SYNOPSIS

No Dues Form Management System of University

PROJECT ID:12

STUDENT NAME: ADITYA BHARDWAJ
UNIVERSITY ROLLNO:2418159
SECTION: F1

1. Problem Description:

The No-Dues Clearance System is a digital platform created to simplify and automate the student clearance process after graduation. Its main goal is to ensure that students have resolved all pending obligations before leaving the institution, making the process smooth, transparent, and efficient.

This system performs a thorough check of the student's records in the following key areas:

Academic/Institutional Fees:
Verifies whether all tuition fees, exam fees, and other academic-related charges have been paid in full.

Library Dues
Confirms that the student has returned all borrowed books

Hostel Charges
Checks for any dues related to hostel room allocated

How It Works:
Once a student submits a clearance request, the system automatically gathers data from different departments.

Each department updates the student's status – marking it as "Cleared" or "Pending" based on dues.

If all areas are cleared, the system forwards the request to the admin for final approval.

If there are any pending dues, the system highlights them and the request is put on hold or rejected until all issues are resolved.

Benefits of the System:
Reduces Manual Work: No need for students to visit multiple departments physically.

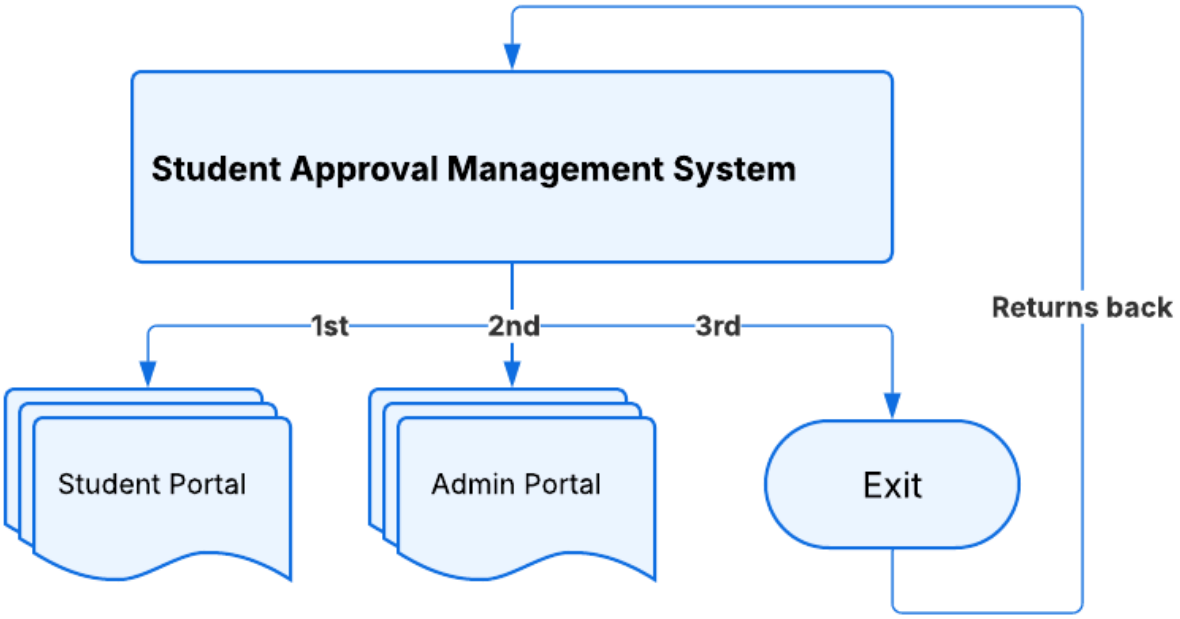
Minimizes Human Errors: Automated checks ensure accuracy and consistency in the verification process.

