***“LIBRARY MANAGEMENT SYSTEM”***

A MINI- PROJECT REPORT ON

Submitted in partial fulfilment of the requirements

For the degree of

Bachelor of Engineering

In

Information Technology

by

**Pranali Waghmare Roll No.-18IT1011**

**Purva Tol Roll No.-18IT2017**

**Ritvika Sanap Roll No.-18IT1040**

Supervisor

**NilimaDongre**



DepartmentofInformation Technology

Dr.D.Y.PatilGroup’s

**RamraoAdik Institute of Technology**

Dr. D. Y. Patil Vidyanagar, Sector 7, Nerul, Navi Mumbai 400706.

(Affiliated to University of Mumbai)

( 2020)

# rait logo

RamraoAdik Institute of Technology

(Affiliated to the University of Mumbai)

Dr. D. Y. Patil Vidyanagar,Sector 7, Nerul, Navi Mumbai 400706.

CERTIFICATE

This is to certify that, Mini Project entitled

“Library Management System”

is a bonafide work done by

1.Ritvika Sanap

2.Purva Tol

3.Pranali Waghmare

and is submitted in the partial fulfilment of the requirement for the

degree of

Bachelor of Engineering

in

Information Technology

to the

University of Mumbai

Supervisor

Prof. Nilima M. Dongre

Project Guide Head of the department

Nilima Dongre Dr. Ashish Jadhav

Certificate of Approval by Examiners

This Mini Project report entitled “Library Management System” is a bonafide work done by Student Names under the supervision of Prof. Nilima Dongre approved for the award of Bachelor Degree in Information Technology, University of Mumbai.

Examiners:

1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . ..

2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . ..

Supervisors:

1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . ..

2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . ..

Principal:

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Date:

Place:

# **DECLARATION**

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

**Name and Roll No. of Students Signature**

1.Purva Tol-18IT2017

2.Ritvika Sanap-18IT1040

3.Pranali Waghmare-18IT1011

Date:

Place:

# ACKNOWLEDGEMENT

The project “Library Management System” is creative work of many minds. A proper synchronization between individual is must for any project to be completed successfully. One cannot imagine the power of the force that guides us all and neither can we succeed without acknowledging it.

We take this opportunity to express my profound gratitude and deep regards to our Guide **Nilima Dongre**, Department of the Information Technology Engineering for her or her exemplary guidance, monitoring and constant encouragement throughout the completion of this mini project.

We would like to express our gratitude to **Dr. Ashish Jadhav,** Head of the department, Information Technology Engineering for encouraging and inspiring us to carry out the project in the department lab. We take this privilege to express my sincere thanksare thankful to **Dr. Mukesh D. Patil, Principal RAIT,** for his constant support and motivation.

We also would like to thank all the staff members Department of the Information Technology Engineering for providing us with the required facilities and support towards the completion of the project.

Last but not the least we are thankful to our parents and friends for their constant Inspiration, encouragement and well wishes by which we have made a challenging project.

Ritvika Sanap (18IT1040) Signature

# **PREFACE**

We take great opportunity to present this Mini Project report on “**LIBRARY MANAGEMENT SYSTEM”** and put before readers some useful information regarding our project.

We have made sincere attempts and taken every care to present this matter in precise and compact form, the language being as simple as possible. We are sure that the information contained in this volume certainly prove useful for better insight in the scope and dimension of this project in its true perspective.

The task of the completion of the project though being difficult was made quite simple, interesting and successful due to deep involvement and complete dedication of our group members.

**TABLE OF CONTENTS**

**Declaration …………….…….………………………………………… I**

**Acknowledgement ……...……………………………………………… II**

**Preface …………….…………………………………………………… III**

**Table of Contents ……………………………………………………… IV**

**Table of figures ...…………………………………………………….… V**

**Abstract……....………………………………………………….……... VI**

**TABLE OF CONTENTS**

**Sr. No. Topic Page No.**

1. **INTRODUCTION………………………………………………………………..8**
   1. **INTRODUCTION TO SCRIPTING LANGUAGES………………………10**
   2. **WHY PARTICULAR SCRIPTING LANGUAGE………………………...11**
   3. **PROBLEM STATEMENT………………………………………………….12**
   4. **OBJECTIVES………………………………………………………………..13**
2. **LITERATURE SURVEY……………………………………………………….14**
   1. **MOTIVATION………………………………………………………………15**
3. **PROPOSED SYSTEM…………………………………………………………..16**
   1. **INTRODUCTION OF PROPOSED SYSTEM AND**

