



*Получена: 10.01.2022 г.*

*Приета: 14.02.2022 г.*

## **DEVELOPMENT OF MAPBOX GIS APPLICATION FOR LOCATION AND PEDESTRIAN NAVIGATION WITH WEB ACCESS FROM COMPUTERS AND MOBILE DEVICES**

**T. Ilieva<sup>1</sup>**

*Keywords: Mapbox, GIS application, web and mobile GIS, location and navigation*

### **ABSTRACT**

The paper presents a developed application that is Internet-based and uses online and inline resources. It is installed only with copying on a single web page and can work on a computer or laptop, as well as on mobile devices. HTML and CSS are used for developing the layout of the page, and the functionality is provided by JavaScript in combination with tools for real-time location and navigation, which are available from the Mapbox platform.

### **1. Introduction**

In the last few years, the web-based and mobile GIS are very popular and are widely used in combination with different methods for obtaining the user location in real time, route planning, tracking and optimization.

The GIS as software applications are easy for use and in web or mobile version are compatible with mobile phones or other devices, and what is more, nowadays there are also a lot of libraries and platforms which could be used for free. In this way the more advanced users with minimal programming knowledge could get the user location from the device and put it on a map, to plan routes, track and optimize them.

This current realization of GIS application for obtaining location and pedestrian navigation is part of the so-called personal navigation services, which support the individual positioning, the selection and finding of a desired location, and guidance to the location on the basis of different means of transport.

---

<sup>1</sup> Tamara Ilieva, Chief Assist. Prof. Dr. Eng., Dept. "Geodesy and Geoinformatics", UACEG, 1 H. Smirnovski Blvd., Sofia1046, e-mail: Ilieva\_tamara@yahoo.com

Recent studies [1, 2, 3] show that over 75% and up to 90% of smartphone owners regularly use personal navigation apps and 65% of them use the most popular navigation app Google Maps [4], but there are OpenStreetMap [5], HERE [6], Mapbox [7] and many, many others. Also, one of the leading trends for 2022 are the geolocation apps, which are treated as multi-purposeful mobile technology that suits any industry [8].

For the purposes of this research an application has been developed, which could be accessed from computer or mobile device and has functionality for adding location data and personal pedestrian navigation. The Mapbox platform was used. The development process is described below.

## **2. Development of the application, methodology**

The development of the application consists of two main stages – concept and realization. Each of them is described in the next paragraphs.

### **2.1. Concept**

The concept for Internet-based GIS application for location and pedestrian navigation that can be accessed through a desktop and mobile browser includes the following:

- Description of the required functionality;
- Requirements for platform selection and data access;
- Definition of a coordinate system for representing the data.

#### **2.1.1. Description of the required functionality**

The required functionality for the application should include at least the following:

- Base map and data layers;
- Visualization control tools – zoom, rotate screen;
- Geocoding search tools;
- User geolocation;
- Options for route planning and tracking;
- Display instructions for personal pedestrian navigation;
- Options for copying and sharing the data.

#### **2.1.2. Requirements for platform selection and data access**

In compliance with the description of the required functionality the platform which should be used must have libraries, tools and functions for: loading a base map, publishing different data layers from online and inline sources, geocoding, adding visualization, location and navigation controls, displaying navigation instructions.

All these requirements are met by the selected Mapbox platform. A Mapbox Token is used to add a base map and associate API requests with a specific account. This Token is used also to access other features from the platform such as geocoding, routing and more. It is necessary for an account in Mapbox to be created and the Token to be referenced in the application code.

### 2.1.3. Coordinate system for representing the data

The application uses EPSG:4326 coordinate reference system [9] – WGS84 datum, decimal degrees, and longitude and latitude units. The definition of this coordinate system in WKT format is as follows:

```
GEOGCS["WGS 84",  
  DATUM["WGS_1984",  
    SPHEROID["WGS 84",6378137,298.257223563,  
      AUTHORITY["EPSG","7030"]],  
    AUTHORITY["EPSG","6326"]],  
  PRIMEM["Greenwich",0,  
    AUTHORITY["EPSG","8901"]],  
  UNIT["degree",0.0174532925199433,  
    AUTHORITY["EPSG","9122"]],  
  AUTHORITY["EPSG","4326"]]
```

Mapbox Tiling Service and Mapbox SDKs for mobile and web accept GeoJSON that follows the GeoJSON standard [10, 11] for a geographic coordinate reference system equivalent to OGC:CRS84. Most Mapbox upload tools also require coordinates ordered as longitude, latitude, so when using data directly from the EPSG:4326 reference frame the coordinates should be reversed.

