VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Machhe, Belagavi, Karnataka 590018

18ISL67 - FILE STRUCTURES LABORATORY WITH MINI-PROJECT REPORT

On

ONLINE COURSE MANAGEMENT SYSTEM USING B-TREES

Submitted in partial fulfillment of the requirement
for the award of the degree of

Bachelor of Engineering
in
Information Science & Engineering
by

Abhay M Kamat (1BG20IS001)
Ranjana BK (1BG20IS039)
Shweta K (1BG20IS054)

Vidyayāmruthamashnuthe

B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Information Science and Engineering
2022 – 2023

B.N.M. Institute of Technology

An Autonomous Institution under VTU

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the Mini-project entitled Online Course Management System is carried out by **Abhay Kamat** USN **1BG20IS001** the bonafide student of **B.N.M Institute of Technology** in partial fulfillment for the award of **Bachelor of Engineering** in **Information Science & Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2022-2023. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The mini-project report has been approved as it satisfies the academic requirements in respect of mini-project prescribed for the said Degree.

Mrs. Kavya N L
Asst. Professor, Dept. of ISE
BNMIT

Dr. S Srividhya
Prof & Head, Dept. of ISE
BNMIT

Name & Signature of the Examiners with date:

1.

2.

B.N.M. Institute of Technology

An Autonomous Institution under VTU

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the Mini-project entitled Online Course Management System is carried out by **Ranjana BK** USN **1BG20IS039** the bonafide student of **B.N.M Institute of Technology** in partial fulfillment for the award of **Bachelor of Engineering** in **Information Science & Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2022-2023. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The mini-project report has been approved as it satisfies the academic requirements in respect of mini-project prescribed for the said Degree.

Mrs. Kavya N L
Asst. Professor, Dept. of ISE
BNMIT

Dr. S Srividhya
Prof & Head, Dept. of ISE
BNMIT

Name & Signature of the Examiners with date:

1.

2.

B.N.M. Institute of Technology

An Autonomous Institution under VTU

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the Mini-project entitled Online Course Management System is carried out by **Shweta K** USN **1BG20IS054** the bonafide student of **B.N.M Institute of Technology** in partial fulfillment for the award of **Bachelor of Engineering** in **Information Science & Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2022-2023. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The mini-project report has been approved as it satisfies the academic requirements in respect of mini-project prescribed for the said Degree.

Mrs. Kavya N L
Asst. Professor, Dept. of ISE
BNMIT

Dr. S Srividhya
Prof & Head, Dept. of ISE
BNMIT

Name & Signature of the Examiners with date:

1.

2.

Table Of Contents

Table Of Figures

CHAPTER 1

INTRODUCTION

As a part of the File Structures laboratory mini-project, “Online Course Management” a generalized file structures application was developed, and using the functionality of the same web based application was designed to serve as a database of online courses and its details. The application displays the courses along with its description and price.

1.1 Objective

The objective of the project taken up was to generalize the file structure operations such as insert, search and delete, by abstracting away the core functionality of the same and to implement the optimization method of Indexed Sequential File Access on top of the same.

A web-based application online course database was designed, which serves as a database of course details. This application enables students and learners to access their desired course get its description very efficiently and conviniently.

1.2 Scope

The Online Course Management System that includes functionalities like displaying, searching, saving, deleting, and modifying course details has a wide scope of application. Here are some potential areas where this project can be applied:

- Educational Institutions: The system can be used by schools, colleges, universities, and other educational institutions to manage their course offerings. It allows administrators, faculty, and students to access and update course information easily.
- E-Learning Platforms: Online platforms that provide courses and training programs can utilize this system to organize and manage their course catalog. It enables instructors to add, modify, and delete courses, and learners to search for and enroll in courses.
- Corporate Training: Companies conducting internal training programs can benefit from the course management system. It helps HR departments or training managers to keep track of courses, monitor employee enrollments, and update course details as required.

1.3 Motivation

The motivation for developing a Course Management System with features like displaying, searching, saving, deleting, and modifying course details stems from the need to efficiently manage and organize educational courses in an organized manner. Here are a few key motivations for such a project:

- Easy Course Access: By implementing a Course Management System, users can easily access course information such as course name, description, and other relevant details. This makes it convenient for students, instructors, and administrators to quickly find and access the courses they are interested in or responsible for.
- Efficient Course Organization: With a centralized system for course management, it becomes easier to organize and categorize courses. This allows for better navigation and search capabilities, enabling users to locate courses based on different criteria such as course ID, name, or description.
- Streamlined Updates and Modifications: Having the ability to modify and update course details ensures that the information presented remains accurate and up to date. It allows for changes such as course name updates, description revisions, or even changes to course materials or prerequisites. This feature benefits both administrators and learners by providing them with the most relevant and current information.
- Improved User Experience: A well-designed Course Management System enhances the overall user experience. It simplifies the process of finding and enrolling in courses, allows for customization and personalization options, and provides a user-friendly interface for course management tasks. This, in turn, leads to higher user satisfaction and engagement.

Overall, a Course Management System that encompasses functionalities such as displaying, searching, saving, deleting, and modifying course details simplifies the process of managing courses, enhances user experience, and improves overall efficiency and effectiveness in the educational domain.

CHAPTER 2

METHODOLOGY

2.1 B-Tree

B-tree is a data structure used to store and manage large amounts of data efficiently. It is specifically designed to handle disk-based storage systems, where data needs to be read and written from and to the disk. The key idea behind a B-tree is to keep the data balanced and minimize the number of disk accesses required for operations like searching, insertion, and deletion.

In simple terms, a B-tree is like a tree with multiple levels, where each level can hold a certain number of keys and pointers to other nodes. The keys in the tree are arranged in a sorted order, allowing for efficient searching. The tree is balanced, meaning that all the leaf nodes are at the same level, which helps maintain a consistent performance for accessing data

2.2 Tools

Tkinter Module in Python

The Tkinter module is a standard Python library that provides a GUI (Graphical User Interface) toolkit for creating desktop applications. Tkinter is widely used because it is easy to learn and allows developers to build visually appealing and interactive applications with minimal code.

Some key features and functionalities of the Tkinter module include:

- Window Management: Tkinter allows the creation and management of windows and dialog boxes. Developers can create main application windows, pop-up windows, and various types of dialog boxes.

- Widgets: Tkinter provides a wide range of built-in widgets such as buttons, labels, entry fields, checkboxes, radio buttons, dropdown menus, listboxes, and more. These widgets can be placed in windows and used to interact with the user.

- Layout Management: Tkinter offers different layout managers (e.g., grid, pack, and place) to arrange and position widgets within windows. These layout managers help in achieving a desired user interface design.

- Event Handling: Tkinter supports event-driven programming, where actions by the user or system events (e.g., mouse clicks, keyboard input) trigger specific functions or actions. Developers can bind events to specific widgets and define corresponding event handlers.

- Styling and Customization: Tkinter allows customization of widgets' appearance, including changing colors, fonts, sizes, and styles. Developers can create custom widget styles to match the application's design requirements.

- Dialog Boxes and Message Boxes: Tkinter provides pre-built dialog boxes, such as file dialog, color picker dialog, and message boxes for displaying alerts, confirmations, or requesting user input.

- Menu and Toolbars: Tkinter supports the creation of menu bars, cascading menus, and toolbars for easy navigation and access to various application functionalities.

- Drawing and Graphics: Tkinter allows drawing shapes, lines, images, and text on a canvas widget. This feature is useful for creating visualizations, charts, and simple graphics.

- Integration with Other Libraries: Tkinter can be integrated with other Python libraries and frameworks, such as Matplotlib for advanced data visualization or database libraries for data management.

Overall, Tkinter is a versatile and beginner-friendly module for developing graphical applications in Python, making it a popular choice for building desktop GUIs with interactive features and visual appeal.

