International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 2 | ISSN : 2456-3307

College Finder & Recommender Web Application System

Nishank Samant, Nikhil Manjarekar, Ashwini Mhavarkar, Ashutosh Nagaonkar, Prof. Medha Kulkarni
Department of Information Technology, PVPPCOE, Mumbai, Maharashtra, India

ABSTRACT

College finding process has been a hectic work as it comprised of considering all the factors (like Location, Tuition-Fees, Courses offered) that matter to the student who wants to apply to the college. This was a very time consuming process hence we propose College finder system which provide a wealth of tools and information that assist individuals in their postsecondary educational endeavors. This makes it easier for prospective students to locate a college or university suited to their personal needs and tastes as well as get appropriate recommendations based on their current academic profile.

Keywords: College website, Recommendation system, PHP website, Web application, College finder

I. INTRODUCTION

Initially it was a toilsome process to find appropriate colleges in minimum span of time as we need to look out for all aspects/features of colleges like its courses offered ,fees, cutoff's and general information about it. Thus to simplify this process we have developed a college finder and recommendation system which helps users to search colleges as well as filter them according to various filters selected by users to minimize the results and simplify the process. The recommendation system works on the data provided by the user on their profile which basically consists of academic data. The user can also add the specific colleges to their wish-list to review it faster. There is an extra functionality called career section provided where the user can view the career options that are available to opt for according to their liking. This fully functional website is programmed in PHP language and the data is stored in the MySQL Database.

II. METHODS AND MATERIAL

Existing System

Whenever we implement new system it is developed to remove the shortcomings of an existing system. The computerized has more Edge over the manual system. Getting an admission in good College/University is a very tiring and complicated process. It may take weeks even months to find relevant college which probably provide admission to student and even after wasting much time, students not find colleges as per their requirement.

So firstly we will introduce the existing system, the existing system is based on manual system i.e. users have to search each and every college individually for his/her admission process, which takes lot of time to get relevant or deserving colleges list.

The Existing system earlier consisted of following operations-

- Manually Searching through search engines
- Getting data from respective college websites.
- Getting data from third party websites which hardly considered the necessary parameters for the college searching tool.

1. Absence of Recommendation System

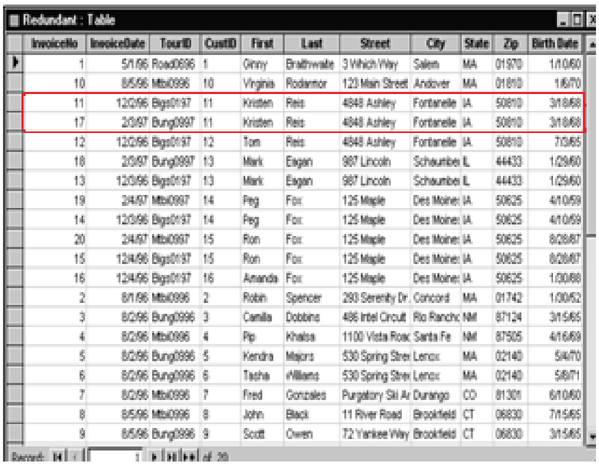
The recommendation system is a very essential tool to provide list of colleges to the users which is evaluated automatically by the program based on the input data provided by the user after logging in the system

1. There are various sites which provide college finding facility but there are very few who provide recommendations to their registered users.
2. This system is necessary to reduce the efforts of the user for getting an insight of which colleges he can apply for and have chance for getting an admit.
3. This system analyzes and generates the list of colleges that a user can apply for based on his /her educational backgrounds or on the scores of the various entrance exams like GATE/SAT/CAT etc.

