TRIP PLANNING QUERY FOR BAGAN USING THE PARTIAL SEQUENCED ROUTE ALGORITHM

WAI MAR LWIN

M.C.Sc. **JUNE, 2019**

TRIP PLANNING QUERY FOR BAGAN USING THE PARTIAL SEQUENCED ROUTE ALGORITHM

By

WAI MAR LWIN

B.C.Sc.(Honours)

A dissertation submitted in partial fulfillment of the requirements for the degree of

Master of Computer Science

(M.C.Sc.)

University of Computer Studies, Yangon

JUNE, 2019

ACKNOWLEDGEMENTS

I would like to express my respectful gratitude and appreciation to Professor **Dr. Mie Mie Thet Thwin**, Rector, the University of Computer Studies, Yangon, for her general guidance, workable environment and overall supporting during the period of study.

I am deeply thankful to my supervisor, **Dr. Myint Myint Sein,** Professor, Head of Geographical Information System (GIS) Lab, the University of Computer Studies, Yangon, for her kindly help, invaluable guidance and comments throughout the period of my thesis.

I also wish to offer my respectful gratitude to Course Coordinator **Dr. Thi Thi Soe Nyunt**, Professor and Head of Faculty of Computer Science, the University of Computer Studies, Yangon, for her support and guidance on this thesis.

I would also like to thank **Daw Aye Aye Khine**, Associate Professor and Head of English Department, the University of Computer Studies, Yangon, for her patient editing my thesis from the language point of view.

I extend many thanks to all my teachers from **Geographical Information System (GIS) Lab**, the University of Computer Studies, Yangon, for their motivating encouragement and stimulating discussions about this research to fulfill all my needs.

Finally, I thank all my teachers who taught and helped me during the period of studies in the University of Computer Studies, Yangon and thank my parents.

STATEMENT OF ORIGINALITY

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

--------------------------------	----------------------------------
Date	Wai Mar Lwin

ABSTRACT

Trip planning query represents an important class of queries in Geographic Information System (GIS). Trip planning has some important features that include selection of points of interest (POIs), path routing, and trip time. In the proposed system, path routing is used to search the route with the minimum road network distance and to systematically make the route for the trips of both tourists and local people who visit Bagan. Users can visit the number of POIs of categories that include pagodas, restaurants and hotels in Bagan. The trip planning system of Bagan is implemented by using the Partial Sequenced Route (PSR) Algorithm. Mobile application development is now quite popular, and Android is one of the most popular mobile device platforms. So, Android platform is used in this trip planning system. The proposed system helps tourists and local people to search the route of trip based on user-interested categories with the minimum road network distance in real world. The system also finds a meeting place for two users in different locations.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

In today's digital world, there are many location-based mobile guide applications to point out the route for the trips, to track movements, to find the famous places, to know the current location and what is around you. Such applications make the user's life easier through offering a real-time guidance. Therefore, the number of Location-based service (LBS) mobile users has steadily increased over the past several years. Even though there is a vast amount of different LBS mobile guide applications developed already, new applications are still needed as majority of existing solution focuses on mobile tourism services.

Since tourism has been developing very rapidly from time to time and particularly in the 21st century, more sophisticated LBS mobile guide applications are required to provide up-to-date information about their current location for tourists in a geographic information system(GIS) environment. GIS is a system designed to capture, store, update, manipulate, analyze, manage, and present all forms of geographically referenced data on Earth's surface. Any information that includes location can be used in GIS. Moreover, GIS has been demonstrated as technology that can effectively solve location problems.

Location-based service in GIS is a great potential application for the mobile communications. Finding the location of the mobile phone is one of the important issues of the modem mobile communication system. LBS are IT services for providing information that has been created, compiled, selected, or filtering taking into consideration the current locations of the users or those persons or mobile objects. The user's current location can be obtained by using positioning technologies. In this system, users can get the route information for the trip that starts from user location to famous places. This trip planning system is focused on the Bagan region.

Bagan is one of the interesting places for tourists because there are many ancient pagodas in that place. And it is an ancient city located in the Mandalay Region of Myanmar and is the site of the many collections of Buddhist pagodas in the world. The main tourist destination in Myanmar is Bagan, capital of the first Myanmar Empire; one of the richest archaeological sites in South-east Asia. Moreover, this site also shows the development in architecture and design over the several centuries. Tourists, who visit Bagan, are requiring the Location-based Services to systematically visit many nearest tourist-interested famous places. Therefore, location-based mobile guide application for Bagan is developed to point out the route for trip in this research.

1.1 Objectives of the Thesis

The main objective of thesis is to systematically plan the trips for both tourists and local people by using the partial sequenced route algorithm. Other objectives are

- To study how the Partial Sequence Route is designed and applied for trip planning
- To provide the convenient trips for both tourists and local people
- To know how to benefit in trip planning by using the Partial Sequenced Route
- To find a meeting point for two users in different locations
- To implement a practical application for trip planning

1.2 Motivation of the Thesis

There are many ancient pagodas in Bagan, so most tourists are interested in studying the architecture and design of pagodas. But, tourists can have the route problem due to many collections of pagodas in that place. So, the trip planning system is made to systematically visit to many pagodas with the minimum travel distance. In this system, both tourists and local people can plan to visit the Points of Interest (POIs) categories (i.e; Pagodas, a restaurant and a hotel) in Bagan.

1.3 Overview of the Proposed System

The proposed system is deployed on an android operating system which is an open source technology. Design of the location-based mobile guide for Bagan is as follows:

If the route is searched for trip, the system finds the sequence route based on the user-selected categories from user's current location by using the Nearest Neighbor-based

Partial Sequenced Route (NNPSR) Algorithm. And the system points out the route for the trip not only in details but also on map.

If the meeting place is searched, the system finds the place to meet for two users in different locations. And the meeting place, the user location and friend location are showed on the map.

1.4 Organization of the Thesis

The organization of thesis is as follows:

Chapter 1 introduces the location-based mobile guide for Bagan, objectives of the thesis, motivation of the thesis and overview of the system. Chapter 2 presents the methods which may be used for a problem instance of the trip planning query. Chapter 3 gives the proposed algorithm with the details of its step. Chapter 4 describes the implementation of the system and experimental results. The last chapter is Chapter 5, and there are Conclusion, System Limitation and Further works in it.

CHAPTER 2

BACKGROUND THEORY

Nowadays, there are a number of online trip planning systems that make it possible to automatically generate a selection and routing plan for visiting Points of Interest (POIs) that satisfy the tourist personal interests. Tourists who visit to a city or region during a trip, are not possible to visit every POIs available that exist in that particular area during a constrained travel time and budget. Thus, they have to plan their trips by selecting some POI categories that they more interest. And, the location-based mobile guides are required to plan the trip according to tourists' preference.

The tourism sector is one of the country's most important economic sectors, and the increasing popularity of mobile devices presents an opportunity for developing mobile tourism services. The functions supported by mobile tourist guides are related to finding relevant attractions and services, or supporting the exploration of an area. To satisfy tourists' needs (i.e.; maps, personal navigation systems and location-based mobile guides), more sophisticated tourism information technologies are required. Today, there is a vast variation of different mobile solutions already developed to support travelers before, during and after the trip, for instance in a form of city attractions, sightseeing, exhibition or museum guides. However, majority of these solutions focus either on recommending tourist attractions or on providing some tourist services.

As pointed out in [8], mobile devices are becoming a primary platform for information access. More and more people use these communication and information access tools, and the functionalities and the challenges provided by these devices are growing. Mobile tourism is an emerging trend in the field of tourism and involves the use of mobile devices as electronic guides [9]. Mobile guides (based on Personal Digital Assistant (PDAs), smartphones and mobile phones) play an increasingly important role in tourism, giving tourists ubiquitous access to relevant information especially during the trip [10].

There are many cultural tourist attractions that are historical places, monuments, museums or art galleries, ancient temples, castles and libraries. Because Bagan is a historical place where has many cultural tourist attractions, it is an attractive place for tourists. So, the location-based mobile guide application is developed to point out the trip based on the tourism personal interests in Bagan Region.

2.1 Location-based Services

Location-based services (LBS) are the information, entertainment or security services, which are accessible with mobile devices through the mobile network and which use information on the geographical position of the mobile devices. Because LBS are largely dependent on the mobile user's location, the primary objective of the service provider's system is to determine where the user is.

LBS uses positioning, telecommunication, and mobile computing technologies to deliver information and assistance to users based on their geographic position. To discover the position of the mobile, LBS must use positioning methods in real time. Locations can be represented in spatial terms or as text descriptions. A spatial location can be represented in the latitude longitude- altitude coordinate system. Latitude is defined as 0-90 degrees north or south of the equator and longitude as 0-180 degrees east or west of the prime meridian, that passes through the Greenwich, England. Altitude is represented in meters above sea level. A text description is usually defined as a street location, including city. The location of the device can be retrieved by using

(1) Mobile Phone Service Provider Network

(2) Satellites

(1) Mobile Phone Service Provider Network

The current cell ID is used to locate the Base Transceiver Station (BTS) that the mobile phone is interacting with and the location of that BTS. It is the most basic and cheapest method as it uses the location of the radio base station that the cell phone is connected to. The cell ID is a generally unique number used to identify each BTS or sector of a BTS within a Location area code. It can locate any device with a GSM/WCDMA/CDMA modem. The good side is that it is low cost and the infrastructure is already there. The downside is that it is not as accurate as other geopositioning technologies – all depending on the proximity of base transceiver stations. A GSM

(Global System for Mobile communication) cell may be anywhere from 2 to 20 kilometers in diameter.

(2) Satellites

The Global Positioning System (GPS) uses a constellation of 24 satellites orbiting the earth. GPS finds the user position by calculating differences in the times the signals, from different satellites, take to reach the receiver. GPS signals are decoded, so the smart phone must have in-built GPS receiver. Assisted GPS (A-GPS) is the new technology for smart phones that integrates the mobile network with the GPS to give a better accuracy of 5 to 10 meters. This fixes the position within seconds, has better coverage and can be used inside the buildings, consumes less battery power and requires fewer satellites.

LBS Applications

LBS applications can be categorized as:

- Navigation: Showing how to reach the specified target location such as route description
- Emergency: Nearest medical center, Emergency calls, Warning about an unsafe areas
- Entertainment: Social networking, Mobile games
- Information services: requesting the tourist guide, traffic information and city guide
- Marketing: location-based mobile advertising
- Tracking: tracking of stolen property/ vehicles, Transportation and logistics, person surveillance

2.2 Google Earth Service

Today, Google Earth is among the most popular geospatial software in the world. It is a free program from Google that allows to "fly" over a virtual globe and view the Earth through high-resolution graphics and satellite images that include cities and landscape of many geographic areas. In addition, it can be used to interactively create and display custom data and geographic data. One of the most useful aspects of Google Earth from a geoscience education point of view is the availability of a variety of geoscience-related datasets.

A digital elevation model within Google Earth allows the users to perform some basic measurements (latitude and longitude, elevation, and size), which has led some users to consider it a variety of GIS software. The globe can be used by entering addresses or coordinates.

The Three Versions of Google Earth

- Free - Intended for home and personal use, this product has many features, including displaying satellite and aerial imagery, a growing set of layers of mappable data, the ability to display third party data, tools for creating new data, and the ability to import GPS data.

- Pro - This version, developed for commercial use, adds movie making, as well as importing ESRI shapefiles and MapInfo tab files, can measure areas of circles and polygons, and can print and save high-resolution images.

- Enterprise - This product makes imagery and other geospatial data available to employees within organizations such as corporations.

Each of these versions of Google Earth can be used to read and create data in KML (Keyhole Markup Language) format, which enables educators, students, and other users to share data. In the proposed system, the route for the trip is presented on the map of Google Earth together with the location information from current location to user-selected categories.

2.3 Spatial Database

A spatial database is a database that is enhanced to store, query and access spatial data or data that defines a geometric space. These data are often associated with geographic locations and features, or constructed features like cities. Spatial database systems offer the underlying database technology for geographic information systems and other applications. The spatial operations are also included to be applied on such kinds of data like distance, area, perimeter, direction, and overlap of geometries. Data on spatial databases are stored as

- Coordinates: latitude and longitude coordinates
- Points: a point of interest, a house, a monument
- Lines: a road segment, a road network
- Polygons: a county, a voting area

In the spatial database of the proposed system, road network and POIs datasets of Bagan are contained. Networks in GIS operate on the line features, but it also includes the surrounding area and the associated attribute. Networks consist of a system of roads, water pipelines, railways, rivers, telephone lines, electric lines, city bus routes and other utility network. These networks can be modeled to develop life of people in the geographical information system (GIS). Routes and network are the interconnected features that are used for searching the route on the road network.

Road Networks are an important part of our everyday movement from place to place. A network is essentially a set of lines known as segments or edges connected or joined by a set of vertices known as nodes or junctions. The road network datasets are of high interest in several fields (e.g. traffic visualization, route planning, navigation systems and etc.). Both road network and POIs datasets with their attributes are stored in the spatial database of the proposed system.

2.4 Difference between Euclidean and Network Distance

Most of the research on spatial databases did focus on Euclidian information when processing spatial queries [4]. The Euclidean distance is the "ordinary" straight-line distance between any two points that can be calculated by only using the coordinates of the two points in Euclidean space. The road network is represented as a graph consisting of a set of nodes (junctions) and links (roads). Network distance is the distance between any two points in the network via the network, that is, the minimum distance one has to travel in the network to reach the second point from the first. Two points in a network may have multiple routes connecting them, but the most interesting route is the one which has the smallest total length and is thus called the shortest route. Figure 2.1 shows difference between Euclidian distance and Network Distance.

In figure 2.1, given a query point q and a set of static POI categories (eg; hotels) on the road network, a nearest neighbor (NN) query finds one hotel closest to q in both Euclidean and Network distance. The numbers in the figure correspond to the network distances. The nearest neighbor of q will be hotel A in road network distance and hotel B in Euclidean distance. In network distance, since q's movement is constrained, which means that q has to follow a path on the network when moving and might not just move freely to hotel B, which actually is the hotel with highest network distance to q, hotel A would be the appropriate answer to this nearest neighbor query.

