

**IMPLEMENTATION OF THE DIJKSTRA METHOD CHOOSING
THE SHORTEST PATH OF ISLAMIC TOURISM
MALANG CITY**

UNDERGRADUATE THESIS

**Created by:
YUSUF ABDI HIDIGOW
NIM. 14650111**



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019
IMPLEMENTATION OF THE DIJKSTRA METHOD CHOOSING
THE SHORTEST PATH OF ISLAMIC TOURISM
MALANG CITY**

UNDERGRADUATE THESIS

**A Thesis is submitted to
Fakultas Sains dan Teknologi,
Universitas Islam Negeri Maulana (UIN)
Malik Ibrahim Malang
for the Requirements for the Degree of
Bachelor of Computer (S.Kom)**

**Created by:
YUSUF ABDI HIDIGOW
NIM. 14650111**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019**

APPROVAL PAGE

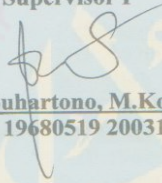
IMPLEMENTATION OF THE DIJKSTRA METHOD
CHOOSING THE SHORTEST PATH OF ISLAMIC
TOURISM MALANG CITY

UNDERGRADUATE THESIS

Created by:
YUSUF ABDI HIDIGOW
NIM. 14650111

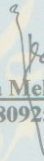
Has been reviewed and approved for the examination
Date ; 17 september 2019

Supervisor I



Dr. Suhartono, M.Kom
NIP. 19680519 200312 1 001

Supervisor II




Roro Inda Melani, M.T, M.Sc
NIP. 19780925 200501 2 008

Know,

Head of Department Information Technology
Faculty of Science and Technology
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Cahyo Crysdian
NIP. 1974024 200901 1 008

CONFIRMATION PAGE

IMPLEMENTATION OF THE DIJKSTRA METHOD CHOOSING
THE SHORTEST PATH OF ISLAMIC TOURISM
MALANG CITY

UNDERGRADUATE THESIS

Created By:

YUSUFABDI HIDIGOW
NIM. 14650111

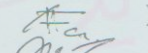
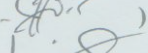


Has been conducted in the presence of the board of undergraduate
thesis examiners and stated accepted as one of the requirements to
obtain the title Bachelor of Computer (S.Kom)

Date: 27 September 2019

Board of Examiner

1. Main Examiner : Fatchurrohman M.Kom
NIP. 19700731 200501 1 002
2. Chief Examiner : Khadijah Fahmi Hayati Holle M.Kom
NIDT. 19900626 20160801 2 077
3. Secretary Examiner : Dr.Suhartono.M.Kom
NIP.196805192003121001
4. Member Examiner : Roro Inda Melani.M.T.M.Sc
NIP. 197809252005012008

Signature

()
()
()
()

Authorize,
Head of Department of Information Technology
Faculty of Science and Technology
Universitas Islam Negeri Maulana Malik Ibrahim Malang


Dr. Carlo Crysdian
NIP. 1974024 200901 1 008

LETTER OF STATEMENT

The undersigned below:

Name : Yusuf Abdi Hidigow
NIM : 14650111
Department : Teknik Informatika
Faculty : Sains dan Teknologi

States that an undergraduate thesis mentioned above is my original work except for the quotations and statements whose resources are acknowledged on the references. Any shortcoming in this present work, therefore, are entirely my own responsibility.

Moreover, this work is not plagiarism result and if it is found that this statement is false, my academic records will attest to invalidation and I will be responsible for that as well.

Malang, October 2019

Signed by,



Yusuf Abdi Hidigow
NIM. 14650111

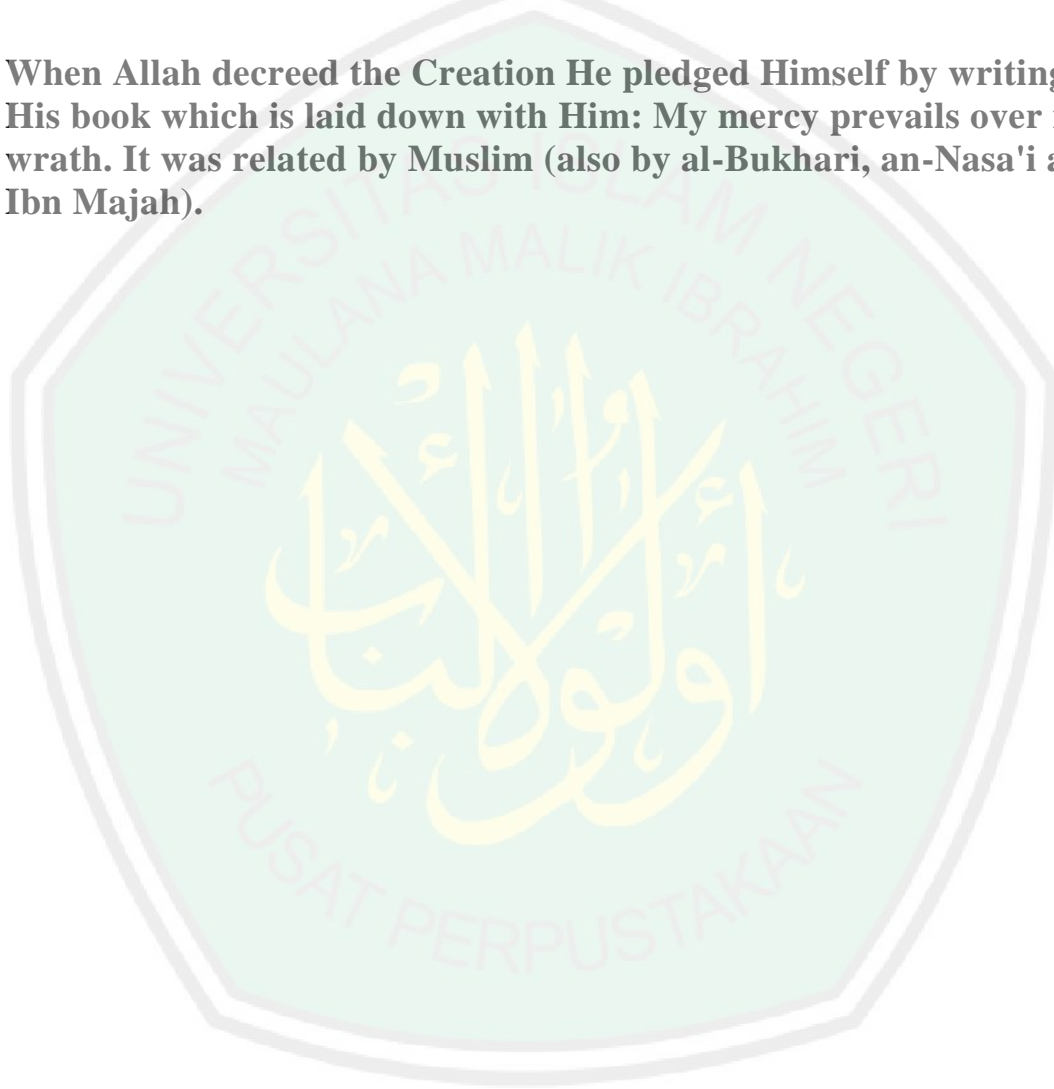
MOTIVATION

:عَنْ أَبِي هُرَيْرَةَ قَالَ: قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ

"لَمَّا قَضَى اللَّهُ الْخُلُقَ، كَتَبَ فِي كِتَابِهِ عَلَى نَفْسِهِ، فَهُوَ مَوْضُوعٌ عِنْدَهُ: إِنَّ رَحْمَتِي تَغْلِبُ غَضَبِي "

(رواه مسلم (وكذلك البخاري والنسائي وابن ماجه

When Allah decreed the Creation He pledged Himself by writing in His book which is laid down with Him: My mercy prevails over my wrath. It was related by Muslim (also by al-Bukhari, an-Nasa'i and Ibn Majah).



OFFERINGS PAGE

In This Chance I Want To Thank :

My Family :

AlhamduLillah thanks to the prayers, support and motivation of all sides, the pole can pass and get what is the dream of me. Hopefully it can be blessing and useful.

My Lecturers :

Thank you for all the knowledge and upbringing that has been given so far, as well as patience in guiding me. Hopefully blessing and become a charity for all the lecturers.

My Classmates :

Teknik Infomatika Angkatan 2015 All Students Thank you for encouraging and accompanying me in joy and sorrow, hopefully blessings and the benefits of their knowledge.

My Organizational Friends:

foreign friends especially (Somali brothers) i hope all of them hopeful and success life thanks all of them to help and supporting me in every aspect Lots of knowledge, experience and benefits that I receive from them. Hopefully it can be blessing and all the benefits.

PREFACE

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamu'alaikum Wr. Wb.

Praise be to God, the researchers pray for the presence of Allah SWT who bestows all His mercy and gifts, as well as prayers and greetings, hopefully they are still bestowed upon the Prophet Muhammad, who has brought guidance to all humanity, namely Islam. So this thesis can be completed with the title : **“IMPLEMENTATION OF THE DIJKSTRA METHOD CHOOSING THE SHORTEST PATH OF ISLAMIC TOURISMMALANG CITY, ”**

With all humility, the author realizes that in completing this thesis can not be separated from the role of various parties who have provided much support, motivation, guidance, and guidance. On this occasion the authors would like to thank:

1. Dr. Cahyo Crys dian, M.CS, as Chairman of the Maulana Malik Ibrahim State Islamic Informatics Engineering Department Malang.
2. Dr. Suhartono,M.Kom, as the Supervisor I, who has provided guidance, input, and direction in the process of completing this thesis.
3. Roro Inda Melani,M.T,M.Sc ,as the Supervisor II, who has provided guidance, input, and direction in the process of completing this thesis.
4. My Parents and all the family who always pray for me and provide moral and spiritual support. provide motivation and prayer that always accompanies so that this thesis is finished.
5. All foreign friends , who always give encouragement and encouragement to quickly finish this thesis.
6. Friends of all who have helped in the completion of this thesis.
7. Informatics Engineering Generation 2014 All Students who have given motivation and enthusiasm so that this thesis can be completed.
8. foreign friends especially (Somali brothers) who have helped in the process of completing a thesis

Nothing is perfect except for Allah SWT, for all the shortcomings and imperfections, the authors realize that the completion of this thesis there are still deficiencies. Hopefully this thesis will provide many benefits.

Wassalamu'alaikum Wr. Wb.

Malang, 27 September 2019

Author



TABLE OF CONTENTS

TITLE PAGE	i
SUBMISSION PAGE	ii
APPROVAL PAGE	iii
CONFIRMATION PAGE	iv
STATEMENT PAGE	v
MOTO PAGE	vi
OFFERINGS PAGE	vii
PREFACE PAGE	viii
TABLE OF CONTENTS	x
LIST OF FIGURES	xiii
LIST OF TABLES	xvi
ABSTRAC	xvii
ABSTRAK	xviii
ملخص	xix
CHAPTER I INTRODUCTION	1
1.1 Background of research	1
1.2 Formulation of the problems	2
1.3 Research question	3
1.4 Objective of research	4
1.5 Benefit of research	4
1.6 Scope of problem	4
CHAPTER II LITERATURE REVIEW	5
2.1 Decision support system (DSS)	5
2.2 Dijkstra algorithm	6
2.3 Design of system	8
2.3.1 Basic principles of design	8
2.3.2 System design approach	9
2.4 Php programming	10
2.5 Mysql	11
2.6 Islamic tourism	12
2.7 Criteria Islamic Tourism	13
2.8 Related research	14

CHAPTER III DESIGN IMPLEMENTATION	17
3.1 Material of research.....	17
3.1.1 Primary data.....	17
3.1.2 Secondary data.....	17
3.2 Tool of research.....	17
3.2.1 Hardware.....	17
3.2.2 Software.....	18
3.3 Flowchart system.....	18
3.4 Planning of system	19
3.4.1 Dad context.....	19
3.4.2 Dad level 0.....	20
3.4.3 Entity relationship diagram.....	22
3.4.4 Database.....	22
3.5 Case of study	23
3.6 Design interface.....	27
CHAPTER IV TRIAL AND DISCUSSION	32
4.1 System implementation	32
4.1.1 Hardware.....	32
4.1.2 Software.....	32
4.2 Interface implementation.....	32
4.2.1 Login page	32
4.2.2 Region Page.....	35
4.2.3 Object page	37
4.2.4 Node page	42
4.2.5 Tourist route page	46
4.3 Discussion.....	51
4.3.1 Test login page.....	51
4.3.2 Test region Page	52
4.3.3 Test add of region page.....	53
4.3.4 Test update of region Page	54
4.3.5 Test delete of region Page.....	55
4.3.6 Test object page	56
4.3.7 Test add of object page	57
4.3.8 Test update of object page	58
4.3.9 Test delete of object Page	59
4.3.10 Test node page	60
4.3.11 Test add of node page	61
4.3.12 Test update of node page	62
4.3.13 Test delete of node page	63