2. Data Redundancy

Take a few moments to inspect Figure 1. Notice that all of the fields now appear in a single table. Although this may seem simpler, it turns out to be a terrible idea. Look at the third and fourth rows in Figure 1. These two rows reflect the sales made to Kristin Reis. But look more closely and you'll notice that Kristin's information (First, Last, Street, City, State, Zip and Birth Date) is stored twice because she has been the customer in two different sales. If she had been the customer in 50 sales, this design would require that we store her name, address and birth-date 50 different times. Storing the same field values more than once (unnecessarily) is referred to as data redundancy. Three problems are caused by data redundancy. The first is that storing values multiple times wastes space.

Under a proper design, Kristin's information is stored only once, in her record in the Customers table. The second problem is that when a field value changes, multiple occurrences need to be updated. For example, if Kristin moves, we'll need to change the values for her Street, City, State and Zip in multiple records. The third problem occurs if we forget to change the values in any of the records. The database would then have inconsistent data.


Redundant : Table

InvoiceNo	InvoiceDate	TourID	CustID	First	Last	Street	City	State	Zip	Birth Date
1	5/1/96	Road0696	1	Ginny	Braithwaite	3 Which Way	Salem	MA	01970	1/10/60
10	8/5/96	Mtb0996	10	Virginia	Rodarmor	123 Main Street	Andover	MA	01810	1/6/70
11	12/2/96	Bigs0197	11	Kristen	Reis	4848 Ashley	Fontanelle	IA	50810	3/18/68
17	2/3/97	Bung0997	11	Kristen	Reis	4848 Ashley	Fontanelle	IA	50810	3/18/68
12	12/2/96	Bigs0197	12	Tom	Reis	4848 Ashley	Fontanelle	IA	50810	7/3/65
18	2/3/97	Bung0997	13	Mark	Eagan	987 Lincoln	Schaumbe	IL	44433	1/29/60
13	12/3/96	Bigs0197	13	Mark	Eagan	987 Lincoln	Schaumbe	IL	44433	1/29/60
19	2/4/97	Mtb0997	14	Peg	Fox	125 Maple	Des Moine:	IA	50625	4/10/59
14	12/3/96	Bigs0197	14	Peg	Fox	125 Maple	Des Moine:	IA	50625	4/10/59
20	2/4/97	Mtb0997	15	Ron	Fox	125 Maple	Des Moine:	IA	50625	8/28/67
15	12/4/96	Bigs0197	15	Ron	Fox	125 Maple	Des Moine:	IA	50625	8/28/67
16	12/4/96	Bigs0197	16	Amanda	Fox	125 Maple	Des Moine:	IA	50625	1/30/88
2	8/1/96	Mtb0996	2	Robin	Spencer	293 Serenity Dr.	Concord	MA	01742	1/30/52
3	8/2/96	Bung0996	3	Camilla	Dobbins	486 Intel Circuit	Rio Ranchc	NM	87124	3/15/65
4	8/2/96	Mtb0996	4	Pip	Khalsa	1100 Vista Roac	Santa Fe	NM	87505	4/16/69
5	8/2/96	Bung0996	5	Kendra	Majors	530 Spring Stre	Lenox	MA	02140	5/4/70
6	8/2/96	Bung0996	6	Tasha	Williams	530 Spring Stre	Lenox	MA	02140	5/8/71
7	8/2/96	Mtb0996	7	Fred	Gonzales	Purgatory Ski Ar	Durango	CO	81301	6/10/60
8	8/5/96	Mtb0996	8	John	Black	11 River Road	Brookfield	CT	06830	7/15/65
9	8/5/96	Bung0996	9	Scott	Owen	72 Yankee Way	Brookfield	CT	06830	3/15/65

Record: 1 of 20

Figure 1: Data Redundancy Problem in table

Other Problems in Existing System

- Browser compatibility issues.
- Searching tedious.
- Do not cover all the modules that should be present for a college finding tool.

III. RESULTS AND DISCUSSION

Proposed System

To overcome the drawback of existing system we have developed new system, which is a web based application in which the student/parent can simply search or explore the colleges anonymously. It also provides a special unit which specifies various career opportunities and course details for Masters/MBA applications. It is most reliable & time saving system for searching relevant and appropriate colleges from the comfort of your desktop.

