

The hypothesis of better accuracy than basic equivalent circuit by new microstrip model usage was proven. The possibility of large table models numerical and circuit approximation was demonstrated. The advanced microstrip X-junction transmission equivalent circuit was designed and tested. The advanced X-junction circuit model is simple enough to be implemented in regular modern circuit CAD/CAM software.

1. Петренко А.И., Кирюша Б.А. Проектирование СВЧ устройств с помощью ALLTED // *Электроника и связь*. –2001. – № 13. – С. 37. 2. Gupta K. C. *Emerging trends in millimeter-wave CAD* // *IEEE Trans. Microwave Theory Tech.*. – Vol. 46, Apr. 1998. –P. 475–483. 3. Rubaiyat Islam. *Theory and Applications of Microstrip / Negative-Refractive-Index Transmission Line (MS/NRI-TL) Coupled-line Couplers*, thesis of Doctor of Philosophy Graduate Department of The Edward S. Rogers Sr. Department of Electrical and Computer Engineering University of Toronto, 2011. 4. Кирюша Б.А. *Нова модель оптимального повороту на 90 градусів* // *Електроника и связь: сб., тем. Вып. «Проблемы электроники»*. – Ч.2. – 2006. – С.117–120.

UDC 004.8

V. Bondarchuk, S. Tkachenko
Lviv Polytechnic National University,
Computer-Aided Design Department

DEVELOPMENT OF THE SUBSYSTEM OF THE AUTOMATED GENERATION FOR THE PUBLIC TRANSPORT ALTERNATIVE ROUTES WITH CHANGES

© Bondarchuk V., Tkachenko S., 2012

In this paper made the survey modern subsystems of the generation for the public transport routes with changes. Own created subsystem is more functionality and most useful. Geographical coordinates of points on the map are the input information for own subsystem. This coordinates for user input. User should input two or more points on the map. If user will input two points – it will be the start and the end of way and if input more points – it will be way with intermediate points. Public transport routes are model for input data. Model saves in data base. Subsystem result of work shows on the map. If the route goes along the street it is highlighted.

Key words: transport routes, subsystem, Dijkstra's algorithm.

Розглянуто існуючі картографічні підсистеми для генерування маршрутів з пересадками для громадського транспорту. Створено власну підсистему, що має більші функціональні можливості, а тому зручніша для користувача. Вхідною інформацією для підсистеми є географічні координати точок на карті, які задає користувач. Можна або задати дві точки – початок руху та кінець, або вказати проміжні точки, через які має пролягати маршрут. Моделлю для вхідних даних є маршрути громадського транспорту, що зберігаються в базі даних. Результати роботи підсистеми показують в графічному вигляді на карті. Вулиці, якими проходить маршрут, підсвічуються.

Ключові слова: транспортні шляхи, підсистема, алгоритм Дейкстри.

Introduction

Several years ago, to replace paper maps and atlases came electronic or interactive maps that allow you to find any objects in seconds – just type the name of the object, click on the appropriate button, and will appear snippets cards. But not only speed refers to the benefits of interactive maps, they have implemented an interactive viewing information. In general, there are three types of maps – maps created

by flash-technology, as well as two types of GIS maps (based on the technology of geographic information systems) – conventional GIS maps and GIS maps with visualization. The minimum degree of interactivity often characteristic flash-cards and is displaying some information on the selected object by hovering or clicking the mouse.

In general, interactive GIS maps can display not only the city, houses and streets. Many systems have the opportunity to put their own objects on a map and store them on the server or in my database. In particular, we can apply transport links.

Routes public transport has changed in Ukraine since January 1, 2012. This was a big problem for residents. The median resident of the city goes from home to work by making one or more transplants. When a resident of the city goes one route to work is good. However, if a courier, a guest or a tourist city, then find and comprehend at once a huge amount of information to decide which route to go, is not easy. This causes the relevance of this paper.

You can put public transport routes on an interactive map and make it freely available on the Internet, it is much easier life passengers. However, you can perform calculations on them: give the user the opportunity to choose the starting point of the movement, final and intermediate points through which he wants to move. You can also calculate other less profitable routes. This subsystem is simply irreplaceable assistant for people who frequently move around the city by public transport in different directions. With this purpose developed this subsystem. This is done through an online mapping system Google maps, using the tools Google API functions in conjunction with the technology of Web programming JavaEE.

Currently already have similar subsystems, but they allow you to search only routes between two points.

