



# INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X

Impact factor: 4.295

(Volume 4, Issue 3)

Available online at: [www.ijarit.com](http://www.ijarit.com)

## FTPAl: Natural language interface assistant for PC file and task management

T. R. Chamara Denipitiya

[it14073588@my.sliit.lk](mailto:it14073588@my.sliit.lk)

Sri Lanka Institute of Information Technology,  
Colombo, Sri Lanka

E. A. I Udayakantha

[e.a.i.udayakantha@gmail.com](mailto:e.a.i.udayakantha@gmail.com)

Sri Lanka Institute of Information Technology,  
Colombo, Sri Lanka

### ABSTRACT

*FTPAl is a natural language interface PC application that can be used to do certain file and task management related computer operations efficiently and conveniently. It is possible to perform these operations by using scripting languages such as Windows PowerShell, but it is a somewhat difficult task for ordinary computer users, if it is possible to perform these computer operations by using simple English language commands it would solve most of these issues. Therefore, FTPAl was developed using natural language techniques to address these issues and reduce the effort needed to write certain shell scripts. After commanding the FTPAl application in simple English, FTPAl is able to interpret and execute all the requested commands. FTPAl application is useful for businesses organizations to manage files and tasks efficiently which may lead to increase of revenue by reducing the time needed to do certain file management related tasks. As well as it is useful for people with less technical knowledge to perform file management tasks. It is possible to use this same concept to reduce the effort needed for other programming languages as well. The ultimate goal of this application is to provide an easy environment to perform file and task management related tasks efficiently and conveniently without using any programming languages.*

**Keywords:** *File manager assistant, File manager, Natural language interface file manager, Artificial intelligence, Natural language processing, File and task management, Windows PowerShell*

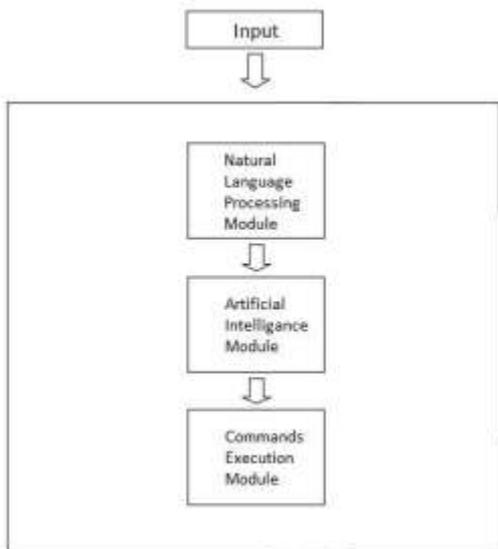
### 1. INTRODUCTION

File management related computer tasks are performed by various business organizations, businesses, and personal users every day. It is possible to categorize these operations as easy and hard operations. Some of these operations might need less technical knowledge while some file management operations need more technical knowledge. As well as some of these tasks are aren't even possible to perform without

having the proper training of languages, tools and as well as it is time-consuming. It is possible to perform some advanced files and tasks management related tasks using scripting languages such as Windows PowerShell, but it is a somewhat hard task for normal computer users to perform if it is possible to do these same tasks by using simple English language commands it would solve most of these issues. Therefore, FTPAl was developed using natural language techniques to address these issues and reduce the effort needed to write certain shell scripts. FTPAl natural language processing functionality makes it easy to use FTPAl compared to other file and task managers. Artificial Intelligence and natural language processing techniques were used to develop FTPAl assistant application.

Assisting users with file managing tasks is another main objective of FTPAl application. There are many assistant applications to perform various kinds of tasks. Applications such as Apple Siri, Google assistant, Cortana, Amazon Echo and ReQall are some of the most popular examples of personal assistant applications. These applications use natural language processing to match text or voice. Assistant applications usually do tasks such as work management, information and electronic mail organization and calendar and task scheduling.

FTPAl act as a bridge to reduce the effort needed to perform file management related operations by transforming user queries into shell commands with the help of natural language processing tools and techniques. Most of the file management applications consist of graphical user interfaces. FTPAl mainly focused on interacting with users using both natural language interface and graphical user interface.



**Fig.1 overview**

As shown in figure 1, the system consists of a natural language processing and artificial intelligence module. Above diagram shows how queries are processed inside FTPal application. It is required to summarize input queries less than ten words before entering them into the system. After that Input queries are passed into the natural language processing module, all the queries are categorized as nouns/verbs and plural/singular terms. Then the gathered data is passed into an artificial intelligence module to identify what commands must be executed. After that process, all the required commands are identified, combined and sent to execution module. Ultimately execution module executes all the requested commands.

## 2. LITERATURE REVIEW

In the literature review, FTPal research team studied numerous published research papers and documents from various sources. The research team used that knowledge to gather information needed to develop FTPal application.

