AKSHAR

(Advanced Key System for Handling Archived Records)

An Office File Management Software

A Project Report
Submitted to the APJ Abdul Kalam Technological University
in partial fulfillment of requirements for the award of the degree

Bachelor of Technology
in
Computer Science and Design

by
Saleem Malik PV (KKE22CSD051)
Muhammed Ismail M (KKE22CSD040)
Lavya Shaji (KKE22CSD035)

DEPARTMENT OF COMPUTER SCIENCE
GOVERNMENT ENGINEERING COLLEGE KOZHIKODE
KERALA

October 18, 2024

DEPT. OF COMPUTER SCIENCE
GOVERNMENT ENGINEERING COLLEGE KOZHIKODE
2024

CERTIFICATE

This is to certify that the report entitled **Akshar-File Management Software** submitted by **Saleem Malik PV** (KKE22CSD051), **Muhammed Ismail M** (KKE22CSD040), and **Lavya Shaji** (KKE22CSD035) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Design is a bonafide record of the project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Dr. Bindima T
(Head of Department)
Department of CSD
Govt. Engineering College
Kozhikode

Dr. PC Reghuraj
(Project Coordinator)
Principal
Govt. Engineering College
Kozhikode

DECLARATION

We hereby declare that the project report **Akshar-File Management Software**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of Dr. PC Reghuraj.

This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that we have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Saleem Malik PV
Muhammed Ismail M
Lavya Shaji

Kozhikode
October 18, 2024

Abstract

Akshar-File Management Software is designed to revolutionize the organization and retrieval of documents within educational institutions. As colleges grow, the manual management of paper records becomes increasingly cumbersome and inefficient. When the principal presented us with the challenge of improving file management within the college, we were faced with a pivotal question: *If our Computer Science Department couldn't solve a technical problem within our own institution, what was the value of our education and the department itself?* This question became a driving force for us, infusing the project with a deep sense of mission and commitment.

This project aims to address these challenges by developing a robust digital solution using Python, CustomTkinter, and SQLite3. The software offers intuitive interfaces for both administrators and users, facilitating tasks such as file addition, deletion, updating, and search functionalities.

The primary objective is to streamline the document management process, ensuring that staff members can locate files quickly and accurately without the delays associated with traditional methods. By implementing a secure authentication system and database management, the software maintains data integrity and confidentiality, complying with institutional standards and regulations.

Moreover, the system's scalability ensures seamless operation as the volume of records grows, accommodating future expansion and increasing efficiency over time. Features include comprehensive documentation for ease of use and maintenance, ensuring sustained usability and support.

This project is not just a technical endeavor but a testament to our capabilities and the very essence of our academic journey. It reflects our commitment to innovation in educational technology, offering a practical solution to enhance productivity and reduce operational costs in managing college records.

Acknowledgement

We take this opportunity to express our deepest sense of gratitude and sincere thanks to everyone who helped us complete this work successfully. We express our sincere thanks to Dr. PC Reghuraj, Principal, Govt. Engineering College Kozhikode, for providing us with all the necessary facilities and support.

We would like to express our sincere gratitude to Dr. Bindima T, Head of Department, Department of CSD, Govt. Engineering College Kozhikode, for the support and cooperation.

We would like to place on record our sincere gratitude to all the college staff for their guidance and support throughout this work. In particular, we thank Mr. Ashraf (Sr Supdt), Mr. Praseed (Jr Supdt), and other office staff with whom we interacted.

Finally, we thank our family and friends who contributed to the successful fulfillment of this project work.

Saleem Malik PV
Muhd. Ismail M
Lavya Shaji

Contents

Chapter 1

INTRODUCTION

1.1 Purpose

The purpose of the Akshar-File Management Software is to provide an efficient and user-friendly system for organizing and retrieving files within Government Engineering College-Kozhikode. It aims to replace the time-consuming manual search for files with an efficient and user-friendly digital solution, improving productivity and allowing college staff to focus on more important tasks.

1.2 System Overview

Akshar is a file management software developed using Python, CustomTkinter, and SQLite 3. CustomTkinter is used for the graphical user interface (GUI), SQLite3 is used as the database, and Python is used as the programming language to create and connect the GUI and database. The system supports two types of users: admin and user. Admin users have the ability to add, delete, update, and search records, while regular users can only search for records.

1.3 Problem Statement

Any public office, such as the college, is required to maintain records of transactions and other activities for at least twenty years as per the RTI Act 2005. Manually searching through files whenever higher officials or college staff need a specific document is time-consuming and inefficient, leading to wasted time and effort. Recognizing this challenge, the principal, Dr. PC Raghuraj, decided to centralize all public college-related files in a dedicated room on the second floor of the main block. However, he needed software to show where the file is located in the room by locker, row, and bundle numbers whenever the file is searched by key content or file number. This software was required to address these inefficiencies, improve the productivity of staff members, and help the office implement the RTI Act 2005 more efficiently.

1.4 Software Requirement Specifications

The Akshar-File Management Software must meet the following requirements:

1.4.1 Requirements Specified by the Principal

1. **Platform:** The software must be a desktop application compatible with Windows (common office operating system).
2. **User Interface:** The software must have an intuitive and user-friendly interface to ensure ease of use for all staff members.
3. **Offline Capability:** The software must function without the need for an internet connection.
4. **File Location Identification:** The software must show the location of a document (locker, row, bundle numbers, and other details) when searched by file number or key content.

1.4.2 Additional Requirements Identified by the Development Team

1. **Roles:** The system must support two user roles:
   - **Admin:** Capable of adding, deleting, updating, and searching records.
   - **User:** Limited to searching records only.
2. **Security:** Implement user authentication to restrict access to admin functionalities.
3. **Database:** Use a proper database like SQLite3 to ensure a lightweight and efficient data storage solution.
4. **Scalability:** Design the system to handle an increasing number of records without degradation in performance.
5. **Performance:** Ensure fast and efficient search capabilities to minimize the time required to locate files.
6. **Warnings and Notifications:** Display warnings and notifications to users for actions such as file updates, deletions, or access attempts to enhance user awareness and prevent accidental actions.
7. **Documentation and maintenance :** Provide clear user and technical documentation for effective software use and management. Ensure ease of maintenance and updates for future developers.

1.5 Goal & Vision

The goal of this software is to streamline file organization and retrieval, improve efficiency, and save time for college staff members. With its user-friendly interface and advanced search capabilities, users can easily locate files without spending valuable time searching through physical records.

Chapter 2

REQUIREMENTS SPECIFICATIONS

2.1 User Characteristics

The software caters to two types of users with distinct roles and permissions:

- **Admin:** Has full access rights including adding, deleting, updating, searching, and viewing records.
- **User:** Has limited rights and can only search and view records.

2.2 Functional Requirements

The system should provide the following functionalities:

- Admin functionalities: Adding, deleting, updating, searching, and viewing records. Authentication required via password for accessing admin functionalities.
- User functionalities: Searching and viewing records. No Authentication required.
- Interaction with SQLite3 database for efficient data management.
- GUI development using CustomTkinter to ensure a user-friendly interface.

2.3 Dependencies

The software requires the following dependencies:

- Python 3.x for core functionality.
- CustomTkinter library for custom graphical user interface.
- SQLite3 library for database operations.

2.4 Performance Requirements

The system must demonstrate robust performance:

- Efficient handling of large datasets to ensure quick search results.
- Responsiveness in user interactions for seamless user experience.

2.5 Hardware Requirements

The hardware specifications for optimal performance include:

- A standard PC with at least 4GB RAM to support smooth operations.
- Minimum of 500MB free disk space for storage and processing requirements.

2.6 Constraints & Assumptions

The software operates under specific constraints and assumptions:

- **Operating system constraint:** Designed primarily for the Windows operating system.
- **User familiarity assumption:** Assumes users have basic familiarity with standard computer operations.
- **Configuration assumption:** Assumes that the computer is set to the recommended scale and layout for Windows. If not configured as such, users should adjust settings by navigating to: *Settings → System → Display → Scale and layout* and ensuring the following:
  - 'Change the size of text, apps, and other items' is set to *Windows recommended.*
  - 'Display resolution' is set to *Windows recommended.*
  - 'Display orientation' is set to *Landscape.*

Chapter 3

DESIGN

3.1 Naming and Logo Design

During the development phase, we explored various names for the software. The initial name was ”Records,” but we decided to find a more engaging and representative name. After considering several options, we shortlisted the following:

1. **Samriti:** Streamlined Access Management for Records & Information
2. **Akshar:** Advanced Key System for Handling Archived Records
3. **Sutharia:** Software Utility to Harness Accessible Information for All

We ultimately chose **Akshar** because of its fluency, brevity, and ease of pronunciation. For the logo, we initially generated a design using modern tools like generative artificial intelligence. However, the color combination did not align with our vision, so we made some refinements to improve the design. The final logo effectively represents key elements such as the name, a locker, and the concept of searching.

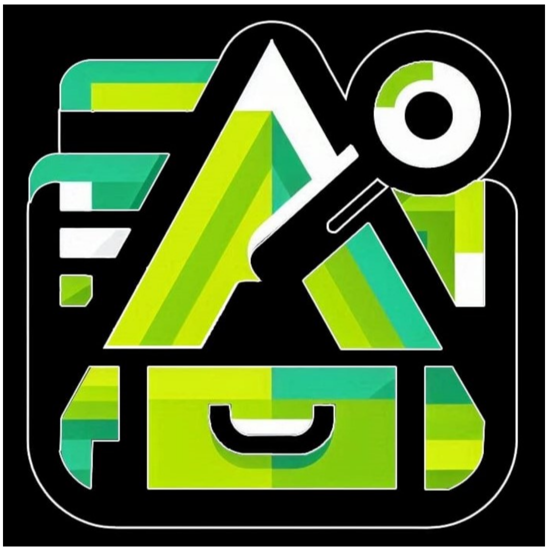


Figure 3.1: Initial Logo Design

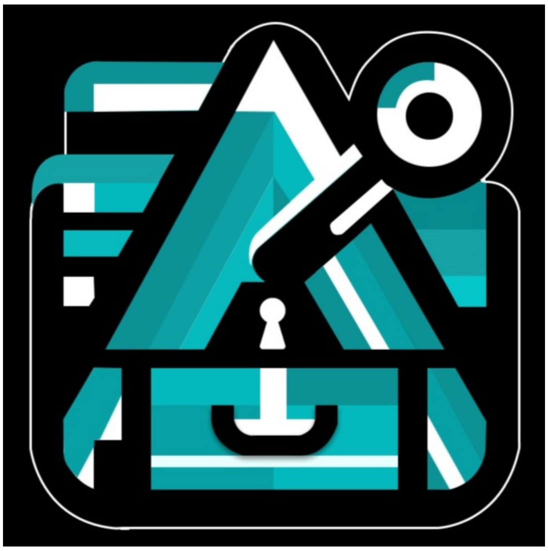


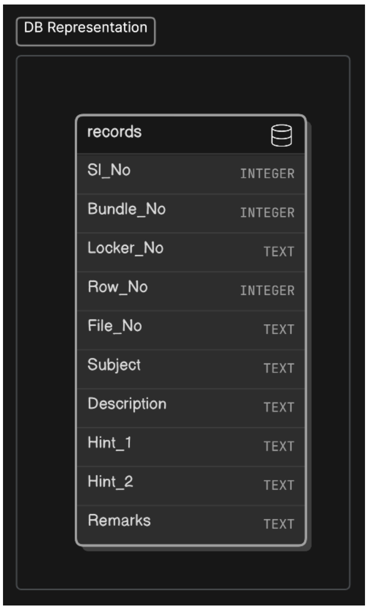
Figure 3.2: Final Logo Design

3.2 Function Oriented Design for Procedural Approach

The software is designed with a modular approach following procedural programming principles. Each module handles specific functionalities such as database operations, user authentication, and GUI management.