**ARCHITECTURE…………………………………………………………...16**

* 1. **HARDWARE AND SOFTWARE REQUIREMENTS……………………17**

1. **IMPLEMENTATION……………………………………………………………21**
   1. **SYSTEM BLOCK DIAGRAM……………….……………………………...21**
   2. **MODULE DESCRIPTION………………………………………………...22**
   3. **CODE………………………………………………………………………….26**
2. **RESULT…………………………………………………………………………...36**
   * 1. **OUTPUT SNAPSHOTS**
     2. **TESTING AND VALIDATION**
3. **CONCLUSION AND FUTURE SCOPE………………………………………..37**
   1. **CONCLUSION……………………………………………………………….37**
   2. **FUTURE SCOPE…………………………………………………………….37**
   3. **BENEFITS TO SOCIETY…………………………………………………..38**
4. **REFERENCES……………………………………………………………………47**

**ABSTRACT**

Library management system is a project which aims in developing a computerized system to maintain all the daily work of library. Manual process of keeping student records, book records, account details, managing employee is very difficult. There are various problems also faced by the student in library such as finding any particular book, information whether book is available or not, searching of books. To eliminate this manual system, library management system has been developed. The following mini-project is a system that allows user to enter, store and manipulate records and information of books. For any educational establishment, the need of an efficient library management system or database is necessary to store the records of books in the library. But at the same time, it becomes essential that the people involved with the use of this system can easily use this system. Keeping this in mind a user-friendly system that is easily accessible when it comes to entering, updating, searching as well as deleting any records or information of a book.

**CHAPTER -1**

**INTRODUCTION**

**INTRODUCTION**

* 1. **INTRODUCTION TO SCRIPTING LANGUAGES**

Usually shells are interactive that mean, they accept command as input from

users and execute them. However, some time we want to execute a bunch of

commands routinely, so we have type in all commands each time in terminal.

Shell scripts are similar to the batch file in MS-DOS. Each shell script is saved

with .shfile extension eg . myscript.shA shell script have syntax just like any other programming language. If you have any prior experience with any programming language like Python, C/C++etc. it would be very easy to get started with it.

* 1. **WHY PARTICULAR SCRIPTING LANGUAGE**
* There are many reasons to write shell scripts –
* To avoid repetitive work and automation
* System admins use shell scripting for routine backups
* System monitoring
* Adding new functionality to the shell etc.
  1. **PROBLEM STATEMENT**
* Prone to costly errors, a single mistake can change the command which

might be harmful

* Slow execution speed
* Design flaws within the language syntax or implementation
* Not well suited for large and complex task
  1. **OBJECTIVES**
* The command and syntax are exactly the same as those directly entered in
* command line, so programmer do not need to switch to entirely different
* syntax: Writing shell scripts are much quicker
* Quick start

**CHAPTER -2**

**LITERATURE SURVEY**

**2.1 MOTIVATION**

1.To store all the information in the database from where user will place their query and get the results on the basis of their query. Only valid users will be able to access this Library Management System.

2.Through this Library Management System, it will be easy to manage accounts and various details of records of book.

3. Library management system ensures easy access for the users for information

4. Manipulation of data and records of the books as per the requirements making the system more and more flexible that will increase the efficiency of the structure involved in managing the records of books.

5. Keeping the text involved clear and simple to overcome the language barrier for several employees

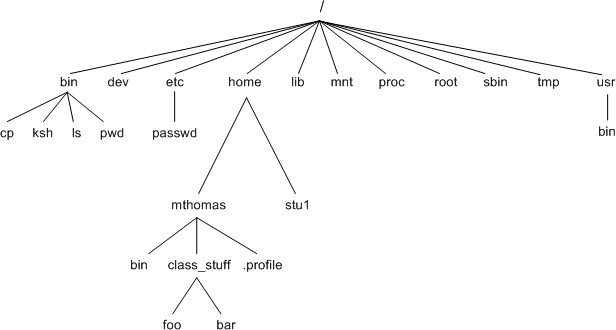
6.If any new parameter or field is to be added it can be done easily since the program has been developed with the help of bash shell scripting language known for being very easy to understand, modify and is very flexible

7.Library Management System replacing the traditional record storage and book keeping will also become economically more feasible

**CHAPTER -3**

**PROPOSED SYSTEM**

**3.1INTRODUCTION OF PROPOSED SYSTEM ANDARCHITECTURE**



The File Operations are performed over the ubuntu operating system and there are various types of directories present in this system.

