# A Network Analysis for Finding the Shortest Path in Hospital Information System with GIS and GPS

**Hamid Ali Abed Alasadi[1,a,*], Mohammed Talib Aziz[2,b], Mohammed Dhiya[2,c] and Ahmed Abdulmajed [4,e]**

[1]Department of Communication, Iraq University Collage, Basra, 61004, Iraq.
*a. hamid.alasadi@iuc.edu.iq*

*Abstract:* The Spatial decision support system (SDSS) has turned to be an important aspect of our daily lives for its role in developing decision techniques based on Geographic Information System (GIS). This paper will be presenting a proposed intelligent SDSS mobile application for better health services and faster aid through integrating two types of technology: an improved Dijkstra algorithm that insures faster response and the geographic information system technology. This is done by studying and comparing the A* algorithm and Dijkstra algorithm to arrive at the best Dijkstra algorithm improvement method. The application goal is to serve as a health aid application that will give instruction to avoid the covid-19 virus and provide the required information and help the infected to find the right donors or apply for receiving request by using an online database system for adding and viewing lists of donators and receivers in Basra governorate. The other aim of the application is to seek out the closest hospital, shortest path to the current health centre, time, and distance.

## 1. Introduction

These days transportation is becoming more and more of an issue due to the rapid increase in population and the demand of transportation, and along with this issue, comes the serious problem of emergency fast transportations. Hence, it is only a rationale to find a way to efficient travel from one place to another in an easy way at the shortest time, cheapest cost and, least effort. Obtaining a system that can give fast decisions in real time is now more important than ever. Making the decision about the nearest hospital and the shortest path to it is to be through a system by using an algorithm in graph theory known as Dijkstra's algorithm for the purpose of offering routine services to all those who are in need of it and guide them to the most correct choice of route [1-2].

Among the areas promising new considerable abilities that lead to decision making lies in the utility of spatial information in Geographic Information System (GIS). An interactive the Spatial decision support system (SDSS) computer-based system is designed to help a user or group of users enjoy higher effectiveness of decision making while solving a semi-structured spatial decision problem [3-7].

After the spread covid-19 virus, an issue of general health arise, an issue that this project will be attempted to give instructions to avoid this virus along with information about it and to have an online active database to help the infected to find a possible donor. Many people, whether patients or others, may benefit from this system.

What follows includes an outline of the study. The area of the study and the work stages are discussed in Sections 2 and 3 whereas preparation and data collection are so discussed in Section 4. In Sections 5 and 6 software and algorithms are presented. The product, the results and discussion are presented in sections 7 and 8, respectively. Finally, the conclusion of the work is offered in Section 9.

## 2. Study Area

The study area of hospitals in this research is the region of north Basra. The area of north Basra region is about 108 km2 located between coordinates 30°26' N 47°47' E and 30°33' N 47°47' E. This city contains a number of hospitals providing services for the population. Most of these hospitals are public ones while others are private as shown in table 1 [8-9]. Many regions and streets around this area within Basra are taken to test any location within these very regions and to find the shortest path from this location to all hospitals inside the area.

Table 1: List of Basra Hospitals.

| Hospital Name | The coordinates | Type |
|---|---|---|
| Basra | 47.805831,30.494041 | Public |
| Al-Sadr Teaching | 47.850590,30.507148 | Public |
| Basra Women's and Children's | 47.835348,30.504715 | Public |
| Ports | 47.782781,30.552273 | Public |
| Al-Fayhaa | 47.782787,30.533714 | Public |
| Basra Children's Specialized | 47.797626,30.477518 | Public |
| Al-Shifa | 47.809786,30.537111 | Public |
| Al-Musawi | 47.826417,30.504866 | Private |
| Al-Mouwasat | 47.812760,30.491754 | Private |

## 3. Work stages

Many stages and tasks needed to be performed in the making of this project that will be mentioned in this section. The steps of the work are illustrated in below Figures 1.a, 1.b and 1.c as the Utilization of Geographic Information System, Spatial Decision Support System (SDSS), Graphical User Interface (GUI), respectively.

