keyboard*. Boolean παράμετρος που ορίζει αν ο marker θα αλληλεπιδρά με τα πλήκτρα tab και enter του πληκτρολογίου.
- *Title*. Το κείμενο που θα εμφανίζεται ως tooltip όταν ο χρήστης κάνει hover πάνω από τον marker.
- *ZIndexOffset*. Καθορίζει τη θέση του marker στον z άξονα, σε σχέση με τους υπόλοιπους markers.
- *Opacity*. Ορίζει τη διαφάνεια του marker.
- *Pane/ShadowPane*. Το επίπεδο στο οποίο θα τοποθετηθεί ο marker και η σκιά του αντίστοιχα.

Επιπλέον, ο προγραμματιστής μπορεί να ορίσει αν ο marker είναι draggable μέσω του ομώνυμου options attribute, το οποίο προσφέρει πληθώρα μεθόδων για την τροποποίηση της συμπεριφοράς του. Επίσης, διαθέτει options για τη διαχείριση της αλληλεπίδρασης με τον κέρσορα, όπως το riseOnHover, αλλά και μεθόδους που κληρονομεί μέσω άλλων στοιχείων, όπως την click και την dblclick. Τέλος, προσφέρονται μέθοδοι για την τροποποίηση των options του, όπως οι getLatLang/ setLatLang για τη θέση του marker, οι getIcon/ setIcon για τη διαχείριση του icon και τα setZIndexOffset και setOpacity για τη θέση του στον κατακόρυφο άξονα και το επίπεδο διαφάνειας του. Στην Εικόνα 18 φαίνεται ένας marker στον χάρτη.



Εικόνα 18: OpenStreetMap Marker

## 4.4 Infowindow/Popup

Το infowindow/popup αποτελεί ένα dialog box, το οποίο προβάλλει περιεχόμενο, κείμενο ή/και εικόνες σε μορφή ενός αναδυόμενου παραθύρου. Η πιο συνηθισμένη και εύληπτη χρήση του, είναι η προσκόλληση του σε έναν marker και η προβολή του, όταν ο εν λόγω marker επιλεχθεί. Ωστόσο, μπορεί και να τοποθετηθεί σε οποιαδήποτε άλλη προκαθορισμένη θέση, ανεξάρτητα αν αυτή σχετίζεται με κάποιον marker.

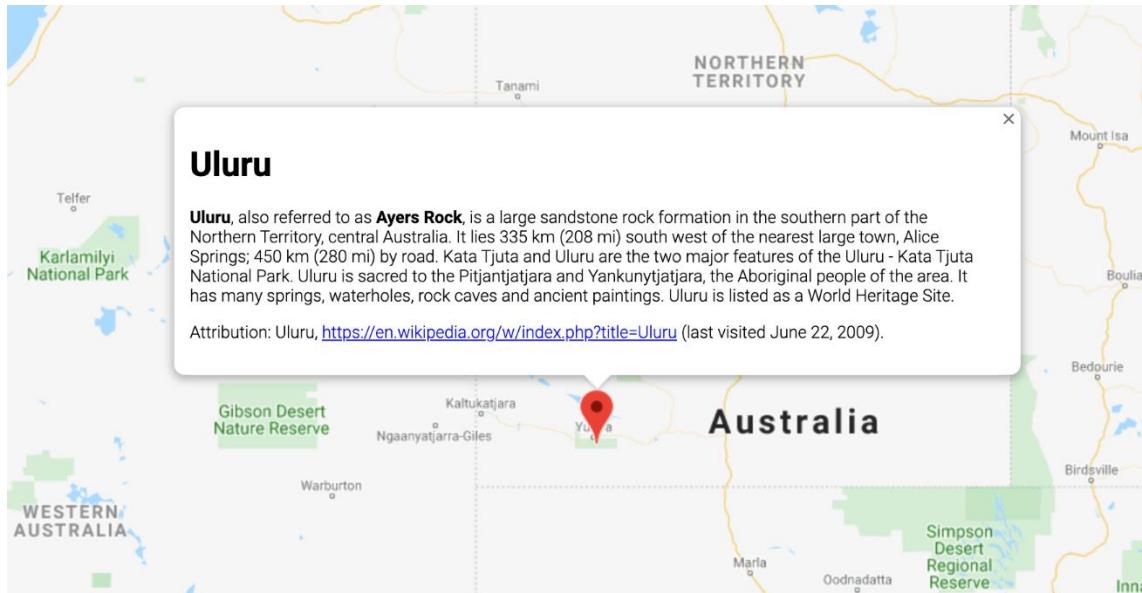
### 4.4.1 Google Maps - Infowindow

Οι παράμετροι που δέχεται ένα infowindow είναι οι:

- *Content*. Το περιεχόμενο του infowindow, το οποίο μπορεί να είναι ένα string ή ένα DOM element.
- *Position*. Καθορίζει τις συντεταγμένες για τη θέση του στον χάρτη.
- *PixelOffset*. Ορίζει την απόσταση του infowindow από το σημείο στο οποίο έχει “αγκιστρωθεί”, πχ την απόσταση του από έναν marker.

Εναλλακτικά, για την θέση του περιεχομένου, μπορεί να κληθεί η μέθοδος `setContent()` με την επιθυμητή παράμετρο. Ο ορισμός ενός infowindow δεν ταυτίζεται, όμως, με την εμφάνιση του στον χάρτη. Για να γίνει αυτό, πρέπει να κληθεί η μέθοδος `open()` του infowindow, η οποία με τη σειρά της δέχεται μια σειρά από options. Πιο συγκεκριμένα, πρέπει να οριστεί ο χάρτης με τον οποίο σχετίζεται, μέσω του `map`

αντικειμένου, το anchor, που ορίζει τον marker στον οποίο θα αγκιστρωθεί το infowindow και τέλος, το shouldFocus, σε περίπτωση που ο προγραμματιστής επιθυμεί να μετατοπιστεί την εστίαση του χάρτη στο σημείο που ανοίγει το infowindow. Στην Εικόνα 19 φαίνεται ένα infowindow όπως εμφανίζεται στον χάρτη. Το κλείσιμο του παραθύρου μπορεί να γίνει είτε μέσω του interface, κάνοντας κλικ στο “x” στην πάνω, δεξιά γωνία ή προγραμματιστικά με την close() μέθοδο.

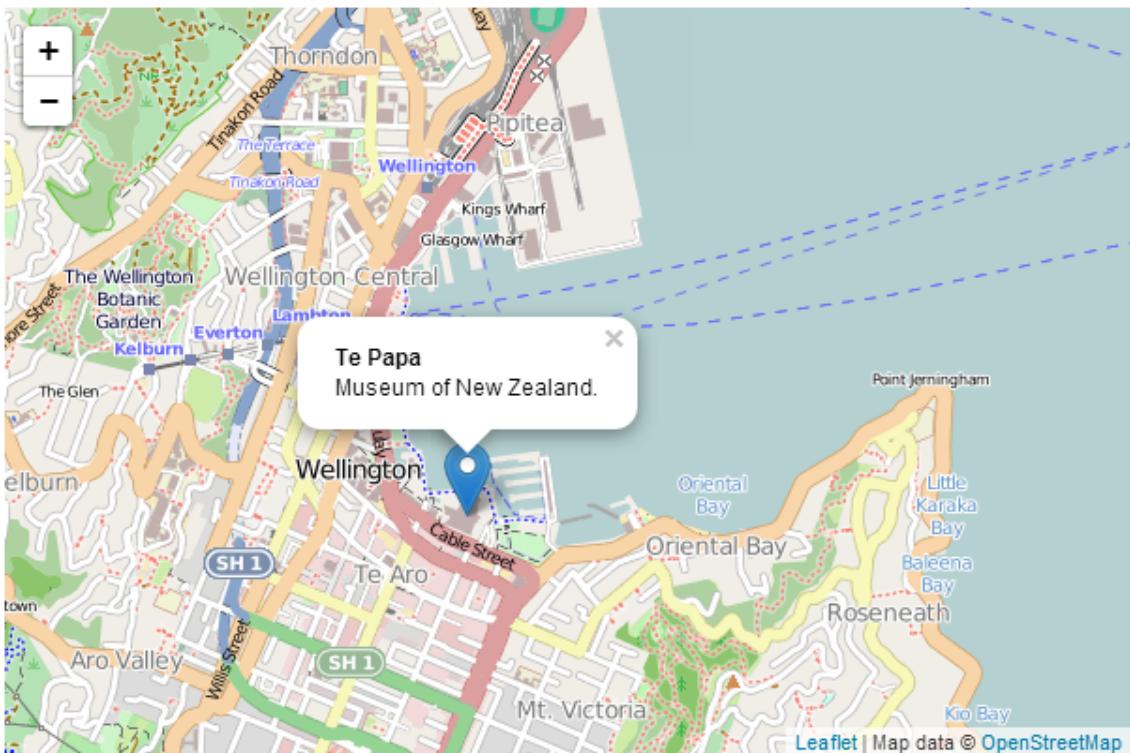


Εικόνα 19: Google Maps Infowindow

#### 4.4.2 OpenStreetMap - Popup

Για τη συσχέτιση ενός marker ή κάποιου άλλου overlay, όπως ένα polyline, η οποία είναι και η πιο συνηθισμένη πρακτική, απαιτείται η bindPopup() μέθοδος, η οποία λαμβάνει σαν παράμετρο το περιεχόμενο/κείμενο του popup. Ωστόσο, για τη δημιουργία ενός popup σε συγκεκριμένο σημείο του χάρτη, η διαδικασία είναι πιο σύνθετη. Αρχικά απαιτείται ένα popup αντικείμενο, το οποίο δέχεται, προαιρετικά, ως παραμέτρους, ένα options αντικείμενο και ένα layer αντικείμενο, για την παροχή αναφοράς στο layer στο οποίο ανήκει. Τα options αυτά έχουν να κάνουν με το max/min Width και Height του παραθύρου, το panning ή όχι του χάρτη στο σημείο που ανοίγει το popup, καθώς και την τελική θέση του παραθύρου σε σχέση με τον χάρτη, μετά το panning, αλλά και με τις επιλογές για το κλείσιμο του (όταν ανοίγει άλλο παράθυρο, όταν γίνει κλικ στον χάρτη ή με το πάτημα του Esc πλήκτρου). Έπειτα, δηλώνονται οι συντεταγμένες του μέσω της setLatLang() μεθόδου, το περιεχόμενο μέσω της setContent() και τέλος καλείται η openOn() μέθοδος, που δέχεται ως παράμετρο τον χάρτη.

Το περιεχόμενο του pop up μπορεί να είναι και HTML κώδικας, οποίος επιδέχεται μορφοποίησης μέσω CSS, αν οριστεί η αντίστοιχη κλάση μέσω του className attribute του options αντικειμένου. Για τη διαχείριση του pop up, υπάρχουν μέθοδοι που μεταβάλλουν τη θέση του στον Z άξονα του pane στο οποίο ανήκει, όπως οι bringToFront() και bringToBack() μέθοδοι για τη λήψη και θέση των συντεταγμένων, τη λήψη και θέση του περιεχομένου, αλλά και για να ελέγξουν αν είναι ανοιχτό ή όχι. Η Εικόνα 20 δείχνει ένα pop up όπως φαίνεται στον χάρτη.



Εικόνα 20: OpenStreetMap Popup

## 4.5 Σχήματα

Τα σχήματα αποτελούν αντικείμενα του χάρτη σε συγκεκριμένες συντεταγμένες. Τα σχήματα μπορεί να είναι ευθείες γραμμές, πολύγωνα, τετράγωνα και κύκλοι, ενώ έχουν πληθώρα λειτουργιών. Παραδείγματα ρεαλιστικών χρήσεων αποτελούν την διαγραφή μιας πορείας μεταξύ σημείων για την απεικόνιση της μεταξύ τους απόστασης, τη χρήση κύκλων για την γραφική προβολή πυκνότητας πληθυσμού ή την απεικόνιση της έκτασης μιας περιοχής με γραφικό τρόπο.

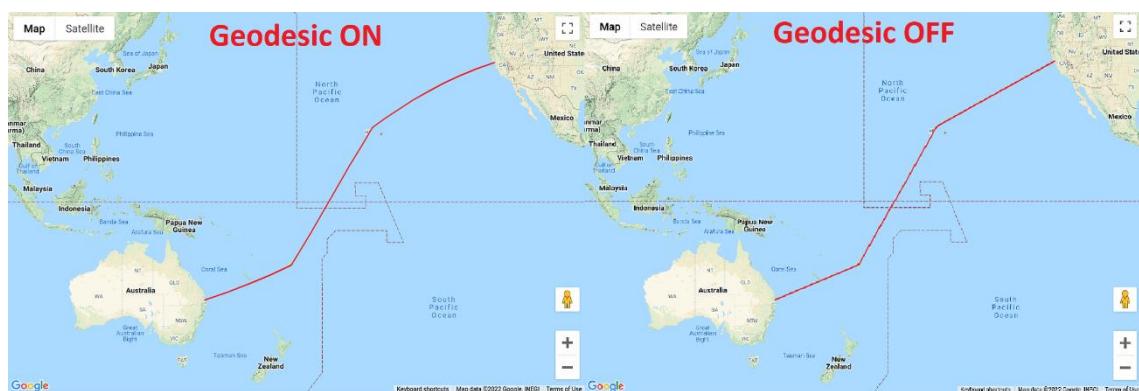
#### 4.5.1 Polylines – Google Maps

Τα Polylines χρησιμοποιούνται για τη σχεδίαση ευθειών γραμμών. Ένα αντικείμενο polyline αποτελείται από έναν πίνακα από geopoints, τα οποία συνδέει μεταξύ τους. Οι βασικότερες παράμετροι είναι οι:

- *Path*. Ο πίνακας των geopoints.
- *Geodesic*. Μια boolean παράμετρος που αποφασίζει για το αν το polyline πρέπει να σχεδιαστεί με geodesics, αντί για ευθείες γραμμές. Το geodesic είναι η συντομότερη διαδρομή μεταξύ δύο σημείων στην επιφάνεια της γης και λαμβάνει υπόψη το σφαιρικό της σχήμα. Στην Εικόνα 21 φαίνεται γραφικά η διαφορά των δύο περιπτώσεων.
- *StrokeColor*. Το χρώμα των γραμμών.
- *StrokeOpacity*. Η διαφάνεια των γραμμών.
- *StrokeWeight*. Το πάχος των γραμμών.

Επίσης, τα polylines, όπως και όλα τα σχήματα, μπορούν να τεθούν ως *editable*, ώστε να μπορούν να επιδεχθούν αλλαγές από τον χρήστη, και *draggable*, για να μπορούν να μετακινηθούν σε νέα θέση.

Για την τοποθέτηση τους στον χάρτη, πρέπει απλά να συσχετιστούν με αυτόν, μέσω της μεθόδου *setMap()* του αντικειμένου, ενώ για την απομάκρυνσή τους, τίθεται σε *null* παράμετρος της μεθόδου. Μέσω της μεθόδου *getPath()*, ο προγραμματιστής μπορεί να αποκτήσει πρόσβαση στον πίνακα με τα geopoints, που καθορίζουν τη διαδρομή του polyline, και έχει τη δυνατότητα να προσθέσει και να αφαιρέσει τα σημεία που επιθυμεί.



Εικόνα 21: Σύγκριση των επιλογών geodesic ενός polyline στο Google Maps

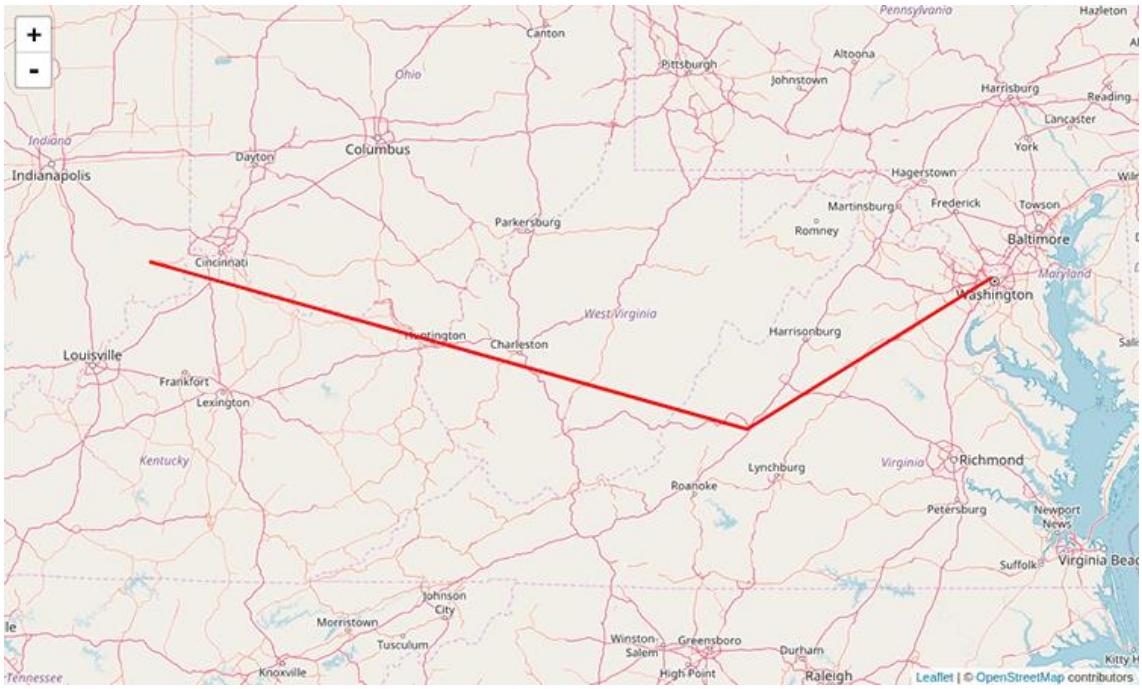
#### 4.5.2 Polylines – OpenStreetMap

Ένα polyline αντικείμενο, δέχεται ως παράμετρο έναν πίνακα από latLng αντικείμενα και ένα options αντικείμενο. Ο πίνακας αυτός, μπορεί να είναι πολυδιάστατος και το

αποτέλεσμα είναι ένα MultiPolyline Shape, δηλαδή πολλά, ξεχωριστά polylines ως ένα σχήμα. Για την προσθήκη του στον χάρτη, καλείται η addTo() μέθοδος του αντικειμένου, με παράμετρο τον χάρτη. Στην Εικόνα 22 απεικονίζεται ένα polyline, όπως φαίνεται στον χάρτη.

