

A Set of Predefined Command, Terminal User Interface for Linux Using Python

Deepshikha Sahu¹, Nilesh Thakur², Kirti Deshmukh³, Ayusha Nagpure⁴, Ashish Buwade⁵, Prof. Pankaj Singh Sisodiya⁶

¹⁻⁶Department of Computer Science and Engineering, Shri Balaji Institute of Technology and Management, Betul, M.P.

ABSTRACT: The research paper explores software (Terminal User Interface). The aim of this project development is to reduce the complexity of commands which are used for installation of various types of software such as HADOOP, AWS, LINUX, YUM and Many more and also it saves the time. The development of this software is based on high-level programming language python. As the name suggests Terminal User Interface this software provide a very friendly environment to the user. This software is very useful for those users who have the issues of memorizing the commands, in TUI we have merged the multiple commands for different software and make it more easy, reliable and

Flexible for the users. In this software users just have to enter his/her choice and than our system internally runs the commands and perform the task given by the user In this process all the commands for installation of such big program (software) are predefined, TUI is mainly designed to interact the user at the text user interface(Terminal).

The main objective of this software is to decrease the complicity of commands which are used to perform several tasks in Linux terminal.

Each terminal has its own set of features and this terminal user interface works great and convenient to work. There are many standard Linux command that are establish in that allow user to perform their task effortlessly. A CLI is used whenever a large vocabulary of commands or queries, coupled with a wide (or arbitrary) range of options, can be entered more rapidly as text than with a pure GUI. This is typically the case with operating system command shells. Command line interface are also used by systems with insufficient resources to support a graphical user interface. Some computer system languages such as Python, Rexx provide an interactive CLI mode to allow for rapid evaluation of code. Command line Interface are often used by programmers and system administrators, in engineering and scientific environment, and by technically advanced personal computer users. Command line Interfaces are also popular among people with visual disabilities since the command and response can be displayed using refreshable Braille displays.

TUI (Terminal User Interface) is the easiest way to avoid problems or issues which are arrive when user work on terminal for installing different software on their system.

To avoid this problem TUI (Terminal User Interface) is understandable for users. It perform task in convenient way.

The major purpose of TUI (Terminal User Interface) is to work in straightforward way by which user will familiarize them with the terminal as it is a standard way to interact with a Linux Server.

The user will not have to remember or to recall all the commands because the commands are already merged in TUI (Terminal User Interface). This type of interface is much faster than any other type of interface. It process the task of user very faster and in very shorten time period.

TUI (Terminal User Interface) intent to provide a friendly environment to Linux user.

Keywords: - PYTHON3, VISUALSTUDIO, Linux commands, Oracle Virtual Box, RedHat8

1. INTRODUCTION-

In today's world user's wants everything to be fast easy as well as secure and Terminal User interface provide all the functionality that user's want in today's era. In this software having a good and intelligently designed text user interface can make a difference on how much our program will be loved and used by its potential users, a lot of users have seen many examples of text user interfaces from a simple text editor, to other programs that use a text user interface to present the user with its functionality, we are using many of them as examples in our this project series, this system is developed by using LINUX(REDHAT), normally user found some difficulty when performing some task on Linux, but Linux is an open-source operating system it is more secure than other operating system in spite of that user doesn't prefer Linux hence TUI comes into the picture. TUI provide user friendly interface where a user can easily perform any task assume as they do on Google (Chrome). Basically if we want to install /run any program or software in our Linux operating system, we have to write proper command for it but for user it's quite difficult to remember all the commands for overcome this kind of problem we design Terminal User Interface. The user will operate TUI by just entering their requirement and our Terminal User Interface executes the specific commands internally. In this user doesn't need to remember commands, by just clicking on the commands system will start running the process of installation.

In computing, Terminal User Interfaces (TUI) (alternately text-based user interfaces, to reflect a dependence upon the properties of computer terminals and not just text). Is a retronym describing a type of user interface (UI) common as an early form of human-computer interaction, before the advent of graphical user interfaces (GUIs). Like GUIs, they may use the entire screen area and accept