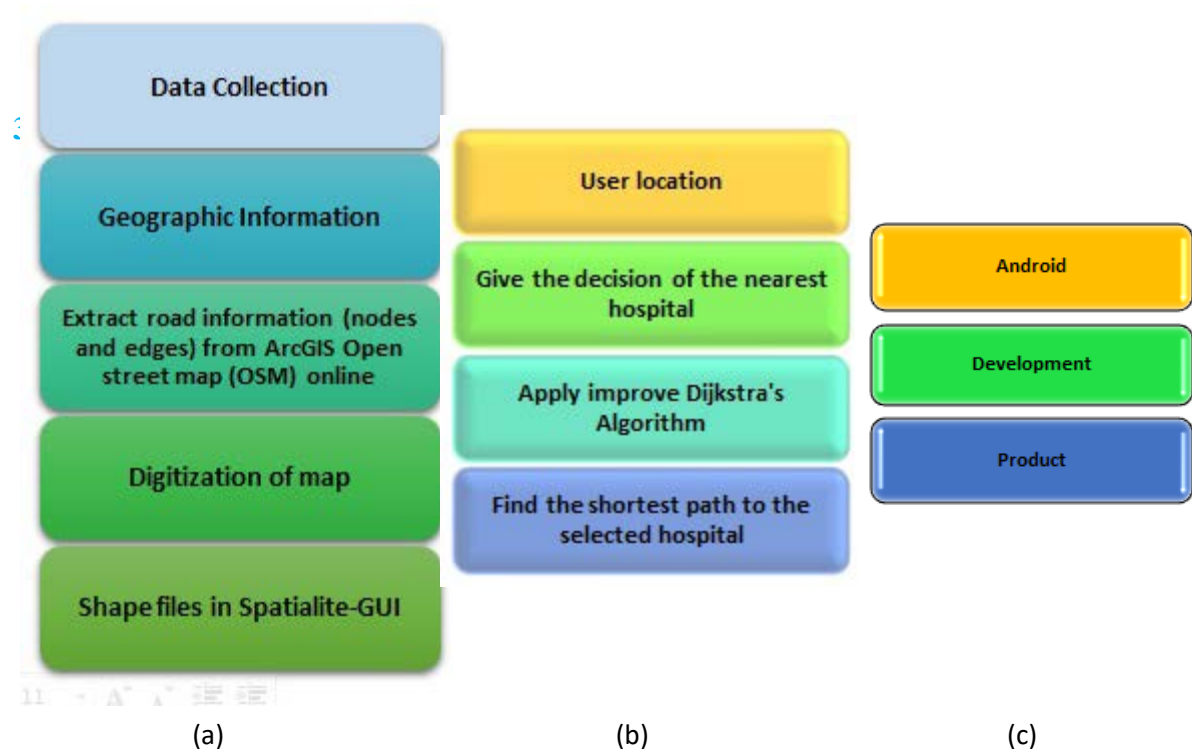|       (a)       |       (b)       |       (c)       |

Figure 1: The work stages.

## 4. Preparations and Data

### 4.1 Preparations

This section will go through the process of obtaining and shaping the format of the desired data along with preparing the software that will be required for this paper.For this project to be completed there are many requirements essential for this work and without them this project won't be possible from the first place [10-23].

- Open Street Map: Open Street Map (OSM) is a project for collaboratively creating a free editable map of the world. The data from such a project can be used in different ways to have paper and electronic maps similar to the Google ones including geocoding of address and place names.The map of the study area is extracted from the OSM free to use data via the QGIS.

- QGIS: is a free open-source cross-platform desktop geographic information system (GIS) application that supports viewing, editing, and analysing geospatial data. There are numerous features and tools that this application can provide. These include the OSM data extractor which can be used to obtain data for further use.

- PostgreSQL: PostgreSQL or Postgres, is a free open-source relational database management system. Using this OSM, one can analyze and create a graphed map out of the processed OSM data.

- Spatialite-GUI: This is a simple tool used for managing databases.

- Android Studio: Android Studio is the official integrated development environment (IDE) for Google's Android operating system. It is built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.

- Java, Xml, PHP and MySQL: Java is a general-purpose programming language, which is class-based, object-oriented, designed to have the fewest implementation dependencies. It is used to allow application developers to write once, run anywhere

meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. XML Extensible Mark-up Language is a language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. PHP is a popular general-purpose scripting language that is especially suited to web development. PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. MySQL is an open source relational database management system.MySQL has stand-alone clients that help users to interact directly with a MySQL database using SQL, but more frequently MySQL is used with other programs to implement applications requiring relational database capability.

- WampServer: refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

- ArcGIS is a geographic information system (GIS) for working with maps and geographic information maintained by the Environmental Systems Research Institute (Esri). It is used for creating and using maps, compiling geographic data, analyzing mapped information, sharing and discovering geographic information, using maps and geographic information in a range of applications, and managing geographic information in a database.

