ATMIYA UNIVERSITY

RAJKOT

A

ReportOn

TOURIST GUIDE

Undersubjectof

PROJECT

B.TECH,Semester– VII

(ComputerEngineering)

Submitted By:

1.	ASHISH SAKARIYA	190002097
2.	JIGAR SATHWARA	190002102
3.	PRIYANK VORA	190002126

Prof. Rupal Shilu

(FacultyGuide)

Prof. Tosal M.Bhalodia

(Head of the Department)

AcademicYear

(2021-22)

<u>CANDIDATE'SDECLARATION</u>

We hereby declare that the work presented in this project entitled "**TOURIST GUIDE"** submitted towards completion of project in **7th Semester** of B.Tech. (Computer Engineering) is an authentic record of our original work carried out under the guidance of "**Prof. Rupal Shilu".**

We have not submitted the matter embodied in this project for the award of another degree.

Semester:
7thPlace:Rajkot

Signature:

ASHISH SAKARIYA (190002097)

JIGAR SATHWARA (190002102)

PRIYANK VORA (190002126)

ATMIYA

UNIVERSITYRAJKOT

CERTIFICATE

Date:

This is to certify that the "**TOURIST GUIDE**" has been carried out by **ASHISH SAKARIYA** under my guidance in fulfillment of the subject Project in COMPUTER ENGINEERING (7^{th}Semester) of Atmiya University, Rajkot during the academic year 2022.

Prof. Rupal Shilu | Prof.Tosal M.Bhalodia

(Project Guide) | **(Head of the Department)**

ATMIYA

UNIVERSITYRAJKOT

CERTIFICATE

Date:

This is to certify that the "**TOURIST GUIDE**" has been carried out by **JIGAR SATHWARA** under my guidance in fulfillment of the subject Project in COMPUTER ENGINEERING (7th Semester) of Atmiya University, Rajkot during the academic year 2022.

Prof. Rupal Shilu	Prof.Tosal M.Bhalodia
(Project Guide)	**(Head of the Department)**

ATMIYA

UNIVERSITYRAJKOT

CERTIFICATE

Date:

This is to certify that the "**TOURIST GUIDE**" has been carried out by **PRIYANK VORA** under my guidance in fulfillment of the subject Project in COMPUTER ENGINEERING (7th Semester) of Atmiya University, Rajkot during the academic year 2021.

Prof. Rupal Shilu — **(Project Guide)**

Prof. Tosal M.Bhalodia — **(Head of the Department)**

ACKNOWLEDGEMENT

We have taken many efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly indebted to Prof. Nirali Borad for their guidance and constant supervision as well as for providing necessary information regarding the Mini Project titled**“TOURIST GUIDE".** We would like to express our gratitude towards staff members of the Computer Engineering Department, Atmiya University for their kind cooperation and encouragement which helped us in completion of this project.

We even thank and appreciate our colleague in developing the project and people who have willingly helped us out with their abilities.

ASHISH SAKARIYA (190002097)

JIGAR SATHWARA (190002102)

PRIYANK VORA (190002126)

ABSTRACT

The ultimate goal of the paper is to explore the requirements of travelers in India and our proposed and developed solution of android application including some basic guidance for the travelers in India. Every year thousands of foreigners from diverse countries come to visit India for different purposes. Most of them come for religious, study, and business purposes. Besides, tourists also visit different places of natural beauty and history of the country. However, being foreigners, the travelers face different types of problems including limited transportation information, problems in understanding different languages and so on. Based on travelers' requirements, we have come up with our online mobile application which can solve their problem during visiting India. The paper illustrates the features, development method, result, and uses of our android application named “Travel Guide ".

Index

List of Figures

List of Tables

Chapter 1

Introduction and Overview

Mobile technology is a field of rapid development. Especially over the past few years, the improvements in mobile technology have been tremendous. The increase in performance of these devices, as well as the additional hardware in form of GPS, sensors, etc., make them powerful devices which can develop comprehensive services to aid people in their everyday tasks. However, the small screens, wireless network and battery presents challenges that need to be addressed carefully.

As one of the biggest sectors in the world, the tourism industry is a very attractive sector for mobile technology[Kabassi, 2010]. Tourists spend considerable time planning their activities, both before and during their visit. The plans made do not take personal preferences into account other than that the tourist chooses attractions based on what he/she thinks are good places to visit. In some cases, reviews from other tourists are available but the user has no knowledge whether this other tourist shares the same passions or preferences.

Most mobiles today have features that are helpful in tourist-situations. GPS (or other means to locate the user's position), accelerometer and maps to mention a few, provides useful information that can be used to aid users. Further, these features combined with a recommender system provide good assistance to users.

In fall 2010 two master theses were written, one targeted recommender systems for tourist applications while the other designed and developed a tourist application. This project aims to develop these further and merge them to achieve an even better application. The end product is a personalized tourist application for smartphones using Android where users are able to find points of interest and their opening hours, receive recommendations, read reviews, give feedback and more.