3.3 Database Design

The database is implemented using SQLite3 with a table structure designed to store records effectively. Below is the SQL schema for the `record` table:



DB Representation

records	
Sl_No	INTEGER
Bundle_No	INTEGER
Locker_No	TEXT
Row_No	INTEGER
File_No	TEXT
Subject	TEXT
Description	TEXT
Hint_1	TEXT
Hint_2	TEXT
Remarks	TEXT

Figure 3.3: Diagrammatic representation of the `records` database table structure.

3.4 GUI Design for Front-End

3.4.1 Tabs

The GUI features four main tabs positioned at the top of every window, allowing seamless navigation across different sections of the software. Each tab is tailored to offer straightforward access to specific functionalities. When a tab is selected, its color changes to indicate which window is currently active. (Note: The User Tab is active in Figure 3.4.)

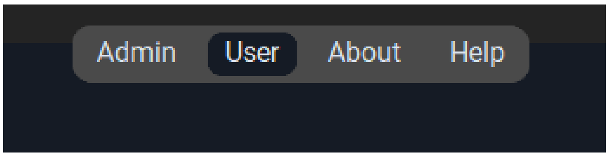


Figure 3.4: Illustration of tabs in the GUI.

- **Admin Tab:** Clicking on this tab navigates to the Admin window, requiring password authentication for access.

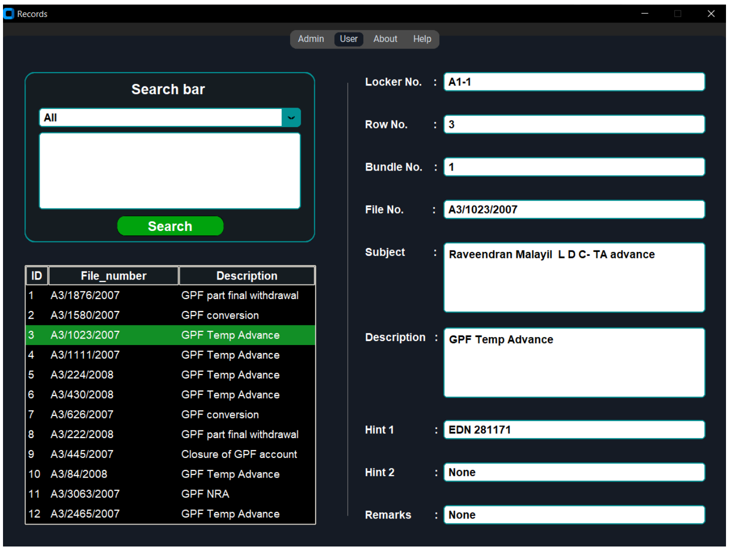
- **User Tab:** Clicking on this tab navigates to the User window, accessible without requiring a password.

- **About Tab:** Clicking on this tab directs to the About window, providing information about the software.

- **Help Tab:** Clicking on this tab leads to the Help window, offering assistance and support information.

3.4.2 User Window

The User Window is accessed when the User tab is selected from the main navigation, offering specific functionalities for searching and viewing records.



Search field: All

ID	File_number	Description
1	A3/1876/2007	GPF part final withdrawal
2	A3/1580/2007	GPF conversion
3	A3/1023/2007	GPF Temp Advance
4	A3/1111/2007	GPF Temp Advance
5	A3/224/2008	GPF Temp Advance
6	A3/430/2008	GPF Temp Advance
7	A3/626/2007	GPF conversion
8	A3/222/2008	GPF part final withdrawal
9	A3/445/2007	Closure of GPF account
10	A3/84/2008	GPF Temp Advance
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance

Locker No. : A1-1

Row No. : 3

Bundle No. : 1

File No. : A3/1023/2007

Subject : Raveendran Malayil L D C- TA advance

Description : GPF Temp Advance

Hint 1 : EDN 281171

Hint 2 : None

Remarks : None

Figure 3.5: Illustration of the User window interface.

- **Search Bar:** Positioned in the top left corner, the search bar allows users to search records based on various fields such as Sl_No, Bundle_No, Locker_No, Row_No, File_No, Subject, Description, Hint_1, Hint_2, Remarks, or all fields(All) simultaneously. (Note: In Figure 3.5, all fields(All) is selected for searching.)

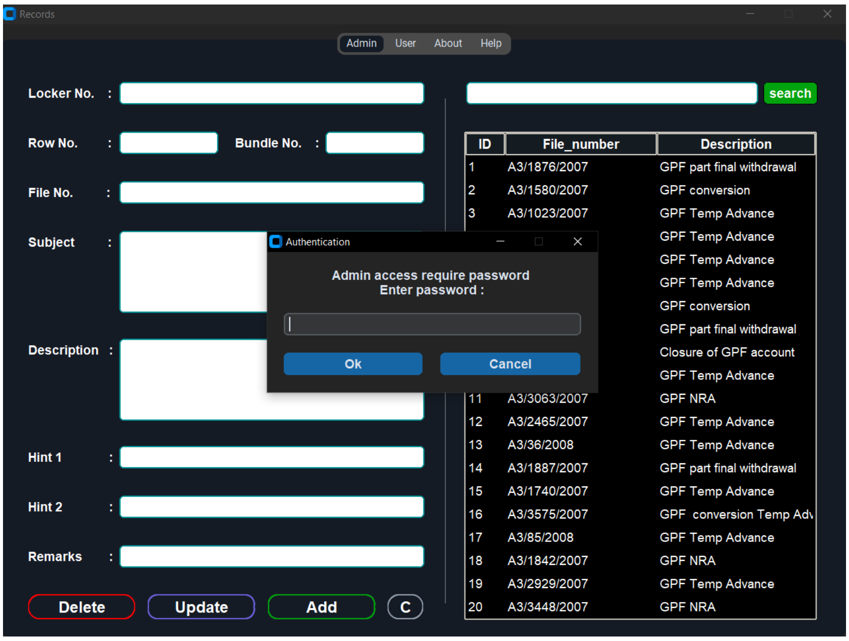
- **Table:** Located in the bottom left section, the table displays summarized information like ID, File Number, and Description for all records. Clicking on a record in the table populates detailed information on the right-hand side.

- **Separator:** A thin divider separates the left and right sections of the window, distinguishing between the search bar,table view and detailed record display.

- **Fields Section:** Located on the right-hand side, this section displays comprehensive details of the selected record from the table. Information includes Locker No, Row No, Bundle No, File No, Subject, Description, Hint 1, Hint 2, and Remarks. Users cannot edit these fields.

3.4.3 Authentication Pop-Up Window

The Authentication Pop-Up Window is opened when the Admin tab is selected from the main navigation, Authentication is required to access admin functionalities



Admin access require password
Enter password :

ID	File_number	Description
1	A3/1876/2007	GPF part final withdrawal
2	A3/1580/2007	GPF conversion
3	A3/1023/2007	GPF Temp Advance
		GPF Temp Advance
		GPF Temp Advance
		GPF Temp Advance
		GPF conversion
		GPF part final withdrawal
		Closure of GPF account
		GPF Temp Advance
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance
13	A3/36/2008	GPF Temp Advance
14	A3/1887/2007	GPF part final withdrawal
15	A3/1740/2007	GPF Temp Advance
16	A3/3575/2007	GPF conversion Temp Adv
17	A3/85/2008	GPF Temp Advance
18	A3/1842/2007	GPF NRA
19	A3/2929/2007	GPF Temp Advance
20	A3/3448/2007	GPF NRA

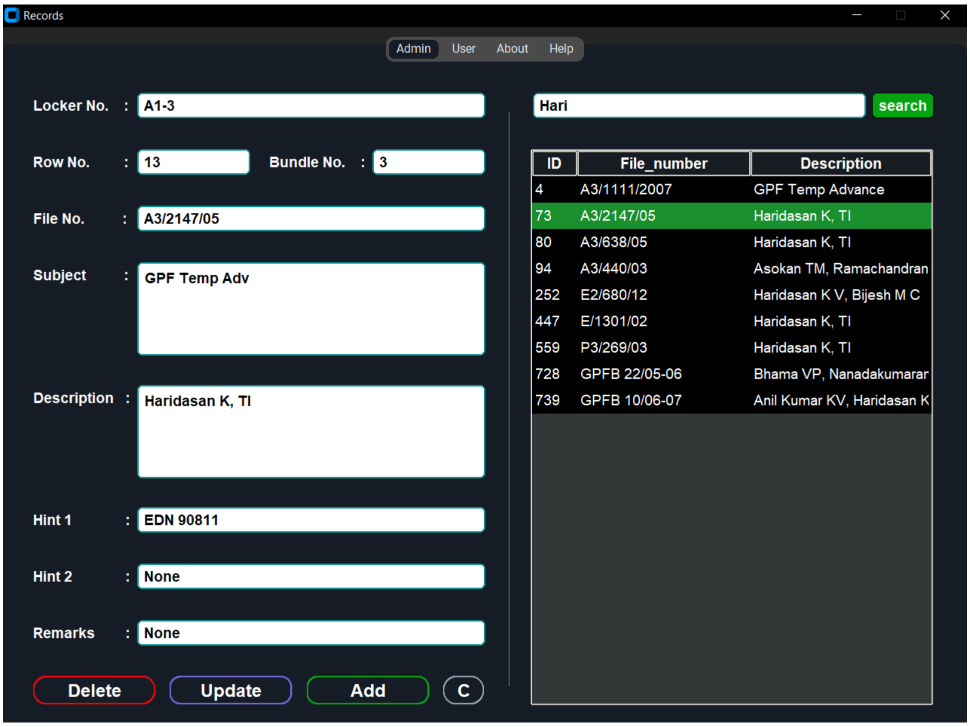
Figure 3.6: Illustration of the Authentication Pop-Up Window.

- **Close Button(X):** Redirects to the user section if clicked.
- **Password Field:** Used to enter password.
- **Ok Button:** Proceeds to the admin section if the password is correct.
- **Cancel Button:** Redirects to the user section if the password is incorrect.

3.4.4 Admin Window

The Admin Window is accessed when the Admin tab is selected from the main navigation and the password is entered correctly in the Authentication Pop-Up Window. Admins can add, delete, update, search, and view records.