Currently subsystem

At the present time for the cities of Ukraine are two known subsystems that generate the list of routes for public transport. One of them is located at <http://marshrutky.com.ua/>, another – <http://www.eway.in.ua>. Each subsystem we consider separately, their advantages, disadvantages and functionality.

Marshrutky.com.ua subsystem uses data from the online mapping system Google maps. This helps Google maps API functions. Subsystem was developed by PHP technology on the server side and JavaScript, Flash on the client side. Main page of the subsystem shown on fig. 1.



Fig. 1. Main page marshrutky.com.ua subsystem

The database contains public transport routes to cities such as Lviv, Rivne, Kyiv, Khmelnytsky, Vinnitsa, Sumy, Kharkiv, Dnipropetrovsk, Donetsk. Working with subsystem starts with choosing the city for which you need to generate an alternative route. When the city is selected, the user must specify a starting point on the map where you move, and end point – where to finished. After those action subsystem

will show the result. Street name and house number can be input data. Result of the subsystem will be highlight streets, which will be moving buses. If buses (trams, trolleybuses) different routes, then they are displayed in different colors. The results are shown in Fig. 2.

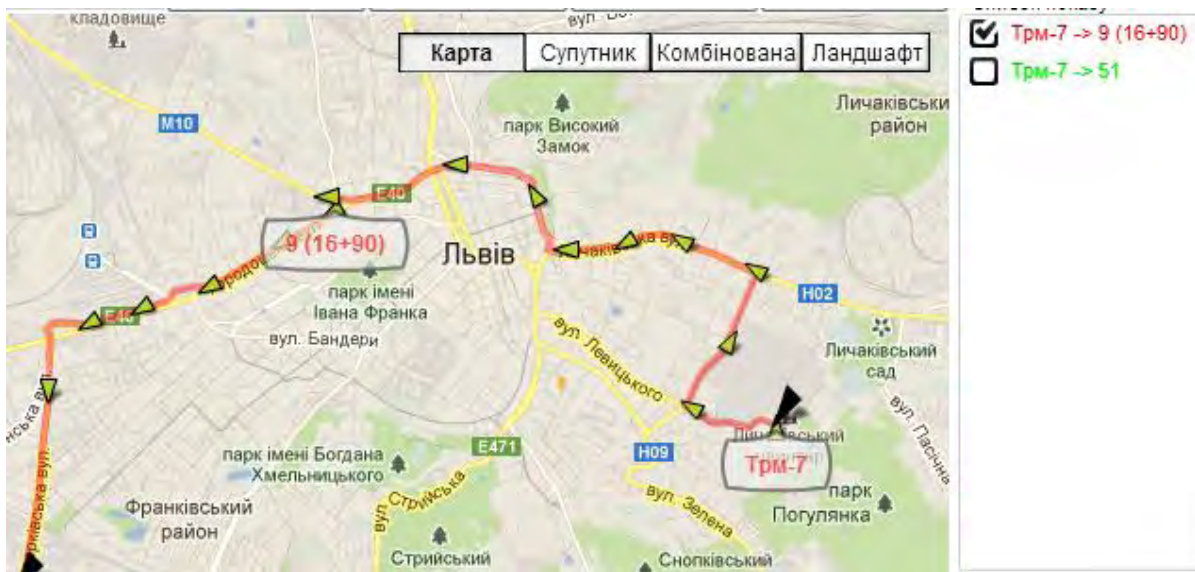


Fig. 2. Working results of the subsystem *marshrutky.com.ua*

Advantages:

- easy to use and friendly user interface;
- user friendly working result;
- no too much information;
- map display in full screen;
- there are stops;
- there are forum for discuss problem of public transport.

Disadvantages:

- you can not dynamically change the start and end points;
- no search criteria;
- you can not enter more than 2 points on the map;
- no internationalization.

Subsystem *eway.in.ua* also uses data from the online mapping system Google maps. subsystem developed using PHP technology on the server side, JavaScript-on the client side. Main page subsystems shown in Fig. 3.

The database contains public transport routes to cities such as Lviv, Lutsk, Rivne, Ternopil, Ivano-Frankivsk, Kyiv, Khmelnytsky, Vinnitsa, Kharkiv, Dnipropetrovsk, Donetsk and others. Working with subsystem starts with choosing the city for which you need to generate an alternative route. When the city is selected, the user must specify a starting point on the map, from which it will move, and end point – where he wants to reach. After specifying the endpoint to choose your search criteria route, type of transport that need to go, and click on "Calculate". Input data can be a street name and house number. The result of the subsystem will highlight street, what will moving buses. Buses of different routes displayed in different colors. The results are shown in Fig. 4.