CHAPTER 3

SYSTEM REQUIREMENTS SPECIFICATION

3.1 User Requirements

The users of this application can be categorized as- application developer and end user.As an application developer, the user requirements are as follows :

1.Python 3.7

2.Visual Studio 2022

3.Tkinter Module GUI

For an end user the user requirements are as follows :

1. RAM > 4GB
2. Processor : Intel® CoreTM i5-6500 CPU @ 3.20GHz pr greater
3. System type : 64-bit operating system, x64-based processor
4. Modern web browser

3.2 Software requirements

The software for the development and deployment of the project includes-

1. Operating system: Windows XP/7/8/10

2. Browser: Chrome browser

3. Front-end: Tkinter GUI

4. Back-end: Python 3.7

3.3 Hardware requirements

The hardware for the development of the projects include-

1. Processor: Intel® core™ i5-6500 CPU @ 3.20GHz 3.20GHz
2. Installed memory (RAM): 8.00 GB (7.88 GB usable)
3. System type: 64-bit operating system, x64-based processor

3.4 Functional Requirements

Detailed description of Functional Requirements :

1. The system should implement the file structures operations of inserting, searching, traversing and deleting the records.
2. The system should implement Indexed Sequential File Access mechanism as the optimization mechanism for file structures operations.
3. The system should create files to store records in the persistent storage.
4. The system should enable indexing on the respective files and store the same in persistent memory as index files.
5. The system should listen to user requests for operations on file structure operations.
6. The system should have a User Interface to facilitate the user in communicating with the system

3.5 Non - Functional Requirements

Detailed description of Non - Functional Requirements :

1.The system should load meta data of the application on start up.

2.The system should intimate the user if the record he/she has searched for doesn’t exist.

3.The application should be user friendly.

4.The application shouldn’t crash.

5.The application should be platform independent.

CHAPTER 4

SYSTEM DESIGN AND DEVELOPMENT

The System Design Document describes the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces. Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

4.1 Architecture Diagram

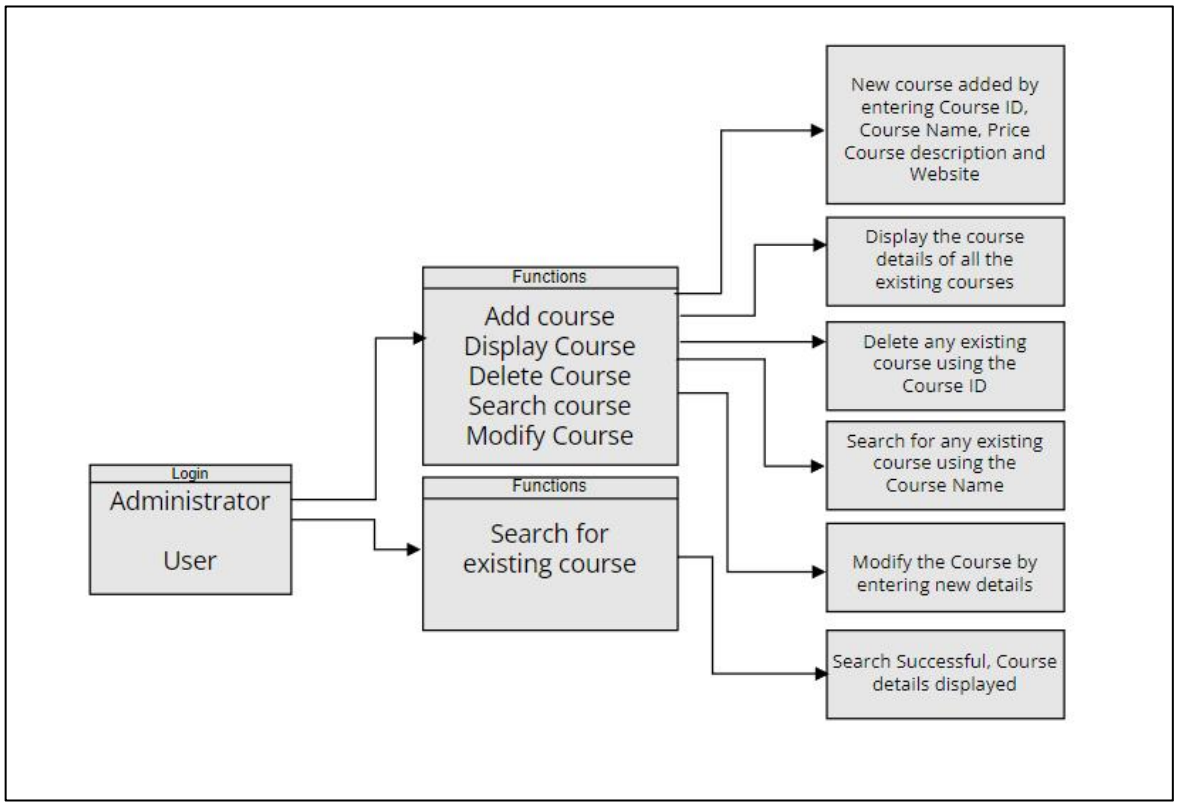


Fig 4.1 Architecture Diagram for Online Course Management System

The Figure 4.1 shows the flow of the mini project that the user will experience. System user has two options in the menu user and Admin. The user can search for the courses. The Admin has to login using right credentials. When the admin opts for add course option, the details are taken from the admin and insert function is implemented on it. The course details along with the primary key is entered into the data file which is entry sequenced. After which, the current record information is displayed on the GUI. Similarly, when user opts for search option the course name is taken from the admin and

search function is implemented on it. The name is looked up in the data file and the corresponding details from the data file is displayed. There the admin has the option to either delete the current record or modify the current record. If the admin opts for delete, the course details are deleted from the data file. In case the details are not present, the appropriate message is displayed. In case of unsuccessful search and modify, an appropriate message is displayed. Display Option prsent in the home screen allows the admin to view the details of all courses in the form of a table. The admin can also view the B-Tree traversal of the file using the Display B tree option present on the screen.

4.2 Design of the Fields and records

The system makes use of variable length record and fixed size fields. In variable length record, it will take the minimum space need for that field and at the end of the record there will be a delimiter placed indicating that it is the end of the record. The fields are separated using a delimiter ('|') whereas hash ('#') denotes end of a record.

A delimiter is a sequence of one or more characters used to specify the boundary between separate, independent regions in plain text or other data streams. In this application, there is 1 file that stores the data which is:

1. Data File

This file has respective attributes which identifies the uniqueness of each record. It consists of the all the course details that are made by the user while entering a record. Attributes which identifies the uniqueness of each record are Course ID, Name, Price and Description.

4.3 User Interface

Fig 4.2 Admin Home Interface

The above fig 4.2 shows the admin home interface.The user interface (UI), in the industrial design field of human–computer interaction, is the space where interactions between humans and machines occur. The goal of this interaction is to allow effective operation and control of the machine from the human end, whilst the machine simultaneously feeds back information that aids the operators' decision-making process.