- **Fields Section:** Located on the left-hand side, this section displays comprehensive details of the selected record from the table. Information includes Locker No, Row No, Bundle No, File No, Subject, Description, Hint 1, Hint 2, and Remarks. Admins can edit these fields.
- **Separator:** A thin divider separates the left and right sections of the window, distinguishing between the Fields Section and the search bar and table view.
- **Search Bar:** Positioned in the top right corner, the search bar allows users to search records based on all fields (Note: All fields include Sl_No, Bundle_No, Locker_No, Row_No, File_No, Subject, Description, Hint_1, Hint_2, Remarks) simultaneously.



Locker No. : A1-3

Row No. : 13 Bundle No. : 3

File No. : A3/2147/05

Subject : GPF Temp Adv

Description : Haridasan K, TI

Hint 1 : EDN 90811

Hint 2 : None

Remarks : None

Hari

ID	File_number	Description
4	A3/1111/2007	GPF Temp Advance
73	A3/2147/05	Haridasan K, TI
80	A3/638/05	Haridasan K, TI
94	A3/440/03	Asokan TM, Ramachandran
252	E2/680/12	Haridasan K V, Bijesh M C
447	E/1301/02	Haridasan K, TI
559	P3/269/03	Haridasan K, TI
728	GPFB 22/05-06	Bhama VP, Nanadakumarar
739	GPFB 10/06-07	Anil Kumar KV, Haridasan K

Figure 3.7: Illustration of the Admin window interface.

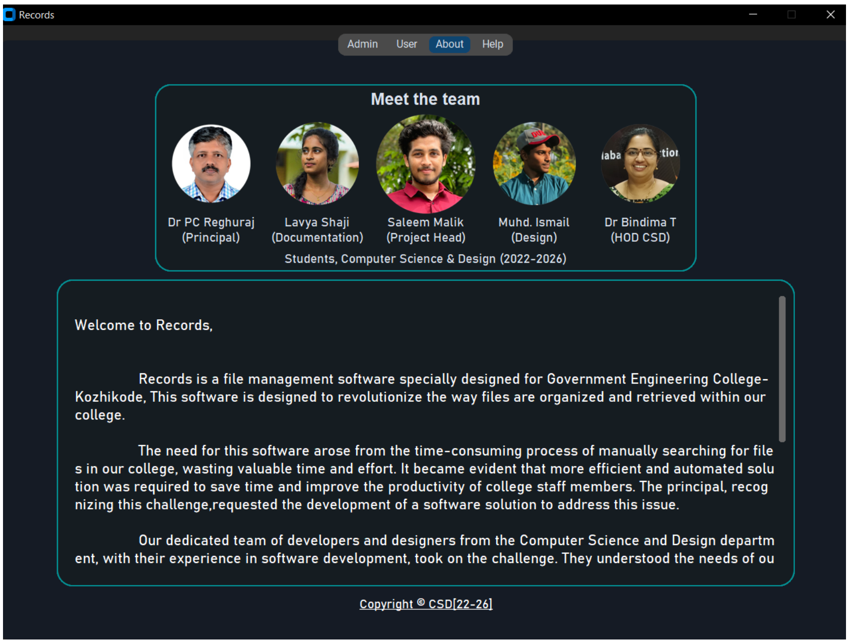
- **Table:** Located below the search bar, the table displays information such as ID, File Number, and Description for all records. Clicking on a record in the table populates detailed information in the Fields Section on the left-hand side.
- **Delete Button:** To delete a record, the button is red in color.
  - Proper way to delete a record:
    1. Select a record from the table to delete.
    2. Make sure you selected the correct record to delete by reviewing the Fields Section.
    3. Press the "Delete" button.
    4. Confirm by pressing Ok in the warning pop-up window.
- **Update Button:** To update a record, the button is violet in color.
  - Proper way to update a record:
    1. Select a record from the table to update.
    2. Make sure you selected the correct record to update by reviewing the Fields Section.
    3. Make changes to the record by editing the Fields Section.
    4. Press the "Update" button.
- **Add Button:** To add a record, the button is green in color.
  - Proper way to add a record:
    1. Click the "C" (clear) button.

2. Add information in the Fields Section.
3. Press the "Add" button.

- **Clear Button:** To clear the Fields Section, the button is grey in color.
  - Press the "C" (clear) button.

3.4.5 About Window

The About Window is accessed when the About tab is selected from the main navigation. This section provides detailed information about the software, including the team behind its development, the purpose, goals, and the reasons for its creation.



Welcome to Records,

Records is a file management software specially designed for Government Engineering College-Kozhikode, This software is designed to revolutionize the way files are organized and retrieved within our college.

The need for this software arose from the time-consuming process of manually searching for file
s in our college, wasting valuable time and effort. It became evident that more efficient and automated solu
tion was required to save time and improve the productivity of college staff members. The principal, recog
nizing this challenge,requested the development of a software solution to address this issue.

Our dedicated team of developers and designers from the Computer Science and Design departm
ent, with their experience in software development, took on the challenge. They understood the needs of ou

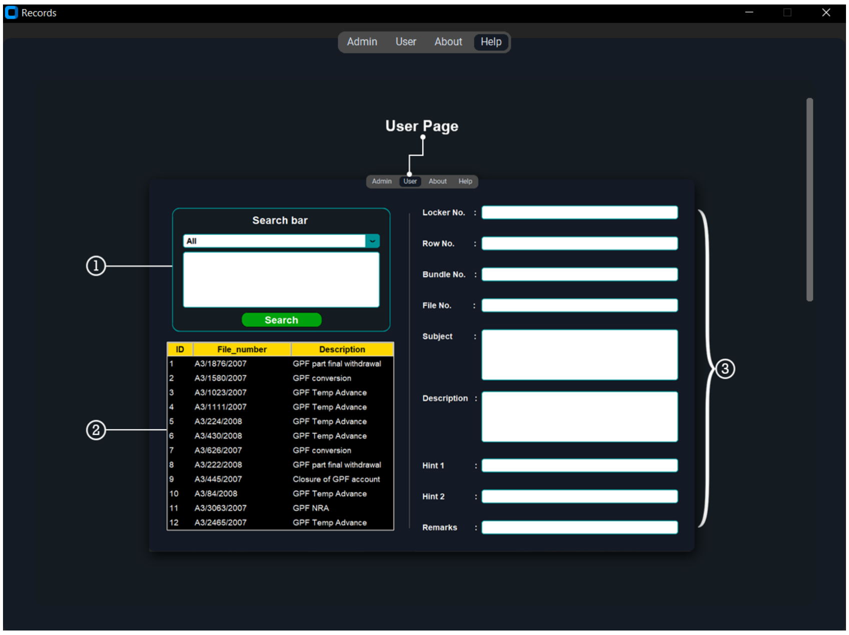
Figure 3.8: Illustration of the About window interface.

- **Team Section:** This section highlights the individuals who contributed to the development of the software. It includes their names, roles, and brief descriptions of their responsibilities. The team section provides recognition to developers, designers, and other key contributors who made significant efforts to bring the project to fruition.
- **About Section:** The About section offers a comprehensive overview of the software. It answers key questions such as:
  - **What is this software?** A description of the software's main functionalities and features.
  - **What is its goal?** The primary objectives the software aims to achieve, such as improving efficiency, providing better data management, or enhancing user experience.
  - **Why was it made?** The motivation behind creating the software, including the problems it solves and the needs it addresses.

- **Who worked on it?** Details about the contributors, including developers, designers, testers, and other stakeholders who played a crucial role in the software's development.
- **Contact Information:** Offers ways to get in touch with the development team for support, feedback, or inquiries. This might include email addresses, social media links, or a link to a support page.

3.4.6 Help Window

The Help Window is accessed when the Help tab is selected from the main navigation. This section provides comprehensive resources and support options to assist users in navigating and utilizing the software effectively.



ID	File_number	Description
1	A3/1876/2007	GPF part final withdrawal
2	A3/1580/2007	GPF conversion
3	A3/1023/2007	GPF Temp Advance
4	A3/1111/2007	GPF Temp Advance
5	A3/224/2008	GPF Temp Advance
6	A3/430/2008	GPF Temp Advance
7	A3/626/2007	GPF conversion
8	A3/222/2008	GPF part final withdrawal
9	A3/445/2007	Closure of GPF account
10	A3/84/2008	GPF Temp Advance
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance

Locker No. :
Row No. :
Bundle No. :
File No. :
Subject :
Description :
Hint 1 :
Hint 2 :
Remarks :

Figure 3.9: Illustration of the Help window interface.

- **User Manual:** A simple yet understandable user manual is available within the Help section, providing step-by-step instructions on how to use different features of the software. It includes visual aids such as screenshots and diagrams to enhance understanding.

3.4.7 Additional Design Elements

In addition to functionality and database design, the design chapter includes details on the following design elements:

- **Fonts Used:** The user interface and documentation utilize Arial and Bahnschrift fonts for clarity and readability.
- **Colors Used:** The GUI employs a specific color scheme to ensure consistency and brand identity:

- Background Color: #161C25
- Text Color: #fff
- Border Color: #0C9295
- Specific Window Colors:
    - Admin Window:
        - Add Button: Green
        - Delete Button: Red
        - Update Button: Violet
        - Clear Button: Grey
        - Search Button: Green

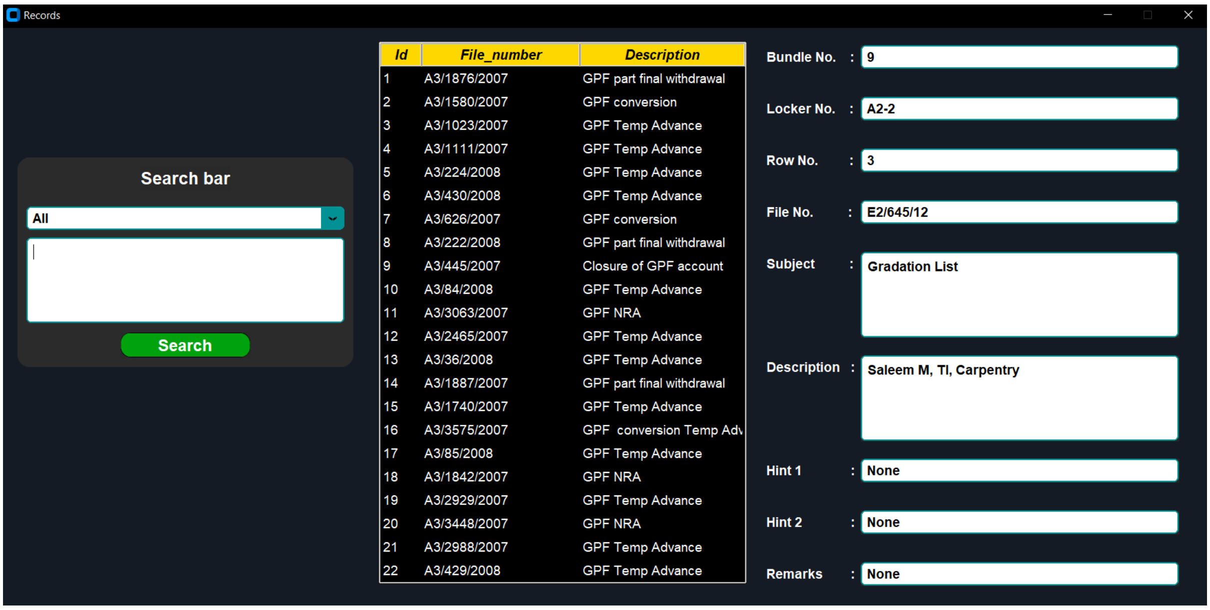
- **Window Sizes:** The GUI is designed to adapt to different screen resolutions:
    - For larger windows (like 1536x864), window sizes are set to 1025x740.
    - For smaller windows (like 1366x768), window sizes are adjusted to 1150x690 to optimize display and usability.