## 2.2. Realization

Before adding any functionality, an HTML file is prepared containing the main elements of the application. Their styling is defined with CSS.

In the <head> section of the HTML file the links that will be used to access the online libraries and sources are specified. In its <style> section the different visualization styles are set. In the <body> all the elements are added. All the functionalities which have to be performed are in the <script> section.

### 2.2.1. Adding a base map

A Mapbox Public Token is needed to add a base map. In this case first the necessary links in the <head> section are set and then the map is called up by using mapboxgl.Map.

Style, center, and zoom are set in order to define the initial appearance of the map [12].

### 2.2.2. Adding a vector tile layer from Mapbox online source

The details for the vector tile data sources must be specified according to the TileJSON specification. One of the ways to do this is to add each selected Mapbox hosted layer by using its URL (mapbox:// + tileset ID).

The necessary links are added to the <head> HTML section. In the <body> section the repository from which vector layers will be added to the map is first specified and after that the necessary layers from the repository could be added one by one. A description for the layer visualization must be specified [13].

In this case only one Mapbox vector tile layer is added – terrain contours, mapped to 10-meter height increments [14].

### **2.2.3. Adding inline user defined data and display a popup with information for objects**

In the application some user defined objects were added to the map - university, bank, etc. They are visualized with icons and on click the information entered about them appears in a popup [15].

For this purpose, Mapbox standard icons are used, the so-called Maki icons – a predefined set of SVG icons for different type of objects [16]. The icons are added as inline GeoJSON at exact coordinates, and for each of them was given the corresponding name and size (for example 'bank-15').

Information about each of the objects is displayed in a popup when you click on them.

### **2.2.4. Adding visualization controls**

Visualization controls are added – zoom in, zoom out and screen rotation. This is implemented by using `map.addControl` and adding new `mapboxgl.NavigationControl` [17].

### **2.2.5. Adding Geocoder**

Mapbox's geocoding API allows geocoding to be made, which means that a text request such as typing Sofia, Bulgaria will become longitude and latitude coordinates. The map is shifted, and these geographical coordinates become its center.

The Mapbox geocoder has a built-in ranking, which affects what results are returned and in what order. Geocoding API requests can also be made even more specific by including additional request parameters.

In this case the `mapbox-gl-geocoder` plugin is loaded and after that a `MapboxGeocoder` object is defined from Mapbox Geocoding API to enable places search [18].

### **2.2.6. Adding Geolocation**

In order for the application to receive data from the location of the device on which it is open, a button is added that the user can press to share his current location and see it marked on the map with a blue dot.

`GeolocateControl` is used to add a button that allows access to the browser geolocation API. When it gets the access, it changes the appearance of the map by zoom into that location [19].

### **2.2.7. Planning routes and getting direction instructions**

The Mapbox Directions API [20] allows users to plan routes, get direction instructions, calculate the travel distance and time. With this API you can calculate the optimal route for driving, walking or cycling. An option for calculating a walking route has been added to the application.

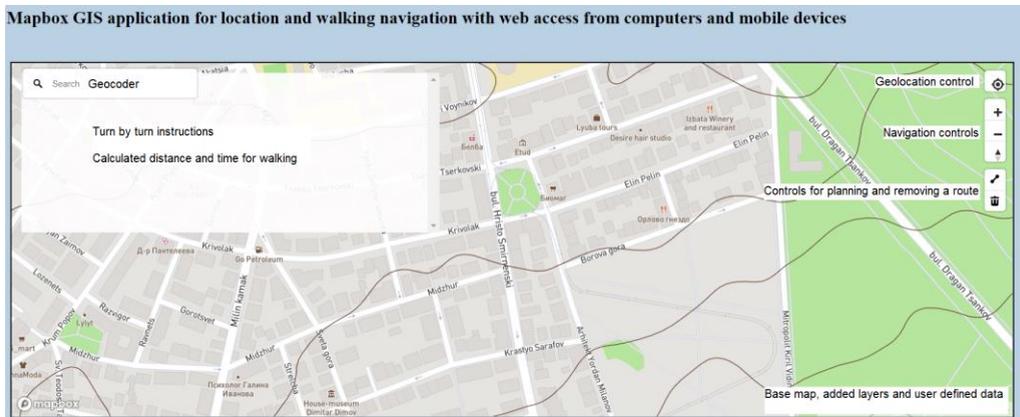
The control for drawing a line was added first, which can be set by connecting up to 25 points. This drawn line is turned into a route and also information on how to get from the first end of the route to the other is provided by using turn by turn directions and coordinate guidance information – these instructions could be copied and sent to another user. Information is given about the length of the route and the approximate time for walking in minutes [21 – 23].