A. Insertion of a record

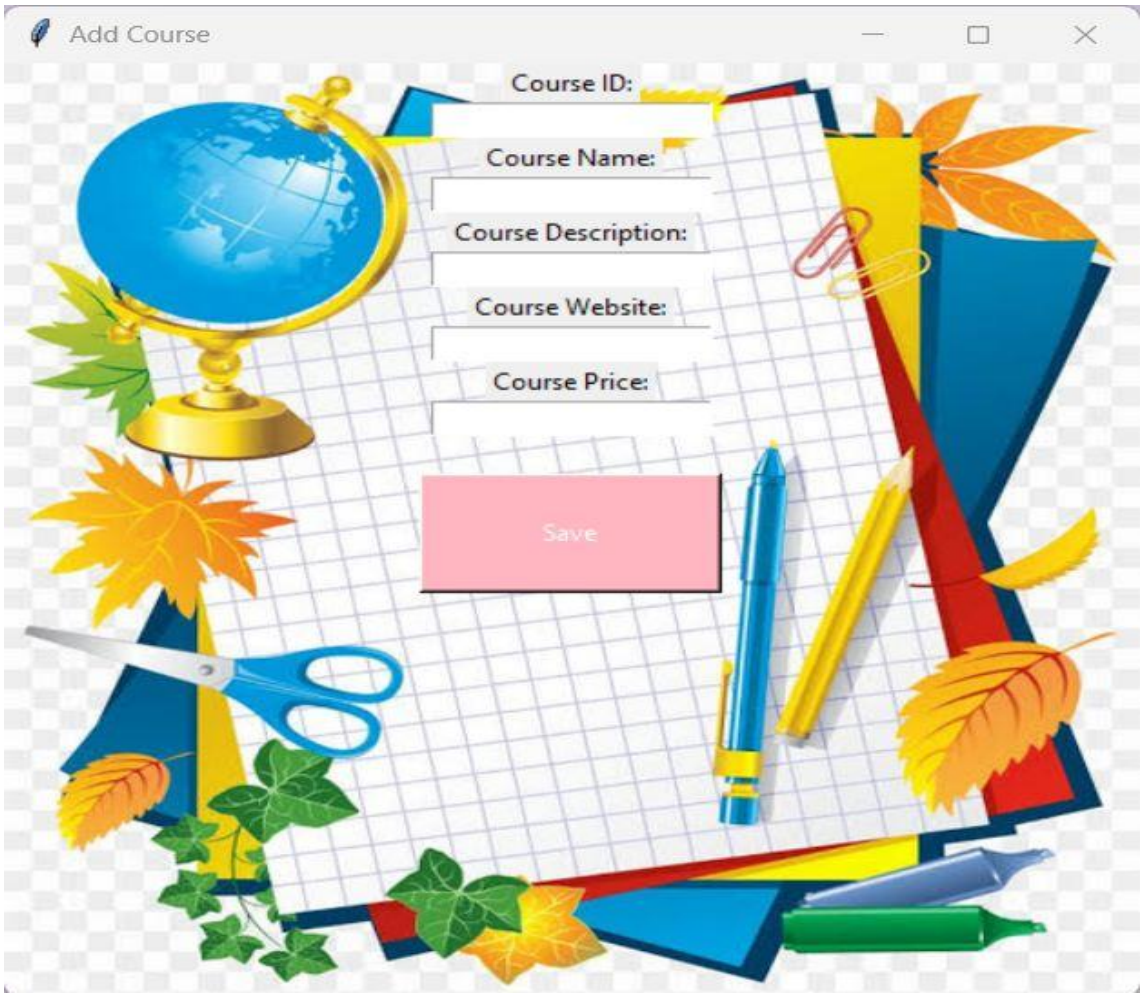


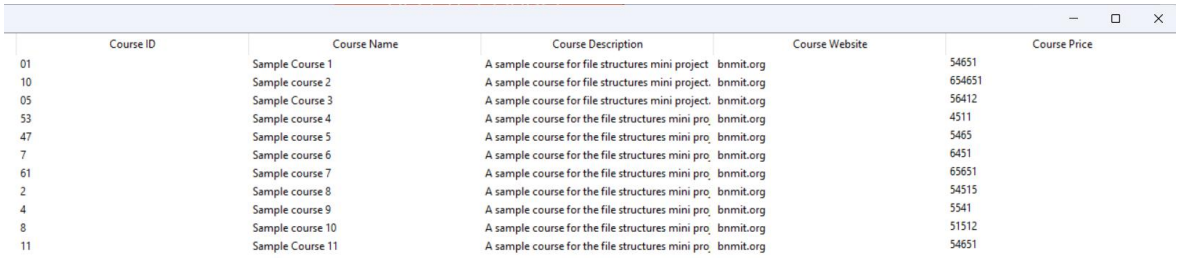
Fig 4.3 Insert Course Interface

The above fig 4.3 shows the insert course interface. If the operation desired is Insertion that is to the read the data, the user is required to enter respective option as his/her choice, from the menu displayed, after which a new screen is displayed. For the Course entry, the user is prompted to enter the Course-ID followed by the Name, price and the description of the course.

The user is prompted for each input until the values for the field has been read. This means that, the user is prompted for ID for particular field is raised until the user enters exactly 4 characters each of which is a digit, else it displays invalid number

message and user is prompted to read ID again. After all the values are accepted, btree() is called that is used to store the records in the form of a B Tree and the record is also stored in the data file and the user can press any key to return back to the menu screen.

B. Display of a record



Course ID	Course Name	Course Description	Course Website	Course Price
01	Sample Course 1	A sample course for file structures mini project	bnmit.org	54651
10	Sample course 2	A sample course for file structures mini project.	bnmit.org	654651
05	Sample Course 3	A sample course for file structures mini project.	bnmit.org	56412
53	Sample course 4	A sample course for the file structures mini pro	bnmit.org	4511
47	Sample course 5	A sample course for the file structures mini pro	bnmit.org	5465
7	Sample course 6	A sample course for the file structures mini pro	bnmit.org	6451
61	Sample course 7	A sample course for the file structures mini pro	bnmit.org	65651
2	Sample course 8	A sample course for the file structures mini pro	bnmit.org	54515
4	Sample course 9	A sample course for the file structures mini pro	bnmit.org	5541
8	Sample course 10	A sample course for the file structures mini pro	bnmit.org	51512
11	Sample Course 11	A sample course for the file structures mini pro	bnmit.org	54651

Fig 4.4 Display Courses Interface

The above fig 4.4 shows the display course interface. The records are displayed based on the ID of the option Courses are maintained in the B tree of indexes, which here is, in an ascending order. Only record with references on B tree indexes are displayed. The record is displayed in various ways. One way is to display one record with all the details together. This option displays the records in sorted order. The other option is to display the record in the form of list under common header containing different fields. Also, records are displayed in form of nodes of the B tree.

C. Deletion Of Record

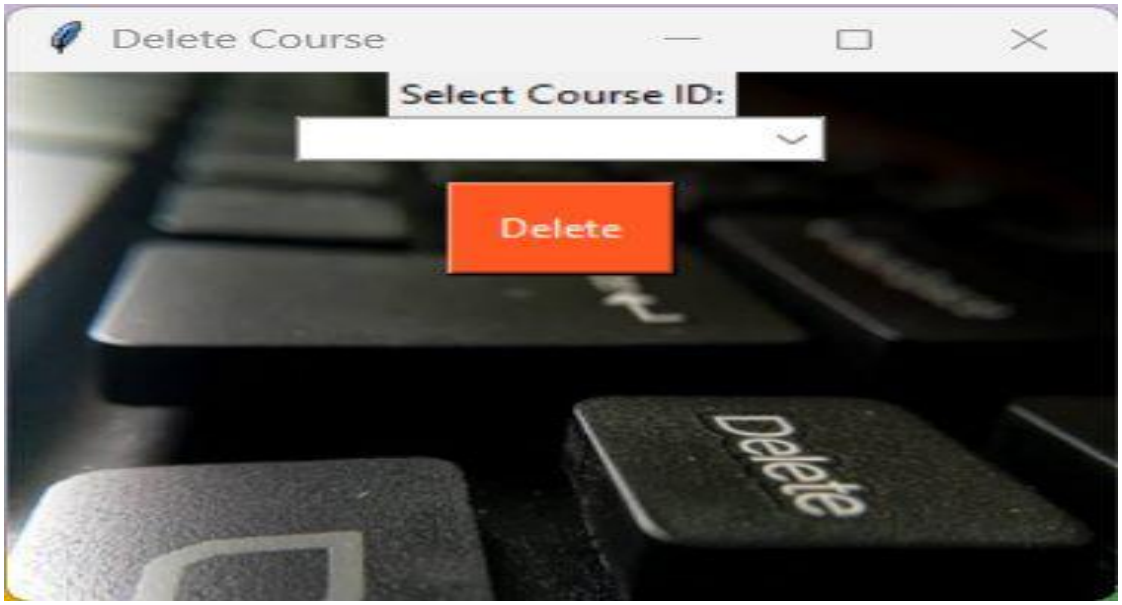


Fig 4.5 Delete Course Interface

The above fig 4.5 shows the delete course interface. If the operation desired is Deletion that is to the delete the data, the user is required to enter the ID, whose record from the file is to be deleted. If there are no records in the file, a “No Records to Delete” message is displayed, and the user is prompted to press any key to return back to the menu screen.

The ID entered is used as a key to search for a matching record. If ID doesn’t match, “Record Not Found” message is displayed. If successful, “Record Found” message is displayed then user is prompted with alert to delete the record if successfully user enter to delete, “Record Deleted” message is displayed. In each case, the user is then prompted to press any key to return back to the menu screen. The reference to a deleted record is removed from the data file. The reference to record is removed from the B Tree.