mouse and other inputs. They may also use color and often structure the display using special graphical characters such as and, referred to in Unicode as the "box drawing" set. The modern context of use is usually a terminal emulator. In Unix-like operating systems, TUIs are often constructed using the terminal control library *curses*, or *ncurses*. (a mostly compatible library), or the alternative *S-Lang* library. The advent of the *curses* library with Berkeley Unix created a portable and stable API for which to write TUIs. The ability to talk to various text terminal types using the same interfaces led to more widespread use of "visual" Unix programs, which occupied the entire terminal screen instead of using a simple line interface. This can be seen in text editors such as vi, mail clients such as pine or mutt, system management tools such as SMIT, SAM, FreeBSD Sysinstall and web browsers such as lynx. Some applications, such as w3m, and older versions of pine and vi use the less-able term cap library, performing many of the functions associated with curses within the application. Custom TUI applications based on *widgets* can be easily developed using the dialog program (based on *ncurses*), or the Whiptail program (based on *S-Lang*). In addition, the rise in popularity of Linux brought many former DOS users to a Unix-like platform, which has fostered a DOS influence in many TUIs. The program *minicom*, for example, is modeled after the popular DOS program *Telnet*. Some other TUI programs, such as the *Twin* desktop, were ported over. Most Unix-like operating systems (Linux, FreeBSD etc.) support virtual consoles, typically accessed through a Ctrl-Alt-F key combination. For example, under Linux up to 64 consoles may be accessed (12 via function keys), each displaying in full-screen text mode. The free software program GNU Screen provides for managing multiple sessions inside a single TUI, and so can be thought of as being like a window manager for text-mode and command-line interfaces. A Command Line Interface (TUI) is also known as Command-line User Interface and/or Console User Interface (TUI). It is a type of human-computer interface (i.e., a way for humans to interact with digital computers or personal computer) that relies solely on textual request and response transaction process. It can be expressed as, TUI is a user interface to a computer's operating system or application software in which the user responds to a visual prompt by instructing with typing in a series of commands on a specified line and receives responses back from the system. The primary means of TUI was interaction with most computer systems until the introduction of the video display terminal in the mid-1960s, and continued to be used throughout the 1970s and 1980s on OpenVMS (Open Virtual Memory System) for Unix systems and personal computer systems including MS-DOS (Microsoft Disk Operating System), CP/M (Control Program for Microcomputers) and Apple DOS (Apple Disk Operating System). These user interfaces for handling operating system functionality is usually implemented with a command line shell, which is a program that accepts commands as text request and converts commands to appropriate operating system functions. A Command Line Interface is a powerful way of user interacting with an operating system. One of the first TUI was the MS-DOS. It was the operating system for the original personal digital computer (PC), which had been built in the 1980s. As in the early computers had one or two floppy disk drives and a monochromatic screen, and they were completely based on text. As the development of the hardware increased the performance and functionality of the PC, there was a urgent requirement to enhance the functionality of operating system. After introduced MS-DOS, Microsoft developed the first Windows operating system. This operating system developed by Microsoft provided a Graphical User Interface TUI with MS-DOS which had to be loaded first. Today's versions of Windows are operating systems in their own right but it is still possible to revert back to a Windows Shell to enter text-based commands to handle the operation of the computer.

1.1 Python

Python might not have bright years in the past years (which is mainly launch in the year 1991) but it has seen a continuous and amazing trend of growth in the 21st century. The growth rate of python was not something easy to achieve. Python programming language is better used for app development, web app or web development, database development, network development, game development, scientific computing, system administration, etc. There are several traits of this programming tool which has provided it with an incredibly successful journey. Python has a main role in the new-edge technologies of current and upcoming times like Data analysis, IOT, ML and AI. In TUI(Terminal User Interface) we have used Python for design as well as development. In TUI we mainly use OS module for performing operation which is given by the user for providing interaction between user and Operating System. Python Speech Recognition module for converting speech to text and text to speech

1.2 LINUX

This is a quite surprising aspect since we are actually using a non-Linux based system to write and read it. Hence, in the world of Microsoft, why are we even working or using Linux? The answer is simple. Linux has given us one of the major additions to the technology world in the form of the legitimized concept of open source and server that are helping companies to grow. With time, the UNIX fanatics are also bending toward the concept of the open source due to its victorious work. Linux has a very comprehensive and capable networking stack, supporting many protocols and features. Linux can be used both as an endpoint

node on a network, and also as a router, passing traffic between interfaces according to networking policies. Linux is not only good due to the impact that it has on industries but also on the background of the business. It has helped an individual to grow along with its company and managed to keep up with the changing environment. Especially when we are talking about entrepreneurs then the way industry was trying to work in have changed drastically. Along with this, it has managed to make the share reach skyrocket with the help of pressure generation, hosting industry, and even the platform that is competing with each other. This has to be a new attribute that is added to this industry due to the involvement of the latest advancement and helps in clicking up with the technology that is now taking over this world.