A. Architecture of Proposed System

The components of the system are as explained below:

- **College Database:** The database is a repository for storing the college details as well as user details. The user related details can be updated through logging in their account.
- **College Finder Website:** The Website acts as an interface between user and database and helps to display information to the user.

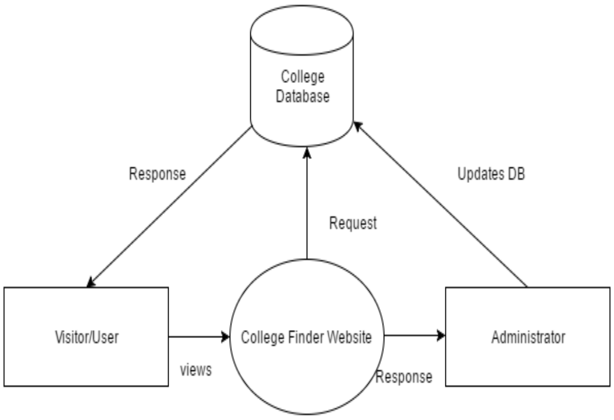


Figure 2 : College Finder System Architecture

- **User:** The user retrieves information from the database about college data and manage their account.

- **Administrator:** The administrator updates the database as well as manage the users.

B. Operations of Proposed System

The proposed system has following operations as explained below:

1. Searching

Searching of colleges is used to find an appropriate college quickly. This feature is used to give accurate search results. The user enters a keyword related to the college and the colleges satisfying that keywords are retrieved.

2. Filtering

Filtering of colleges is used to find colleges based on various properties. This feature is used to give relevant colleges satisfying one or all the properties. The user enters options that they require (like….Location, Degree offered).

3. Recommendation

It helps to recommend colleges to user based on the input data provided by the user after logging in their profile. The data of what user expects in the college is saved on the database and the colleges satisfying the data are retrieved.

4. Wish-List

It helps to add colleges to wish-list so that it is easier to review shortlisted colleges. The user has to login their profile in order to store the colleges to the wish-list.

C. Algorithms for Proposed System

1) Searching Algorithm:

The one thing that really annoys us when surfing around sites is searching on sites that give you irrelevant results. You can't help but think something has gone terribly wrong with the search script. With over 70% of users likely to just 'bounce' off such a site after not being able to find what they were after immediately, we need to take a look into the why's of getting such irrelevant search results. The most basic search practice out there is to do something like:

- Break string into words.
- Compose the search query targeting known data fields like title, description, features, word by word, imploding into the query. At this point the where statement can look like 'where (description like '%black%' or features like '%black%' or title like '%black%') and (description like '%pack').
- Display the results and hope for the best.

It's fair to say, certain words should not be used to score results, they are just too generic to be considered. Unless you are typing something like 'the north face', 'the' should be dismissed, in the same way as 'this' should be removed.

So, what is the alternative? Oddly enough, the most accurate search results are achieved via manual tagging and backed up by product knowledge. It goes like that:

- Assign tags to each product. You can build an aliases table for common tags and errors.
- Build the search algorithm to break down the string into parts and analyse them. Drop all common words that won't help and keep the 'useful' bits only.

What words you have left, treat as tags and fetch all tuples they have been applied to.

Improved Querying tools that can be incorporated for search process -

- **Keywords**: Document creators (or trained indexers) are asked to supply a list of words that describe the subject of the text, including synonyms of words that describe this subject. Keywords improve recall, particularly if the keyword list includes a search word that is not in the document text.
- **Concept Search**: A search that is based on multi-word concepts, for example Compound term processing. This type of search is becoming popular in many e-Discovery solutions.
- **Proximity Search**: A phrase search matches only those documents that contain two or more words that are separated by a specified number of words; a search for "Wikipedia" WITHIN2 "free" would

retrieve only those documents in which the words "Wikipedia" and "free" occur within two words of each other.

