*Second Asia Pacific Conference on Contemporary Research (APCCR, Malaysia, 2016)*
*ISBN:978 0 994365 606*
www.apiar.org.au

# IGUIDE - INTELLIGENT TOUR GUIDING SYSTEM

T.G.I. Bingun [a], C.M. Palliyaguru [b], V.M.P. Godakandage [c], K.K.A.W. Madhubhashana [d], S.A.U.S. Samaratunge [e], T.D. Perera [f]
[abcdef] Sri Lanka Institute of Information Technology, Malabe, Sri Lanka
*Corresponding email*: wimamadubashana@gmail.com

## Abstract

Tourism is an industry based on a wide range of technological advancements where people from different places and cultures are interacting with increasing interest. Unlike predecessors, nowadays, people can affordably and efficiently travel across the world comparatively safe. Since tourism is one of the prevalent and emerging industry, its benefits and challenges are keenly observed by governments. Tourism affects the economic, socio-cultural, ecological and informative resources of nations. The tourism industry in Sri Lanka doesn't comprise a proper intelligent travel and tourism customer support system. The main goal of our research project is to provide the web and mobile application, which will guide the tourist by providing proper information and system is available for Q&A purpose. The system is acting as a human counterpart, by assisting tourists to visit their preferred places under the guidance of our system by communicating with the system. For this, users' texts are classified by using a supervised machine learning algorithm named Naïve Bayes, Semantic Web, optimal route generation, by facilitating a better experience to the tourist according to his/her choices using Pathfinding algorithms, Ontology and Reasoning in Modern Applications.

***Keywords:*** Domain Knowledge, Pathfinding, Semantic Web, Reasoning.

## 1. Introduction

According to the World Tourism Organization, Yearbook of Tourism Statistics, Compendium of Tourism Statistics and data files, the number of tourists who visit Sri Lanka increases (The World Bank, n.d.) every year. 1,798,380 tourists visited Sri Lanka between 1st of January 2015 to 31st of December 2015 (Sri Lanka Tourist Development Authority (SLTDA), n.d.).

Table 1 : Number of tourists visited Sri Lanka

| Year | Number of Tourists per Year |
|---|---|
| 2011 | 856,000 |
| 2012 | 1,006,000 |
| 2013 | 1,275,000 |

Tourists who arrive in an unknown country may face several problems throughout their journey. At the time they arrive at the airport, they would be stressed out thinking about the journey ahead. Including, identifying the places to visit within the country, traveling to places of their interest, the difficulty of getting location-based information and fulfilling their accommodation requirements. In the modern world, using tour guiding systems have become a lot more popular among people. Traditionally, people used maps and compasses to direct

themselves to specific places in an unknown environment. But with the improvements in science and technology, people tend to use guiding systems integrated into their mobile devices. There are millions of tourists who visit a country to see it as they please. But, there is only a handful of tour guiding systems available and most of them are unable to meet the user requirements. Therefore, the need for such systems has become very high.

IGuide system is developed to help tourists who visit Sri Lanka to spend their leisure time and have a safe journey around Sri Lanka. Tourists are not the only group that is benefitted by this system. IGuide aims to overcome and address various navigation issues faced by the people in Sri Lanka. To address these problems, the research team have implemented a complete guiding system, where the user will be able to visit his preferred places under the guidance of the system.Through our system, tourists can effectively achieve all these tasks. It will guide the tourist to the desired destination conveniently. It will provide the tourist with location based information which will let the user know where they should visit next and track the tourist (tourist will be tracked anonymously to protect their privacy). This helps to increase the security of the tourist in case he gets lost.

## 2. Methodology

As a pre-preparation for the research, a comprehensive study was carried out to identify major problems faced by tourists who visit an unknown country. There are no any proper existing applications to satisfy the tourist's requirements. They express a moderate level of dissatisfaction towards infrastructure, transport facility, arrangements, unexpected expenditure, lack of information availability and service related factors. I-Guide is the proposed solution for these drawbacks.

A traditional feasibility study was followed in order to check whether the IGuide is financial, technically and operationally feasible. It is proved due to the open source technologies and does not have any technological constraints and dependencies. Then, the motivation was directed to the functional and non-functional approach of IGuide. In the design phase, those requirements of I-Guide were mapped to a high-level architecture. In the implementation phase, I-Guide is developed based on 7 major components.