3.5 Design Iterations

This section describes the various design iterations for both the User and Admin windows, showing the development process that led to the final design.

3.5.1 User Window Iterations

Below are the various design iterations of the User window that led to the final output.

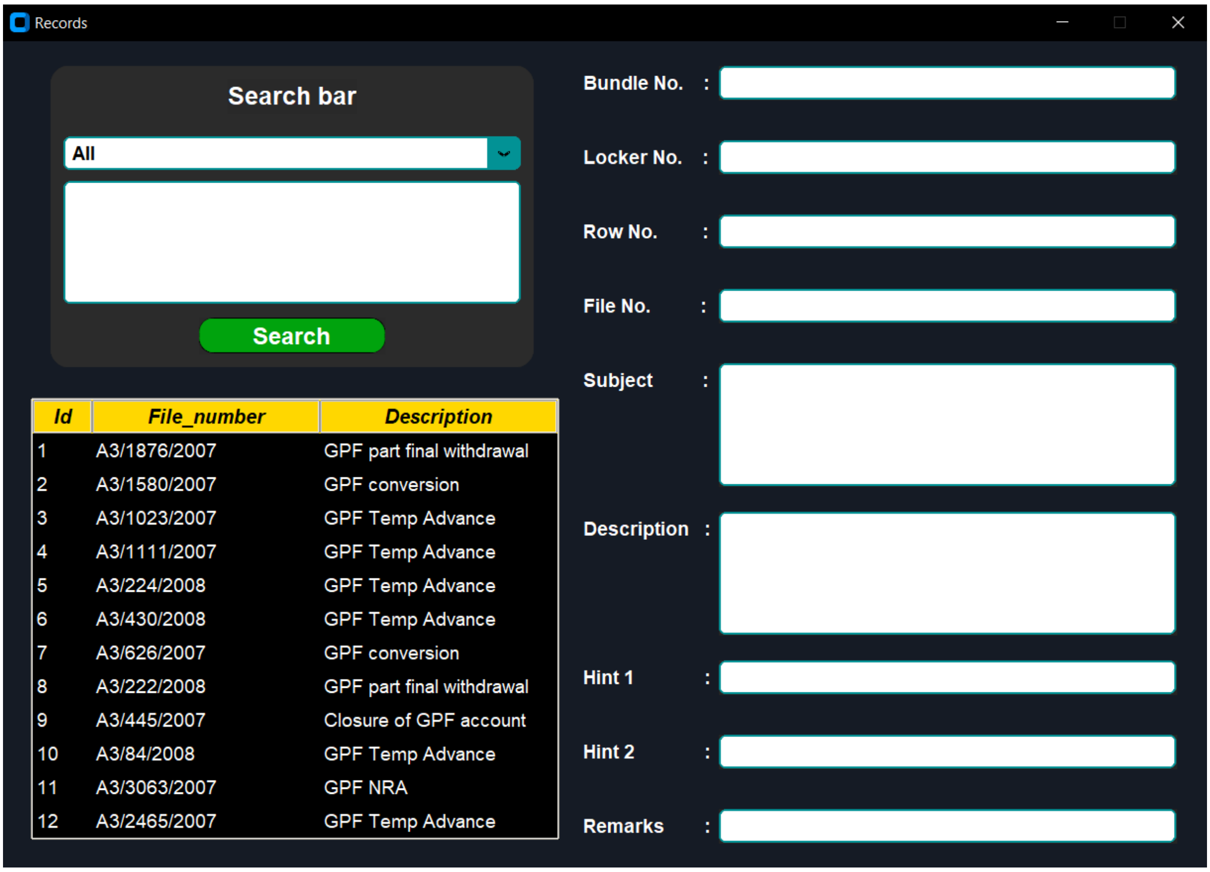


All

Id	File_number	Description
1	A3/1876/2007	GPF part final withdrawal
2	A3/1580/2007	GPF conversion
3	A3/1023/2007	GPF Temp Advance
4	A3/1111/2007	GPF Temp Advance
5	A3/224/2008	GPF Temp Advance
6	A3/430/2008	GPF Temp Advance
7	A3/626/2007	GPF conversion
8	A3/222/2008	GPF part final withdrawal
9	A3/445/2007	Closure of GPF account
10	A3/84/2008	GPF Temp Advance
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance
13	A3/36/2008	GPF Temp Advance
14	A3/1887/2007	GPF part final withdrawal
15	A3/1740/2007	GPF Temp Advance
16	A3/3575/2007	GPF conversion Temp Adv
17	A3/85/2008	GPF Temp Advance
18	A3/1842/2007	GPF NRA
19	A3/2929/2007	GPF Temp Advance
20	A3/3448/2007	GPF NRA
21	A3/2988/2007	GPF Temp Advance
22	A3/429/2008	GPF Temp Advance

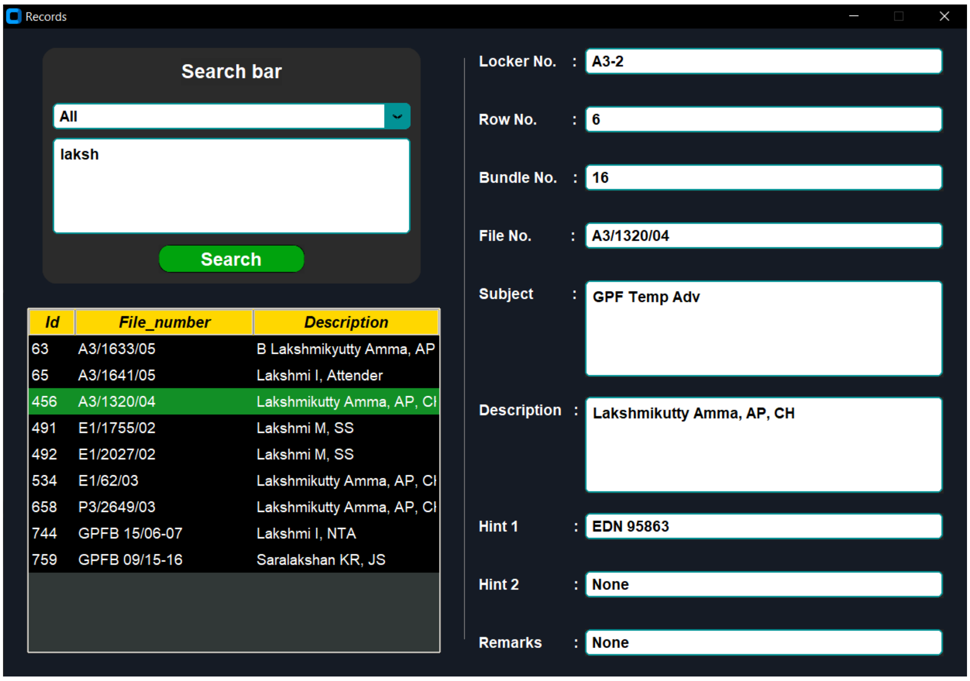
Bundle No.	:	9
Locker No.	:	A2-2
Row No.	:	3
File No.	:	E2/645/12
Subject	:	Gradation List
Description	:	Saleem M, TI, Carpentry
Hint 1	:	None
Hint 2	:	None
Remarks	:	None

Figure 3.10: User Window Iteration 1



Id	File_number	Description
1	A3/1876/2007	GPF part final withdrawal
2	A3/1580/2007	GPF conversion
3	A3/1023/2007	GPF Temp Advance
4	A3/1111/2007	GPF Temp Advance
5	A3/224/2008	GPF Temp Advance
6	A3/430/2008	GPF Temp Advance
7	A3/626/2007	GPF conversion
8	A3/222/2008	GPF part final withdrawal
9	A3/445/2007	Closure of GPF account
10	A3/84/2008	GPF Temp Advance
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance

Figure 3.11: User Window Iteration 2



laksh

Id	File_number	Description
63	A3/1633/05	B Lakshmikyutty Amma, AP
65	A3/1641/05	Lakshmi I, Attender
456	A3/1320/04	Lakshmikutty Amma, AP, CI
491	E1/1755/02	Lakshmi M, SS
492	E1/2027/02	Lakshmi M, SS
534	E1/62/03	Lakshmikutty Amma, AP, CI
658	P3/2649/03	Lakshmikutty Amma, AP, CI
744	GPFB 15/06-07	Lakshmi I, NTA
759	GPFB 09/15-16	Saralakshan KR, JS

Locker No. : A3-2

Row No. : 6

Bundle No. : 16

File No. : A3/1320/04

Subject : GPF Temp Adv

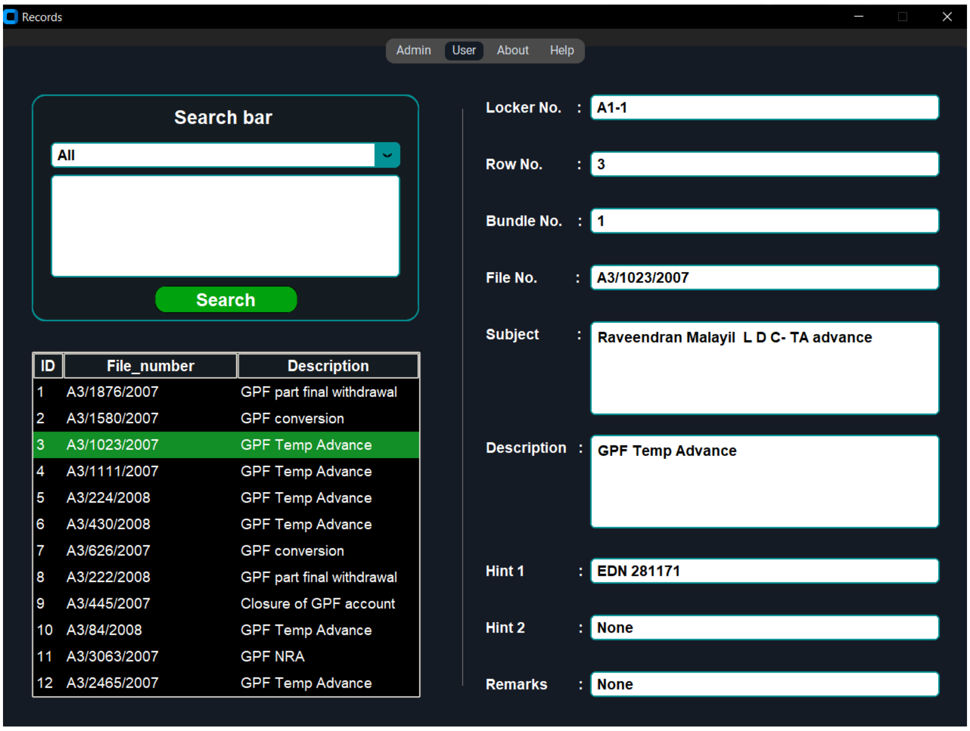
Description : Lakshmikutty Amma, AP, CH

Hint 1 : EDN 95863

Hint 2 : None

Remarks : None

Figure 3.12: User Window Iteration 3



ID	File_number	Description
1	A3/1876/2007	GPF part final withdrawal
2	A3/1580/2007	GPF conversion
3	A3/1023/2007	GPF Temp Advance
4	A3/1111/2007	GPF Temp Advance
5	A3/224/2008	GPF Temp Advance
6	A3/430/2008	GPF Temp Advance
7	A3/626/2007	GPF conversion
8	A3/222/2008	GPF part final withdrawal
9	A3/445/2007	Closure of GPF account
10	A3/84/2008	GPF Temp Advance
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance