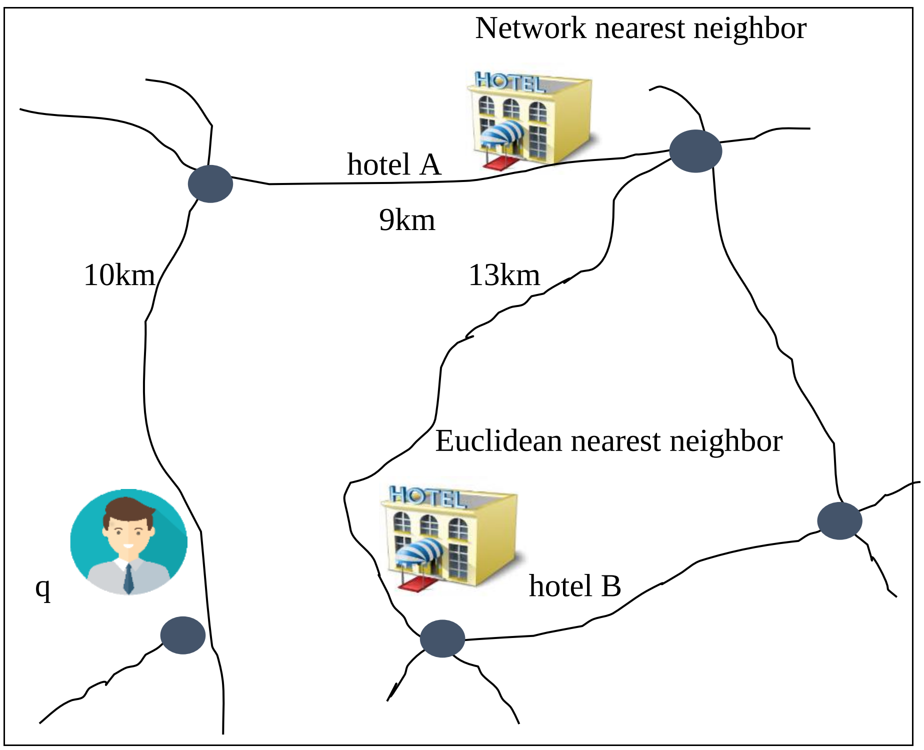

Figure 2.1 Euclidean Vs. Network Distance

Euclidean distance is not an appropriate distance measure for many real-world applications where users can only move on predefined paths, such as streets on a road network. Furthermore, there might exist multiple paths between two points on a road network. Therefore, simply verifying distances is not sufficient for queries on road networks.

In order to demonstrate a practical application effectively, the network distance is used as the distance measure. So, the proposed system uses the real-world road network distance to search the sequence route for the trip.

2.5 Methods for solving the Partial Sequenced Route

There are three algorithms of the Partial Sequenced Route query

1. NNPSR with the Light Optimal Route Discoverer (NNPSR-LORD)
2. An Advanced A* Search-based Partial sequenced route (AASPSR)
3. **3.** The Nearest Neighbor-based Partial Sequenced Route (NNPSR)

2.5.1 NNPSR with the Light Optimal Route Discoverer (NNPSR_LORD)

The Light Optimal Route Discoverer (LORD) algorithm [3] can guarantee to retrieve a route of minimum distance. Since a complete POI sequence after each execution of the NNPSR algorithm is obtained, the trip by applying LORD on the POI sequence found by NNPSR can be further optimized. LORD is a threshold-based algorithm and requires less memory space compared with Dijkstra's shortest path solution.

The first step in LORD is to issue consecutive nearest neighbor queries to find the greedy route that follows the given POI category sequence from the starting point. Then, the length of the greedy route becomes a constant threshold value T_c. In addition, LORD also keeps a variable threshold value T_v whose value reduces after each iteration and LORD discards all the POIs whose distances to the starting point are more than T_v.

Afterward, LORD iteratively builds and maintains a set of partial sequenced routes in the reverse sequence (i.e., from the end points toward the starting point). During each iteration of LORD, POIs from the following category are added to the head of each of these partial sequence routes to make them closer to the starting point. The two thresholds are utilized to prune non-promising routes for reducing the search space.

2.5.2 An Advanced A* Search-based Partial Sequenced Route (AASPSR)

Although the NNPSR and NNPSR-LORD algorithms do not consider the location of the destination when greedily generating the route sequence, AASPSR algorithm considers the start point (S) and the destination (D) of the trip. The A* search based Partial Sequenced Route (ASPSR) algorithm considers the location of the destination in its function and limits the trip planning within a range defined by S and D (e.g., an ellipse whose two focal points are S and D).

In ASPSR, the POI p is retrieved with the minimum cost of $Dist_E(S, p) + Dist_E(p, D)$ in each category included in L_{zero} (the user selected categories with a zero count). Afterward the POI p with the lowest cost is added into the route list L_{route} (A list of the POI sequence of a trip plan) and the category of p is withdrawn from L_{zero}. Then L_{zero} is be updated and the location of p is set as the new query point. The process is reiterate until all the user selected categories are covered.

However, there could be roundabout ways when a trip is planned by ASPSR. Because the POIs which are closer to the major axis of the ellipse (S and D are the two focal points) have a lower distance cost, POIs is picked sequentially in ascending order of their costs. Consequently, a detour is occur where the user has to travel far away from D to visit POI p before reaching D at last. Therefore, ASPSR is improved to solve the above problem.

The improved version of ASPSR is named as the Advanced A* Search-based Partial Sequenced Route query (AASPSR) algorithm. AASPSR(k) is used to denote AASPSR algorithm with parameter k to specify the number of POIs which retains for each category. It can be considered as a hybrid scheme to combine ASPSR and NNPSR.

AASPSR(k) first computes $C_i.P^*$ for each category C_i in C such that every POI in $C_i.P^*$ is a POI with the top-k minimum traveling distance sum from S to D in C_i. In particular, if $k = 1$, only one POI with the shortest traveling distance sum from S to D for C_i is added to $C_i.P^*$. On the contrary, if $k = \infty$, then all the POIs on the underlying road network is included in $C_i.P^*$ for each C_I.

After $C_i.P^*$ has been generated for each category, NNPSR is launched to generate a route only on these selected POIs in each $C_i.P^*$. Starting with S, the nearest POI p is searched in $C_i.P^*$ ($C_i \in L_{zero}$). Afterward, p is inserted into L_{route} and the location of p is used as the query point of the nearest neighbor query. Next the category of p is removed from L_{zero}. The whole process is repeat until L_{zero} becomes empty. The complete algorithm of AASPSR(k) is illustrated in [1].

2.5.3 The Nearest Neighbor-based Partial Sequenced Route (NNPSR)

NNPSR uses the nearest neighbor (NN) query to generate a trip satisfying the user-selected categories. There are many different kinds of NN queries in geographical information system such as range nearest neighbor (RNN) query, k nearest neighbor (KNN) query, Continuous K nearest neighbor queries (C-KNN), aggregate nearest neighbor (ANN) query, Obstructed Nearest Neighbor (ONN) and so on. Above queries use either Euclidean distance or road network distance to find nearest point. NNPSR uses road network distance to find the sequence route for the trip.

NN search on road networks has many real world applications. Given a query location and a set of static objects (e.g., pagoda, restaurant, and hotel) on the road network, the NN finds the closest spatial objects to the query location. The objective of the NN query is to find the object(s) with the minimum distance to a query point (user location). Although searching one location by using NN query is useful, sometime the users really want to search a route of several (and possibly different types of) locations in some sequence for their trip with the minimum total traveling distance . In modern geographic information systems, searching the route for the trip represents an important class of NN queries. In route query related applications, users may want to define a number of traveling rules (traveling preferences) when they plan their trips.

In [1], NNPSR algorithm applies Activity-On-Vertex (AOV) Network to guide the search to retrieve a near-optimal route satisfying all the traveling rules in the road networks. And it launches successive nearest neighbor queries to answer a partial sequenced route (PSR) query. The following theorem provides the relationship of an AOV network and the compatibility of the traveling rules.

Theorem 1 *The partial sequence rules are compatible if and only if the corresponding AOV network is a directed acyclic graph.*

If the set of rules is not a directed acyclic graph, algorithm reports cycles in the set of travelling rules. After having the AOV networks, NNPSR is launched to compute a trip plan satisfying all the user-defined traveling rules. It can only picks a single POI in each category, and can’t search the multiple POIs in each category [1].

In the proposed system, NNPSR algorithm can search the multiple POIs in each category. And, there are two priorities to choose categories as the user-defined traveling rules for a trip. The details of NNPSR algorithm are widely described in chapter 3.

2.6 Meeting Place Searching

Aggregate Nearest Neighbor (ANN) queries developing from Nearest Neighbor (NN) queries are the relatively new query type in spatial database. There are many ANN queries to find a meeting place such as continuous aggregate nearest neighbor, Exact and Approximate Flexible Aggregate Similarity Search, K-Nearest Neighbor Temporal Aggregate Queries, group nearest neighbor, Aggregate center queries. Because of the multiple query points, ANN queries are much more complex than NN queries.

In the proposed system, the meeting place is also searched for two users in different locations by using Aggregate center query. The details for the meeting place searching of the proposed system are described in chapter 3.

CHAPTER 3

DESIGN OF THE PROPOSED SYSTEM

3.1 The Nearest Neighbor-based Partial Sequence Route (NNPSR)

The partial sequenced route query enables a user to plan a trip which starts from the current location, goes through a sequence of points-of-interest (POIs), and ends at a destination location. Given n disjoint sets of POI category {C1, …..,Cn}, each containing a number of POIs, the PSR query is to search for a route that satisfies the following three requirements:

1. The route will traverse through at least one POI in each category that the user selects.
2. The total traveling distance is minimized.
3. The route conforms with the user-selected categories.

The Nearest Neighbor-based Partial Sequence Route is defined as finding the sequenced route that satisfies the user-defined traveling rule by implementing nearest neighbor (NN) search. NNPSR uses the NN search (incremental network expansion) to find a nearest point of interest. The incremental nearest neighbor query searches the neighbor POIs by gradually enlarging the search area of road network from the query location. The POIs of each category can be found by expansion the segments on the road network. The search is terminated when a specified POI number of categories has been found. The NNPSR network model not only finds out the nearby categories but also indicates the sequenced route to be travelled by the user. Finally, a POI sequence route of categories is obtained for a trip with the minimum network distance. The result sequence route is presented by highlighting the areas of interest on the map display.

In the proposed system, NNPSR algorithm is devised by utilizing both the L_{zero} list and the nearest neighbor (NN) search (i.e., the incremental network expansion [4] based implementation) to generate a trip satisfying the user-selected categories. There are two priorities to choose categories as the user-defined traveling rules for a trip in the proposed system. If user chooses pagoda category of first priority or second priority, the number of pagoda that the user wants to go, can be selected at least one to five pagoda in spinner box and is stored in P_{no}. All the number that stored in P_{no} are searched to include in the route of trip by using NN search.

With NNPSR, the nearest POI of category is first searched to the query point q (as the starting point) whose category is included in L_{zero}. The retrieved nearest POI P_{NN} is stored in a route list L_{route} and the category of P_{NN} (i.e., P_{NN}.C) is removed from L_{zero}. In addition, the query point q is also updated to the location of P_{NN}. The process repeat until the user-selected categories are contained in the route list L_{route}. The pseudo code of NNPSR algorithm is described in Figure 3.1. The Symbolic Notations of NNPSR algorithm are described in Table 3.1.

Nearest Neighbor (NN) Search in the Road Network

In real-word, users usually move only in the underlying road networks (i.e., street) rather than traveling freely through obstacles (e.g., buildings, rivers, etc.). So, when searching the route for the trip, the real-world road network distance is used in this system. The nearest neighbor (NN) query is a very important query type for supporting Geographic Information System applications and has many real life applications, such as map services.

NN search is defined as finding the nearest POI with the minimum network distance to a query point. And it is demonstrated by applying the incremental network expansion technique [4] in the proposed system. Given a query location *q* and a set of static POIs, the incremental nearest neighbor query finds the nearest POI on the road network by simply expansion the path from one junction node to the next until the nearest POI on road segments is reached.

Network expansion is performed from *q*. First, $n_i n_j$ (the road segment) that covers *q* (query point) is identified and all POIs on $n_i n_j$ are retrieved. If there exists at least one POI on $n_i n_j$, POI p_k with the smallest network distance d_N (q, p_k) is updated as P_{NN} (the nearest neighbor point), and d_N (q, p_k) is added into d_{Nmax} (distance). This distance offers an upper bound to restrict the search space. If no POI is found on $n_i n_j$, zero is added into p_k and infinity is added into d_{Nmax}.

Then a priority queue, Q =< (n_i ,d_N (q, n_i)), (n_j , d_N (q, n_j)) >, is initiated with nodes and the accumulated distance from q. The node n which is closest to q is de-queued

```
Nearest Neighbor-based Partial Sequenced Route query(C, S)
Set L route = Ø and q= S ;
Add the user-selected categories of C into L zero ;
Add the number of POI for each user selected categories into P no ;
while L zero ≠ Ø do
    P = Ø;
    for each Ci ∈ L zero do
        P = Ci .P ∪P ;
    end for
  while  P no ≠  0 do
    Identify the road segment ni nj covering q;
    Find all the POIs in P on ni nj;
    if there exists at least one POI in P on ni nj then
        Update PNN with the POI pk with the smallest dN (q, pk);
        d Nmax =    dN (q, pk);
    Else
        pk =  Ø ,   d Nmax =  ∞  ;
    end if
    Q =< (ni , dN (q, ni )), (nj , dN (q, nj ))>;
    De-queue the node n in Q with the smallest dN (q, n);
    while dN (q, n) <   d Nmax do
        for each non-visited adjacent node nk of n do
            Find all the POIs in P on the road segment n nk ;
            Update PNN from the POI pk with the smallest network distance found so far  ;
            Update d Nmax with dN (q, pk);
            En-queue (nk, dN(q, nk)) in Q;
        end for
        De-queue the node n in the updated Q with the smallest dN (q, n);
    end while
    L route = L route ∪ PNN   ;
    q = PNN .L;
    P no = P no -1  ;
    Update  P no;
  end while
  Remove PNN .C from L zero;
  Update L zero,  P no;
end while
return L route
```

Figure 3.1 Algorithm for Nearest Neighbor-based Partial Sequenced Route (NNPSR)

Symbol	Meaning
C	The set of all the user selected categories
P	A set of POIs
Q	The priority queue
S	The starting point of a NNPSR query
q	The query point of a nearest neighbor query
n	Junction node
POI	Point of Interest
P_{no}	The number of POI in each category
d_{Nmax}	Distance to restrict the search space
$P_{NN}.C$	The category of a POI P_{NN}
$P_{NN}.L$	The location of a POI P_{NN}
C_i	A POI category
$C_i.P$	All the POIs of a category
L_{zero}	the user selected categories with a zero count
L_{route}	A list of the POI sequence of a trip plan
P_{NN}	The query result of a nearest neighbor query
$d_N(x, y)$	The network distance between points x and y
PSR	The partial sequenced route

Table 3.1 Symbolic Notations

in Q with the smallest $d_N(q, n)$. If $d_N(q, n)$ is less than d_{Nmax}, expansion process is started for each non-visited adjacent node n_k of n. All the POIs on the road segment n n_k is searched. P_{NN} and d_{Nmax} is updated and (n_k, $d_N(q, n_k)$) is en-queued in Q. The node n in the updated Q with the smallest $d_N(q, n)$ is de-queued. By processing iteratively above steps, when $d_N(q, n)$ is greater than d_{Nmax}, the algorithm terminates and returns P_{NN} as the nearest neighbor to *q* with a network distance of d_{Nmax}.