4.3.14 Test route page.....	64
4.3.15 Test result of route page	65
4.4 Analysis of results	65
CHAPTER V CONCLUSIONS AND SUGGESTIONS	67
5.1 Conclusions	67
5.2 Suggestions.....	68
REFERENCES	69



LIST OF FIGURES

Figure 3.1 Flowchart of shortest distance determination system	19
Figure 3.2 Dad context	20
Figure 3.3 Dad level 0	21
Figure 3.4 Entity relationship diagram	22
Figure 3.2 Add distance weight of nodes	22
Figure 3.3 Retrieval of neighboring node 1	24
Figure 3.4 Choose minimal distance weight of node 2	25
Figure 3.4 Retrieval of neighboring end node	25
Figure 3.5 Retrieval of other neighboring node 1	26
Figure 3.6 Retrieval of neighboring node 3	26
Figure 3.7 Retrieval of neighboring node 4	27
Figure 3.8 Retrieval of neighboring end node	27
Figure 3.10 Form login admin	28
Picture 3.11 Form data city	28
Figure 3.11 Form input data city	29
Figure 3.12 Form data object	29
Figure 3.13 Form input data object	30
Figure 3.14 Form data node	30
Figure 3.15 Form input data node	31
Figure 3.16 Form report favourite route	31
Figure 3.17 Form dashboard traveler	32
Figure 3.18 Form direction result	32
Figure 4.1 login page	33
Figure 4.2 Line code login page	34
Figure 4.3 Line code authenticate user	34
Figure 4.4 Dashboard page	35
Figure 4.5 Region page	36
Figure 4.6 Line code display data region	37
Figure 4.7 Line code add region	37

Figure 4.8 Line code update region	38
Figure 4.9 Line code delete region	38
Figure 4.10 Object page.....	39
Figure 4.11 Line code display data object.....	41
Figure 4.12 Add new object page.....	42
Figure 4.13 Line code add object	43
Figure 4.14 Update object page.....	44
Figure 4.15 Line code update object	44
Figure 4.16 Line code delete object.....	44
Figure 4.17 Node page.....	45
Figure 4.18 Line code display data node	46
Figure 4.19 Add new node page	47
Figure 4.20 Line code add node	48
Figure 4.21 Update node page.....	48
Figure 4.22 Line code update node	49
Figure 4.23 Line code delete node.....	50
Figure 4.24 Route page.....	51
Figure 4.25 Line code display option route.....	52
Figure 4.26 Line code progress route	53
Figure 4.27 Result page	54
Figure 4.28 Line code result progress route	55

LIST OF TABLES

Table 2.2 Data type of mysql.....	14
Table 3.1 Data city or region	22
Table 3.2 Data object.....	22
Table 3.3 Data node.....	23
Table 4.1 Testing for login lage.....	51
Table 4.2 Testing for region page.....	52
Table 4.3 Testing for add of region page.....	53
Table 4.4 Testing for update of region page.....	54
Table 4.5 Testing for delete of region page.....	55
Table 4.6 Testing for object page	56
Table 4.7 Testing for add of object page	57
Table 4.8 Testing for update of object Page.....	58
Table 4.9 Testing for delete of object Page	59
Table 4.10 Testing for node Page.....	60
Table 4.11 Testing for add of node Page.....	61
Table 4.12 Testing for update of node page	62
Table 4.13 Testing for delete of node page	63
Table 4.14 Testing for route page.....	64
Table 4.15 Testing for result of route page	65
Table 4.16 Testing data.....	66
Table 4.17 Trial result.....	66

ABSTRAC

Yusuf Abdi Hidigow 2019. Implementation Of The Dijkstra Method Choosing The Shortest Path Of Islamic Tourism Malang City. Thesis, Study Program Technic Information, Faculty Of Science And Technology, Universitas Negeri Maulana Malik Ibrahim Malang.

Supervisor I: Dr. Suhartono, M.Kom, Supervisor I: Roro Inda Melani, M.T, M.Sc

Keywords: Implementation of dijkstra method, choosing the shortest path, Islamic tourism

Malang City is one of the favorite city as a tourist destination. With many tourism places scattered throughout Malang in this case in particular is a tour with Islamic theme. It is a problem for newcomers and tourists in choosing the right route and place of residence. Based on the problem, the author will try to create the shortest route search system for Islamic tourism Malang which is expected to provide an efficient route, and equipped with Google map that contains a map of shelter location.

The algorithm used to find the shortest route to the Islamic tourism destination of Malang is Dijkstra's algorithm. Dijkstra found the shortest route based on the smallest distance from one shelter to another. The Input of the system is the initial shelter and shelter, while the output of the system is the shortest route/path from the initial shelter to the shelter. Distance between shelters obtained from Google Maps by entering the coordinates of the initial shelter and neighbors. From the results of the test case that has been made taken 3 tourist attractions as the tourist destination data, namely the mosque, the Masji Jami ' and Museum Brawijaya. Based on where the Tujun was taken 4 places as the point of Origin namely Arjosari, Blimbing, Kepanjen, Turen. With the result of the route from Arjosari stop to the mosque with a distance of 33km, from blimbing to Masjin Jami ' with distance of 6km, from the Dampit to the mosque, with a distance of 12 km, from Turen to the Museum of Brawijaya with a distance of 27km and from the long to the Masjin, with a distance of 27km. These results indicate that the results of this system are the same as the manually searched results, this means that the system successfully implements Dijkstra's algorithm to find the shortest route of Islamic tourism destinations Malang. In addition to displaying the shortest route, the system can also display alternate routes taken from alternate tables. The system also has maps from Google Maps to display location.

ABSTRAK

Yusuf Abdi Hidigow 2019. Implementasi Metode Dijkstra Memilih Jalur Terpendek Pariwisata Islam Kota Malang. Skripsi, Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing I: Dr. Suhartono, M.Kom, Pembimbing II: Roro Inda Melani, M.T, M.Sc.

Kata Kunci: Penggunaan metode dijkstra, memilih jalur terpendek, wisata Islami

Kota Malang merupakan salah satu kota favorit sebagai destinasi wisatawan. Dengan banyaknya tempat Pariwisata yang tersebar di seluruh Malang dalam hal ini khususnya adalah wisata dengan tema Islami. Hal tersebut yang menjadi masalah bagi pendatang baru dan wisatawan dalam memilih rute dan tempat tinggal yang tepat. Berdasarkan permasalahan tersebut, penulis akan mencoba membuat sistem pencarian rute terpendek untuk Pariwisata Islami Malang yang diharapkan dapat memberikan rute yang efisien, dan dilengkapi dengan peta Google yang berisi peta lokasi tempat berteduh.

Algoritma yang digunakan untuk mencari rute terpendek destinasi Pariwisata Islami Malang adalah algoritma Dijkstra. Dijkstra menemukan rute terpendek berdasarkan jarak terkecil dari satu tempat penampungan ke yang lain. Input dari sistem adalah tempat penampungan awal dan tempat penampungan, sedangkan output dari sistem adalah rute / jalur terpendek dari tempat penampungan awal menuju tempat penampungan. Jarak antara shelter yang diperoleh dari Google Maps dengan memasukkan koordinat awal shelter dan tetangga.

Dari hasil percobaan kasus yang telah dibuat diambil 3 tempat wisata sebagai data tujuan wisata yakni Masjid Tiban, Masjid Jami' dan Museum Brawijaya. Berdasarkan tempat tujuan tersebut diambil 4 tempat sebagai titik asal yakni Arjosari, Blimbing, Kepanjen, Turen. Dengan hasil rute dari Arjosari ke Masjid Tiban dengan jarak 33km, dari Blimbing ke Masjid Jami' dengan jarak 6km, dari Dampit ke Masjid Tiban dengan jarak 12 km, dari Turen ke Museum Brawijaya dengan jarak 27km dan dari Kepanjen ke Masjid Tiban dengan jarak 27km. Dari hasil tersebut menunjukkan dapat dilihat bahwa hasil sistem ini sama dengan hasil pencarian yang dilakukan secara manual, ini berarti sistem tersebut berhasil mengimplementasikan algoritma Dijkstra untuk mencari rute terpendek destinasi Pariwisata Islami Malang. Selain menampilkan rute terpendek, sistem ini juga dapat menampilkan rute alternatif yang diambil dari tabel alternatif. Sistem ini juga memiliki peta dari Google Maps untuk menampilkan lokasi destinasi Pariwisata Islami yang tersebar di seluruh Malang.

الملخص

مالانج يوسف عدي هيدجو . 2019 تنفيذ طريقة ديكسترا في اختيار أقصر طريق للسياحة الإسلامية في مالانج . أطروحة ، برنامج دراسة هندسة المعلوماتية ، كلية العلوم والتكنولوجيا ، جامعة مولانا مالك إبراهيم الحكومية الإسلامية في مالانج.

المستشار الأول: د. سوهارتونو ، م. كوم ، المشرف الثاني: رورو إندا ميلاني ، ماجستير ، ماجستير

الكلمات المفتاحية: استخدام طريقة ديكسترا ، واختيار أقصر طريق ، السياحة الإسلامية

Malang هي واحدة من المدن المفضلة كوجهة سياحية . مع وجود العديد من الأماكن السياحية المنتشرة في جميع أنحاء مالانج في هذه الحالة على وجه التحديد هي جولة ذات موضوع إسلامي . هذه مشكلة للقادمين الجدد والسائحين في اختيار الطريق الصحيح ومكان العيش . بناءً على هذه المشكلات ، سيجادل المؤلف إنشاء أقصر نظام للبحث عن السياحة الإسلامية في مالانج والذي من المتوقع أن يوفر طريقاً فعالاً ، ومجهزاً بخريطة google تحتوي على خريطة لموقع الملجأ . الخوارزمية المستخدمة للعثور على أقصر طريق إلى وجهات مالانج للسياحة الإسلامية هي خوارزمية ديكسترا . وجدت ديكسترا أقصر طريق يعتمد على أصغر مسافة من مأوى إلى آخر . المدخلات من النظام هي الملجأ الأولي والمأوى ، في حين أن إخراج النظام هو المسار / أقصر الطرق من الملجأ الأولي إلى الملجأ . المسافة بين الملاجئ التي تم الحصول عليها من خرائط جوجل عن طريق إدخال الإحداثيات الأولية للملاجئ والجيران . من نتائج القضايا التجريبية التي تم إجراؤها ، تم أخذ 3 مناطق سياحية كبيانات عن الوجهات السياحية وهي مسجد تيبان ومسجى جامي ومتحف براويجايا . بناءً على الوجهة ، يتم اعتبار 4 أماكن هي الأصل ، وهي Arjosari و Blimbing و Kepanjen و Turen . مع نتائج الطريق من Arjosari إلى المسجد Tibib على مسافة 33 كم ، من Blimbing إلى 'Masjin Jami على مسافة 6 كم ، من دامبيت إلى المسجد Tibib على مسافة 12 كم ، من تركيا إلى متحف على مسافة 27 كم ومن Kepanjeng إلى Masjin Tiban . من هذه النتائج ، يمكن ملاحظة أن نتائج هذا النظام هي نفسها نتائج عمليات البحث التي يتم إجراؤها يدوياً ، وهذا يعني أن النظام نجح في تنفيذ خوارزمية ديكسترا للعثور على أقصر طريق إلى وجهات السياحة الإسلامية في مالانج . بالإضافة إلى عرض أقصر طريق ، يمكن لهذا النظام أيضاً عرض طرق بديلة مأخوذة من جدول بديل . يحتوي هذا النظام أيضاً على خريطة من خرائط Google لعرض موقع وجهات السياحة الإسلامية المنتشرة في جميع أنحاء مالانج .