Locker No. : A1-1

Row No. : 3

Bundle No. : 1

File No. : A3/1023/2007

Subject : Raveendran Malayil L D C- TA advance

Description : GPF Temp Advance

Hint 1 : EDN 281171

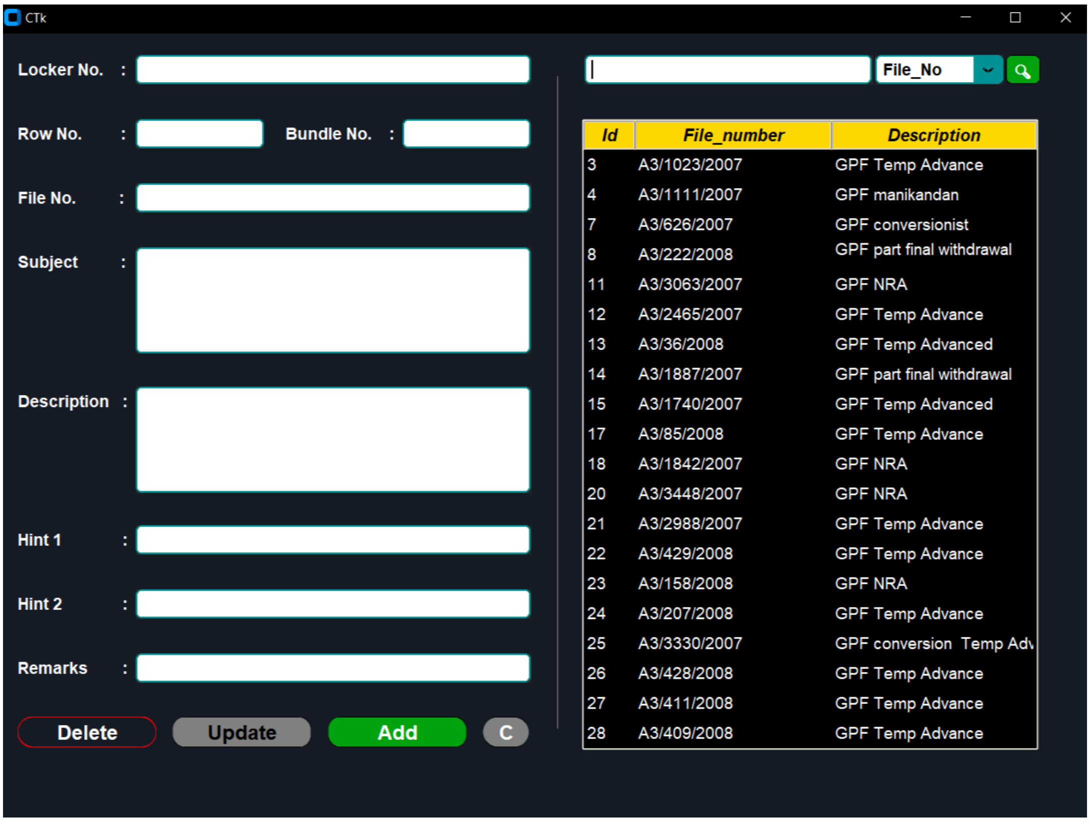
Hint 2 : None

Remarks : None

Figure 3.13: User Window Iteration 4

3.5.2 Admin Window Iterations

Below are the various design iterations of the Admin window that led to the final output.



Locker No. :

Row No. : Bundle No. :

File No. :

Subject :

Description :

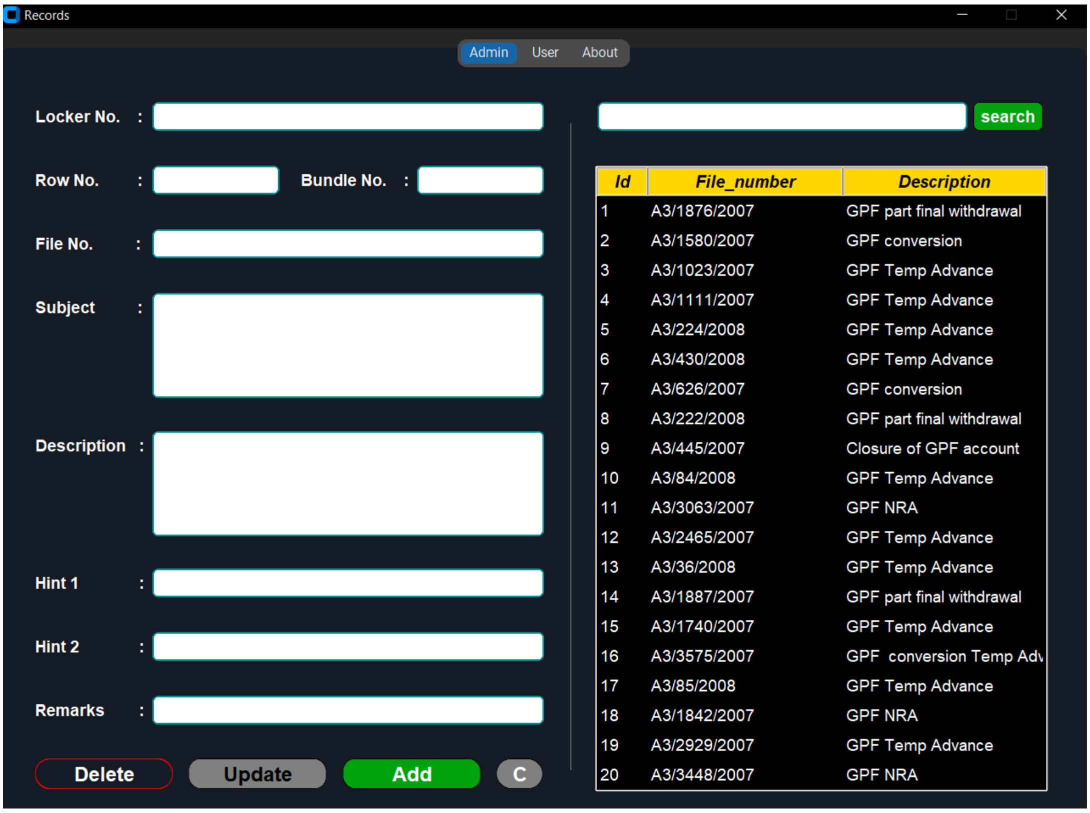
Hint 1 :

Hint 2 :

Remarks :

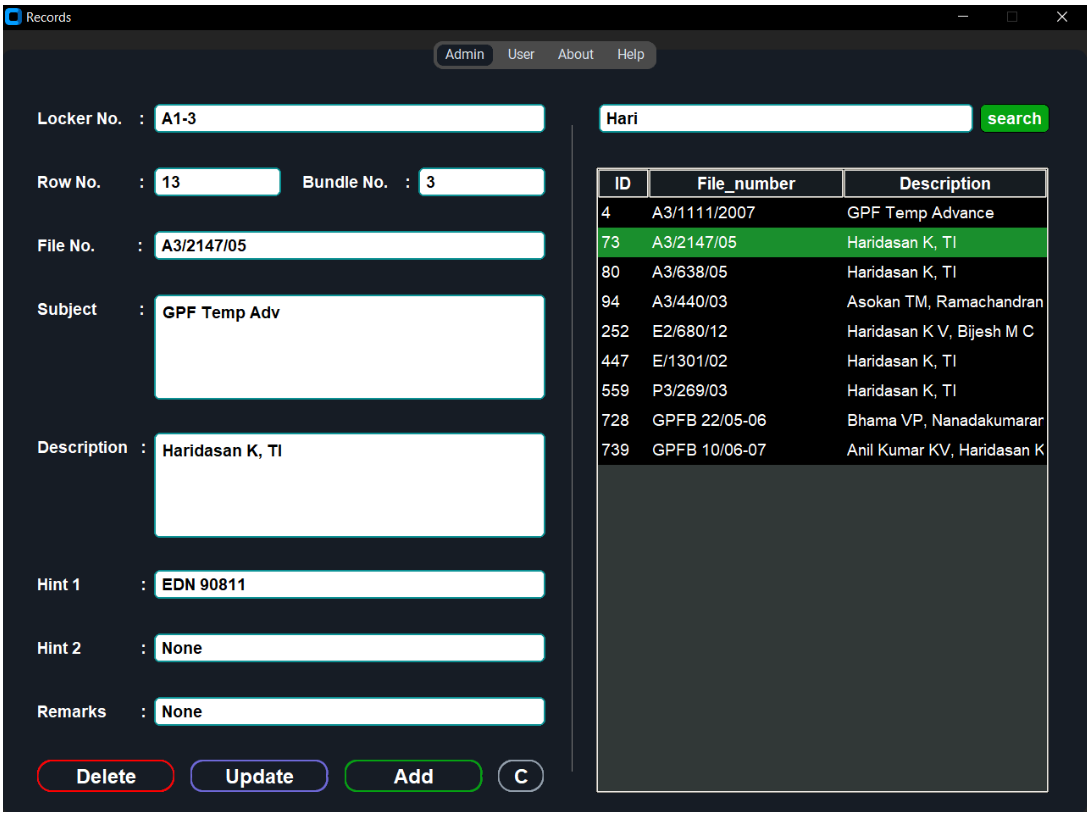
Id	File_number	Description
3	A3/1023/2007	GPF Temp Advance
4	A3/1111/2007	GPF manikandan
7	A3/626/2007	GPF conversionist
8	A3/222/2008	GPF part final withdrawal
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance
13	A3/36/2008	GPF Temp Advanced
14	A3/1887/2007	GPF part final withdrawal
15	A3/1740/2007	GPF Temp Advanced
17	A3/85/2008	GPF Temp Advance
18	A3/1842/2007	GPF NRA
20	A3/3448/2007	GPF NRA
21	A3/2988/2007	GPF Temp Advance
22	A3/429/2008	GPF Temp Advance
23	A3/158/2008	GPF NRA
24	A3/207/2008	GPF Temp Advance
25	A3/3330/2007	GPF conversion Temp Adv
26	A3/428/2008	GPF Temp Advance
27	A3/411/2008	GPF Temp Advance
28	A3/409/2008	GPF Temp Advance