3.2 Meeting Place for the Proposed System

Aggregate Nearest Neighbor (ANN) queries return the meeting place which minimizes the total traveling distance for two users in different locations. For finding a meeting place in the system, users need to choose meeting place at, the friend’s location and type. After choosing, midpoint is calculated for two users. And a meeting place that is near to midpoint is searched in spatial database. And, three points: friend’s location, user location, and the convenient place to meet for two users are displayed on the map.

3.3 Overview of the Proposed System Architecture

In the proposed system, there are three categories which users can select for their trip. They are the pagodas, the restaurants, and the hotels. The proposed system architecture is shown in Figure 3.2.

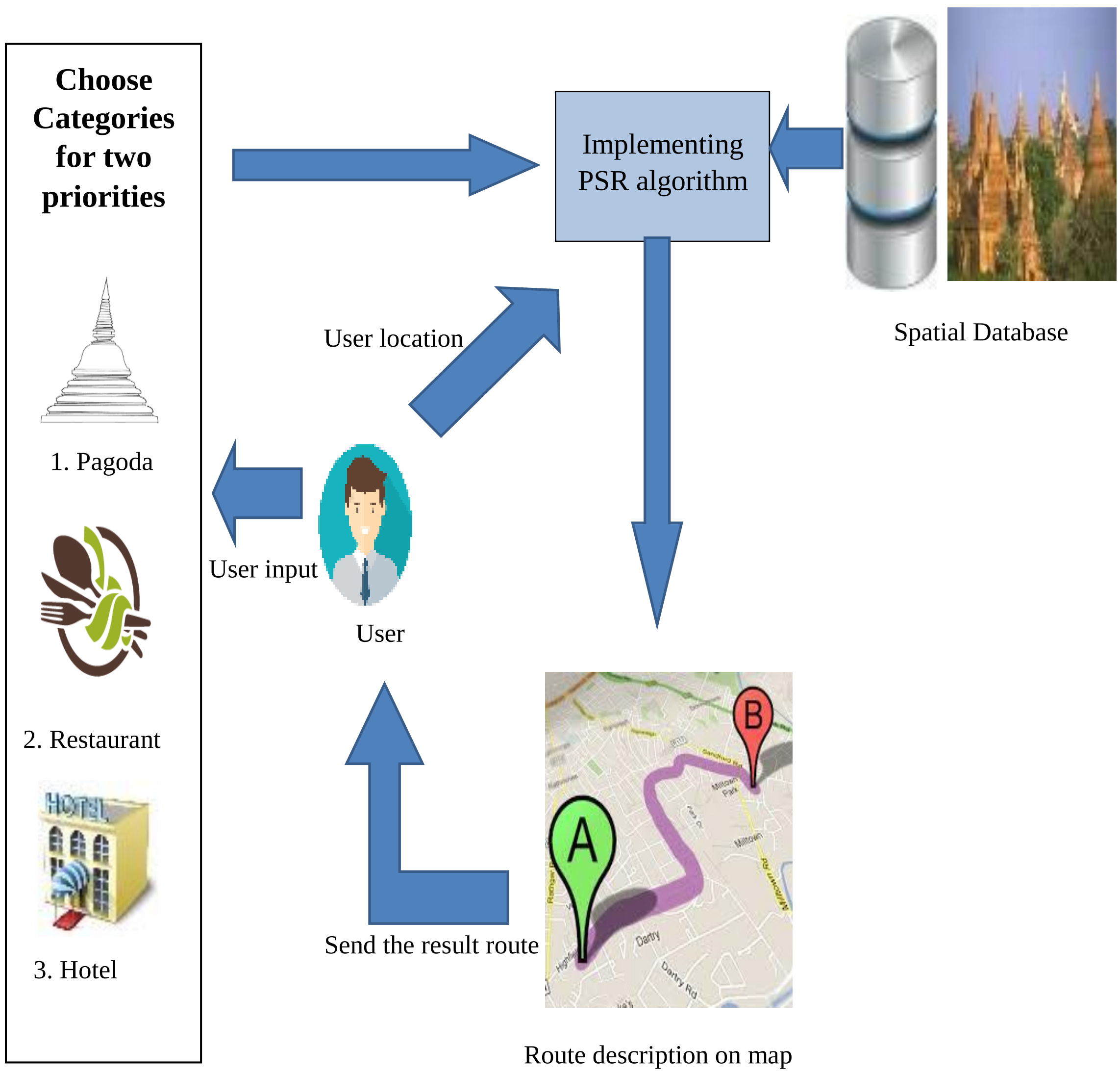

Figure 3.2 System Architecture for Bagan Trip Planning System

In the start of the system, users can choose categories for two priorities. And then, the partial sequenced route (PSR) algorithm is applied with two inputs of the user-selected categories and user's current location. Data from the spatial database of Bagan Region are used to implement PSR algorithm. Finally, the route for trip is shown on map.

3.4 System Flow Diagram

In the proposed system, the user's current location and the user-selected categories are inputs of the system. And, the sequenced route with minimum network distance for the trip is output of the system. The detailed process of the system flow diagram is shown in Figure 3.3.

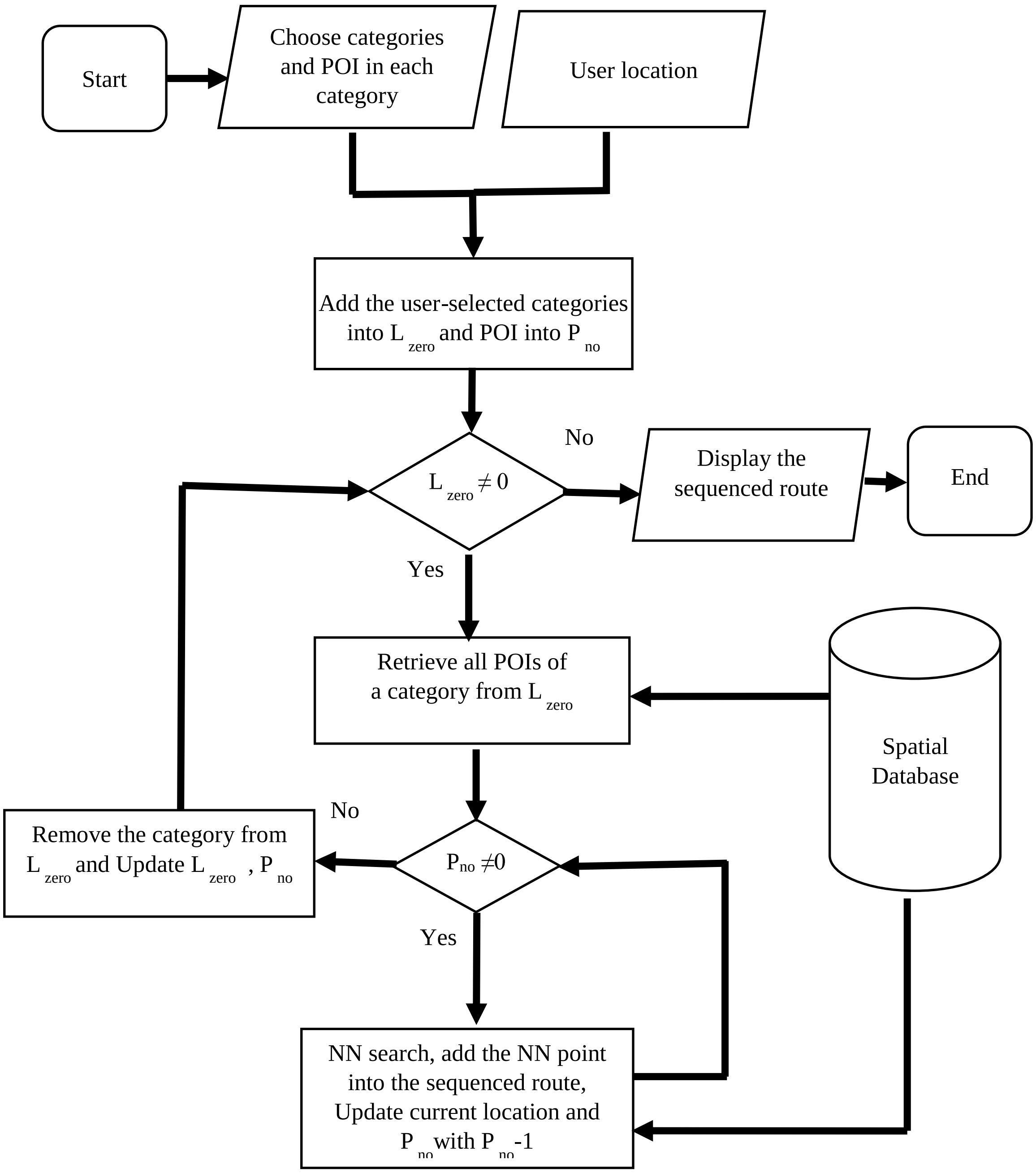

Figure 3.3 System Flow Diagram

3.5 Bagan Region for the Trip Planning System

NNPSR Algorithm is applied with the dataset of Bagan Region as shown in Figure 3.4.

1. New Bagan
2. Old Bagan
3. Nyaung U
4. Myin Ka Bar
5. West Pwazaw Village
6. Minnanthu Village and
7. Aung Myay Thar Quarter

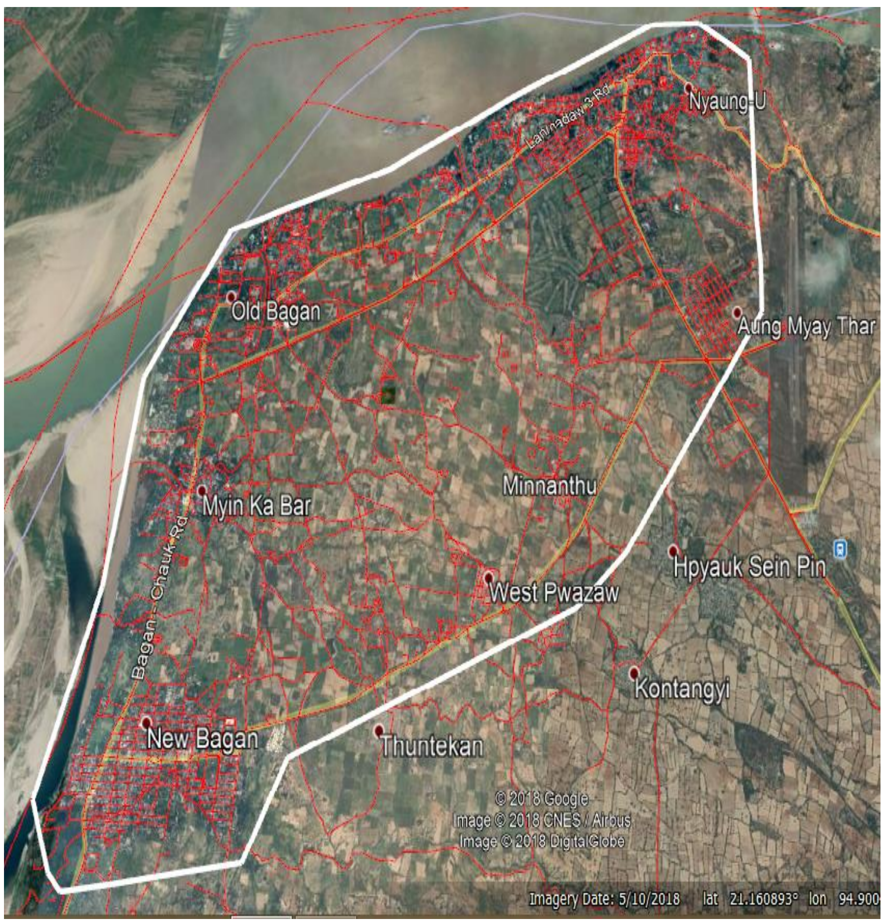

Figure 3.4 Bagan Region for the proposed system

3.6 Real Dataset of the Bagan Region

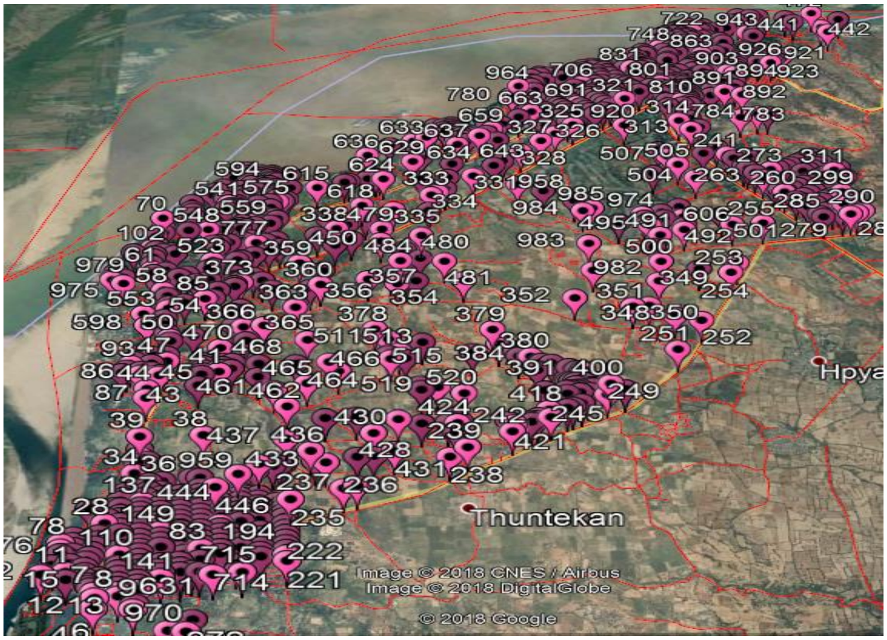

Figure 3.5 Road Network

The road network of Bagan that includes the junction nodes, the road segments, and distances is shown in Figure 3.5.

