Application for Online and offline Data Browsing on Windows Platform

Akshay T N
Computer Science and Engineering Department
AMC Engineering College
Bangalore, India

Aparna K S
Computer Science and Engineering Department
AMC Engineering College
Bangalore, India

Anjali Chaudhary
Computer Science and Engineering Department
AMC Engineering College
Bangalore, India

Jayashubha J, AP
Computer Science and Engineering Department
AMC Engineering College
Bangalore, India

***Abstract*— Windows Presentation Foundation (or WPF) [1] is a graphical subsystem for rendering user interfaces in Windows-based applications by Microsoft. It is a powerful framework for building Windows applications. In this paper, we address two features which can be developed using such a domain: a data browser and offline access of data browsed on data browser. A user can search for a particular data and this is shown with the help of tabs where each tab of the data browser has links to different websites. The user hence does not have to search for the data on different browsers and sometimes miss out on the information. Also an offline feature is added that helps in viewing data offline by saving the data at regular intervals.**

Keywords—WPF, data browser, offline access.

I. INTRODUCTION

In the earlier GUI frameworks, [1] there was no real separation between how an application looks like and how it behaved. Both GUI and behaviour was created in the same language, e.g. C# or VB.Net which would require more effort from the developer to implement both UI and behaviour associated with it.

In WPF [1][5], UI elements are designed in XAML while behaviors can be implemented in procedural languages such C# and VB.Net. So it is very easy to separate behavior from the designer code. With XAML, the programmers can work in parallel with the designers. The separation between a GUI and its behavior can allow us to easily change the look of a control by using styles and templates.

Data binding [2] [3] is a mechanism in WPF applications that provides a simple and easy way for Windows Runtime apps to display and interact with data. In this mechanism, the management of data is entirely separated from the way data. It allows the flow of data between UI elements and data object on user interface. When a binding is established and the data or your business model changes, then it reflects the updates automatically to the UI elements and vice versa. It is also possible to bind, not to a standard data source, but to another element on the page.

The data browser [2] is a great tool for viewing and editing your data. A great amount of work can be done on a data browser such as viewing of data according to a particular data rules or according to a search by the user.

A data browser can also be provided with an offline access facility that would facilitate the use of the data that has been viewed by the user previously to be downloaded and this download can be used for offline viewing of data.

Our key commitments are as per the following:

1. We propose an interface through a standalone application wherein it consists of a window consisting of tabs with links to all the websites we want.

2. One the user enters the search, the data from different websites is fetched and loaded [4] onto those tabs that are present so that one does not have to navigate through web pages separately using a web browser

3. We have provided admin access so that the admin alone can change the data that is linked with the application.

4. We have provided an offline access facility wherein the content of that particular webpage gets downloaded automatically for offline usage.

II. RELATED WORK

[1] https://msdn.microsoft.com/en-gb/ - Visual Studio and WPF application building have been referred from the works done in this website. We have gained the knowledge regarding the WPF application domain and Visual Studio usage for the same using the above website. Visual Studio 2013 has been used for developing the above application and the information on how to use and implement the code part has been studied and referred to from the above website.

[2] https://www.codeprojects.com - We have referred the coding and the implementation parts of the offline module and the database module from the above website.

The reference as to how to add styles, how to build and connect the database through C# have been referred from here.

[3] https://www.stackoverflow.com - We have referred the coding and the implementation parts of the online module and the database module from the above website. The reference as to how to add styles, how to build and connect the database through C# have been referred from here.

[4] https://www.wikipedia.com - We have integrated the search database using the plant species names from Wikipedia so that a name search would do the job of searching the information of that particular page on Wikipedia inside the data browser of the application designed.

[5] Suresh Patil, Sameer Soni, Pranali Dhele and Dr B.B.Meshram, ‘‘Application Development using WPF’’, ISSN:2278-1323, International Journal of Advanced Research in Computer Engineering & Technology, vol 1, Issue 4, June 2012.

III. SYSTEM MODEL

The System model mainly consists of four parts – the interacting user, a database, a data browser window and offline access storage. The data flows from the search the user keys in to the database, then the matched data is searched in the data browser and loads on to all of the tabs which are linked to different websites. Once the loading is done, the data that is loaded is saved for offline access.

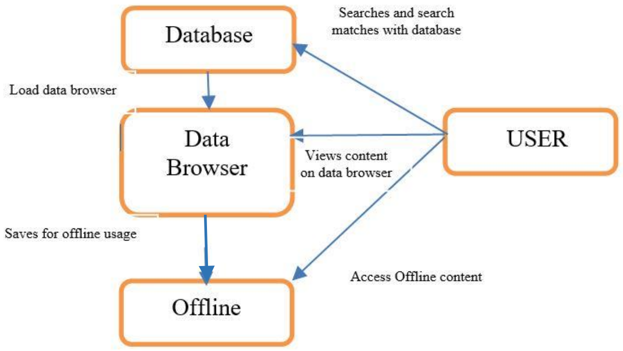


Fig.1 System Architecture

To lessen the burden on user, the data browser consists of tabs, each tab linked to different websites and which loads the search of the user on the websites. All the websites can be viewed at once in a single window without having to worry about the websites search manually. The loaded data is saved for offline purpose which can be viewed by the user later.