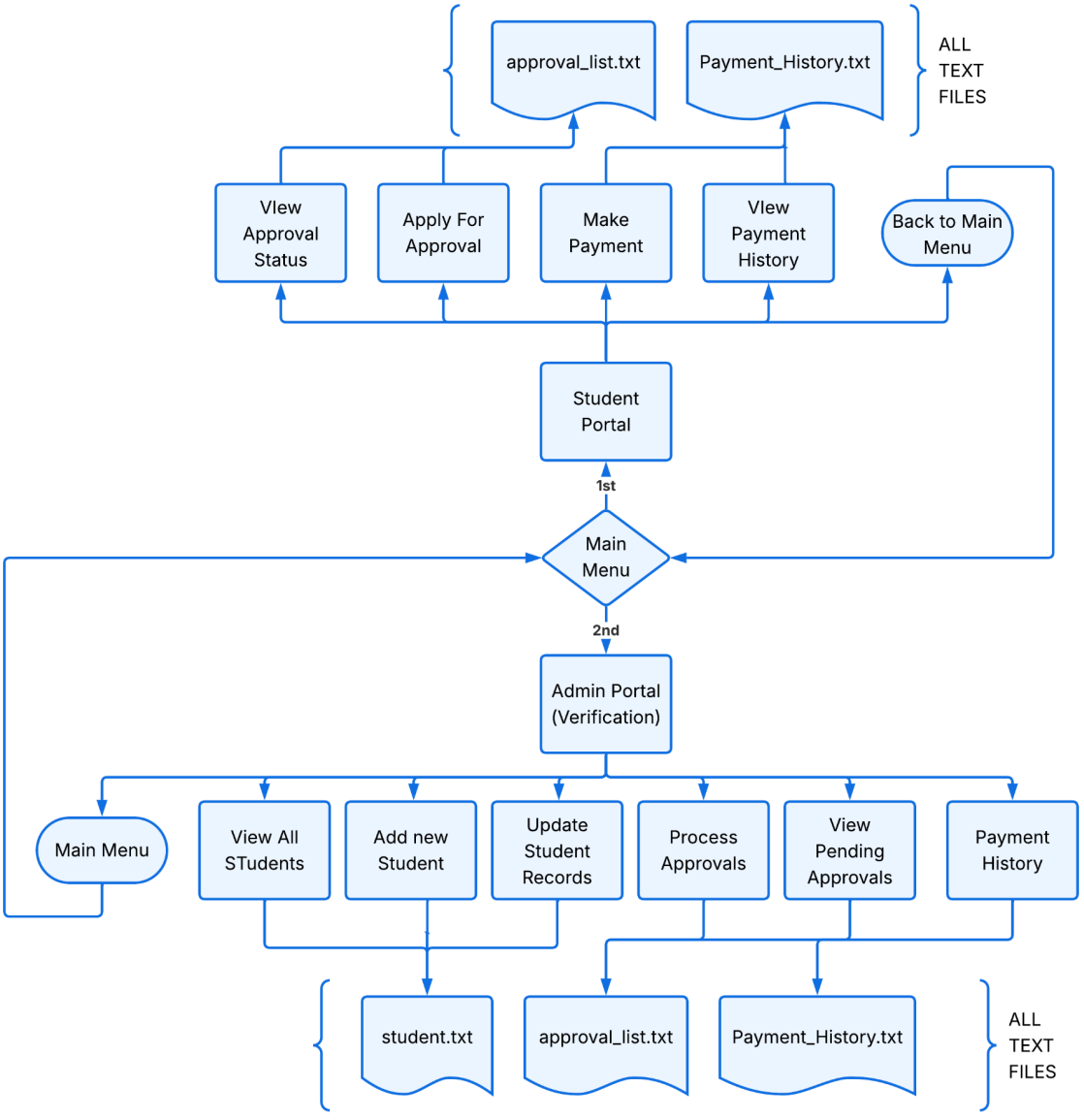
Promotes Transparency: Students can clearly see which areas are pending and what needs to be done.