1.3 Command Line Interface

A command line interface processes commands to a computer program in the form of lines of text. The program which handles the interface called as command-line interpreter. OS implements command-line interface in a shell to make interactive access to operating system functions. In this type of access was primarily provided to users by computer terminals which starts in the mid of 1960s. In this generation, many users rely upon graphical user interfaces and menu-driven interactions. However, some programming and maintenance tasks may not have a graphical user interface and may still use a command line. The command line interface features a black box with white text. The user responds to a prompt in the command line interface by typing a command in shell. The output or response from the system can include a message, table, list, or some other confirmation of a system or application action.

2. OBJECTIVES

In this modern generation user are avoiding command line interface because of its non user friendly feature. But it is fast easy as well as secure and Terminal User interface provide all the functionality that user's want in today's era. This type of problem can be solved if we make CLI user friendly. So In this software user don't need to remember tough and long commands, but Linux is an open-source operating system it is more secure than other operating system in spite of that user doesn't prefer Linux hence TUI comes into the picture. TUI provide user friendly interface where a user can easily perform any task as same as they do on Google (Chrome). Basically if we want to install /run any program or software in our Linux operating system, we have to write proper command for it but for user it's quite difficult to remember all the commands for overcome this kind of problem we design Terminal User Interface. The user will operate TUI by just entering their requirement and our Terminal User Interface executes the specific commands internally.

3. TOOLS

The tools which we have used in our projects are jenkins, python, Docker.

3.1 Jenkins

Jenkins is free or we can also say that open source automation server. Which helps automate the parts of software development related to build, test and developing, facilitating continue integration and continuous delivery. This is server based system which is runs on servlet containers. Jenkins permit automating build and tests quickly deploying. Jenkins helps to automate the parts of software development related to building, testing and, continuous delivery. It is mainly a server based system that runs in servlet containers like Apache Tomcat. It also supports version control tools, including AccuRev, CVC and can execute Apache, Perforce, and RTC and can execute Apache Maven, Apache Ant and sbt based projects as well as arbitrary shell scripts and windows batch commands.

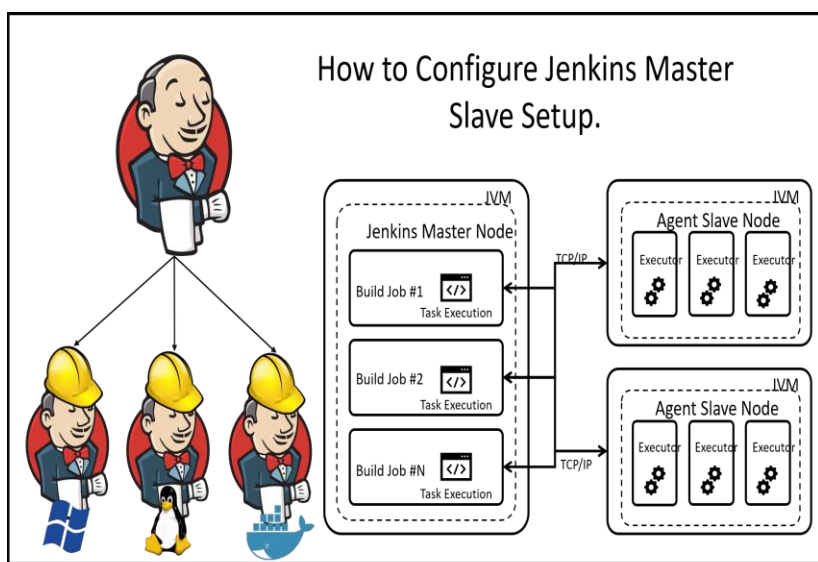


Fig.3.1 Jenkins

3.2 Python

Python is a high level programming language. Mainly Object oriented programming and structured programming are fully supported, and it also supports aspect oriented programming and functional programming. Many other factors are supported via extensions, programming logic and design by contract.

Python is also a dynamically typed language and a combination of reference counting and detects garbage collector for memory management. It also have a

feature of dynamic name resolution, which binds method and variable names during program execution.

Python design offers some support for functional programming in the Lisp tradition. It has reduce, map and filter functions; list comprehensions, tuple, frozen set, dictionaries, sets, and generator expression. The standard library has two modules which are itertools and functools, that implement functional tools borrowed from Haskell and standard Machine learning.