Control for removing the planned route is also added.

### 3. User interface. Performed tests and results

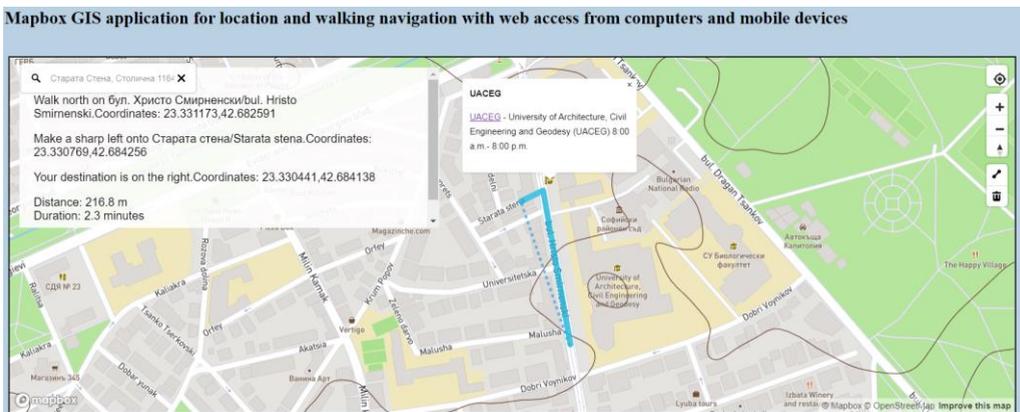
Here the user interface of the application and how it works on a computer and mobile device are presented.

Figure 1 shows the user interface of the application in desktop version with the labeled elements that were added for the developed functionalities described in section 2.



**Figure 1. User interface of the application in desktop version with labeled elements for the developed functionalities**

Figure 2 shows how the Geocoder works on search request for the Starata Stena street. The planned route with turn-by-turn direction instructions is added, the travel distance and time are shown. Also, the added user defined data for the UACEG is shown in popup.



**Figure 2. User interface of the application in desktop version – geocoder, planned route with turn-by-turn direction instructions, travel distance and time, user defined data in popup**

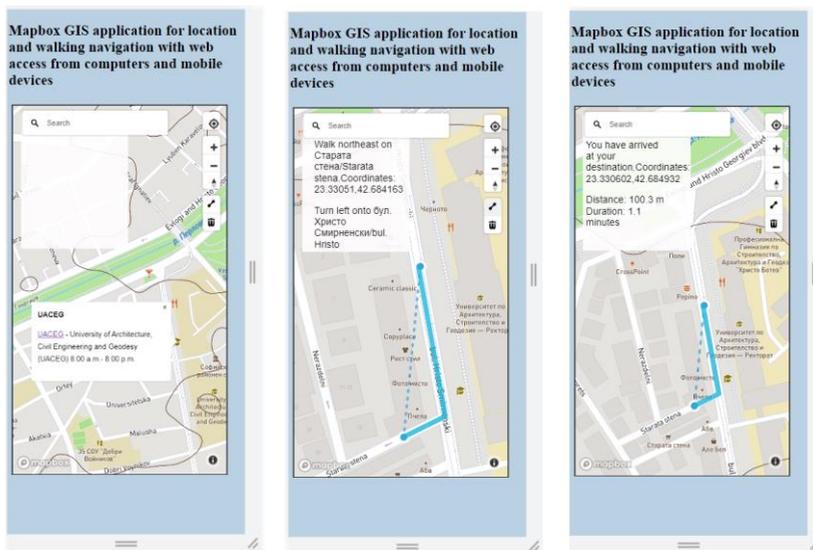
The turn-by-turn instructions with coordinate guidance information which are shown on Figure 2 could be copied and sent to another user – this is presented in the next example:

Walk north on бул. Христо Смирненски/bul. Hristo Smirnenski.  
Coordinates: 23.331173,42.682591  
Make a sharp left onto Старата стена/Starata stena.

Coordinates: 23.330769,42.684256  
Your destination is on the right.  
Coordinates: 23.330441,42.684138  
Distance: 216.8 m  
Duration: 2.3 minutes

Figure 3 shows responsive desktop views for mobile device. The added user defined data for the UACEG is shown again in popup and a planned route with turn-by-turn direction instructions, the travel distance and time are calculated.