CHAPTER I

INTRODUCTION

1.1. Background of Research

Information is something that is needed by certain individuals and the general public. Information is also a need for certain parties in decision making. Information is data that has been processed into a form that is meaningful to users, which is useful in making current decisions or supporting information sources (Kusrini, 2007). An information must be presented properly and can be justified. An information system in the selection of the shortest tourist route functions as a tool as well as information media for travellers to determine transportation routes so that the traveller can take the intended tourist destination quickly. To fulfil this subject, it is necessary to have an accurate and clear route information to help the traveller's needs. The impact of the current bottleneck on tourism is where many people turn and will look for new solutions to meet their fun needs (Luru, 2017).

Malang City, which is one of the cities in East Java province, is one of the favourite tourist destinations visited by travel lovers. Tourism presented is in the form of natural, educational and religious tourism which is very attractive to visitors. Located in a cool highland area 90 kilometers south of Surabaya, old buildings inherited from colonialism are deliberately maintained so that they become a hallmark of Malang (Merinda, 2016). Malang as one of the biggest cities is often visited by travellers almost every year, especially if it coincides with national holidays visitors flock to visit tourist attractions that are there. The most influential

climate in the city is flanked by mountains in the east and west. It is the second largest city in East Java, located at an altitude between 440 - 667 masl and surrounded by mountains such as Arjuno, Kawi, Tengger and Kelud (Harisuseno, 2017). Besides being famous as a city of tourism because of its charming nature and cool air, Malang is also a city of art because of the many unique arts of this city. According to the International Agency Intrix, Malang City is one of the easiest cities in the world. Local residents consider congestion in the city already included in the severe category. This congestion also eliminates a sense of comfort for tourists visiting.

1.2. Formulation of the Problems

Tourist visits are an important asset that greatly influences income for the local service. Tourism as one of the drivers of the economy is proven to be able to contribute to the prosperity of a country. Tourism development can stimulate business activities to produce significant social, cultural and economic benefits (Rai, 2014). Maximum service can increase the number of visits, including the participation of local agencies in overcoming traffic lanes. Increasing the role of the local government in the form of actions in planning, implementing, monitoring and controlling impacts on national tourism development (Liga, 2013). Many tourism activities depend on transportation and communication. Distance and time factors greatly influence the desire of people to travel. The main function of transportation is closely related to "accessibility" where the frequency of its use, the speed it has can cause long distances as if to be closer. This means shortening the time and will

certainly reduce the cost of travel. Thus transportation can make it easier for people to visit a certain area. Due to the occurrence of these problems there is a need for an information system media solution that can be useful for tourists visiting to be able to choose the fastest route to the desired tourist location, so the author tries to create a fastest path to religious tourism destinations in malang city which is expected to overcome the problem.

Dijkstra's algorithm is an algorithm used to solve the shortest distance problem for a directed graph. This algorithm was published in 195 in the Numerische Mathematic journal entitled "A Note on Two Problems in Connexion with Graphs" and is considered a greedy algorithm. The Dijkstra algorithm works by making a path to an optimal node at each step. So in step n, there are at least n nodes that we already know the shortest path. In this study the author will try to implement the Dijkstra method in building an information system that can be useful for tourists to determine the fastest path to religious tourism destinations in malang city.

1.3. Research Question

Based on the background of the above explanation, then the problem can be formulated as follows

1. How to design the shortest route search system on Islamic tourism in Malang.
2. How to implement the dijkstra algorithm on searching the shortest route in Islamic tourism in Malang.

1.4. Objective of Research

Building an information system that can be useful to help tourists visit religious tourism sites in Malang to determine the fastest traffic route by implementing the Dijkstra method.

1.5. Benefit of Research

It is hoped that this research can provide several benefits including

Determine the fastest transportation route to the religious tourism area in Malang City. Can be a reference for travel route information for tourists in making tour trips to the city.

1.6. Scope of Problem

There are several limitations to the problem in this study including:

- a. Data used is data on the names of districts in Malang along with the distance.
- b. Tourism object used is in the form of religious tourism objects in the city and district of Malang.
- c. Travel route objects that are taken and passed are limited to public transport roads, not small roads or breakthroughs.
- d. Programming languages will used in this study is the PHP langu and MySQL database.

CHAPTER II

LITERATURE REVIEW

2.1. Decision Support System (DSS)

DSS is a combination of models and data in solving semi-structured and unstructured problems involving users or users, where no one knows for sure how these decisions should be made (Turban, 2001). SPK(system pendukung keputusan) can be seen as an achievement or a philosophy rather than an appropriate methodology (Volonio, 2010). From some understanding, SPK can be interpreted, namely a system that is able to solve problems correctly and precisely, which aims to help decision making in choosing one of the various alternative decisions that are the result of processing information data.

At this time along with the development of increasingly advanced information technology where the need for accurate and effective information in one example of the selection of travel routes as one medium of information that is very useful for the traveller who will visit a tourist spot. The situation of congested traffic jams that often occurs in almost all major cities and to large cities, especially if it coincides with a weekend or a national holiday. Of course this can affect the interest of the community to travel. As we know that people in general are more likely to remain silent in their homes when they have time off. In this case the author tries to try to do a research to implement a method of dijkstra in determining the shortest distance to religious tourism places in Malang city and regency so that it can be a reference and information that is very useful for the

traveller in traveling or for the local transportation office is tasked with focusing on regulating solid transportation traffic.

2.2. Dijkstra Algorithm

Dijkstra algorithm is an algorithm method for determining the shortest path in a weighted non-directed graph without explicitly identifying all possible trajectories (Marsudi, 2016). This algorithm is used to determine the shortest path between point pairs in a graph. The dijkstra algorithm works by making a path to an optimal node at each step. So that at the n - th step, there are at least n nodes that we already know as the shortest path. The steps in the dijkstra method are as follows:

- Determine the point that will be the initial node (n), then weight (w) the distance on the first node (n_1) to the closest node (n_x) one by one
- Give the distance weight value (d) for each point to another point, then set the value 0 to the initial node with infinite value to other nodes that have not been filled.
- Attach all nodes that have not been traversed and set the initial node as the departure node.
- From the departure node, consider the closest node that has not been traversed and calculate the distance from the departure point. If this distance is smaller than the previous distance, then delete the old data then save the distance data with the new distance.
- Mark the nodes that have been traversed as nodes passed. The node passed will

never be checked again, the distance saved is the last distance and the smallest weight.

- Set the node has not been skipped by weighing the smallest distance as the next departure node.
- All values of each weighting ($w_{u,v}$) distance between the departure node and the arrival node are known with

$$w(u,v) = n(u) \rightarrow n(v) \dots\dots\dots(2.1)$$

Where u = departure node, v = arrival node

There are several main notations that need to be considered in the work of a Dijkstra algorithm.

- Initialization process, formation of an array or set of N with s members

$$N = \{ s \} \dots\dots\dots(2.2)$$

Where s = source node

At this initialization stage the D_j value (the distance in the table results between node s and node j , with j not equal to s) is entered in the actual value.

If it is not connected directly it will be deemed undefined.

For distances D_s must be 0

$$D_j = C_{sj} \quad j \neq s \quad D_s = 0 \dots\dots\dots(2.3)$$

Where D = distance that will be available in the algorithm results table

C = distance value on the available map.

- Find a neighbour node (node other source), for looping each line entered node i which is not included in the array or set N for later node i be used as an

extension of node s , with node i being the neighbouring node of node s

$$D_i = \min D_j \dots \dots \dots (2.4).$$

- An update for each j is a set of N , for each node j updated the smallest value is comparing the previous D_j value with the sum ($D_i + C_{ij}$), which is the sum of the distance nodes s to node i with the actual distance from node i to node j

$$D_j = \min \{ D_j, D_i + C_{ij} \} \dots \dots \dots (2.5)$$

2.3. Design of System

Defined about as drawing, planning, and making sketches or arrangements of several separate elements into one whole unit. System design is designing a system solution that has been conceptualized when analysing the system based on the needs and problems of the system, so that the system can be applied using computer technology (Mulyani, 2016). Design here concerns the configuration of the software and hardware components needed from the system to be built. The purpose of system design is to provide a clear picture and complete design to the designer and the technical experts involved.

2.3.1 Basic Principles of Design

a. Monolithic System Design

Defining design is more emphasized on system integration where the process of selecting types of sources can integrated in order to obtain effective system especially in the use of costs.

b. Modular System Design

Defining design is more emphasized on functions that have low independence into separate modules (functional sub systems), such as

- 1) Data Collection, process of gathering and measuring information about targeted variables in a strong system.
- 2) Data Processing, process of converting data into information or knowledge for users.
- 3) Data Storage, a component of the recording media that maintains digital data that is used at a certain time
- 4) Information Report, presentation of packaged digital data reports to present processed data to users.

2.3.2 System Design Approach

a. Driven Approach Model

It is one approach in system design using model depiction as documentation and blueprint for developing a system.

b. Rapid Application Development

Is a model of software development systems or systems that are classified into incremental techniques and emphasize short, short, and fast development cycles.

c. FAST-System Design

Is a model of the software development system or system that uses a framework that is flexible enough for various types of projects and strategies.

2.4. PHP Programming

Hypertext Preprocessor or in short PHP is a programming language used to create dynamic web, of course the PHP programming language is different from HTML. PHP is an open source server-side web programming language that is an integrated script with HTML (Anhar, 2010). In the PHP programming language the script or code that is created cannot be displayed on a browser web page, but must be processed first by a web server and then displayed in the form of a web page in a web browser, PHP scripts can also be inserted in HTML and scripts PHP always starts with `<?php` and ends with `?>`. Database management is usually used for PHP programming such as MySQL.

PHP is also called the server side script programming language, because PHP is processed on the server computer. Almost every UI develop that is currently available is always compatible in reading PHP programming syntax. Initially PHP was short for Personal Home Page (personal site) first created by Rasmus Lerdorf in 1995 (Winarno, 2014). At that time PHP was still called Form Interpreted (FI), whose form was in the form of a set of scripts used to process form data from the web. Furthermore Rasmus released the source code for the public and named it PHP / FI. With the release of the source code being open source, many developers are interested in developing PHP. In November 1997, PHP / FI 2.0 was released which was implemented in program C.A company called Zend rewrote PHP interpreters to be cleaner, better, and faster. Then in June 1998, the company released a new interpreter for PHP and inaugurated the

release as PHP 3.0 and stands for PHP modified Hypertext Preprocessing. Some advantages of the PHP programming language is a scripting language that does not compile in its use. PHP is open source making it easier for other developers to develop a system that has been built with the PHP programming language.

2.5. MySQL

MySQL is a database management system software (English: Database Management System) or a multithreaded, multi-user DBMS, with around 6 million installations worldwide. MySQL AB is a software company founded in Sweden in 1995 where the company was the creator of MySQL and made MySQL available as free software under the General Public License (GPL) license. GPL is a software license that guarantees the freedom to use, study, share and modify software (Solihin, 2010). MySQL can be categorized as an open source software or database creation software and can run on various operating systems both Windows and Linux.

Reliability of a database system can be known from the optimal way of working in processing SQL commands made by users and application programs that use them. MySQL supports database transaction operations and non-transaction operations. In non-transaction operations, MySQL can be said to be superior in terms of performance compared to other database software, but there is no guarantee of the reliability of stored data, so non-transaction operations are only suitable for types of applications that do not require data reliability such as blogging, CMS, and the like. For system requirements aimed at businesses it is

strongly recommended to use it, only as a consequence MySQL's performance is not as fast as performance in non-transaction operations.

Various MySQL data types that are often used include:

Table 2.1 Data Type of MySQL

Type	Function
Integer	Used to store data in the form of integers with range for numbers marked -2147483648 up to 2147483647
Bigint	Used to store data in the form of the largest integer with a range for numbers marked -9223372036854775808 up to 9223372036854775807
Char	Used to store data in the form of characters that can handle data up to 255 static characters
Varchar	Used to store data in the form of characters that can handle data up to 255 dynamic characters
Text	Used to store data in the form of characters that can handle data up to 65535 dynamic characters
Decimal (M,D)	Used to store data in the form of fractions of -99,99 to 99,99 for example Decimal (5,2) where there are 2 (two) digits behind the comma
Datetime	Used to store data in the form of a date combination and a clock with a range from 1000-01-01 00:00:00 to 9999-12-31 23:59:59
Longblob	Used to store data in the form of a maximum file size of 4294967295 bytes
Enum ('nilai1', 'nilai2', '..')	Used to store data in the form of enumeration or a choice of a maximum of 65535 characters

2.6. Islamic Tourism

In the Presidential Regulation No. 50/2011 concerning the National Tourism Development Master Plan article 14 paragraph 1 explains that tourist attraction

includes attraction of natural, cultural and man-made tourism. Tourist Attraction Man-made products are developed in various sub-types or categories of tourism activities, one of which is religious tourism. While tourist attraction is defined as everything that is unique, beauty and value that is the target or destination of tourist visits. Islamic tourism in the sense of religious tourism emphasizes uniqueness, beauty and religious values. Such as objects in the form of mosques, relics of historical buildings of religious value, pilgrimage and others. Therefore religious tourism is often closely related to historical tourism, which is part of cultural tourism. So in the Ministry of Tourism under the Deputy for Development of Cultural Tourism Destinations, there is the Field of Development of Historical and Religious Tourism.