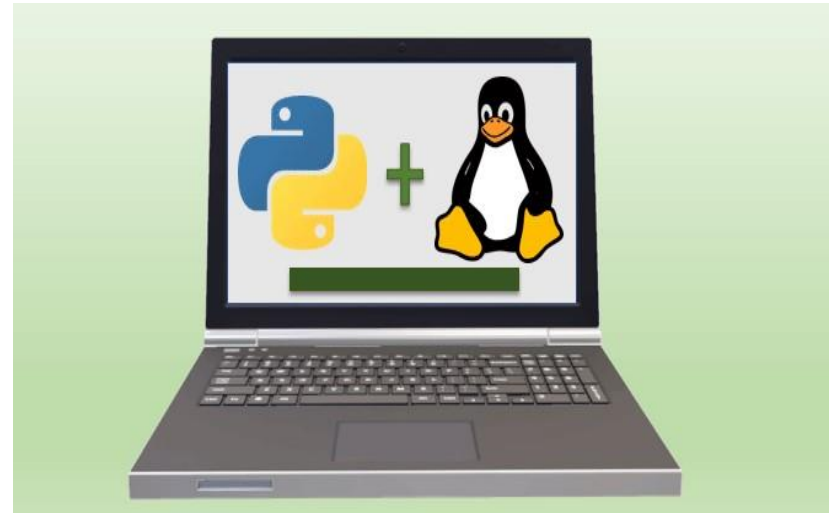


Fig 3.2 Python

3.3 Docker

Docker is an open source containerization technology which used to deploy the operating system within some seconds. Docker has its own commands. Docker is mainly use to deploy the web server, here we are using this containerization technology to deploy and test our environment. Overall the docker is as open source project that is used to automate the deployment of the software application which is running inside the containers, here containers provides the isolation for the application on the other hand we can say that it provide the isolated part for the application.

Docker has mainly two important tools which are docker. Docker compose is a tool for defining and running multiple container Docker applications. It uses YAML files to configure the application services and performs the creation and start-up process of all the containers with a single command. The docker composer command line interface allows user to run commands on multiple containers.

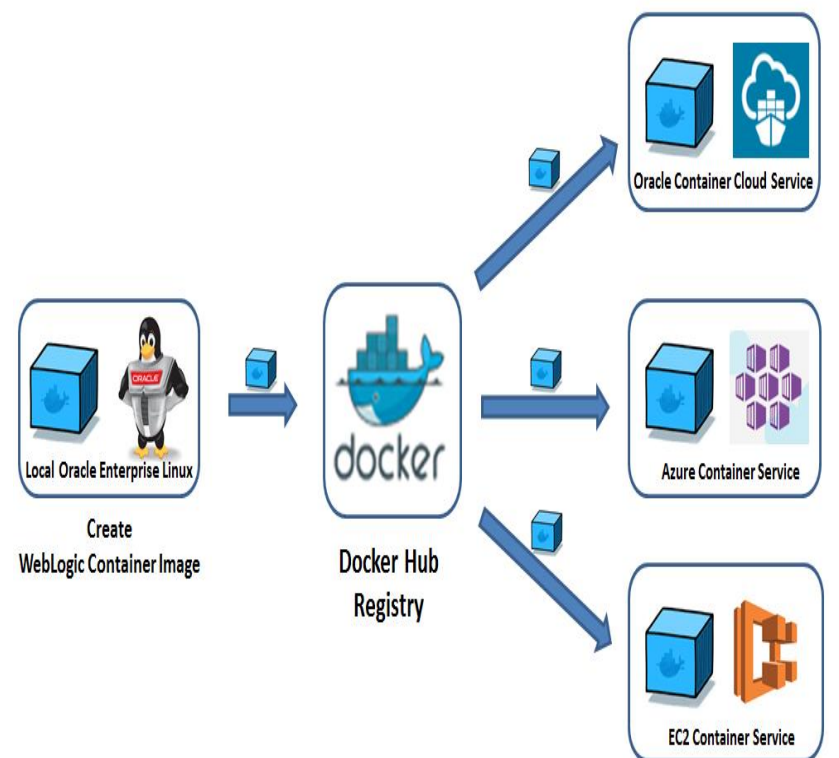


Fig 3.3 Docker

For ex. building images running containers that were stopped and many more. Commands related to image manipulation, or user interactive options are not relevant in Docker Compose because they address one container. The docker compose.yml file is used to define the services of application and also includes various configuration options. For example, the build option defines configuration options such as docker file path, the command option allows override to default docker commands. Docker swarm provide native clustering functionality for Docker containers, which turns group of docker engines into a single virtual Docker engine.

4. METHODOLOGY

Terminal User Interface developed by using Linux command in python program.

For developing Terminal user interface we are using REDHAT8 and PYTHON3 as a developing tool. While developing TUI

OS module of PYTHON3 and PYYTXS3 pays a very big and important role, more than 70% development of TUI is done by using above libraries now let's understands how TUI actually works.