It can be said that in recent year's considerable amount of time and effort have been spent on developing intelligent systems. Most of the so-called applications are focused on assisting users with artificial intelligence. Applications such as Apple Siri, Google assistant, Cortana, Amazon Echo and ReQall are some of the most popular examples of personal assistant applications [1]. Among them, Apple Siri was the first modern virtual assistant installed on a mobile device [2]. Virtual assistant applications make work via text/voice or by taking/uploading photos. These applications use natural language processing to match text or voice. Assistant applications usually do tasks such as work management, information and electronic mail organization and calendar and task scheduling. As well as most of these applications are a result of researches in artificial intelligence, natural language processing and machine learning [3].

It is possible to develop an assistant application that can act as a remote chatbot with the help of artificial intelligent techniques. By using such systems, businesses can minimize the effort needed for customer care service waiting queue. In addition, such systems are very fast and efficient compared to human customer care officers. In the respective paper, they suggest a new language called CyberMate Scripting

Language (CSL), which can be used to model business information to the system easily [4].

## 3. METHODOLOGY

Prototyping software development life cycle model was used as the systems development method for this research project. There are many software development life cycle models which can be used as a system development method, therefore research team had to analyze each of these SDLC cycle models and choose a most suitable model for the research project. As a result, the research team identified and analyzed factors such as available time, the complexity of the research project and available team members and finally decided that the prototype SDLC is most suitable for this project.

### A. Planning

The planning phase is one of the most important parts in the SDLC. In this phase, research team discussed why the FTPal system should be developed. The problem of the project was identified and discussed it among the group members. Moreover, all of these steps were considered for the identification of the basic requirements needed for the project. Main issues that came into consideration were database issues, Interface integration issues, and Time management.

The feasibility analysis helps to identify risks associated with the project. Therefore, feasibility analysis was carried out to identify these risks. Technical feasibility analysis was carried out to determine all the technical related limitations. This gathered information was used to decide whether to proceed with the FTPal project or not. All of the team members were involved this risk identifying the process.

A Work Break down Structure was designed to identify and understand all the requirements that are needed. Since the project duration was one year, a Gantt chart was designed to schedule all the tasks associated with the project. With the help of the Gantt chart, the research team decided to remove and add some tasks. Even though there were some useful features that research team had identified earlier, those were removed in order to ensure that this research project is completed within one year period of time. Each member of the research project was assigned to work and develop at least one function of the system. All the Research team members were involved in the process of identifying weaknesses and strengths that our group members had. In addition to that, research team members decided to divide most of the functions based on the skills they possess. After considering all the things Mentioned earlier all the tasks were allocated among the research team members.

### B. Requirement Gathering and Analysis

After completing the project planning, the research group was entered the phase of requirement gathering and analysis. In this phase, requirements that are needed for the FTPal application project were clearly defined. In order to develop the system, the research group needed to understand all the processes related to file and task management activities. The primary method used for data gathering was a literature review. FTPal application literature review was created by gathering required information through related research papers. The research team analyzed past research work was done by the researchers for the literature review including a

set of research papers related to pc assistant software, natural language interface and artificial intelligence.

The secondary method used for data gathering was a Questionnaire. Since this FTPal application is a desktop application for Windows users, all Windows PC users were considered as the population for this project and Nonprobability sampling method was used to gather related data for FTPal application.

Following things were found by analyzing the data gathered. Majority of respondents were employed and only 7% percent of people were not employed. 75 percent respondents were male and only 25% percent were female. The mean duration of experience is 2.75 years with a standard deviation of 1.2. As well as most people have not installed a third party software to do file and task management related tasks. 12.5 people have installed such applications and 87.5 have not installed a third party software.

Once the requirement gathering is finalized System Requirement Specification document was created. Ultimately all the requirements were gathered and analyzed.

**C. Design**

FTPal application consists of a Natural Language Interface (NLI) for the user to interact with the application. Variety of natural language commands are available to perform file and task management related tasks. In the design phase research group mostly concerned about the design of the assistant application. Flows, Prototypes and UML schemas were created as a part of the design phase.

Most of the FTPal interfaces were designed considering natural language processing functionalities. When designing the project whole system was divided into two main interfaces as file management interface and task management interface. In order to separate commands from an interface to another, the idea of separating into two interfaces became more useful throughout the whole project. After that, all group members were decided to show the results in the section below to the NLP input command. It helped to understand the final outcome of the command given by the user. User-friendly designs were mostly used for all the project designs.