1. Question-Based Feedback Engine (Natural Language Understanding)
2. Location-Based Information
3. Optimal Route Generation Based on the Tourist's Visiting Places (Pathfinding)
4. Live and Smart Notifications
5. Location-Based Review System
6. Universal Ontology Structure
7. Predict tourism categories using a survey according to user preferences

The high-level architecture of the I-Guide system is shown in below Figure 1.

*ISBN:978 0 994365 606*
www.apiar.org.au

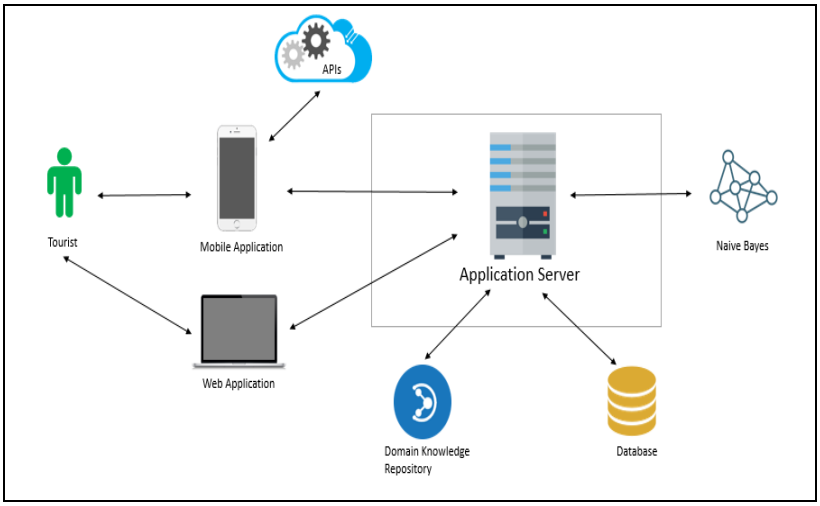


Figure 1 : High level Diagram

### 2.1 Question-Based Feedback Engine (Natural Language Understanding)

To interact with guiding systems by asking questions and getting relevant responses, Natural Language Understanding techniques were followed. Naive-Bayes-Classifier approach and two algorithms which were used for this purpose, which is more appropriate in fulfilling this requirement.

Question-based feedback engine starts with providing of questions by the user. The question will go through a set of analyzing techniques which are implemented to identify the human language. The first step is removing unwanted characters and converting the question to lowercase letters. Then, it will be sent to the Naive-Bayes-Classifier, to classify the question according to the category.

**Text Classification**

**Naïve-Bayes Classifier** is a Machine Learning Algorithm that needs to be trained for supervised learning tasks like classification, prediction etc... Bayes theorem is used to calculating the posterior probability, P(c|x), from P(c), P(x), and P(x|c). Naive Bayes classifier assumes that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors. This assumption is called class conditional independence (Naive Bayesian, n.d.).

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

$$P(C|X) = P(X_1|C) * P(X_2|C) * \ldots * P(X_n|C) * P(C)$$

1. P(c|x) is the posterior probability of class (target) given predictor (attribute).
2. P(c) is the prior probability of class.
3. P(x|c) is the likelihood which is the probability of predictor given class.
4. P(x) is the prior probability of predictor.

**Classification** is the mechanism used for identifying the correct class for a given input. In basic classification tasks, a set of classes are defined according to the task domain and each input is considered in isolation from all other inputs.

**Supervised classification** is built based on a training dataset with the correct class for each input. The framework used by supervised classification is described in the below diagram (Figure 2) with two processes (Learning to Classify Text, n.d.).

a) In the training process, a feature extractor is used to convert each input value to a feature set. These feature sets capture the basic information about each input which is used to classify it. Feature sets and classes are sent to the machine learning algorithm to generate a model.

b) In prediction process, the same feature extractor is used to convert inputs to feature sets. These feature sets are then sent to the classifier model and it generates the predicted class for that input.

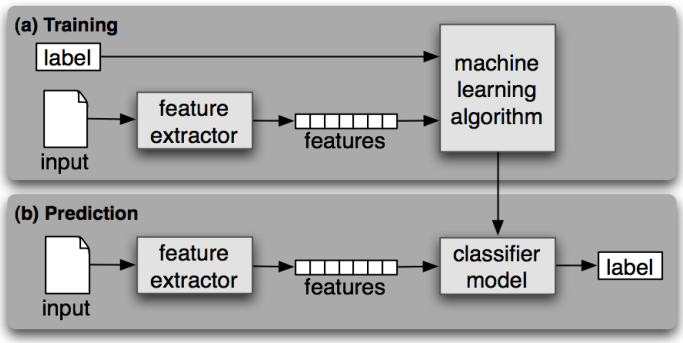


Figure 2 : Dataset Training Process [4]

In Order to identify the category of the question, the classification model has to be trained based on the above-mentioned classification technique by using the training data set.

