Traveller: A Novel Tourism Platform for Students Based on Cloud Data

Qi-Ying Hu[1], Chang-Dong Wang[1(✉)], Jia-Xin Hong[1], Meng-Zhe Hua[1], and Di Huang[2]

[1] School of Mobile Information Engineering, Sun Yat-sen University, Zhuhai, China
changdongwang@hotmail.com
[2] SYSU-CMU Shunde International Joint Research Institute (JRI), Shunde, China

Abstract. Tourism is one of the top choices of contemporary university students for leisure on vacation. Desires that students want to travel extensively become increasingly stronger. However, few correct analysis on the tourism market of students in e-commerce results in its immature development, causing a difficulty in making a personalized travel plan to them. Concerning such a situation, it makes sense to provide students with more high-quality plans based on personalized analysis on their tourism preferences. Thus, it is necessary to design a safe, reliable, stable platform for students' travel, named *Traveller*. Cloud computing service is mainly designed for PC machines at present, but it is a trend of future development to combine the cloud computing and mobile application. From the viewpoint of technique, powerful computing capacity and storage capacity are sufficient to improve the user experience. From the viewpoint of mobile user, a sharp increase of users in recent years has become a driving force to push the advancement of mobile application. As a consequence, mobile computing will soon prove its prevalence. In this paper, tourist information will be collected by communicating through a mobile terminal to send requests to the server in the cloud, making full use of cloud server's features such as efficiency and convenience to update.

Keywords: Travel · Cloud-platform · Database · Application

1 Introduction

In step with the improvement of people's living standards, the tourism industry has become an indispensable part of people's life [1]. It is also universally popular among university students. As a special group, university students with independent economy, self-ability, relatively more spare time and adventurous spirit play an essential role in the whole tourism market. Moreover, the students are different from the other groups in many aspects. Students prefer group tourism and they also prefer natural attractions when choosing sites. Due to the limitation of school and the family, they generally lack travel experience, worry and hoping to get a more intuitive advice. At the same time, the young generation depends

S. Guo et al. (Eds.): CollaborateCom 2015, LNICST 163, pp. 26–35, 2016.
DOI: 10.1007/978-3-319-28910-6_3

more on the mobile phones, so the Apps for tourism find favor with them along with the gradual development of intelligent terminal technology.

Currently some Apps have been developed to meet this needs, like Qunar [2], Ctrip [3], Umetrip [4], etc. However, they have the following drawbacks [5]. Firstly, the pertinence of the Apps is not strong, making the App lack individuation and the features greatly reduced in practice. Secondly, Because the function of the App is complex and the target population is not explicit, it is obvious that only a few can find favor with people among the numerous Apps. Finally, for best reliability and performance, the traditional App should be constantly updated, which causes trouble for users to download installation package of the latest version over and over otherwise they cannot experience the benefits from the new features and functions.

Cloud computing is an emerging ICT framework and service model. It is an evaluation of distributed processing, parallel processing and network computing. It has advantages in high reliability, rapid deployment and on-demand service, which makes it extremely popular at present. Based on these merits, we intend to make use of cloud computing to develop a searching platform which can provide users with tourist information including student discount. That is why we want to design a mobile application on the basement of cloud platform.

Mobile cloud computing is a new computing paradigm to meet the urgent need not for applications and services of resource-constrained mobile devices. To overcome resource constraints on mobile devices, a general idea is to ship code and data to the cloud and back to prolong the battery life and speed up the application. Besides saving energy, cloud computing also enables enhanced mobile experiences by employing a client-server framework that consists of two parts, which run on the mobile device and the cloud, respectively. Overall, mobile cloud computing can solve the problem of energy consumption and bring convenience to the mobile users [6].

In summary, this project is aimed at developing a searching platform for students' traveling convenience based on mobile cloud computing. In addition to implementing some common functions in current applications, this cloud application can provide more personalized services. The main purpose of this application is to help students make a good choice. It can provide users with some useful information about traveling depending on their situations and preferences.