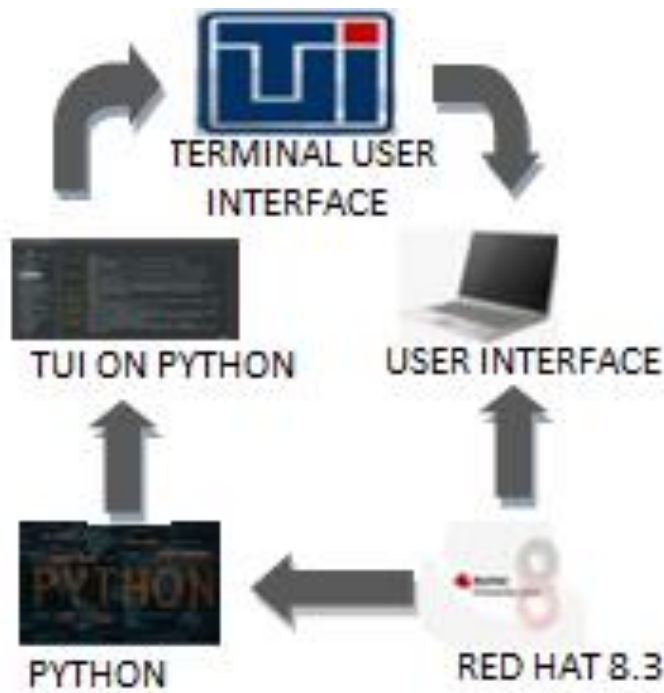


FIG 4. TUI

Firstly when a user open the TUI user see a search bar on the home page of the TUI.

In the search bar user type their query and the he/she only need to enter their choice for performing any task of installing any software, then our system internally check whether the command that the user entered is available or not . if the entered choice matched the command that are predefined than Terminal user interface prompt for asking that user want to install/execute the commands YES/NO if user click on the YES button than TUI execute the command and install the software , otherwise software will exit when user click on the NO button.

5. SYSTEM MODEL

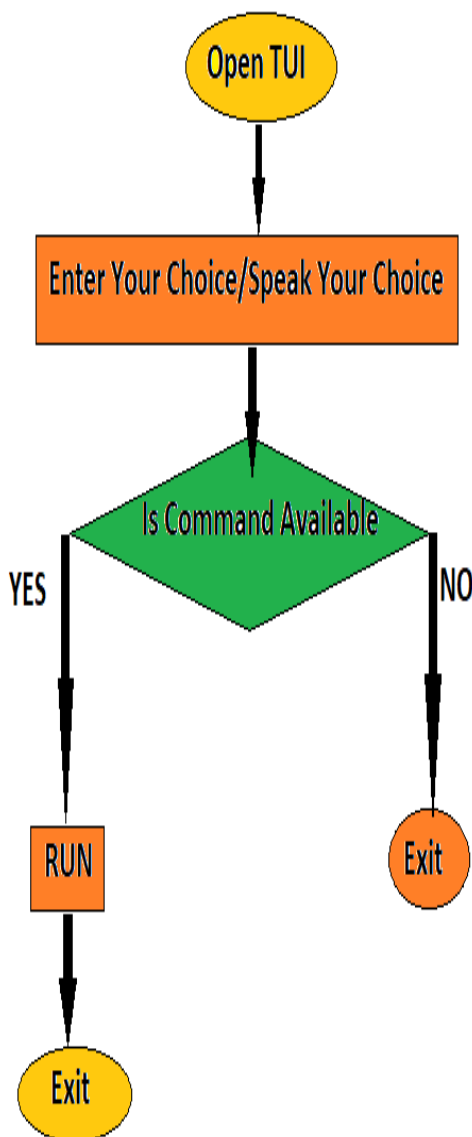


FIG 5.1 FLOW CHART

Above diagram (flow chart diagram) represents a workflow or process of our terminal user interface. A flow chart is simply a graphical representation of steps in sequential order and presenting flow of algorithm. The flow chart is showing the steps which the written in boxes of various kind, and their order by arrows.

When you just open terminal user interface you will see menu which contains many choices, according to which task you want to perform then it will checks your command is exist or not. If it is available then it will successfully run your task and then exit, if your command is not available then you are simply exit.