Figure 3.14: Admin Window Iteration 1



Id	File_number	Description
1	A3/1876/2007	GPF part final withdrawal
2	A3/1580/2007	GPF conversion
3	A3/1023/2007	GPF Temp Advance
4	A3/1111/2007	GPF Temp Advance
5	A3/224/2008	GPF Temp Advance
6	A3/430/2008	GPF Temp Advance
7	A3/626/2007	GPF conversion
8	A3/222/2008	GPF part final withdrawal
9	A3/445/2007	Closure of GPF account
10	A3/84/2008	GPF Temp Advance
11	A3/3063/2007	GPF NRA
12	A3/2465/2007	GPF Temp Advance
13	A3/36/2008	GPF Temp Advance
14	A3/1887/2007	GPF part final withdrawal
15	A3/1740/2007	GPF Temp Advance
16	A3/3575/2007	GPF conversion Temp Adv
17	A3/85/2008	GPF Temp Advance
18	A3/1842/2007	GPF NRA
19	A3/2929/2007	GPF Temp Advance
20	A3/3448/2007	GPF NRA

Figure 3.15: Admin Window Iteration 2



Locker No. : A1-3

Row No. : 13 Bundle No. : 3

File No. : A3/2147/05

Subject : GPF Temp Adv

Description : Haridasan K, TI

Hint 1 : EDN 90811

Hint 2 : None

Remarks : None

Hari

ID	File_number	Description
4	A3/1111/2007	GPF Temp Advance
73	A3/2147/05	Haridasan K, TI
80	A3/638/05	Haridasan K, TI
94	A3/440/03	Asokan TM, Ramachandran
252	E2/680/12	Haridasan K V, Bijesh M C
447	E/1301/02	Haridasan K, TI
559	P3/269/03	Haridasan K, TI
728	GPFB 22/05-06	Bhama VP, Nanadakumarar
739	GPFB 10/06-07	Anil Kumar KV, Haridasan K

Figure 3.16: Admin Window Iteration 3

3.6 Diagrammatic Representations

The software design includes a variety of graphical representations to provide a clear understanding of the system architecture, functionality workflows, and data structures. Below are the detailed diagrammatic representations included in the design:

3.6.1 Software Architecture

This diagram illustrates different users and which windows they can open, detailing the functionalities provided by each window.

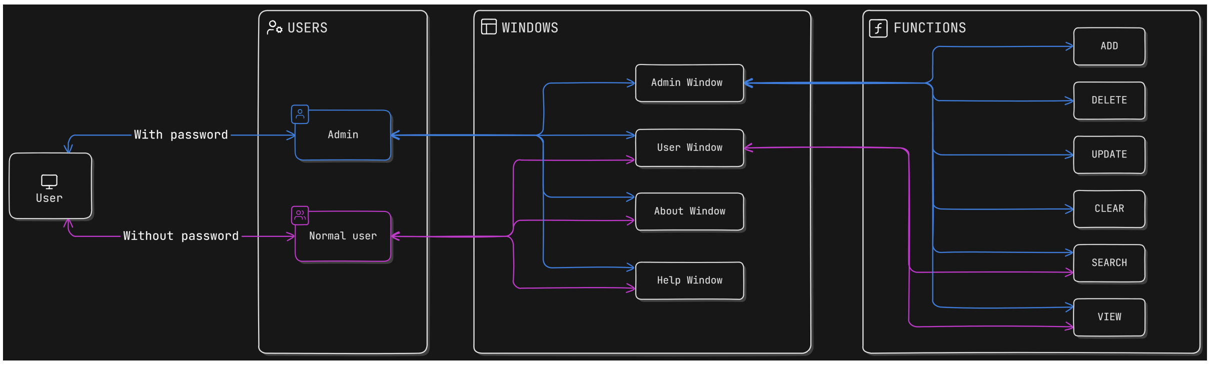


Figure 3.17: Illustration of the software architecture.

3.6.2 Functionality Flow Charts

These diagrams outline the workflow and interactions for various functionalities within the software.

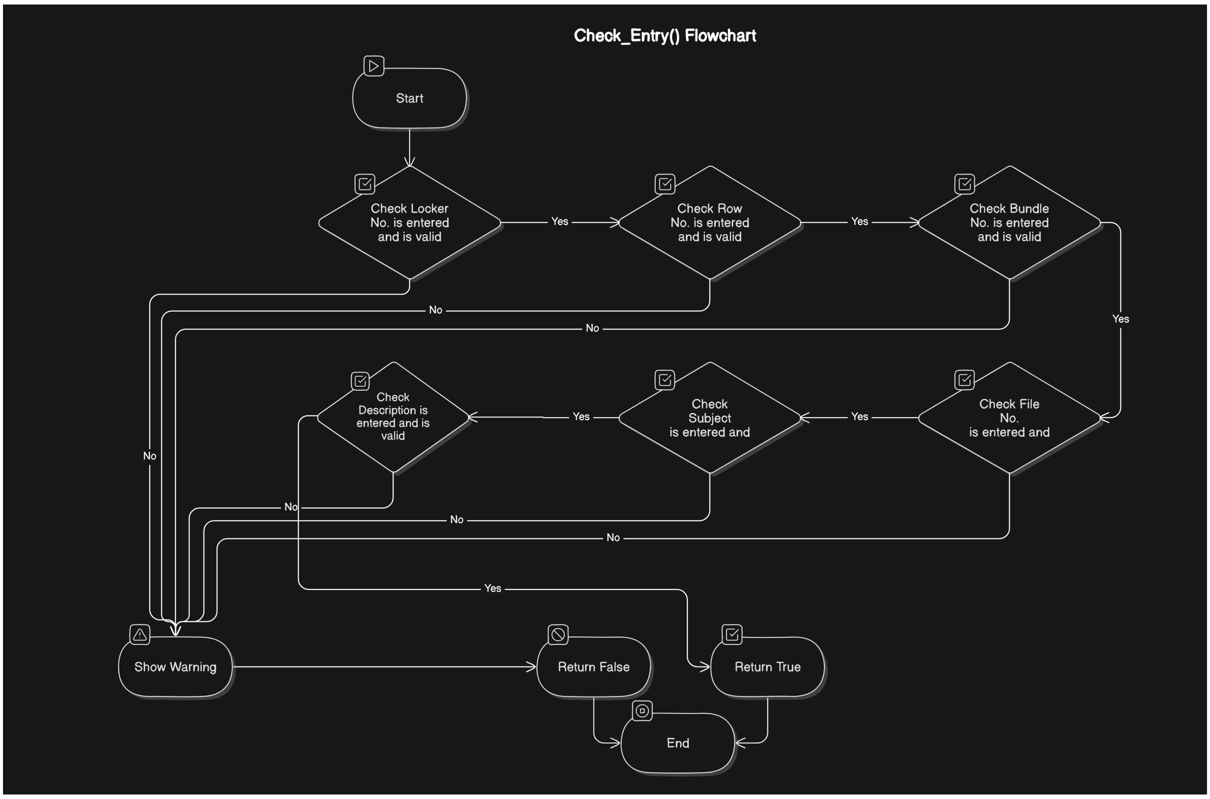

Check_Entry() Flowchart
Start
Check Locker No. is entered and is valid
Yes
Check Row No. is entered and is valid
Yes
Check Bundle No. is entered and is valid
Yes
No
No
No
Check File No. is entered and
Yes
Check Subject is entered and
Yes
Check Description is entered and is valid
No
No
No
No
Yes
Show Warning
Return False
Return True
End

Figure 3.18: Flow chart for the `check_entry()` function.

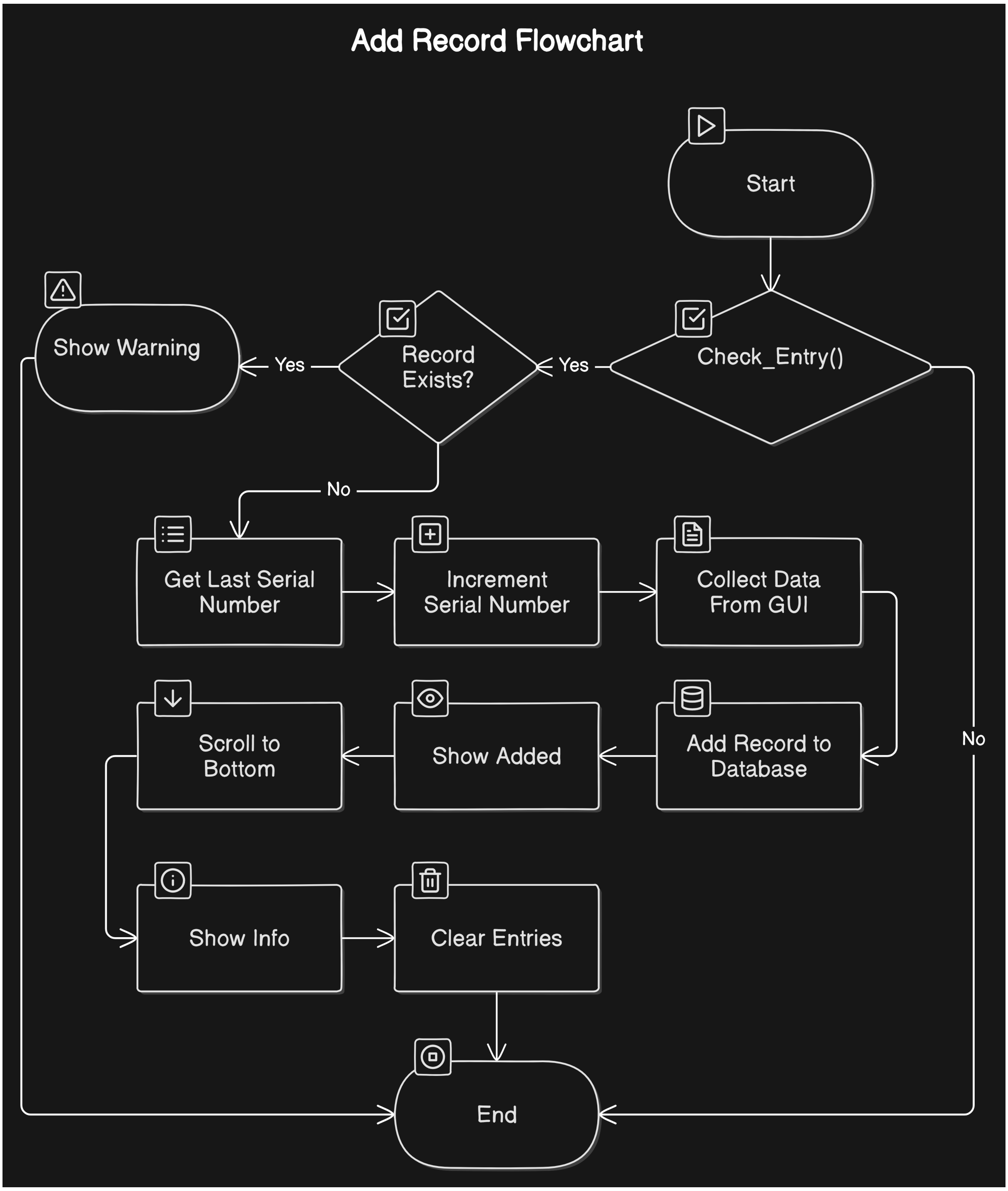


Figure 3.19: Flow chart for the `add()` function.

