INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

Make Search Easy

Anirudh Polawar
agpolawar1@gmail.com
Vishwakarma Institute of Technology, Pune, Maharashtra

S. V. Patil
saraswati.jadhav@vit.edu
Vishwakarma Institute of Technology, Pune, Maharashtra

Prajwal Kotalwar
prajwalkotalwar@gmail.com
Vishwakarma Institute of Technology, Pune, Maharashtra

Nikhil Ghodke
nikhil.ghodke17@vit.edu
Vishwakarma Institute of Technology, Pune, Maharashtra

Prathamesh Gindodia
prathamesh.gindodia17@vit.edu
Vishwakarma Institute of Technology, Pune, Maharashtra

ABSTRACT

There is a tremendous increase in the number of files and their transaction within the mobiles across the world. One has to report and maintain all the work through documentation. It gets difficult for the peoples to maintain and search through these files. This paper presents an approach to design a file manager integrated with the power of advanced search enabling the user to go through multiple files within the android through filtering them. Make Search Easy is a simple application like a file manager but has a powerful search and filtering option. The main feature which makes this application stand out in many other applications is its advanced search option. The user can search for a required letter, word, line or a paragraph within a document without actually knowing the location of a document within a directory or entire phone memory. Making it easy for the user to attend the search within the multiple files at the same time and through the easy way.

Keywords— *File manager, Advanced search, File filtering, Stack algorithm*

1. INTRODUCTION

File transaction is rapidly increasing among the mobiles making it laborious for the peoples to search through the multiple files. There are about millions of files on our mobile phone itself. A file manager is a mobile application, which is usually on both Android and IOS that provides a user interface to manage files and folders [8] [9]. A file, of whatever form it may be, is of vital importance to a user and so is the organization and management of files. As the number of files is quite large and continuously increasing, the files should be placed in such a way that it must be easily accessible as per the user's request. For this purpose, people had been using the traditional file management systems, where files are organized in a hierarchical structure. In a hierarchical file management system, files are stored under folder and subfolders of appropriate names. The 'make search easy' android application is basically an additional layer between the data stored on the device and the user to easily access the data. It's used to handle all the functions related to files from creation, storage, retrieval to deletion. With the help of this android application, we can do many of the operations which are to be done on files. The most common operations performed on files or groups of files include creating, opening (e.g. viewing, playing, editing or printing), renaming, moving or copying, deleting and searching for files. We can open a text, PDF or any other file for reading or writing. It can also be used to view images, videos, etc. With the help of our application which can also be called as a file manager, we can also listen to music and audios.

The most common problem user face during the interfacing with multiple files is searching the required file. As the number goes on increasing there is an increase in burdensome for the search. It is not possible for the user to search using the name as the key for the searching of the file. The naming of the file most of the times become hard to remember for the user. Due to the hierarchical structure of the location of the files, it also becomes difficult to keep track of the location of the files for the search to perform. Going through the individual file and search for the required information takes much time for the user and it is the only option left with the user by considering the above consequences. The file manager can be equipped with advanced search and filtering option making it easier for the user to search within the multiple files along with their contents. Hierarchical structure limits the user to access the same types of files in one place. The filter is designed and integrated with the application to remove this problem.

2. PROPOSED APPLICATION

2.1 Objective

People are facing problems to searching for the required information or file in the mobile or computer due to increase in a number of the file. In this proposed application we are going

© 2019, www.IJARIIT.com All Rights Reserved

to develop an application which will help the user for the search. This application is useful to reduce the laborious work of the user.

2.2 Features

- Smart filter for the files to improve organization hierarchical structure.
- This application uses the structure of a traditional file manager for the organization of the files.
- Information extracted by the application is arranged on the basis of percentage match and priority of the information location.
- This application is able to handle the search among the huge number of files efficiently.

3. WORKING AND DESIGN

During the first launch of the application, it will traverse through all the directories present in the phone and collect the different files along with their path and store it in different compartments and list them as per the user demands [7]. This will provide the filtered file of different types. The user can access the same type of files at one place which will remove the problem related to the hierarchical structure of the file organization. The different options are provided to the user in the form of a list view for the selection of type.