**Training Classifier Model in Naïve Bayes Algorithm**

In order to train the classifier model, it was provided with predefined questions based on the most common categories related to tourism i.e. Hotels, Restaurant, Worship places, etc.... at runtime and identify the category of any unknown question.

Question analyzing consists of two main steps.

1. Question Analyzer Algorithm
2. Question Word Corrector Algorithm

**1. Question Analyzer Algorithm**

In this process, the question is analyzed by using the question analyzer, which is our own algorithm to remove unwanted words (in, a, this, by, an, more, the), special characters (@, #, /), split the words with spaces to finally identify the variations of words which describe any city, place name, district which is included in ontology. When considering the variations, algorithm is implemented in a way of producing the output in three levels, such as one word, word pair and triple words which are mentioned in below example.

```
Question: Give me a description about Kandy Hill Resort

[kandy, hill, resort, kandy hill, kandy hill resort, hill resort]
```

Figure 3 : Word Variations

After identifying the relevant question category and possible variations for the filtered text, next step is querying it from the ontology by using SPARQL queries. Even though these possible variations were identified in the given text, due to spelling mistakes and contrast in pronunciation (tourists' pronunciation in English differs and they may not aware of the exact names of the visiting places in Sri Lanka), it cannot identify the specific data from the ontology. In order to overcome this problem, an algorithm is implemented to automatically predict the relevant place according to the tourist's filtered text.

## 2. Question Word Corrector Algorithm

The main goal of this algorithm is to identify misspelled words and suggest the most matching terms (visiting place names) which are from the ontology. First, it will extract all the classes / individuals from the ontology by using SPARQL queries to compare it with the filtered variations word list. The technique which is used to compare and find the accurate text based on the variation word list is described in two main parts.

1. Take the filtered variations word list and get one word at a time by iterating through the whole array list and make the letter combinations of that word. It is shown in below Figure 4.

```
d
da              a
dal             al              l
dala            ala             la              a
dalad           alad            lad             ad
dalad           alad            lad             ad
dalad m         alad m          lad m           ad m
dalad ma        alad ma         lad ma          ad ma
dalad mal       alad mal        lad mal         ad mal              l
dalad mali      alad mali       lad mali        ad mali             li
dalad malig     alad malig      lad malig       ad malig            lig
dalad maliga    alad maliga     lad maliga      ad maliga     ...   liga
dalad maligaw   alad maligaw    lad maligaw     ad maligaw          ligaw
dalad maligawa  alad maligawa   lad maligawa    ad maligawa   ...   ligawa
```

Figure 4 : Letter Combinations of a Word

2. Iterate through all these letter combinations with ontology words and compare it to find the most matching word which is in the ontology, by getting the matching letter count and the probability.

Place name or class with the highest probability and the highest matching letter count is the most accurate result. After identifying all these correct keywords, final step is to get correct specific data from the ontology. In order to do that, SPARQL queries were generated by including those relevant keywords and obtained the result.

### 2.2 Location-Based Information

This is another important requirement which is achieved by the research team. This process depends on the user's current location. Therefore, to get this feature, GPS and internet connection were used. By using GPS technology, it will identify the latitude and longitude to specify its current location and create a circular area around that location (with a radius of 1000m). By using the 'Google places API', system will identify all the nearby places in that circular area, which is within a range of 1000m.

Since this is a dynamic process, system should be able with identifying the user's location changes. Scheduled timer is used for this purpose. Timer is used to continuously track the users' current location based on GPS and by calling the API method, it obtains places which are around that area.

### 2.3 Optimal Route Generation for Visiting Places (Pathfinding)

Generally, pathfinding consists of two main steps.

1. Find Alternative Paths between Two Nodes
2. Shortest path generation

#### 1. Find Alternative Paths between Two Nodes

The first step is to identify possible routes / paths between tour places. In order to achieve this requirement, the 'Google Directions API' was used. Google Directions API is a service which is provided by google to obtain routes between two locations. Therefore, this API has been used to take possible routes between user's visiting places. Initially, it takes user's current location as the first node and from that, has to find all the possible routes to each and every node (locations).