Τα polylines, όπως και όλα τα σχήματα στην leaflet.js, κληρονομούν μεθόδους και attributes από τις υπερκλάσεις Path, και κατ' επέκταση Layer. Έτσι, όλα τα σχήματα μοιράζονται κάποια κοινά attributes. Τα attributes αυτά έχουν να κάνουν με την εμφάνιση των σχημάτων, όπως το color για το χρώμα των γραμμών, το weight για το πάχος τους, το opacity για τη διαφάνεια, τα lineCap και lineJoin για σχήμα των γραμμών στις άκρες και στις γωνίες αντίστοιχα, τα dashArray και dashOffset για την εμφάνιση των γραμμών ως διακεκομμένες και την απόσταση των εν λόγω τμημάτων. Τα options attributes του polyline, αυτού καθαυτού, είναι δύο. Το smoothFactor, το οποίο ορίζει το πόσο θα απλοποιείται το σχήμα ανάλογα με το επίπεδο zoom, όπου όσο μεγαλύτερη τιμή τίθεται, τόσο λιγότερη λεπτομέρεια αναδεικνύεται και κατ' επέκταση, τόσο καλύτερο performance επιτυγχάνεται. Τέλος το noClip, για την αποφυγή clipping, την αλληλοεπικάλυψη, δηλαδή, των γραμμών.

Οι μέθοδοι που διαθέτει επιτρέπουν την επεξεργασία των geopoints που απαρτίζουν το polyline με τις get/set/addLatLang μεθόδους, τη λήψη του κέντρου του, για την ακρίβεια, του centroid του (γεωμετρικό κέντρο) με την getCenter(), αλλά και την επιλογή του κοντινότερου σημείου στο polyline από ένα Point που ορίζει ο χρήστης. Τέλος, είναι δυνατή η εξαγωγή του polyline ως GeoJSON αρχείο μέσω της μεθόδου toGeoJSON(), η οποία δέχεται ως παράμετρο την ακρίβεια των δεκαδικών στις συντεταγμένες του polyline.



Εικόνα 22: OpenStreetMap Polyline

### 4.5.3 Polygons – Google Maps

Τα polygons μοιάζουν με τα polylines στο γεγονός ότι αναπαριστούν μια διαδρομή μεταξύ σημείων. Η κύρια διαφορά είναι πως, στα polygons, η διαδρομή είναι κλειστή, ενώνεται, δηλαδή, ο τελευταίος με τον πρώτο κόμβο. Πιο συγκεκριμένα, οι παράμετροι που μπορεί να πάρει ένα polygon αντικείμενο είναι οι εξής:

- *Path*. Ο πίνακας των geopoints.
- *StrokeColor*. Το χρώμα των γραμμών.
- *StrokeOpacity*. Η διαφάνεια των γραμμών.
- *StrokeWeight*. Το πάχος των γραμμών.
- *FillColor*. Το χρώμα γεμίσματος ή φόντου του σχήματος.
- *FillOpacity*. Η διαφάνεια του γεμίσματος του σχήματος.

Η εμφάνιση στον χάρτη γίνεται, όπως και σε όλα τα σχήματα, μέσω της `setMap()` μεθόδου και η διαγραφή του από αυτόν, με το πέρασμα της παραμέτρου `null`. Επίσης, μπορούν, κατ' επιλογή, να τεθούν οι ιδιότητες `draggable` και `editable`. Για την δημιουργία μια τρύπας στο εσωτερικό του σχήματος, όπως φαίνεται στην Εικόνα 23, πρέπει το όρισμα `path` να αποτελεί έναν πίνακα από πίνακες με geopoints, με τον πρώτο να προσδιορίζει τις εξωτερικές συντεταγμένες, και τον δεύτερο τις εσωτερικές. Τέλος, για την πρόσβαση του πίνακα των geopoints, χρησιμοποιείται η μέθοδος `getPath()`.

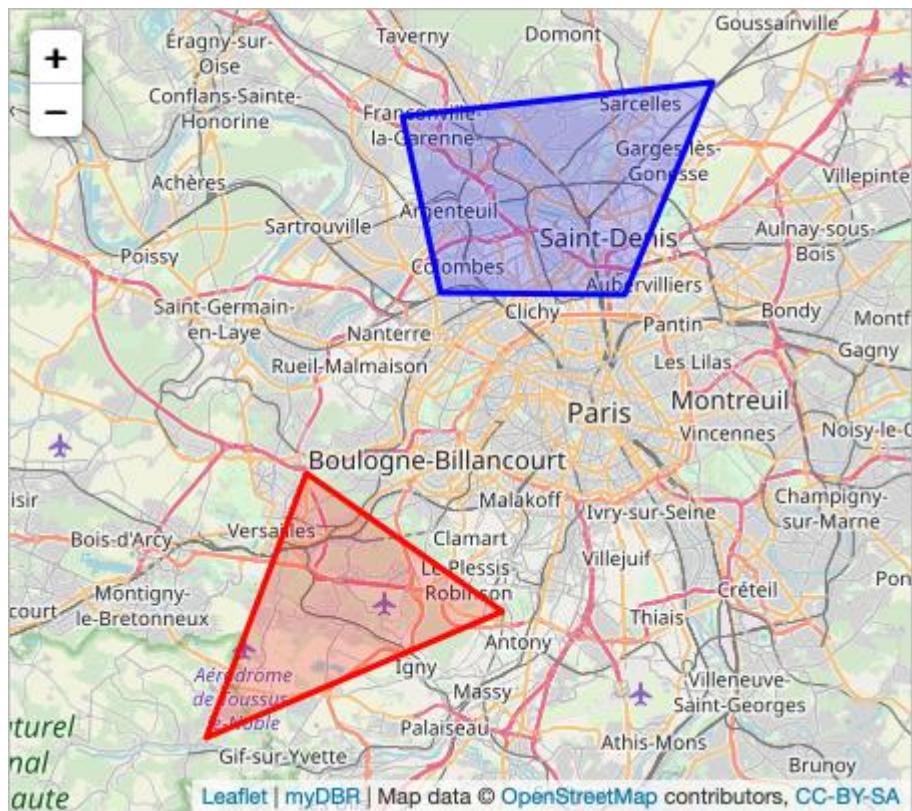


Εικόνα 23: Google Maps Polygon με μια τρύπα στο εσωτερικό του

#### 4.5.4 Polygons – OpenStreetMap

Τα polygons, όπως και τα polylines, δέχονται ως παράμετρο ένα options αντικείμενο και έναν πίνακα με LatLng σημεία. Το τελικό σχήμα δημιουργείται με τη διαδοχική ένωση των σημείων αυτών, με το τελευταίο συνδέεται με το πρώτο, ενώ δεν ενδείκνυται η επανάληψη του πρώτου στοιχείου στο τέλος του πίνακα. Επίσης, υποστηρίζεται η δημιουργία τρυπών στο εσωτερικό του σχήματος, με το πέρασμα ως παράμετρο ενός πίνακα πινάκων, με τον πρώτο εμφωλευμένο πίνακα να αντιστοιχεί στο εξωτερικό σχήμα και τους υπόλοιπους στα εσωτερικά. Επιπλέον, υποστηρίζεται και η δημιουργία MultiPolygon shape, πολλών polygons ως ένα σχήμα, με το πέρασμα ενός πολυδιάστατου πίνακα. Παράδειγμα αποτελεί η Εικόνα 24.

Οσον αφορά τα options που μπορεί να δεχτεί, όπως προαναφέρθηκε, είναι κοινά με τα υπόλοιπα σχήματα. Είναι δυνατή, δηλαδή, η μορφοποίηση του χρώματος, του σχήματος, της μορφής και άλλων ιδιαίτερων χαρακτηριστικών των γραμμών του σχήματος. Τέλος, οι μέθοδοι που είναι διαθέσιμοι, είναι ακριβώς οι ίδιες.

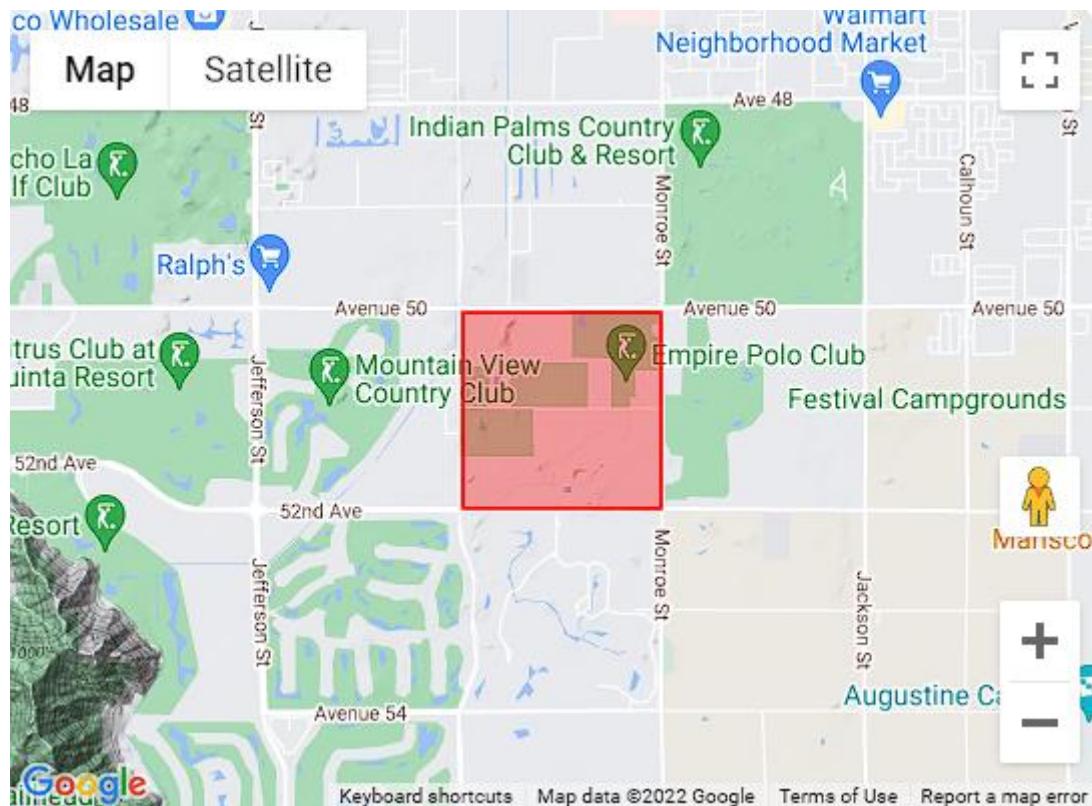


Εικόνα 24: Polygons στο OpenStreetMap

#### 4.5.5 Rectangles – Google Maps

Τα rectangles είναι ουσιαστικά polygons, ωστόσο προσφέρεται ξεχωριστή κλάση για τη δημιουργία τους. Η βασική διαφορά με τα polygons είναι πως δεν δέχονται σαν όρισμα έναν πίνακα για το path, αλλά έχουν μια ξεχωριστή παράμετρο, την bounds, η οποία αποτελεί ένα αντικείμενο με τέσσερις παραμέτρους, north, south, east, west, για τις θέσεις του κάθε σημείου. Πιο συγκεκριμένα, το τεράγωνο σχεδιάζεται με βάση τον συνδυασμό των south και west σημείων σε σχέση με τον συνδυασμό των north και east.

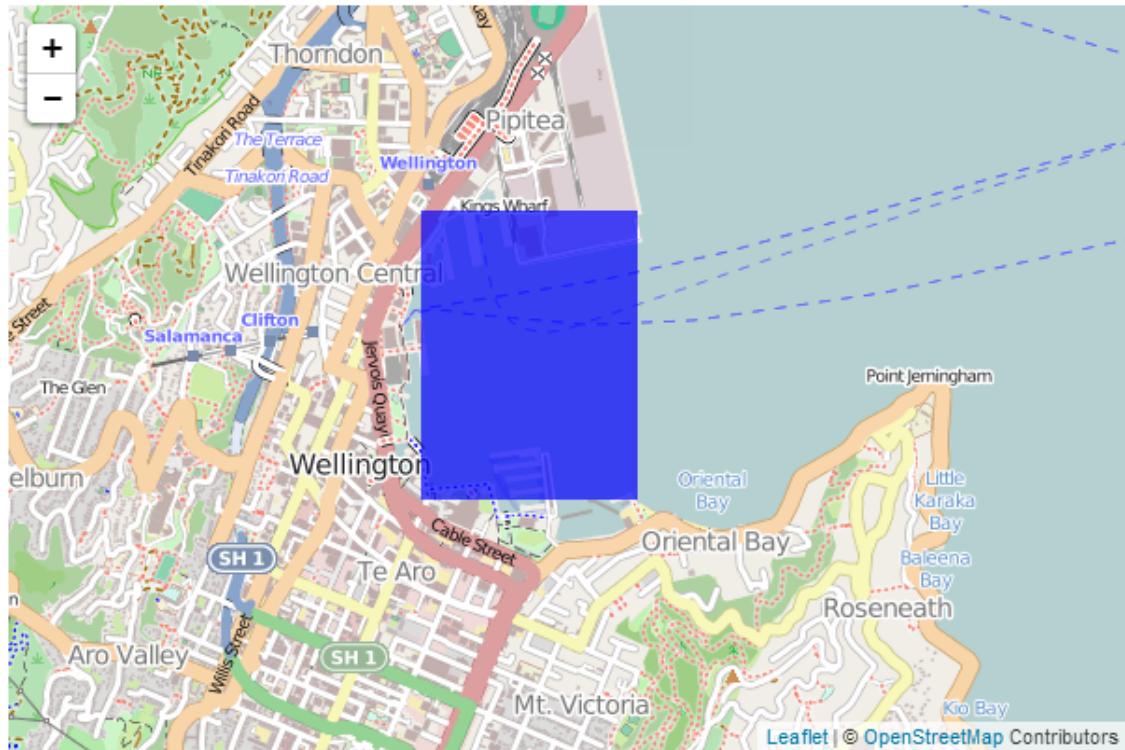
Τα rectangles διαθέτουν όλα τα υπόλοιπα ορίσματα όπως τα strokeColor, strokeOpacity, strokeWeight, fillColor και fillOpacity, ενώ μπορούν να τεθούν και ως draggable και editable. Η τοποθέτησή τους στον χάρτη γίνεται είτε απευθείας κατά τη δημιουργία τους μέσω του map ορίσματος ή μέσω της setMap() μεθόδου. Η Εικόνα 25 δείχνει ένα rectangle.



Εικόνα 25: Rectangle σε Google Maps

#### 4.5.6 Rectangles – OpenStreetMap

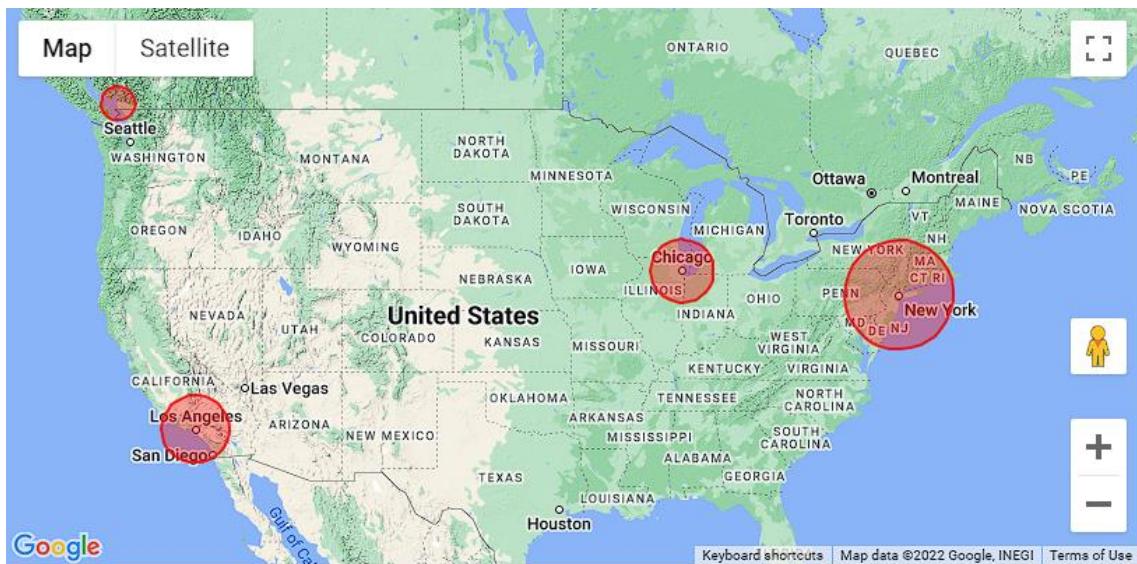
Τα rectangles εμφανίζουν μια μικρή διαφοροποίηση ως προς την δημιουργία τους και την σύνθεση τους. Αντί για ένα LatLng αντικείμενο για την παροχή σημείων, δέχονται ένα LatLngBounds αντικείμενο, το οποίο αντιπροσωπεύει μια ορθογώνια παραλληλόγραμμη περιοχή. Το αντικείμενο αυτό αποτελείται από δύο LatLng αντικείμενα με το καθένα να έχει τις συντεταγμένες για τον βορρά και τη δύση, και την ανατολή και τον νότο αντίστοιχα. Οι συντεταγμένες αυτές, μπορούν να παρασχεθούν και ως απλοί, εμφωλευμένοι πίνακες, απευθείας στο αντικείμενο rectangle. Αυτή είναι και η μοναδική διαφορά που έχουν από τα υπόλοιπα σχήματα. Η δομή και οι επιλογές του options αντικειμένου, καθώς και οι μέθοδοι που διαθέτει, παραμένουν ίδια. Παράδειγμα ενός rectangle, μπορεί να φανεί στην Εικόνα 26.