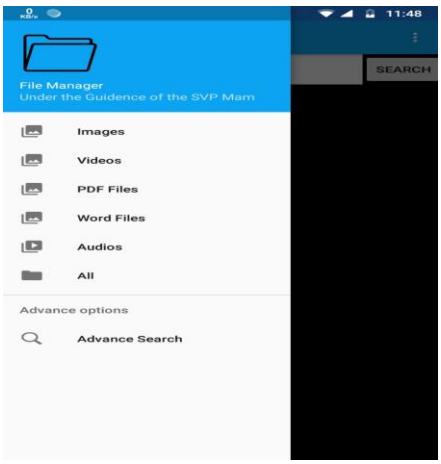

Fig. 1: Working and design

Once the user selects the option, the app enables the user to perform the operation on the same type of all the files present inside the phone memory. Traversing of the files on the first launch will increase the efficiency of the application and performs the task in less amount of time [5] [6]. The application keeps the list of all the path which will increase the speed to perform the operation on the multiple files.

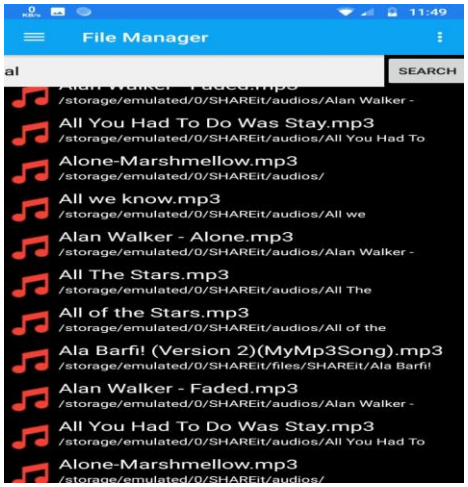

Fig. 2: Working and design

When the application is first launched, the root directory is selected as the current directory and all the files are listed in the root directory. The default root directory is the absolute path of the external directory and list view of the subdirectory. Different files and directories are displayed with different logos so that it makes easy for the user to differentiate between the files.

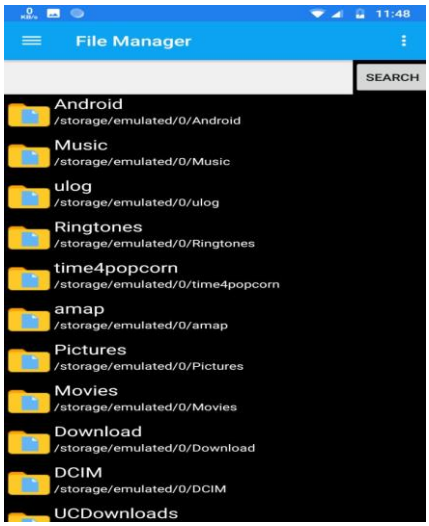

Fig. 3: Working and design

Advance search option switches the application to advance search mode where a user can search in many files at the same time. It will take require input from the edit text and search the entered data within all the files and display the output as the list view with the number of the character match in the file name.

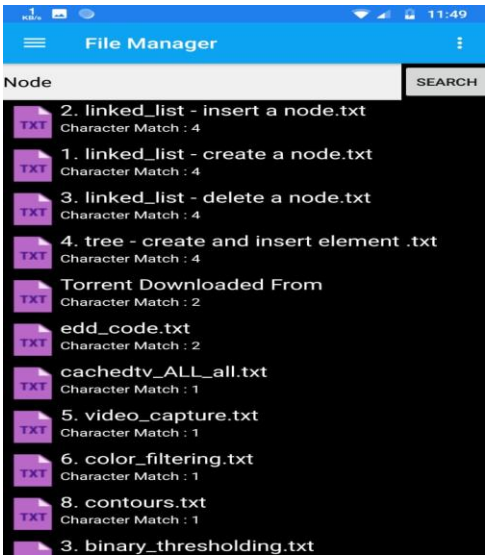

Fig. 4: Working and design

The application selects the current directory as the root and searches for it within the selected root directory. This will avoid applying the search in an unwanted directory and saves time. It will do the push and pop operations to the directory, then traverse through it and search the input within the stack directory [4]. The output is given such that user can tally the input with each file. This will enable the user to search within the multiple files at the same time which save time and efforts. A user can use the file contents and can manipulate them as per their own needs.