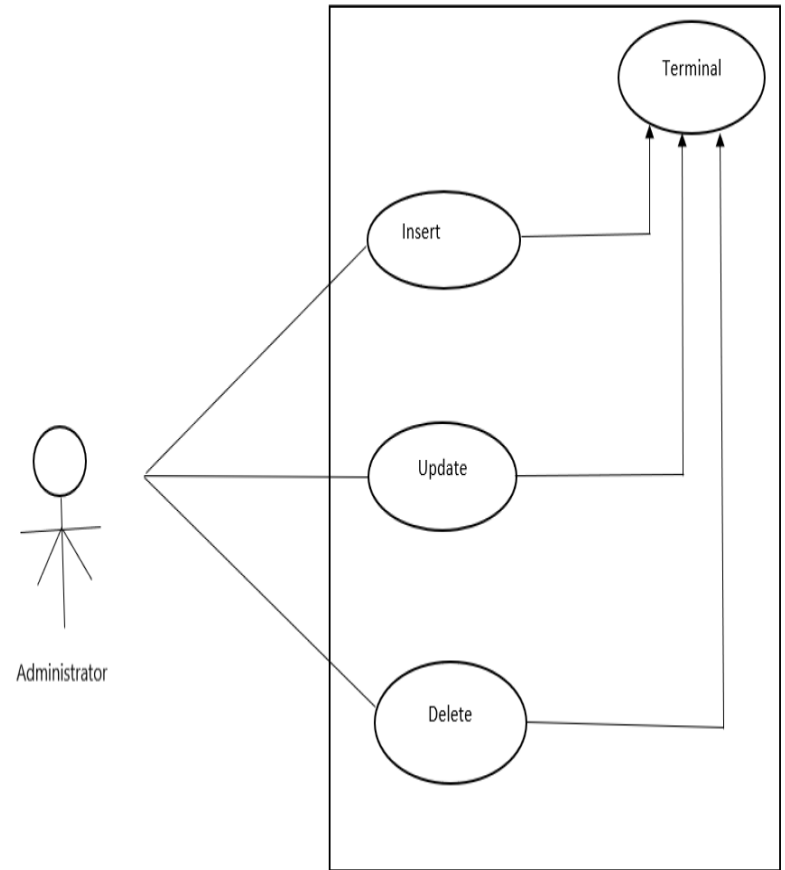
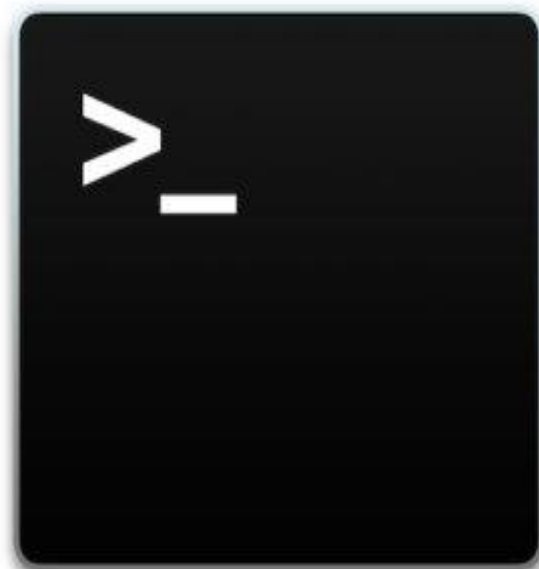


Fig 5.2 Use Case Diagram

6. Existing System

In existing system we use terminal in Linux , terminal is just like command prompt in windows a shell is a program that receives commands from the user and gives it to the OS to process, and it shows the output. Linux's shell is its main part. It is distros come in GUI (graphical user interface), but basically, Linux has a CLI (command line interface). The basic commands that we use in the shell of Linux.



7. FUTURE DEVELOPMENT

In future we can modify TUI as personal assistant just like OK GOOGLE, by adding voice based search for this we need when the program is run, the terminal will ask to user with the help of personal assistant with generating speech from the processed text output. After that user will give input by text or speech, which they want to execute. If the data is received to software in the form of human speech, the speech keywords are captured and analyzed then it converts into the text. This feature saves the time of user which is spending while typing the text .The conversion of Speech-to-Text and Text-to-Speeches performed by task manager. For this it uses natural language processing to match user text or voice input to executable commands then it identifies the

response to be taken against the command. Finally once the response is identified, output is generated by processing the data which is stored in software. Once the command executed then software again asking to user for input.

work fast with the developing technologies and bring a new sustaining change in the field of automation. Here paper pretence as an illustration for many advance application.

9. ACKNOWLEDGEMENT

I would like to extend my sincere thanks to our project coordinator Mr. Pankaj Singh Sisodiya for their guidance and supervision in completion of this project. We are extremely thankful to the Head of Department, prof. Vinay Sahu for their encouragement which helped us a lot.

We would also like to express our special thanks of gratitude to our principal Dr. Amit J. Modak for providing us the opportunity to work on this project. My thanks and appreciation also goes to my group members in developing this project.