Εικόνα 26: Rectangle σε OpenStreetMap

#### 4.5.7 Circles – Google Maps

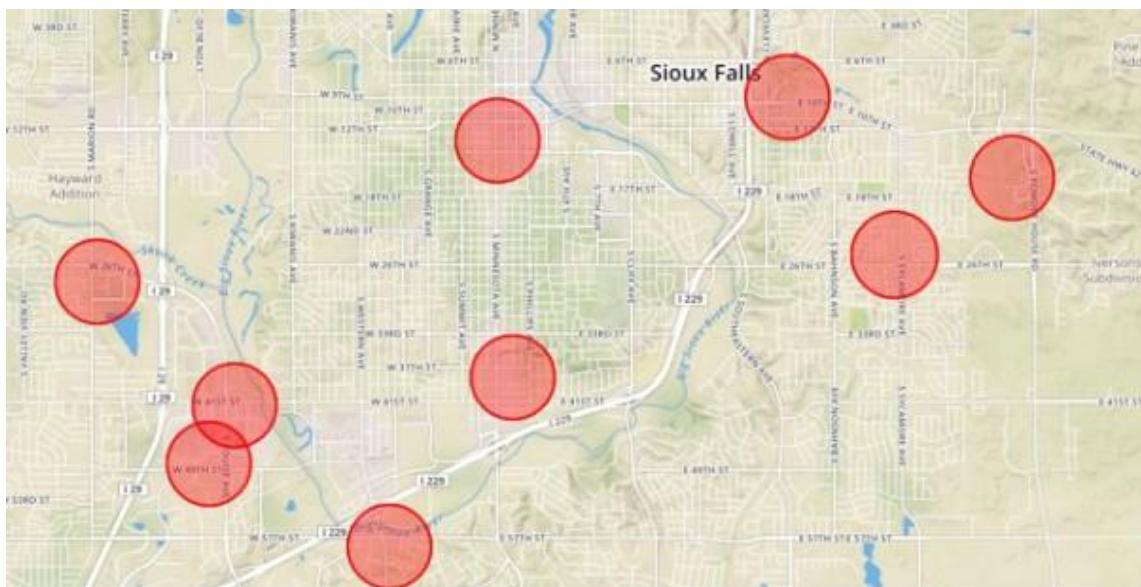
Τα circles, όπως και τα rectangles, παρόλου που είναι κατά βάση polygons, έχουν τη δική τους ξεχωριστή κλάση. Δεν διαθέτουν path ή bounds ορίσματα. Αντιθέτως, έχουν δύο ξεχωριστά, το center, που καθορίζει το κέντρο το κύκλου και είναι ένα geopoint, και το radius, το οποίο καθορίζει την ακτίνα του σε μέτρα. Αν εξαιρεθούν αυτές οι διαφορές, μοιράζονται όλα τα υπόλοιπα ορίσματα με τα λοιπά σχήματα, διαθέτουν δηλαδή strokeColor, strokeOpacity, strokeWeight, fillColor και fillOpacity. Μπορούν τεθούν ως editable και draggable, ενώ συσχετίζονται με κάποιον χάρτη, είτε κατά την αρχικοποίηση τους ή μέσω της setMap() μεθόδου. Η Εικόνα 27 δείχνει τέσσερις κύκλους διαφορετικών εμβαδών.



Εικόνα 27: Circles σε Google Maps

#### 4.5.8 Circles - OpenStreetMap

Τα circles διαφοροποιούνται από τα υπόλοιπα σχήματα στο ότι διαθέτουν ελαφρώς διαφορετική δομή. Για τη δημιουργία ενός αντικειμένου απαιτείται ένα latLng αντικείμενο ή, εναλλακτικά, ένας πίνακας δύο στοιχείων, που αντιπροσωπεύει το κέντρο του κύκλου. Επιπλέον, το options δέχονται ένα επιπλέον όρισμα, το radius, για τον προσδιορισμό της ακτίνας του κύκλου. Τα υπόλοιπα ορίσματα είναι κοινά με τα λοιπά σχήματα. Διαθέτει, επίσης, δύο ακόμα μεθόδους, την getRadius() και την setRadius(), για την λήψη και θέση της ακτίνας του. Η παρακάτω Εικόνα 28, δείχνει ένα πλήθος κύκλων.



Εικόνα 28: Circles σε OpenStreetMap

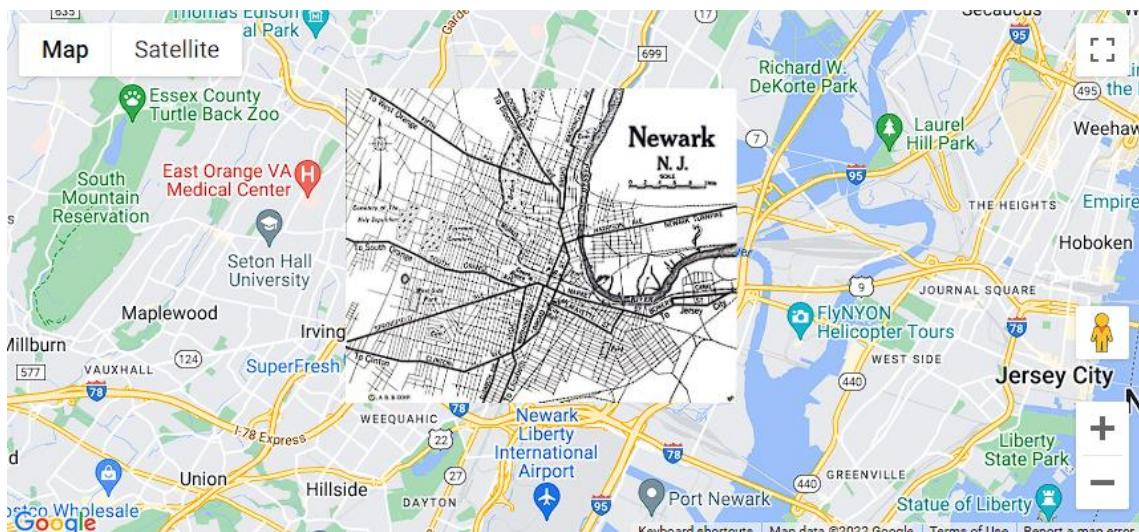
## 4.6 Overlays/ Layers και Layers/ LayerGroups

Οι όροι overlay, layer και layergroup χρησιμοποιούνται για να περιγράψουν δύο διαφορετικά χαρακτηριστικά που μοιράζονται και οι δύο πλατφόρμες. Οι ονομασίες μπορούν να επιφέρουν σύγχυση, καθώς όμοια χαρακτηριστικά περιγράφονται από όρους που περιγράφουν κάτι άλλο στην απέναντι πλατφόρμα. Η πιο ακριβής αντιστοίχιση θα ήταν μεταξύ των overlays του google maps και των layers του OpenStreetMap, και συγκεκριμένα του leaflet.js, και των layers του google maps με τα layerGroups του OpenStreetMap.

### 4.6.1 Google Maps Overlays και Layers

Ένα overlay στο google maps είναι ό, τι μπορεί να προστεθεί στον χάρτη, από έναν marker μέχρι ένα σχήμα. Λοιπά overlays, που δεν αναφέρθηκαν είναι τα εξής:

- *Symbols*. Αποτελεί ένα vector-based εικονίδιο το οποίο μπορεί να χρησιμοποιηθεί ως εικόνα για έναν marker ή να προβληθεί σε ένα polyline.
- *Ground Overlay*. Χρησιμοποιείται για την τοποθέτηση εικόνων στον χάρτη. Σχετικό παράδειγμα φαίνεται στην Εικόνα 29.
- *Custom Overlay*. Χρησιμοποιείται για τη δημιουργία custom overlay.



Εικόνα 29: Ground Overlay μιας εικόνας σε Google Maps

Τα Layers είναι αντικείμενα τα οποία ομαδοποιούν τα διάφορα στοιχεία του χάρτη, ώστε να μπορούν να τα διαχειριστούν ενιαία. Αποτελούν, δηλαδή, συλλογές αντικειμένων που προστίθενται στον χάρτη και, συνήθως, προσδιορίζονται από κάποια κοινή σχέση. Για την παρουσίαση των αντικειμένων των layers, το API τα κάνει render

ως ένα αντικείμενο, συνήθως ως ένα tile overlay, και τα προβάλλει στον εκάστοτε διαθέσιμο χώρο του χάρτη. Τα Layers που διατίθενται είναι τα εξής:

- *Data Layer*. Αποτελεί ένα container για γεωγραφικά δεδομένα. Αυτό σημαίνει πως υποστηρίζει την προβολή custom δεδομένων ή/και GeoJSON δεδομένων, τόσο custom όσο και από τρίτους. Το API της google, επιτρέπει την τοποθέτηση πληθώρας overlays, όπως markers, polylines κ.α., τα οποία συνδυάζουν γεωγραφικά δεδομένα με δεδομένα για την μορφοποίηση του στυλ τους. Αντί, λοιπόν, να τοποθετηθούν αυτά τα overlays, μπορούν να εισαχθούν τα αντίστοιχα δεδομένα στο Data Layer μέσω κάποιου GeoJSON αρχείου, και το API, αφού το κάνει parse, θα μεταφράσει τις δομές του σε markers, polylines, polygons ή σε οτιδήποτε άλλο μπορεί να ορίσει ένα GeoJSON αρχείο. Για την φόρτωση ένος GeoJSON αρχείου στο Data Layer ενός χάρτη, χρησιμοποιείται η `loadGeoJson()` μέθοδος, η οποία δέχεται ως παράμετρο το URL του αρχείου, ενώ η πρόσβαση στο Data Layer γίνεται μέσω του data attribute που διαθέτει ο χάρτης. Επίσης, είναι δυνατή η μορφοποίηση των εισαχθέντων δεδομένων μέσω της `setStyle()` μεθόδου του data attribute του χάρτη, η οποία δέχεται είτε ένα StyleOptions αντικείμενο είτε ένα function, το οποίο υπολογίζει και ορίζει τη μορφοποίηση για κάθε στοιχείο του GeoJSON αρχείου.
- *Heatmap Layer*. Χρησιμοποιείται στην παρουσίαση δεδομένων μέσω Heatmap απεικόνισης. Τα heatmaps χρησιμοποιούνται συνήθως για απεικόνιση πληροφορίας σχετική με πυκνότητα ή συχνότητα γεγονότων. Απαιτείται χρήση της βιβλιοθήκης visualization.
- *KML Layer*. Χρησιμοποιείται για την απεικόνιση Keyhole Markup Language και GeoRSS στοιχείων στον χάρτη.
- *Traffic Layer*. Χρησιμοποιείται για την προβολή στοιχείων σχετικά με την κατάσταση της κίνησης.
- *Transit Layer*. Χρησιμοποιείται για την προβολή του δικτύου δημοσίων συγκοινωνιών στον χάρτη.
- *Bicycling Layer*: Χρησιμοποιείται συγκεκριμένα για την προβολή ποδηλατοδρόμων και στοιχείων σχετικά με την μετακίνηση με ποδήλατο.

#### 4.6.2 OpenStreetMap Layers και LayerGroups

Ως layer στο leaflet.js ορίζεται οτιδήποτε μπορεί να προστεθεί στον χάρτη. Έτσι, οι markers, τα popups και τα σχήματα, όλα αποτελούν layers. Τα layers μπορούν να ομαδοποιηθούν σε LayerGroups, με οποιαδήποτε κριτήρια αποφασίσει ο προγραμματιστής. Ως αποτέλεσμα, ένα LayerGroup θα μπορούσε να περιέχει όλους του markers ή όλα τα layers που σχετίζονται με κάποια συγκεκριμένη λειτουργία. Τα LayerGroups διαθέτουν μεθόδους για την προσθήκη και αφαίρεση layer, για τον έλεγχο ύπαρξης ενός layer στο συγκεκριμένο group καθώς και την εύκολη ανάκτηση τους. Αποτελούν εξαιρετικά modular οντότητες που προσφέρουν τεράστια ευελιξία στη διαχείριση και οργάνωση των στοιχείων ενός χάρτη.

To leaflet υποστηρίζει και την εισαγωγή GeoJSON δεδομένων μέσω της geoJSON μεθόδου, η οποία δημιουργεί ένα layerGroup με τα στοιχεία που γίνονται parse από την GeoJSON δομή. Η μέθοδος δέχεται ένα geoJson αντικείμενο για τον προσδιορισμό των δεδομένων και ένα αντικείμενο options για την παραμετροποίηση του. Το options αντικείμενο μπορεί, μεταξύ άλλων, να προσδιορίσει τη μορφοποίηση των GeoJSON στοιχείων αλλά και να καθορίσει ποια από αυτά θα καταλήξουν στον χάρτη, μέσω μηχανισμών φιλτραρίσματος.

# 5 Εφαρμογή επίδειξης

Στο κεφάλαιο αυτό, θα παρουσιαστεί η εφαρμογή επίδειξης της παρούσας πτυχιακής εργασίας. Αυτή η εφαρμογή αποτελεί ένα web application επίδειξης των προγραμματιστικών χαρακτηριστικών των δυο ειδών χαρτών που παρουσιάστηκαν και αναλύθηκαν στο 4<sup>o</sup> κεφάλαιο. Ταυτόχρονα με την επίδειξη των χαρακτηριστικών αυτών, η εφαρμογή αναδεικνύει, για κάθε περίπτωση, το κομμάτι κώδικα των δυο βιβλιοθηκών χαρτών, που εξυπηρετεί κάθε περίπτωση χαρακτηριστικού. Παρακάτω, ακολουθούν υποενότητες τόσο για την γραφική παρουσίαση της εφαρμογής όσο και για την ανάλυση του κώδικα, συγκριτικά, για τα όμοια χαρακτηριστικό των δυο χαρτών.

## 5.1 Γραφικό περιβάλλον εφαρμογής

Το γραφικό περιβάλλον της εφαρμογής έχει σχεδιαστεί με τέτοιο τρόπο, ώστε ο χρήστης να μπορεί να περιηγηθεί ανάμεσα στα κοινά χαρακτηριστικά των δυο βιβλιοθηκών χαρτών.

Η παρούσα εφαρμογή έχει υλοποιηθεί με την γλώσσα προγραμματισμού JAVASCRIPT και με την υποβοήθηση των HTML (hypertext markup language) και CSS (Cascading Style Sheets).

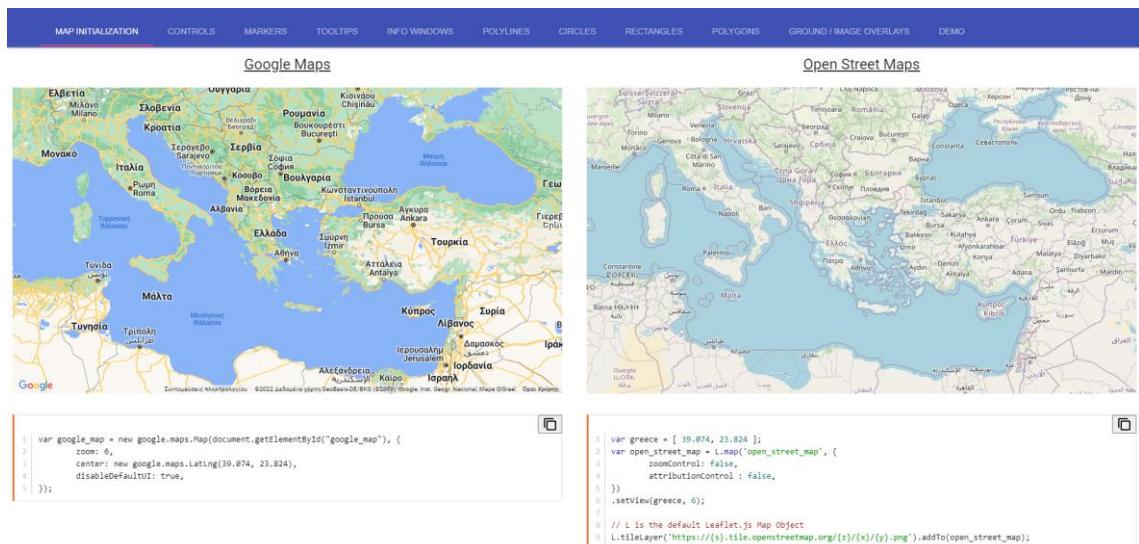
Η δυνατότητα περιήγησης είναι εφικτή μέσω ενός κεντρικού μενού μπάρας, όπου αναγράφονται, ένα προς ένα, όλα τα στοιχεία. Κάνοντας κλικ πάνω σε ένα στοιχείο του μενού, ο χρήστης ανακατευθύνεται στην αντίστοιχη σελίδα. Το μενού μπάρα της εφαρμογής παρουσιάζεται στην Εικόνα 30.