The main components here are:

1. /boot : Contains the bootloader
2. /home : Contains the home directories of users.
3. /bin : All the executable binaries and commands used by all the users on the system are located here.
4. /sbin : This contains the system executable binaries typically used by system administrators.
5. /lib : Contains the system libraries that support the binaries in /bin and/sbin.
6. /etc : Contains the configuration files for network, boot-time, etc.
7. /dev : This has the device files i.e. usb, terminal device or any other device attached to system are shown here.
8. /proc : Contains information about the process running.
9. /tmp : This is the temporary directory where many processes create the temporary files . required. This is purged each time the machine is booted.

**3.2HARDWARE AND SOFTWAREREQUIREMENTS**

The following mini-project of Library Management system programmed using bash shell scripting language does not involve a conventional graphical user interface and thus it's specific requirements with their corresponding information is mentioned below

* Shell can be accessed by user using a command line interface. A special program called Terminal in linux/macOS or Command Prompt in Windows OS is provided to type in the human readable commands such as “cat”, “ls” etc. and then it is being execute. The result is then displayed on the terminal to the user. It will list all the files in current working directory in long listing format. Working with command line shell is bit difficult for the beginners because it’s hard to memorize so many commands. It is very powerful, it allows user to store commands in a file and execute them together. This way any repetitive task can be easily automated. These files are usually called batch files in Windows and Shell Scripts in Linux/macOS systems

⦁A command-line interface (CLI) is an operating system shell that uses alphanumeric characters typed on a keyboard to provide instructions and data to the operating system, interactively. For example, a teletypewriter can send codes representing keystrokes to a command interpreter program running on the computer; the command interpreter parses the sequence of keystrokes and responds with an error message if it cannot recognize the sequence of characters, or it may carry out some other program action such as loading an application program, listing files, logging in a user and many others. Operating systems such as UNIX have a large variety of shell programs with different commands, syntax and capabilities. Some operating systems had only a single style of command interface; commodity operating systems such as MS-DOS came with a standard command interface but third-party interfaces were also often available, providing additional features or functions such as menuing or remote program execution.

**CHAPTER –4**

**IMPLEMENTATION**

Start

Display menu

User Input

Quit

While choice is not f

Switch case

View books

Remove books

Edit books

Find books

Add new book record

Display menu

User input

**4.2 MODULEDESCRIPTION**

Because of easy-to-use approach a user can easily access the system. After the system code is executed using the ./<filename.sh> statement a 6-optioned menu will appear in the screen . User can select any option by entering the given number in the required field the 6 options of the menu are asfollows

"1. Add a new book record."

"2. Search for a book"

"3. Edit a book record."

"4. Remove a book record."

"5. View the entire record"

"6. Quit "

"Options"

the menu's options and the subsequent description one by one are given below

1. Add a new book record:

Under this option of the menu user, will be able to add a new book record to an existing system or can create a new system that never existed before by making a first entry this option makes use of the simple "read " command to accept the information of the book. The information contains book category, book title and author’s name.

1. Search for a book:

With the help of this option the user will be able to view information of a certain book as per the need. For this the user has to enter the title of the book to be searched (case sensitivity is not necessary), the information of the required book is then displayed. This option is selected by entering '2'. Here we make use of a simple command known as “grep” that will find the required book with the entered title.

1. Edit the record of a book:

With the help of this option of the menu the user can manipulate and make any changes to the data entered for any book. This is done by entering the title of the book whose data is to be updated. Then the system will display the information of the book that will help the user to check the data. This is done since the update function works such that the user has to enter the wrong data in the field asked and then has to enter the data with which it is to be replaced. This is done by using the “sed” command. If any record is not to be updated then the user can skip it by hitting the enter key

1. Remove a book record:

Through the use of this option the user will be able to delete information of the book one at a time, multiple decisions are excluded in this process. Keeping the case sensitivity in mind the system will accept the title of the book to be deleted from the user. After the user enters the title, its information will be deleted and a message that the information has been deleted will be displayed on the screen. Then the system will ask user to hit enter to return to the menu again or will ask if the user wants to quit, to hit the number in the menu for the quit command that is 6.

1. View all the records:

This option is used to display all the information that is stored at that particular moment in the system. For this we make the use of “cat” command and it will display the information of the books in the order in which they were entered the system. To view the records the user can choose that option by entering ‘5’

1. Quit:

This option is used by the user to quit from the given system. This option has been designed in such a way that it will clear the entire command line interface once this option is selected.