1.1 Background and Motivation

This project involves two fields, namely tourist applications and artificial intelligence (AI). Recommendation systems face some tough challenges. The cold-start problem is one of them where the problem is giving accurate recommendations to a user who is new to the recommender system. One way to address this is to use *demographic user data*. In this approach, users are required to enter some information about themselves which is used to provide good recommendations. Several papers present recommender systems based on demographic information. Lam et al. [2008] gets the most promising results[Lillegraven and Wolden, 2010] by training a probabilistic model and depending their recommendations entirely on this. *Ask to rate* is another way of dealing with this. Rashid et al. [2002] states that the most direct way of acquiring information

Research Method

is to present items to the user and have them rate these items. Other approaches include fast learners, trusted users, tagging and geographic position data.
presents GUIDE which is a tourist service spanning the city of Lancaster using a client-server model. GUIDE employed a map-based interface and provided information on tourist attractions filtered by context information. Location and previously visited places of the user are among the context information used to provide points of interest (POIs). The user evaluation showed a high level of acceptance, but also presented some challenges regarding its customization strategy in relation to context information.

The Mobile Tourist Service Recommender application, MTSR, is a personalized mobile tourist application which is developed for smartphones with Android OS. MTSR also uses a map-based interface and provides search-functionality for POIs, recommendations, complete travel plans, and more. Context awareness is central in MTSR's design, and user's preferences are used to provide good recommendations. In addition to searching for recommended POIs, the system will send suggestions that the user might enjoy. These suggestions are known as proactive recommendations. MTSR also provides general information about places, as well as ratings from other users regarding the POI in question. The recommendation algorithm used is designed especially for tourist applications and is developed by Lillegraven and Wolden.

Chapter 2

DESIGN & PLANNING

WATERFALL MODEL

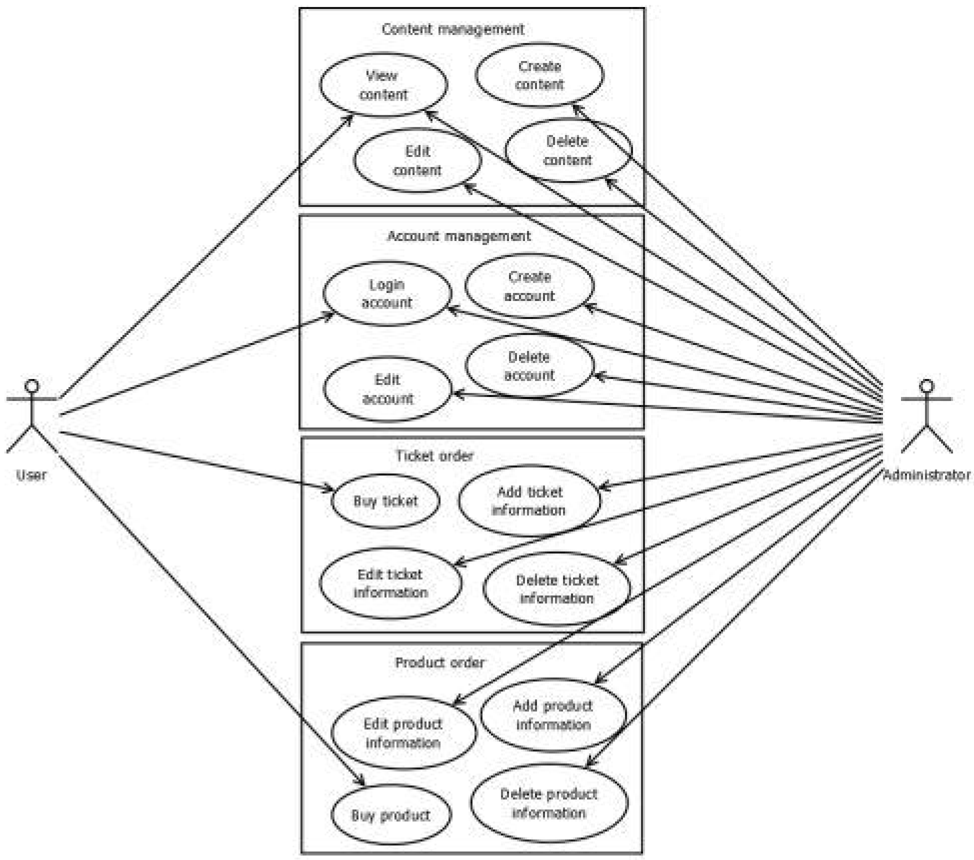
The waterfall model was selected as the SDLC model due to the following reasons:

Requirements were very well documented, clear and fixed.Technology was adequately understood.

Simple and easy to understand and use.There were no ambiguous requirements.