- Nox emulator: It is a free to use android emulator that runs on windows.

## 4.2 Data Collection

What is required for the SDSS system is spatial data such as locations (coordinates latitude and longitude) for road nodes, edges of road network and hospitals. This kind of data can be obtained through the process of digitizing a base map of specific area and provide locations coordinates for the hospitals by GPS by the following steps.

- Step 1: Geographic Information, this part is concerned with all the necessary geographic data and satellite images that are going to be explained just below.
- Step 2: Extracting road information (nodes and edges) from ArcGIS Open street map (OSM) online. The open street map, as far as the term suggests, refers to the data that are free for both kinds of organizations: commercial and public. This is far from being specific. It is rather an open source map which enables users to obtain information of OSM and update or correct the data of the map as shown in figure 2. The map which is to be converted into OSM data and made into the data is shown in figure 3 whereas the data that is needed for the project within the row OSM file data is shown in table 1. The extraction of the OSM file will be done by the QGIS software.

The process Digitization of map will produce the desired data for each node, line, edge, and their exact coordination from the OSM file that has all the information needed but in a format that not yet possible to be used in the upcoming stages.

First is the process of converting the OSM file into a shapefile (SHP), this process take place in the QGIS program that has a tool capable of such process. In this section the main goal is to convert the data that the SHP file contain which is now nothing but lines with a coordination to both start and end points for each line into a usable graph that marks each lines crossing, turning points, edges and also lines for each street along with the length of leach one.

Figure 2: OSM map.



Figure 3: Extracted data.

Table 2: A Sample of the row data.

| | ogc_fid | osm_id | name | highway | waterway | aerialway | barrier | man_made | z_order | other_tags |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4706 | 806479729 | | residential | | | | | 3 | |
| 2 | 4707 | 806480404 | | secondary | | | | | 6 | "oneway"=>"y... |
| 3 | 4704 | 806479727 | | residential | | | | | 3 | |
| 4 | 4705 | 806479728 | | residential | | | | | 3 | |
| 5 | 4708 | 806482273 | | residential | | | | | 3 | |
| 6 | 4690 | 806479713 | | residential | | | | | 3 | |
| 7 | 4691 | 806479714 | | residential | | | | | 3 | |
| 8 | 4688 | 806479711 | | residential | | | | | 3 | |
| 9 | 4689 | 806479712 | | residential | | | | | 3 | |
| 10 | 4694 | 806479717 | | residential | | | | | 3 | |

Using a tool in PostgreSQL program called "osm pgrouting" the row lines in the SHP file can be processed into the desired network of the streets. this data now is separated into two tables in one data base, as shown in the figure 3 and table 1 is the map data before processing, in the table 2, 3 and figure 4 is the nodes and lines needed in one database separated into two tables, (27250) line and (18726) node.

Table 3: Lines.

| | length | length_m | name | source | target | source_osm | target_osm |
|---|---|---|---|---|---|---|---|
| | Filter | Filter | Filter | Filter | Filter | Filter | Filter |
| 1 | 0.0005450062... | 57.106313185... | NULL | 7041 | 275 | 4026434958 | 610933712 |
| 2 | 0.0009410668... | 95.645028780... | NULL | 7169 | 1 | 6058095968 | 4886594079 |
| 3 | 0.0005814208... | 60.534622658... | NULL | 5055 | 1 | 4886594076 | 4886594079 |
| 4 | 0.0002974122... | 29.824059123... | NULL | 3337 | 2 | 6553530661 | 6553530660 |
| 5 | 0.0006108027... | 63.184424023... | NULL | 4334 | 3 | 4141779129 | 609220599 |
| 6 | 0.0007800817... | 80.900042802... | NULL | 2934 | 3 | 609220594 | 609220599 |
| 7 | 0.0010711797... | 112.91380325... | الشارع التجاري | 5222 | 4 | 1873241738 | 610955609 |
| 8 | 0.0006170862... | 61.782460832... | NULL | 1422 | 5 | 4886594158 | 610975027 |
| 9 | 0.0004492071... | 45.868293340... | NULL | 1423 | 6 | 4148805999 | 4148806003 |
| 10 | 0.0004667213... | 46.681716124... | NULL | 4530 | 7 | 4149911013 | 5801978822 |

Table 4: Points.