#### 2. Shortest Path Generation

The second step in pathfinding process is the generation of the shortest path from the tourist's current location to the destination, by covering all the tour places. The developed algorithm starts with a beginning point and a set of possible remaining nodes. At each step, the lowest distance from the starting point is examined. When the node with the lowest distance is identified, that node is marked and it is not reconsidered. All nodes that are adjacent to it are added to the remaining node set if they have not been already examined. Then again start from that node and this process is continuously done until the destination point is reached.

### 2.4 Live and Smart Notifications

The current location of the tourist is identified by the system through GPS technology. This uses the 'Haversine' formula to calculate the great-circle distance between two points – that is, the shortest distance over the earth's surface (Calculate distance, bearing and more between latitude/longitude points, n.d.).

*Haversine formula:*

$$a = \sin^2(\Delta\varphi/2) + \cos\varphi_1 \cdot \cos\varphi_2 \cdot \sin^2(\Delta\lambda/2)$$
$$c = 2 \cdot \operatorname{atan2}(\sqrt{a}, \sqrt{(1-a)})$$
$$d = R \cdot c$$

To calculate the distance, formula needs latitude and longitude of the source and the destination. Since source is the tourist's current location, it will be taken from GPS and destination is tourist's next visiting place, which is gained by the domain knowledge. In order to do this process continuously, a parallel thread was used on user's mobile application which runs every time when the user is using the mobile phone. A range of 3 kms was specified and when the tourist reaches that range, it automatically sends a notification to the tourist by providing relevant information about the next tour place and the distance to that place.

### 2.5 Location Based Review System

This section focuses on implementing a reviewing system where users can share their experience and give the feedback about the places that they visit. The specialty of this feature is that this reviewing system is only available when only user visits a particular place.

After tourist starts their journey system will continuously track and get his current location details time to time by sending these details to the distance handler algorithm. From those details and using an algorithm, system identifies whether the tourist reached to a place that included in his tour or not. If yes and cross the radius of 300m after visited the place (Haversine formula is used to identify the circular area), then by using a timer system will notify user to share the experience by using this reviewing system. Word filter algorithm is another algorithm used to identify admins defined banned words when users are adding their reviews. This reviewing system is based on voice and users can use his voice to share their experience as a review. Speech to Text is used to recognize the voice and converted into text. This is done through using several methods as following,

- Receive a request for voice recognition
- Check the availability of application for speech recognizing
- If speech recognizing is available, then call the intent for it and receive the results
- Noise cancellation

Android API provides the voice recognition plugin for recognizing the voice of a user and get the relevant results from that. To do that particular task, first algorithm converts analog voice input into a digital voice input. Then that converted signals are passed to acoustic and language models. From the language model, we can identify the language, pitch and the accent of that input. In the acoustic model it creates a statistical report of sound that came up from each word. Speech Recognition engine will recognize the speech and send to speech engine as it contains methods to identify that and also noise cancellation is done through this process.

This whole system is developed in order to give users to get a good idea about the places and make decision making faster when they are selecting places to their tours. By considering the ratings and modes our system will suggest places as Top rated and most awesome places and also when a user logged in to the web application system will take users country and suggest recommended places according to the country.

### 2.6 Universal Ontology Structure

An Ontology is a formal naming and definition of types, properties and relationships of the entities that really exist for a particular domain (Noy & McGuiness, n.d.). This defines a common vocabulary for all the researches who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the specific domain and relations among those. Domain Ontology (or domain-specific ontology) represents concepts which belong

to part of the world. Particular meanings of terms applied to that specific domain knowledge are provided by domain ontology.

We constructed an Ontology for several reasons

- We need a Hierarchy in order to create the class structure
- Ontologies provide a way for automated reasoning to occur in order to discover new relationships between entities and this cannot be done through a relational database
- Querying performance is high in ontology
- To enable the reuse of the ontology
- To enable place selection to use the knowledge base

This contains a collection of classes, sub classes, individuals, data properties, object properties which are interrelated and the properties are describing the features and attributes of the classes and restrictions. This constructed Ontology is used in Tour Customization, Question-based engine and Category Prediction components in the system.