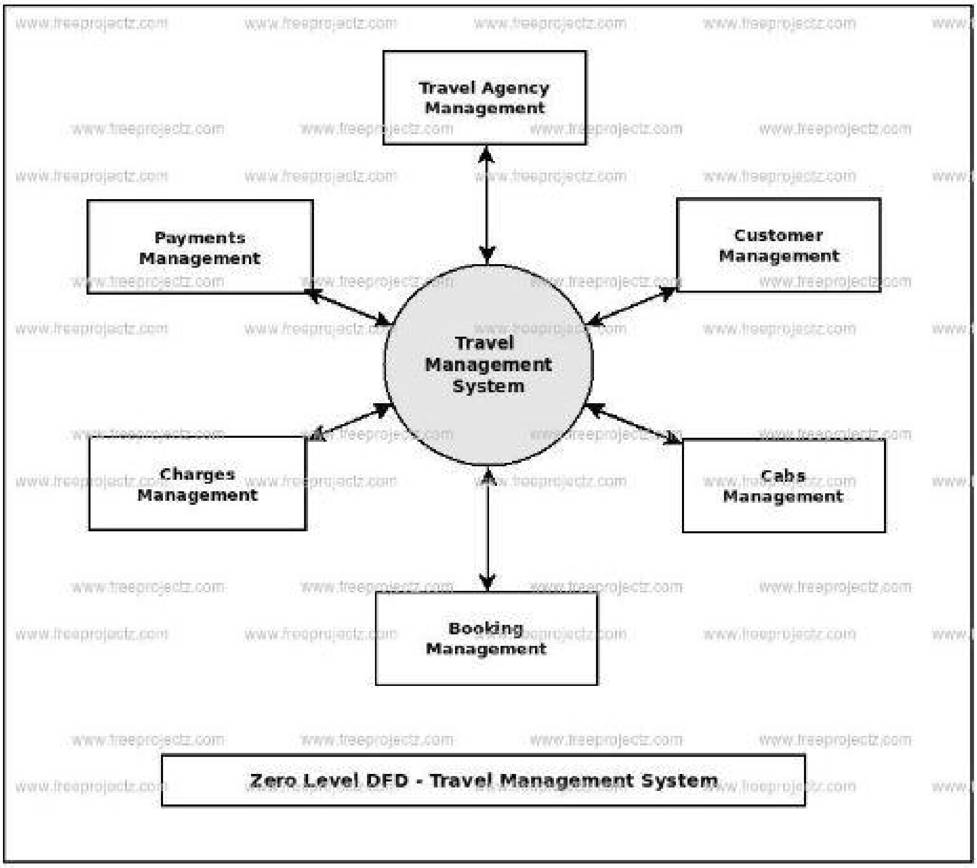
Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review

process.Clearly Defined stages. Well understood milestones, easy to arrange task.