1.1 Related Work

Many mobile tourism applications have been released into the market in recent years, but most of them aim to bring users convenience to buy tickets or check schedules. Umetrip [4] is a mobile application released by Civil Aviation Administration of China to query flight information, then users' account can also save each flight they have taken or going to take, further more it can remind users of the latest flight they will take. Ctrip [3] can be defined as an online travel agency, providing ticket service, tourist information and gather strangers together to have a trip. Users can buy tickets online to get some discount, share their own travel experience and read some strategies posted by others. Airbnb [7] is aimed

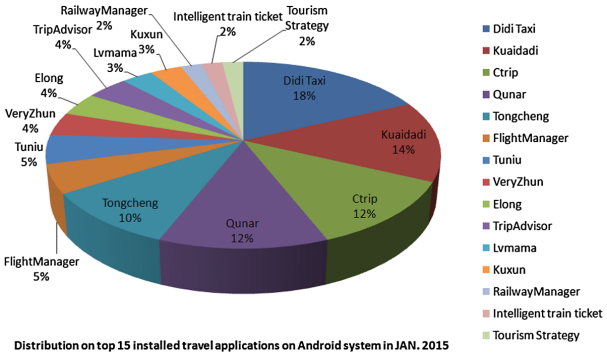


Fig. 1. Ranking of travel applications in China.

at making visitors feel like home at other places. Visitors can find local houses to stay for a short time to experience the local way of life. The living conditions are similar to those in hotels. Serving as private driver in more than 50 countries, Uber [8] provides different levels of car such as UberX and UberBlack for users. It makes users feel more comfortable and convenient to take this kind of car than a taxi. HHtravel helps users to have a top-level trip, such as Michelin restaurants and five-star hotels. The HHtravel [9] staffs can also organize a trip according to users' requirements. Lonely Planet Traveller [10] is transformed from a magazine with the same name. It introduces countries in a very detailed way by showing information about eating, playing, traffic, living, and even festival dates. Applications listed above are only a few of the whole travel application market. For more details we can see Fig. 1 about travel application download ranking in China in January 2015 [11].

In China, users prefer to use Ctrip [3], Qunar [2] and Tongcheng [12] (totally more than 30 %) which are defined as online travel agencies.

In conclusion, there are so many applications aimed at different travel aspects, such as tickets service, flight information, traffic, attraction information and so on, but few applications focus on university students who have gradually become main potential customers of tourism. Moreover, students have a vast number of requirements that cannot be satisfied by those applications. For example sometimes students want to check whether a specific scenery spot has student discount, yet unfortunately the information cannot be found easily. Also those most fascinating travel places are not actually what students are really interested in. They want to receive more recommendations from the same generation. In general there is no such an application that is personalized enough for university students. We expect to develop an application to provide convenience to students, giving them more opportunities to go out and see more. Meanwhile a students-friendly application can also save time for schedule with travel experience comments by those who have similar hobbies and preferences.

1.2 Our Work

To address the above issues, we develop a tourism platform for students based on the Aliyun [13], which sets up server in a virtual machine by using Apache and the remote database constructed in the server. Then the client can connect and call the database through the server program, as well as doing insertions, deletions, modifications and query to the database. In cloud server, we write Python code to make server receive requests from client, and obtain data from database according to the request. The cloud platform can run the code all the time to satisfy the need that the users get tourist information whenever they are. The configuration on it is easy to implement and the cost is relatively cheap.

Our idea is to develop a new application named *Traveller*, providing students with favourable tourist information. Students can log in the system and search for the destination they want to travel, get information of surroundings, and they can also grade the viewpoints where they have been to and make recommendations to others. The personalized recommendation is achieved on the server so that students can get more suitable and appropriate recommendations according to their own preferences. Even more, a student can make more like-minded friends by using this App and travel with them.

As to these functions, the framework of the application can be developed in two aspects: client and server. Eclipse is a good extensible development platform, which is open-source and able to support varied coding language. We can use Java to implement the function and this tool can help to construct the interactive interface of the application and implement the encapsulation of the data transmission via GET or POST request. For the server terminal, the cloud is used for its power in function and convenience in operation. All data will be stored in cloud and transmitted to the client in need.

Traveller is aimed at serving university students, so the objective is clear and functions are highly targeted. The current market lacks such App, so our application has strong market competitiveness. In addition, by keeping all the data in the cloud platform, it is easy to update all data and features, avoiding the need to constantly update the App to obtain the new properties.

The rest of the paper is organized as follows. In Sect. 2, the background knowledge of cloud computing is introduced. In Sect. 3, the proposed platform is described in detail. Section 4 reports the experimental results. We conclude our paper in Sect. 5.

2 Background

Based on the cloud platform, the main use in our application development is cloud computing. PaaS (Platform as a Service) [14] is referred to provide service for client. Utility computing is used to compute recommendation algorithms and also combine computing with storage. Socket is used as physical connection between the server and client which can transmit data.