| | id | osm_id | eout | lon | lat |
|---|---|---|---|---|---|
| | Filter | Filter | Filter | Filter | Filter |
| 1 | 1 | 4886594079.0 | NULL | 47.813015 | 30.5355977 |
| 2 | 2 | 6553530660.0 | NULL | 47.7840728 | 30.5190313 |
| 3 | 3 | 609220599.0 | NULL | 47.7572202 | 30.5429645 |
| 4 | 4 | 610955609.0 | NULL | 47.8024236 | 30.5244874 |
| 5 | 5 | 610975027.0 | NULL | 47.8071972 | 30.5378722 |
| 6 | 6 | 4148806003.0 | NULL | 47.767845 | 30.524457 |
| 7 | 7 | 5801978822.0 | NULL | 47.7556354 | 30.5205305 |
| 8 | 8 | 611465042.0 | NULL | 47.7974555 | 30.5107775 |
| 9 | 9 | 608237203.0 | NULL | 47.8084093 | 30.5173565 |
| 10 | 10 | 610941641.0 | NULL | 47.8000179 | 30.5271187 |

Figure 4: Map after processing.

Using the Spatialite-GUI to modify and manipulate the two tables in the database that produced by PostgreSQL to shape them into a final database that contain the finalized finished data that can be used now directly in the application and with the algorithms.

## 5. Algorithms

This section will go through the progression of planning the implementing of the algorithms that will take a place in this project. This is mainly used to find the shortest path extending from a starting point or node to the end node in a weighted graph. When Dijkstra's algorithm is applied, it creates a tree of shortest path from a starting source to all the other nodes in the graph that is connected to it directly. And as shown in figure 5 from the source point the Dijkstra's algorithm will start comparing the weighted between every node connected to the start node and chose the lowest weight and then repeat the process in the new progressed node and check for the total weight each time for reassessment. In case the path that has been taken is discovered not to be the best, one step back is required to the last node where the path is the lowest. Then the process is repeated from there without including the false angulated path.
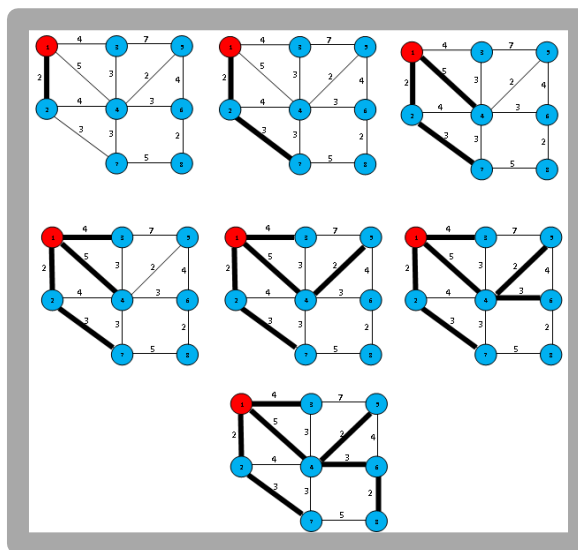


Figure 5: Dijkstra's algorithm.

Though the Dijkstra's Algorithm seems very efficient, it can actually take a long time routing the path from start to end in a huge array of data that will be used as shown in figure 6, and that is not acceptable at all, so to reduce the time a solution has to be made.
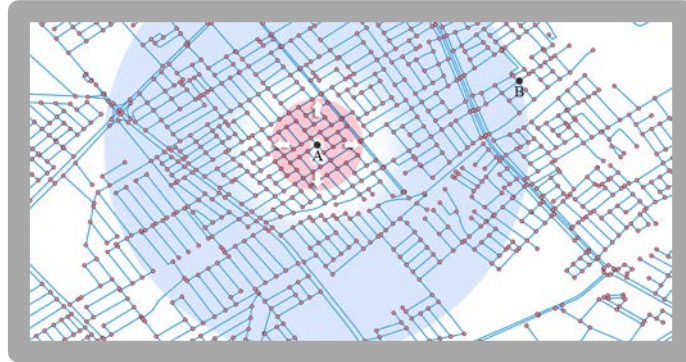


Figure 6: Dijkstra's progression.

The red circle is the progression of the Dijkstra's algorithm and the blue one indicates the actual graph nodes and lines the algorithm has to process before calculating the correct root from A to B.