Ex: Places -> Country -> Province -> District -> Category

As shown in above example by using the Top-down development process our approach is to construct a rich domain knowledge hierarchy that covers all the areas that come under tourism in Sri Lanka. When we take Sri Lanka as our top level domain we divided that into provinces level and district levels. Likewise by adding classes, sub classes, individuals and data properties (Wikipedia, n.d.) we built the ontology for 2 domains.

Reasoning model is created to map all the relationships in the domain knowledge as that is important when we consider querying process using SPARQL queries. We used standard reasoning method to reason the domain knowledge that we created. Under this method, Pallet reasoner is used to reason the domain knowledge in protégé level as it is the most suited one because this can directly connect to Jena to overcome the limitations so in a case of full OWL reasoning is possible in Jena. In java we used Jena Ontology API to reason the ontology model (Jena Ontology API, n.d.).

Finally, there is no existing reusable travel specific ontology can be found in any of the resources available and because of that we have to construct this universal hierarchy from ground level. This knowledge base structure can be used as a universal travel specific ontology domain knowledge structure in any application which is related to this tourism industry who need a class hierarchy like this. This contains a hierarchy which can be expanded according to the requirements of an application. Furthermore, currently this hierarchy is created for the domain of Sri Lanka but, in the future this can be categorized even from the country level and develop from there onwards.

### 2.7 Predict tourism categories using a survey according to user preferences

This will provide suitable tourism categories for a particular user according to user preferences by letting them minimize the effort when they are creating their tour package. In order to predict, we used a data mining model and optimize all the information to provide the predicted categories with details.

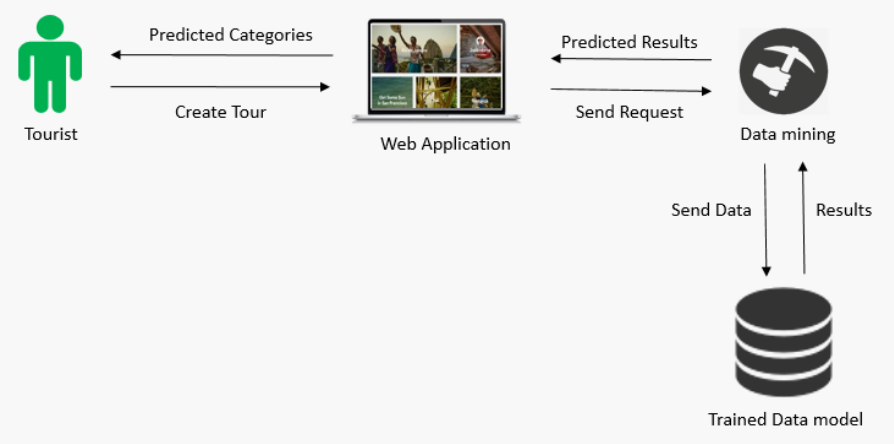


Figure 5 : Prediction system

When a logged in user wishes to create a tour that particular user will be directed to a survey with a set of questions to fill out. These questions are mainly related to likings and preferences. All the answers provided by a user will be directed to a data mining model to find predicted tourism categories.

**Classification Algorithm**

Classification is a data mining function that assigns items in a collection to target categories or classes. The goal of classification is to accurately predict the target class for each case in the data (Classification, n.d.).

**How Data mining used for Predictions**

A set of survey answers collected from different users are used as the data set in order to build the data mining model and classification is the data mining technique used to deliver the predictions to users.In order to do the prediction, Algorithm will be trained using a training data set containing a set of attributes and respective predicted column. From the training data set algorithm will discover relationships among attributes that would make it possible to make predictions. Then, the algorithm is given a test dataset which has the same attribute but without the prediction attribute and algorithm will analyze that input to give the best predictions.

As mentioned earlier by using the trainee and test data sets system will able to predict respective categories to users when creating their tour. Possible outcomes of this prediction will be Wildlife, Ayurveda, Historical, Landmarks etc.

**Dynamic Ontology updating through user interface**

Initially, at the development phase, we have created ontology using protégé. The system is equipped with functionality in admin end to develop the ontology.

In order to query, access or modify data in ontology we need to use query engine. In IGuide system, we are using apache Jena (McGuiness, 2016). System services are running as Restful java web services which will use to access ontology using Jena in iGuide mobile and web applications.