**4.3 CODE**

menu\_choice=""

record\_file="bookRecords.ldb"

temp\_file=/tmp/ldb.$$

touch $temp\_file; chmod 644 $temp\_file

trap 'rm -f $temp\_file' EXIT

get\_return(){

printf '\tPress return\n'

read x

return 0

}

get\_confirm(){

printf '\tAre you sure?\n'

while true

do

read x

case "$x" in

y|yes|Y|Yes|YES)

return 0;;

n|no|N|No|NO)

printf '\ncancelled\n'

return 1;;

\*) printf 'Please enter yes or no';;

esac

done

}

set\_menu\_choice(){

clear

printf 'Options:-'

printf '\n'

printf '\ta) Add new Books records\n'

printf '\tb) Find Books\n'

printf '\tc) Edit Books\n'

printf '\td) Remove Books\n'

printf '\te) View Books\n'

printf '\tf) Quit\n'

printf 'Please enter the choice then press return\n'

read menu\_choice

return

}

insert\_record(){

echo $\* >>$record\_file

return

}

#!!!!!!!!!...........................!!!!!!!!!!!!!!!!

#This function ask user for details information about book for keeping records

add\_books(){

#prompt for information

printf 'Enter Books category:-'

read tmp

liCatNum=${tmp%%,\*}

printf 'Enter Books title:-'

read tmp

liTitleNum=${tmp%%,\*}

printf 'Enter AutherName:-'

read tmp

liAutherNum=${tmp%%,\*}

#Check that they want to enter the information

printf 'About to add new entry\n'

printf "$liCatNum\t$liTitleNum\t$liAutherNum\n"

#If confirmed then append it to the record file

if get\_confirm; then

insert\_record $liCatNum,$liTitleNum,$liAutherNum

fi

return

}

find\_books(){

echo "Enter book title to find:"

read book2find

grep $book2find $record\_file> $temp\_file

# set $(wc -l $temp\_file)

# linesfound=$1

linesfound=`cat $temp\_file|wc -l`

case `echo $linesfound` in

0) echo "Sorry, nothing found"

get\_return

return 0

;;

\*) echo "Found the following"

cat $temp\_file

get\_return

return 0

esac

return

}

remove\_books() {

# $temp\_file

#set $(wc -l $temp\_file)

#linesfound=$1

linesfound=`cat $record\_file|wc -l`

case `echo $linesfound` in

0) echo "Sorry, nothing found\n"

get\_return

return 0

;;

\*) echo "Found the following\n"

cat $record\_file ;;

esac

printf "Type the books titel which you want to delete\n"

read searchstr

if [ "$searchstr" = "" ]; then

return 0

fi

grep -v "$searchstr" $record\_file> $temp\_file

mv $temp\_file $record\_file

printf "Book has been removed\n"

get\_return

return

}

view\_books(){

printf "List of books are\n"

cat $record\_file

get\_return

return

}

edit\_books(){

printf "list of books are\n"

cat $record\_file

printf "Type the tile of book you want to edit\n"

read searchstr

if [ "$searchstr" = "" ]; then

return 0

fi

grep -v "$searchstr" $record\_file> $temp\_file

mv $temp\_file $record\_file

printf "Enter the new record"

add\_books

}

rm -f $temp\_file

if [!-f $record\_file];then

touch $record\_file

fi

clear

printf '\n\n\n'

printf 'Mini library Management'

sleep 1

quit="n"

while [ "$quit" != "y" ];

do

#funtion call for choice

set\_menu\_choice

case "$menu\_choice" in

a) add\_books;;

b) find\_books;;

c) edit\_books;;

d) remove\_books;;

e) view\_books;;

f) quit=y;;

\*) printf "Sorry, choice not recognized";;

esac

done

# Tidy up and leave

rm -f $temp\_file

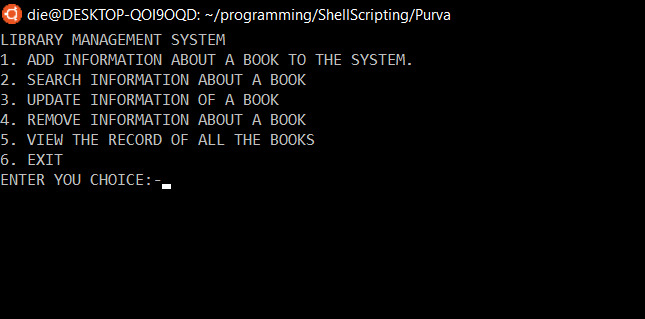
echo "Finished"