The improved algorithm will potentially reduce the search area significantly as shown in the figure 7. This is only a fraction of the area in the previous figure. It will allow much faster calculations thus reducing the time required for the routing process to come up with a valid path. The improving method is basically creating a temporary data set before initiating the Dijkstra's algorithm itself and treating it as if it were the graph that the algorithm has to work with. This data set will be created after acquiring the start and end nodes. With the coordinates from the start and finish nodes the data set can be excluded from the main data by selecting only the nodes that located inside the square created by the two nodes.
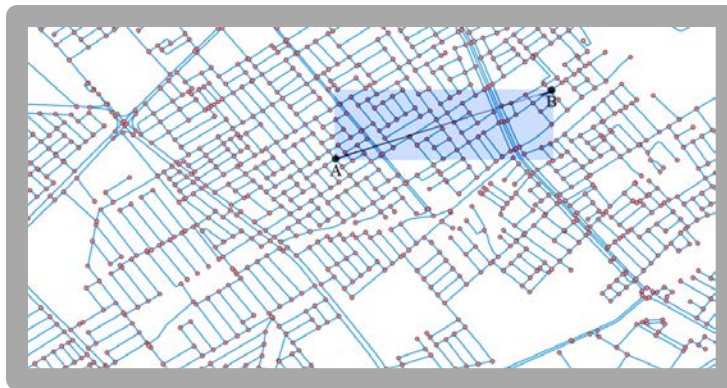


Figure 7: Improved search area.

The Dijkstra's algorithm is not the only solution when it comes to finding the shortest path in a weighted graph. There are some alternative algorithms that may or may not come to the same solution but at a faster time.

The A* algorithm is one of the best search algorithms here [24-25]. It can be even faster than the Dijkstra's algorithm since it has a feature the Dijkstra's has not and that is why it can tell if the node is closer to or farther from the destination node. The unimproved Dijkstra's algorithm has to search the entire area in order to come to a solution as shown in figure 6 the A* algorithm requires more data in order to be executed. This is shown in figure 8, and the data represented in the figure by the red numbers is the distance from each node to the destination node, this distance will be used and

added to the weights when comparing the different nodes and ways so the farther it gets when calculating, the greater the weight will be. This will reduce the search time in large graphs significantly, however, there is an issue with the new needed values and it is that they should be created first, unlike the Dijkstra's algorithm which goes through the graph with only the nodes and the weights. However, this issue can be easily avoided by only creating the needed values when it needed along the execution and not previous to it, so instead of measuring the entire graph it will only measure a small fraction of it, thus reducing the required time for creating it considerably.
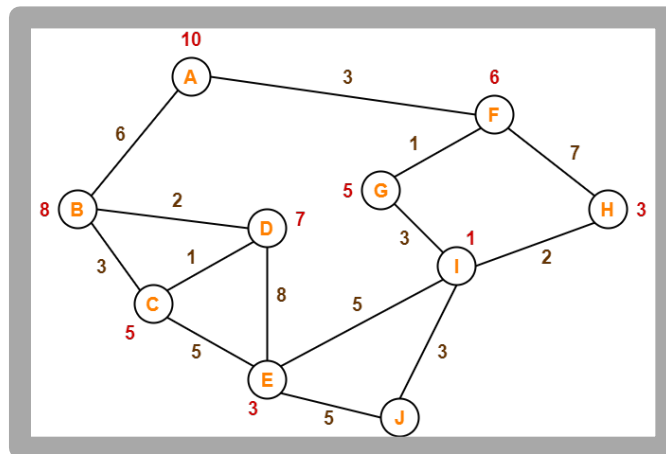


Figure 8: A* algorithm.

## 6. The Product

This is the final stage and the one when all the parts come together to form the project that has been intended. All the previous stages are worthless unless it is implemented in an application that is easily accessible by common user which is the mine objective for this entire project form the first place.

Using android studio to create a Java based app is fairly simple. However, creating the actual map that will be shown to the user is a bit more complex, since the project will be using ArcGIS maps that will show the open street map. This step will require the ArcGIS API which can be easily implemented in the project after getting and implementing the correct development kit which is free on their website for developer use only. After creating the map there are few things that the API offers that will be helpful in this project, such as giving the exact coordination for the user location and updating it in real time, the second tool that will be used is drawing a line using coordination points  after creating the correct path.

The product in the final form in which this project will be shaped and used, the APK file (The install package) will be exported from the android studio as the ready to use application. After installing the application into an android operating system and running the app a message box will be shown regarding the Covid-19 vireos as shown in the figure 9. The main interface will be represented as a map of the user location, and shown on it the path that shall be taken, as shown in figure 10.