2.7. Criteria Islamic Tourism

According to the Global Muslim Travel Index (GMTI), there are several criteria that make a religious tourism object, among others

- Generally moslem tourists travel with family so this criterion becomes the main requirement.
- Conditions of terrorism-free crime and other security guarantees are very important.
- The number of moslem visits means seeing how many visited to that place.
- Guarantee of halal food which is most related to halal and haram food.
- Prayer rooms in the form of mosques and instructions on the place and time of prayer are the daily needs of a moslem.

- Accommodation options in the form of lodging and other accommodations need to guarantee halal.

2.8. Related Research

This research is aimed at Malang City Central Bureau of Statistics (2015) data (2015) recorded that every year Malang City experienced a new population increase of 9.98% or 81,934 people dominated by students. In Malang City there are 26 gas stations in several locations with different facilities and services in each location (ESDM in Figures, 2012). Migrants (visitors) as new residents must find it difficult to find gas stations according to what is needed in the midst of heavy

Implementasi Algoritme Dijkstra pada Webgis untuk Pencarian Lokasi SPBU di Kota Malang, Mega Yuda Rukmana¹, Fatwa Ramdani² From Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya (2018).

The researcher used Botanical gardens is an institution that keeps documentation about living plants collection Purwodadi Botanical Gardens located on Surabaya Malang Km. 65 Highway, Purwodadi Village, Purwodadi District, Pasuruan Regency. Purwodadi Botanical Garden has an area of 85 hectares. Purwodadi Botanical Garden has plants collection of 2.002 species, 178 family, 962 genus and 11.669 specimens. With abundance of plants, an application is required that can lead the way from user's location to intended plant's location.

Implementasi Algoritma Dijkstra Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis Android, Muhammad Syamsuddin Yusuf¹, Hanifah Muslimah Az-Zahra², Diah Harnoni

Apriyanti³ from 1,2 Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya (2017).

Research on communications technologies that support by Dijkstra algorithm, as a determinant of the distribution are more effective for the volunteer. The system also equipped by the data inventory and needs of each refugee, so there is no buildup and lack support in their respective refugee camps, as well as appropriate aid, according to the type of the needs of victims. verified by BNPB and MDMC as , *Implementasi algoritma dijkstra sebagai solusi efektif pembuatan sistem bantuan bencana real time*, Siti Nandiroh¹, Haryanto² dan Hafidh Munawir³ from Universitas Muhammadiyah Surakarta (2013).

Determining the shortest path from one place to another is one of the many problems encountered in everyday life. Dijkstra's algorithm is one of the algorithms to find the shortest path from a vertex to another vertex on a positive weighted grap. The use of Dijkstra's algorithm will also produce an overview of the problem in a mathematical model. *Implementasi Algoritma Dijkstra dalam Pencarian Rute Terpendek Tempat Wisata Kota Medan*, Simamora, Aloysius Gestart Parulian Universitas Sumatera Utara (2018).

Research using Utilization of the internet for searching a location is very needed by the peoples with high mobility and low knowledge about route of an areas. Especially in areas where there are many tourist attractions which most visitors from the outside of the region or even from abroad, who may have difficulty to finding the location of tourist attractions and other public places. Ijen crater is one of the most popular tourist destinations located in the

Bondowosodistrict.

Implementasi algoritma dijkstra untuk pencarian lintasab terpendek lokasi tempat-tempat umum di kabupaten bondowoso berbasis web, Lutfi Ali Muharrom, Universitas Muhammadiyah Jember (2014).

Universitas Sumatera Utara which is located in Padang Bulan is campus that has a lot of activities both academic and non-academic within it. Along with that many activities, then someone's mobility level in campus also becomes higher. So, a geographic information system based on mobile that can produce the most effective direction guideto the destination point is needed to do those activities. One of the methods that we can use in term of searching the shortest path is Dijkstra's algorithm, *Implementasi algoritma dijkstra dalam penentuan ruteterpendek berbasis mobile GIS (studi kasus: Universitas Sumatera Utara*, Johannes Hutabarat (2014).

CHAPTER III

DESIGN IMPLEMENTATION

3.1 Material of Research

The material used in designing the shortest path selection system is in the form of primary data and secondary data.

3.1.1 Primary Data

Primary data is data obtained directly from the object of research, in this case the author uses direct activities in the form of visits to objects of religious tourism

3.1.2 Secondary Data

Secondary data is data obtained directly from the object of research, in this case the author explores information in the form of distance between regions, cities, and districts

3.2 Tool of Research

The tool used to support the author in designing the system for selecting the shortest path is in the form of hardware and software

3.2.1 Hardware

- Operating System : Windows 10 Professional
- Processor : Intel Core i5 2430M 2,40 Ghz
- Memory : 4096 Mb RAM
- Hard Disk : 500 Gb
- Graphic Card: Intel HD Graphic 3000

3.2.2 Software

- Text Writing : Microsoft Word 2016
- Browser : Google Chrome 69.0
- IDE Developed : Notepad++
- Web Service : Xampp 3.2.2
- Diagram Design : Pacestar UML Diagrammer

The flow chart of the test results about dijkstra starts from inputting tourist data and alternative value and process a matrix normalization alternative value of and determination value dijkstra positive negative, process matrix value an finally object selection system test values, the calculation process using the dijkstra method to find results figure will be presented in section on manual calculation.

3.3 Flowchart System

A flowchart for making a catenated matrix of nodes

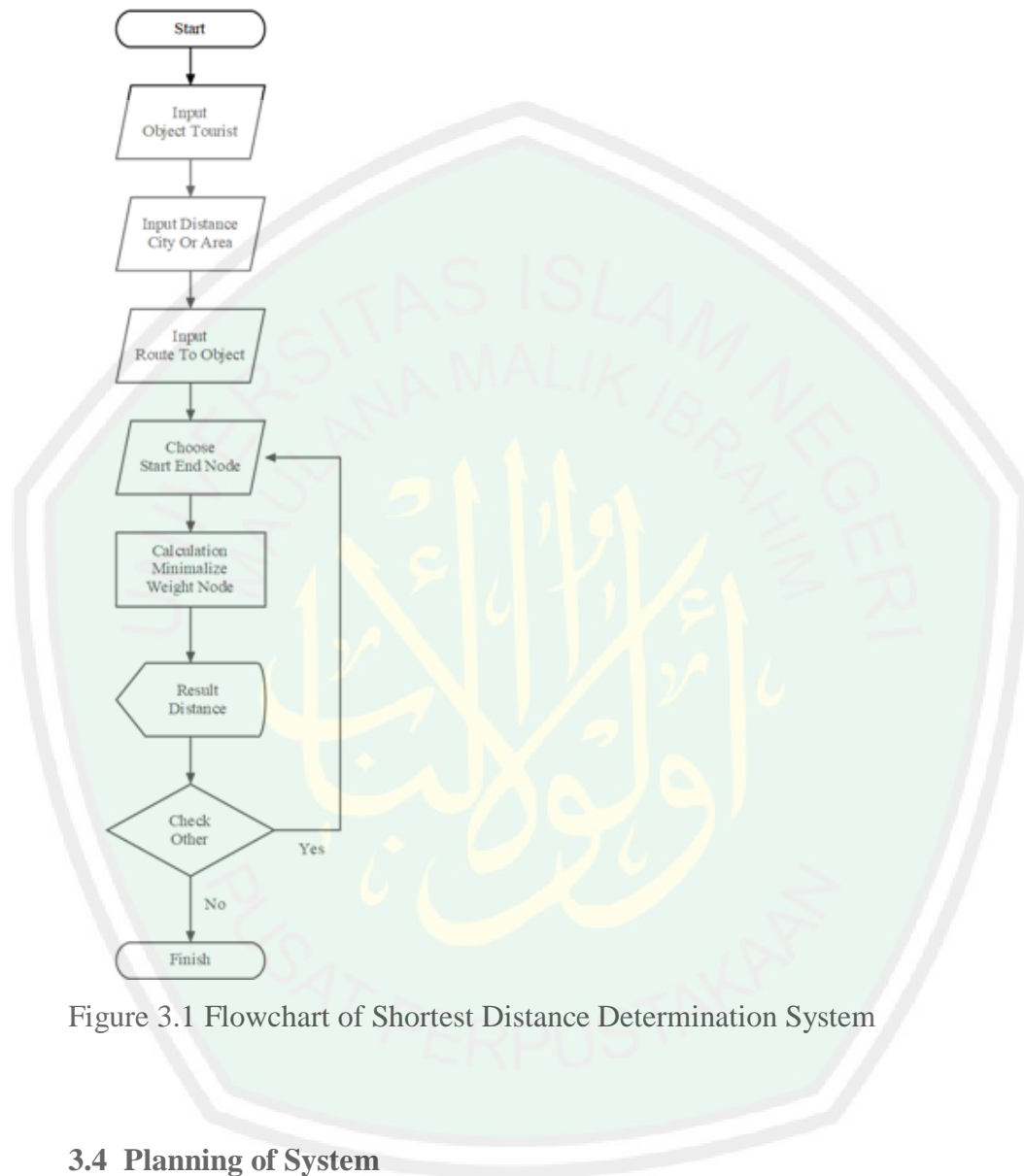


Figure 3.1 Flowchart of Shortest Distance Determination System

3.4 Planning of System

3.4.1 DAD Context

The flow diagram below explains how the flow of the selection of outstanding shortest path begins with the activity of entering data on data city and data on the value of each criterion by each node in to the system. Then it was

continued by the tourists bureau to carry out the system calculation process in obtaining one of most outstanding place among several registration places. The result of This election will be delivered by the tourists bureau in the form of announcements that will be posted on the information wall. Figure 3.2 DAD context.



Figure 3.2 DAD Context

3.4.2 DAD Level 0

In data flow diagram the diagram below explains how the selection path of outstanding node, starting with activities by each place entering \data on city that will be stored in the table data city system database, and data value for each criterion that will be stored in the database system table. then it is continued by the nodes bureau in processing the calculation system. the system will perform calculation so that the results of one the most outstanding node appear among several nodes .the results of this election will be delivered by nodes bureau in the Form of announcements that will posted on the tourism place information.

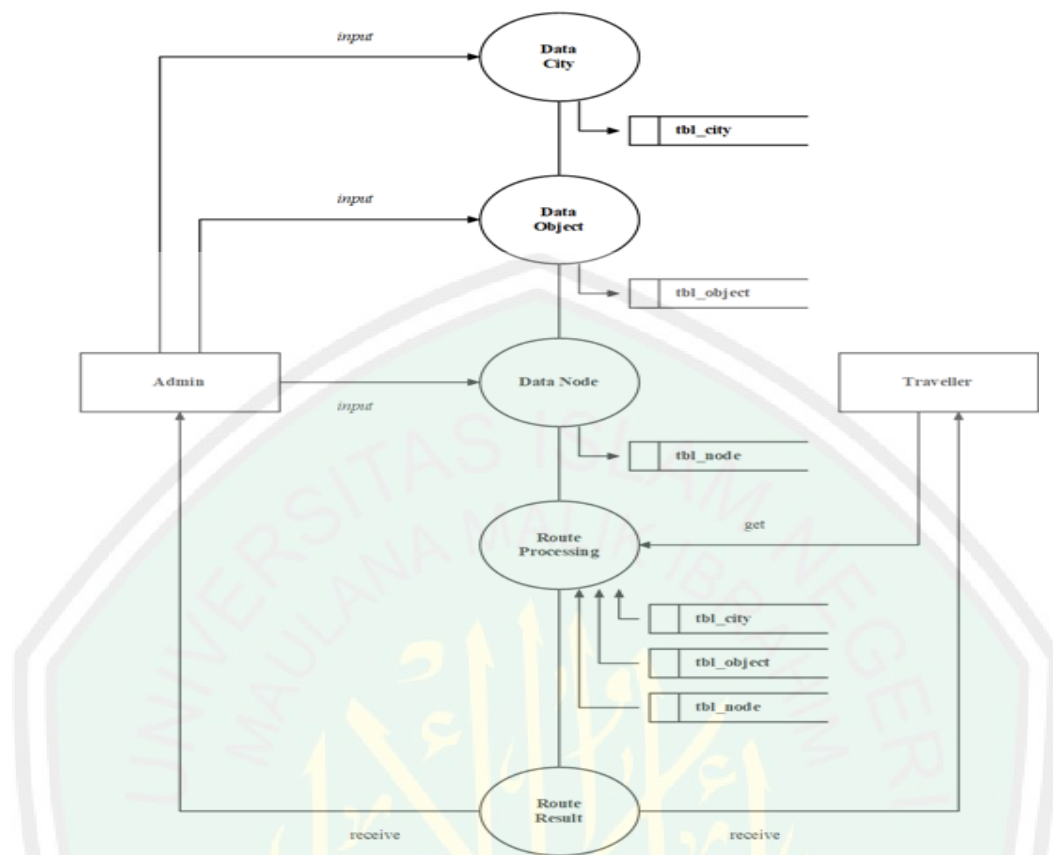


Figure 3.3 DAD Level 0

An entity relationship model (also called an entity relationship diagram) is the graphical representation of entities and their relationships to each other. This type of diagram is used in databases or information system design ERD help you to visualize interrelationship between elements in database and illustrate how they are involved in a system.