Εικόνα 30: Μενού μπάρα χαρακτηριστικών χαρτών

Κάθε σελίδα/πτέρυγα χαρακτηριστικού αποτελείται από δύο instances χαρτών, ενός Google Maps και ενός Open Street Maps. Οι δύο αυτοί χάρτες τοποθετούνται ο ένας δίπλα στον άλλο και παρουσιάζουν κάθε φορά ένα χαρακτηριστικό, όπως αυτό εξυπηρετείται από την εκάστοτε βιβλιοθήκη.

Παράλληλα, ακριβώς κάτω από τα δύο instances χαρτών, παρουσιάζονται δύο πλαίσια κώδικα, ένα για τον Google Maps και ένα για τον Open Street Maps χάρτη. Στη κάθε περίπτωση, τα δύο αυτά πλαίσια αναδεικνύουν τον κώδικα που παρέχεται από τη κάθε βιβλιοθήκη και που χρειάζεται ένας προγραμματιστής, ώστε να υλοποιήσει το χαρακτηριστικό προς παρουσίαση. Ταυτόχρονα, όπως φαίνεται στην Εικόνα 32: Αντιγραφή πλαισίου κώδικα., δίνεται η δυνατότητα αντιγραφής αυτού του πλαισίου για επαναχρησιμοποίηση. Το κεντρικό σώμα της εφαρμογής με τα πλαίσια χαρτών και κώδικα, παρουσιάζεται στην Εικόνα 31: Σώμα εφαρμογής..



Εικόνα 31: Σώμα εφαρμογής.

Τέλος ύστερα από την παρουσίαση όλων των κοινών χαρακτηριστικών, υπάρχει μια τελική καρτέλα, με τίτλο «*DEMO*», η οποία δίνει τη δυνατότητα στο χρήστη να πειραματιστεί με τους χάρτες και να δοκιμάσει όλα τα χαρακτηριστικά που υπάρχουν στις προηγούμενες καρτέλες, ταυτόχρονα και με κάποια επιπλέον στοιχεία.



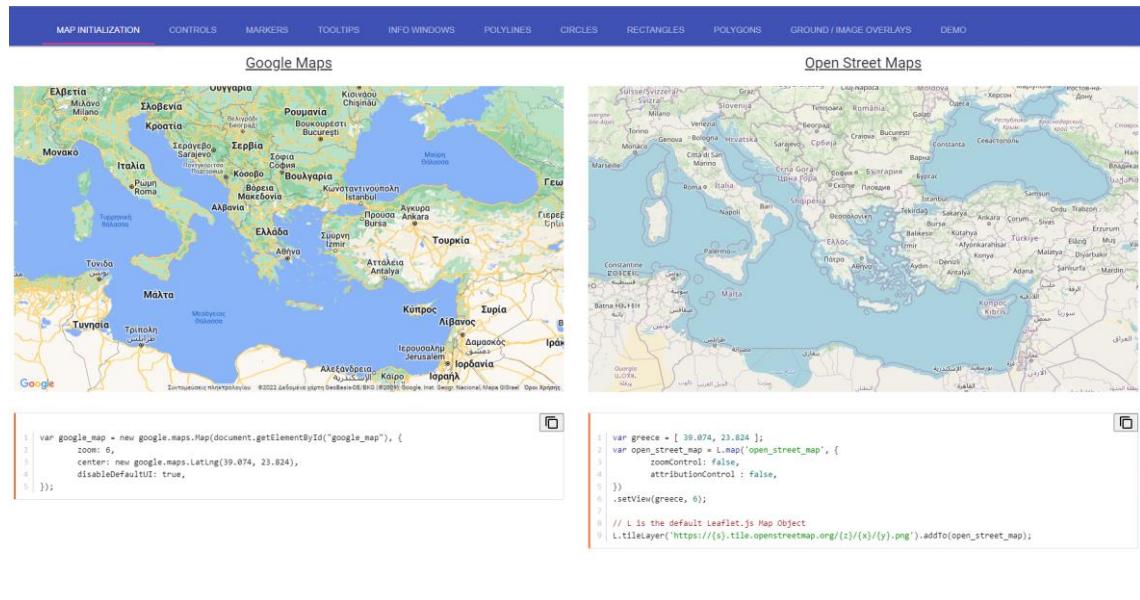
Εικόνα 32: Αντιγραφή πλαισίου κώδικα.

Σε αυτό το σημείο, αξίζει να αναφερθεί πως στόχος της εφαρμογής, είναι η χρήση της από άλλους προγραμματιστές και η αξιοποίησή της ως ένα documentation κοινών χαρακτηριστικών των δυο χαρτών, με τελικό σκοπό την υποβοήθηση εκμάθησης των βιβλιοθηκών που τα υποστηρίζουν.

## 5.2 Αρχικοποίηση Χάρτη

Η πρώτη επιλογή του μενού μπάρας αποτελεί αύτη της αρχικοποίησης χάρτη. Αυτό το βήμα είναι ίσως και το πιο σημαντικό από τα υπόλοιπα χαρακτηριστικά που αναλύθηκαν, γιατί χωρίς την αρχικοποίηση του στοιχείου του χάρτη σε μια HTML σελίδα, δεν μπορούν να εξυπηρετηθούν όλες οι υπόλοιπες δυνατότητες που προσφέρουν οι βιβλιοθήκες.

Η σελίδα της αρχικοποίησης των δυο χαρτών φαίνεται στην Εικόνα 33: Αρχικοποίηση Χαρτών.



Εικόνα 33: Αρχικοποίηση Χαρτών

Η παραπάνω εικόνα απεικονίζει την αρχικοποίηση των δυο χαρτών χωρίς την ύπαρξη στοιχείων ελέγχου/controls στα πλαίσια τους. Και οι δυο χάρτες είναι κεντραρισμένοι στις συντεταγμένες  $[ 39.074, 23.824 ]$ , ακριβώς πάνω από την Ελλάδα, με επίπεδο zoom 6. Όλα τα υπόλοιπα παραδείγματα χαρακτηριστικών που ακολουθούν, θα αρχικοποιούν τους χάρτες με βάση τα παραπάνω δύο στοιχεία. Η απόκρυψη των controls είναι διαχειρίσιμη μέσω του αντικειμένου options κατά την αρχικοποίηση τους.

### 5.2.1 Αρχικοποίηση Χάρτη Google Maps

Πιο συγκεκριμένα, η αρχικοποίηση του Google Maps χάρτη πραγματοποιείται με το πλαίσιο κώδικα που φαίνεται στην Εικόνα 34: Πλαίσιο κώδικα αρχικοποίησης χάρτη Google Maps, χωρίς στοιχεία ελέγχου..

```
1 var google_map = new google.maps.Map(document.getElementById("google_map"), {  
2     zoom: 6,  
3     center: new google.maps.LatLng(39.074, 23.824),  
4     disableDefaultUI: true,  
5 });
```

Εικόνα 34: Πλαίσιο κώδικα αρχικοποίησης χάρτη Google Maps, χωρίς στοιχεία ελέγχου.

Ουσιαστικά, αρχικοποιείται ένα instance αντικειμένου Google Maps, πάντα με την έκφραση `new google.maps.Map()`. Μέσα στην παρένθεση, ο constructor της κλάσης περιμένει ένα DIV HTML element, το οποίο το αποκτά με τη javascript μέθοδο `document.getElementById("google_map")`. Επιπλέον, προαιρετικά περιμένει και ένα JSON αντικείμενο options, το οποίο, αν δεν προστεθεί, ο χάρτης αρχικοποιείται με κάποιες default τιμές που έχουν οριστεί από τη βιβλιοθήκη. Στη περίπτωση του παραδείγματος παραπάνω, το αντικείμενο options παίρνει τιμές για επίπεδο `zoom` 6, `center` ένα αντικείμενο `google.maps.LatLng()` με κέντρο την Ελλάδα και `disableDefaultUI` true, ώστε να αποκρύψει τα βασικά στοιχεία ελέγχου του χάρτη.

### 5.2.2 Αρχικοποίηση Χάρτη Open Street Maps

Η αρχικοποίηση του χάρτη Open Street Maps, πραγματοποιείται με το πλαίσιο κώδικα που παρουσιάζεται στην Εικόνα 35.

```
1 var greece = [ 39.074, 23.824 ];  
2 var open_street_map = L.map('open_street_map', {  
3     zoomControl: false,  
4     attributionControl : false,  
5 })  
6 .setView(greece, 6);  
7  
8 // L is the default Leaflet.js Map Object  
9 L.tileLayer('https://s.tile.openstreetmap.org/{z}/{x}/{y}.png').addTo(open_street_map);
```

Εικόνα 35: Πλαίσιο κώδικα αρχικοποίησης χάρτη Open Street Maps, χωρίς στοιχεία ελέγχου.

Σε αυτήν την περίπτωση, αρχικοποιείται ένα αντικείμενο `open_street_map`. Αυτό γίνεται καλώντας το κεντρικό αντικείμενο αναφοράς της βιβλιοθήκης, το `L`. Έτσι καλείται η μέθοδος του αντικειμένου `L`, η `map()`, η οποία περιμένει ως παραμέτρους ένα στοιχείο HTML DIV με `id` `open_street_map` και ένα JSON αντικείμενο `options`. Στην

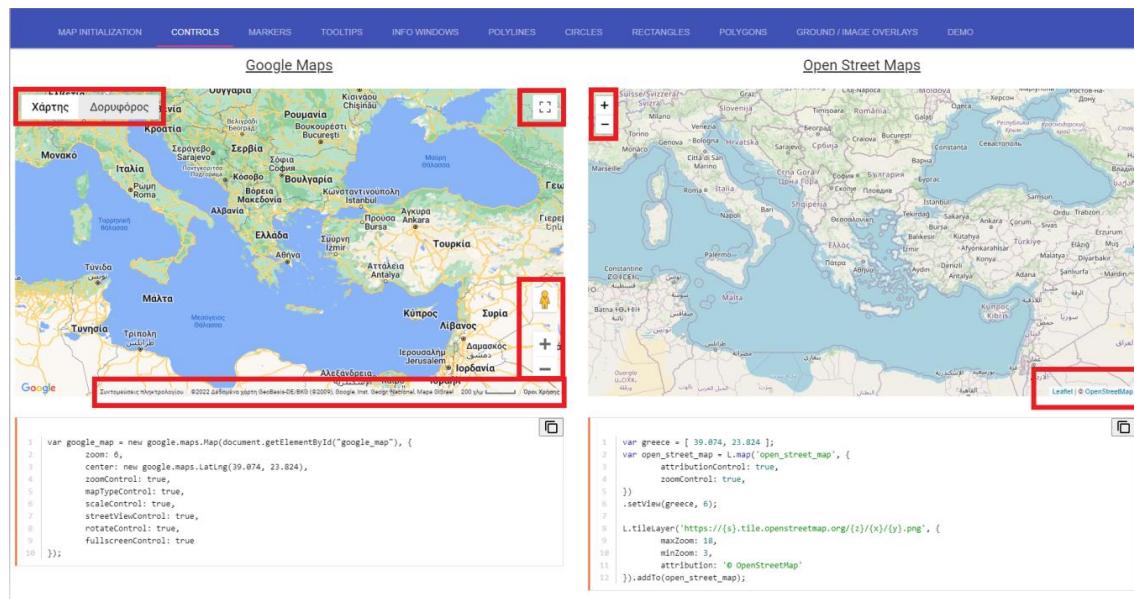
περίπτωση του παραδείγματος, το αντικείμενο αυτό, περιλαμβάνει δύο τιμές για τη διαχείριση απόκρυψης των στοιχείων ελέγχου. Αυτά είναι το *zoomControl* και το *attributionControl*, τα οποία ορίζονται με την τιμή *false*.

Έπειτα, καλείται η μέθοδος *.setView()* με παραμέτρους τις συντεταγμένες τις Ελλάδας, που έχουν οριστεί παραπάνω, μαζί με το επίπεδο *zoom*, που ορίζεται στο 6.

Τελευταίο βήμα για την αρχικοποίηση, είναι τη προσθήκη των map tiles από κάποιον πάροχο. Το βήμα αυτό, γίνεται ώστε να εμφανίζονται στο χάρτη, τα τετράγωνα περιοχών, σύμφωνα με τις παραμέτρους που έρισε το καινούργιο instance παραπάνω. Έτσι, καλείται η μέθοδος *L.tileLayer('https://s.tile.openstreetmap.org/{z}/{x}/{y}.png')*, και εισάγεται στο instance του χάρτη με τη μέθοδο *addTo()*:

## 5.3 Controls Χάρτη

Η δεύτερη επιλογή στο μενού, παρουσιάζει επίσης τη διαδικασία αρχικοποίησης των δυο χαρτών, αλλά με παραμετροποίηση των βασικών στοιχείων ελέγχου τους. Η σελίδα Controls φαίνεται στην Εικόνα 36, παρακάτω.



Εικόνα 36: Controls Χαρτών

### 5.3.1 Controls Google Maps Χάρτη

Η αρχικοποίηση των βασικών στοιχείων ελέγχου ενός Google Maps χάρτη έρχεται με την παραμετροποίηση των παρακάτω τιμών του options JSON αντικειμένου:

- *ZoomControl*.

- *MapTypeControl*.
- *ScaleControl*.
- *StreetViewControl*.
- *RotateControl*.
- *FullscreenControl*.

Οι δυνατότητες των παραπάνω αυτών παραμέτρων έχουν αναλυθεί εκτενέστερα στην υποενότητα του 4<sup>ου</sup> κεφαλαίου, 4.2.1. Το τελικό πλαίσιο κώδικα, για τον ορισμό αυτών των παραμέτρων, παρουσιάζεται στην Εικόνα 37.

```

1 var google_map = new google.maps.Map(document.getElementById("google_map"), {
2   zoom: 6,
3   center: new google.maps.LatLng(39.074, 23.824),
4   zoomControl: true,
5   mapTypeControl: true,
6   scaleControl: true,
7   streetViewControl: true,
8   rotateControl: true,
9   fullscreenControl: true
10 });

```

Εικόνα 37: Στοιχεία ελέγχου αντικειμένου options, χάρτη Google Maps.

### 5.3.2 Controls Open Street Maps Χάρτη

Το πλαίσιο κώδικα που επιτρέπει την παρουσίαση των βασικών στοιχείων ελέγχου αυτού του χάρτη, φαίνεται στην Εικόνα 38.

```

1 var greece = [ 39.074, 23.824 ];
2 var open_street_map = L.map('open_street_map', {
3   attributionControl: true,
4   zoomControl: true,
5 })
6 .setView(greece, 6);
7
8 L.tileLayer('https://s.tile.openstreetmap.org/{z}/{x}/{y}.png', {
9   maxZoom: 18,
10   minZoom: 3,
11   attribution: '© OpenStreetMap'
12 }).addTo(open_street_map);

```

Εικόνα 38: Στοιχεία ελέγχου αντικειμένου options, χάρτη Open Street Maps.

Οι δύο παράμετροι του αντικειμένου options, *attributionControl* και *zoomControl*, κατά την αρχικοποίηση, ορίζονται με τη τιμή `true`, ώστε να εμφανιστούν τα κουμπιά zoom και η *attribution* καρτέλα, κάτω δεξιά στον χάρτη.

Τέλος, με την εμφάνιση αυτών των δυο καρτελών, ορίζονται και τρεις ακόμη παράμετροι σε ένα δεύτερο JSON options αντικείμενο, κατά τη κλήση της μεθόδου `tileLayer()`. Οι παράμετροι του δεύτερου options αντικειμένου είναι οι εξής:

- *MaxZoom*. Ορίζει το μέγιστο επίπεδο zoom μέσω του κουμπιού «+».
- *MinZoom*. Ορίζει το ελάχιστο επίπεδο zoom μέσω του κουμπιού «-».
- *Attribution*. Ορίζει το όνομα της ταμπέλας attribution.

## 5.4 Markers

Η παρουσίαση των markers, αποτελεί την τρίτη επιλογή στο μενού. Η απεικόνιση των markers πάνω στους χάρτες Google Maps και Open Street Maps, φαίνονται στην Εικόνα 40 και στην Εικόνα 42 αντίστοιχα.

Σε αυτό το σημείο πρέπει να αναφερθεί πως για τις περιπτώσεις των markers και των tooltips, info windows, polyline circles, rectangles και polygons, που αναφέρονται παρακάτω, φορτώνεται κάθε φορά ένα αρχείο javascript με όνομα `geopoints.js`. Το αρχείο αυτό, εξάγει ένα JSON αντικείμενο, το οποίο περιέχει μια λίστα από σημεία αναφοράς ή αλλιώς γεω-σημεία (geopoints). Κάθε γεω-σημείο της λίστας, αποτελεί κι αυτό ένα JSON αντικείμενο που συγκρατεί κάποια χαρακτηριστικά. Πιο συγκεκριμένα, τα γεω-σημεία των παραδειγμάτων, αναφέρονται σε ελληνικά στάδια αθλητισμού, και συγκρατούν τις παρακάτω πληροφορίες:

- *Title*. Αποτελεί τον τίτλο του σταδίου.
- *Description*. Ορίζει μια γενική περιγραφή για το στάδιο.
- *Latitude*. Ορίζει το γεωγραφικό πλάτος του γηπέδου.
- *Longitude*. Ορίζει το γεωγραφικό μήκος του γηπέδου.
- *Capacity*. Ορίζει τη χωρητικότητα του σταδίου σε άτομα.