D. Searching of a record

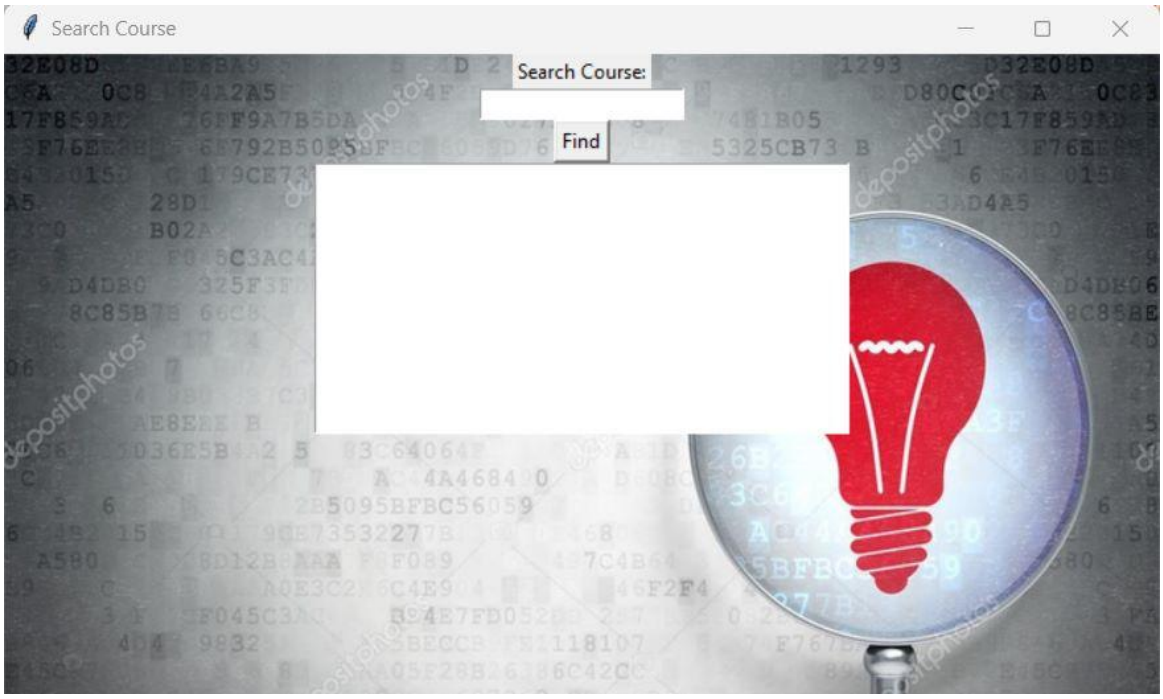


Fig 4.6 Search Course Interface

The above fig 4.6 shows the search course interface. If the operation desired is Search, the user is required to enter the respective ID course that has to be searched. If the record is not found in the data file, a “RECORD DOES NOT EXISTS” message is displayed, and the user is prompted to press any key to return back to the menu screen. If the matching key is found, the corresponding record is to be retrieved. A “RECORD FOUND” message is displayed with details of the record that was to be searched. In each case, the user is then prompted to press any key to return back to the menu screen.

E. Modification of a Record

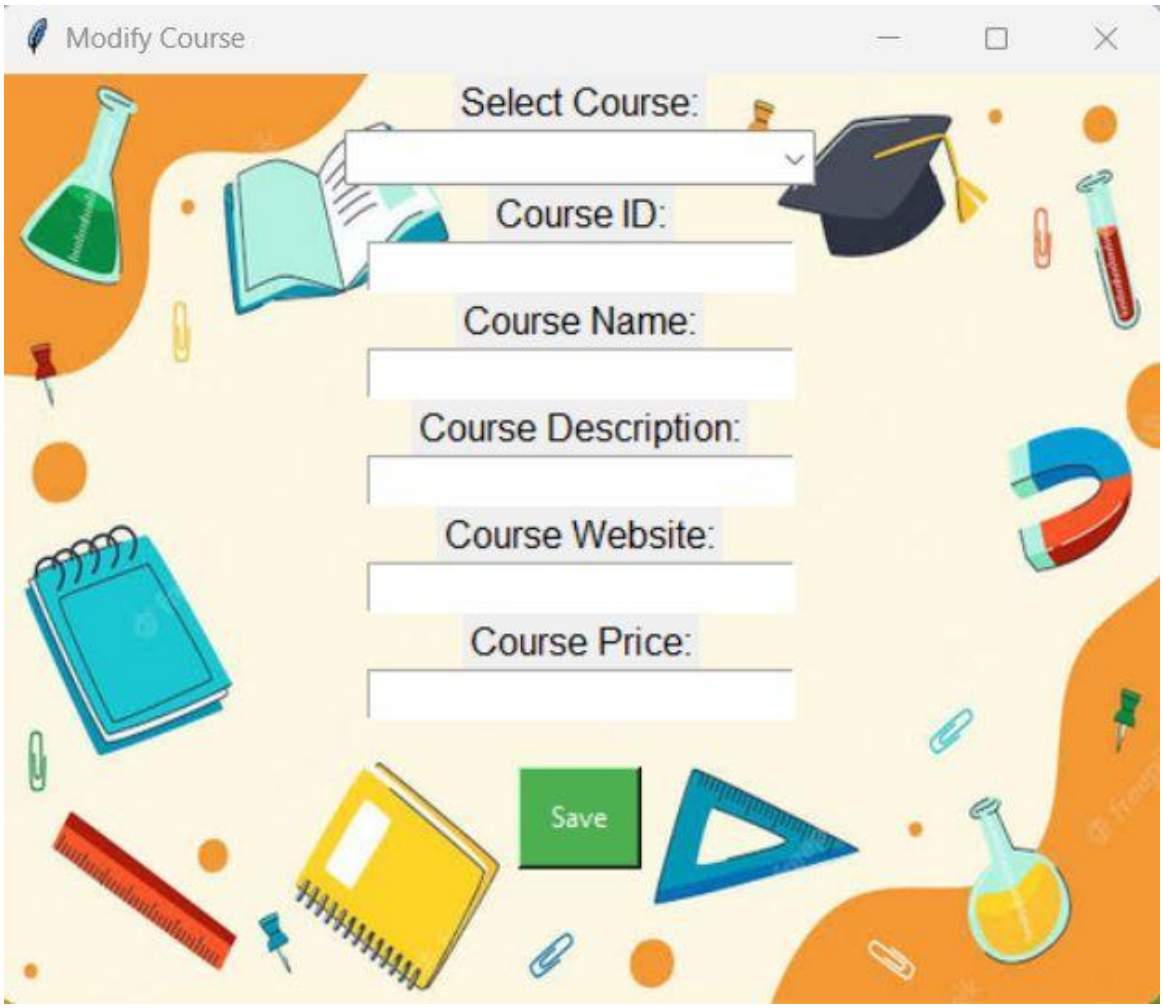


Fig 4.7 Modify Course Interface

The above fig 4.7 shows the modify course interface. If the operation desired is Modify, the user is required to enter the course ID. The Modify operation is implemented as a deletion followed by an insertion. If there are no records in the file, a “No Record to Modify” message is displayed, and the user is prompted to press any key to return back to the menu screen. If there is at least a record in the file, the user is prompted for the ID, whose matching record is to be modify. ID entered is used as a key to search for a matching record. If none is found, a “Record Not Found” message is displayed and the user is then prompted to press any key to return back to the menu screen. If one is found, a “Record Deleted” message is displayed, after which a new screen is displayed.

CHAPTER 5

IMPLEMENTATION

Implementation is the process of defining how the system should be built, ensuring that it is operational and meets quality standards. It is a systematic and structured approach for effectively integrating a software-based service or component into the requirements of end users.

5.1 List of Modules

The different types of operations supported by Indexed Sequential File Access are: - insert, search, delete and traversal.