10. LITERATURE SURVEY

In this research paper[1]Xerox introduce The Star user interface differs from that of other office computer systems by its emphasis on graphics, its adherence to a metaphor of a physical office, and its rigorous application of a small set of design principles. The graphic imagery reduces the amount of typing and remembering required operating the system. The office metaphor makes the system seem familiar and friendly; it reduces the alien feel that many computer systems have. The design principles unify the nearly two dozen functional areas of Star, increasing the coherence of the system and allowing user experience in one area to apply in others.

11. RESULT

Proposed system of TERMINAL is useful interface user's for understand terminal much more . It provide a very friendly and fast interface for operating Linux. Terminal User Interface make installation very simple and fast in Linux based system

12. LIMITATIONS

For someone who has never used a CLI, it can be very confusing. Commands have to be typed precisely. If there is a spelling mistake then the command will not respond or fail. If user can miss-type an instruction, it is often necessary to start from scratch again. There are a large number of commands which need to be learned-in the case of UNIX it can be more than hundred. Use can't just guess what the instruction might be and user can't just 'have a go'.

13. REFERENCES

- [1] Smith, D. C., E. F. Harslem, C. H. Irby, R. B.Kimball, and W. L.Verplank. "Designing theStar User Interface." *Byte*,April 1982.
- [2] Hung L-H,Kristiyanto D, Lee SB,Yeung KY (2016)GUIDOCK: UsingDocker Containers witha Common GraphicsUser Interface toAddress theReproducibility ofResearch. *PLoS ONE*11(4): e0152686.<https://doi.org/10.1371/journal.pone.0152686>
- [3] Ono K. VIZBI 2015Tutorial: Cytoscape,iPython, Docker, andreproducible workflow; Availablefrom: <https://github.com/idekerlab/cyREST/wiki/VIZBI-2015-Tutorial>.
- [4] Introducing Rocker: Docker for R;. Availablefrom: <http://dirk.eddelbuettel.com/blog/2014/10/23/>. Research ofmonitoring terminal oftrain based onembedded Linux andMiniGUI Liu Yongxin Tan Nanlin SuShuqiang (School ofMechanical, Electronicand Control Engineering,Beijing JiaotongUniversity, Beijing100044).
- [5] GitSurvey2012 - Git SCM Wiki. (2012). GitSurvey2012 - Git SCM Wiki. Available: <http://git.wiki.kernel.org/index.php/GitSurvey2012>. Last accessed 4th April 2015..
- [6] Morgan, K., Morris, R. & Gibbs, S. (1991). When does a Mouse become a Rat? or . . . Comparing Performance and Preferences in Direct Manipulation and Command Line Environment. *The Computer Journal*. 34 (3), p265-271.
- [7] Kelleher, J. (2014, 17-19 Jan). Employing git in the classroom. Paper presented at 2014 World Congress on Computer Applications and Information Systems (WCCAIS). Retrieved 4th April, 2015, at IEEE Xplore. DOI: 10.1109/WCCAIS.2014.6916568.
- [8] Treweek, P. (1996). Comparing Interfaces: Should We Assume that Ease of Use Influences Users' Preference?. Presented at Sixth Australian Conference on Computer-Human Interaction. Retrieved 4th April, 2015, at IEEE Xplore. DOI: 10.1109/OZCHI.1996.560004.
- [9] Perez De Rosso, S. & Jackson, D. (2013). What's wrong with git?: a conceptual design analysis. Presented at 2013 ACM international symposium on New ideas, new paradigms, and reflections on programming & software. Retrieved 4th April, 2015, at ACM Digital Library. DOI: 10.1145/2509578.2509584.
- [10] Eric S. Raymond. (2005). The Art of Unix Programming. Available: <http://www.catb.org/esr/writings/taoup/html/ch11s04.html>. Last accessed 6th April 2015.
- [11] Juergen Haas. Linux: GUI vs. Command line. Available: http://linux.about.com/cs/softofficeutility/a/gui_cli.htm. Last accessed 6th April 2015.