Τμήμα του αρχείου `geopoints.js` φαίνεται στην Εικόνα 39.

```
const geopoints = [
  {
    Title : "Γήπεδο Τούμπας",
    Description: "Το Γήπεδο της Τούμπας ...",
    Latitude : 40.61291551292003,
    Longitude : 22.972182669299624,
    Capacity : 29000
  },
  {
    Title : "Ολυμπιακό Στάδιο «Σπύρος Λούης»",
    Description: "Το Ολυμπιακό Στάδιο Αθηνών ...",
    Latitude : 38.03631715564138,
    Longitude : 23.787502853343906,
```

```

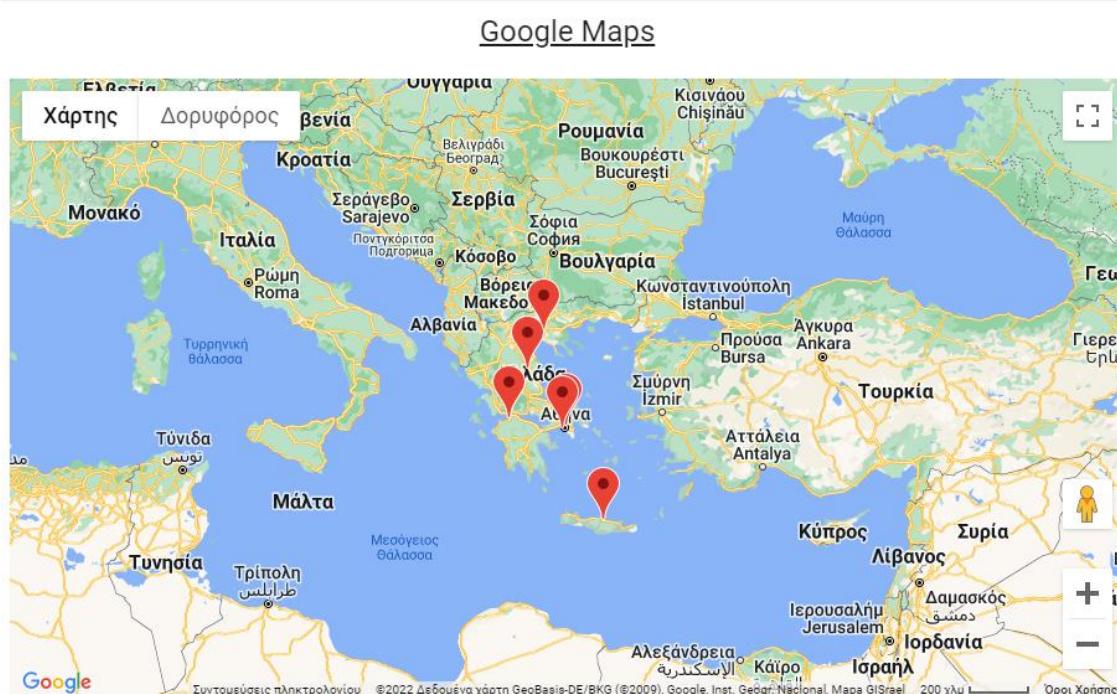
Capacity : 69618
},
{
Title : "Πήπεδο Γεώργιος Καραϊσκάκης",
Description: "Το Στάδιο «Γεώργιος Καραϊσκάκης» ...",
Latitude : 37.946729309659105,
Longitude : 23.664388869245307,
Capacity : 32115
},

```

Εικόνα 39: Τμήμα αρχείου geopoints.js.

#### 5.4.1 Markers Google Maps Χάρτη

Στον χάρτη Google Maps, τοποθετούνται όλα τα σημεία που έχουν φορτωθεί μέσω του αρχείου geopoints.js.



Εικόνα 40: Markers Google Maps χάρτη

Για την εισαγωγή ενός marker πάνω στον χάρτη πρέπει να χρησιμοποιηθεί, το instance του αντικειμένου χάρτη που αναφέρθηκε στην υποενότητα 5.2.1. Η δημιουργία marker, προϋποθέτει τη δημιουργία ενός καινούργιου τύπου αντικειμένου, το *Marker*. Η αρχικοποίηση γίνεται με το *new google.maps.Marker()*. Ο constructor του αντικειμένου, έχει ως παράμετρο ένα JSON αντικείμενο, με δύο παραμέτρους. Η πρώτη παράμετρος είναι το *position*. Αυτή αποτελεί ακόμα ένα JSON αντικείμενο με παραμέτρους *lat* και *lng*, οι οποίες παίρνουν τις τιμές *geopoint.Latitude* και *geopoint.Longitude* από το αρχείο με τα γεω-σημεία. Η δεύτερη παράμετρος είναι αυτή του *map*, όπου δέχεται το instance της μεταβλητής *map*, που έχει δημιουργηθεί, ώστε να εμφανιστεί εν τέλει ο marker πάνω

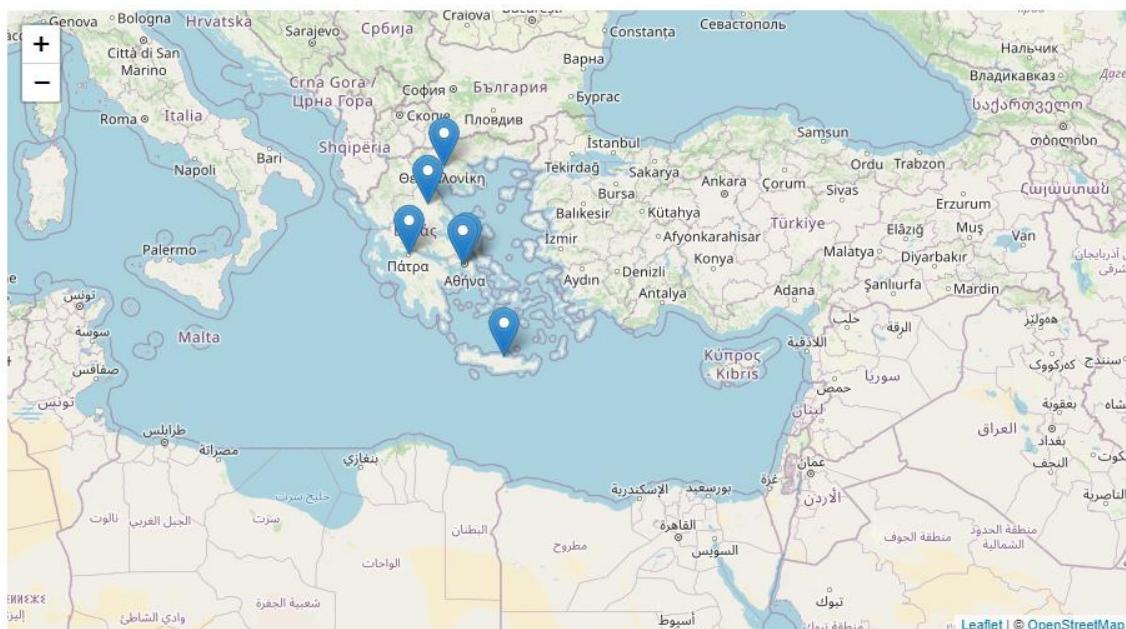
στον καμβά του Google Maps χάρτη. Σχετικό παράδειγμα κώδικα, βρίσκεται στην Εικόνα 41.

```
1 let marker = new google.maps.Marker({  
2     position: { lat: geopoint.Latitude, lng: geopoint.Longitude },  
3     map: google_map  
4 });
```

Εικόνα 41: Πλαίσιο κώδικα Google Maps χάρτη

#### 5.4.2 Markers Open Street Maps Χάρτη

Ο χάρτης Open Street Maps, με τη σειρά του, φορτώνει τα γεω-σημεία, αλλά με διαφορετική υλοποίηση.



Εικόνα 42: Markers Open Street Maps χάρτη

Η εισαγωγή marker, πραγματοποιείται με την κλήση της μεθόδου `.marker()`, μέσω του βασικού σημείου αναφοράς  $L$  της βιβλιοθήκης. Η μέθοδος αυτή, δέχεται την παράμετρο πίνακα συντεταγμένων `[geopoint.Latitude, geopoint.Longitude]`. Τέλος ακολουθεί η μέθοδος `.addTo()`, στην οποία εισάγεται το instance του αντικειμένου χάρτη που πραγματοποιήθηκε πριν, ώστε να μπορέσει να εμφανιστεί ο marker στον καμβά του υπάρχοντα χάρτη. Σχετικό παράδειγμα κώδικα, φαίνεται στην Εικόνα 43.

```
L.marker([geopoint.Latitude, geopoint.Longitude]).addTo(open_street_map);
```

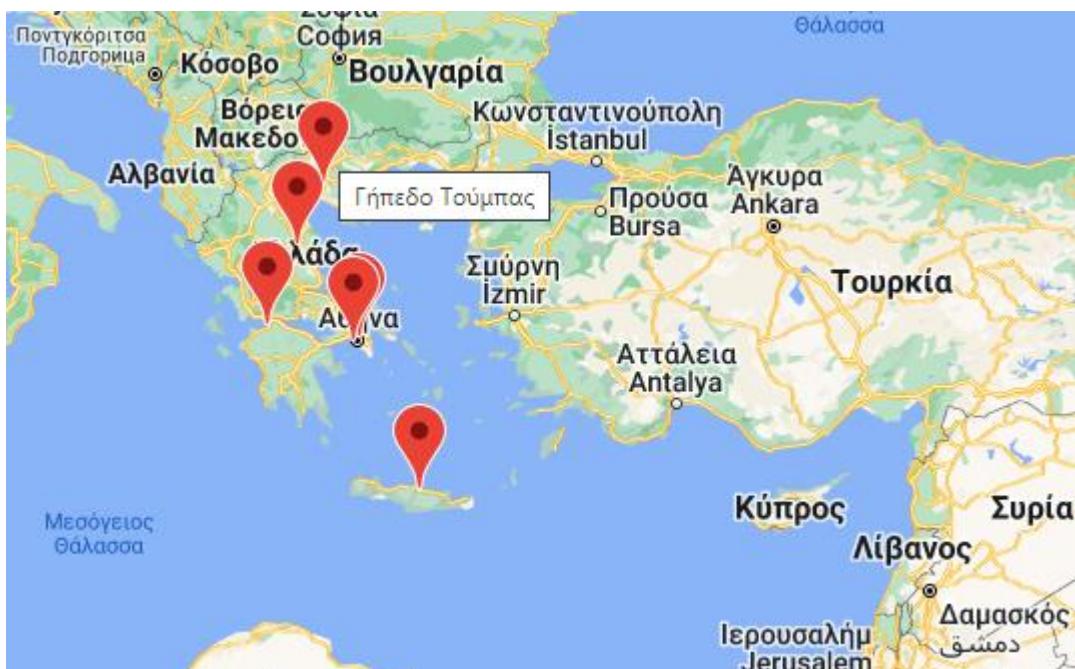
Εικόνα 43: Πλαίσιο κώδικα Open Street Maps Χάρτη

## 5.5 Tooltips

Η τέταρτη επιλογή του μενού είναι αυτή των Tooltips. Στη σελίδα αυτή φορτώνονται πάλι τα προκαθορισμένα γεω-σημεία από εξωτερικό αρχείο και τοποθετούνται ως markers επάνω στους δύο χάρτες. Μόνο που σε αυτό το παράδειγμα, οι markers εμφανίζουν ένα tooltip πληροφορίας, όταν ένας χρήστης τοποθετήσει τον κέρσορα επάνω σε έναν marker.

### 5.5.1 Tooltips Google Maps Χάρτη

Ένα tooltip σε google χάρτη φαίνεται στη Εικόνα 44.



Εικόνα 44: Tooltip Google Maps χάρτη

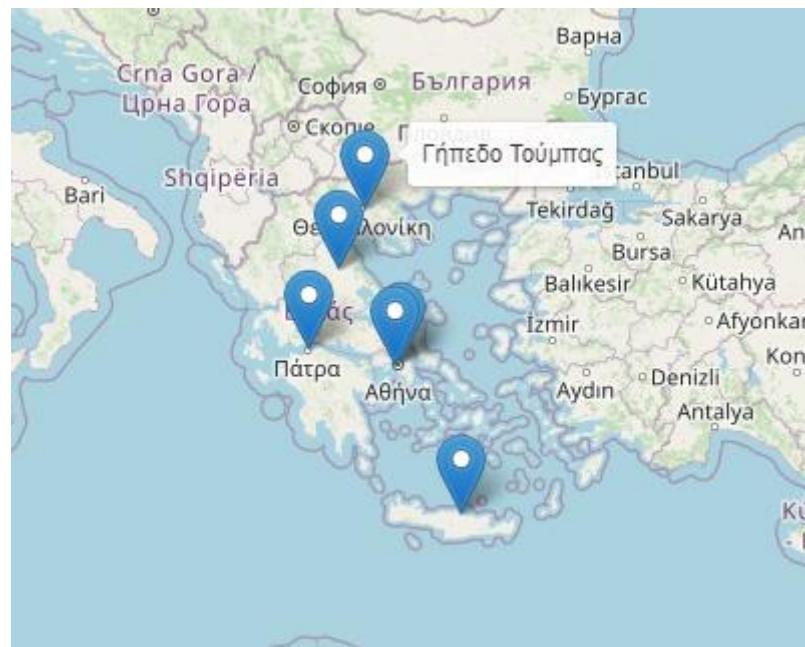
Για να ρυθμιστεί το tooltip, ακολουθείται η ίδια διαδικασία που εφαρμόστηκε στην υποενότητα 5.4.1, όμως το JSON αντικείμενο-παράμετρος του Marker, παίρνει ακόμα μια τιμή, την title, στην οποία ορίζεται ο τίτλος του εκάστοτε γεω-σημείου. Το παράδειγμα φαίνεται στην Εικόνα 45, παρακάτω.

```
1 let marker = new google.maps.Marker({
2     position: { lat: geopoint.Latitude, lng: geopoint.Longitude },
3     map: google_map,
4     title: geopoint.Title
5});
```

Εικόνα 45: Πλαίσιο κώδικα tooltip σε Google Maps χάρτη

### 5.5.2 Tooltips Open Street Maps Χάρτη

Ένα tooltip σε open street χάρτη φαίνεται στη Εικόνα 46.



Εικόνα 46: Open Street Maps tooltip

Η διαδικασία ρύθμισης του tooltip στο χάρτη γίνεται ακολουθώντας την ίδια διαδικασία της υποενότητας 5.4.2. Επιπλέον, καλείται η μέθοδος `.bindTooltip()`, η οποία εισάγει τον τίτλο του εκάστοτε γεω-σημείου, στον marker που δημιουργείται. Το πλαίσιο κώδικα για το συγκεκριμένο παράδειγμα παρουσιάζεται στην Εικόνα 47.

```
1 | L.marker([geopoint.Latitude, geopoint.Longitude])
2 |   .addTo(open_street_map)
3 |   .bindTooltip(geopoint.Title)
```

Εικόνα 47: Πλαίσιο κώδικα εισαγωγής tooltip σε Open Street Maps χάρτη

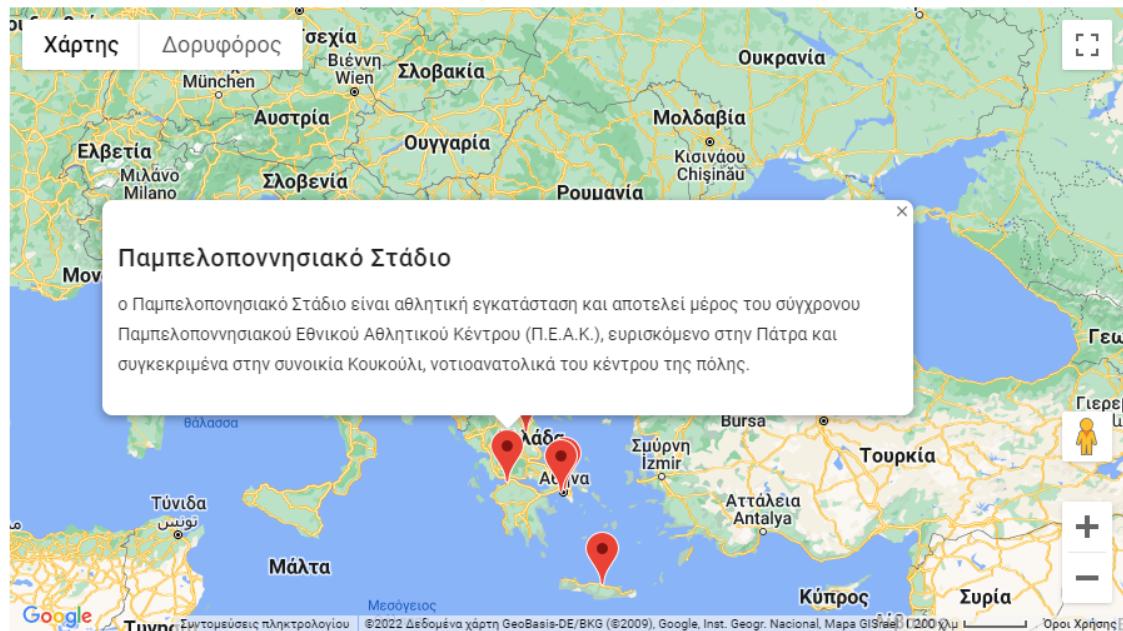
## 5.6 Info Windows

Τα info windows αποτελούν την πέμπτη επιλογή του μενού, με τη σελίδα να φορτώνει ξανά τα προκαθορισμένα γεω-σημεία.

### 5.6.1 Info Windows Google Maps Χάρτη

Προαπαιτούμενο αποτελεί η αρχικοποίηση των markers όπως στις προηγούμενες δυο υποενότητες και εισάγονται info windows πάνω σε αυτούς. Τα info windows εμφανίζονται κάνοντας κλικ σε ένα marker και φαίνονται όπως στην Εικόνα 48.