The different operations supported by the online course management system that are used to implement the Course data objects are listed below:

1) Insert
2) Search
3) Display
4) Delete
5) Modify
6) Traversal

5.2 Description of Module

1. Insert module is used to insert details of the product to the file created by the admin. This module inserts the information in an organized manner that is pre-defined and helps in proper management of all the information.
2. Search module is used to search for specific course details that the user or admin wishes to review. This module searches the file with the help of the course id and name entered by the admin and display's the information of the corresponding movie.
3. Display module is used to review all the information that is stored in a file at once. This module can be used to see all course details that is stored in the file.
4. Delete module is used to delete a particular record that is stored in the file by using the index of that course record.
5. Modify module is used to modify the details of a particular record by using the course id as index.

6. Traversal module is used to display the id of all the movies in a tree format. This type of representation uses a technique called the B-Tree for displaying the movie id.

5.3 Pseudocode

Pseudo code is a detailed yet readable which represents the programming language statements for each module which is easier to understand of our application and its main idea is to give out its functionality.

A. Insertion Module Pseudocode

The following pseudocode represents the insertion module.

1. Even inserting at-least 1 entry into the leaf container does not make it full then add the record.

2. Else, divide the node into more locations to fit more records.

    a. Assign a new leaf and transfer 50 percent of the node elements to a new placement in the tree.

    b. The minimum key of the binary tree leaf and its new key address are associated with the top-level node.

    c. Divide the top-level node if it gets full of keys and addresses.

        i. Similarly, insert a key in the center of the top-level node in the hierarchy of the Tree.

    d. Continue to execute the above steps until a top-level node is found that does not need to be divided anymore.

3. Build a new top-level root node of 1 Key and 2 indicators.

B. Display Module Pseudocode

The following pseudocode represents the display module.

1. Open the datafile.

2. Check if datafile is empty; If empty break. Else continue the execution.

3. Set the indentation to left using set ios flags.

4. Retrieve the address of the first field in the first record.

5. While until the condition is true

    a. Unpacks the fields of the first record.

    b. Displays the contents of the record.

6. Increments the pointer to the next record.

7. Exits the loop when the file pointer reaches end of file.

C. Deletion Module Pseudocode

The following algorithm represents the deletion module pseudocode. It sketches how deletion works with various cases of deleting keys from a B-tree.

1. Start at the root and go up to leaf node containing the key K

2. Find the node n on the path from the root to the leaf node containing K

    a. If n is root, remove K

        i. if root has more than one key, done

        ii. if root has only K

            1. if any of its child nodes can lend a node

            Borrow key from the child and adjust child links

            2. Otherwise merge the children nodes. It will be a new

            root

        iii. If n is an internal node, remove K

            1. If n has at least ceil(m/2) keys, done!

            2. If n has less than ceil(m/2) keys,

            3. If a sibling can lend a key,

            4. Borrow key from the sibling and adjust keys in n and the

            parent node

            5. Adjust child links

        Else

1. Merge n with its sibling

2. Adjust child links

iv. If n is a leaf node, remove K

1. If n has at least ceil(M/2) elements, done!

In case the smallest key is deleted, push up the next key

2. If n has less than ceil(m/2) elements

3. If the sibling can lend a key

Borrow key from a sibling and adjust keys in n and its parent node

Else

4. Merge n and its sibling

5. Adjust keys in the parent nod

6. End.

D. Search Module Pseudocode

In B- Tree, a search is one of the easiest procedures to execute and get fast and accurate results from it.

1. Call the binary search method on the records in the B+ Tree.

2. If the search parameters match the exact key

a. The accurate result is returned and displayed to the user

Else

1. if the node being searched is the current and the exact key is not found by the algorithm

2. Display the statement "Record set cannot be found."

CHAPTER 6

RESULTS AND DISCUSSIONS

6.1 Snapshots of the project and description

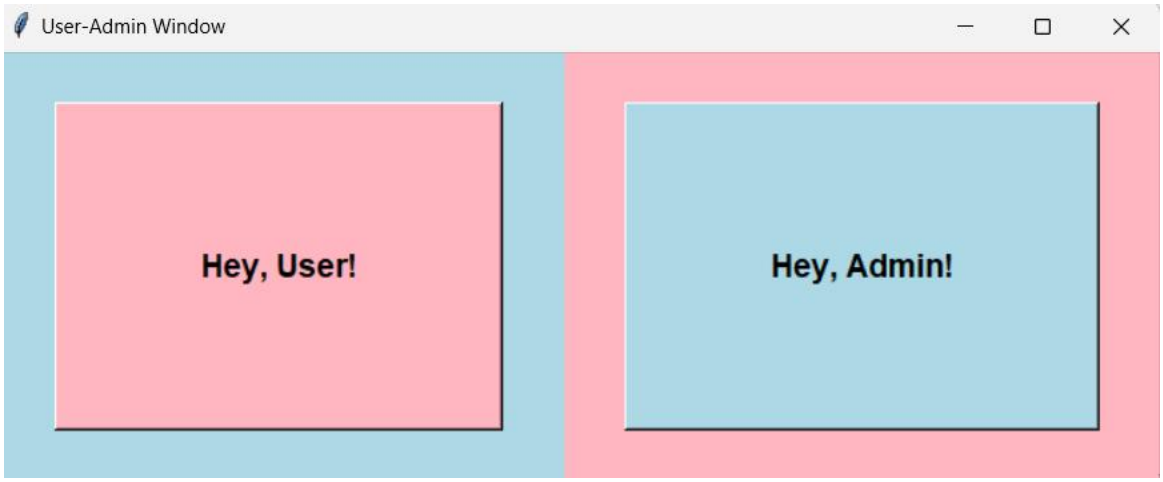


Fig 6.1 User-Admin Window

The fig 6.1 shows a window that gives the operator a choice to operate the system as an user or the admin.

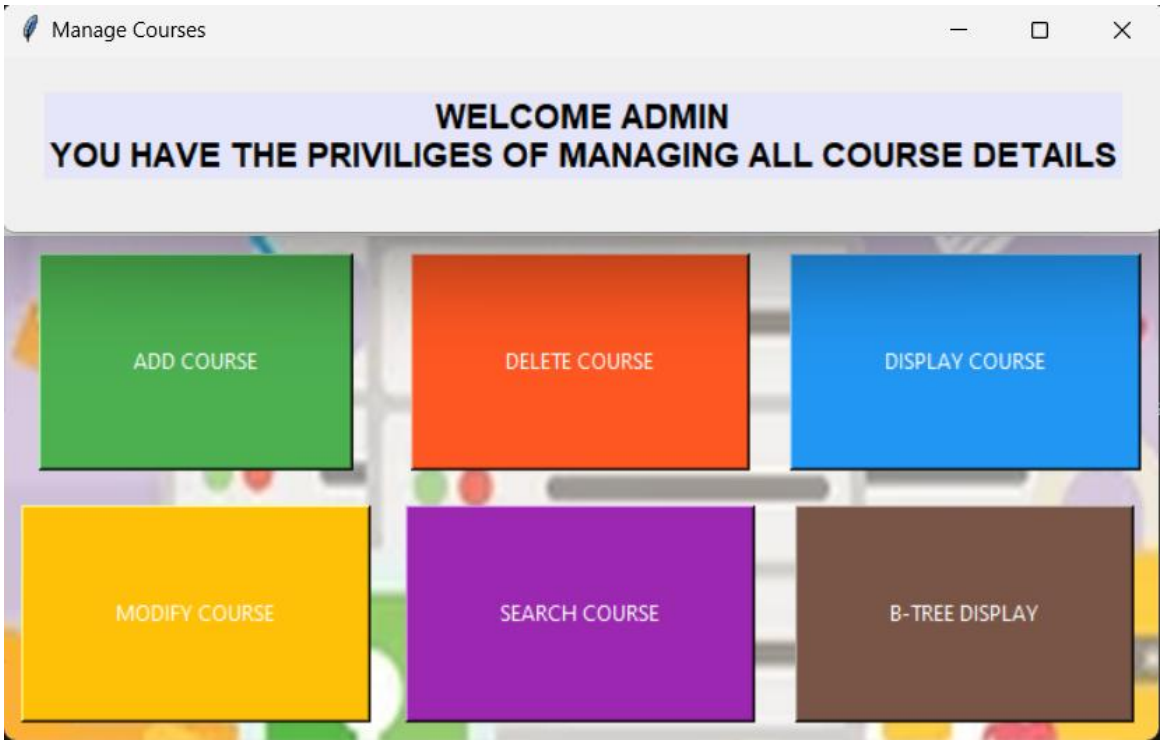


Fig 6.2 Admin Home Page

The figure 6.2 shows the admin function page that gives the admin choice to add, delete, display, modify, search the courses along with the B tree structure display.