Figure 4 shows some of the developed functionalities in responsive mobile views – the screen rotation, geolocation, planned route with turn-by-turn direction instructions, calculated travel distance and time



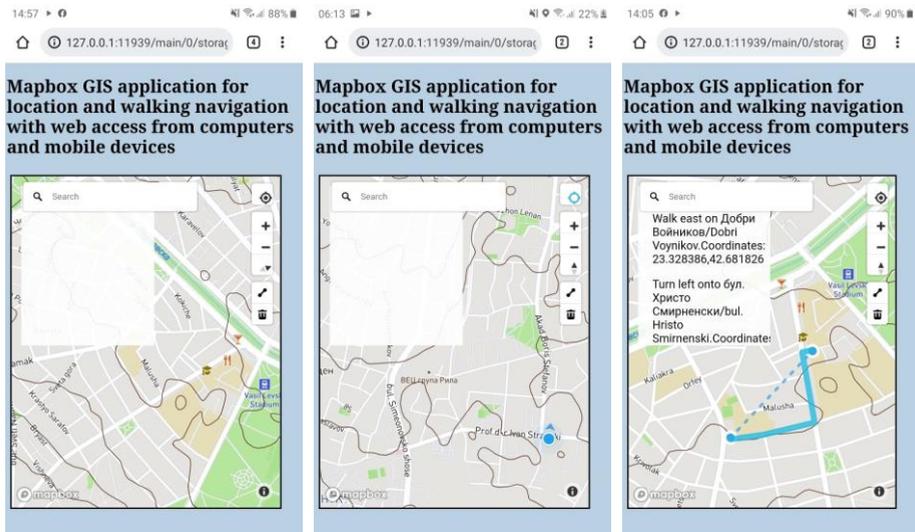
**Figure 3. Responsive desktop views – user defined data in popup, planned route with turn-by-turn direction instructions, calculated travel distance and time**

All of the developed functionalities are tested on desktop version from laptop, on responsive desktop views and mobile phone. As it could be also seen from the examples given above, the user interface is accessible and all developed functionalities of the internet-based application seem to work fine.

It is important to note here that for the time being the developed application works only online. In case of disconnection from the Internet, the content that is loaded in the application remains active until the next reload, and meanwhile the geolocation is performed with data from the internal built-in GNSS receiver of the device.

Mapbox has client-side cache management APIs for Web applications, giving developers more control over caching – there are methods that clear a user's cached tiles on-demand, determine when map data and tileset updates take effect.

By default, the ambient cache in Maps SDKs is instantiated at 50MB, but on mobile devices, the APIs also include a method that allows resizing the ambient cache – to increase the cache size so that more map resources are cached and retrieved quickly or to decrease it in order to reduce the amount of space that map tiles use on a device [24].



**Figure 4. Responsive mobile views – screen rotation, geolocation, planned route with turn-by-turn direction instructions, calculated travel distance and time**

Mapbox has also an option for using preliminary downloaded offline map pack for defined region, given by its geographic bounds or by the bounds of the currently displayed map [25].

The application currently does not have any developed functionality for caching or using offline data, but this can be added in the future.

## 4. Conclusion

This paper presented the development of Mapbox GIS application for location and pedestrian navigation. Based on what was written above the most important is summarized in the next bullets:

- The design of the application is responsive and could be accessed from both computers and mobile devices via web browser.
- The developed functionalities are in compliance with predefined requirements and the given concept.
- The platform that was used enables all the needed libraries, tools and functions for: loading base map; publishing different data layers from online and inline sources; geocoding; adding visualization, location and navigation controls; displaying and sharing navigation instructions.

This is only an example of how the internet-based GIS could be used for personal location and pedestrian navigation. The development model is open, so the application could be upgraded in the future by adding new data layers, as well as different type of transportation and new functionality to it in order to meet the specific user requirements.