## Google Maps



Εικόνα 48: Info window Google Maps χάρτη

Το πλαίσιο κώδικα που χρησιμοποιείται για την εισαγωγή info window σε Google Maps φαίνεται στην Εικόνα 49.

```
1 let contentString = "Text you want to display";
2
3 let infowindow = new google.maps.InfoWindow({
4     content: contentString
5 });
6
7 let marker = new google.maps.Marker({
8     position: { lat: geopoint.Latitude, lng: geopoint.Longitude },
9     map: google_map
10 });
11
12 marker.addListener("click", () => {
13     infowindow.open({
14         anchor: marker,
15         map: google_map,
16         shouldFocus: true,
17     });
18});
```

Εικόνα 49: Πλαίσιο κώδικα εισαγωγής info window σε Google Maps χάρτη

Για την δημιουργία ενός Info Window, χρησιμοποιείται η αρχικοποίηση του αντικειμένου *InfoWindow()* της βιβλιοθήκης. Έτσι καλείται η γραμμή κώδικα *new google.maps.InfoWindow({content: contentString});*, στην οποία καταχωρείται ως παράμετρος μία μεταβλητή *contentString*. Η μεταβλητή αυτή, περιέχει τη πληροφορία

που θα εμφανιστεί μέσα στο Info Window και μπορεί να έχει ακόμα και formatting μέσω HTML κώδικα. Έπειτα, γίνεται η αρχικοποίηση του marker.

Για την εμφάνιση όμως του Info Window, κάνοντας κλικ πάνω στον marker, ακολουθεί ο κώδικα στην Εικόνα 50.

```
marker.addListener("click", () => {
  infowindow.open({
    anchor: marker,
    map: google_map,
    shouldFocus: true,
  });
});
```

Εικόνα 50: Click Listener Marker.

Ο κώδικας αυτός δημιουργεί έναν *listener*, οποίος εκτελεί την μέθοδο *.infowindow.open()*, όταν ο χρήστης κάνει κλικ πάνω στον σχετικό marker. Η παράμετρος που παίρνει η παραπάνω μέθοδος είναι ένα JSON αντικείμενο, που ορίζει σε ποιο αντικείμενο marker και σε ποιο χάρτη θα ανοίξει αυτό το info window, με τις παραμέτρους *anchor* και *map* αντίστοιχα.

### 5.6.2 Info Windows Open Street Maps Χάρτη

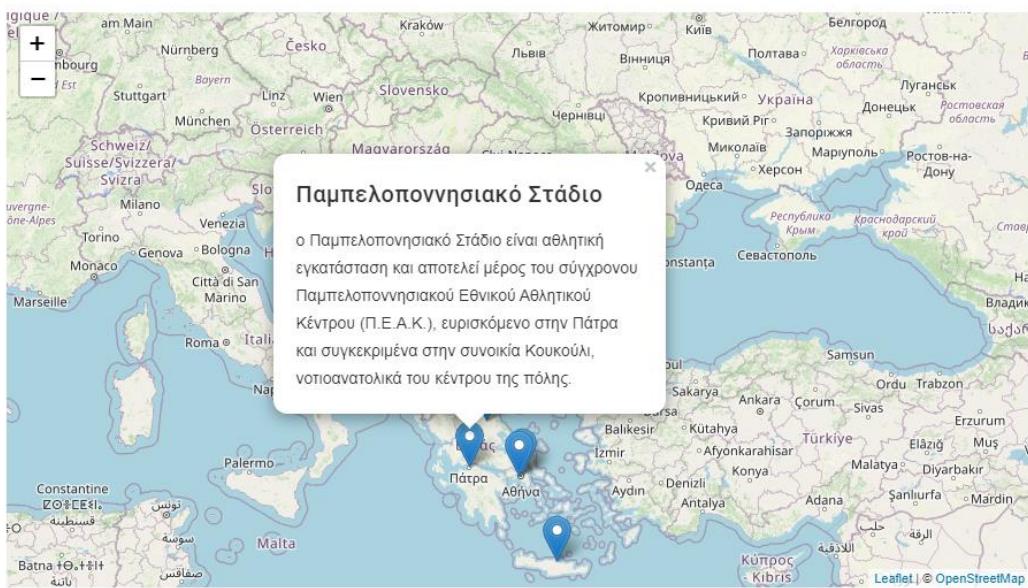
Η εισαγωγή info windows σε αυτό το παράδειγμα είναι ακόμα πιο απλή. Γίνεται αρχικοποίηση όλων των markers και για κάθε γεω-σημείο και καλείται ακόμα μια μέθοδος, η *.bindPopup()*. Ακολουθεί παράδειγμα στην Εικόνα 51.

```
1 | let contentString = "Text you want to display";
2 |
3 | L.marker([geopoint.Latitude, geopoint.Longitude]).addTo(open_street_map)
4 | .bindPopup(contentString)
```

Εικόνα 51: Πλαίσιο κώδικα εισαγωγής info window σε Open Street Maps χάρτη

Ένα παράδειγμα info window που παρουσιάζεται στην εφαρμογή είναι το παρακάτω, στην Εικόνα 52.

## Open Street Maps



Εικόνα 52: Εισαγωγή Info Window σε Open Street Maps χάρτη

## 5.7 Polylines

Τα polylines, ως έκτη επιλογή, εισάγουν το χρήστη στα shapes των δυο βιβλιοθηκών.

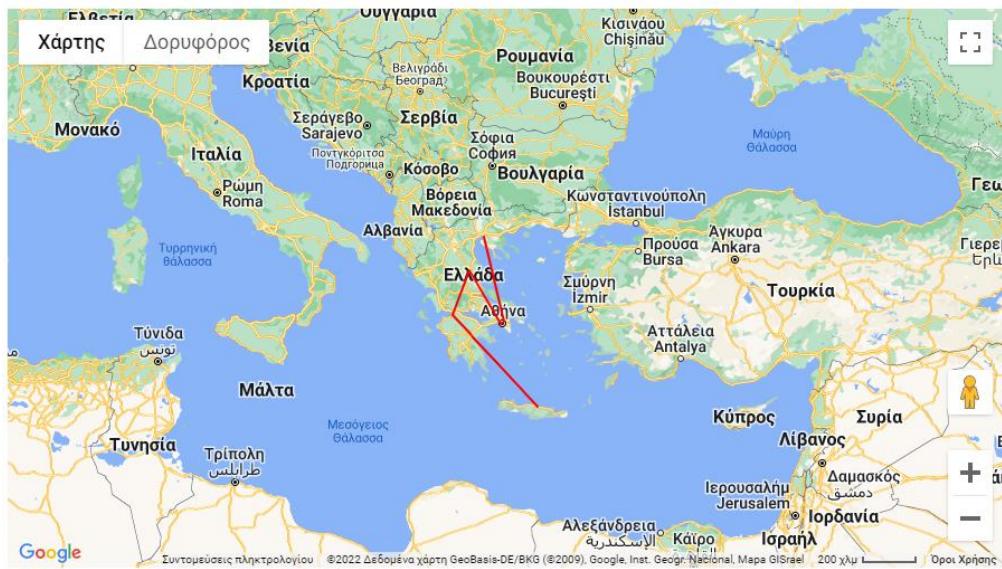
### 5.7.1 Polylines Google Maps Χάρτη

Τα polylines που παρουσιάζονται στο παράδειγμα της εφαρμογής ακολουθούν ένα μονοπάτι το οποίο ορίζεται από τις συντεταγμένες των γεω-σημείων, που φορτώνονται από το αρχείο geopoints.js.

Ο κώδικας που ακολουθείται στο παράδειγμα, αρχικοποιεί πρώτα μια μεταβλητή *travelCoordinates*, η οποία με μια μέθοδο *.forEach()* αποθηκεύει όλες τις συντεταγμένες των σημείων αναφορών των γεω-σημείων. Υστερα, δημιουργείται ένα νέο αντικείμενο τύπου *Polyline* της βιβλιοθήκης. Το αντικείμενο δέχεται ως παράμετρο ένα JSON αντικείμενο, το οποίο έχει ως παραμέτρους μεταβλητές απεικόνισης του polyline. Η πιο σημαντική μεταβλητή όμως είναι αυτή του *path*, η οποία ορίζει τη διαδρομή του polyline και στην οποία ορίζεται η *travelPlanCoordinates* που κρατάει αυτή τη πληροφορία.

Το παράδειγμα χάρτη και πλαισίου κώδικα παρουσιάζεται στην Εικόνα 53 και στην Εικόνα 54 αντίστοιχα.

## Google Maps



Εικόνα 53: Polyline Google Maps χάρτη

```
1 var travelPlanCoordinates = [];
2
3 geopoints.forEach(geopoint => {
4     travelPlanCoordinates.push({ lat: geopoint.Latitude, lng: geopoint.Longitude });
5 });
6
7 const travelPath = new google.maps.Polyline({
8     path: travelPlanCoordinates,
9     geodesic: true,
10    strokeColor: "#FF0000",
11    strokeOpacity: 1.0,
12    strokeWeight: 2,
13 });
14
```

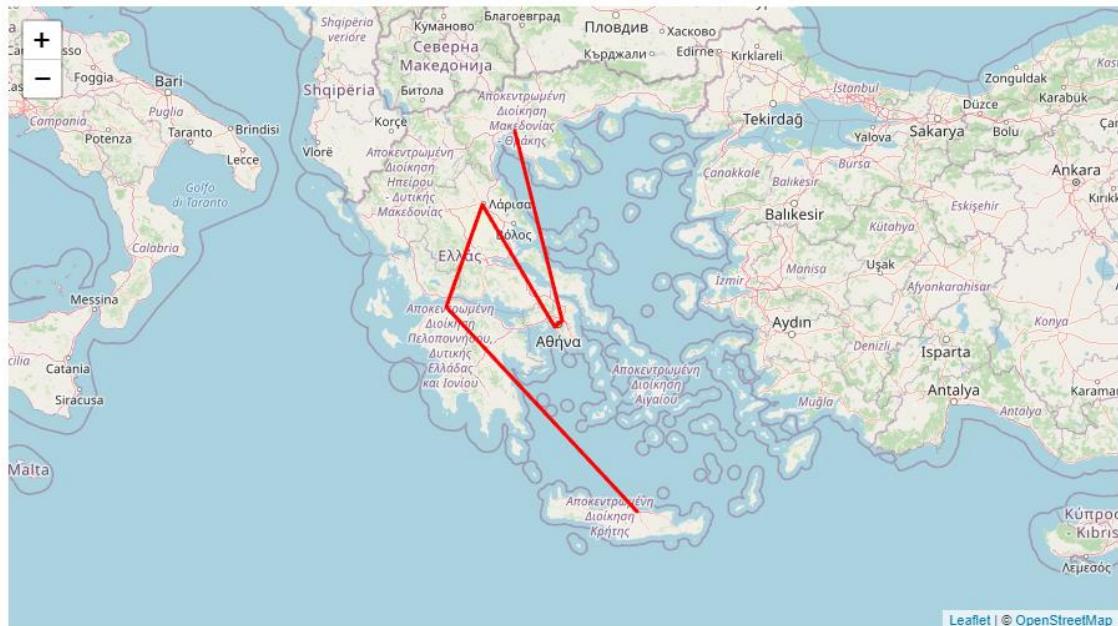
Εικόνα 54: Πλαίσιο κώδικα εισαγωγής Polyline σε Google Maps χάρτη

### 5.7.2 Polylines Open Street Maps Χάρτη

Σε αυτό το παράδειγμα ακολουθείται ή ίδια λογική με τη προηγούμενη υποενότητα, όσον αφορά τη διαδρομή polyline με τη μεταβλητή *travelPlanCoordinates*. Η σημαντική διαφορά είναι πως καλείται η μέθοδος *polyline()* μέσω του αντικειμένου *L*. Η μέθοδος αυτή δέχεται δύο μεταβλητές. Η πρώτη ορίζει τη διαδρομή του polyline. Σε αυτή αντιστοιχίζεται η μεταβλητή *travelPlanCoordinates*. Η δεύτερη μεταβλητή ορίζει ένα JSON αντικείμενο που δέχεται παραμέτρους που μορφοποιούν το σχηματιζόμενο polyline. Τέλος, ακολουθεί η μέθοδος *addTo()*, όπως και στα προηγούμενα παραδείγματα χρήσης χαρακτηριστικών της βιβλιοθήκης *Leaflet.js*.

Ακολουθούν παραδείγματα χάρτη και πλαισίου κώδικα στην Εικόνα 55 και στην Εικόνα 56 αντίστοιχα.

### Open Street Maps



Εικόνα 55: Polylines Open Street Maps χάρτη

```

1 var travelPlanCoordinates = [];
2 geopoints.forEach(geopoint => {
3     travelPlanCoordinates.push({ lat: geopoint.Latitude, lng: geopoint.Longitude });
4 });
5
6 var polyline = L.polyline(travelPlanCoordinates, {color: 'red'}).addTo(open_street_map);
7
8 // zoom the map to the polyline
9 map.fitBounds(polyline.getBounds());

```

Εικόνα 56: Πλαίσιο κώδικα εισαγωγής Polylines σε Open Street Maps χάρτη

## 5.8 Circles

Το χαρακτηριστικό circles αποτελεί την έβδομη επιλογή στο μενού. Ακολουθείται η λογική φόρτωσης των γεω-σημείων και η χρησιμοποίηση των στοιχείων τους, για την επίδειξη κύκλων επάνω στους χάρτες. Και στα δυο παραδείγματα σχηματίζονται κύκλοι στον χάρτη, με κεντρικό σημείο τις συντεταγμένες κάθε γεω-σημείου και εμβαδό, ανάλογα με τη χωρητικότητα του κάθε σταδίου.

### 5.8.1 Circles Google Maps Χάρτη

Ο σχηματισμός κύκλων επάνω στον Google Maps χάρτη φαίνεται στην Εικόνα 57.

## Google Maps



Εικόνα 57: Circles σε Google Maps χάρτη

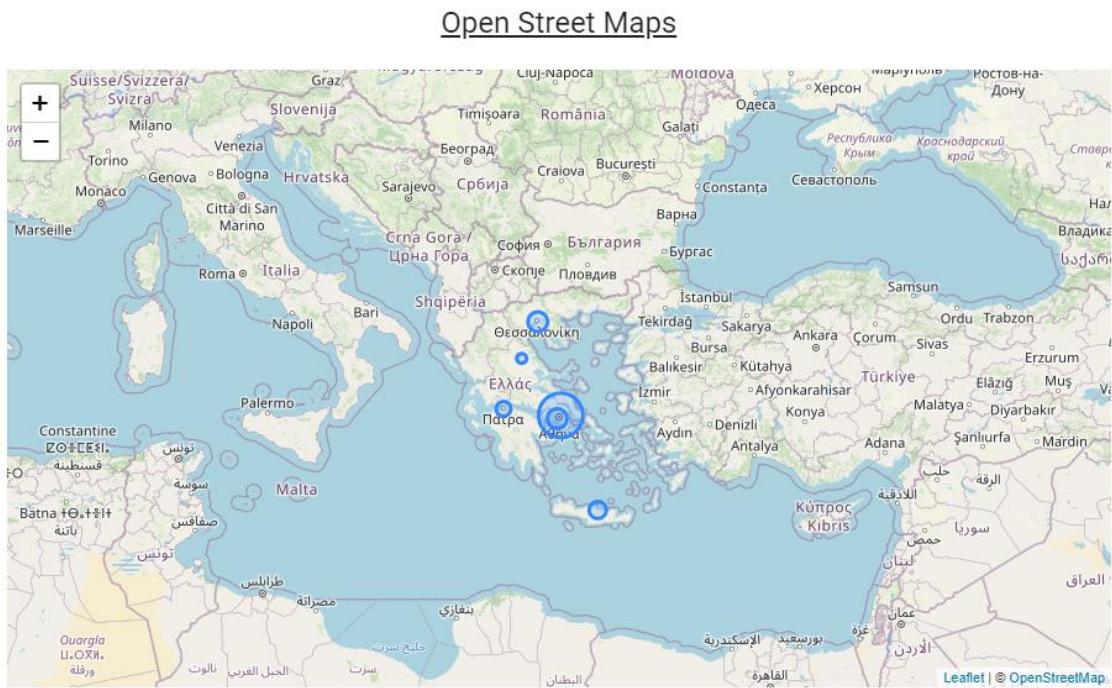
```
1 | geopoints.forEach(geopoint => {
2 |     var cityCircle = new google.maps.Circle({
3 |         strokeColor: "#FF0000",
4 |         strokeOpacity: 0.8,
5 |         strokeWeight: 2,
6 |         fillColor: "#FF0000",
7 |         fillOpacity: 0.35,
8 |         map: google_map,
9 |         center: { lat: geopoint.Latitude, lng: geopoint.Longitude },
10 |         radius: geopoint.Capacity,
11 |     });
12 | });
```

Εικόνα 58: Πλαίσιο κώδικα εμφάνισης κύκλων σε Google Maps χάρτη

Για κάθε γεω-σημείο που φορτώνεται, λειτουργεί ο κώδικας της Εικόνα 58. Δηλαδή, δημιουργείται, κάθε φορά, ένα καινούργιο instance αντικειμένου Circle της google maps βιβλιοθήκης με το κώδικα *new google.maps.Circle()*. Το αντικείμενο αυτό, δέχεται ως παράμετρο ένα JSON αντικείμενο options. Οι τρεις σημαντικοί παράμετροι αυτού, είναι το *map*, το *center* και το *radius*, στις οποίες ορίζονται το instance του χάρτη, οι συντεταγμένες του γεω-σημείου και η χωρητικότητα του σταδίου αντίστοιχα. Οι υπόλοιπες παράμετροι που παρατηρούνται, αφορούν τη μορφοποίηση του κύκλου που σχηματίζεται.

## 5.8.2 Circles Open Street Maps Χάρτη

Ο σχηματισμός κύκλων επάνω στον Open Street Maps χάρτη φαίνεται στην Εικόνα 59.