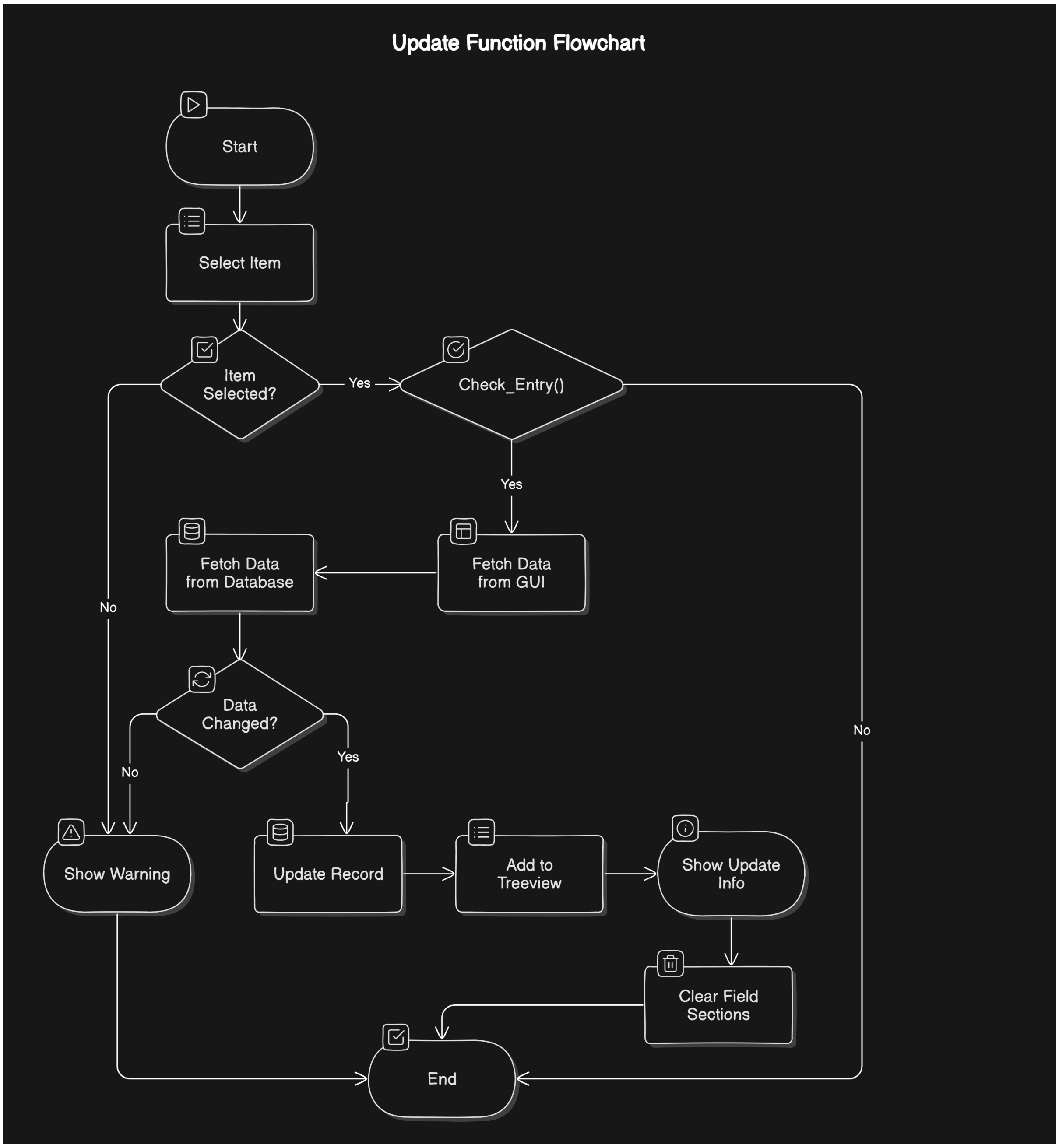


Figure 3.20: Flow chart for the `update()` function.

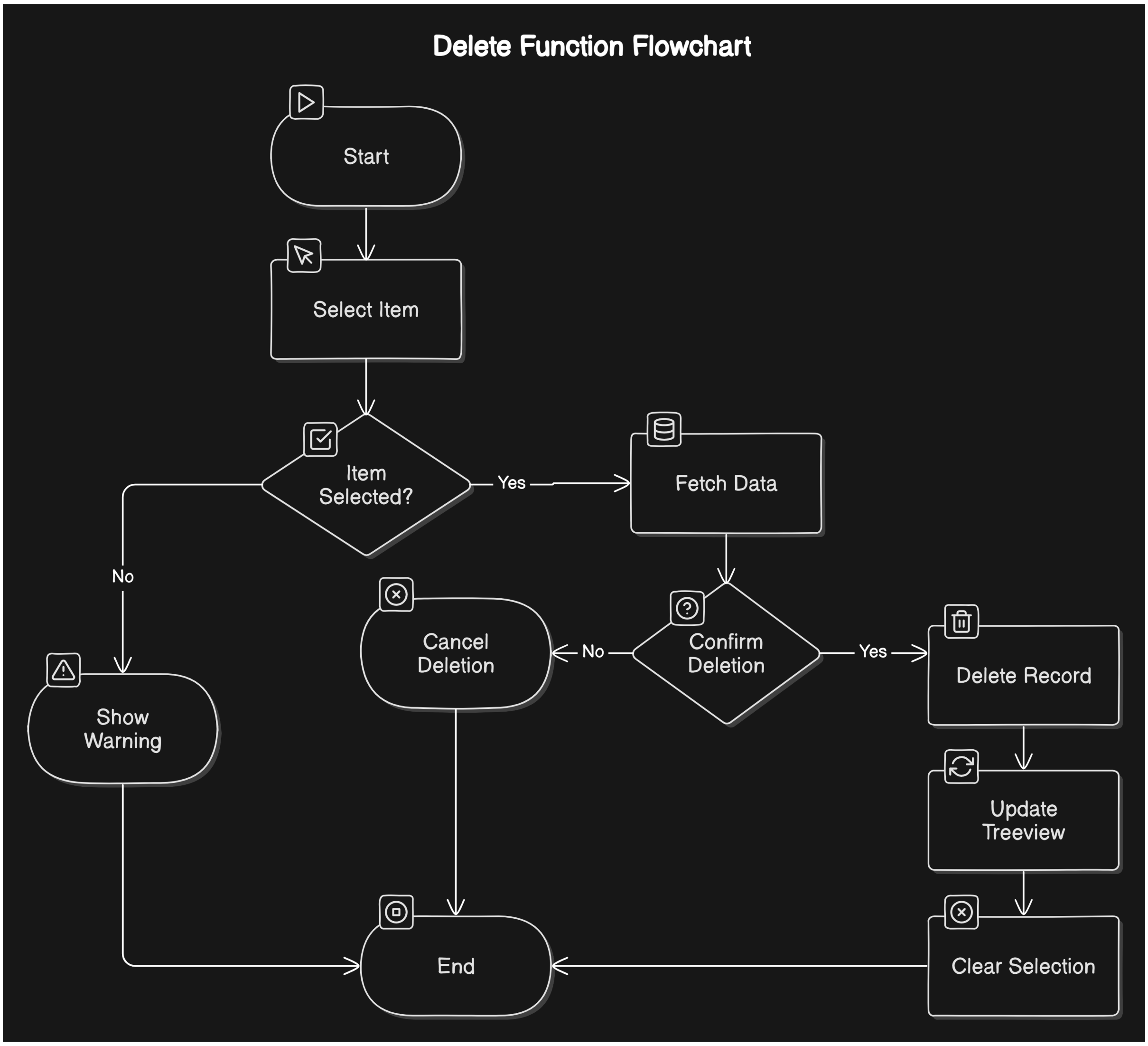


Figure 3.21: Flow chart for the `delete()` function.

Chapter 4

DIRECTORY STRUCTURE

4.1 Directory Structure and File Descriptions

The project directory structure includes the following files:

```
project_directory/

 database_admin.py        % Admin database functionalities
 database_image_bytes.py  % Functions related to handling image data in the database
 database_user.py         % User database functionalities
 function_admin.py        % Admin functions related to GUI interaction
 function_user.py         % User functions related to GUI interaction
 main.py                  % Main application file
 records.db               % SQLite database file
 window_about.py          % GUI window for "About" information
 window_admin.py          % Admin GUI window
 window_admin_1.py        % Extended Admin GUI window
 window_help.py           % GUI window for "Help" information
 window_user.py           % User GUI window
 window_user_1.py         % Extended User GUI window
```

Each file serves a specific purpose within the project:

- **database_admin.py:** Contains functions for administrative tasks related to the SQLite database, such as adding, deleting, updating and searching records.
- **database_image_bytes.py:** Includes functions for handling image data in the database, such as storing and retrieving images.contains binary data of images in the About Window.
- **database_user.py:** Includes functions for user operations on the database, particularly searching for records based on specified criteria.
- **function_admin.py:** Implements GUI interaction functions for administrative tasks like adding, deleting, and updating records through a custom tkinter interface.
- **function_user.py:** Implements GUI functions for user operations, including searching and displaying records based on user input.

- **main.py:** The main application file that integrates the GUI components and handles the main execution flow of the records management system.
- **records.db:** SQLite database file storing the records managed by the application.
- **window_about.py:** Provides the GUI window displaying information about the application.
- **window_admin.py:** The primary GUI window for admin functionalities.
- **window_admin_1.py:** An extended GUI window for additional admin functionalities in small Display resolutions.
- **window_help.py:** Provides the GUI window displaying help information.
- **window_user.py:** The primary GUI window for user functionalities.
- **window_user_1.py:** An extended GUI window for additional user functionalities in small Display resolutions.

This structured approach helps in organizing and understanding the roles of each file within the project directory, facilitating easier navigation and maintenance of the codebase.

Chapter 5

TESTING

5.1 Test Plan

The testing process includes unit testing of individual functions and integration testing of the overall system. Tests were conducted to ensure all functionalities work as expected and the software performs well under different conditions.

5.2 Test Report

- **Functionality Testing:** All admin and user functions were tested for correctness. Functions were tested with valid, invalid, and edge case inputs.
- **Performance Testing:** The system was tested with a large number of records to ensure it handles data efficiently.
- **User Interface Testing:** The GUI was tested for usability and responsiveness. Ensured all interface elements work as intended.