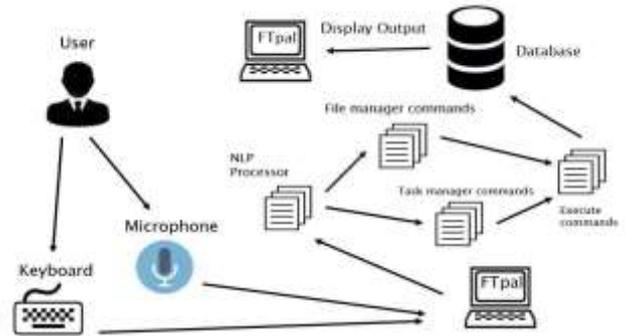
Finally, research group came up with the user interface shown below in figure 2.



**Fig. 2: Main interface**

When the application was developed research team considered things such as cohesion and coupling. The system was developed to produce the high cohesion and low coupling. Since research group used the prototype model for the development, the research team was able to make

changes easily and quickly to the system. The FTPal application was developed in Java language and used MySQL to design the database for the project. The main Idea behind the FTPal is to develop a way to convert user queries into shell commands, therefore ultimately a simple algorithm was designed for that purpose.



**Fig. 3: Architecture diagram**

Above diagram shows how main natural language processing and artificial intelligence modules are processed. After input commands are passed into the natural language processing module, they are categorized as nouns/verbs and plural/singular terms. After that process information is passed into an artificial intelligence module to identify what commands must be executed.

**D. Implementation**

After the design phase was completed actual implementation of the system was started. In this phase, group members divided work in order to start the implementation of the system. After dividing work, group members developed all components individually and after certain a time period, all of the components were merged into one project.

For the development, Java language was used by the group. In order to recognize natural language behaviors, NLP library called as Stanford Core NLP was used. After identifying file manager commands they are passed to commands execution module and executed using the Windows PowerShell.

The research team used POS tagging module of NLP library to develop the natural language functionality of FTPal application. A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads the text in some language and assigns parts of speech to each word (and another token), such as noun, verb, and adjective. [5].



**Fig. 4: POS tags**

Figure 4 shows how FTPal uses NLP tags in order to categorize words for the natural language module. Each of these words is assigned a POS tag value and returned by the natural language framework, FTPal identifies those tags and generates a relevant PowerShell command for the user input. FTPal first generates relevant shell commands for the verb terms and then those commands are combined with other relevant commands. Ultimately those generated commands are sent to the command execution module.

**Table 1: shows all tags that NLP framework supports and meaning of corresponding POS tag term.**

Tag	Description
IN	Preposition , subordinating conjunction
JJ	Adjective
NN	Noun, singular or mass
NNS	Noun, plural
NNP	Proper noun, singular
POS	Possessive ending
PRP	Personal pronoun
RB	Adverb
RBR	Adverb, comparative
RBS	Adverb, superlative
VB	Verb, base form
VBD	Verb, past tense
VBG	Verb, gerund or present participle
VBN	Verb, past participle
VBP	Verb, non-3rd person singular present
VBZ	Verb, 3rd person singular present

In addition to that, the application is capable of identifying a limited amount of commonly used similar words. For this purpose, a simple database table which consists of similar words was used. Whenever a user input is executed, FTPal application checks for the similar words. FTPal application is capable of converting user input queries into Windows

PowerShell commands. Therefore a simple algorithm was developed for that purpose. In order to develop the algorithm most of the PowerShell commands and user queries were analyzed by the research team.

At this stage, FTPal application supports only Windows operating system. Therefore, the research team studied the Windows environment before starting the actual implementation. The reason that FTPal application does not support operating systems other than Windows OS is that command execution module was developed using Windows PowerShell commands. Even though current implementation does not support other OS's it can be changed whenever needed simply by editing shell commands according to the OS of the respective user with the help of Java programming language.

**E. Testing**

After the system was developed, it is necessary to run tests to check whether the final output is being achieved and to make sure to provide the system with better quality. Software tests were carried out to reduce the number of defects. Prototype methodology was used for the development and it became useful when testing the system. Several types of testing were carried out by the research group as described below.

1) *Unit testing*

Unit testing was the testing of a single component to verify it whether it functions properly as expected. Research group tested all the units defined in the specification. Unit testing is necessary for the project to ensure all the units/components meet its defined requirements in the specification.

2) *Integration testing*

Integration testing mainly focused on testing whether components function together without any issues. Data exchange between modules is tested in this type of testing. User interface testing was performed along with user interface testing by the testing functionality of each interface and its relation to other interfaces. The research team used big bang integration test method as the testing method, the whole system was considered as one and tested the system.

3) *System testing*

System testing is used to check whether the system has met both functional and non-functional requirements specified in the specification. When the system testing was carried out system was tested for the objectives specified in the SDLC planning phase. Several non-functional requirements were also tested by the research group.

**4. RESULT and DISCUSSIONS**

**A. Research Findings**

This section describes the findings that research team gathered during the time of the research project.