The red lines represent the shortest path while the red lines represent the improved search area (The blue lines will not be shown in the actual app but they are available only in the beta version). In the bottom right corner of the shown the estimated travel time and the name of the hospital.
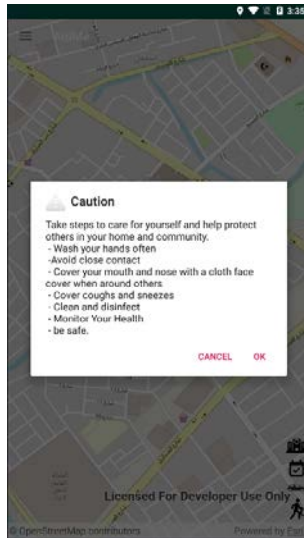
Figure 9: Preview 1.          Figure 10: Preview 2.

In the top left corner of the screen is the slide bar icon. This slide bar as shown in figure 11 is activated by sliding the left side of the screen inward or by tapping on the icon. Hence, the bar will include four options, the first one will lunch the contact application in the mobile in order for the user to call someone, while the second one will dial the emergency hotline for Basra. The Donor option will open a list of the available donors, and the same is for the receiver option as shown in figure 12. A list of donors or receivers will be displayed. As shown on the top of the frame there is a register button that will launch the registration form as shown in figures 13 and 14.
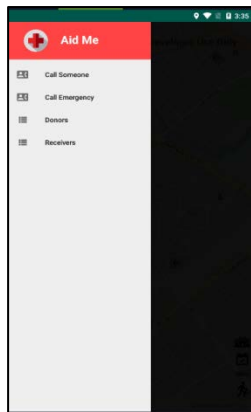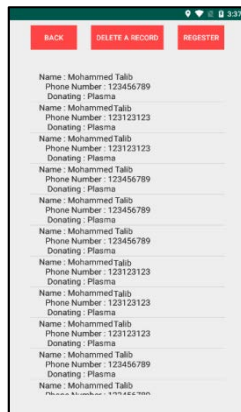





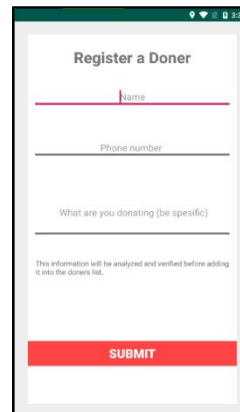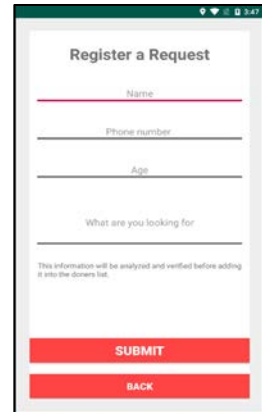Figure 11: Preview 3.     Figure 12: Preview 4.     Figure 13: Preview 5.     Figure 13: Preview 6.

After filling the form and submitting it a passcode will be provided as shown in figure 14. The removing will require the given passcode, after tapping on the delete a record button shown in figure 14. The frame shown in figures 15, 16 and 17 will appear and after inserting the correct passcode, a message will appear confirming the removing of the record, in case the passcode is wrong (does not exist in the database), a message will appear.
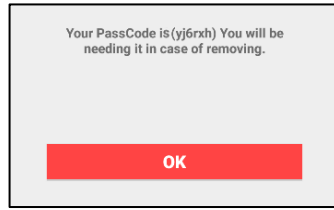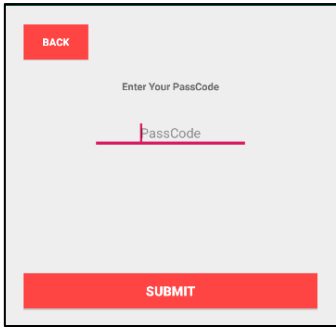
Figure 14: Preview 7.
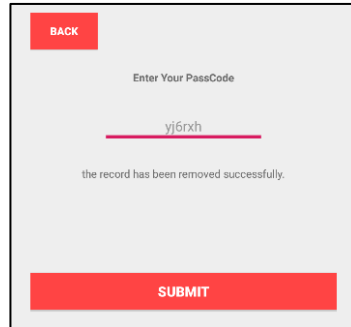


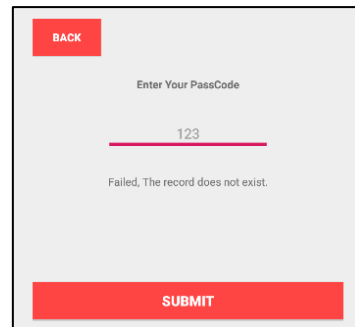Figure 15: Preview 8.          Figure 16: Preview 9.          Figure 17: Preview 10.