Faster Process: Helps students get their clearances quickly, especially important during graduation or transfer.

Department-wise Coordination: Encourages better coordination between academic, hostel, and library staff.

2. Steps of implementation


Student Approval Management System
1st
2nd
3rd
Returns back
Student Portal
Admin Portal
Exit


approval_list.txt
Payment_History.txt
ALL TEXT FILES
VIew Approval Status
Apply For Approval
Make Payment
VIew Payment History
Back to Main Menu
Student Portal
1st
Main Menu
2nd
Admin Portal (Verification)
Main Menu
View All STudents
Add new Student
Update Student Records
Process Approvals
View Pending Approvals
Payment History
student.txt
approval_list.txt
Payment_History.txt
ALL TEXT FILES

3. Proposed modules:

- **Authentication Module**

Purpose: Handles secure login for both students and administrators
Functions:

- Student login (using roll number)
- Admin login (with username/password)
- Password masking for security
- Session management

- **Student Management Module**

Purpose: Manages all student records and data
Functions:

- Add new students to the system
- Update student information (name, roll number)
- View all student records
- Search/filter student records
- Delete student records (with confirmation)

- **Dues Management Module**

Purpose: Tracks and manages all student financial and resource obligations
Functions:

- Record fee payments/dues
- Track library book returns
- Manage hostel dues
- Automatic due calculations
- Payment receipt generation

- **Approval Request Module**

Purpose: Handles the complete approval workflow
Functions:

- Student application submission
- Request status tracking (-1, 0, 1)
- Automatic validation of eligibility
- Application history
- Duplicate request prevention

- **Approval of Processing Module**

Purpose: Administrative interface for processing requests
Functions:

- View pending requests dashboard
- Batch processing options
- Approval/Rejection with comments
- Automatic dues clearance on approval
- Audit trail of all decisions

- Reporting Module

Purpose: Generates various system reports
Functions:

- Approval status reports
- Dues summary reports
- Student clearance certificates
- Administrative analytics

3. Required Topics from the Subject

- **Strings:**
  Used extensively for data input, comparison, and display purposes. Examples include comparing passwords for login and checking student names.
- **Pointers:**
  Pointers are used to manage dynamic memory and handle file operations effectively. They also allow passing references to functions, which is crucial for struct manipulation.
- **Structures:**
  Structures help in grouping different data types together. They are useful in creating complex data records like student details, employee info, etc.
- File Handling (Creating .txt file):
  Used to store and retrieve data from external text files. It allows saving outputs or loading inputs. Example: writing user data to a .txt file using fopen(), fprintf(), and fclose().
- **String Manipulation:**
  Involves modifying or analyzing string data using functions like strcpy(), strcat(), strlen(), and strcmp(). Useful for form validation, text formatting, etc.
- **Loops:**
  Used for repeating a block of code multiple times. Types include for, while, and do-while. Useful in tasks like iterating over arrays, repeated calculations, and menu-driven programs.
- **Switch:**
  The switch-case statement is used for decision-making with multiple options. It's helpful in writing menu-based programs or handling various user choices efficiently
- **Arrays:**Arrays are used to store multiple student names, room numbers, or fee values in a structured manner. They simplify data manipulation and looping through multiple records.

4. Platform Required:

Visual Studio: The project can be executed using Visual Studio because of their user-friendly interface and debugging support.

CodeBlocks: CodeBlocks is a free, open-source, cross-platform Integrated Development Environment (IDE) primarily designed for C, C++, and Fortran programming, offering a customizable environment with a plugin architecture for extending functionality.

5. Books and Link Sources:

- Let Us C by Yashavant Kanetkar – Excellent for understanding C fundamentals and file handling.
- Computer Fundamentals and Programming – Good for structured programming and practice examples.
- GeeksforGeeks – File Handling in C – Detailed explanation and examples on file I/O.
- TutorialsPoint – C Programming – Covers all C topics with code snippets and practice exercises.