4. METHODOLOGY

4.1 File Arrangement and Filtering

List view is provided for the selection of the different types of the file. The navigation bar along with drawable bar is designed

© 2019, www.IJARIIT.com All Rights Reserved

which has filtered files arranged in sorted order is provided to the user. Filtering of this file is done using the push and pop algorithm which is used in the stack at the first launch of the app [1].

4.2 Advance search

Advance search is kept optional for the user. The user can activate it manually by the option provided. In advance, option mode current directory is selected as the root directory and software traverse through all the subdirectories. Push and pop algorithm is used for traversing all the subdirectories [2] [3]. Software visits the current directory and pushes all the subdirectories within it. Once the operation is completed it pop from the stack and replace the current directory with the stack top. This algorithm covers all the subdirectories and software will traverse through all the files. Operation of search is applied to the name of the file as well as contained within the file which will enable the software to extract all the required information.

5. ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our teachers, as well as our college VIT Pune who provided us with the opportunity to work on this project which also helps us do a lot of research, we are really thankful to them. Secondly, we would also like to thanks our colleagues and friends who help us a lot finishing this project within the limited time frame.

- This research was supported/partially supported by our Institute and our guides. We are thankful to our colleagues who provided the expertise that greatly assisted the research, although they may not agree with all of the interpretations provided in this paper.
- We are also grateful to our institute respected teachers for assistance with their knowledge, and SV Patil who moderated this paper and in that line improved the manuscript significantly.
- We have to express our appreciation to other non-teaching staff for sharing their pearls of wisdom with us during the course of this research.

6. REFERENCES

[1] Y. Peng and Z. Hao, "FA-Stack: A Fast Array-Based Stack with Wait-Free Progress Guarantee," in IEEE Transactions on Parallel and Distributed Systems, vol. 29, no. 4, pp. 843-857, 1 April 2018. doi: 10.1109/TPDS.2017.2770121

[2] L. Santiago, L. A. J. Marzulo, B. F. Goldstein, T. A. O. Alves and F. M. G. França, "Stack-Tagged Dataflow," 2014 International Symposium on Computer Architecture and High-Performance Computing Workshop, Paris, 2014, pp. 78-83. doi: 10.1109/SBAC-PADW.2014.21

[3] M. Stadler, M. Thalmann, T. Rower, N. Felber and W. Fichtner, "An embedded stack microprocessor for SDH telecommunication applications," Proceedings of the IEEE 1998 Custom Integrated Circuits Conference (Cat. No.98CH36143), Santa Clara, CA, USA, 1998, pp. 17-20. doi: 10.1109/CICC.1998.694899

[4] M. Hague, A. S. Murawski, C. -. L. Ong and O. Serre, "Collapsible Pushdown Automata and Recursion Schemes," 2008 23rd Annual IEEE Symposium on Logic in Computer Science, Pittsburgh, PA, 2008, pp. 452-461. doi: 10.1109/LICS.2008.34

[5] E. H. -. Sha, Y. Jia, X. Chen, Q. Zhuge, W. Jiang and J. Qin, "The design and implementation of an efficient user-space in-memory file system," 2016 5th Non-Volatile Memory Systems and Applications Symposium (NVMSA), Daegu, 2016, pp. 1-6. doi: 10.1109/NVMSA.2016.7547176

[6] E. H. -. Sha, X. Chen, Q. Zhuge, L. Shi and W. Jiang, "Designing an efficient persistent in-memory file system," 2015 IEEE Non-Volatile Memory System and Applications Symposium (NVMSA), Hong Kong, 2015, pp. 1-6. doi: 10.1109/NVMSA.2015.7304365

[7] V. Raveendran, M. Pauly, N. Paul, P. S and R. Varghese, "An approach to file manager design for speech interfaces," 2016 IEEE International Conference on Current Trends in Advanced Computing (ICCTAC), Bangalore, 2016, pp. 1-5. doi: 10.1109/ICCTAC.2016.7567335

[8] S. Kozawa, M. Niibori, Y. Ohtaki and M. Kamada, "File Manager that Shows off Notes Out of the Files," 2014 17th International Conference on Network-Based Information Systems, Salerno, 2014, pp. 611-614. doi: 10.1109/NBiS.2014.53

[9] C. E. Palazzi and M. Ferrarese, "FTP4Android: A local/remote file manager for Google Android platform," 2011 IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, 2011, pp. 373-377. doi: 10.1109/CCNC.2011.5766494.

© 2019, www.IJARIIT.com All Rights Reserved