Chapter 6

USAGE INSTRUCTIONS

6.1 End-User Instructions

This section provides essential instructions for end-users to effectively utilize the Akshar-File Management Software. Follow these guidelines and recommendations to ensure optimal performance and ease of use.

6.1.1 Getting Started

- **Installation:**

  There is no formal installation process for this application. To get started, simply copy the ‘Akshar.exe‘ file and the ‘records.db‘ file to your desired location. Double-click the ‘Akshar.exe‘ file to open the application. It will run immediately. This app is specifically designed for Windows OS (common office operating system).

- **Initial Setup:**

  Ensure that both the ‘Akshar.exe‘ file and the ‘records.db‘ file are up-to-date and located in the same directory. Additionally, make sure that the administrator has acquired the password from the developer.

6.1.2 Basic Usage

- **Navigating the Interface:** Familiarize yourself with the main sections of the application:
  - **User Window:** Search and view records.
  - **Admin Window:** Add, update, search, and delete records.
  - **About Window:** About section of the software.
  - **Help Window:** General help guide.
- **Searching for Records:** Use the search functionality to locate records quickly. Utilize filters and keywords to refine search results and find specific documents.
- **Modifying Records:** Enter relevant details accurately when adding or updating records. Ensure all mandatory fields are completed to avoid data entry errors.Confirm twice while deleting deleting records

6.1.3 End User Recommendations

- **Regular Backups:** Schedule regular backups of your database to prevent data loss. Use the built-in backup feature or follow the recommended backup procedures to ensure your data is protected.
- **Software Updates:** Keep the software up-to-date by applying the latest patches and updates provided by the developers. This ensures you benefit from new features, improvements, and security enhancements.
- **Performance Monitoring:** Monitor system performance and report any issues to the support team. Regularly check for bugs or errors and address them promptly to ensure smooth operation.
- **Training and Support:** Take advantage of training resources and user manuals to fully understand the software's capabilities. Contact support if you encounter any issues or need assistance with advanced features.
- **Data Integrity:** Ensure data integrity by following best practices for data entry and management. Avoid manual modifications to the database to prevent corruption or inconsistencies.
- **Feedback:** Provide feedback on the software's performance and suggest improvements. Developer feedback is valuable for future updates and enhancements to better meet user needs.

6.1.4 Troubleshooting

- **Common Issues:** Refer to the troubleshooting guide for solutions to common problems, including:
  - **Records Not Showing in the Table:**
    - **Reason:** This issue may occur because the 'Akshar.exe' file and the 'records.db' file are not in the same directory, or the database name has been changed from 'records.db'.
    - **Solution:**
      - Move the 'records.db' file to the same directory as the 'Akshar.exe' file if it is not already there.
      - Rename the database file to 'records.db' if the name has been changed.
      - Use a backed-up database if the previous 'records.db' file was deleted due to accidental deletion or system failure.
  - **Cannot Log In to the Admin Window:**
    - **Reason:** The administrator may have forgotten the password.
    - **Solution:** Contact the developer to retrieve the password.
  - **The UI is Crashed or Not Aligned Properly:**
    - **Reason:** The computer is not set to the recommended scale and layout for Windows.

        - **Solution:** Users should adjust settings by navigating to: *Settings* → *System* → *Display* → *Scale and layout* and ensuring the following:
            - *Change the size of text, apps, and other items* is set to *Windows recommended.*
            - *Display resolution* is set to *Windows recommended.*
            - *Display orientation* is set to *Landscape.*

- **Contact Support:** If you encounter issues that cannot be resolved through the troubleshooting guide, contact the support team for assistance. Provide detailed information about the issue to facilitate prompt resolution.

6.1.5 Best Practices

- **Regular Database Maintenance:**

  Since the software does not use the internet, we cannot host the database on a server and must use it locally on the computer. This can lead to potential issues, such as database loss if the system crashes. To mitigate this risk, we recommend regularly backing up the database. Specifically, back up the database to the college server every two months, or whenever modifications are made, to ensure that you have recent and reliable copies in case of system failures.

- **Security Measures:**

  Do not share the admin password with normal users. Allowing unauthorized access could enable malicious users to modify or corrupt the database. To maintain the integrity and security of the system, ensure that only authorized personnel have access to administrative credentials.

By following these instructions and recommendations, users can make the most of the Akshar-File Management Software, ensuring efficient and secure management of records.

6.2 Running the Application

1. Ensure Python 3.x is installed on your system. You can download it from the official Python website: `https://www.python.org/`

2. Install the required libraries using pip:

```
    pip install customtkinter
    pip install pillow
    pip install tk

```

3. Navigate to the project directory in your terminal or command prompt.

4. Run main.py script or go to terminal and type:

```
    py main.py

```

6.3 Making the Executable

1. Navigate to the project directory in your terminal or command prompt.

2. Install the required package using the following command:

```
   pip install pyinstaller

```

3. Run the main application using the following command:

```
   pyinstaller --noconfirm --onefile --windowed --add-data "C:\Users\< USER >\AppData\Local\Programs\Python\Python310\Lib\site-packages\customtkinter;customtkinter/" "main.py"

```

4. **Explanation of the PyInstaller Command:**

   - `--noconfirm`: This option automatically confirms the removal of any previous output directory without asking the user. It ensures a clean build process by deleting old build files.
   - `--onefile`: This option packages the entire application, including the Python interpreter and all dependencies, into a single executable file. This makes distribution and execution easier as it requires only one file to run the application.
   - `--windowed`: This option ensures that the application runs without opening a console window. It is particularly useful for GUI applications where a console window is not needed.
   - `--add-data " C: USER>310-packages;customtkinter/"`: This option includes additional data files or directories that are not automatically detected by PyInstaller. The format is `source;destination`, where the source is the path to the directory or file to include, and the destination is the directory within the bundled application where these files will be placed. Here, it includes the `customtkinter` library. The placeholder `< USER >` should be replaced with your actual Windows username.
   - `"main.py"`: This specifies the main Python script of your application. PyInstaller will use this script as the entry point to package the application.

6.4 Developer Recommendations

- Maintain the database schema and ensure any changes are well-documented.

- Regularly update the `README.md` file with any changes in installation or usage instructions.

- Follow a consistent code style and naming conventions across all files for better readability and maintenance.

- Use version control systems like Git to manage code changes and collaborate with other developers.

- Write unit tests for critical functions to ensure the reliability and stability of the application.

This structured approach ensures that both end-users and developers can efficiently install, run, and maintain the application.

Chapter 7

FUTURE WORK

The Akshar-File Management Software has laid a solid foundation for efficient document management within educational institutions. However, several areas present opportunities for further enhancement and development:

- **Backup and Restore Feature:** Implementing a comprehensive backup and restore mechanism is crucial to safeguard data against loss due to system failures, accidental deletions, or other unforeseen events. This feature will allow users to schedule regular backups and restore the database to a previous state if needed. Additionally, implementing version control for backups can provide multiple restore points, enhancing data security and reliability.

- **Enhanced Graphical User Interface (GUI):** Further improving the GUI can significantly impact user experience. This involves making the interface more intuitive and user-friendly, incorporating feedback from current users. Features such as customizable dashboards, more responsive design elements, and improved accessibility options (e.g., support for screen readers) can make the system more efficient and inclusive. Additionally, integrating modern UI frameworks and incorporating interactive elements such as drag-and-drop functionalities could enhance usability.

- **Integration with Existing Systems:** Future work could involve integrating the Akshar-File Management Software with other institutional systems such as Etlab, and other office softwares. This integration can streamline workflows, reduce redundancy, and provide a unified platform for managing various aspects of college administration.

- **Advanced Search and Reporting Capabilities:** Enhancing the search functionality to support more complex queries and filters can improve the efficiency of locating records. Adding advanced reporting features, such as generating custom reports and analytics on document usage and status, can provide valuable insights and aid in decision-making processes.

- **Mobile Accessibility:** Developing mobile-friendly versions or applications for the software can extend its usability beyond desktop environments. This would allow users to access and manage records from smartphones and tablets, increasing convenience and accessibility for on-the-go use.

- **User Training and Support:** Implementing comprehensive training programs and support resources for users can enhance the effectiveness of the software. This includes creating detailed user manuals, video tutorials, and a helpdesk system to assist users with troubleshooting and technical issues.

- **Performance Optimization:** Continuously monitoring and optimizing the performance of the software is essential to handle increasing volumes of data efficiently. This involves refining database queries, improving indexing strategies, and optimizing the overall system architecture to ensure fast and reliable performance.
- **Scalability Enhancements:** As the institution grows, the software should be scalable to accommodate more records and users. Future work could include upgrading the system architecture to support distributed databases or cloud-based solutions, ensuring that the software can handle future growth and additional requirements.

Chapter 8

CONCLUSION

The Akshar-File Management Software marks a pivotal achievement for Government Engineering College - Kozhikode. Developed by the inaugural batch of the Computer Science and Design Department, this project aimed to address the urgent need for a user-friendly system to manage and retrieve records, a challenge presented shortly after the department's establishment in 2022.

When the principal presented us with the challenge of improving file management within the college, we were faced with a pivotal question: **if our Computer Science and Design Department could not tackle a technical problem within our own institution, what was the purpose of our education and our department? This question became a driving force for us.** We took on the project with a sense of mission and commitment, understanding that this was not just a task but a testament to our capabilities and the very essence of our academic journey.

The development of this software has been nothing short of an adventure. We embarked on this project with determination and enthusiasm, navigating through numerous obstacles and learning curves. **Despite lacking extensive industrial exposure and the guidance of experienced seniors, we embraced the challenge with resilience.** Our learning journey involved leveraging resources from YouTube, Google, official documentation, and forums like Stack Overflow. We built our knowledge from the ground up, often improvising and adapting our approach to meet the project requirements.

Our team worked tirelessly to create a solution that not only meets the immediate needs of the college but also sets a standard for future projects. Although our codebase may not meet the highest professional standards, we have made every effort to ensure that it is readable and maintainable. We hope that future students and developers will find our work valuable and learn from the solutions and strategies we employed.

This project has been a testament to our growth, perseverance, and the spirit of innovation that defines our department. **It symbolizes the successful application of our learning and a meaningful contribution to our college.** Through this endeavor, we have not only addressed a critical need but also demonstrated the practical impact and relevance of our academic pursuits.

We are proud of what we have accomplished and believe that this software will serve as a valuable asset for the college, streamlining file management and enhancing efficiency. Our journey has been marked by dedication, creativity, and an unwavering commitment to excellence, embodying the true spirit of our Computer Science and Design Department.

Chapter 9

REFERENCE

- **Python Documentation:** https://docs.python.org/3/
- **SQLite Documentation:** https://sqlite.org/docs.html
- **CustomTkinter Documentation:** https://github.com/TomSchimansky/CustomTkinter
- **YouTube Videos:**
  - **TabView Widget:** https://youtu.be/df30Qro3Iu4?si=vGpQSm0PjjeFtdBO
  - **Employee Management System:** https://youtu.be/B0BOayNs4jI?si=UvuUUXXXhdYpQDpn