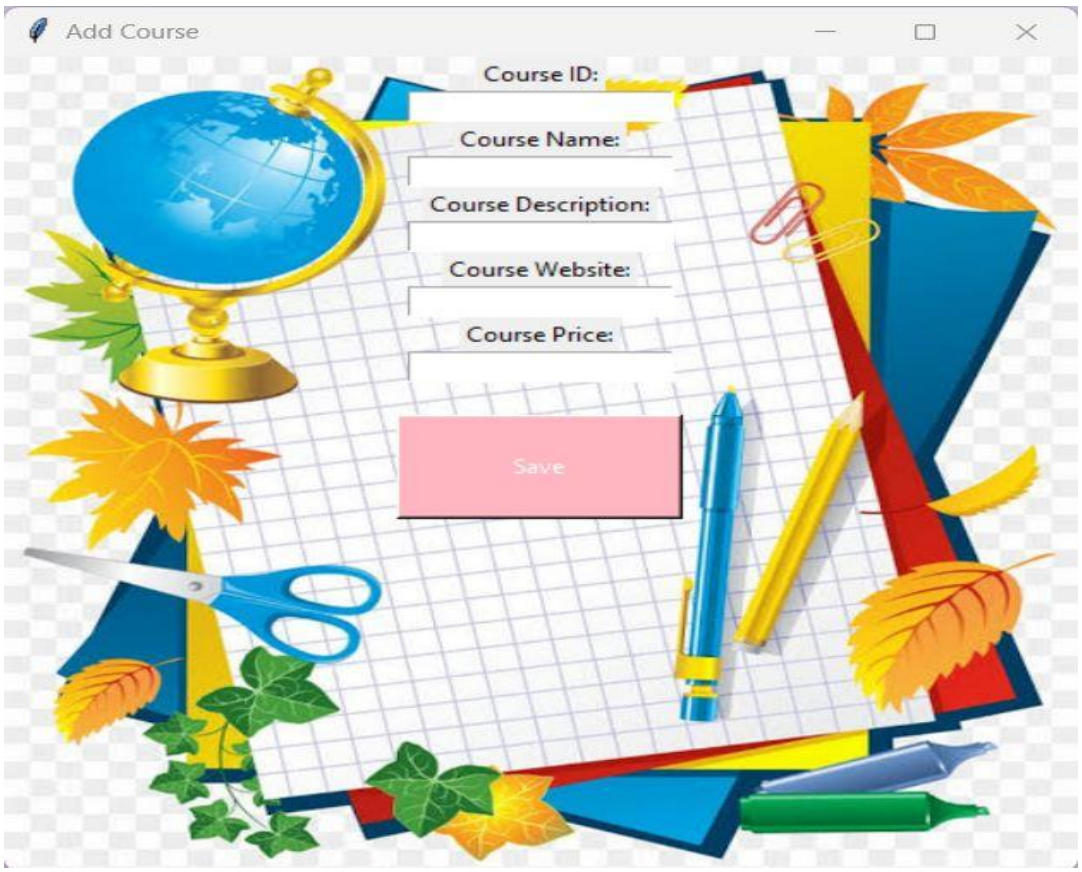


Fig 6.3 Add Course Page

The fig 6.3 shows the add course page that gives the admin the privilege to enter a new course record into the system.

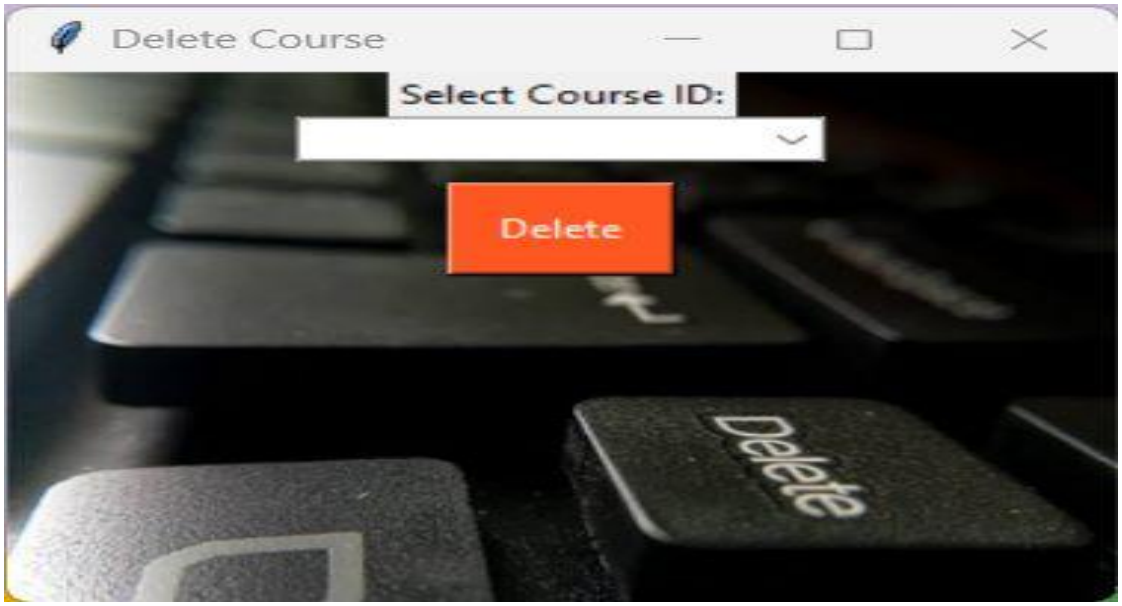


Fig 6.4 Delete Course Page

The fig 6.4 shows the delete course page, here the admin can delete unwanted courses using the respective course IDs from the drop down.

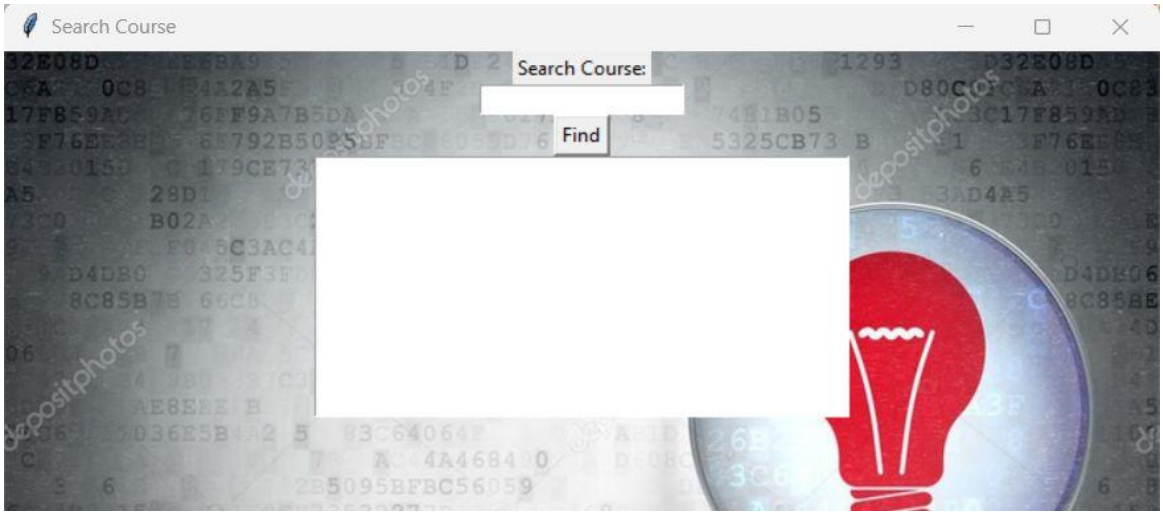


Fig 6.5 Search Course Page

The fig 6.5 shows the search course page which allows the admin to search for a particular course to retrieve its details.