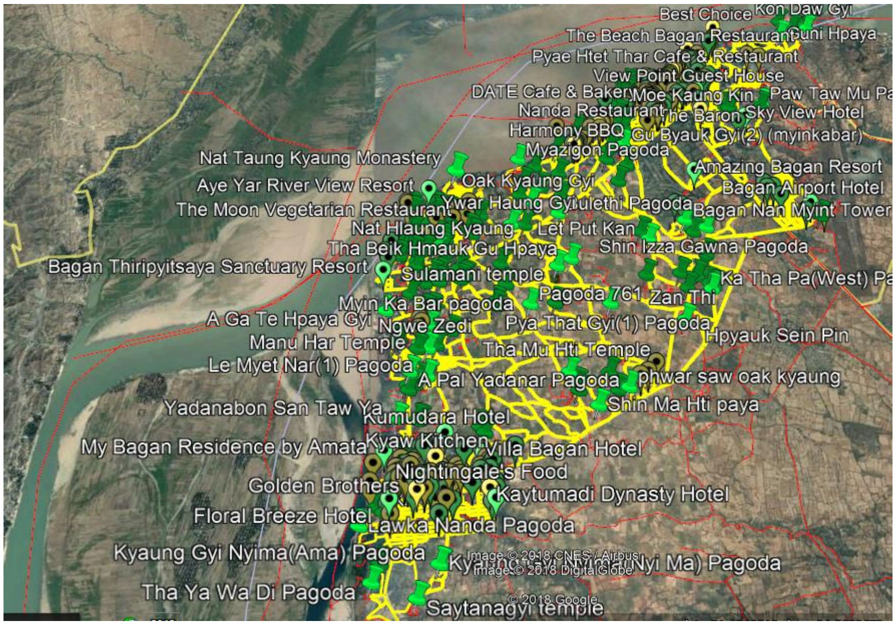

Figure 3.6 Points of Interest Distribution

The distribution of Pagodas, Restaurants and Hotels on the road network of Bagan is displayed in Figure 3.6.

3.7 Database Design for Road Network and Categories

A road network can be modeled as an undirected weighted graph G = (V,E,W), where V is the set of vertices (junction nodes) , E is the set of edges (road segments) between two nodes in V. And, W associates each edge to a positive real number. Interesting objects (POIs) are located on edges e Є E. In this system, the distance is measured with kilometer scale.

Therefore, the junction nodes, road segments, and points-of-interest (POIs) (i.e; Pagodas, restaurants and hotels) with their attributes are stored in the spatial database for the trip planning of the proposed system. The database contents are shown in Figure 3.7.

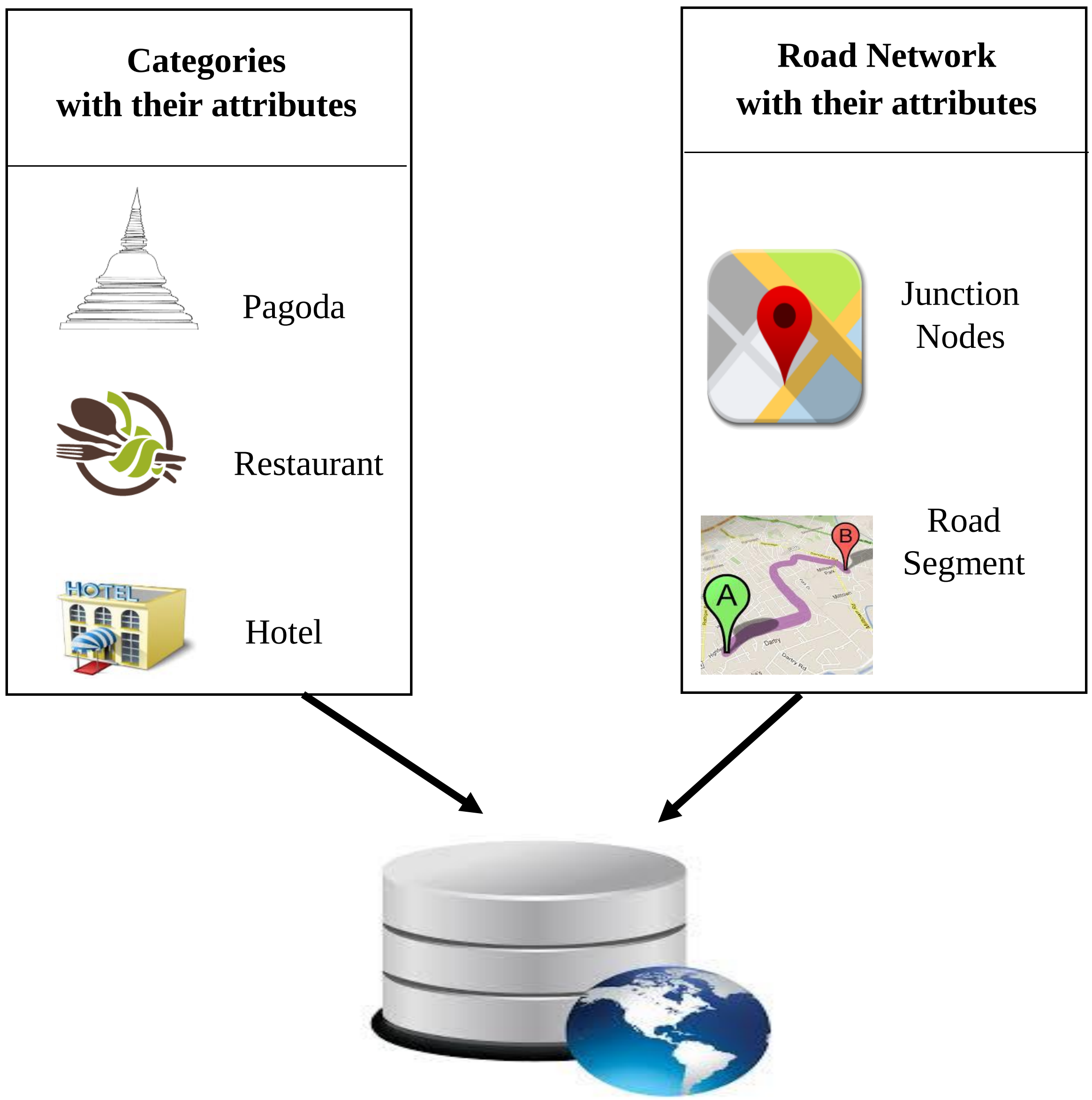

Figure 3.7 Contents of Spatial Database

3.7.1 Creating Dataset for the System

To create the database, the data are collected from Bagan Region only. Firstly, about 461 points-of-interest are collected for three categories of places. And, about 986 junction nodes and 1652 road segments are collected to implement this proposed system. The data are obtained from Google Earth, Trip Advisor, Myanmar travel and tour websites. In this system, the database contains the twelve tables named 'start_junction_node', 'end_junction_node', 'road_segment', 'poi_pagoda', 'poi_hotel' 'poi_restaurant', 'segment_pagoda', 'segment_restaurant', 'segment_hotel', 'PagodaTo Pagoda ', 'PagodaToRestaurant' and 'HotelToRestaurant'.

Start and end junction node tables are the main tables, and store start nodes and end nodes with location (latitude and longitude) to define road segments of Bagan Region. Road segment table is the main table, and stores start node, end node and distance of segment. The poi pagoda, restaurant and hotel tables are the main tables, and store name and location (latitude and longitude) of pagoda, restaurant and hotel. The segment pagoda, restaurant, and hotel tables store segment, pagoda, start node and distance to retrieve the distance from start node of segment to pagoda, restaurant, and hotel. PagodaToPagoda table stores segment, pagoda and distance to retrieve the distance between two pagoda on the same segment. PagodaToRestaurant table stores segment, pagoda, restaurant and distance to retrieve the distance between pagoda and restaurant on the same segment. HotelToRestaurant table stores segment, hotel, restaurant and distance to retrieve the distance between hotel and restaurant on the same segment.

Table 3.2 shows some of the dataset for the "start_junction_node" table in the database.

Table 3.2 The "start_junction_node" table

start_node_ID	latitude	longitude
1	21.117305°	94.856035°
2	21.119366°	94.859577°
3	21.118622°	94.855077°
4	21.119498°	94.854833°
5	21.119510°	94.854282°
6	21.125145°	94.855868°

start_node_ID	latitude	longitude
7	21.125760°	94.855757°
8	21.125256°	94.857655°
9	21.125753°	94.857622°
10	21.127231°	94.853467°
11	21.128061°	94.853390°
12	21.124013°	94.855822°
13	21.123818°	94.858905°
14	21.126256°	94.855445°
15	21.126778°	94.855176°
16	21.126776°	94.855516°
17	21.126571°	94.855274°
18	21.127353°	94.854931°
19	21.127924°	94.854864°
20	21.128500°	94.854799°
21	21.129190°	94.854897°
22	21.129700°	94.855130°
23	21.130217°	94.855271°
24	21.130840°	94.855446°
25	21.131411°	94.855574°
26	21.132082°	94.855699°
27	21.132757°	94.855868°
28	21.133325°	94.856014°
29	21.133950°	94.856175°
30	21.135137°	94.856720°
31	21.135907°	94.857097°
32	21.136300°	94.857289°
33	21.136878°	94.857448°
34	21.139097°	94.857978°
35	21.140324°	94.858152°
36	21.141146°	94.858304°
37	21.142000°	94.858254°
38	21.143534°	94.863368°
39	21.143594°	94.858112°

Table 3.3 shows some of the dataset for the "end_junction_node" table in the database.

Table 3.3 The "end_junction_node" table

start_node_ID	latitude	longitude
1	21.117305°	94.856035°
2	21.119366°	94.859577°
3	21.118622°	94.855077°
4	21.119498°	94.854833°
5	21.119510°	94.854282°
6	21.125145°	94.855868°
7	21.125760°	94.855757°
8	21.125256°	94.857655°
9	21.125753°	94.857622°
10	21.127231°	94.853467°
11	21.128061°	94.853390°
12	21.124013°	94.855822°
13	21.123818°	94.858905°
14	21.126256°	94.855445°
15	21.126778°	94.855176°
16	21.126776°	94.855516°
17	21.126571°	94.855274°
18	21.127353°	94.854931°
19	21.127924°	94.854864°
20	21.128500°	94.854799°
21	21.129190°	94.854897°
22	21.129700°	94.855130°
23	21.130217°	94.855271°
24	21.130840°	94.855446°
25	21.131411°	94.855574°
26	21.132082°	94.855699°
27	21.132757°	94.855868°
28	21.133325°	94.856014°
29	21.133950°	94.856175°
30	21.135137°	94.856720°

start_node_ID	latitude	longitude
31	21.135907°	94.857097°
32	21.136300°	94.857289°
33	21.136878°	94.857448°
34	21.139097°	94.857978°
35	21.140324°	94.858152°
36	21.141146°	94.858304°
37	21.142000°	94.858254°
38	21.143534°	94.863368°
39	21.143594°	94.858112°
40	21.151332°	94.862875°
41	21.151639°	94.864344°
42	21.152057°	94.864392°
43	21.148457°	94.858337°
44	21.149654°	94.858330°
45	21.151386°	94.859245°
46	21.151594°	94.859370°
47	21.153271°	94.859815°
48	21.154068°	94.859775°
49	21.164168°	94.861809°
50	21.156444°	94.859868°
51	21.157121°	94.859610°
52	21.158323°	94.859287°

Table 3.4 shows some of the dataset for the “road_segment” table in the database.

Table 3.4 The “road_segment” table

segment-ID	start-node-ID	end-node-ID	distance
1	1	2	0.5
2	1	3	0.18
1518	1	617	0.94
1	2	1	0.5
3	2	13	0.52
1519	2	969	0.15
2	3	1	0.18

segment-ID	start-node-ID	end-node-ID	distance
4	3	4	0.11
1520	3	617	0.11
4	4	3	0.11
5	4	5	0.06
6	4	12	0.5
5	5	4	0.06
7	5	17	1.02
1521	5	617	0.39
1522	6	7	0.07
1523	6	8	0.18
8	6	12	0.12
1522	7	6	0.07
9	7	9	0.2
10	7	14	0.06
1523	8	6	0.18
1524	8	9	0.06
11	8	625	0.18
9	9	7	0.2
1524	9	8	0.06
1525	9	103	0.06
12	9	631	0.16
13	10	11	0.1
14	10	18	0.15
1526	10	952	0.5
13	11	10	0.1
15	11	72	0.1
1527	11	952	0.2
6	12	4	0.5
8	12	6	0.12
16	12	13	0.37
3	13	2	0.52
16	13	12	0.37
18	13	625	0.05
10	14	7	0.06
19	14	17	0.04
20	14	103	0.22
21	15	17	0.03

segment-ID	start-node-ID	end-node-ID	distance
22	15	16	0.04
23	15	18	0.07
22	16	15	0.04
24	16	17	0.04
25	16	105	0.05
7	17	5	1.02
19	17	14	0.04
21	17	15	0.03
24	17	16	0.04
14	18	10	0.15

The pagodas's data are stored in table and the sample data are shown in table 3.5.