Entity Relationship Diagram is model to explain the relationship between data in a database based on data base objects that have relations between relations. To model data structures and relationships among the data as well as using some notations and symbols to see that explains Figure 3.4 Entity Relationship Diagram.

- Identify all the entities in the system. An entity should appear only once in a particular diagram. Create rectangles for all entities and name them properly.
- Identify relationship between entities. Connect the relationship with arrows, and add a name in the middle to describe the relationship.
- Add meaningful names for the entities so they can be understood quickly and easily by all.

3.4.3 Entity Relationship Diagram

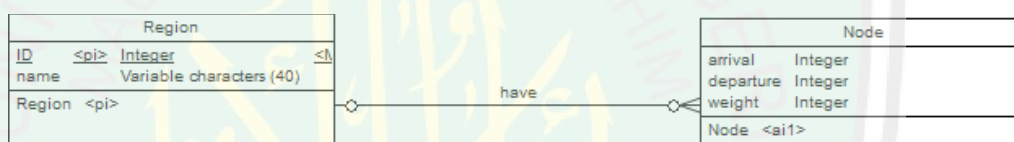


Figure 3.4 Entity Relationship Diagram

3.4.4 Database

In designing the system, the implementation of the dijkstra method for selecting the fastest path that is built has the structure of the data table as follows:

Table 3.1 Data City or Region

Field	Size	Attribute	Description
ID	Integer(11)	Primary Key	ID Number
name	Varchar(30)	-	Name

Table 3.2 Data Object

Field	Size	Attribute	Description
ID	Integer(11)	Foreign Key	ID Number
name	Varchar(40)	-	Name

description	Text(1000)	-	Description
picture1	Longblob	-	Picture 1
picture2	Longblob	-	Picture 2
picture3	Longblob	-	Picture 3

Table 3.3 Data Node

Field	Size	Attribute	Description
arrival	Integer(11)	Foreign Key	Arrival
departure	Integer(11)	Foreign Key	Departure
weight	Integer(11)		Distance (Km)

3.5 Case of Study

System design and examples of problem solving in determining the shortest distance to be built are as follows:

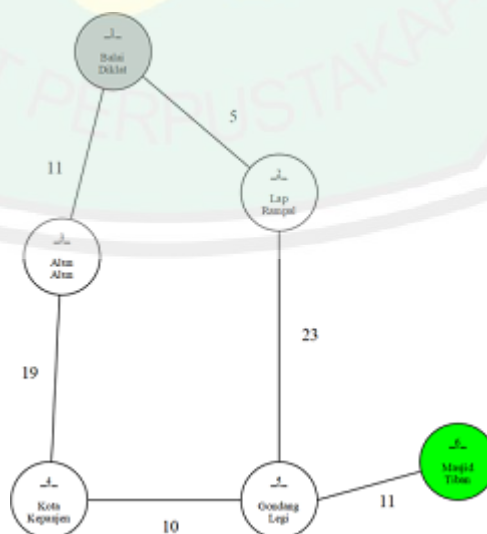


Figure 3.2 Add distance weight of nodes

For the first option explain node 1 is set to be the departure node, so djikstra continues to recalculate the neighboring node that is directly connected to node 1 where node 2 is obtained with a distance value of 5.

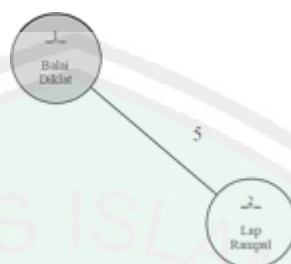


Figure 3.3 Retrieval of neighboring node 1

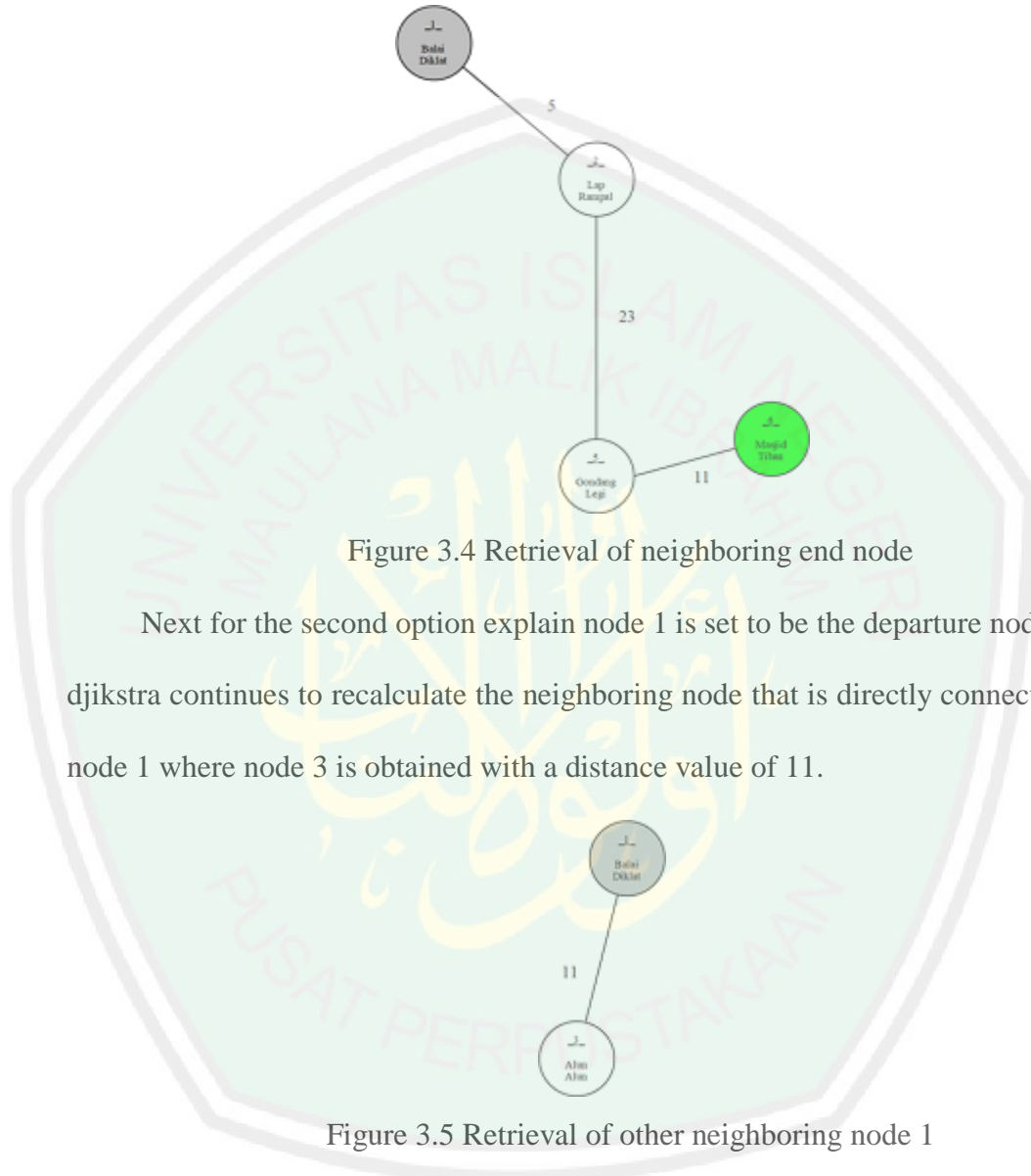
Explain node 2 is set to be the departure node and is marked as the node that has been reached, so djikstra continues to recalculate the neighboring or closest node that is directly connected to node 2 where node 5 is obtained with the smallest distance weight value of $5 + 23 = 28$.



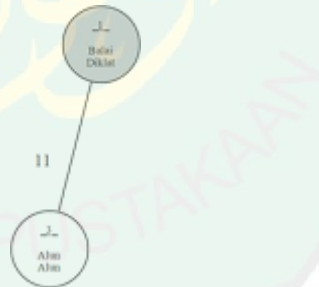
Figure 3.4 Choose minimal distance weight of node 2

Explain node 5 is marked as a node that has been reached, so djikstra continues to recalculate the neighboring or closest node connected directly to node 5 where node 6 is the destination node with a distance weight value

of $5 + 23 + 11 = 29$. Then the results of the calculation in the first option above can be seen with a total distance of 29.



Next for the second option explain node 1 is set to be the departure node, so djikstra continues to recalculate the neighboring node that is directly connected to node 1 where node 3 is obtained with a distance value of 11.



Explain node 3 is marked as a node that has been reached, so djikstra continues to recalculate the neighboring or closest node connected directly to node 3 where node 4 is the destination node with a distance weight value of $11 + 19 = 30$.



Figure 3.6 Retrieval of neighboring node 3

Explain node 4 is marked as a node that has been reached, so djikstra continues to recalculate the neighboring or closest node connected directly to node 3 where node 5 is the destination node with a distance weight value of $11 + 19 + 10 = 40$.

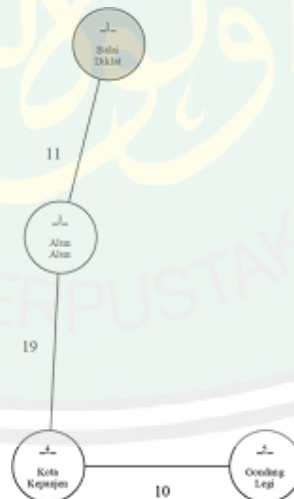


Figure 3.7 Retrieval of neighboring node 4

Explain node 5 is marked as a node that has been reached, so djikstra continues to recalculate the neighboring or closest node connected directly to node 5 where node 6 is the destination node with a distance weight value

of $11 + 19 + 10 + 11 = 41$. Then the results of the calculation in the second option above can be seen with a total distance of 41.

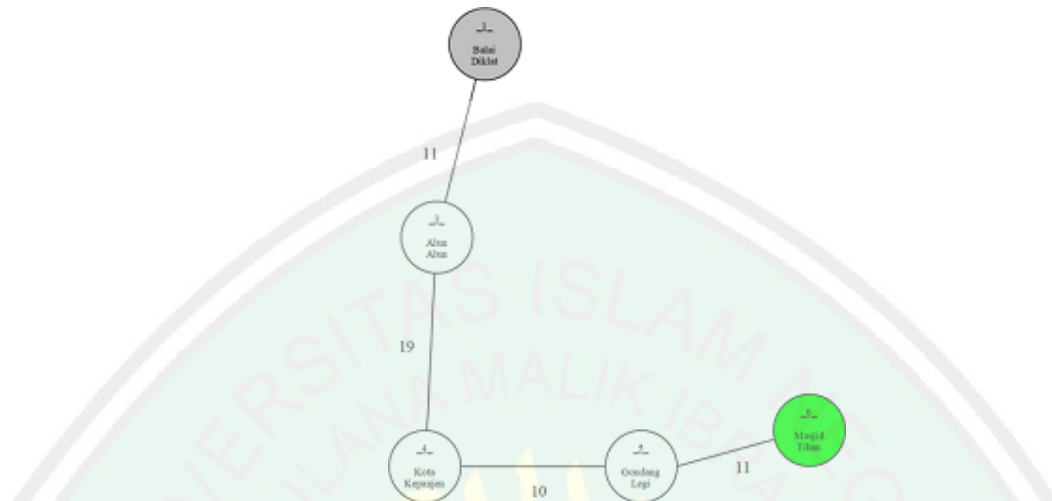


Figure 3.8 Retrieval of neighboring end node

Then the results of the calculation in the first option above can be seen with a total distance of 29, dijkstra will take the smallest result between the first and the second option to be chosen is the first option.