IV.IMPLEMENTATION

Implementation of the project:

- The design part of the data browser consists of a data browser which consists of separate tabs for each website that needs to be loaded. If there are 11 websites that can be linked, there would be 11 tabs respectively.
- It also consists of a search tab and a search button for facilitating the search of the item the user wants to search. This search bar is connected to the database for seeing if the data is present in the database.
- If matched, the item is searched for certain websites that need prior storage of names or elements for search. Else, the data can be searched online itself.
- So once the User initiates the search, the search is done and later part is to store the data in a file or folder for later usage.
- The files get downloaded automatically reducing the burden of the user having to download the page when he wants to and saves it as a file for offline reading.

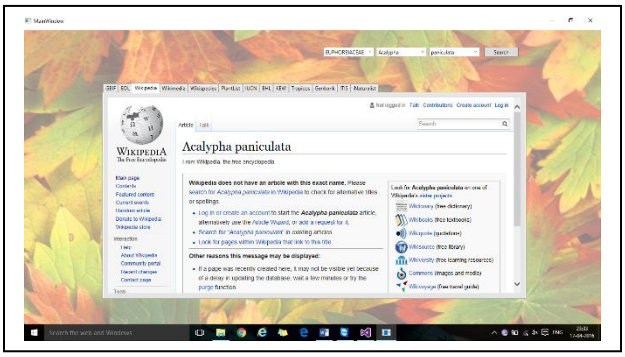
The implementation hence works in the above given way where the data is first searched, then the database is matched for the searched entry. Later, the tabs are loaded accordingly and while viewing data on the foreground, the details and content of that page displayed on the data browser is downloaded for offline usage.

V. RESULT

The working procedure is shown in the screenshot below. Once the search is done, the database is checked for the data correctness. Once the checking is done, the checked entry window is loaded and according to the code written using c# as the programming language. Once loaded, the content is downloaded for offline usage. The figure describes how the page looks with several tabs each linked to predefined websites and loads according to the search done. The below table specifies the number of tabs loaded when a particular search is done and also the range of relevant information loaded for each tab. For example, the Rose when searched is loaded in all the tabs with relevant information but at the same time, when Acacia Rugata is typed, it loads only on 9 out of 11 websites as the data regarding the particular content search is not present in some of the websites whereas the 9 websites have that information and they are loaded. The same applies to the other plant also.

TABLE 1: Results table

Sl.No	Content Searched	No.of tabs loaded with relevant information	Percentage range of relevance to the contents loaded (%)
1	Rose	11	100
2	Acacia Rugata	9	75-100
3	Acrocephalus Indicus	9	75-100



Acalypha paniculata

From Wikipedia, the free encyclopedia

Wikipedia does not have an article with this exact name. Please search for *Acalypha paniculata* in Wikipedia to check for alternative titles or spellings.

- Log in or create an account to start the ***Acalypha paniculata*** article, alternatively use the Article Wizard, or add a request for it.
- Search for "*Acalypha paniculata*" in existing articles
- Look for pages within Wikipedia that link to this title

Other reasons this message may be displayed:

- If a page was recently created here, it may not be visible yet because of a delay in updating the database; wait a few minutes or try the purge function.

Look for **Acalypha paniculata** on one of Wikipedia's sister projects:

Fig.1 Result page screenshot where the content is loaded in the specified webpage inside the application tab.

V1. CONCLUSION AND FUTURE WORK

Using this method, it is easier for the user to search for data as the data browser is designed in such a way wherein different websites are linked to each tab and much more websites can be linked in the same way. Also the saving of data for offline usage is done in a way wherein it downloads the whole website content and saves the whole website for navigating offline.

REFERENCES

[1] https://msdn.microsoft.com/en-gb/ - Visual Studio and WPF application building

[2] https://www.codeprojects.com - We have referred the coding and the implementation parts of the offline module and the database module

[3] https://www.stackoverflow.com - We have referred the coding and the implementation parts of the online module and the database module

[4] https://www.wikipedia.com - We have integrated the search database using the plant species names from Wikipedia

[5] Suresh Patil, Sameer Soni, Pranali Dhele and Dr B.B.Meshram, ‘‘Application Development using WPF’’, ISSN:2278-1323, International Journal of Advanced Research in Computer Engineering & Technology, vol 1, Issue 4, June 2012.

[6] Liu Xianhong, ‘‘Application of WPF to MIS development’’

[7] Yasushi Shinjo, Fei Guo, Naoya Kaneko, Takejiro Matsuyama, Tatsuya Taniuchi, Akira Sato, ‘‘A Distributed Web Browser as a Platform for Running Collaborative Applications’’

[8] B.S.Manjunath and W.Y.Ma, “Texture features for Browsing and Retrieval of Image Data”, IEEE transactions on pattern analysis nd machine intelligence, vol. 18, no.8, August 1996.

[9] Ann Chervenak, Ian Foster, Carl Kesselman, Charles Salisbury and Steven Tuecke, “The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets”, Journal of Network and Computer applications (2000) 23, 187-200.