Data transmits in different terminals by HTTP, which are a set of rules for computer communication through network [15]. Computer specialists designed

HTTP to satisfy the needs from client (such as mobile client), which requests information and service from Web server. All HTTP connections are formed with a set of requests and responses [16].

As a script, Python [17] deals with the requests coming from Android system while database system can utilize MySQL of the Ali virtual machine to aid this behaviour. What should be done first is to install MySQL and Python in the Ubuntu system. MySQL is useful to establish the database and it will be called in the Python script. This process needs the support of the library named MYSQLdB [18,19].

The server will get the requests send from the Android terminal, pick up the related datum, such as information about view, and return them as the format of string. Based on a subset of JavaScript, JSON [20] (JavaScript Object Notation) is a lightweight format to interchange data, it uses a text format which is completely independent to language, it also has similar habits like the C language family (including C, C++, Java, JavaScript, Python etc.). These properties make JSON ideal language to exchange data. It is not only easy to read and write, but also easy for machines to parse and generate.

3 The Proposed Platform

Based on the cloud server of Ali, we develop our platform in the Ubuntu system with SQL database [21]. Aliyun is a distributed application environment, providing developers an environment to develop server-side applications rapidly, and simplify the system maintenance work. For developers, it can be logically divided into two parts. One is management system which is responsible for providing management interfaces and storing data. The other is running system that is responsible for calculating, serving as a middle-ware to transmitting data between client and database, which promises the data security for different users.

- For the client-side, Android applications will send a request to get data from sever or send data to the server to update databases. Users can search for any tourist attractions or grade for those they have been to.
- For the server-side, when the server receiving a request, it will handle it by either getting data from databases or updating data to databases.

There is a highlight in our development. It is a double recommendation systems.

- The outer recommendation system is a global one. It can offer the most popular scenic spots to all users.
- The inner one is a personalized recommendation system. It will recommend scenic spots to users based on their preference.

3.1 Get Data

Users can log in the App by inputting username and password. The terminal will send a URL [22] of the username to the server to get the corresponding

```
for item in res:
    data = dict()
    data['cityname'] = cityname
    data['viewname'] =item[1]
    data['price'] = item[2]
    data['picture']=item[4]
    data['discount'] = item[3]
    temp.append(data)
result['result']=temp
return json.dumps(result)
```

(a) The server returns the data in JSON format.

```
totalscore = viewscore*0.5 + foodscore*0.2 + servicescore*0.3
totalscore = str(totalscore)
con = Connect()
param = dict()
q1 = "lock table evaluate write"
r1 = Exec(con,q1,param)
para['totalscore'] = totalscore
query = "replace into evaluate values('%s','%s','%s',%f','%f','%f')
res = Exec(con,query,para)
q2 = "unlock tables"
r2 = Exec(con,q2,param)
```

(b) The server returns the data in JSON format.

Fig. 2. Illustration of login.

password in the cloud and check whether the user can enter or not. After logging in successfully, the user will enter the searching interface. Using the name of city or attraction input by the user, the client packages URL and uploads it to the server, the server returns the result of all attractions searched in the database. Then the client receives data and parses the text information and pictures which will be displayed. Finally, in the list of results, user can click on appropriate spot, then it can jump to the details interface. The method of getting detailed information is the same as above.

The client is realized by JAVA language on Eclipse platform [18,23], it can get requests, set the connection timeout and read timeout settings, put the data on the URL and connect to the server. The server parses the URL string and extracts its data, through the JSON format statement it searches for the needed information and returns to the client as well as in JSON format [24,25], as is shown in Fig. 2(a).

3.2 Update Data

A function of this application is to grade a scenic spot if the user has already logged in and has been to this site. Score is divided into scenery score, food score and service score. After the user clicks the OK button, the client packages URL including the three scores, the name of view and username, and uploads it to the server. The server receives data and inserts it into the database by using the triggers [26] or update the data in the database if there exists the same item. Figure 2(b) illustrates the update of the score data.

3.3 Data Storage

The main function of our application is to search tourist information. Therefore one of the most significant components is data storage. Using cloud platform to store data is convenient and costless. A database with several tables satisfying different requests is created in our virtual machine by using MySQL. When there is a request to get view information, the database can be accessed easily [19].

3.4 Recommendation Algorithms

Our application development is based on the cloud platform from Ali and uses the Ubuntu System as server. In the server, we use MySQL to design database. To deal with HTTP request sent from Android client, we use Python as script language, which can also connect to database and retrieve data and then send back to client [27]. In our application, the key functions are two recommendation systems.