3.6 Design Interfac

Figure 3.10 Form Login Admin

Design interface diagram is describes how database system works when You log in. where action access logins by entering a username and password Which is then checked by the system through the database , if the data is value

Figure 3.11 Form Input Data City

Then the system will display the main page. On the main page, the action The users data pages, such in

Figure 3.12 Form Data Object

In this page display it functions to process tourists location data

Figure 3.13 Form Input Data Object

This page display it serves to process tourist object data both adding changing or deleting.

Figure 3.14 Form Data Node


Figure 3.15 Form Input Data Node

This page display to create new data object

Object	Route	Distance (Km)

Figure 3.16 Form report favourite route

The following images are the route page in displaying option arrival and destination for traveler.



WELCOME TO MALANG

ISLAMIC
TOURISM

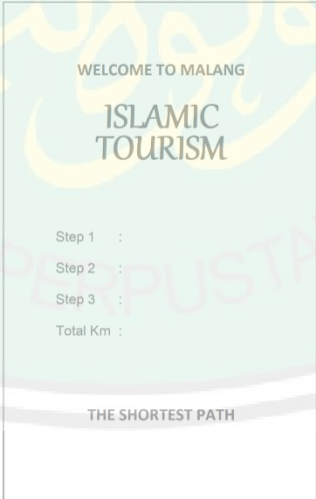
Your Location

Going To

DIRECTION

THE SHORTEST PATH

UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM
PUSAT PERPUSTAKAAN



WELCOME TO MALANG

ISLAMIC
TOURISM

Step 1 :
Step 2 :
Step 3 :
Total Km :

THE SHORTEST PATH

Figure 3.17 Form dashboard traveler

Figure 3.18 Form direction result

CHAPTER IV

TRIAL AND DISCUSSION

4.1 System Implementation

The tool used to support the author in designing the system for selecting the shortest path is in the form of hardware and software.

4.1.1 Hardware

- Operating System : Windows 10 Professional
- Processor : Intel Core i5 2430M 2,40 Ghz
- Memory : 4096 Mb RAM
- Hard Disk : 500 Gb
- Graphic Card: Intel HD Graphic 3000

4.1.2 Software

- Text Writing : Microsoft Word 2016
- Browser : Google Chrome 69.0
- IDE Developed : Notepad++
- Web Service : Xampp 3.2.2
- Diagram Design : Pacestar UML Diagrammer

4.2 Interface Implementation

4.2.1 Login Page

When the information system for selecting the shortest path for selecting Islamic tourism is carried out, there are two types of user pages that can be

accessed here, namely the admin page as a processor and provider of data and pages for tourists or users who want to obtain the fastest track information. The admin page starts with authenticating the username and user password as admin to enter the system environment and can do further data processing.

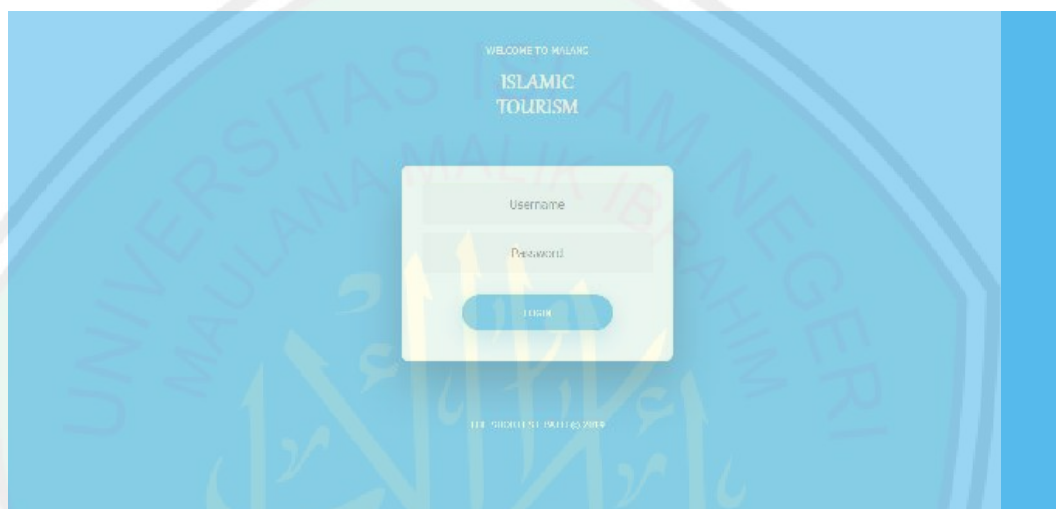


Figure 4.1 Login page

The following is an important line of code login page for the admin to authenticate user.

```
<div class="headlabel">
  welcome to malang
  <h1>islamic<br />tourism</h1>
</div>
<div class="wrapper">
  <div id="FormContent">
    <form method="POST" action="login/in">
      <br />
      <input type="text" name="user" maxlength="10" placeholder="Username" onblur="this.placeholder='Username'"
        onfocus="this.placeholder=''" required>
      <input type="password" name="pass" maxlength="10" placeholder="Password" onblur="this.placeholder='Password'"
        onfocus="this.placeholder=''" required>
      <br /><br />
      <input type="submit" value="Login">
    </form>
  </div>
</div>
<div class="footlabel">the shortest path &copy; 2019</div>
```

Figure 4.2 Line code login page

The following is a line of important code login page when the admin authenticate user.

```
public function in() {
    if ($this->input->post('user') == 'admin' && $this->input->post('pass') == 'admin') {
        $this->session->set_userdata('user', 'admin');

        redirect('dashboard');
    }
    else {
        echo "<script>alert('Check Username And Password');window.history.back();</script>";
    }
}
```

Figure 4.3 Line Code Authenticate User

After successfully entering the admin user page, a dashboard page that contains a map of Islamic tourist areas in Malang is displayed. On the page display the admin user environment there are several menu options, including Region, Object, and Node, each of which serves to process further data.

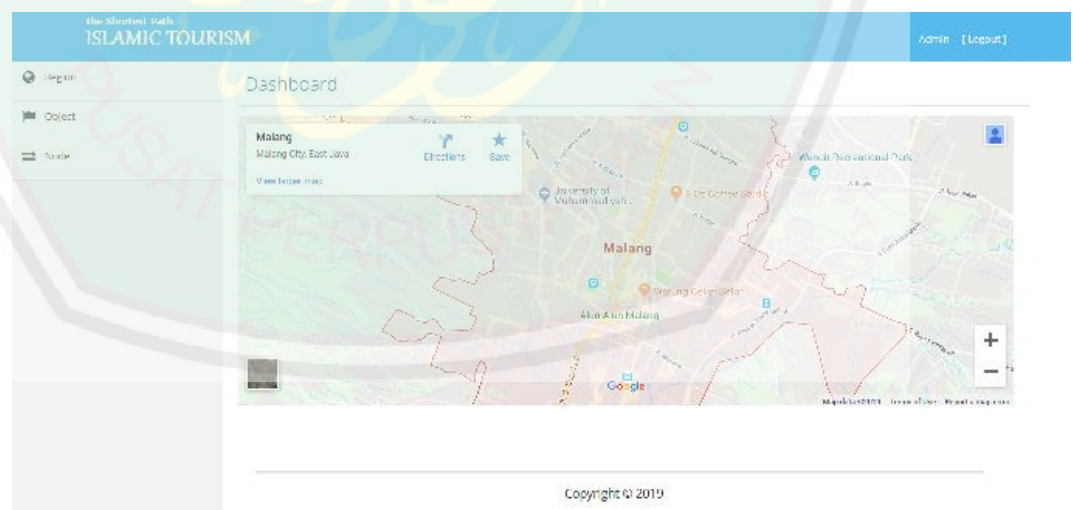


Figure 4.4 Dashboard Page

4.2.2 Region Page

In this region's page display it functions to process tourist location data and certain destination points to make a distance that will be travelled by visiting tourists, both adding change or deleting. To create a new region data is done by filling in the region column, then clicking the Add New button. And there is an Edit icon button that is used to change the region data that has been entered or changed, and the Delete icon button used to delete data.

Name	Actions
Arjosari	[Edit] [Delete]
Karangwatu	[Edit] [Delete]
Kojan	[Edit] [Delete]
Kuning	[Edit] [Delete]
Kroyok	[Edit] [Delete]
Kusambi	[Edit] [Delete]
Turan	[Edit] [Delete]
Kepanjen	[Edit] [Delete]

Figure 4.5 Region Page

The following is an important line of code region page in displaying of data that has been stored.

```

<table class="table table-bordered table-hover">
  <thead>
    <tr>
      <th>Name</th>
      <th>Action</th>
    </tr>
  </thead>
  <tbody>
    <?php
      $this->db->order_by('id', '');
      $sql = $this->db->get('regions');
      foreach ($sql->result() as $row) {
    >
    <tr>
      <td><?php echo $row->name ?></td>
      <td>
        <div class="hidden-sm hidden-xs btn-group">
          <button onClick="window.location.href='<?php echo site_url('region') ?>?
            p=edit&id=<?php echo $row->id ?>'" class="btn btn-xs btn-info"
            title="Edit" style="width: 5em; margin-right: 1em;">
            <i class="ace-icon fa fa-pencil bigger-120"></i>
          </button>
          <button onClick="window.location.href='<?php echo site_url('region/del') ?
            >?id=<?php echo $row->id ?>'" class="btn btn-xs btn-danger"
            title="Delete" style="width: 5em;">
            <i class="ace-icon fa fa-trash-o bigger-120"></i>
          </button>
        </div>
      </td>
    </tr>
    <?php } ?>
  </tbody>
</table>

```

Figure 4.6 Line Code Display Data Region

The following is a line of important code region page when admin adds region data.

```

public function add() {
  $sql = $this->db->get_where('regions', array('name' => $this->input->post('name')));
  if ($sql->num_rows()<1) {
    $data = array(
      'name' => $this->input->post('name')
    );
    $this->db->insert('regions', $data);
  }
  redirect('region');
}

```

Figure 4.7 Line Code Add Region

The following is a line of important code region page when admin updates region data.

```
public function edit() {
    $sql = $this->db->get_where('regions', array('name' => $this->input->post('name'), 'id!=' => $_GET['id']));
    if ($sql->num_rows()<1) {
        $data = array(
            'name' => $this->input->post('name')
        );
        $this->db->where('id', $_GET['id']);
        $this->db->update('regions', $data);
    }
    redirect('region');
}
```

Figure 4.8 Line Code Update Region

The following is a line of important code region page when admin deletes region data.

```
public function del() {
    $this->db->delete('regions', array('id' => $_GET['id']));
    redirect('region');
}
```

Figure 4.9 Line Code Delete Region

4.2.3 Object Page

On the object page display it serves to process tourist object data, both adding changes or deleting. To create a new data object is done by clicking the Add New button. And there is an Edit icon button that is used to change the region data that has been entered or changed, and the Delete icon button used to delete data.