Εικόνα 59: Circles σε Open Street Maps χάρτη

Το πλαίσιο κώδικα, στην περίπτωση των κύκλων σε Open Street Map χάρτη είναι μικρότερο και πιο απλό. Για κάθε γεω-σημείο που φορτώνεται, καλείται η μέθοδος `.circle()` του κεντρικού αντικειμένου αναφοράς της βιβλιοθήκης leaflet.js. Η μέθοδος αυτή, δέχεται ως είσοδο έναν πίνακα με συντεταγμένες του γεω-σημείου και ένα JSON αντικείμενο `options`, το οποίο, στην περίπτωση του παραδείγματος, έχει μόνο την τιμή `radius` που ορίζει την ακτίνα του κύκλου. Όπως και στο παράδειγμα του google maps χάρτη, έτσι κι εδώ, τοποθετείται η τιμή `capacity`, δηλαδή η χωρητικότητα του γηπέδου. Τέλος ακολουθεί η μέθοδος `.addTo()` η οποία εισάγει τους κύκλους στο χάρτη, με αναφορά στο `instance` του αντικειμένου του Open Street Map χάρτη. Ο σχετικός κώδικας που αναλύθηκε μόλις, παρουσιάζεται στην Εικόνα 60.

```
1 geopoints.forEach(geopoint => {
2     L.circle([geopoint.Latitude, geopoint.Longitude], {radius: geopoint.Capacity}).addTo(open_street_map);
3 });

```

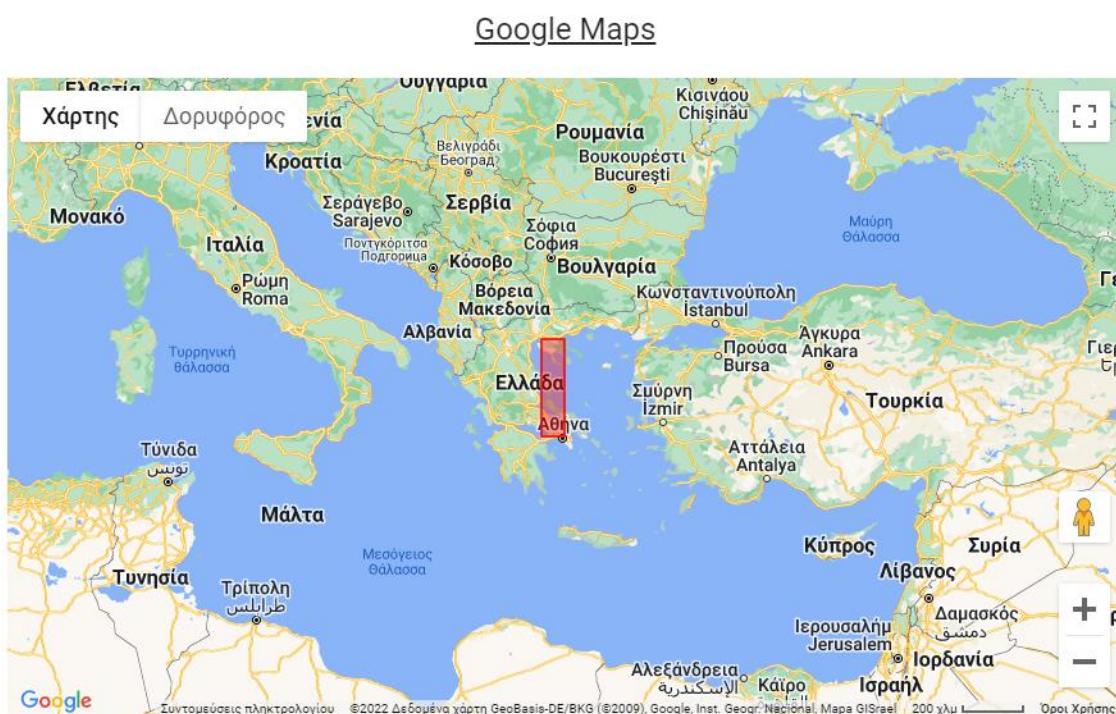
Εικόνα 60: Πλαίσιο κώδικα εισαγωγής circles σε open street maps χάρτη

## 5.9 Rectangles

Η εισαγωγή rectangles στους δυο χάρτες, είναι παραπλήσια με αυτή των circles στη προηγούμενη υποενότητα. Γίνεται φόρτωση όλων των γεω-σημείων από το αρχείο geo-points.js και ακολουθείται η διαδικασία σχεδιασμού, μόνο για το πρώτο, όμως, γεω-σημείο της λίστας.

### 5.9.1 Rectangles Google Maps Χάρτη

Ο σχεδιασμός ενός rectangle επάνω σε google maps χάρτης παρουσιάζεται στην Εικόνα 61.



Εικόνα 61 Rectangle σε Google Maps χάρτη

Το πλαίσιο κώδικα είναι ίδιο με αυτό του circles. Στην περίπτωση αυτή, χρησιμοποιείται το αντικείμενο *Rectangle()*. Το αντικείμενο δέχεται ένα JSON αντικείμενο ως παράμετρο, το οποίο, με τη σειρά του, δέχεται παραμέτρους μορφοποίησης, το αντικείμενο αναφοράς χάρτη και το JSON αντικείμενο *bounds*. Μετά τη δημιουργία του instance του αντικειμένου *Rectangle* στη μεταβλητή *rectangle*, καλείται η μέθοδός του *.setMap()*, με αναφορά στον χάρτη. Αυτή, «ζωγραφίζει» το ορθογώνιο επάνω του. Το πλαίσιο κώδικα που υποστηρίζει τα παραπάνω, φαίνεται στην Εικόνα 62.

```

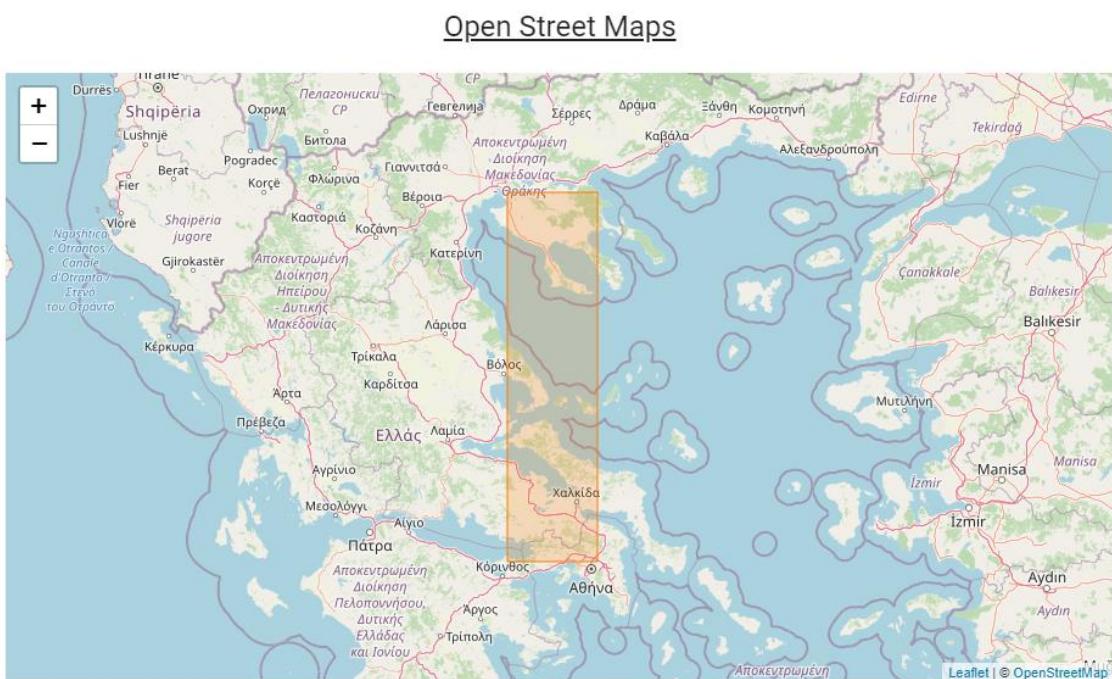
1  var rectangle = new google.maps.Rectangle({
2      strokeColor: "#FF0000",
3      strokeOpacity: 0.8,
4      strokeWeight: 2,
5      fillColor: "#FF0000",
6      fillOpacity: 0.35,
7      google_map,
8      bounds: {
9          north: geopoints[0].Latitude,
10         south: geopoints[0].Longitude,
11         east: geopoints[1].Latitude,
12         west: geopoints[1].Longitude,
13     }
14 };
15
16 rectangle.setMap(google_map);

```

Εικόνα 62: Πλαίσιο κώδικα εισαγωγής rectangle σε Google Maps χάρτη

### 5.9.2 Rectangles Open Street Maps Χάρτη

Ο σχηματισμός rectangle επάνω σε open street maps χάρτη φαίνεται στην Εικόνα 63.



Εικόνα 63: Rectangle σε Open Street Maps χάρτη

Η εισαγωγή rectangle, γίνεται με την μέθοδο `.rectangle()`. Αυτή, δέχεται ένα δισδιάστατο πίνακα με συντεταγμένες, με όνομα `bounds`, και ένα JSON αντικείμενο με παραμέτρους μορφοποίησης, όπως έιναι το `color`. Στη συνέχεια καλείται η μέθοδος `.addTo()`, η οποία εισάγει το rectangle επάνω στον χάρτη. Η μέθοδος `.fitBounds()` στη σειρά 4 της Εικόνα 64, διατάζει τον χάρτη να εστιάσει στο πλαίσιο των συντεταγμένων της μεταβλητής `bounds`.

```

1 var bounds = [[geopoints[0].Latitude, geopoints[0].Longitude], [geopoints[1].Latitude, geopoints[1].Longitude]];
2
3 L.rectangle(bounds, {color: "#ff7800", weight: 1}).addTo(open_street_map);
4 open_street_map.fitBounds(bounds);

```

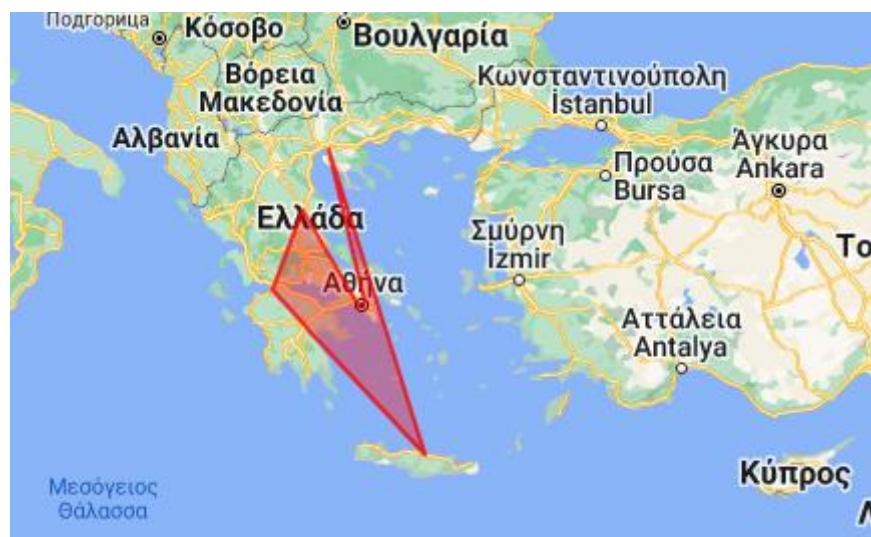
Εικόνα 64: Πλαίσιο κώδικα εισαγωγής rectangle σε Open Street Maps χάρτη

## 5.10 Polygons

Τα polygons, αποτελούν το τελευταίο είδος σχημάτων που καλύπτει η εφαρμογή επίδειξης. Ακολουθούν την ίδια λογική με τον κώδικα των polylines και η διαφορά τους είναι πως, το εσωτερικό των σημείων, που καλύπτει ένα polygon, καλύπτεται με χρώμα επάνω στον χάρτη. Στην περίπτωση των δυο χαρτών σχηματίζεται ένα polygon, από το σύνολο των συντεταγμένων των σημείων που φορτώνονται από το αρχείο geopoint.js.

### 5.10.1 Polygons Google Maps Χάρτη

Το πολύγωνο που σχηματίζεται από τις συντεταγμένες των γεω-σημείων επάνω στον Google Maps χάρτη, παρουσιάζεται στην Εικόνα 65.



Εικόνα 65: Polygon Google Maps χάρτη

Ο κώδικας υλοποίησης του παραπάνω παραδείγματος, ξεκινά την αρχικοποίηση μιας μεταβλητής *polygonsCords*, στην οποία αποθηκεύονται αντικείμενα συντεταγμένων απ' όλα τα γεω-σημεία που έχουν φορτωθεί. Αυτό γίνεται με τη μέθοδο *.forEach()* στις σειρές 4 με 6 της Εικόνα 66.

Υστερα αρχικοποιείται το αντικείμενο *Polygon*, το οποίο δέχεται ως παράμετρο ένα JSON αντικείμενο. Αυτό, με τη σειρά του, έχει παραμέτρους μορφοποίησης και τη παράμετρο *path*, στην οποία ορίζεται η μεταβλητή *polygonCords*.

Τέλος, το πολύγωνο ζωγραφίζεται στον χάρτη με τη μέθοδο *.setMap()*; και ορίζοντας ως παράμετρο το instance του google maps χάρτη.

```
1 // Define the LatLang coordinates for the polygon's path.
2 const polygonsCords = [];
3
4 geopoints.forEach(geopoint => {
5     polygonsCords.push({ lat: geopoint.Latitude, lng: geopoint.Longitude });
6 });
7
8 // Construct the polygon.
9 const courtsPolygon = new google.maps.Polygon({
10     paths: polygonsCords,
11     strokeColor: "#FF0000",
12     strokeOpacity: 0.8,
13     strokeWeight: 2,
14     fillColor: "#FF0000",
15     fillOpacity: 0.35,
16 });
17
18 courtsPolygon.setMap(google_map);
```

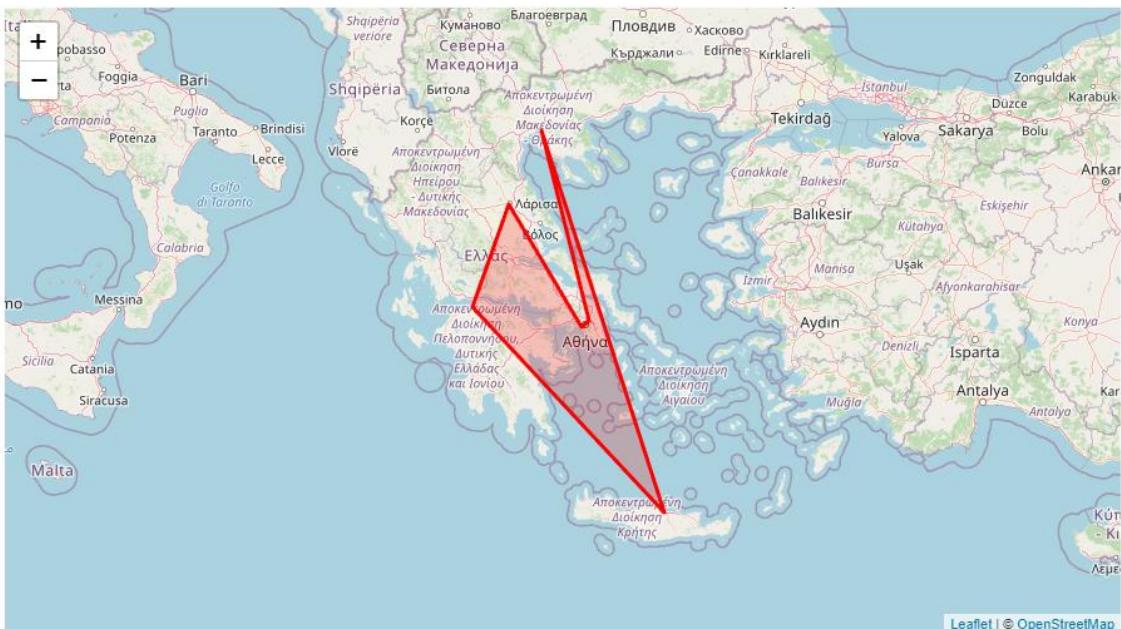
Εικόνα 66: Πλαίσιο κώδικα εισαγωγής polygon σε Google Maps χάρτη

### 5.10.2 Polygons Open Street Maps Χάρτη

Ο κώδικας της εισαγωγής polygon σε Open Street Map χάρτη, είναι ίδιος με αυτόν της εικόνας 65. Η διαφορά είναι πως η εισαγωγή του πολυγώνου γίνεται με τη μέθοδο *.Polygon()*, η οποία δέχεται μια μεταβλητή συνόλου συντεταγμένων και ένα JSON αντικείμενο με παραμέτρους μορφοποίησης. Τέλος καλείται η μέθοδος *.addTo()*, για εισαγωγή στο χάρτη.

Παραδείγματα απεικόνισης πολυγώνου και πλαισίου κώδικα φαίνονται στην Εικόνα 67 και στην Εικόνα 68 αντίστοιχα.