Table 3.5 The "poi_pagoda" table

pagoda_ID	pagoda_name	latitude	longitude
1	A Ga Te Phaya Gyi	21.158845°	94.853813°
2	A Pal Yadanar Pagoda	21.149051°	94.857635°
3	A Twin Zigon Pagoda	21.172179°	94.857747°
4	Alodawpyae pagoda	21.176854°	94.882570°
5	Alodawpyi Kyaung	21.188932°	94.893863°
6	Ananda Oak Kyaung	21.171838°	94.867177°
7	Ananda Temple	21.170906°	94.867683°
8	Ashin Arahan Monastery	21.140267°	94.868218°
9	Aung Myay Thar Monastery	21.171217°	94.924756°
10	Aung Mye Bontha	21.196521°	94.905863°
11	Bagan Archeological Museum	21.167725°	94.855943°
12	Bagan Myo Temple	21.172090°	94.876275°
13	Bocho Mi Gubyauk	21.121860°	94.853446°
14	Bu Phaya	21.176305°	94.857860°
15	Bulethi Pagoda	21.173821°	94.881886°
16	Chan Tha Gyi	21.199297°	94.907597°
17	Chauk Phaya Hla Kyaung	21.207212°	94.925433°

pagoda_ID	pagoda_name	latitude	longitude
18	Chedaw Ya Phaya	21.196537°	94.905660°
19	Dama YaziKa	21.144995°	94.883421°
20	Dhamma Yan Gyi	21.162235°	94.872931°
21	East Hpet Leik Pagoda	21.129576°	94.853200°
22	East Panet The Gu	21.195843°	94.894173°
23	Gawdaw Palin Pagoda	21.170020°	94.856409°
24	Gu Byauk Gyi(2) (myinkabar)	21.185730°	94.893613°
25	Gu Byauk Nge(2)(Wetkyi-in)	21.184928°	94.890974°
26	Gu Pyauk Gyi(1)(myinkabar)	21.157274°	94.860723°
27	Gu Pyauk Nge(1)(Wetkyi-in)	21.158825°	94.861081°
28	Gu Taw Thit	21.164321°	94.861291°
29	Guni Hpaya	21.205747°	94.923058°
30	Hgnet Pyit Taung	21.191861°	94.912429°
31	Hpaya Ko Hsu	21.189916°	94.893135°
32	Hsin Byu Shin Pagoda	21.162642°	94.895360°
33	Hsu Taung Pyi Pagoda(1)	21.173467°	94.868100°
34	Hsu Taung Pyi pagoda(2)	21.204702°	94.910752°
35	Htilo Min Lo Pagoda	21.178604°	94.879232°
36	Htupalaysutan	21.135724°	94.856371°
37	Inn pagoda	21.165622°	94.872164°
38	Ka tha Pa Htupa Pagoda	21.160842°	94.905235°
39	Ka tha Pa(East) Pagoda	21.160296°	94.905914°
40	Ka Tha Pa(West) Pagoda	21.159868°	94.905523°
41	Kan U Phaya	21.193161°	94.892976°
42	Khaymingha	21.177309°	94.875665°
43	Kon Daw Gyi	21.206904°	94.927962°
44	Kyan Ma Ba	21.169241°	94.857025°
45	Kyan Sit Thar Umin	21.191512°	94.893105°
46	Kyauk--Sa-Ga-Gyi	21.128676°	94.853855°
47	Kyaunggyi Ama	21.121386°	94.862177°

pagoda_ID	pagoda_name	latitude	longitude
48	Kyaunggyi Nyima	21.121376°	94.861981°
49	Law ka Hteik Pan Pagoda	21.165215°	94.865433°
50	Lawka Nanda Pagoda	21.127037°	94.850353°
51	Lay Myat Hna(2) Pagoda	21.160248°	94.901746°
52	Lay Myet Nar(1) Pagoda	21.148279°	94.860366°
53	Loka Man Aung	21.191918°	94.911555°

The restaurants's data are stored in table and the sample data are shown in table 3.6.

Table 3.6 The "poi_restaurant" table

restaurnat_ID	restaurant_name	latitude	longitude
1	A Little Bit of Bagan Restaurant	21.191547°	94.900960°
2	A'ung Sar. Seafood Thailand	21.194211°	94.903578°
3	Angle Reataurant	21.191661°	94.899432°
4	Aroma(2)	21.191735°	94.900746°
5	Aung Yadana Restaurant	21.193584°	94.896671°
6	Aye Mya Thida	21.132913°	94.865773°
7	Aye Tha Har	21.191608°	94.901194°
8	Aye Yeik Thar Yar (Nyaung U)	21.192062°	94.895282°
9	Aye Yeik Thar Yar (New Bagan)	21.131682°	94.857561°
10	Ayeyarwady River Terrace Restaurant	21.188282°	94.883046°
11	Bagan ZAY	21.193362°	94.899354°
12	BBQ Garden Restaurant	21.200185°	94.919166°
13	Best Choice	21.206403°	94.911819°
14	Bibo Restaurant Nyaung U	21.192699°	94.898799°
15	Black Bamboo Nyaung U	21.193443°	94.900007°

restaurnat_ID	restaurant_name	latitude	longitude
16	Black Rose (1)	21.132862°	94.861090°
17	Black Rose (2)	21.130617°	94.860507°
18	Burger Hut (Myanmar)	21.131956°	94.861307°
19	Cafe Friends Nyaung U	21.201344°	94.908454°
20	CHEF HOUSE Restaurant	21.192183°	94.900043°
21	Cheri Land Nyaung U	21.198826°	94.904317°
22	CP Five Star	21.197729°	94.903306°
23	DATE Cafe & Bakery Nyaung U	21.193462°	94.899634°
24	Daw Hla Than Salad Shop	21.198875°	94.904666°
25	Delicious restaurant	21.131518°	94.860768°
26	Diamond Arrow Restaurant & Lacquerware Wotkshop	21.154618°	94.859609°
27	East Wak Kyi Inn Restaurant,	21.188698°	94.888978°
28	Eden BBB	21.189012°	94.891522°
29	Fatty Fried Chicken	21.195516°	94.907571°
30	Feel	21.170502°	94.925770°
31	Food Affair Restaurant	21.195688°	94.902495°
32	Full Moon Nyaung U	21.195732°	94.901174°
33	Golden Bagan Restaurant(New Bagan)	21.134699°	94.861881°
34	Golden Bamboo Restaurant	21.132336°	94.856806°
35	Golden Bangan Restaurant(Nyaung U)	21.193570°	94.899918°
36	Golden Brothers	21.130491°	94.859361°
37	Golden Emperor Restaurant	21.184931°	94.885881°
38	Golden Land Restaurant Nyaung U	21.193410°	94.899481°
39	Golden Myanmar 2 Restaurant	21.173029°	94.867536°

restaurnat_ID	restaurant_name	latitude	longitude
40	Green Elephant Restaurant	21.131939°	94.857511°
41	Golden Myanmar Restaurant(1)	21.174139°	94.865884°
42	Han Yar Pyae Cafe	21.197158°	94.902695°
43	Harmony BBQ	21.184775°	94.885773°
44	Harmony Restaurant	21.133646°	94.860318°
45	Hollywood Restaurant	21.194233°	94.899766°
46	HTI Bar & Restaurant	21.192867°	94.899017°
47	Indian Hut Restaurant (Bagan)	21.191658°	94.901067°
48	Kabyar Nya Restaurant	21.127245°	94.855484°
49	Kaday Kywe Restaurant	21.132843°	94.861305°
50	Kan Daw Gyi Nyaung U	21.193320°	94.898949°

The hotels's data are stored in table and the sample data are shown in table 3.7.

Table 3.7   The "poi_hotel" table

hotel_ID	hotel_name	latitude	longitude
1	Amata Garden Resort Bagan	21.129680°	94.858822°
2	Amazing Bagan Resort	21.179074°	94.905439°
3	Ananta Bagan	21.189550°	94.891468°
4	Areindmar Hotel	21.133189°	94.857481°
5	Arthawka Hotel	21.133085°	94.869578°
6	Aung Mingalar Hotel	21.191643°	94.895026°
7	Aung Su Pyae Hotel	21.195974°	94.900424°
8	Aureum Palace Hotel	21.173474°	94.901611°
9	Aye Yar River View Resort	21.177855°	94.861304°
10	Bagan Airport Hotel	21.172361°	94.923823°

hotel_ID	hotel_name	latitude	longitude
11	Bagan beauty Hotel	21.132328°	94.856530°
12	Bagan Central Hotel	21.132043°	94.858371°
13	Bagan Empress Hotel	21.130336°	94.860447°
14	Bagan Guest House	21.132486°	94.859558°
15	Bagan Hotel	21.169088°	94.855195°
16	Bagan Lodge	21.136044°	94.873959°
17	Bagan May Hotel	21.185092°	94.886055°
18	Bagan Nova Guest House	21.130776°	94.861343°
19	Bagan Princess Hotel	21.186113°	94.887868°
20	Bagan Regalia Hotel	21.192998°	94.900049°
21	Bagan Relax Guest House	21.193722°	94.904081°
22	Bagan Sense Hotel	21.176299°	94.916671°
23	Bagan Star Hotel	21.192478°	94.907657°
24	Bagan Thande Hotel	21.167482°	94.854230°
25	Bagan Thiripyitsaya Sanctuary Resort	21.164239°	94.853960°
26	Bagan Umbra Hotel	21.184918°	94.885489°
27	Bagan view Hotel	21.134821°	94.859088°
28	Bamboo House	21.173149°	94.902628°
29	Bawga Theiddhi Hotel	21.130270°	94.871049°
30	Blue Bird Hotel	21.131182°	94.870831°
31	Crown Prince Hotel	21.133794°	94.856494°
32	Diamond Lion Hotel	21.175531°	94.918149°
33	Duwun Motel	21.132566°	94.859856°
34	Eden Motel	21.198834°	94.908378°
35	Ever New Guest House	21.173947°	94.919177°
36	Floral Breeze Hotel	21.129373°	94.855329°
37	Gold Star Hotel	21.196929°	94.900012°
38	Golden Crown Motel	21.192486°	94.903105°
39	Golden House Hotel	21.195760°	94.907443°

hotel_ID	hotel_name	latitude	longitude
40	Golden Myanmar Guest House	21.200284°	94.907188°
41	Golden Rose Guest House	21.188516°	94.889235°
42	Grand Empire Hotel	21.200399°	94.906569°
43	Green Land Motel	21.175458°	94.917108°
44	Glorious Bagan Hotel	21.130709°	94.858797°
45	Heritage Bagan Hotel	21.171094°	94.922903°
46	Hotel Nan Eain Thu	21.132956°	94.870110°
47	Hotel Razagyo	21.133520°	94.869973°
48	Hotel Yadanarbon Bagan	21.134133°	94.870634°
49	Hotel Preinmar Bagan	21.192945°	94.891673°
50	Innwa Motel	21.200378°	94.907519°
51	Kaday Aung Hotel	21.137078°	94.860483°
52	Kaytumadi Dynasty Hotel	21.128784°	94.870890°
53	Kumudara Hotel Pagoda View	21.138412°	94.863555°

Table 3.8 shows some of the dataset for the “segment_pagoda” table in the database.

Table 3.8 The “segment_pagoda” table

segment_ID	pagoda_ID	start_node_ID	distance
1599	1	553	0.21
1599	1	598	0.3
77	2	43	0.07
77	2	44	0.06
122	3	68	0.05
122	3	69	0.04
581	4	334	0.08
581	4	626	0.23
1037	5	663	0.14

segment_ID	pagoda_ID	start_node_ID	distance
1037	5	664	0.15
892	6	552	0.03
892	6	558	0.03
1602	7	764	0.04
1602	7	777	0.05
1605	8	959	0.03
1605	8	960	0.11
500	9	290	0.08
500	9	298	0.22
1298	10	818	0.06
1298	10	819	0.11
112	11	63	0.02
112	11	65	0.03
584	12	336	0.11
584	12	337	0.1
7	13	5	0.3
7	13	17	0.72
129	14	70	0.17
129	14	535	0.28
1609	15	335	0.24
1609	15	962	0.13
1326	16	840	0.19
1326	16	841	0.07
1596	17	472	0.09
1596	17	488	0.31
1298	18	818	0.09

Table 3.9 shows some of the dataset for the "segment_restaurant" table in the database.

Table 3.9 The "segment_restaurant" table

segment_ID	restaurant_ID	start_node_ID	distance
1048	1	669	0.03
1048	1	674	0.04
1052	2	672	0.04
1052	2	673	0.17
1059	3	676	0.09
1059	3	677	0.13
1044	4	667	0.2
1044	4	674	0.01
1102	5	703	0.12
1102	5	704	0.02
328	6	194	0.05
328	6	202	0.12
1048	7	669	0.02
1048	7	674	0.05
1099	8	702	0.13
1099	8	703	0.09
185	9	113	0.03
185	9	130	0.13
1001	10	634	0.18
1001	10	655	0.05
1090	11	695	0.03
1090	11	697	0.03

Table 3.10 shows some of the dataset for the "segment_hotel" table in the database.

Table 3.10   The "segment_hotel" table

segment_ID	hotel_ID	start_node_ID	distance
207	1	126	0.04
207	1	127	0.02
828	2	505	0.26
828	2	506	0.06
1035	3	661	0.14
1035	3	662	0.08
191	4	116	0.03
191	4	133	0.13
370	5	215	0.13
370	5	229	0.08
1099	6	702	0.08
1099	6	703	0.14
1106	7	707	0.04
1106	7	708	0.1
1631	8	490	0.28
1631	8	974	0.2
869	9	537	0.17
869	9	539	0.21
499	10	290	0.03
499	10	291	0.19
48	11	26	0.09

segment_ID	hotel_ID	start_node_ID	distance
48	11	114	0.08
187	12	114	0.13
187	12	131	0.1
211	13	128	0.15
211	13	146	0.02
217	14	131	0.07
217	14	149	0.1

Table 3.11 shows some of the dataset for the "PagodaToPagoda" table in the database.

Table 3.11 The "PagodaToPagoda" table

segment_ID	pagoda_ID1	pagoda_ID2	distance
129	89	14	0.03
129	14	89	0.03
1298	10	18	0.03
1298	18	10	0.03
122	3	67	0.03
122	67	3	0.03
1596	17	116	0.13
1596	17	146	0.16
1596	17	43	0.23
1596	116	17	0.13
1596	116	146	0.03
1596	116	43	0.1
1596	146	17	0.16
1596	146	116	0.03
1596	146	43	0.07

segment_ID	pagoda_ID1	pagoda_ID2	distance
1596	43	17	0.23
1596	43	116	0.1
1596	43	146	0.07
1616	22	93	0.01
1616	93	22	0.01
92	26	66	0.06
92	66	26	0.06
1544	27	107	0.05
1544	107	27	0.05
1036	31	85	0.03
1036	85	31	0.03
1652	38	39	0.09
1652	38	40	0.18
1652	39	38	0.09

Table 3.12 shows the dataset for the "PagodaToRestaurant" table in the database.