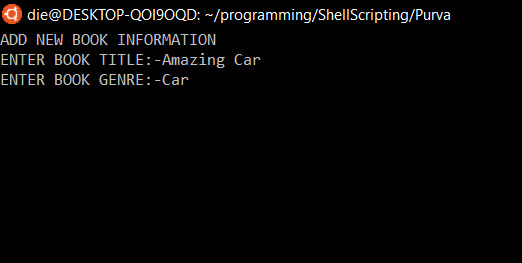
exit 0

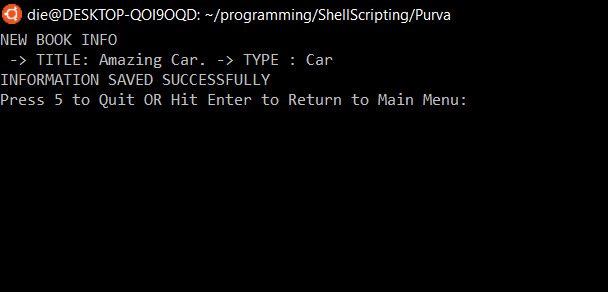
## CHAPTER –5

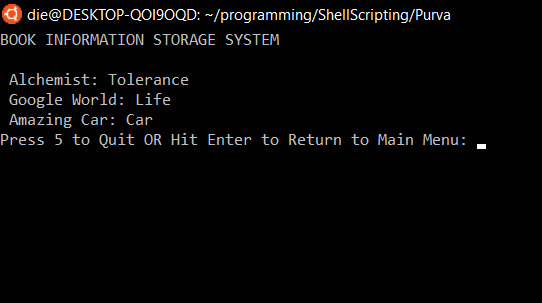
**RESULT**

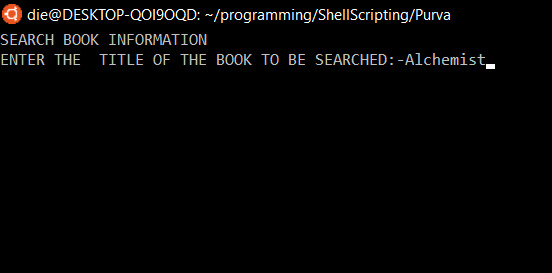
**5.1.1 OUTPUT SNAPSHOTS**

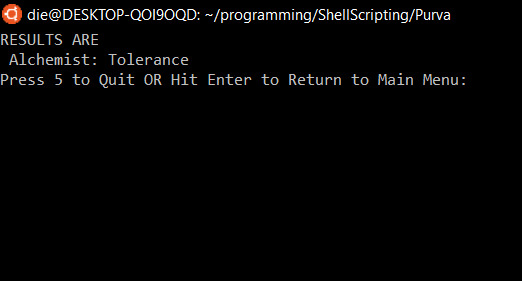


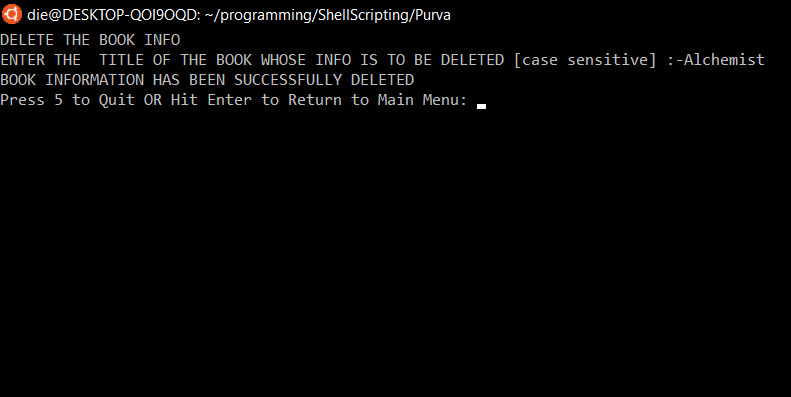


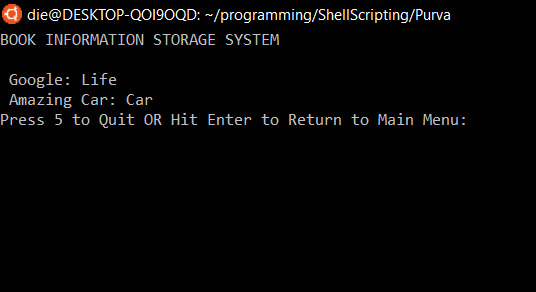


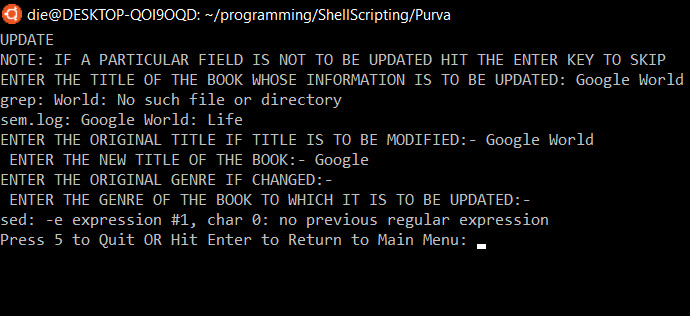












* 1. **Testing and validation**

##### Testing Details

Testing is a verification process for quality assessment and improvement. Testing is basically done to find errors, faults in the system. The basic goal of software development process is to produce the software that has very few or no errors

In an effort to detect errors soon after they are introduced each phase ends with verification activity such as reviews. However, most of these verification activities in the early phase of the software development are based on human evaluation and cannot detect all the errors. Testing plays an important role in quality assurance for the software. It is a dynamic method for the verification and validation, where the system to be tested is executed and the behavior of the system is observed.

##### Black Box Testing

Black Box Testing is also known as Behavioral Testing, is a software testing method in which the internal structure/ design/ implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional. This method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see. RamraoAdik Institute of Technology

##### White Box Testing

White Box Testing (also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing) is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. Programming know- how and the implementation knowledge is essential. White box testing is testing beyond the user interface and into the nitty-gritty of a system.

##### Unit Testing Unit

Testing is a level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of software. It usually has one or a few inputs and usually a single output. In procedural programming a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/super class, abstract class or derived/child class.

##### Integration Testing

Integration is the process by which components are aggregated to create larger components. Testing

the data flow or interface between two features is known as integration testing. Testing that occurs at lowest level is called unit/ module testing. As the units are tested and low-level bugs are found and fixed, they are integrated and integration testing is performed against groups of modules. Integration Testing

is also called as Structural Testing.