Jena provides functionalities in order to access different levels of the ontology hierarchy. For each request which needed for ontology access system will load reasoned ontology model. Using this ontology model system will query data using SPARQL queries (McGuiness, 2016). Using Flyweight design pattern, we have optimized ontology loading in coding level. Ontology model will be load once and system will share the ontology model in all requests. The system will reload the model only if any modification or changes done to the ontology. This will improve the efficiency of the system and reduce the response time.

After loading the reasoned ontology model we can query information by name of the element append with the Ontology URI. Unique Resource Identifier is using to uniquely identify an ontology. Jena has exposed methods to extract data properties and relations of ontology attributes. First, we have developed an algorithm to get the hierarchy of a given word. This algorithm returns whole class hierarchy till root class which given word belongs to.

When system receives the places/provinces/districts as input, system will remove unwanted elements such as words include # as a prefix.

## Conclusıon

In this paper it is been discussed the use of our system as a virtual guidance system, which will help travelers to customize their tours accordingly. We have tried to provide a system which will make it possible to remove the need for third parties for a traveler who is visiting a place which is unknown to them. With our system, the user will be able to get a good understanding of the places they are about to visit and through the use of the mobile application, they will be constantly updated about the status of their tour.

This research paper provided some techniques that can be used to detect the optimal route detection algorithm, make the system understand the human natural language by using Natural Language Understanding techniques, semantic web, Ontology based domain knowledge, notification system with parallel threading to notify users about some important factors and discuss some strengths of these approaches.

Current systems which are similar does not provide the level of customization provided by our system and most of them do not provide all the services provided by our mobile application. Through our product, it was needed to provide the best possible service to a traveler and IGuide wants to make sure they are able to have the best possible experience.

## Future works

As the next stage of this research project, the main focus is on enhancing the application for both IOS and windows phones, using cross-platform development. Furthermore, accuracy level and the performance level can be increased using newly coming technologies by adapting to those and system will be modified according to the requirements. Since the system comprises of a question-based feedback engine, the focus will be given to increase the domain that the users can ask questions, by enhancing the Naïve Bayes classifier model. Considering security levels of the application, converging on implementing user logins for each user level.

*ISBN:978 0 994365 606*
www.apiar.org.au

## References


i. 'Calculate distance, bearing and more between Latitude/Longitude points'. (n.d.) [Online] Available at: http://www.movable-type.co.uk/scripts/latlong.html

ii. 'Learning to Classify Text.' (n.d.) [Online] Available at: http://www.nltk.org/book/ch06.html

iii. 'Naive Bayesian' (n.d.) [Online] Available at: http://www.saedsayad.com/naive_bayesian.htm

iv. *Classification* (n.d.) [Online] Available at: https://docs.oracle.com/cd/B28359_01/datamine.111/b28129/classify.htm

v. *Dijkstra Algorithm* (n.d.) [Online] Available at: http://students.ceid.upatras.gr/~papagel/project/kef5_7_1.htm

vi. Foy, N.F. & McGuinness, D.L. (2016) *Ontology Development 101: A Guide to Creating Your First Ontology*. National Institute of General Medical Sciences, 2016. [Online] Available at: http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html

vii. Foy, N.F., McGuinness, D.L. (n.d.) *Ontology Development 101: A Guide to Creating Your First Ontology*. [Online] Available at:

viii. http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html

ix. Hutt, K. (2005) *A Comparison of RDF Query Languages*. [Online] Available at: http://www.ewp.rpi.edu/hartford/~rhb/cs_seminar_2005/SessionE1/hutt.pdf

x. *Introduction to A* Pathfinding* (n.d.) [Online] Available at: https://www.raywenderlich.com/4946/introduction-to-a-pathfinding

xi. *Processing OWL Ontologies using Java, Jena Ontology API* (n.d.) [Online] Available at: http://www.slideshare.net/rajighawi/java-and-owl

xii. Sri Lanka Tourism Development Authority (SLTDA) (n.d.) *Tourism Research and Statistics*. [Online] Available at: http://www.sltda.lk/statistics

xiii. The World Bank – IBRD/IDA (n.d.) *International tourism, number of arrivals*. [Online] Available at: http://data.worldbank.org/indicator/ST.INT.ARVL

xiv. Wikipedia (n.d.) 'Ontology Components.' *Wikipedia* [Online] Available at: https://en.wikipedia.org/wiki/Ontology_components

xv. Wikipedia (n.d.) 'Pathfinding.' [Online] Available at: http://en.wikipedia.org/wiki/Pathfinding