- **Regular-Expression**: A regular expression employs a complex but powerful querying syntax that can be used to specify retrieval conditions with precision. Fuzzy search will search for document that match the given terms and some variation around them (using for instance edit distance to threshold the multiple variation).
- **Wildcard Search**: A search that substitutes one or more characters in a search query for a wildcard character such as an asterisk. For example, using the asterisk in a search query "s*n" will find "sin", "son", "sun", etc. in a text.

2) Sorting/Filtering Algorithm:

For Sorting/Filtering algorithms we can consider the shopping websites which provide sorting operations performed on products to show relevant results. On System websites, users can find colleges in two different ways: searching and browsing. Searching obviously means using the site search whilst browsing involves drilling down through the catalogue provided by the website.

Getting sorting and filtering right improves searching ability and allows users to find the product they want in less time, from this product listing. If users can't find the exact product they require in the minimal time, there's a good chance they'll go to another site where they can.

What is sorting and filtering?

Sorting is a method of changing the order of any product listing, where by users can choose which criteria they want the colleges to be listed by. So, price-conscious web users may choose to list the products in order of tuition fees, from cheapest to most expensive. Filtering is a way of reducing the number of products in a product listing. Users choose which criteria are important to them and view only relevant colleges. For example, price-conscious users may choose to view only colleges having tuition fees under 1 Lac rupees (thereby filtering out all products over 1 Lac rupees).

3) Recommendation Algorithm:

For getting accurate and relevant colleges as per users wish there is a need of Recommendation Algorithm to make the user experience much effective. The Recommendation algorithm helps to shortlist the colleges based on individual users demand and thus provide refined results which really matter to users. This algorithm basically works on the input data provided by the users after logging in their account and completing their profile.

The colleges are recommended based on the details as given below:

- Cut-off Marks(10th,12th,Grad Degree)
- Location
- Preferred degree(MBA,MTECH)
- Exam cut-offs (GATE, CAT, etc.)

To achieve faster query results the ranking of cut-off marks is carried out. Like assigning ranks to marks that fall in a certain range. For example:

Let's consider the marks range being assigned rank.

Less than 50%	> Rank 1
50%-60%	> Rank 2
60%-70%	> Rank 3
70%-85%	> Rank 4
Above 85%	> Rank 5

Now the college database values will be assigned these ranks corresponding to the cut off marks they require. Now when user selects the range in which their marks fall, the corresponding rank will be assigned to the user and hence the user will be recommended with the colleges that have ranks equal or lower corresponding to the user specified rank in their profile.

After filling such details they can get an estimate of which colleges they can apply for to get admission. This really helps to truncate down the search and provide minimal but optimal results quickly.

IV.CONCLUSION

This project assists in automating the existing manual system. This is a paperless work which can be monitored and controlled remotely. It reduces the man power required and provides accurate information. All years together gathered information can be saved and can be accessed at any time. Therefore the data stored in the repository helps in taking decision by management. So it is better to have a Web Based system.

V. REFERENCES

[1] Phyo Thu Thu Khine,Htwe Pa Pa Win,Khin Nwe Ni Tun“Keyword searching and browsing system over relational databases”,Digital Information Management (ICDIM), 2011,pages 121-126.

[2] 12th International Conference on Database Systems for Advance Applications (DASFAA), 2007. J. Zhan and S. Wang, "ITREKS: Keyword Search over Relational Database by Indexing Tuple Relationship".

[3] C. D. Manning, P. Raghavan and H. Schütze, An Introduction to Information Retrieval, Cambridge University Press, 2009, pages 569.

[4] S. Agrawal, S. Chaudhuri, and G. Das, "DBXplorer: A System for Keyword-Based Search over Relational Databases", ICDE, 2002, pp. 5-16.