Advantages:

- easy to use and friendly user interface;
- there are stops;
- search criteria;
- you can select types of public transport;

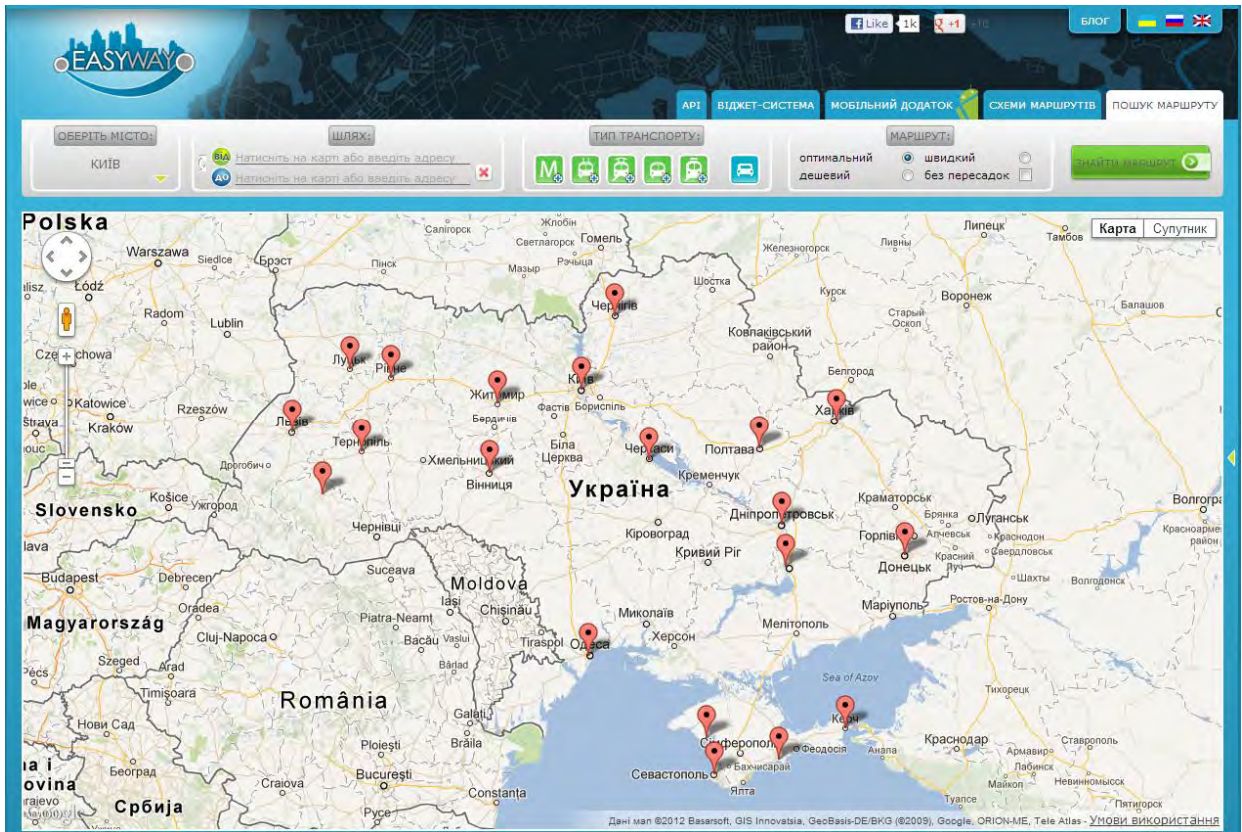


Fig. 3. Main page *eway.in.ua* subsystem

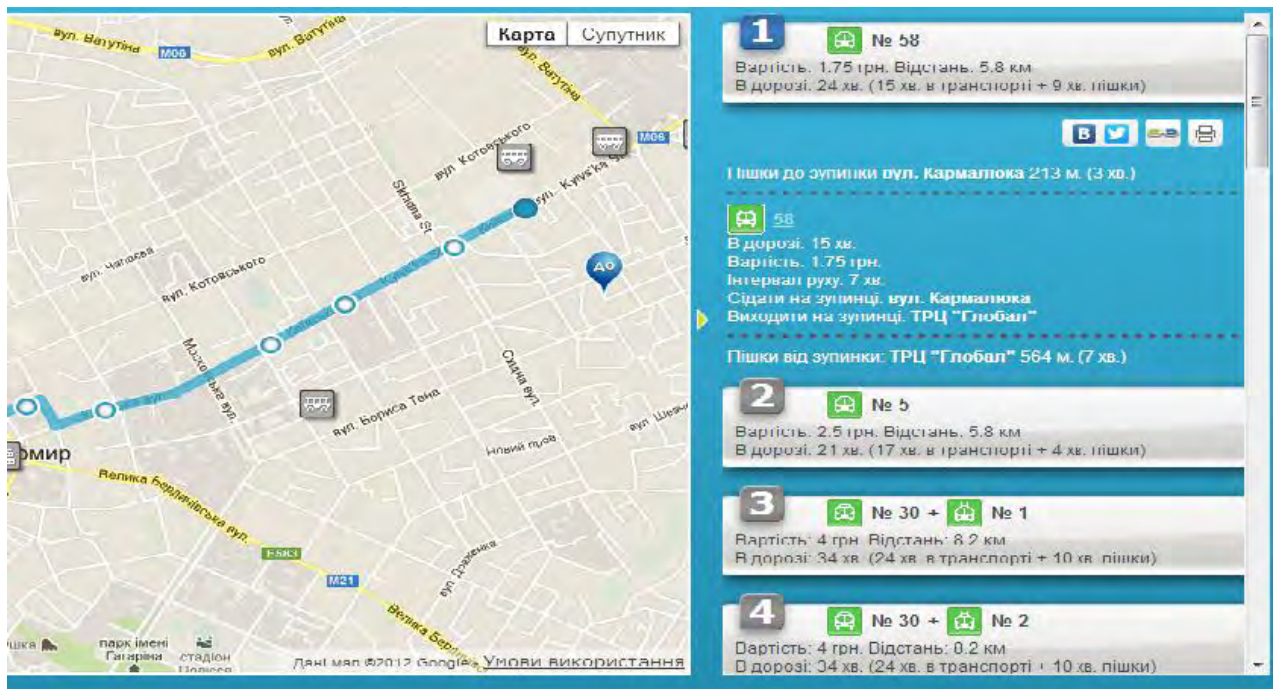


Fig. 4. Working results of the subsystem *eway.in.ua*

Disadvantages:

- you can not enter more than 2 points on the map;
- no full screen map;
- many information in result work subsystem.

Development of the subsystem

Subsystem will use data from the online mapping system Google maps. Subsystem is implemented using JavaEE technologies on the server side and HTML, CSS, JQuery on the client side. Project architecture is schematically shown in Fig. 5.

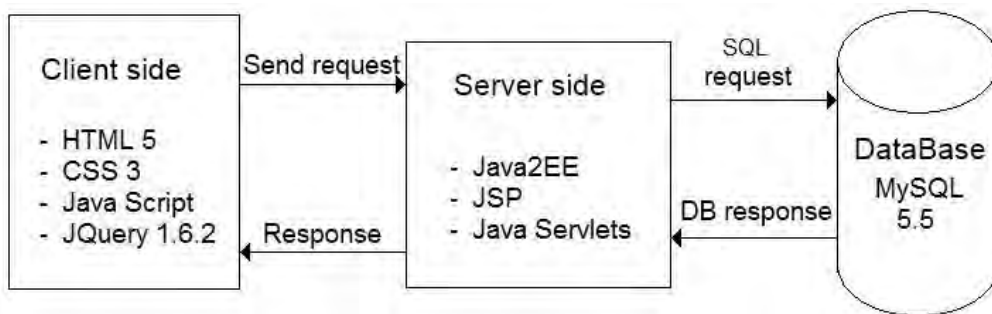


Fig. 5. Project architecture

The principle of this architecture is simple. The client sends a request to the server. The server accepts it, get from the database required data, processes them according to requests received from the client and, after processing, sends them to the client. When the client receives the data, it simply represents them.

Routes will be stored as the coordinates of latitude and longitude. One route would be a 100 to 900 nodes, which are consistently on the street and joined the line. The distance between the points of the route can vary from 10 meters – if the route is drawn on turning the streets – and up to 400 meters – if this is a direct street. Graphically, this is shown in Fig. 6.