1. The first one is global recommendation system, which can be used before users log in the system. This option can provide all users scores of tourist attractions which are highly recommended by users who have been to them.
2. The second recommendation is particularly designed for certain users. When a user logs in his/her account and presses the menu button, the other hand-picked option will be presented to the user. For this option, we accomplished a personalized recommendation system to help users to find what they are really interested in.

Global Recommendation System. For a certain tourist attraction, we collect scores from users who have been to this place. The scores are divided into three parts, score for scenery, score for food and score for service. Three parts carry different weight in total scores. Scenery is the most importance, taking a percentage of 50, while food takes 20 percent and service takes 30 percent. We build the system based on all users' grading and take an average. After that we sort the average scores and pick out top five for all users. Global recommendation is not designed for a particular user so every one can reach the same view.

Personalized Recommendation System Based on Slope One. Slope one algorithm [28] is a kind of grading-based collaborative filtering algorithm. It is a simple algorithm in recommendation for it only needs a little bit of linear algebra without statistics involved. It is used to predict target users' grading for which are not done before according to any other users who have already graded. In slope one, what we need is a crowd of users, a group of views and some gradings by users who have been to the certain place. What we do is to recommend some places to a target client, which he may be interested in but has not graded yet. The algorithm consists of the following three steps [29]:

1. Calculate any two views' scores difference.
2. Input a user's ID, get his grading record and predict those not grading.
3. Sorted by grading and output top five views as what user is interested in.

4 Experiments

In this section, we will validate the proposed method. We test the performance of connection and data transmission between Android client and cloud server. For this purpose, we construct a database and make a simulation.

4.1 Data Description

We do some tests about the connection and data transmission between client and server. To this end, some data about tourism information are used in our experiments.

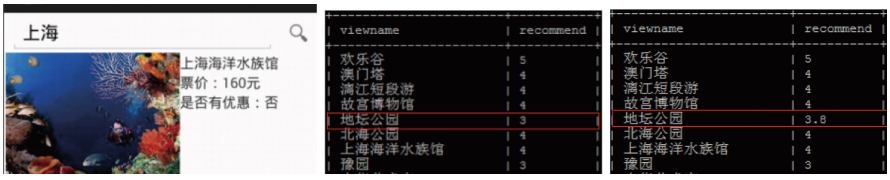
There are four tables in the database. The View table consists of information of all viewpoints. It contains 5 types of items including the name of city, the name of view, discounts for students, recommend score, and photo about the view. The student table contains information of all students including student ID and password after encrypting. Viewinfo, used to do recommendation, incorporates information of some viewpoints which have high recommend scores. The recommend score comes from the table Evaluate, which contains the scores of all students.

We test our schemes over these aspects:

1. whether the client can get the data from the server and send the data to the server successfully.
2. whether the data in the database has changed after being modified.
3. whether the slope-one algorithm works or not.

4.2 Results

Now the results of our experiments can be shown as follows:



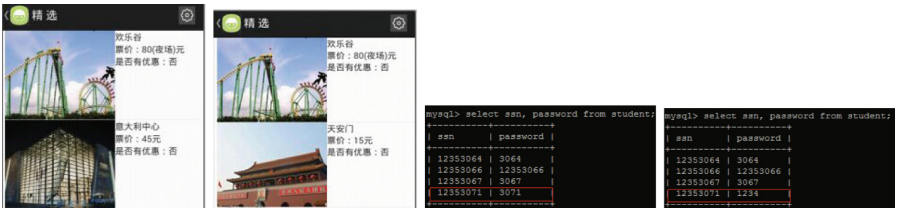
(a) Searching for Shanghai. (b) Recommendation index before scoring. (c) Recommendation index after scoring.

Fig. 3. Recommendation index when scoring.

Test of Login and Search. User can log in successfully when password is correct. From Fig. 3(a), we can get the sites given the name of the city. It proves that the method using in connection is correct.

Test of Grade and Recommendation. From Fig. 3(b), 3(c), it can be seen that in this test, after being graded, the score of this site has changed.

From Fig. 4(a), 4(b), the recommendations are different when the user has logged in, proving that the recommendation algorithm is useful.



(a) Recommendation before Login. (b) Recommendation after Login. (c) The password before modifying. (d) The password after modifying in database.

Fig. 4. Detail information.

Test of Modify the Password. It can be seen that in this test, the user can modify the password successfully. In Fig. 4(c) and 4(d), the password in the database will be changed if a user modifies the corresponding password.