Table 3.12 The "PagodaToRestaurant" table

segment_ID	pagoda_ID	restaurant_ID	distance
500	9	30	0.04
129	89	72	0.01
129	14	72	0.04
129	89	57	0.02
129	14	57	0.01
1503	127	85	0.09

Table 3.13 shows some of the dataset for the "HotelToRestaurant" table in the database.

Table 3.13 The "HotelToRestaurant" table

segment_ID	hotel_ID	restaurant_ID	distance
1044	120	4	0.16
1102	86	5	0.03
1102	86	169	0.01
1102	86	70	0.07
1099	6	8	0.05
1099	6	128	0.07
1099	6	53	0.12
1114	60	21	0.05
1114	60	24	0.07
1114	60	82	0.01
1114	60	105	0.04
1114	60	144	0.11
1114	64	21	0.03
1114	64	24	0.01
1114	64	82	0.07
1114	64	105	0.12
1114	64	144	0.03
1114	83	21	0.08
1114	83	24	0.06
1114	83	82	0.12
1114	83	105	0.17
1114	83	144	0.02
1114	92	21	0.07
1114	92	24	0.05
1114	92	82	0.11
1114	92	105	0.16
1114	92	144	0.1

3.7.2 The Database Design

The database design contains the twelve tables named ‘start_junction_node’, ‘end_junction_node’, ‘road_segment’, ‘poi_pagoda’, ‘poi_restaurant’, ‘poi_hotel’, ‘segment_ pagoda’, ‘segment_restaurant’, ‘segment_hotel’, ‘PagodaToPagoda’, ‘PagodaToRestaurant’ and ‘HotelToRestaurant’. The database design of the system is shown in Figure 3.8.

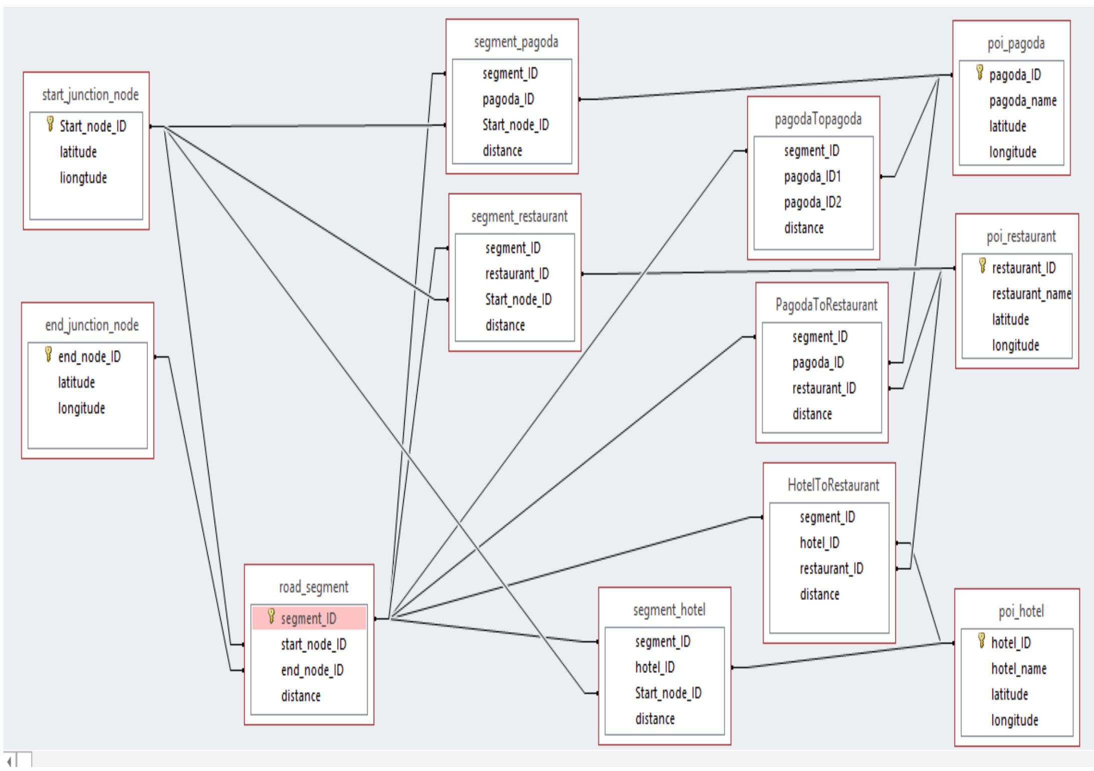

Figure 3.8 Database Design of the proposed System

3.8 Related Works

In this section, the previous works related to the trip planning queries and the sequenced routes are reviewed.

Chen et al. [1] have proposed a generalization of the trip planning query, called multi-rule partial sequenced route (MRPSR) query. In order to plan a route which can fulfill all the user defined partial sequence rules, Activity-on-Vertex (AOV) network is used. If the set of rules in AOV network is not a directed acyclic graph, cycles are reported in the set of travelling rules. If AOV network is a directed acyclic graph, the multi-rule partial sequenced route is searched by using three approximate algorithms : the Nearest Neighbor-based Partial Sequenced Route (NNPSR) algorithm, the Nearest Neighbor-based Partial Sequenced Route with Light Optimal Route Discoverer(NNPSR-LORD) algorithm, and the Advanced A* Search-based Partial Sequenced Route (AASPSR(k)) algorithm. All algorithm can only picks a single POI in each category, and can't search the multiple POIs in each category.

In [2], Sharifzadeh et al. have presented "The Optimal Sequenced Route (OSR) Query". The light optimal route discover (LORD) is used to find the optimal sequenced route and is light to use memory .The constant threshold value (T_c) and the variable threshold value (T_v) are used to limit the search space. Therefore, LORD iteratively builds and maintains the partial sequenced route in the reverse sequence. Finally, it returns a route with the minimum length passing through a set of POIs in a particular order from the source location of a user, where both order and type of POIs are specified by the user. Therefore, it is not suitable for the real-world application because it uses Euclidean distance.

Li F, Cheng D et al. [3] have described "On Trip Planning Queries in Spatial Databases". Nearest Neighbor (NN) algorithm is used to plan a trip in both Euclidean distance and road Network. The goal of NN algorithm is to find the best trip (route). The problem of algorithm expands the point that is closest to the last point in the partial trip without considering the end destination. The algorithm cannot plan a trip with the sequenced order of the user-defined categories.

CHAPTER 4

IMPLEMENTATION OF THE PROPOSED SYSTEM

Android is a mobile operating system developed by Google, based on a modified version of the Linux kernel and other open source software and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android provides the support of mobile map and location service, and which is probably a concern of vast numbers of developers. So far, the development of mobile map and location application is complex and difficult. The Android platform allows developers to write managed code using Java to manage and control the Android device. The Android operating system is open source and its source code was released under the Apache license. More and more smart phones today are using Android systems that are undoubtedly benefit millions of mobile users. Moreover, Android is free and open, providing an easy-to-use development kit containing flexible map display, control function and location support. So, Android platform is used to develop this trip planning system.

This proposed trip planning system has been implemented and evaluated using Android Studio 3.3.2.

4.1 Implementation of the System

In order to use the proposed system, users need to have android phone with the operation system version 4.0 at least and the trip planning application is installed in android phone. After installing the proposed system application in user's android mobile phone, the application icon is appear on the phone screen. User clicks the proposed application icon, and home page of the system is appear. To search the partial sequenced route that starts from the user location, the android phone's GPS location is needed to turn on. If phone's GPS location is off in the start of system, the message is appear to open GPS location to know the user's current place. Figure 4.1 shows the message to turn on device location.

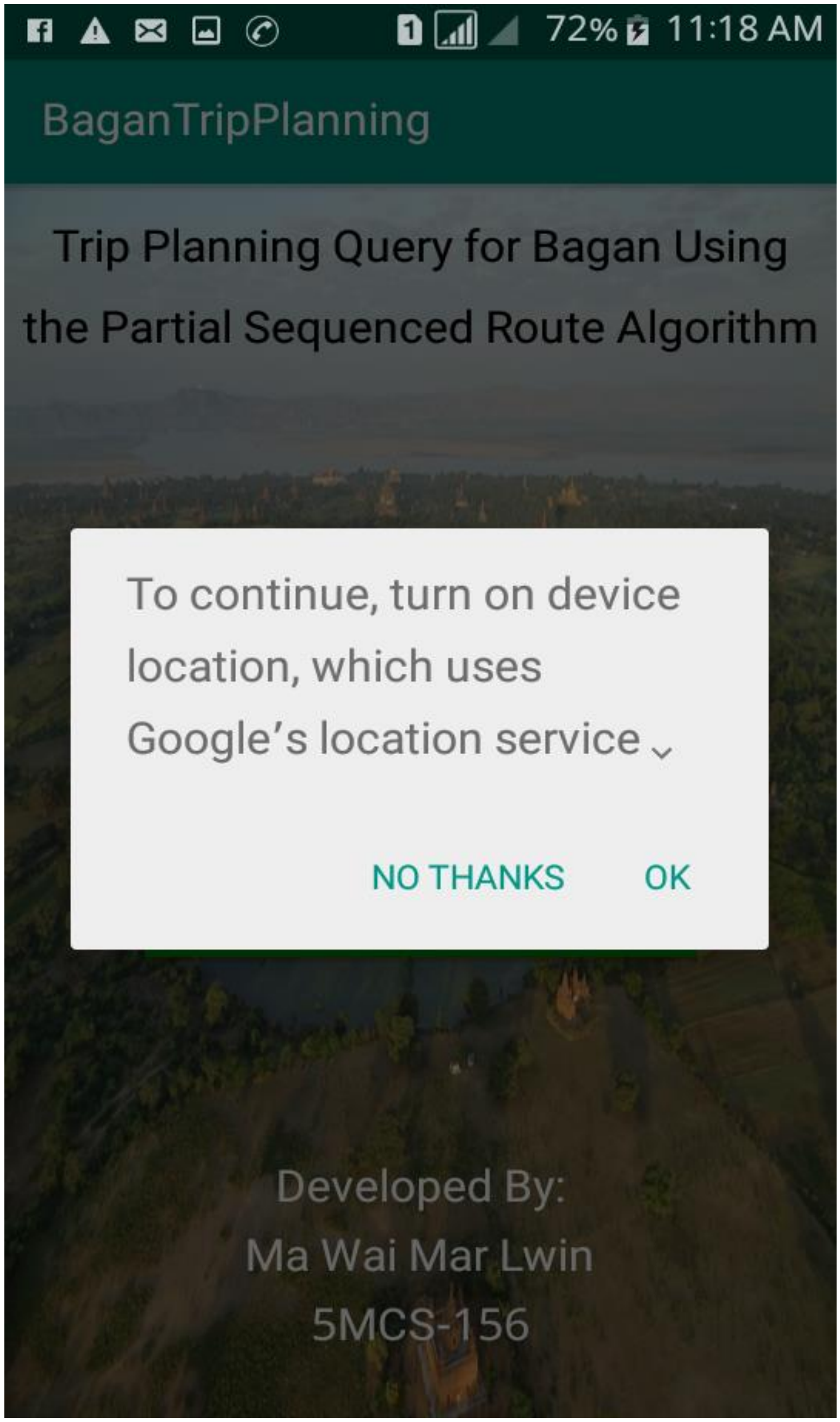

Figure 4.1 Message to turn on device location

After turning on the device location, welcome page or home page of the system is appear as shown in Figure 4.2.

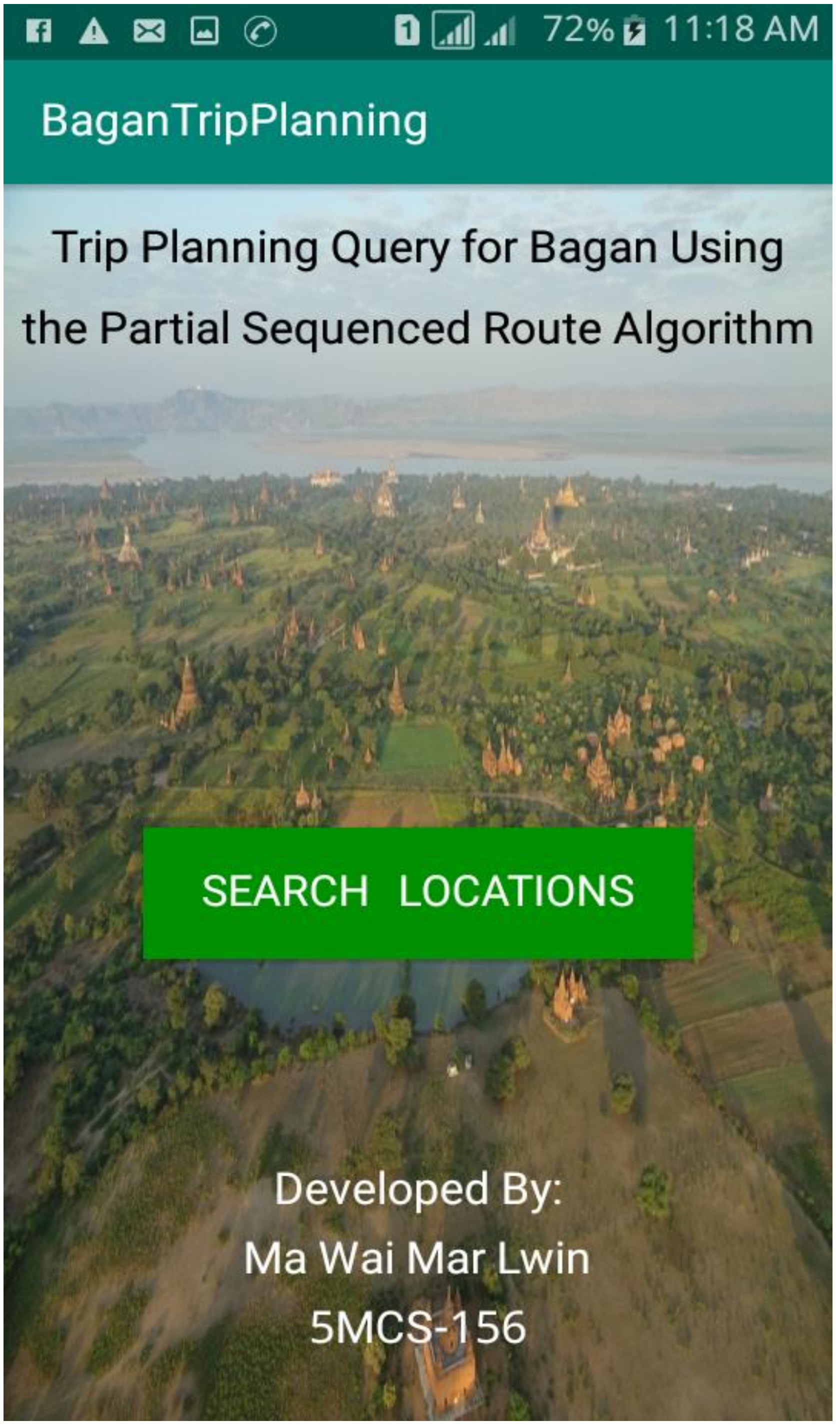

Figure 4.2 Home Page of the Bagan Trip Planning System

When the user clicks the "**SEARCH LOCATIONS**" button, the page is appear to choose the route or meeting place as shown in Figure 4.3.