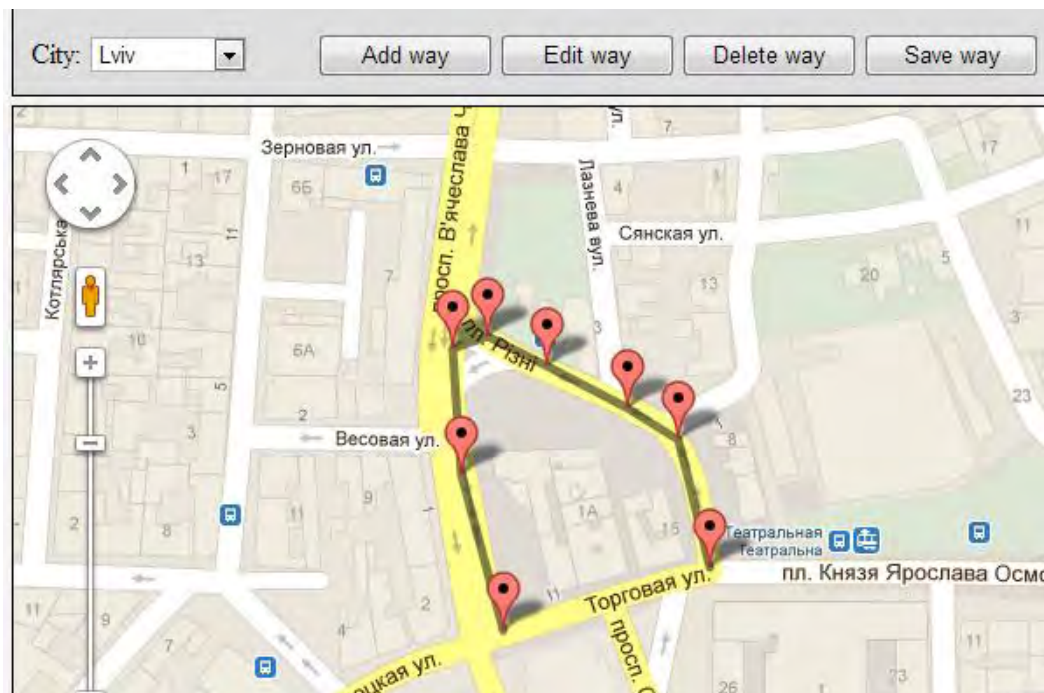


Fig. 6. Creating route for public transport

The algorithm of the subsystem

It may seem like a search for the optimal route for public transport using Dijkstra's algorithm, if you convert all points of all routes in the vertices. However, it is not, because Dijkstra's algorithm works with fixed vertices and in our subsystem vertex will be determined dynamically, depending on the input data set the user [1].

The general principle of the algorithm is that it is necessary to take the starting point, which set the user and find the first five routes that are closest to it. Next, you need to take each found route to see if it leads to the final point. If the leads – leave it otherwise – reject. Those routes that remained to show the user. Where the routes for the first point found, but none of them leads to the final point, it is necessary to leave the ones that are coming to the end point, and find the point in those routes that are left and which is closest to the destination. After this initial point is transferred to the position of the waypoints that are the closest to the final destination. Then search for routes performed again as described at the beginning. Search routes with intermediate points built on finding a way between two points. For example, you must know how to get from point A to point B through point C and D. Initially will search between two points A and B, then between B and C, then between G and B. Steps of algorithm are below.

Step 1. Check whether the input data to the algorithm. If there is, then go to step 2. If no – go to step 7.

Step 2. Find the first 5 routes that are closest, within 250 meters to the starting point. If you find at least one route, then go to step 3. If no route, then go to step 6.

Step 3. Alternately, take each route and see if it leads to the following user-defined point. If even one leads to the next point, then go to step 4. If not is none, then go to step 5.

Step 4. Add in the resulting list of routes. Check whether a point is the last. When is the last, then go to step 8. If not the last one, then go to step 2.

Step 5. Determine the point in a route that is closest to the final. Set the starting point to the place nearest point in the route. Go to step 2.

Step 6. Print a message about "not match any route according to the input data". Shut algorithm.

Step 7. Print the message that the input data is not entered correctly. Shut algorithm.

Step 8. Print a list of routes that can be reached from the starting point at the last through the intermediate point. Shut algorithm.

Working with subsystem will is the same as that of counterparts, except for some differences:

- map will be shown on the full screen, and the results will show and hide in side panels.
- it is possible to set intermediate point on the map.

Conclusions

The problem that exists now with public transport has been shown. Existing subsystem that can partially solve this problem has been shown. The own subsystem was designed with new features. Search algorithm in routing subsystem and technology, by which it is implemented was described.

This subsystem can use guests, tourists, city dwellers, couriers and delivery services. They can easily find the information they need concerning the use of public transport.

1. Bondarchuk V., Tkachenko S. Subsystem of the Automated Generation for the Public Transport Alternative Routes with Changes // Proceedings of the VIIIth International Conference on Perspective Technologies and Methods in MEMS Design. – Lviv: Publisher Lviv Polytechnic, 2012. – 252 p. – P 141.