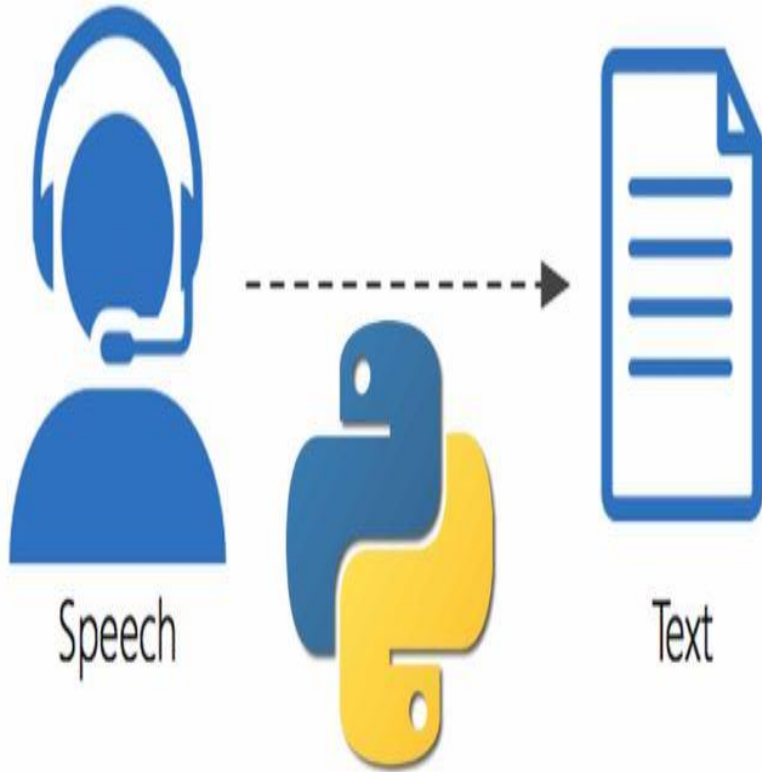


Fig4.Speech-Recognition

8. CONCLUSIONS

TUI (Terminal User Interface) is a type of user interface which is common as an earliest form of human and computer interaction that's must be like a text based user interface. In other words we may call it as a heteronym of a user interface. In Linux, user's face some difficulty while performing some task only because of their lengthy commands and today's era user want everything fast and easy. Here the idea of personal assistant plays a very important role. Personal assistant does things very fast and the use of personal assistant is very easy and comfortable even a child can also operate it. So by merging both of the things an idea of user friendly commands come into the picture hence a personal assistant for Linux is need to be develop. With the advancement in technology, where everyone is going for GUI because of user friendly environment. In any organization whether it be a business, corporate or any educational purposes, the people working in it or dealing with it generally, come up with issues like complicated language and limited range of software. Though they uninstalled because of their discomfort. This TUI is user friendly so user can use linux operating system. This software is useful to handles big command. Within specific time constraint it resolves problems of users.

The terminal user interface TUI (Terminal User Interface) accepts the user commands that the user's type or call it by using personal assistant and that command is proceed to the shell. The command user has typed proceed by the shell and that command is received and elucidate what the user typed or call into the instruction that can be executed by the operating system. Now the steps that user go through had concluded a result as an output is proceed by specific command then the output is displayed at the terminal. The terminal shows error if there should be some problem in the command then message is displayed at the terminal if there is no problem then the command runs or executes the program. If the user knows the correct commands then this type of interface can be much faster than any other type of interface. This type of interface needs much less memory (Random Access Memory) in order to use compared to other types of user interfaces. This type of interface does not use as much CPU processing time as others. A low resolution, cheaper monitor can be used with this type of interface. A TUI does not require Windows to run. Linux provides us a command line interface using terminal where we should have to learn basics of using commands to run or execute that should be burden of remembering commands for a user have load heavily of work. So in today's fast running life of new technologies every user need everything to be fast and that take less time including this they want more securable technologies . Linux is secure but taking the users efforts of remembering power for learning commands and user needs to type every command and that come as a drawback of Linux. To overcome with this drawback TUI (Terminal User Interface) works on user's demand.TUI (Terminal User Interface) uses text bases command as a user search for command to execute or install a or run a program they simply have to write the name of the program or the IDE name that they want to install the TUI (Terminal User Interface) gives various options of what they search and found be useful. To enhance this more in this there is a personal assistant work Personal Assistant will use the natural language processing.

Thus on the basis of the methodology we come to the conclusion that it can be designed to minimize the human efforts to interact with user's dependencies to

[12] ZeroTurnaround. "Developer Productivity Report 2012: Java Tools, Tech, Devs & Data". ZeroTurnaround. Boston, MA. 2012.

[13] Microsoft Corporation, "Survey Results: Open Source Developer Preferences (June 2011)," Microsoft Corporation, 11 July 2011. [Online]. Available: <http://blogs.msdn.com/b/codeplex/archive/2011/07/11/survey-results-open-source-developer-preferences-june-2011.aspx>. [Accessed 18th April 2015].

[14] Webopedia.com.(2015). Available: http://www.webopedia.com/TERM/G/Graphical_User_Interface_GUI.html. [Accessed 18th April 2015].

[15] Wikipedia. (Unknown). History of the graphical user interface. Available: http://en.wikipedia.org/wiki/History_of_the_graphical_user_interface#Apple_Lisa_and_Macintosh_28and_later_2C_the_Apple_IIgs.29. [Accessed 20th April 2015].