4.1.1 Finding a Meeting Place for Two Users

There are two radio buttons to search route and meeting place. If the route is searched for trip, the system finds the sequence route based on the user-selected categories from user's current location by using NNPSR Algorithm. If the meeting place is searched, the system searches the meeting place for two users in different location.

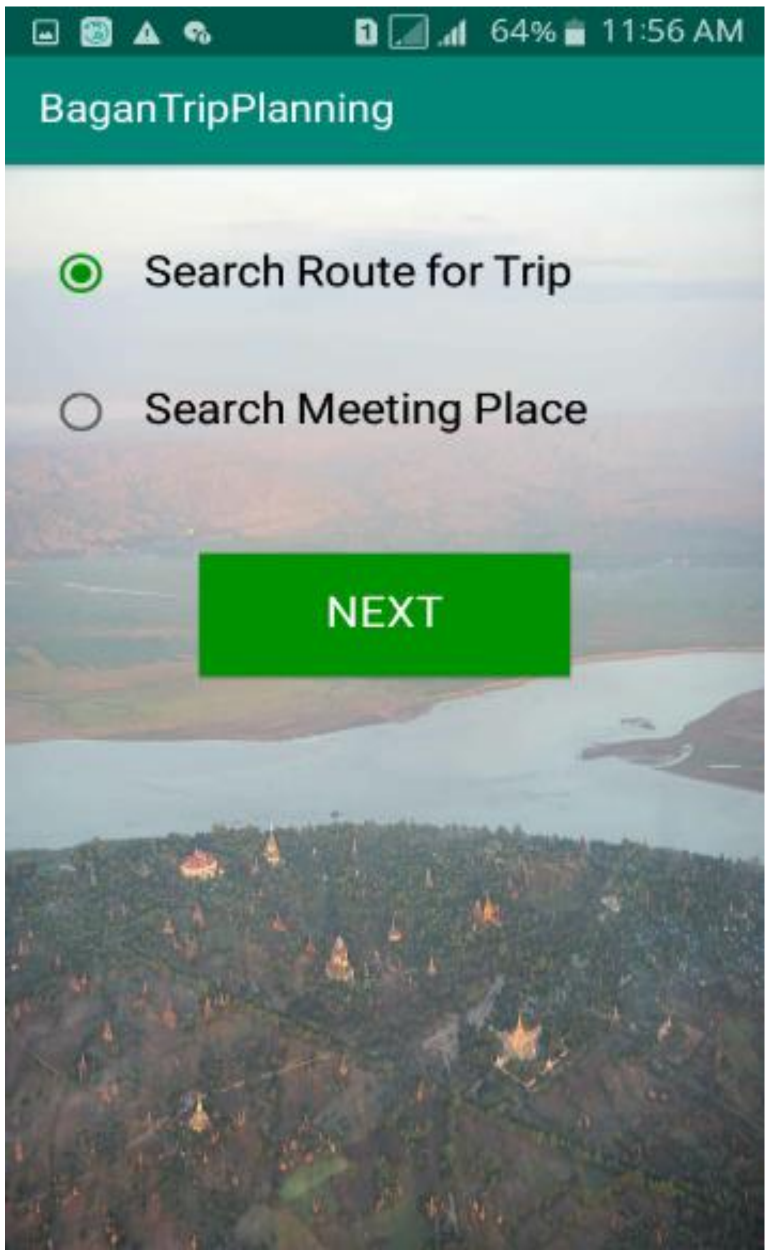

Figure 4.3 Route or Meeting Place Selection

If "**Search Meeting Place**" button is selected, the system shows the page to choose the friend's location, the friend's location type and the place to meet as shown in Figure 4.4.

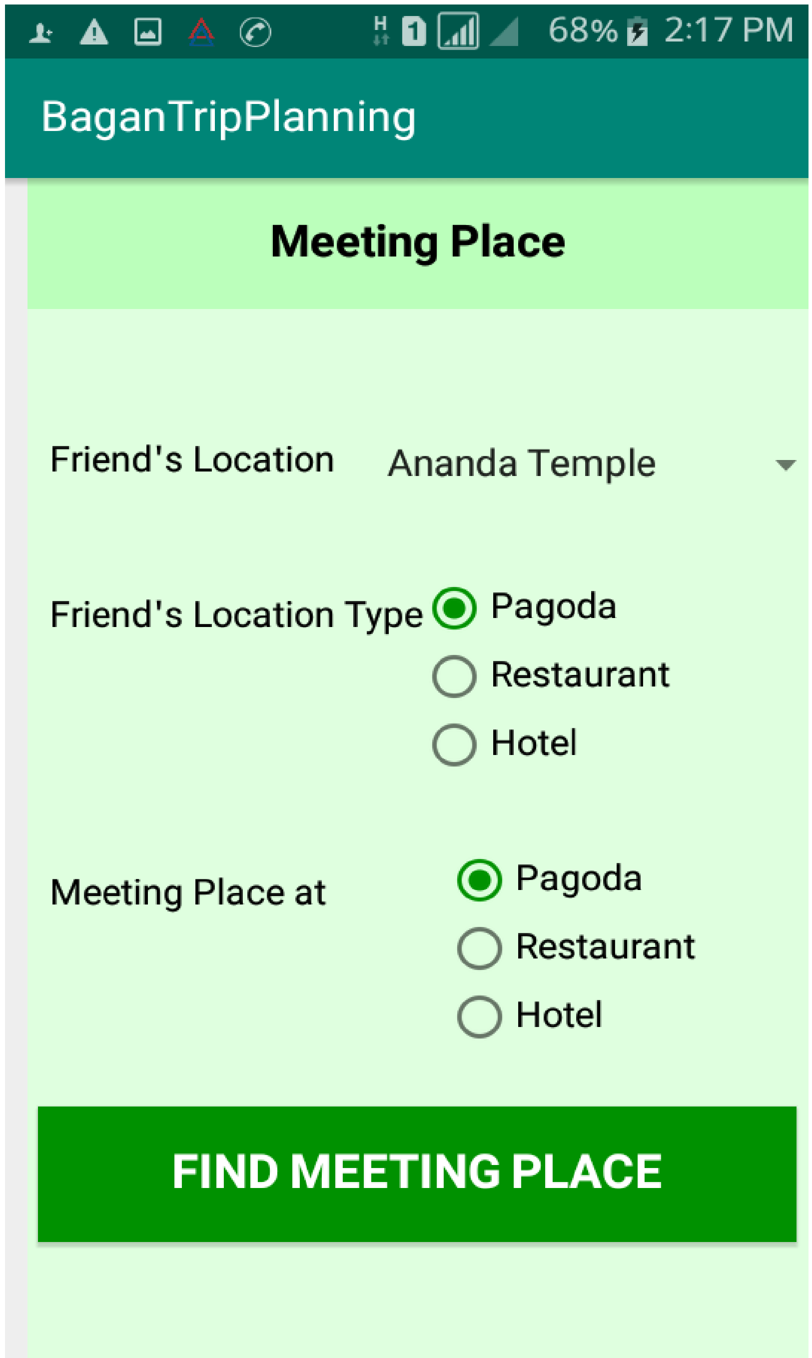

Figure 4.4 Selection of Meeting Place

After choosing meeting place at, the friend's location and type, the system shows three points: user location, friend location, and the meeting place are displayed on map as shown in Figure 4.5.

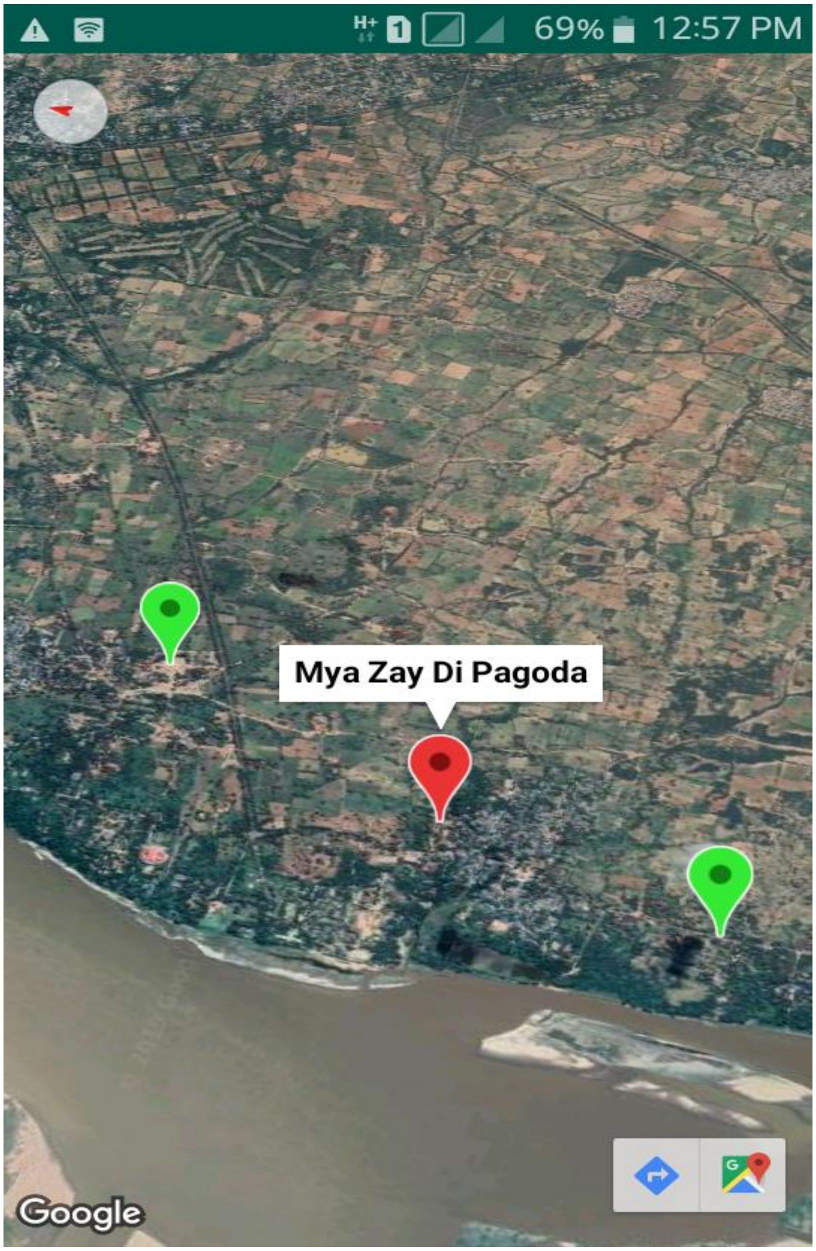

Figure 4.5 Meeting Place for Two Users

In figure 4.5, user's location is latitude 21.145138 and longitude 94.858051. Friend's location type is pagoda. So, Friend's location is Ananda Temple with latitude 21.170906 and longitude 94.867683. Users want to meet at a pagoda.

MidPoint for Two Users is

Latitude (21.145138+21.170906) / 2 = 21.158022

Longitude (94.858051+94.867683) /2 = 94.862867

A pagoda (latitude equal to 21.158) is searched in the online server database. Mya Zay Di Pagoda with latitude 21.158022 is found. If latitude 21.158 is not found, 21.157 or 21.159 is alternately searched. And, user location, friend location and Mya Zay Di Pagoda (a meeting place) are displayed on map as shown in Figure 4.5.

4.1.2 Searching Route for the Trip

Most people would probably like to make the travel priorities for their trip. The travel priority gives many advantages. Therefore, the proposed system includes the priorities to visit the pagodas, the restaurants and the hotels in Bagan region.

There are two priorities to choose categories for a trip in this application. First priority includes three radio buttons of Pagoda, Restaurant, and Hotel. Second priority involves four radio buttons of Pagoda, Restaurant, Hotel and None. If user selects "Pagoda" radio button, the number of 1-5 pagodas can be selected in spinner box. If user selects "Restaurant" radio button and "Hotel" radio button, one restaurant and one hotel are searched for the trip. If the user selects "None" radio button in second priority, the route of first priority is only showed on map.

If "**Search Route for Trip**" button is selected, the page is showed to choose the user-interested categories as shown in Figure 4.6.