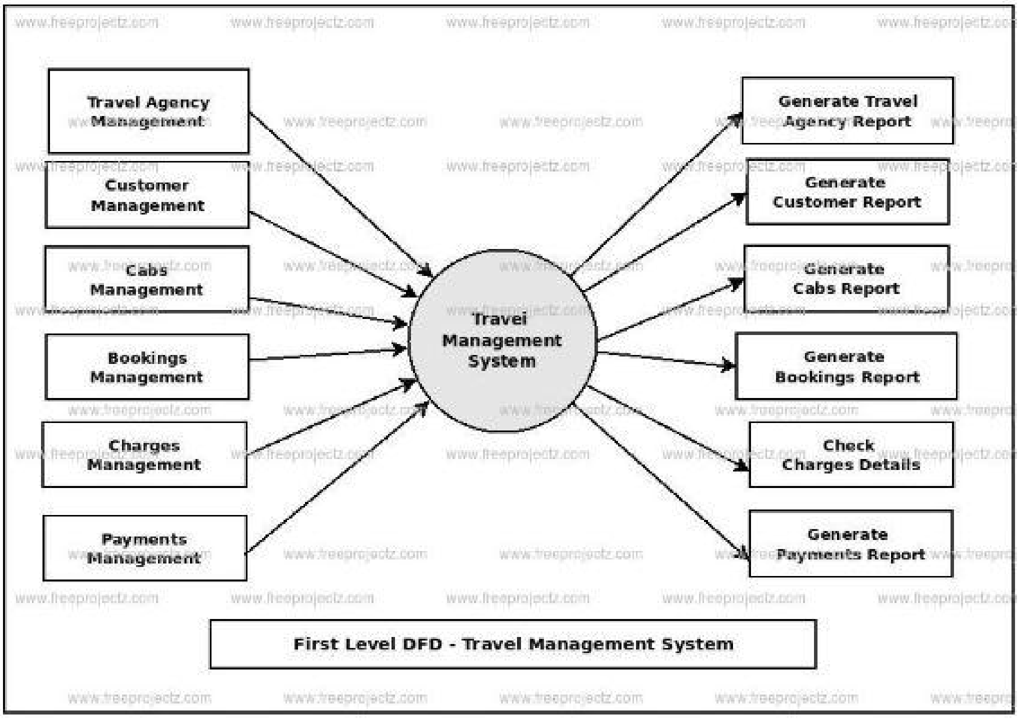


DATA FLOW DIAGRAM

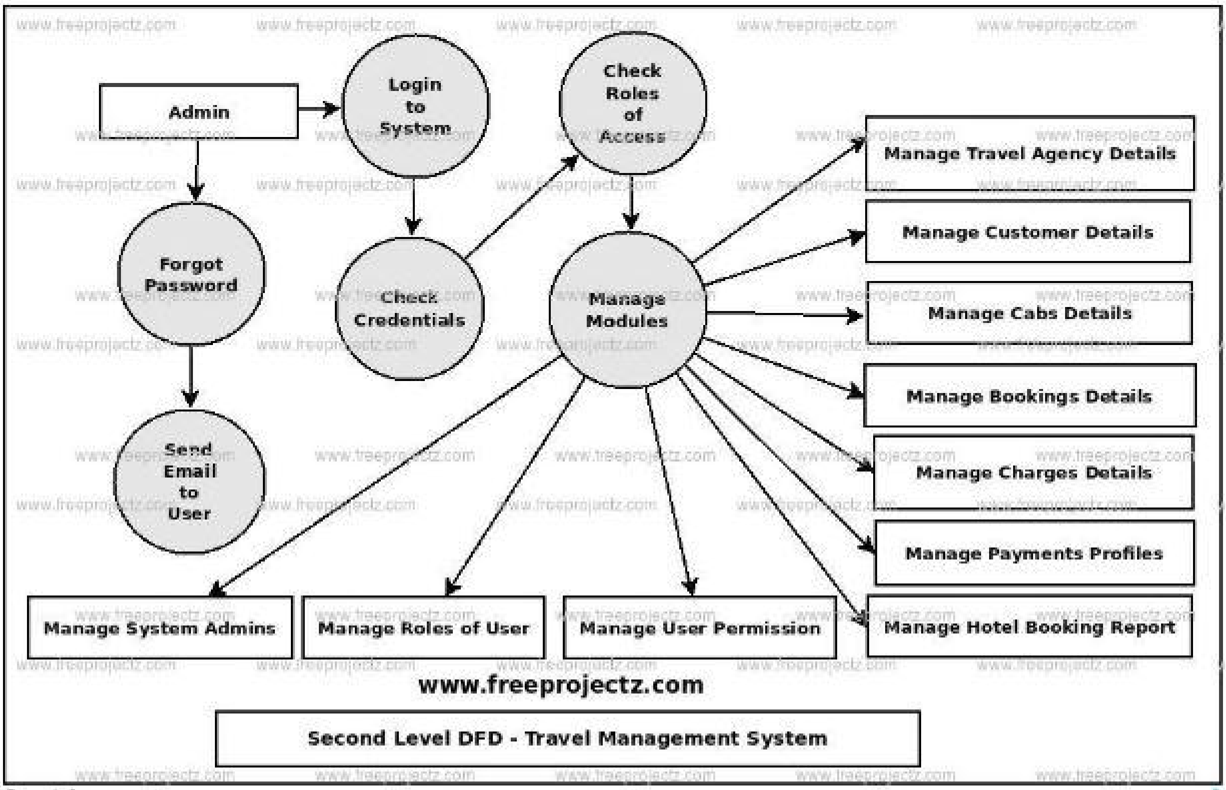
Level 0 DFD



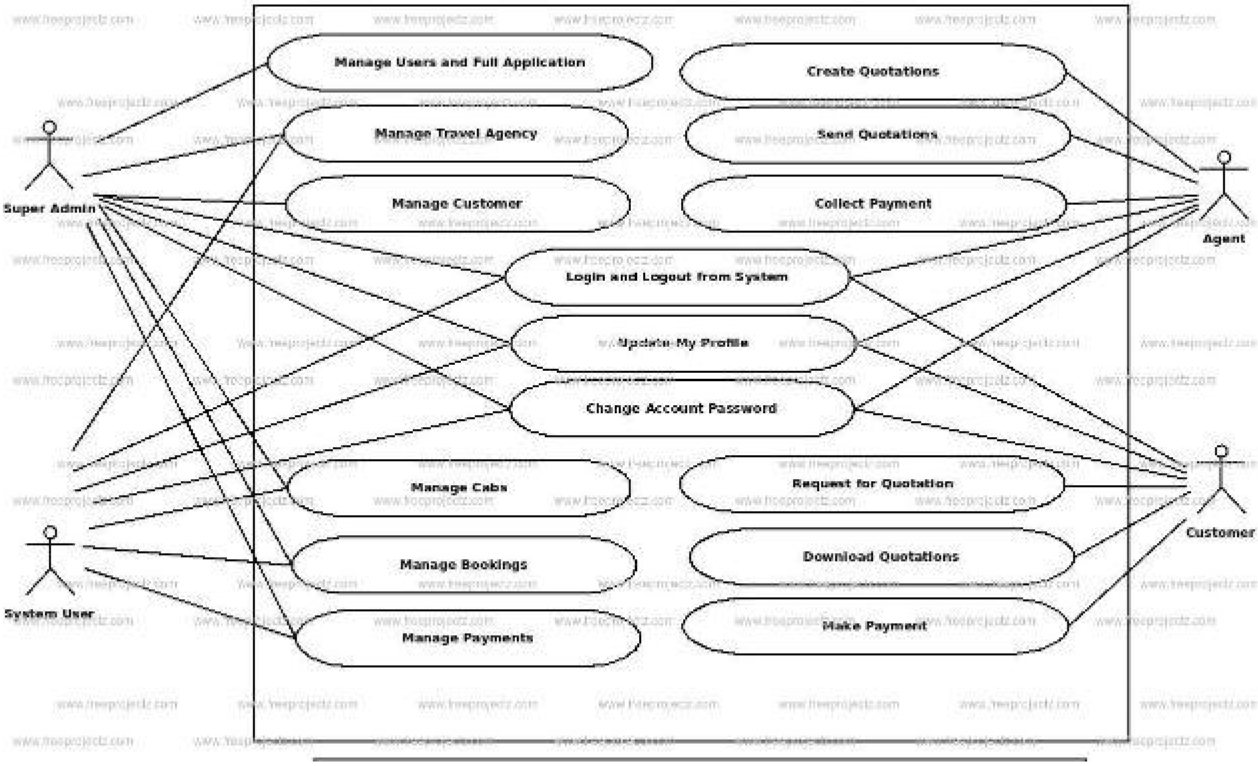
Level 1 DFD



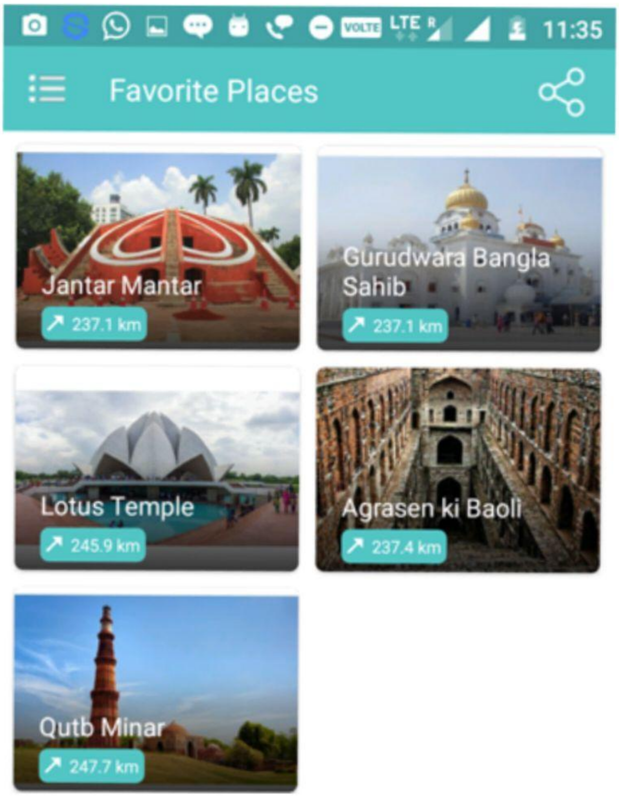
Level 2 DFD

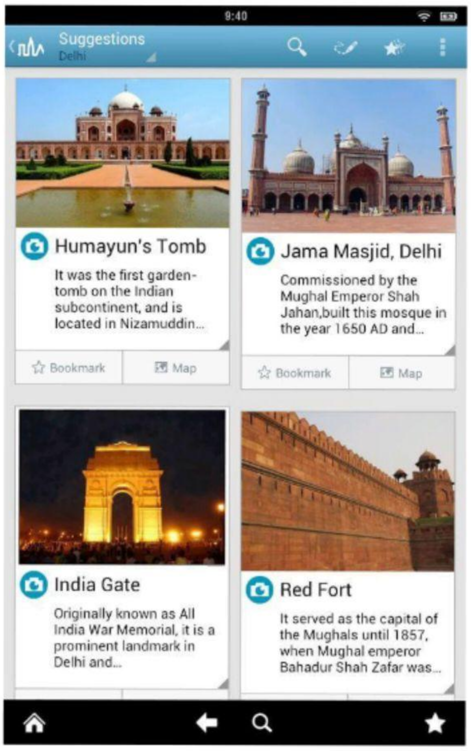


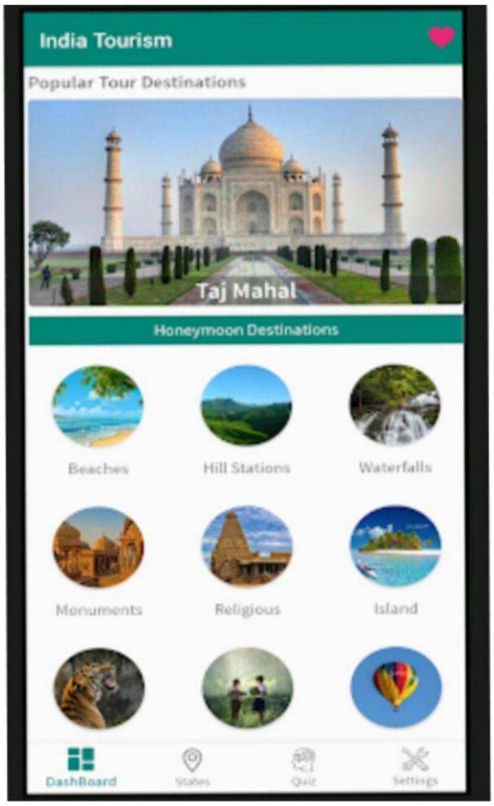
USE CASE DIAGRAM



Design


11:35
Favorite Places
Jantar Mantar
237.1 km
Gurudwara Bangla Sahib
237.1 km
Lotus Temple
245.9 km
Agrasen ki Baoli
237.4 km
Qutb Minar
247.7 km


9:40
Suggestions
Delhi
Humayun's Tomb
It was the first garden-tomb on the Indian subcontinent, and is located in Nizamuddin...
Bookmark
Map
Jama Masjid, Delhi
Commissioned by the Mughal Emperor Shah Jahan,built this mosque in the year 1650 AD and...
Bookmark
Map
India Gate
Originally known as All India War Memorial, it is a prominent landmark in Delhi and...
Red Fort
It served as the capital of the Mughals until 1857, when Mughal emperor Bahadur Shah Zafar was...