## Open Street Maps



Εικόνα 67: Polygon σε Open Street Maps χάρτη

```
1 | var latlngs = [];
2 |
3 | geopoints.forEach(geopoint => {
4 |   latlngs.push([ geopoint.Latitude, geopoint.Longitude ]);
5 | });
6 |
7 | var polygon = L.polygon(latlngs, {color: 'red'}).addTo(open_street_map);
8 |
9 | // zoom the map to the polygon
10| open_street_map.fitBounds(polygon.getBounds());
```

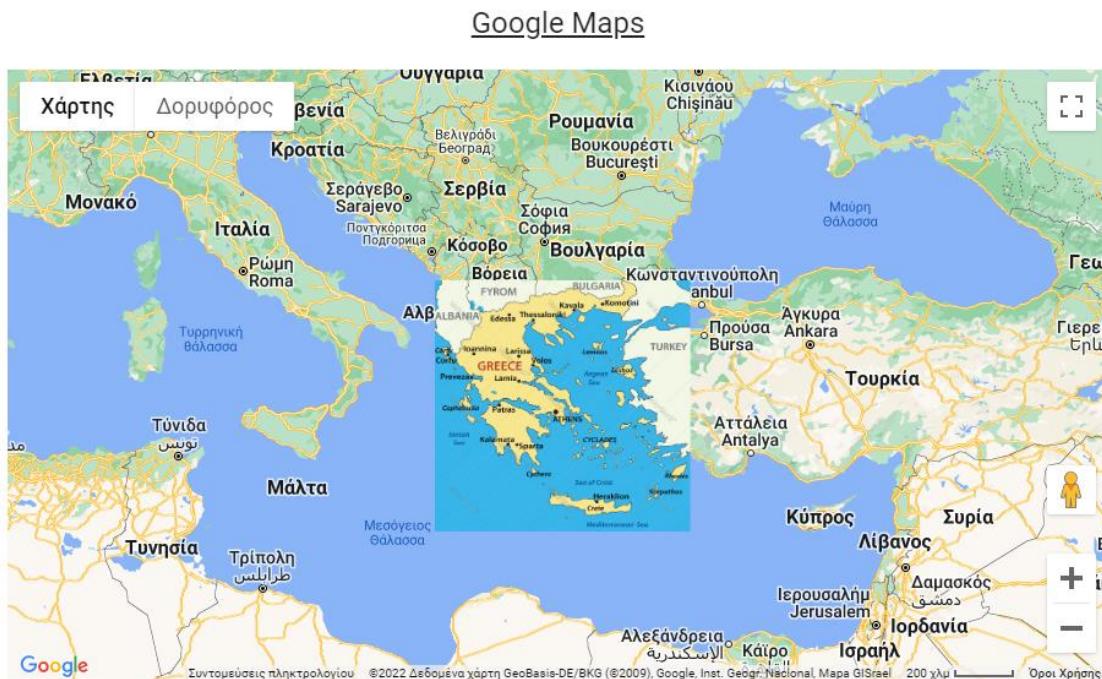
Εικόνα 68: Πλαίσιο κώδικα εισαγωγή polygon σε Open Street Maps χάρτη

## 5.11 Ground / Image Overlays

Τα ground και image overlays αποτελούν το τελευταίο παράδειγμα με επίδειξη κώδικα και το τελευταίο κοινό χαρακτηριστικό ανάμεσα στους δυο χάρτες. Ουσιαστικά, αφορά την τοποθέτηση μιας εικόνας χάρτη της Ελλάδας επάνω από τις γεωγραφικές συντεταγμένες της Ελλάδας, στους δυο χάρτες. Και στις δυο περιπτώσεις, γίνεται αρχικοποίηση μιας μεταβλητής *imageBounds*, η οποία μεταφράζεται σε δυο σημεία. Τα σημεία αυτά αφορούν τις συντεταγμένες που θα καλύψει η προστιθέμενη εικόνα από το πιο νοτιοδυτικό έως το πιο βορειοανατολικό σημείο στους χάρτες.

### 5.11.1 Ground Overlays Google Maps Χάρτη

Η τοποθέτηση της εικόνας στον Google Maps χάρτη φαίνεται στην Εικόνα 69.



Εικόνα 69: Ground overlay σε Google Maps χάρτη

Η τοποθέτηση γίνεται με την αρχικοποίηση ενός αντικειμένου `GroundOverlay`. Αυτό δέχεται παραμέτρους το `path` της εικόνας και τη μεταβλητή `imageBounds`. Ύστερα, με την κλήση της μεθόδου `setMap()`, η εικόνα ζωγραφίζεται στις συντεταγμένες που ορίστηκαν. Παράδειγμα πλαισίου κώδικα, ακολουθεί στην Εικόνα 70.

```
1 var imageBounds = new google.maps.LatLngBounds(
2     new google.maps.LatLng(34.60737710561527, 19.375888049080856), //SW
3     new google.maps.LatLng(41.68075941231813, 28.543628959853727) //NE
4 );
5
6 historicalOverlay = new google.maps.GroundOverlay(
7     "../../ancient-greece.jpg",
8     imageBounds
9 );
10 historicalOverlay.setMap(google_map);
11
```

Εικόνα 70: Πλαίσιο κώδικα εισαγωγής ground overlay σε Google Maps χάρτη

### 5.11.2 Ground Overlays Open Street Maps Χάρτη

Η εισαγωγή `image overlay` σε αυτήν την περίπτωση με την κλήση της μεθόδου `imageOverlay()`. Η μέθοδος αυτή, δέχεται ως παραμέτρους το `path` της εικόνας και τη μεταβλητή `imageBounds`. Τέλος, όπως όλα τα προηγούμενα παραδείγματα, καλείται η

.`addTo()`, για την εισαγωγή του overlay στον χάρτη. Ακολουθούν παραδείγματα στην Εικόνα 71 και στην Εικόνα 72.

### Open Street Maps



Εικόνα 71: Image Overlay σε Open Street Maps χάρτη

```
1 var imageUrl = "./../ancient-greece.jpg";
2 imageBounds = [[41.68075941231813, 19.375888049080856], [34.60737710561527, 28.543628959853727]];
3 L.imageOverlay(imageUrl, imageBounds).addTo(open_street_map);
```

Εικόνα 72: Πλαίσιο κώδικα εισαγωγής image overlay σε Open Street Maps χάρτη

## 5.12 Demo Εφαρμογής επίδειξης

Η επιλογή Demo, αποτελεί τη τελευταία σελίδα της εφαρμογής επίδειξης. Δίνει στον χρήστη τη δυνατότητα να πειραματιστεί με τις δυνατότητες που παρουσιάστηκαν στις προηγούμενες σελίδες.

Κατά τη φόρτωση της σελίδας, ο χρήστης παρατηρεί τους δύο χάρτες, ενώ από κάτω του εμφανίζεται μια λίστα από κουμπιά, με επιλογές στα αριστερά και μια λίστα οδηγιών στα δεξιά.

Η λίστα επιλογών κουμπιών φαίνεται στην Εικόνα 73.

Load geopoints from a file and presents them with clustering

**SHOW POINTS OF INTEREST**

**HIDE POINTS OF INTEREST**

Draw shapes based on the markers that are currently pinned on the map

**DRAW POLYLINES**

**DRAW RECTANGLES**

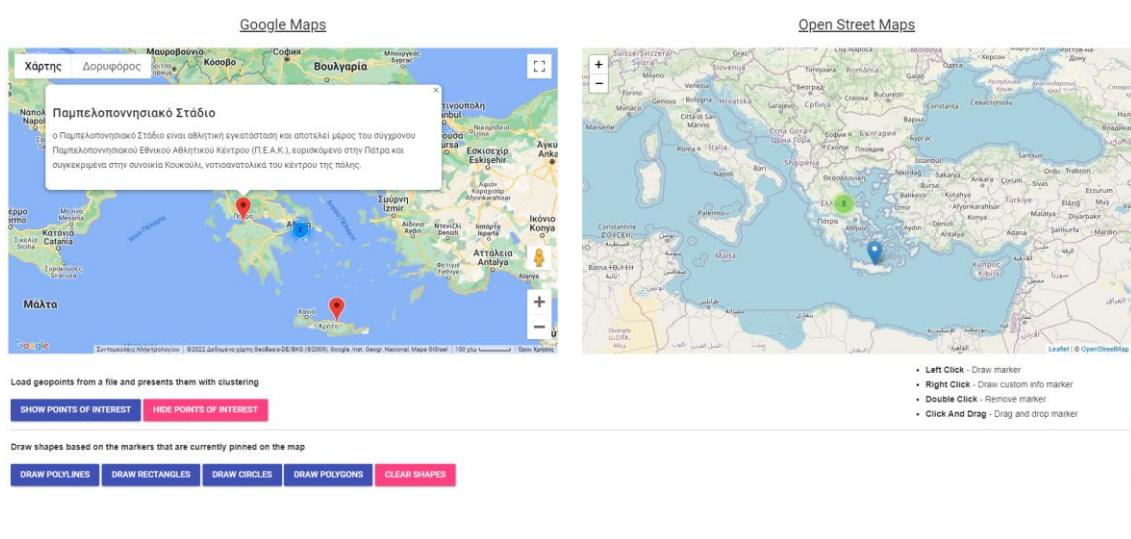
**DRAW CIRCLES**

**DRAW POLYGONS**

**CLEAR SHAPES**

Εικόνα 73: Επιλογές κουμπιών εφαρμογής Demo

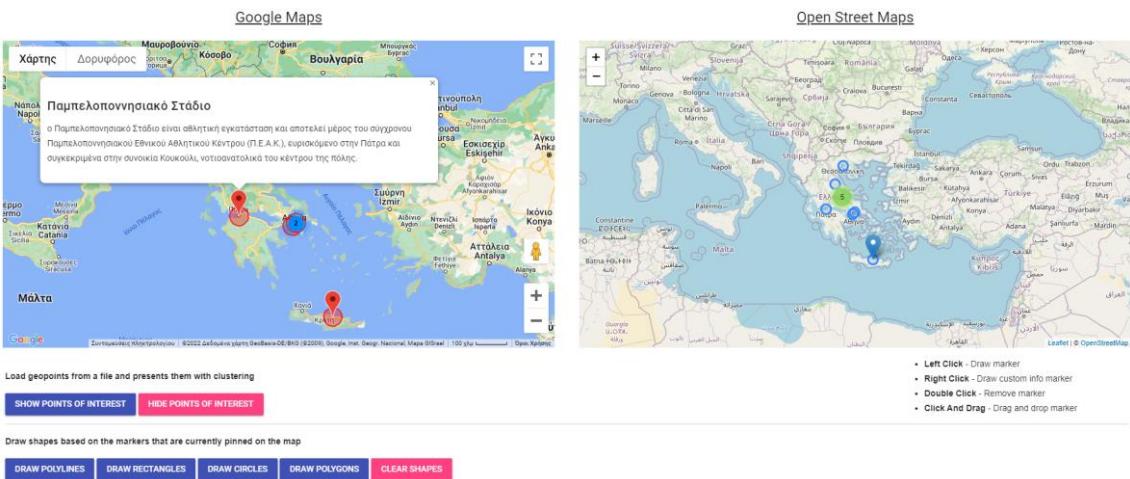
Το κουμπί «SHOW POINTS OF INTEREST», φορτώνει όλα τα γεω-σημεία από το αρχείο geopoints.js. Έτσι, δημιουργούνται markers με tooltips και info windows με τη πληροφορία των γεω-σημείων. Οι markers αυτοί, ζωγραφίζονται επίσης με τη μέθοδο clustering, όπως φαίνεται στην Εικόνα 74.



Εικόνα 74: Πάτημα κουμπιού “SHOW POINTS OF INTEREST”.

Πατώντας το κουμπί «HIDE POINTS OF INTEREST», εξαφανίζονται όλοι οι markers και από τους δυο χάρτες.

Η δεύτερη γραμμή με κουμπιά, αφορά την δημιουργία όλων των shapes που έχουν αναφερθεί έως τώρα. Ανάλογα, με τους markers που υπάρχουν επάνω στους δυο χάρτες, δημιουργείται και το κατάλληλο shape, όταν ο χρήστης πατάει το αντίστοιχο κουμπί. Παρακάτω, στην Εικόνα 75, εμφανίζεται ένα παράδειγμα πατήματος του κουμπιού «Draw Circles», όταν ο χρήστης έχει φορτώσει τους markers από το αρχείο geopoints.js.



Εικόνα 75: Πάτημα κουμπιού “DRAW CIRCLES”

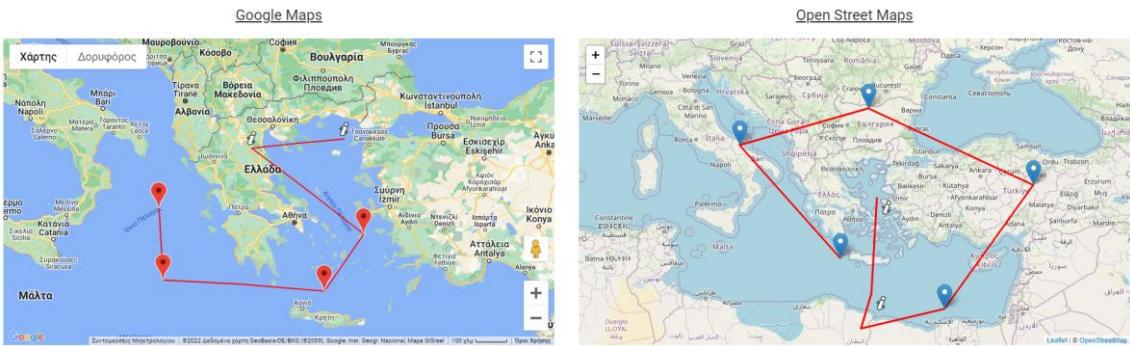
Το κουμπί «CLEAR SHAPES», με τη σειρά του, διαγράφει όλα τα ζωγραφισμένα σχήματα που υπάρχουν επάνω στους δύο χάρτες.

Με τη σειρά της, η λίστα στα δεξιά, δείχνει στον χρήστη μια λίστα επιλογών που έχει με το κέρσορά του επάνω στους δύο χάρτες. Η εκτέλεση των οδηγιών αυτών, πυροδοτεί κάποια βασικά συμβάντα της γλώσσας javascript. Αυτά είναι τα εξής:

- *Aριστερό Click.* Προσθέτει marker επάνω στο χάρτη.
- *Δεξί Click.* Προσθέτει custom marker (info icon) επάνω στο χάρτη.
- *Διπλό Αριστερό Click.* Αφαιρεί marker.
- *Drag and Drop.* Μετακίνηση marker.

Ο χρήστης, με την τοποθέτηση δικών του marker, μπορεί ταυτόχρονα να χρησιμοποιήσει και τα κουμπιά που υπάρχουν στο μενού αριστερά και να δει τα αποτελέσματα των σχημάτων να εφαρμόζονται και στα σημεία που μόλις τοποθέτησε.

Παράδειγμα χρήσης των οδηγιών αυτών φαίνεται στην Εικόνα 76, παρακάτω.



Εικόνα 76: Παράδειγμα χρήσης οδηγιών κέρσορα



## 6 Συμπεράσματα

Η δημιουργία της εν λόγω εφαρμογής-tutorial του κεφαλαίου 5, αποτελεί παράδειγμα του πόσο εύκολη είναι η διαδικασία ενσωμάτωσης ενός ηλεκτρονικού χάρτη σε μια εφαρμογή, αλλά, ταυτόχρονα, και το πόσο εύχρηστη μπορεί να γίνει αυτή, με την προσθήκη χαρτών.

Η χρήση χαρτών μέσω των δύο αυτών βιβλιοθηκών, όχι μόνο μαθαίνει σε έναν καινούργιο προγραμματιστή πως να ενσωματώνει στον κώδικά του εξωτερικές βιβλιοθήκες, αλλά και για το πόσο σημαντικός είναι ο εμπλουτισμός των εφαρμογών του για την εμπειρία του χρήστη, με τη χρήση διαδραστικών χαρτών.

Μελλοντικές εξελίξεις για την εφαρμογή επίδειξης, θα μπορούσαν να είναι η ενσωμάτωση περισσότερων βιβλιοθηκών χαρτών. Επίσης θα μπορούσε να εφαρμοστεί μια ολική αναπροσαρμογή στο UI, δηλαδή το γραφικό περιβάλλον, ώστε να είναι εύχρηστο σε όλων των ειδών τις συσκευές, όπως smartphones και tables. Τέλος, στη Demo παρουσίαση, θα μπορούσαν να προστεθούν κι άλλες επιλογές, ώστε ένας χρήστης να μπορεί να πειραματιστεί περισσότερο και να εξερευνήσει μαζικά, όλες τις δυνατότητες που του δίνουν οι βιβλιοθήκες.



# Βιβλιογραφία

- [1] [LINK](#) - Ασβεστάς Νικόλαος, “Δημιουργία Δικτυακής Εφαρμογής Διαχείρισης Γεωγραφικών Δεδομένων”, pp. 3-5, 2008
- [2] [LINK](#) - Κυπριάδου Άννα, “Σχεδίαση και Υλοποίηση ενός Γεωγραφικού Πληροφοριακού Συστήματος σε Περιβάλλον Arc GIS”, A.E.M. 2135, pp. 28-30, 2015
- [3] [LINK](#) - Σουλακέρης Νικόλας & Ρούσσου Όλγα, “Ατλας πυρκαγιών της Εύβοιας”, Μυτιλήνη, pp. 7-15, 2019
- [4] [LINK](#) – Bąk Tomasz – “Mapbox vs Google Maps — What are the differences?” [07/03/2020]
- [5] [LINK](#) – Ceci L., “Most popular mapping apps in the U.S. 2021, by downloads” [02/03/2022]
- [6] [LINK](#) – Luden Ingrid, “Google Bought Waze For \$1.1B, Giving A Social Data Boost To Its Mapping Business” [06/03/2022]
- [7] [LINK](#) – Olson Christi & Kemery Kelli, “Voice report - From answers to action: customer adoption of voice technology and digital assistants” [02/03/2022]
- [8] [LINK](#) – Google Developers, “Using API Keys” [02/03/2022]
- [9] [LINK](#) – OpenStreetMap Official Wiki, “About OpenStreetMap” [07/03/2022]
- [10] [LINK](#) – OpenStreetMap Stats [07/03/2022]
- [11] [LINK](#) – OpenStreetMap Official Wiki, “Why OpenStreetMap?” [07/03/2022]
- [12] [LINK](#) – Waze Official Documentation, “How does Waze work?” [06/03/2022]