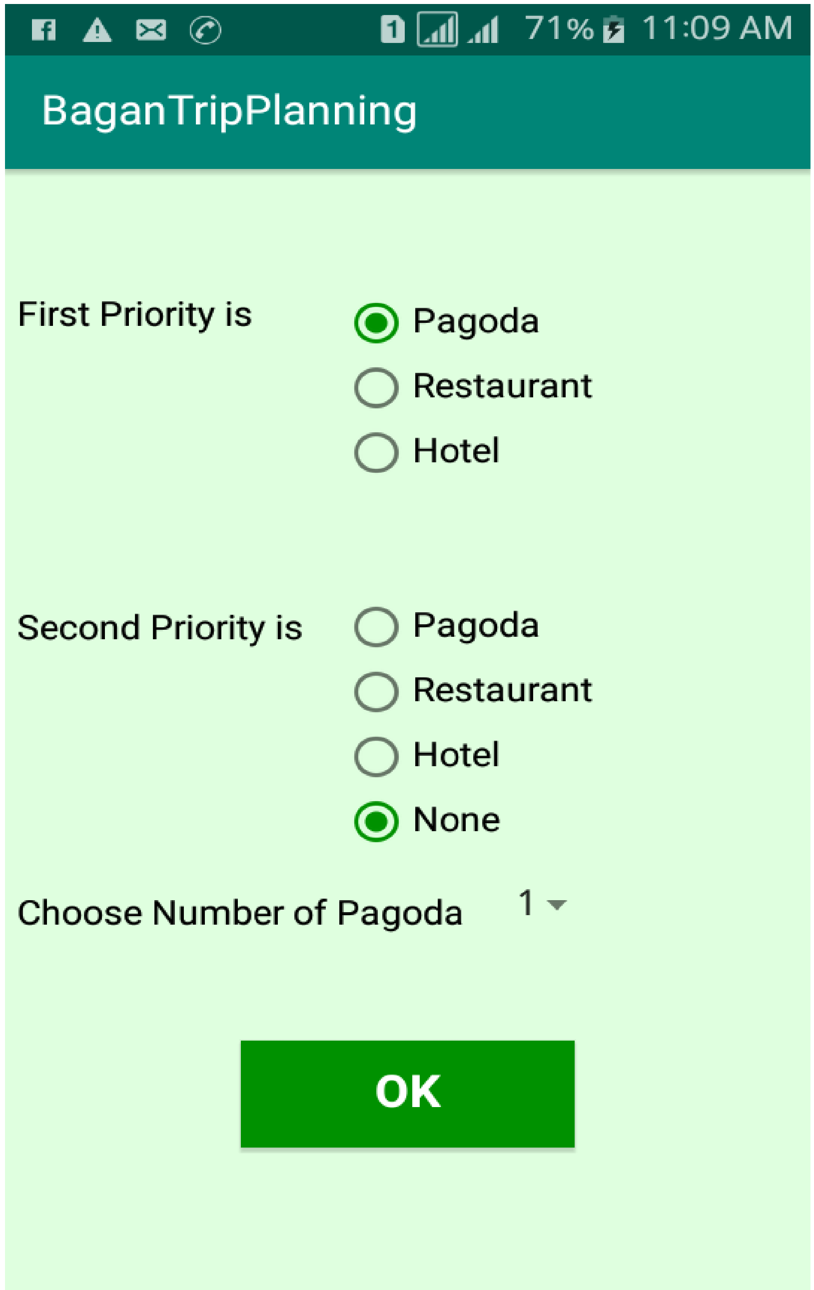

Figure 4.6 Selection the user-interested categories for a trip

In Figure 4.6, one pagoda is selected in first priority and the "None" radio button is selected in second priority. So, the system searches only one pagoda near to user location.

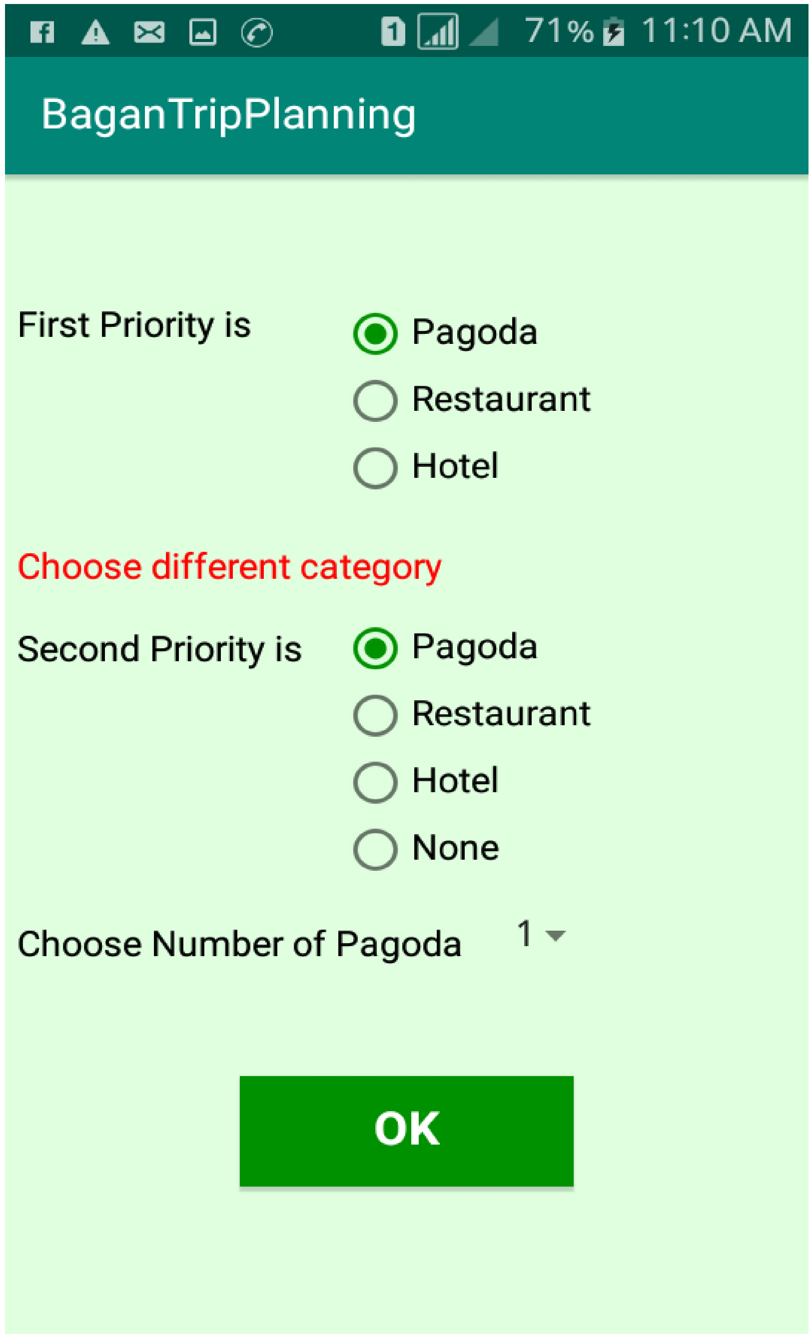

Figure 4.7 Notification Message to choose different categories

Users need to choose differently first priority with second priority. If first priority was equal to second priority, the notification message is appear to choose differently as shown in Figure 4.7.

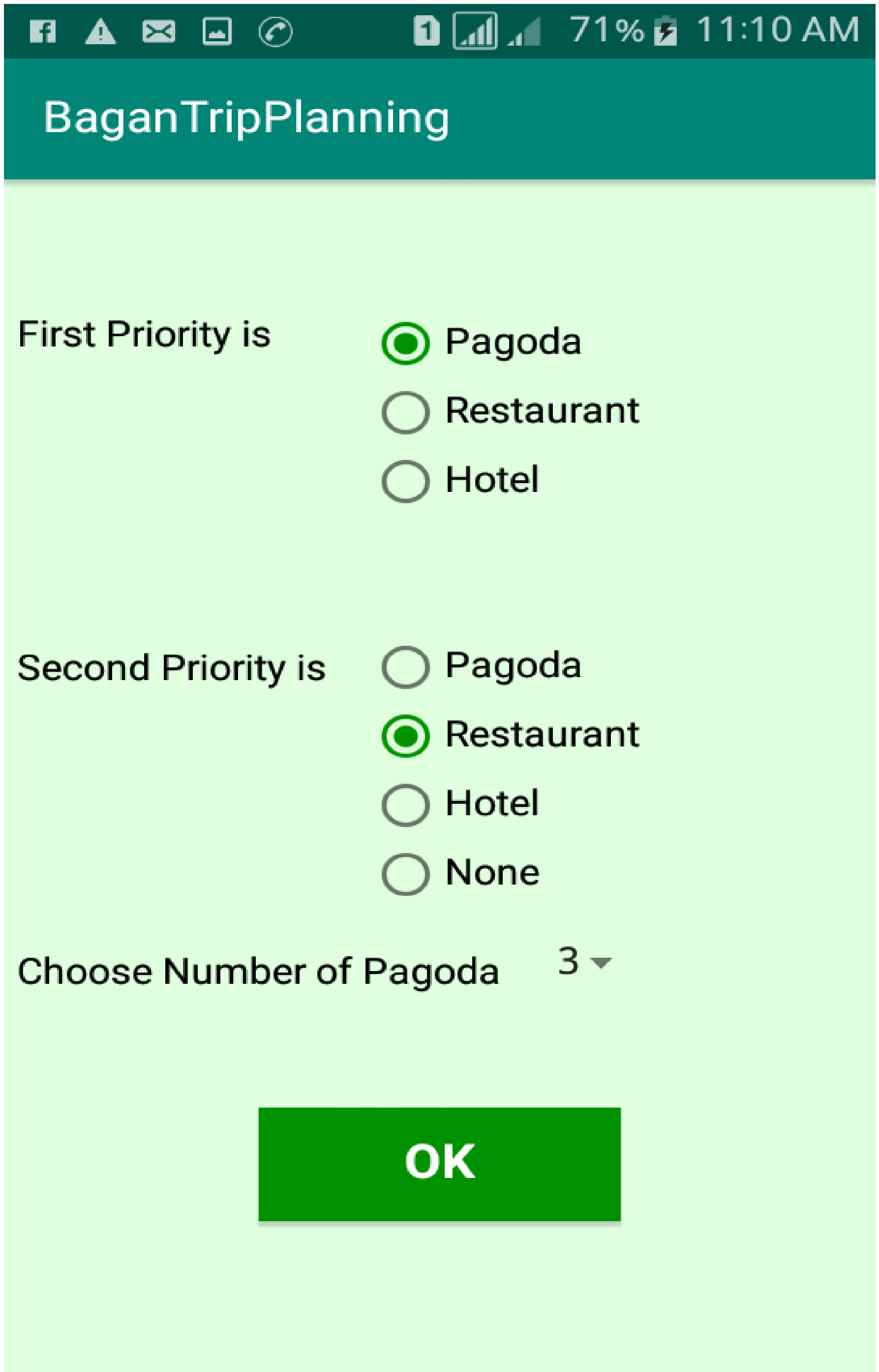

Figure 4.8 Selection of three pagoda and restaurant

In Figure 4.8, user's location is latitude 21.14827 and longitude 94.881825. And user selects "Pagoda" radio button with the number of 3 pagodas as first priority and "Restaurant" radio button as second priority.

After choosing first and second priority, NNPSR algorithm is used to search the route of 3 pagodas and one restaurant. And, the route that starts from user's current location is displayed on Google Earth. The names of POIs and priorities are also displayed in the route description as shown in Figure 4.9.

Fig 4.9 Route with the minimum road network distance for the trip

CHAPTER 5

CONCLUSION

The convergence of multiple technologies, including the Internet, wireless communications, geographic information systems, location technologies, and mobile devices, has given rise to new types of information utilities that may be referred as mobile location-based services (LBS). LBS can be described as applications that exploit knowledge about where an information device (user) is located. These include utility location information and route-guidance information. The technologies and applications of LBS play an ever increasingly important role in the modern, mobile, always-connected society.

Location-based system and map services can provide the tourist guide application for the users. The route-guidance mobile application for the location-based trip planning system is developed in the proposed system. This system can find a meeting place based on the user's current location and friend location, and can search a route for the trip which starts from the source location to the nearest famous places based on the tourists' personal interests. Moreover, users can also review the meeting place and the route for the trip not only in details (pagoda, restaurant and hotel name) but also on map. By using this proposed system, both tourists and local people can plan their trips systematically to visit famous places in Bagan.

5.1 System Limitation

As the system is chosen in the limited categories, it is not suitable for the users who want to visit other categories that don't contain in the proposed system. As the system is intended only for Bagan Region, it is not useful for other cities. If users want to search the route for their trip, they need to open GPS and the internet connection. If there is not open GPS, the system shows the message to open it. If users want to find a meeting place, the system needs to select the place to meet, the friend's location and the category of friend's location.

5.2 Further Extension

Beyond the work in this research, the proposed system can be extended to advanced trip planning software for the two or more cities. And, the research can be extended to add more categories such as handicrafts, lacquerware, souvenir shops, gems and jewellry shops, myanmar traditional umbrellas and textile ware shops, and ATM of banks and so on.

AUTHOR'S PUBLICATION

[1] Wai Mar Lwin, Myat Thiri Khine, Myint Myint Sein, "Trip Planning Query for Bagan Using the Partial Sequenced Route Algorithm", the National Journal of parallel and Soft Computing (NJPSC 2020), Yangon, Myanmar, 2020.

[2] Myint Myint Sein, Myat Thiri Khine, Wai Mar Lwin, "Trip Planning Query Based on Partial Sequenced Route Algorithm", IEEE 8th Global Conference on Consumer Electronics (GCCE), OSAKA Japan, October 15-18, 2019.

REFERENCES

[1] Haiquan Chen, Wei-Shinn Ku, Min-Te Sun, Roger Zimmermann, "The Partial Sequenced Route Query with Traveling Rules in Road Networks", Geoinformatica (an international journal for Geographic Information System(GIS)), July 2011.

[2] Haiquan Chen, Wei-Shinn Ku, Min-Te Sun, Roger Zimmermann, "The Multi-rule Partial Sequenced Route Query", the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, California, November 2008.

[3] Mehdi Sharifzadeh, Mohammad Kolahdouzan, Cyrus Shahabi, "The Optimal Sequenced Route Query", International Journal on Very Large Data Bases, July 2008.

[4] Dimitris Papadias, Jun Zhang, Nikos Mamoulis, Yufei Tao, "Query Processing in Spatial Network Databases", the 29th International Conference on Very Large Data Bases, September 2003.

[5] Feifei Li, Dihan Cheng, Marios Hadjieleftheriou, George Kollios, Shang-Hua Teng, " On Trip Planning Queries in Spatial Databases", the 9th international conference on Advances in Spatial and Temporal Databases, August 2005.

[6] Prof.Titiksha Bhagat, Devsmita Shil, Shital Pofare, "Android City Tour Guide System Based on Web Services", International Journal of Science, Engineering and Technology Research (IJSETR), March 2016.

[7] Man Lung Yiu, Nikos Mamoulis, Dimitris Papadias, "Aggregate Nearest Neighbor Queries in Road Networks", IEEE transactions on knowledge and data engineering, June 2005.

[8] Francesco Ricci, "Mobile recommender systems", International Journal of Information Technology and Tourism, 2010.

[9] Michael Kenteris, Damianos Gavalas, Daphne Economou, "An innovative mobile electronic tourist guide application", Journal of Personal and Ubiquitous Computing, February 2009.

[10] Höpken, Wolfram, Fuchs, Matthias, Zanker, Markus, Beer, Thomas, "Context-based adaptation of mobile applications in tourism", International Journal of Information Technology & Tourism, April 1, 2010.