India Tourism
Popular Tour Destinations
Taj Mahal
Honeymoon Destinations
Beaches
Hill Stations
Waterfalls
Monuments
Religious
Island
DashBoard

Chapter 3

Background

In this chapter, two master theses will be presented. These master theses are the basis of the project, hence we will provide a short presentation of them.

3.1 The Mobile Tourist Service Recommender

The Mobile Tourist Service Recommender (MTSR) was presented by Wium [2010] in Spring 2010. MTSR is a personalized mobile tourist application for smartphones using the Android operative system. Information is primarily displayed in an interactive map where the user can zoom and pan the map. The user may search for points of interests (POIs), complete travel itineraries and rate places he/she has been. The recommender system present is only created to fill the basic need for such a system.

3.1.1 Architecture

The server architecture has three layers (see Figure 3.1) which all have their own tasks and responsibilities. In addition, a database is used to store various information. *The business layer* and *the data access layer* are the most important parts of the architecture in this project. MTSR is made modular, meaning that it should be more convenient to change specific modules. The data access layer provides an interface to the underlying spatial database, and represents tables in the database as Java-objects to the business layer.

In this architecture the server provides recommendations by sending lists to the client. The server is responsible for all calculations, thus conserving client battery and probably reducing wait time. Request- and response-messages uses XML (eXtensible Markup Language) that are parsed and handled by both sides.

As the client only has minor changes which do not affect the architecture, the client will not be addressed in this section.

3.2 Recommender System for Tourism

Lillegraven and Wolden [2010] presents a recommender system designed for tourism using a Bayesian model. The recommender system also addresses the cold-start

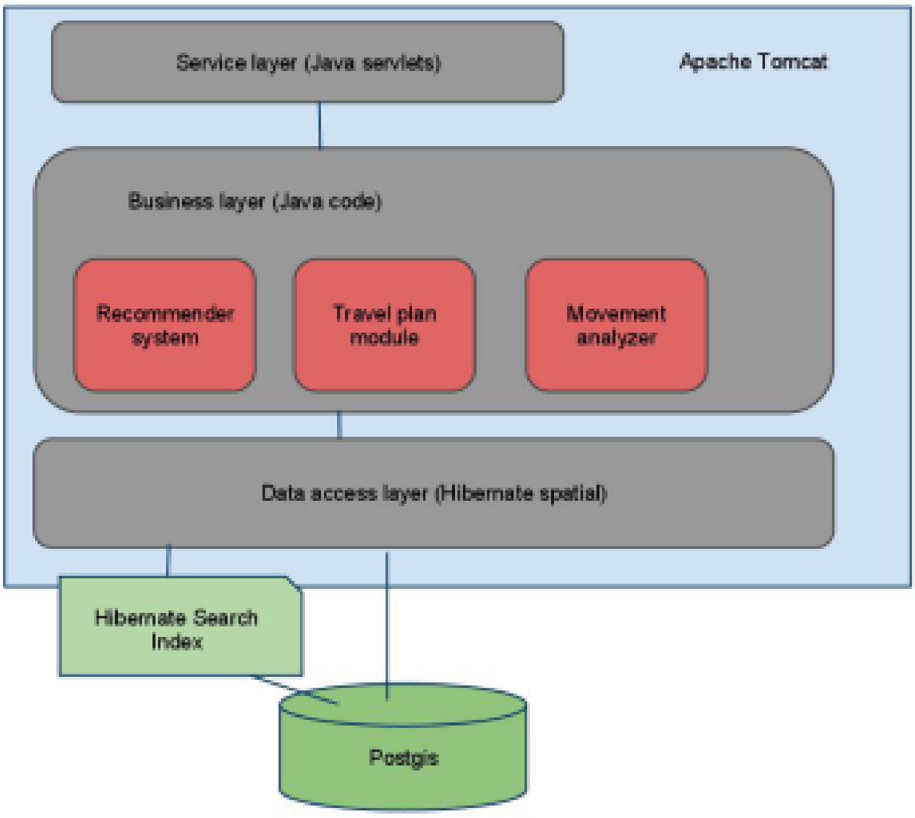


Figure 3.1: Functional server architecture (courtesy of Wium [2010])

demographic survey involving six characteristics, namely age, gender, occupation, type of holiday, nationality and budget. The presented overview of the recommendation process is illustrated in Figure 3.2 and consists of several stages. One of the more interesting parts of this process is that the client is designed to do most of the work. The server updates the Bayesian reasoning model and item information, the rest is left to the client. We will look into this in a coming section.

3.2.1 Architecture

If a new user enters the system, the person concerned will have to submit some demographic information before the server sends updated item information to the client. The client is now in possession of all items, these are sent to the *service filter* in order to minimize the number of valid items that will be used to calculate the preference rating. By using the *Bayesian reasoning model*, a sorted list of items is presented to the user. The ratings made are stored on the client until the user goes online again. At this point the ratings are uploaded to the server which updates the Bayesian reasoning model and sends the latest reasoning model and item list back to the client.

By using several matrices with simple Boolean-variables, they increase the speed of calculations. This, however, requires a specific user- and item-model.