From all results, the application can realize all the basic functions like connecting to the server, getting recommendations and modifying the database.

5 Conclusion

In this paper, we mainly introduce the development of our application based on cloud platform and implemented in Android client. We build database for our application in the cloud and it can be executed in the background; based on this database in cloud we can use an Android mobile-phone to run our application. Using the application, we can input the name of city or specific scenic sport we expect to visit and then get some information of tourist attractions. One novelty is that, our application is particularly designed for students particularly. As to the targeted users, a channel is built to check whether there is a discount for students. Moreover, each user will get some corresponding recommended attractions, which is implemented based on the slope one algorithm. Security is also guaranteed by verifying user's name and password.

In our future work, we want to implement more functions. Firstly it will have a function to provide a world map for users so they can mark where they have been to. Secondly, some information about our friends' wish list can be presented so that we can make a plan to travel with friends.

Acknowledgment. This work was supported by NSFC (61502543), CCF-Tencent Open Research Fund, and the PhD Start-up Fund of Natural Science Foundation of Guangdong Province, China (2014A030310180).

References

1. Zhang, Y.: Analyse the effective development of university students' tourist market. J. Inner Mongolia Agr. Univ. **4**(10), 105–106 (2008)
2. Qunar. http://app.qunar.com/
3. Ctrip. http://app.ctrip.com/
4. Umetrip. https://itunes.apple.com/cn/app/id480161784?mt=8
5. Zhang, L.: The survival and development prospect of China tourism internet, China Tourism Development, pp. 1–3 (2008)
6. Liu, F., Shu, P., Jin, H., Ding, L., Yu, J., Niu, D., Li, B.: Gearing resource-poor mobile devices with powerful clouds: architecture, challenges and applications. IEEE Wirel. Commun. **20**(3), 14–22 (2013)
7. Airbnb. https://itunes.apple.com/us/app/airbnb/id401626263?mt=8
8. Uber. https://itunes.apple.com/us/app/uber/id368677368?mt=8
9. HHtravel. http://www.hhtravel.com/
10. Lonely Planet Traveller. https://itunes.apple.com/us/app/lonely-planet-traveller/id563539341?mt=8
11. Ranking of the tourism Apps. http://www.ctcnn.com/html/2015-02-10/14628276.html
12. TongCheng. http://www.ly.com/
13. Aliyun. http://docs.aliyun.com/
14. Deng, R., Qin, C., Xie, X.: The application status of the mobile cloud computing and analysis of the existing problems. Chongqing Univ. Posts Telecommun. **6**(24), 716–723 (2012)
15. Zhang, W.: Scheme in transmission and storage security of user data on cloud computing, Beijing University of Posts and Telecommunications, pp. 4–71 (2011)
16. Kurose, J.F., Ross, K.W.: Computer Networking A Top-Down Approach, 4th edn. China Machine Press, Beijing (2014)
17. Hetland, M.L.: Beginning Python: From Novice to Professional, 2nd edn. The people's posts and telecommunications press, Beijing (2010)
18. Xu, Q., Wang, Z.: Application of cloud computing: the practice of developing (2012)
19. Wu, J., Ping, L., Ge, X., Wang, Y., Fu, J.: Cloud storage as the infrastructure of cloud computing. In: International Conference on Intelligent Computing and Cognitive Informatics (ICICCI), pp. 380–383 (2010)
20. JSON. http://www.json.org.cn/
21. SQL. http://www.w3school.com.cn/sql
22. Guo, H.: Android Application Development Comments. Electronic Industry Press, Beijing (2010)
23. Fengsheng, Y.: The reveal of Android application and development, pp. 130–200. Beijing: Mechanical Industry Press, Beijing (2010)
24. Smith, B.: Beginning JSON. Apress, New York (2015)
25. The Android use JSON format to transmit data. http://blog.csdn.net/lserein/article/details/6239460
26. Schwartz, B., Zaitsev, P., Tkachenko, V.: High Performance MySQL. Electronic Industry Press, Beijing (2013)
27. Fielding, R., Kaiser, G.: The Apache HTTP server project. IEEE Internet Comput. **1**(4), 88–90 (1997)
28. Wang, Y., Li'ang Yin, B.C., Yu, Y.: Learning to recommend based on slope one strategy. Web Technologies and Applications, pp. 537–544. Springer, Heidelberg (2012)
29. Lemire, D., Maclachlan, A.: Slope one predictors for online rating-based collaborative filtering. In: SDM, pp. 967–968 (2007)