## 7.   Results and Discussion

The output of this system will mainly be represented as the distance (in km) of the calculated path. Using Google maps to have a comparison between this project results and the distances collected from Google maps after using the same coordination in the testing process.

Table 5: The Results.

| Initial location | Hospital name | Dijkstra | Google map | Accuracy % |
|---|---|---|---|---|
| Longitude 47.790752 Latitude 30.549495 | Ports | 1143 m | 1.1 km | 100 |
| | | 1.1 km | | |
| | Al-Fayhaa | 3021 m | 3.2 km | 94.406 |
| | | 3 km | | |
| | Basra Children's Specialized | 9493 m | 10.9 km | 86.238 |
| | | 9.4 km | | |
| | Al-Shifa | 4276 m | 5.5 km | 76.363 |
| | | 4.2 km | | |
| | Basra | 10357 m | 9.6 km | 92.690 |
| | | 10.3 km | | |
| | Al-Sadr Teaching | 9647 m | 9.5 km | 98.958 |
| | | 9.6 km | | |
| | Basra Women's and Children's | 9329 m | 8.9 km | 95.698 |
| | | 9.3 km | | |
| | Al-Musawi | 7351 m | 7.1 km | 97.260 |
| | | 7.3 km | | |
| | Al-Mouwasat | 8930 m | 9.8 km | 90.816 |
| | | 8.9 km | | |

| | | Average Accuracy | 92.492 |
|---|---|---|---|

Since the only variable in this case is the time, the comparison is carried out after measuring the time of that the unimproved Dijkstra algorithm, and that of the improved Dijkstra algorithm and the A* algorithm takes to calculate the correct path is shown in table 6. This measurement was made by using an android built in library that measures the code execution time.

Table 6:  Comparison.

| Initial location | Hospital name | Distance | unimproved | improved | A* |
|---|---|---|---|---|---|
| **Longitude 47.790752** **Latitude 30.549495** | Ports | 1.1 km | 0.48 sec | 0.33 sec | 0.33 sec |
| | Al-Fayhaa | 3 km | 1.17 sec | 0.95 sec | 0.94 sec |
| | Basra Children's Specialized | 9.4 km | 3.22sec | 2.81 sec | 2.17sec |
| | Al-Shifa | 4.2 km | 1.96 sec | 1.28 sec | 1.21 sec |
| | Basra | 10.3 km | 4.29 sec | 3.02 sec | 2.41 sec |
| | Al-Sadr Teaching | 9.6 km | 3.88 sec | 2.94 sec | 2.58 sec |
| | Basra Women's and Children's | 9.3 km | 3.69 sec | 2.77 sec | 2.19 sec |
| | Al-Musawi | 7.3 km | 2.79 sec | 2.14 sec | 2.02 sec |
| | Al-Mouwasat | 2.67 km | 3.37 sec | 2.68 sec | 2.47 sec |
| Average Time | | | 2.76 sec | 2.10 sec | 1.81 sec |

As shown in the table, the time was the highest with the unimproved algorithm.   the time of the improved Dijkstra could not match the A* and that was due to the time taken for calculating and adding of the distance between each edge and the final destination. As there are many factors contributing to this system, and most important of them all is the shape of the required map data. The format in which the data is stored can significantly improve the way this system behaves making it faster and more efficient, however in order to achieve this data format there are more tools that should be used and data to be collected which can't be easily or freely accessed.

In addition to the low process power of the mobile devices compared to the more professional extremely high and different online servers used by most routing services to handle the execution of the process. However, the difference in process powers will not be an issue when using the best data formatting and making any mobile device as efferent as any routing servers.

## 8.  Conclusion

Since the only difference between the two algorithms is the time taken to finish and after the use and application of the two algorithms, the conclusion was that even if the A* algorithm was faster to calculate the correct path the Dijkstra algorithm with a little improvement could behave and enhance the speed of execution making it close to that of the A*algorithm. However, with the improving method itself comes a very small delay that with the repetition process can be noticeable and driving the margin of time between the two algorithms slightly greater.

This application which will be free to download and use on the part of any number of users in the area that the project covers, these users can use their mobile apps when needed to  get the closest hospital immediately along with the best possible root that will offer the fastest travel time.

Hence, it is suggested that the accuracy is not as good as Google maps are. That is why more improvement can be done to the data that are used with this application in order to make it faster and more accurate for the purpose of determining the correct valid path.