## REFERENCES

1. *Rajendran, V.* Location Based Services: Expected Trends and Technological Advancements, <https://geoawesomeness.com/>, June 13, 2017.  
<https://geoawesomeness.com/expected-trends-technological-advancements-location-based-services/>, last visited on January 30, 2022.
2. *Panko, R.* The Popularity of Google Maps: Trends in Navigation Apps in 2018, <https://themanifest.com/>, July 10, 2018.  
<https://themanifest.com/app-development/trends-navigation-apps>, last visited on January 30, 2022.
3. *Wurmser, Y.* Maps and Navigation Apps Report, <https://www.emarketer.com/>, July 26, 2018.  
<https://www.emarketer.com/content/maps-and-navigation-apps>, last visited on January 30, 2022.
4. Google Maps, <https://www.google.com/maps/about/#/>, last visited on January 30, 2022.
5. OpenStreetMap, <https://www.openstreetmap.org/about>, last visited on January 30, 2022.
6. HERE, <https://www.here.com/>, last visited on January 30, 2022.
7. Mapbox, <https://www.mapbox.com/>, last visited on January 30, 2022.
8. *Anwar, R.* Mobile app development trends 2021-2022, <https://easternpeak.com/>, July 17, 2021.  
<https://easternpeak.com/blog/mobile-app-development-trends/>, last visited on January 30, 2022.
9. EPSG Registry, EPSG:4326 coordinate reference system, <https://epsg.org/home.html>, last visited on January 30, 2022.
10. GeoJSON format for encoding geographic data structures, <https://geojson.org/>, last visited on January 30, 2022.
11. *Butler et al.* GeoJSON, Standards Track, RFC 7946, Item 4. Coordinate Reference System, August 2016.  
<https://datatracker.ietf.org/doc/html/rfc7946>, last visited on January 30, 2022.
12. Mapbox examples: Display a map on a webpage, <https://docs.mapbox.com/mapbox-gl-js/example/simple-map/>, last visited on January 30, 2022.
13. Mapbox examples: Add a vector tile source, <https://docs.mapbox.com/mapbox-gl-js/example/vector-source/>, last visited on January 30, 2022.
14. Mapbox tilesets: Mapbox Terrain v2, <https://docs.mapbox.com/data/tilesets/reference/mapbox-terrain-v2/>, last visited on January 30, 2022.
15. Mapbox examples: Display a popup on click, <https://docs.mapbox.com/mapbox-gl-js/example/popup-on-click/>, last visited on January 30, 2022.
16. Mapbox standard icons, <https://github.com/mapbox/mapbox-gl-styles>, last visited on January 30, 2022.
17. Mapbox examples: Display map navigation controls <https://docs.mapbox.com/mapbox-gl-js/example/navigation/>, last visited on January 30, 2022.

18. Mapbox examples: Add a geocoder  
<https://docs.mapbox.com/mapbox-gl-js/example/mapbox-gl-geocoder/>, last visited on January 30, 2022.
19. Mapbox examples: Locate the user,  
<https://docs.mapbox.com/mapbox-gl-js/example/locate-user/>, last visited on January 30, 2022.
20. Mapbox Directions API,  
<https://docs.mapbox.com/api/navigation/directions/>, last visited on January 30, 2022.
21. Mapbox tutorials: Getting started with the Mapbox Directions API,  
<https://docs.mapbox.com/help/tutorials/getting-started-directions-api/>, last visited on January 30, 2022.
22. Mapbox examples: Display navigation directions,  
<https://docs.mapbox.com/mapbox-gl-js/example/mapbox-gl-directions/>, last visited on January 30, 2022.
23. Mapbox tutorials: Driving directions from one place to another,  
<https://github.com/webeasystem/mapbox-driving-directions-from-one-place-to-another>, last visited on January 30, 2022.
24. Mapbox blog: New cache management controls for Mapbox SDKs,  
<https://blog.mapbox.com/new-cache-management-controls-for-mapbox-sdks-2be0302f9ba>, last visited on February 02, 2022.
25. Mapbox blog: Offline maps for all,  
<https://blog.mapbox.com/offline-maps-for-all-bb0fc51827be>, last visited on February 02, 2022.

## РАЗРАБОТВАНЕ НА МАРВОХ ГИС ПРИЛОЖЕНИЕ ЗА ОПРЕДЕЛЯНЕ НА МЕСТОПОЛОЖЕНИЕ И ПЕШЕХОДНА НАВИГАЦИЯ С УЕБ ДОСТЪП ОТ КОМПЮТРИ И МОБИЛНИ УСТРОЙСТВА

Т. Илиева<sup>1</sup>

*Ключови думи:* Марбох, ГИС приложение, уеб и мобилни ГИС, определяне на местоположение и навигация

### РЕЗЮМЕ

Тази статия представя разработено приложение, което е интернет базирано и използва онлайн и вградени ресурси. Инсталира се само с копиране на една уеб страница и може да работи както на компютър или лаптоп, така и на мобилни устройства. HTML и CSS са използвани за оформлението на страницата, а функционалността се осигурява от JavaScript в комбинация с инструменти за местоположение и навигация в реално време, които са достъпни от платформата Mapbox.

---

<sup>1</sup> Тамара Илиева, гл. ас. д-р инж., кат. „Геодезия и геоинформатика”, УАСГ, бул. „Хр. Смирненски“ № 1, 1046 София, e-mail: Ilieva\_tamara@yahoo.com