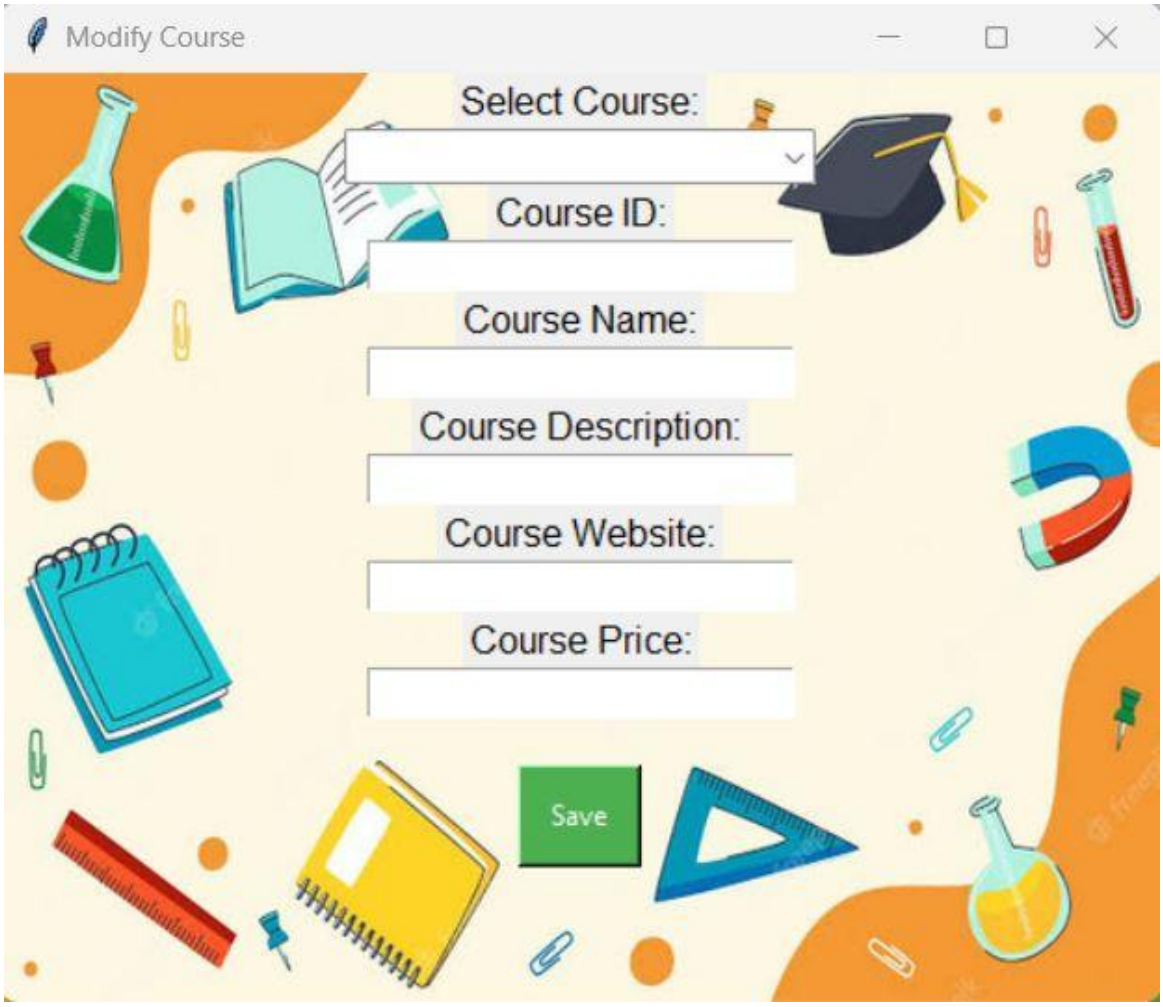
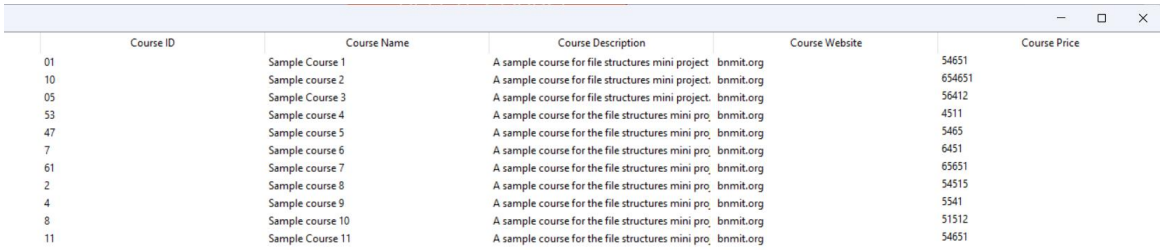


Fig 6.6 Modify Record Page

The fig 6.6 shows the modify course page that allows the admin to modify some or all the attributes of the course.



Course ID	Course Name	Course Description	Course Website	Course Price
01	Sample Course 1	A sample course for file structures mini project	bnmit.org	54651
10	Sample course 2	A sample course for file structures mini project.	bnmit.org	654651
05	Sample Course 3	A sample course for file structures mini project.	bnmit.org	56412
53	Sample course 4	A sample course for the file structures mini pro	bnmit.org	4511
47	Sample course 5	A sample course for the file structures mini pro	bnmit.org	5465
7	Sample course 6	A sample course for the file structures mini pro	bnmit.org	6451
61	Sample course 7	A sample course for the file structures mini pro	bnmit.org	65651
2	Sample course 8	A sample course for the file structures mini pro	bnmit.org	54515
4	Sample course 9	A sample course for the file structures mini pro	bnmit.org	5541
8	Sample course 10	A sample course for the file structures mini pro	bnmit.org	51512
11	Sample Course 11	A sample course for the file structures mini pro	bnmit.org	54651

Fig 6.7 Display Records Page

The fig 6.7 shows the display record page that displays all the records present in the system in a neat tabular format.

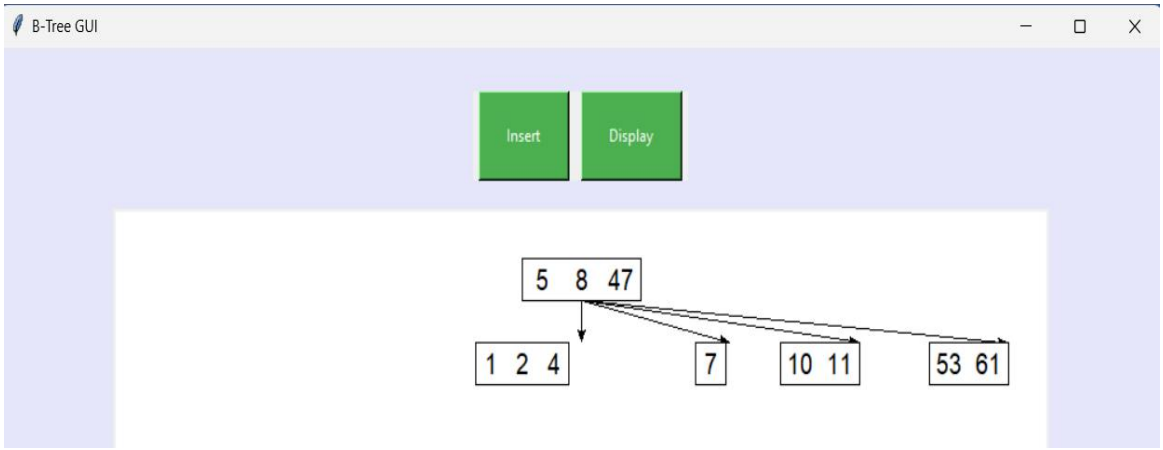


Fig 6.8 Btree Structure Display Page

The fig 6.8 shows the BTree Display page, here course IDs are used as keys to construct and display the B-tree structure.