Name	Region	Action
Masjid Jami	Lowokwaru	Edit Delete
Masjid sabillillah	Blimbing	Edit Delete
Masjid Tiban	Turen	Edit Delete
Muslum Brawijaya	Klojen	Edit Delete

Figure 4.10 Object Page

The following is an important line of code object page in displaying of data that has been stored.

```
<table class="table table-bordered table-hover">
  <thead>
    <tr>
      <th>Name</th>
      <th>Region</th>
      <th>Action</th>
    </tr>
  </thead>
  <tbody>
    <?php
      $this->db->order_by('name', '');
      $sql = $this->db->get('objects');
      foreach ($sql->result() as $row) {
        $sqld = $this->db->get_where('regions', array('id' => $row->region)); $rowd =
        $sqld->row();
      }
    </?php
  </tbody>
</table>
```



```

<tr>
  <td><?php echo $row->name ?></td>
  <td><?php echo $rowd->name ?></td>
  <td>
    <div class="hidden-sm hidden-xs btn-group">
      <button onClick="window.location.href='<?php echo site_url('objects') ?>?
      p=edit&name=<?php echo $row->name ?>&region=<?php echo $row->region ?>'"
      class="btn btn-xs btn-info" title="Edit" style="width: 5em; margin-right:
      1em;">
        <i class="ace-icon fa fa-pencil bigger-120"></i>
      </button>
      <button onClick="window.location.href='<?php echo site_url('objects/del')
      ?>?name=<?php echo $row->name ?>&region=<?php echo $row->region ?>'"
      class="btn btn-xs btn-danger" title="Delete" style="width: 5em;">
        <i class="ace-icon fa fa-trash-o bigger-120"></i>
      </button>
    </div>
  </td>
</tr>
<?php } ?>
</tbody>
</table>

```

Figure 4.11 Line Code Display Data Object

In the page view, the added data object describes how to add a data object. Beginning with completing all the fields contained, namely Name, description, image 1, image 2, image 3, and region, then proceed by clicking the Save button to save the object data.

Figure 4.12 Add New Object Page

The following is a line of important code object page when admin adds object data.

```
public function add() {  
    $sql = $this->db->get_where('objects', array('name' => $this->input->post('name')));  
    if ($sql->num_rows()<1) {  
        $data = array(  
            'name' => $this->input->post('name'),  
            'descr' => $this->input->post('descr'),  
            'pic1' => file_get_contents($_FILES['pic1']['tmp_name']),  
            'pic2' => file_get_contents($_FILES['pic2']['tmp_name']),  
            'pic3' => file_get_contents($_FILES['pic3']['tmp_name']),  
            'region' => $this->input->post('region')  
        );  
        $this->db->insert('objects', $data);  
    }  
    redirect('objects');  
}
```

Figure 4.13 Line Code Add Object

In the page view, change the data object to describe how the process of changing a data object. Beginning with completing all fields contained, namely Name, description, image 1, image 2, image 3, and region, then proceed by clicking the Update button to save the object data.

The screenshot shows a web interface for updating an object. The header includes the site name 'ISLAMIC TOURISM' and the user 'admin'. A sidebar on the left has 'Object' selected. The main form has the following fields:

- Name:** Masjid Ibbeni
- Description:** bla bla bla
- Picture 1:** Browse... No file selected.
- Picture 2:** Browse... No file selected.
- Picture 3:** Browse... No file selected.
- Region:** Turon

An 'Update' button is located at the bottom of the form.

Figure 4.14 Update Object Page

The following is a line of important code object page when admin updates object data.

```
public function edit() {
    if ($_FILES['pic1']['size'] > 0) {
        $data = array(
            'pic1' => file_get_contents($_FILES['pic1']['tmp_name'])
        );
        $this->db->where(array('name' => $_GET['name'], 'region' => $_GET['region']));
        $this->db->update('objects', $data);
    }

    if ($_FILES['pic2']['size'] > 0) {
        $data = array(
            'pic2' => file_get_contents($_FILES['pic2']['tmp_name'])
        );
        $this->db->where(array('name' => $_GET['name'], 'region' => $_GET['region']));
        $this->db->update('objects', $data);
    }

    if ($_FILES['pic3']['size'] > 0) {
        $data = array(
            'pic3' => file_get_contents($_FILES['pic3']['tmp_name'])
        );
        $this->db->where(array('name' => $_GET['name'], 'region' => $_GET['region']));
        $this->db->update('objects', $data);
    }
}
```

```

    $data = array(
        'name' => $this->input->post('name'),
        'descr' => $this->input->post('descr'),
        'region' => $this->input->post('region')
    );
    $this->db->where(array('name' => $_GET['name'], 'region' => $_GET['region']));
    $this->db->update('objects', $data);

    redirect('objects');
}

```

Figure 4.15 Line Code Update Object

The following is a line of important code object page when admin deletes object data.

```

public function del() {
    $this->db->delete('objects', array('name' => $_GET['name'], 'region' => $_GET['region']));

    redirect('objects');
}

```

Figure 4.16 Line Code Delete Object

4.2.4 Node Page

On page views this node serves to process path data, both adding changes or deleting. To create a new path data is done by clicking the Add New button. And there is an Edit icon button that is used to change the path data that has been entered or changed, and the Delete icon button used to delete data.

Arrival	Departure	Weight (Km)	Action
Aripaori	Lowekwora	4	Edit Delete
Lomowetu	Kigen	1	Edit Delete
Kojen	Kobot	12	Edit Delete
Kowad	Lawatig Legi	5	Edit Delete
Gondang Lodi	Turan	3	Edit Delete
Lidempu	Uwari	4	Edit Delete
Kapanien	Gondang Lodi	20	Edit Delete

Figure 4.17 Node Page

The following is an important line of code node page in displaying of data that has been stored.

```
<table class="table table-bordered table-hover">
  <thead>
    <tr>
      <th>Arrival</th>
      <th>Departure</th>
      <th>Weight (Km)</th>
      <th>Action</th>
    </tr>
  </thead>
  <tbody>
    <?php
      $this->db->order_by('arriv', '');
      $this->db->order_by('weight', '');
      $sql = $this->db->get('nodes');
      foreach ($sql->result() as $row) {
        $sqld1 = $this->db->get_where('regions', array('id' => $row->arriv)); $rowd1 =
        $sqld1->row();
        $sqld2 = $this->db->get_where('regions', array('id' => $row->depar)); $rowd2 =
        $sqld2->row();
      }
    >>
```

```

<tr>
  <td><?php echo $rowd1->name ?></td>
  <td><?php echo $rowd2->name ?></td>
  <td><?php echo $row->weight ?></td>
  <td>
    <div class="hidden-sm hidden-xs btn-group">
      <button onClick="window.location.href='<?php echo site_url('node') ?>?
      p=edit&a=<?php echo $row->arriv ?>&d=<?php echo $row->depar ?>'"
      class="btn btn-xs btn-info" title="Edit" style="width: 5em; margin-right:
      1em;">
        <i class="ace-icon fa fa-pencil bigger-120"></i>
      </button>
      <button onClick="window.location.href='<?php echo site_url('node/del') ?>?
      a=<?php echo $row->arriv ?>&d=<?php echo $row->depar ?>'" class="btn
      btn-xs btn-danger" title="Delete" style="width: 5em;">
        <i class="ace-icon fa fa-trash-o bigger-120"></i>
      </button>
    </div>
  </td>
</tr>
<?php } ?>
</tbody>
</table>

```

Figure 4.18 Line Code Display Data Node

In the display page, add data node describes how to add a data node. Begin by completing all fields contained there, namely Arrival, departure, and weight (Km), then proceed by clicking the Save button to save the path data.

The screenshot shows a web application interface for adding a new node. The header is blue with the text 'The Shortest Path ISLAMIC TOURISM' and a user profile 'admin [Logout]'. A sidebar on the left has a 'Node' menu item highlighted. The main content area is titled 'Node' and contains three input fields: 'Arrival', 'Departure', and 'Weight (km)'. A green 'Save' button is located at the bottom of the form.

Figure 4.19 Add New Node Page

The following is a line of important code node page when admin adds node data.

```

public function add() {
    $sql = $this->db->get_where('nodes', array('arriv' => $this->input->post('arriv'), 'depar' => $this->input->post('depar')));
    if ($sql->num_rows() < 1) {
        $data = array(
            'arriv' => $this->input->post('arriv'),
            'depar' => $this->input->post('depar'),
            'weight' => $this->input->post('weight')
        );
        $this->db->insert('nodes', $data);
    }
    redirect('node');
}

```

Figure 4.20 Line Code Add Node

In the page view, change the data path illustrates how the process of changing a data path. Beginning with completing all the fields contained, namely Arrival, departure, and weight (Km), then proceed by clicking the Update button to save the path data.

Figure 4.21 Update Node Page

The following is a line of important code node page when admin updates node data.

```
public function edit() {
    $data = array(
        'weight' => $this->input->post('weight')
    );
    $this->db->where(array('arriv' => $_GET['a'], 'depar' => $_GET['d']));
    $this->db->update('nodes', $data);

    redirect('node');
}
```

Figure 4.22 Line Code Update Node

The following is a line of important code node page when admin deletes node data.

```
public function del() {
    $this->db->delete('nodes', array('arriv' => $_GET['a'], 'depar' => $_GET['d']));

    redirect('node');
}
```

Figure 4.23 Line Code Delete Node

4.2.5 Tourist Route Page

On page views this route serves to obtain information on the fastest path that can be traversed to visit an Islamic tourist destination to be visited. To find out which route routes can be traversed, begins by selecting and completing the current location column and the tourist destination that will be visited.

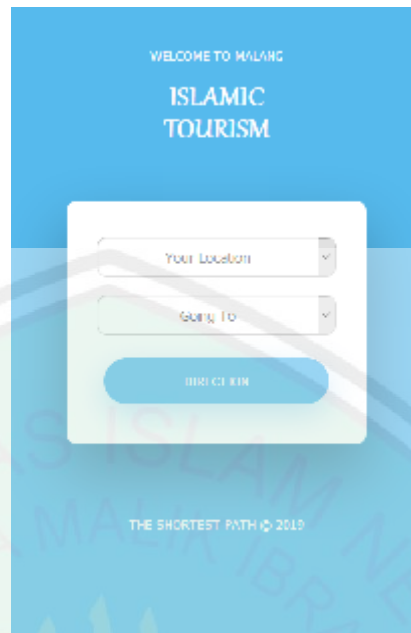


Figure 4.24 Route Page

The following is an important line of code route page in displaying option arrival and destination for traveller.

```

<form onSubmit="return route()" method="POST" action="route/proses">
  <br /><br />
  <select id="arriv" name="arriv" class="form-control" required>
    <option selected value="0">Your Location</option>
    <?php
      $this->db->order_by('id', '');
      $sql = $this->db->get('regions');
      foreach ($sql->result() as $row) {
    ?>
    <option value="<?php echo $row->id ?>"><?php echo $row->name ?>
    <?php } ?>
  </select>
  <br />
  <select id="depar" name="depar" class="form-control" required>
    <option selected value="0">Going To</option>
    <?php
      $this->db->order_by('name', '');
      $sql = $this->db->get('objects');
      foreach ($sql->result() as $row) {
    ?>
    <option value="<?php echo $row->region ?>"><?php echo $row->name ?>
    <?php } ?>
  </select>
  <br />
  <input type="submit" value="Direction">
</form>

```

Figure 4.25 Line Code Display Option Route

The following is a line of important code progress route page after traveller choosing arrival and destination.

```

public function proses() {
  if ($this->input->post('arriv') != $this->input->post('depar')) {
    $this->session->set_userdata('depar', $this->input->post('depar'));
    $this->db->truncate('temp');

    $this->db->order_by('weight', '');
    $sql = $this->db->get_where('nodes', array('depar' => $this->input->post('depar')));
    foreach ($sql->result() as $row) {
      $data = array(
        'arriv' => $row->arriv,
        'depar' => $row->depar,
        'weight' => $row->weight,
        'cek' => '1'
      );
      $this->db->insert('temp', $data);
    }
  }
}

```

```

for ($d = 1; $d <= 100; $d++) {
    $this->db->order_by('weight', '');
    $sql = $this->db->get_where('temp', array('cek' => $d));
    foreach ($sql->result() as $row) {
        $this->db->order_by('weight', '');
        $sqln = $this->db->get_where('nodes', array('depar' => $row->arriv));
        foreach ($sqln->result() as $rown) {
            $datan = array(
                'arriv' => $rown->arriv,
                'depar' => $rown->depar,
                'weight' => $rown->weight,
                'cek' => $d+1
            );
            $this->db->insert('temp', $datan);
        }
    }
}

$arriv = $this->input->post('arriv');
$r = $arriv;
for ($d = 1; $d <= 100; $d++) {
    $sql = $this->db->get_where('temp', array('arriv' => $arriv));
    $row = $sql->row();

    $arriv = $row->depar;
    $r = $r . '#' . $arriv;

    if ($arriv==$this->input->post('depar')) { break; }
}

$this->session->set_userdata('result', $r);
}

redirect('route');
}

```

Figure 4.26 Line Code Progress Route

In the results page display this route will display the fastest route information that can be passed to visit an Islamic tourist destination that will be addressed. The system will automatically calculate and calculate the shortest distance that travelers can pass with the Dijkstra method.