## References

[1] Schulz, F., Wagner, D., Weihe, K.: Dijkstra's algorithm on-line: An empirical case study from public railroad transport. ACM Journal of Experimental Algorithmics 5 (2000)

[2] Ravi, N., Sireesha, V. Using Modified Dijkstra's Algorithm for Critical Path Method in a Project Network. International  Journal of Computational and Applied Mathematics. Volume 5 number 2 pp 217-225. 2010.

[3] M.P. Johnson, "A spatial decision support system prototype for housing mobility program planning," Springer Science & Business Media B.V., 2001, pp. 49.

[4] R. L. Church and A.T. Murray, "Business Site Selection, Location Analysis, and GIS". New York: John Wiley & Sons, 2009, pp.20-22 .

[5] Goodchild, M.F. 2005. "GIS and modelling overview". In GIS, spatial analysis, and modelling, Edited by: Maguire, D.J., Batty, M. and Goodchild, M.F. 1–17. Redlands: ESRI Press.

[6] Laudien, R. and Bareth, G. 2007. Developing and programming Spatial Decision Support Systems with Java and ArcGIS Engine (ESRI) (in German). GIS, 04/2007: 16–21.  [Google Scholar]

[7] Laudien , R. , Röhrig , J. , Bareth , G. and Menz , G. , 2007 . Spatial Decision Support System zur Modellierung der agrarischen Marginalität in Benin (Westafrika) . Proc. AGIT'2007 . Heidelberg : Wichmann .

[8] Al-Asadi SAR, Al Hawash AB, Alkhlifa NH, Ghalib HB (2019) Factors affecting the levels of toxic metals in the Shatt Al-Arab River, Southern Iraq. Earth Syst Environ.

[9] Al-Tememi MK, Hussein MA, Khaleefa UQ, Ghalib HB, Al-Mayah AM, Ruhmah AJ (2015) The Salts diffusion between East Hammar marsh area and Shatt Al-Arab River Northern Basra City. Marsh Bull 10(1):36–45

[10] Maier, Gunther (2014). "OpenStreetMap, the Wikipedia Map". Region. 1 (1): R3–R10. doi:10.18335/region. v1i1.70.

[11] "QGIS Official Website". QGIS. Retrieved 16 October 2013.

[12] "Project name – statement from the core team". archives.postgresql.org. November 16, 2007. Retrieved November 16, 2007.

[13] Ducrohet, Xavier; Norbye, Tor; Chou, Katherine (May 15, 2013). "Android Studio: An IDE built for Android". Android Developers Blog. Google. Retrieved May 16, 2013.

[14] "Write once, run anywhere?". Computer Weekly. May 2, 2002. Retrieved July 27, 2009.

[15] "1.2 Design Goals of the Java™ Programming Language". Oracle. January 1, 1999. Archived from the original on January 23, 2013. Retrieved January 14, 2013.

[16] XML 1.0 Specification". World Wide Web Consortium. Retrieved 22 August 2010.

[17] "PHP: Hypertext Preprocessor". www.php.net. Retrieved 2020-02-12.

[18] "Introduction: What can PHP do?". PHP Manual. Retrieved 2009-03-05.

*[19] "What is MySQL?". MySQL 8.0 Reference Manual. Oracle Corporation. Retrieved 3 April 2020. The official way to pronounce "MySQL" is "My Ess Que Ell" (not "my sequel"), but we do not mind if you pronounce it as "my sequel" or in some other localized way.*

*[20] Sobel, Jason (21 December 2007). "Keeping Up". The Facebook Blog. Facebook. Archived from the original on 18 June 2009. Retrieved 18 June 2009. Facebook's data is stored in MySQL database servers.*

*[21] Susan Perschke (23 April 2012). "WampServer delivers a smart, Windows-friendly platform for Apache, MySQL and PHP-based apps". Network World. Retrieved 20 January 2015.*

*[22] Serdar Yegulalp (30 May 2012). "Review: WAMP stacks for Web developers". InfoWorld. Retrieved 20 January 2015.*

*[23] "ArcGIS Desktop". Retrieved 28 July 2018.*

*[24] Saleh Alija Amani, Analysis of Dijkstra's and A\* algorithm to find the shortest path, 2015.*

*[25] Reddy Harika, "PATH FINDING-Dijkstra's and A\* Algorithm's",* International Journal in IT and Engineering, *pp. 1-15, 2013.*