##### System Testing:

In system testing the behavior of whole system/product is tested as defined by the scope of the development project or product. It may include tests based on risks and/or requirement specifications, business process, use cases, or other high-level descriptions of system behavior, interactions with the operating systems, and system resources. System testing is most often the final test to verify that the system to be delivered meets the specification and its purpose. System testing is carried out by specialist’s testers or independent testers. System testing should investigate both functional and non- functional requirements of the testing.

##### Alpha Testing:

Alpha testing is one of the most common software testing strategy used in software development. It’s specially used by product development organizations. This test takes place at the developer’s site.

Developers observe the users and note problems. Alpha testing is typically performed by a group that is independent of the design team, but still within the company, e.g. in-house software test engineers, or software QA engineers. Alpha testing is final testing before the software is released to the general public. It has two phases:

1. In the first phase of alpha testing, the software is tested by in-house developers. They use either debugger software, or hardware-assisted debuggers. The goal is to catch bugs quickly.
2. In the second phase of alpha testing, the software is handed over to the software QA staff, for additional testing in an environment that is similar to the intended use

## CHAPTER –6

### CONCLUSION AND FUTURE SCOPE

###### CONCLUSION

In the end we would like to conclude that our aim was to make ‘Library Management System’ using bash script. The Library Management System can handle all the details about a library. The details may include genre, title, author’s name that would be beneficial for the officials involved in this record maintaining process making it more efficient and less time-consuming

###### FUTURESCOPE

After certain tweaks in the mini project it can be made more visually appealing at the same time it would have immense potential if taken online where the users will be able to print or store the required information on a device independent of the network

* 1. **REFERENCES**
     + [**www.google.com**](http://www.google.com/)
     + [**www.wikipedia.org**](http://www.wikipedia.org/)
     + [**www.tutorialpoint.com**](http://www.tutorialpoint.com/)
     + [**www.youtube.com**](http://www.youtube.com/)
     + [**www.academia.edu**](http://www.academia.edu/)