Figure 4.27 Result Page

The following is a line of important code result progress route page for traveller.

```

<div class="headbest">
  best shortest path<br />to
  <?php
    $sql = $this->db->get_where('objects', array('region' => $this->session->userdata('depar')));
    $row = $sql->row(); echo $row->name;
  ?>
</div>
<?php
  $ro = $this->session->userdata('result');
  for ($r = 0; $r <= substr_count($ro,"#"); $r++) {
    $sql = $this->db->get_where('regions', array('id' => explode("#",$ro)[$r]));
    $row = $sql->row();

    $s = $r + 1;
    echo "
      <div style='float: left; text-align: left; width: 90%; padding: 4%;'>
        <div style='float: left; width: 30%;'>STEP $s</div>
        <div style='float: left; width: 5%;'></div>
        <div style='float: left; width: 60%;'>" . $row->name . "</div>
      </div>
    ";
  }
?>
<input onClick="window.location.href='<?php echo site_url('route/finish') ?>'" type="submit" value="0 0 K"
style="margin-top: 1.5em;">

```

Figure 4.28 Line Code Result Progress Route

4.3 Discussion

4.3.1 Test Login Page

Table 4.1 Testing for Login Page

User	Admin
Condition	True
Test method	Accessing to login page Username: admin, Password: admin Fill in and complete fields Click button Login
Expected correct	Displaying dashboard page for admin users
OK / NOT OK	OK
Screen shoot result	

Testing on the login page by the admin user, first the user completes the fields then click the Login button. After successfully logging in, the user will be directed to the board page. The fields that are on the login page is username and password. The expected results from this test is displaying a dashboard page for admin users. The condition is wrong if the user fails to authenticate user.

4.3.2 Test Region Page

Table 4.2 Testing for Region Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Region
Expected correct	Displaying region page with stored data
OK / NOT OK	OK
Screen shoot result	

Testing on the region page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Region menu. The expected result of this test is to display all the region data that has been stored. Incorrect condition if the page does not display data that has been stored.

4.3.3 Test Add of Region Page

Table 4.3 Testing for Add of Region Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Region Fill in field Click button Add New
Expected correct	Successfully added new region data
OK / NOT OK	OK
Screen shoot result	

Testing on the add region page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Region menu. The field that are on the region page is Name. The expected result of this test is success to add new region data. Incorrect condition if data cannot be stored.

4.3.4 Test Update of Region Page

Table 4.4 Testing for Update of Region Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Region Select data of region Click icon Edit Fill in field Click button Update
Expected correct	Successfully updated region data
OK / NOT OK	OK
Screen shoot result	

Testing on the update region page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Region menu. The field that are on the region page is Name. The expected result of this test is success to update region data. Incorrect condition if data cannot be update.

4.3.5 Test Delete of Region Page

Table 4.5 Testing for Delete of Region Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Region Select data of region Click icon Delete
Expected correct	Successfully deleted region data
OK / NOT OK	OK
Screen shoot result	

Testing on the delete region page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Region menu. The expected result of this test is success to delete region data. Incorrect condition if data cannot be delete.

4.3.6 Test Object Page

Table 4.6 Testing for Object Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Object
Expected correct	Displaying object page with stored data
OK / NOT OK	OK
Screen shoot result	

Testing on the object page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Object menu. The expected result of this test is to display all the object data that has been stored. Incorrect condition if the page does not display data that has been stored.

4.3.7 Test Add of Object Page

Table 4.7 Testing for Add of Object Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Object Click button Add New Fill in and complete fields Click button Save
Expected correct	Successfully added new object data
OK / NOT OK	OK
Screen shoot result	

Testing on the add object page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Object menu. The fields that are on the object page is Name, Description, Picture 1, Picture 2, Picture 3, and Region. The expected result of this test is success to add new object data. Incorrect condition if data cannot be stored.

4.3.8 Test Update of Object Page

Table 4.8 Testing for Update of Object Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Object Select data of object Click icon Edit Fill in and complete fields Click button Update
Expected correct	Successfully updated object data
OK / NOT OK	OK
Screen shoot result	

Testing on the update object page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Object menu. The fields that are on the object page is Name, Description, Picture 1, Picture 2, Picture 3, and Region. The expected result of this test is success to update object data. Incorrect condition if data cannot be update.

4.3.9 Test Delete of Object Page

Table 4.9 Testing for Delete of Object Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Object Select data of object Click icon Delete
Expected correct	Successfully deleted object data
OK / NOT OK	OK
Screen shoot result	

Testing on the delete object page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Object menu. The expected result of this test is success to delete object data. Incorrect condition if data cannot be delete.

4.3.10 Test Node Page

Table 4.10 Testing for Node Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Node
Expected correct	Displaying node page with stored data
OK / NOT OK	OK
Screen shoot result	

Testing on the node page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Node menu. The expected result of this test is to display all the node data that has been stored. Incorrect condition if the page does not display data that has been stored.

4.3.11 Test Add of Node Page

Table 4.11 Testing for Add of Node Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Node Click button Add New Fill in and complete fields Click button Save
Expected correct	Successfully added new node data
OK / NOT OK	OK
Screen shoot result	

Testing on the add node page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Node menu. The fields that are on the node page is Arrival, Departure, and Weight. The expected result of this test is success to add new node data. Incorrect condition if data cannot be stored.

4.3.12 Test Update of Node Page

Table 4.12 Testing for Update of Node Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Node Select data of node Click icon Edit Fill in and complete fields Click button Update
Expected correct	Successfully updated node data
OK / NOT OK	OK
Screen shoot result	

Testing on the update node page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Node menu. The fields that are on the node page is Arrival, Departure, and Weight. The expected result of this test is success to update node data. Incorrect condition if data cannot be update.

4.3.13 Test Delete of Node Page

Table 4.13 Testing for Delete of Node Page

User	Admin
Condition	True
Test method	Login with user admin Username: admin, Password: admin Select menu Node Select data of node Click icon Delete
Expected correct	Successfully deleted node data
OK / NOT OK	OK
Screen shoot result	

Testing on the delete node page first the user admin login using a username and password. After successfully logging in, the user will be directed to the dashboard page, then user can select Node menu. The expected result of this test is success to delete node data. Incorrect condition if data cannot be delete.

4.3.14 Test Route Page

Table 4.14 Testing for Route Page

User	Traveller
Condition	True
Test method	Accessing to main page Fill in and complete fields
Expected correct	Displaying option choose route for traveller
OK / NOT OK	OK
Screen shoot result	

Testing on the route page by the traveller, first the user completes the fields then click the Direction button. The fields that are on the route page is Arrival and Departure. The expected results from this test is displaying option choose a route page for traveller. The condition is wrong if the user fails to access page.

4.3.15 Test Result of Route Page

Table 4.15 Testing for Result of Route Page

User	Traveller
Condition	True
Test method	Accessing to main page Fill in and complete fields Click button Direction
Expected correct	Displaying best path route page for traveller
OK / NOT OK	OK
Screen shoot result	

Testing on the result of route page by the traveller, first the user completes the fields then click the Direction button. After successfully for process, the user will be displaying best path of route page. The expected results from this test is displaying best path a route page for traveller. The condition is wrong if the user fails to see best path.

4.4 Analysis of Results

The results analysis aims to find out whether the method used in the design is in accordance with the results. At the trial stage, origin and destination destinations are needed. The data will be used to determine the shortest distance using the Dijkstra algorithm. Examples of data that will be used as a trial are shown in Table 4.16 below:

Table 4.16 Test Data

No	Origin	Destination
1	Arjosari	Masjid Tiban
2	Blimbing	Masjid Jami
3	Kepanjen	Masjid Tiban
4	Turen	Musium Brawijaya
5	Kepanjen	Masjid Tiban

From the data in Table 4.16 then a test is performed for each data with different origin and destination according to the table. The results of tracing the shortest route using Dijkstra can be shown in Table 4.17 below:

Table 4.17 Trial Results

No	Origin	destination	Distance
1	Arjosari	Masjid Tiban	33 Km
2	Blimbing	Masjid Jami	6 Km
3	Dampit	Masjid Tiban	12 Km
4	Turen	Musium Brawijaya	27 Km
5	Kepanjen	Masjid Tiban	27 Km

Based on the test results shown in Table 4.17 that the Dijkstra algorithm has succeeded in getting the shortest route based on different origin and destination. This proves that the algorithm can be implemented in the search for the shortest route in Islamic tourism in the city of Malang.

CHAPTER V

CONCLUSIONS AND SUGGESTIONS

5.1 Conclusion

Based on the formulation of the problem and the research with title "Implementation of the Dijkstra Method Choosing The Shortest path of Islamic tourism Malang City" it can be concluded implementation of the Dijkstra Method in building an information system that can be useful for tourists to determine the fastest path to religious tourism destinations in Malang City use The tool used to support the author in designing the system for selecting the shortest path is in the form of hardware that is operating System mac OS Mojave, processor 2,5 GHz Intel Core i5 memori 10 GB 1333 MHz DDR3, Hard Disk 500 Gb and software that is operating System Mac OS Mojave, Browser Google Chrome 69.0, IDE Developed Notepad++, Web Service Xampp 3.2.2, Diagram Design Pacestar UML Diagrammer.

From the results of trial cases that have been made, 3 tourist attractions were taken as data on tourist destinations namely the Tiban Mosque, Masji Jami 'and the Brawijaya Museum. Based on the destination, 4 places are taken as the origin, namely Arjosari, Blimbing, Kepanjen, Turen. With the results of the route from Arjosari to the Masjid Tibib with a distance of 33km, from Blimbing to Masjin Jami 'with a distance of 6km, from dampit to the mosque with a distance of 12 km, from Turen to the Museum of Brawijaya with a distance of 27km and from Kepanjeng to Masjin Tiban with a distance of 27km. From these results it can be

seen that the results of this system are the same as the results of searches done manually. Determination of training data used for classification is done in a manner Trial and error, to get the best accuracy found in testing this system is 71.24%

5.2 Suggestion

1. It is expected that the City Government of Malang in this case the Office of Tourism to create an online service in the form of a system that makes it easy for tourists to obtain information both from locations, types and tourist destinations in the city of Malang..
2. Further research can be done by expanding the scope of research and the object under study besides the tourist area. Future studies can also use other methods that might be more accurate and might provide.

REFERENCES

- Kusrini, 2007. Tuntunan Praktis Membangun Sistem Informasi Akuntansi, Edisi 2007, ANDI Offset, Yogyakarta
- Dachi, Alyakin, Rahmat, 2017. Proses Dan Analisis Kebijakan, Cetakan 2017, Utama, Yogyakarta Budi
- Rusmana, Agus, 2019. *The Future Of Organizational Communication In The Industrial*, 2019, AKSEL Media Akselerasi, Bandung
- Sari, Febrina, 2018. Metode Dalam Pengambilan Keputusan, Cetakan I 2018, Budi Utama, Yogyakarta
- Basyaib, Fachmi, 2006. Teori Pembuatan Keputusan, 2006, Grasindo, Jakarta
- Soetam, Rizky, Wicaksono, 2018. Studi Kasus Sistem Penunjang Keputusan, Seribu Bintang, Malang 2018,
- Marimin, 2004. Teknik Dan Aplikasi Pengambilan Keputusan Kriteria Majemuk, 2004, Grasindo, Jakarta
- Marsudi, 2016. Teori Graf, 2016, UB Press, Malang
- A., Yunaeti, Elisabet, 2017. Pengantar Sistem Informasi, 2017, ANDI Offset, Yogyakarta
- Maniah, 2017. Analisis Dan Perancangan Sistem Informasi, Cetakan I 2017, Budi Utama, Yogyakarta
- Soeherman, Bonnie, 2008. *Designing Information System*, 2008, PT Elex Media Komputindo, Jakarta
- Anhar, 2010. Panduan Menguasai PHP & MySQL, Cetakan I 2010, PT TransMedia, Jakarta
- Winarno, Edy, 2014. Pemrograman Web Berbasis HTML5 PHP Dan Javascript, 2014, PT Elex Media Komputindo, Jakarta
- Solichin, Achmad, 2010. Pemrograman Web Dengan PHP Dan MySQL, 2010, UBL, Jakarta
- Yanto, Robi, 2016. Manajemen Basis Data Menggunakan MySQL, Cetakan I Budi Utama, Yogyakarta 2016,

Sudarma, 2010. Panduan Belajar MySQL Database Server, Cetakan I 2010, PT TransMedia, Jakarta