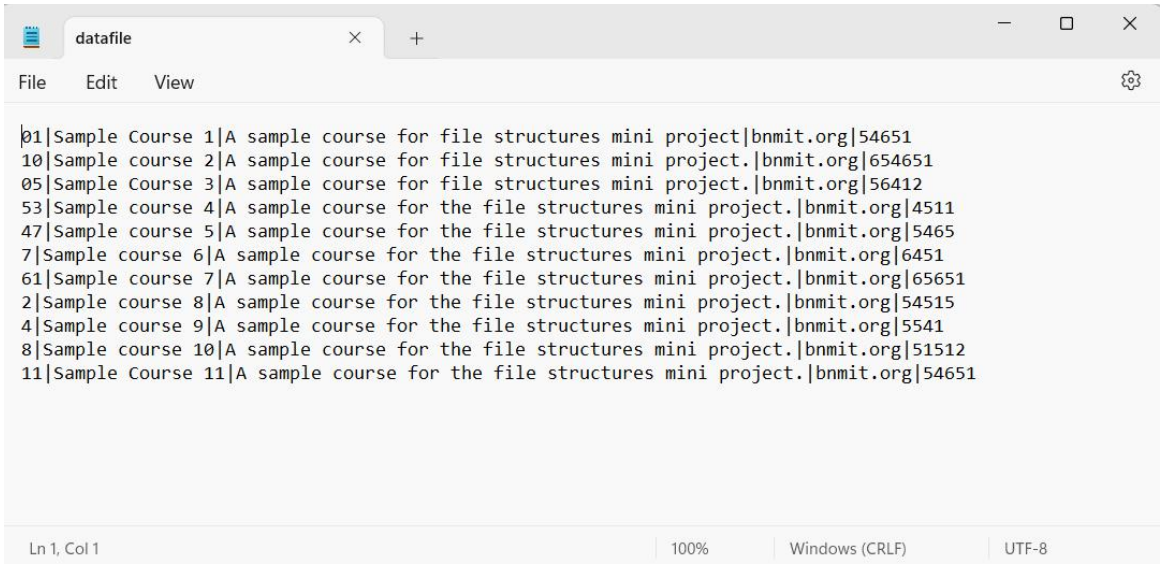


Fig 6.9 data.txt file

The fig 6.9 shows the data file where all course details are stored with a delimiter character used as a separator between the attribute fields.

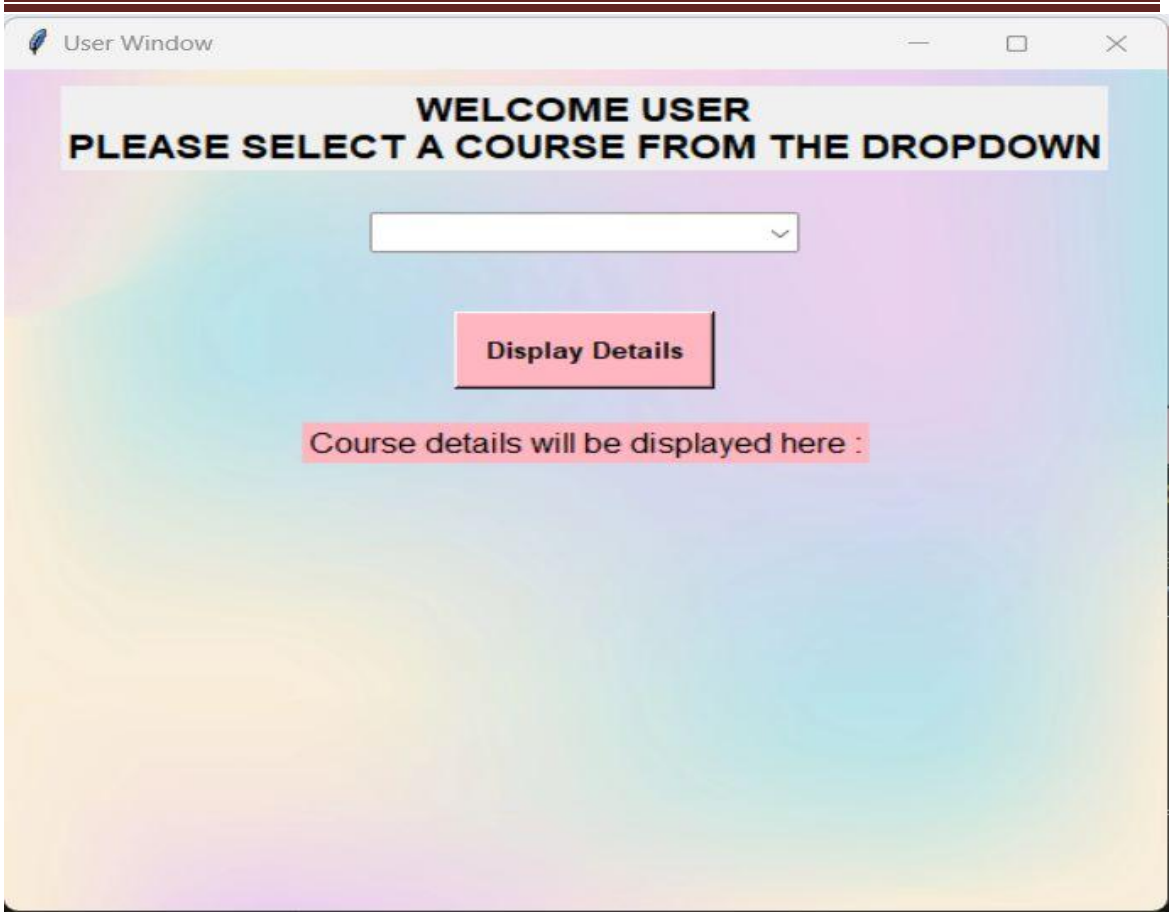


Fig 6.10 User Home Page

The fig 6.10 shows the user home page where the user can search a particular course and view its details.

6.2 Observation about the project

The mini project generalizes the file structure operations such as insert, search, traverse, delete and display by abstracting away the core functionality of the same and to implement the optimization method of Indexed Sequential Access on top of the same.. The application is observed to perform insertion, search, delete and display operations. The user can enter theinformation of any number of movie record, which will be stored in the data file. After entering all the required details, user press insert button the insert() function runs in which respective record details are stored in data file and primary keys are stored in index file along with address/rrn.

The user can enter the Course id to get respective details. After entering the Course id on the search bar, search() function is called which checks for the corresponding Course id in index file and then in data file. If Course id is not present in the index file then a dialogue box appears displaying the record with movie id is not found. If the search is successful, the user also has the option to either modify or delete

the corresponding record. The user can view whole file records by clicking Display button. The whole file is displayed in form of a table with field headers.

This application is a method for creating, maintaining, and manipulating computer files of data so that records can be retrieved sequentially or randomly by one or more keys. In this method, records are stored in the file using the primary key. An index value is generated for each primary key and mapped with the record.

Advantages of an Online Course Management System:

1. Accessibility and flexibility.

2. Improved communication and collaboration.

3. Streamlined administration and logistics.

4. Rich multimedia learning resources.

5. Personalized learning experience.

Disadvantages of an Online Course Management System:

1. Limited face-to-face interaction.

2. Technical challenges and disruptions.

3. Self-discipline and motivation required.

4. Limitations in hands-on or practical experience.

5. Access and equity issues.

CHAPTER 7

CONCLUSION

The mini project generalizes the file structure operations on a file of course records such as insert, search and delete, by abstracting away the core functionality of the same and to implement the optimization method of Indexed Sequential Access on top of the same. The project helps the user to keep track of all the information on various courses provided by the admin in a simple and structured file system, and can view the information stored in the files. The mini project can be used as a personal application for the user of all the courses that the user is currently undertaking. It also gives the user of the proper structure of a Btree based on the course Id he inserts, giving him a better structural view of the file. With further enhancements in the web functionality, this application can be used as a fulfledged web application to supply students and users with various online courses and certifications.

REFERENCES

Text books

- Michael J.Folk, Bill Zoellick, File Structure, 2nd edition
- A.V. Oppenheim and R.W. Schafer, Digital Signal Processing, Englewood, N.J., Prentice Hall, 3 Edition, 1975.

Websites

- https://www.w3schools.com
- https://code.visualstudio.com
- https://www.programiz.com
- https://app.diagrams.net
- https://online.visual-paradigm.com