The research team worked on analyzing user queries for the file manager module commands and found patterns that can to be transformed into shell commands. As well as the minimum amount of verb terms, noun terms needed for each query type to run successfully were identified. The research team came up with several simple algorithms to solve

problems for tasks such as finding the validity of a query, assigning each word a suitable shell command and grouping all generated commands together. In addition to that separate simple algorithm was developed to identify similar words for all user queries. This algorithm is capable of calling the same function if the user has entered the same query using different words.

## B. Evidence

Various type of testing was carried out by the research team in order to ensure that the system meets its specified requirements. These tests include unit testing, integration testing, and system testing. Unit testing was done separately for the project.

```
public int genQueryVerbPosition(String[] array){
    int i=0;
    boolean count = false;
    for(i = 0; i < array.length; i++) {
        if(array[i].equals("VB")) {
            count = true;
            i++;
            break;
        }
    }
    return i;
}
```

**Fig. 5: Verb position generation function**

```
public class executeCommands {
    public int exShellCommand(String shellCommand) throws
    IOException {

        String command = "powershell.exe "+shellCommand;
        Process powerShellProcess =
        Runtime.getRuntime().exec(command);
        powerShellProcess.getOutputStream().close();
        return 1;
    }
}
```

**Fig. 6. Command execution function**

## C. Discussion

In the initial stage of the research project, it became very hard to choose a suitable natural language processing library for the project. Moreover, the research team identified that some NLP libraries support only specific programming language such as python. NLP libraries such as Natural Language Toolkit, Stanford core NLP, and Apache Open NLP were identified as suitable natural language processing frameworks for the research. Since most of the research members are not familiar with Python, the research team decided to choose Stanford core NLP as NLP framework and Java as a programming language.

Most of the technical problems that were faced during the project occurred while integrating the natural language interface library with the system. NLP library had various unwanted built-in functionalities that were not required for the FTPal application. Therefore, all the unwanted functionalities were identified and removed by the research team. As a result, lite version of the library was used instead of using the full version of the NLP library. It helped research team to minimize the effort needed for the development and reduced the FTPal application load time.

After integrating the NLP functionalities, development of the command generation and execution module was started. In order to develop command generation module, the research team was required to develop set of functions that assign each user query term a suitable shell command. After analyzing user commands and shell commands carefully, required functions were developed by the research team to assign commands. Even though it became a hard task to develop the command generation module, research team came up with good solutions that solved above-mentioned challenges.

All of the FTPal file manager related commands are executed through Windows PowerShell. Therefore command execution module is another important module that operates as the backbone of FTPal. It was identified that some PowerShell commands take less time to execute while some commands take more time. Research team carefully analyzed those commands and selected commands that takes less time as suitable shell commands to include in the execution module. As well as it became a challenging task for the research team to capture the output of PowerShell and send it to the Java application. Therefore PowerShell CSV export option was used to capture the output and send it to the application.

## 5. CONCLUSION

FTPal is useful for any organization that needs efficient and convenient file management application. It is easy to use the FTPal application with the help of natural language interface. It can be used to minimize the effort needed for file management related tasks.

Moreover, it is possible to use these same techniques to transform user queries into different programming languages. It will help to minimize the effort needed to do programming related work. As well as currently, FTPal supports up to five words and only one sentence, therefore it is possible to increase the number of words, sentences that application supports and make it easier for users to use. Using these techniques it is possible to generate code for any applicable user query which can be used to do tasks in a less amount of time.

The research team believes that artificial intelligence, natural language processing related tools can be used to minimize the effort needed to do most daily tasks. The ultimate goal of the FTPal was to develop an application with an easy environment to perform files related tasks efficiently and conveniently through a natural language interface without using any programming languages.

## 6. REFERENCES

- [1] Jeff Dunn. (Jun. 2, 2017). It looks like Apple has some work to do if it wants Siri to be as smart as Google Assistant. [Online]. Available: [www.businessinsider.com/siri-vs-google-assistantcortana-alexa-2016-11](http://www.businessinsider.com/siri-vs-google-assistantcortana-alexa-2016-11)
- [2] Jeff Dunn. (6. 2, 2011). iPhone 4S hands-on. [Online]. Available: <https://www.engadget.com/2011/10/04/iphone-4s-hands-on/>
- [3] 25 Tasks You Can Outsource to a General Virtual Assistant. [Online]. Available: <http://www.chrisducker.com/25tasks/>

- [4] N. T. Weerawarna ; H. M. H. R. B. Haththella ; A. R. G. K. B. R. Ambadeniya ; L. H. S. S. Chandrasiri , CyberMate ~ Artificial Intelligent business help desk assistant with instance messaging services, 16-19 Aug. 2011.
- [5] Stanford NLP Group, Stanford Log-linear Part-Of-Speech Tagger. [Online]. Available. <https://nlp.stanford.edu/software/tagger.shtml>