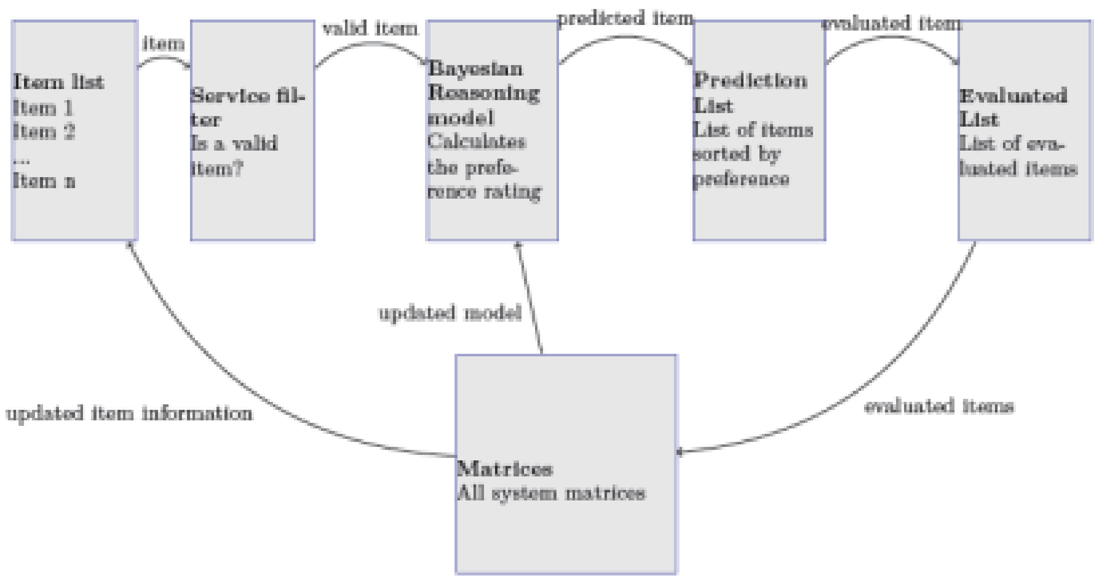


Figure 3.2: An overview of the recommendation process (courtesy of Lillegraven and Wolden [2010])

3.2.2 Concerns

Smartphones are more powerful than they have ever been, but they still face challenges like unstable and variable networks, lack of processing power and battery consumption. The architecture described above requires more computations to be done at the client, decreasing battery life and perhaps increasing the time to handle a request. Lillegraven and Wolden [2010] argue that this approach increases scalability drastically and that the user now does not have to be online to receive recommendations.

12 Recommender System for Tourism

Chapter 4

Implementation

In this chapter we will describe how the new recommender system is integrated, and how this affects the architecture and design of the MTSR system.

4.1 Challenges

During the project we met some challenges which will be presented in this section.

4.1.1 Client- vs. Server-Approach

In Chapter 3, we investigated how the MTSR-system was built before we looked at the architecture of the new recommender system. They presented two very different approaches where recommendations from the original MTSR-system were generated at the server, while they were produced by the client in the second system. Both solutions have their advantages and limitations, however considering the limited capabilities of smartphones and that this matter has not yet been investigated, the safer choice is to have the server handle recommendation requests. The feasibility of the client-approach is left for future work.

4.1.2 User- and Item-Model Issues

As MTSR is modular, it should be easy to replace the old recommender system. However, the new system relies on specialized user- and item-models to achieve the intended advances in the calculation process.

This issue complicates the replacement-process as these models need to be handled in some manner. In addition to replacing the system itself, the database needs to store these additional attributes and some sort of porting has to be done between the matrices of user- and item-data and the original design of these data.

4.2 Architecture and Design

The overall architecture will remain the same for both server and client. The recommender system is replaced and complex matrix-calculations are added. Both the user- and item-model have acquired an additional attribute, namely a matrix that is used by the recommender algorithm.

The old models kept all information in separate variables. To keep consistency between the old

and the new model, we have added porting-functionality between the information-variables and matrices.

4.3 How It Works

Needed demographics are shown in Table 4.2 and consists of six features, namely age, gender, occupation, type of holiday, nationality and budget. Each item has its features displayed in Table 4.2. These attributes are represented in their respective matrices, one matrix for each user and one for each item. Each row in the Boolean-variable column is either 0 or 1 (False or True), meaning that each user will have a matrix with dimension 1x34.

Having a matrix with 34 values for each user seems much, but these are only Boolean-variables and they make the calculations more effective. The same applies for items. We will now describe the new process, and as the solution for users and items behave in the same way we will in some sections only present the user-model.

As we have to keep information in both variables and matrices, we encounter new challenges. We have to keep consistency between them, and we do not have the luxury to only keep one of them (discussed why in Chapter 5). This means that for every occurrence of an update to either a user or an item, the system must perform porting of the updated information to the respective matrix.

User features	
Feature	Boolean variable
Age	<25 25-44 45-64 64<
Gender	Male Female
Occupation	Student Science, Education and Academia Customer Service Sales Technology and Engineering Health care

	Artists Retired Unemployed Other
Type of holiday	One-day stay Regular city break Backpacker
Occupation	Norwegian Other Nordic United Kingdom and Ireland Germany and Benelux Southern and Western Europe Eurasia Central Europe North America Other America China and North-West Asia Other Asia Africa
Budget	Low Medium High

Table 4.1: Needed demographics from the user (courtesy of Lillegraven and Wolden [2010])

16 How It Works

Item features	
Feature	Boolean variable
Night-life	Nightclub Pub Alcoholic beverage
Sightseein g	Art museum Other museum Architectural landmark
Cuisine	Italian cuisine Mexican cuisine Norwegian cuisine Fast-food Asian cuisine
Budget	Low Medium High
Feature average rating location	Other variable continuous between [0.0-1.0] coordinates

Table 4.2: Features attached to items (courtesy of Lillegraven and Wolden [2010])

Chapter 5

Evaluation and Conclusion

5.1 Discussion

The potential in this application is very good. The tourism industry is one of the biggest sectors in the world, meaning that the number of users that would benefit by this application is very high. Also, the rapid development in mobile technology provides good resources that will aid applications to achieve better recommendations, interfaces, accuracy and higher performance.

Keeping information in both variables and matrices makes the process less delicate. Having to store information two places requires more storage and functionality to keep consistency between the two elements. The matrices are not user-friendly considering that when a new user is added the provider of the information (typically the user) would require knowing what each entity in the matrix indicates. By adding some complexity, the system could do this translation for the user. However, we would still have to store both the information variables and matrices as MTSR uses information variables throughout its system and the new recommender system must have matrices to do the calculations as they were meant to.

Converting the system to only use matrices seems feasible. However, before going through with such a major operation, in-depth research of how this will affect performance should be completed. Future iterations should look into this matter.

Servers are supreme in terms of performance and storage compared to smartphones. Lille graven and Wolden [2010] argue that leaving the process of achieving recommendations to the client increases scalability drastically and the necessity to be online disappears. MTSR’s current implementation performs the recommendation process on the server which probably is the best solution. Scalability can also be achieved using multiple servers, and the limited resources of smartphones make it less attractive to let the client do all the work. It will probably drain more battery and maybe increase waiting time, which will lower user satisfaction. Widely popular tourist applications should use multiple servers as these can be deployed when needed or when the coverage area is expanded. As the demand increases, the service will deploy more servers to cope with the requests.

The advantages of a pure client approach is that it does not require wireless access and the client is able to perform operations on its own. As smartphones become more powerful, this approach becomes more feasible and may already be beneficial on some devices. Two versions of the software, one client-based and one with more server-influence, will probably be a good idea in the future but not necessary.

There is no functionality to handle users other than to hard-code their properties. The system would clearly benefit of such functions and is necessary in order to let new users sign up for the

5.2 Summary

The implementation of the improved recommender system presented unforeseen challenges despite the modular nature of MTSR. We were, however, able to solve this by adding an additional attribute to the existing user-model and one to the item-model, and porting between information and matrices. The unexpected challenges, and complexity of MTSR and the new recommender system, made us unable to perform thorough testing in the time at hand. The limited testing so far has showed promising results, but proper testing and further improvements are necessary.

5.3 Contributions

During this project, we have developed MTSR further by implementing a new recommender system. The new recommender system implemented is presented by Lillegraven and Wolden [2010], and uses a Bayesian model to acquire better recommendations.
We have also identified further improvements which are presented in the next section.

5.4 Future Work

As mentioned before, this application has high potential but further testing and development is necessary. In this section we will look at topics that should be addressed.

5.4.1 User Acceptance Testing

In Wium [2010] only a few users tested the framework presented and the test endured over a short period of time. Wium [2010] points out that conducting an evaluation of MTSR applying the so-called Mobile services acceptance model is the natural next step.

5.4.2 Finding Content

MTSR uses a database where the content has to be inserted manually. This is time consuming and highly ineffective. An approach with dynamic content and where data can be fetched from external providers would significantly improve this part of the system. Ratings given by users at the external provider's service will also prove helpful.

5.4.3 Recommender System Testing

The new recommender system should be tested to see that it actually provides better recommendations, and by how much. From a theoretical point of view the new recommender system should outperform the original one. A test, however, is necessary to prove that it actually performs better and to provide test data which can be compared to other systems.

5.4.4 Web Interface

A web interface to manage the server and collect statistics of usage. This interface will mainly have administrative tasks and responsibilities, including correction of data.
Evaluation and Conclusion 19

5.4.5 General Improvements

The master thesis of MTSR presents a few improvements highlighted by the test subjects.

• A more user-friendly search, by using a auto completing search field that also give suggestions on geographical areas that will yield results

• Map that move accordingly to the keyword chosen from the search field

• Option to specify search strategy in terms of ”search for everything”, ”search for stuff recommended for me”, and ”search for top rated points of interest”

Further improvements include:

• The user should be able to set the frequency of proactive recommendations, and also turn the service off

• If a search within a geographical location does not provide any results, the result area should automatically be expanded and display that area

• Users should be able to change the geographical area used for searches easily, perhaps by providing some additional input

• Search results should be categorized by type (restaurant, sight and hotel) for better ordering

• Considering travel itineraries, when users want to add an element they should be able to ask for new recommendations instead of having to find them manually

• The current version of MTSR has no functionality to handle users beyond that they are hard-coded with username and password. Implementing a system regarding this matter must be addressed.

• If feasible, the user should be able to download information prior to their visit. In this way the user will not be depended on the network and its price, but it might raise a battery-concern

Chapter 6

CONCLUSION

Travel applications provide you with all the help that your travel needs, such as location details, ticket booking cost, restaurant details, ticket booking facility, facility to book accommodation, taxi, and explore local experiences. Tourists can also decide when to visit this location.