******

**Department of**

**Computer Science and Engineering**

| **DS Final Project** |
| --- |

**Course Code : CSE - 2322**

**Course Title : Data Structure Lab**

**Section : 3AM**

| **Team Name : Team Console**  **Name of Team Leader : Gazi Shaharabi Anwar Tuhin**  **Team Leader ID : C231035**  **Name of Team Member 1 : SM Muntasir Sarwar**  **Team Leader ID : C231037**  **Name of Team Member 2 : Adnan Mahmud Alvee**  **Team Leader ID : C231022** |
| --- |

**Date of Submission : 25/06/2024**

**Submitted To : Mohammed Shamsul Alam**

| **Remark** |
| --- |

### **Introduction**

The Employee Management System (EMS) is a console-based application designed to help users manage employee information efficiently. This system supports functionalities like adding, viewing, updating, deleting, and searching employee records. Additionally, it includes user authentication, allowing multiple users to maintain their own sets of employee data.

### **Features**

1. **User Authentication**:
   * Sign Up: Allows new users to create an account.
   * Log In: Allows existing users to access their accounts.
   * User credentials are stored in a users.txt file, and each user has a separate file to store employee data.
2. **Employee Data Management**:
   * Insert Data: Add new employee details.
   * Show Data: Display all employee records.
   * Search Data: Search for an employee by ID.
   * Update Data: Modify existing employee records.
   * Delete Data: Remove specific or all employee records.
3. **File Handling**:
   * Employee data is persisted in text files.
   * Utilizes file streams for reading and writing data.
4. **User Interface**:
   * Console-based interface with clear options.
   * Visual feedback through changing console colors and delay effects.

### **Implementation**

**Functions Used - Short Description**

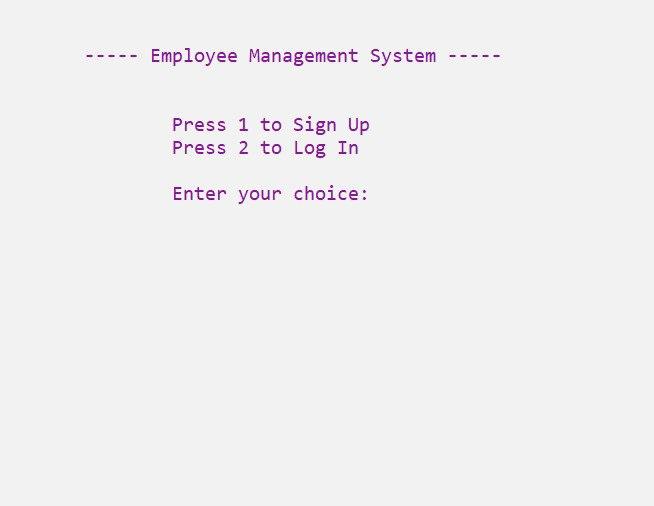
1. **insertEmp(const emp& employee)**:
   * Inserts a new employee record into the linked list.
2. **saveEmpDataToFile()**:
   * Saves all employee data from the linked list to the user's file.
3. **loadEmpDataFromFile()**:
   * Loads employee data from the user's file into the linked list.
4. **signUp()**:
   * Registers a new user by saving their credentials and creating a new file for their employee data.
5. **logIn()**:
   * Authenticates an existing user and loads their employee data.
6. **empdata()**:
   * Collects and inserts new employee data from the user.
7. **show()**:
   * Displays all employee records.
8. **Search()**:
   * Searches for an employee by ID and displays their information.
9. **update()**:
   * Updates an existing employee's information based on their ID.
10. **del()**:
    * Deletes specific or all employee records.
11. **feature()**:
    * Displays the list of features with a delay effect.
12. **sleepMilliseconds(int milliseconds)**:
    * Introduces a delay in the program execution.

### **Future Plan/Possible Extension**

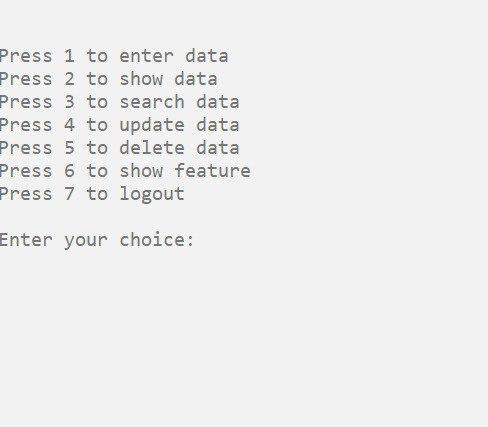
1. **Enhanced Error Handling**:
   * Implement comprehensive error handling to cover more edge cases and provide better user feedback.
2. **Input Validation**:
   * Ensure that inputs like employee IDs are unique and follow a specific format.
3. **Graphical User Interface (GUI)**:
   * The transition from a console-based interface to a GUI for a better user experience.
4. **Database Integration**:
   * Integrate with a database system like MySQL or SQLite for more efficient data management and retrieval.
5. **Concurrent Access Handling**:
   * Allow multiple users to access and modify data simultaneously with proper synchronization.
6. **Reporting and Analytics**:
   * Add features to generate reports and perform analytics on employee data.
7. **Data Encryption**:
   * Encrypt user credentials and employee data files to enhance security.

**Screen Shorts :**

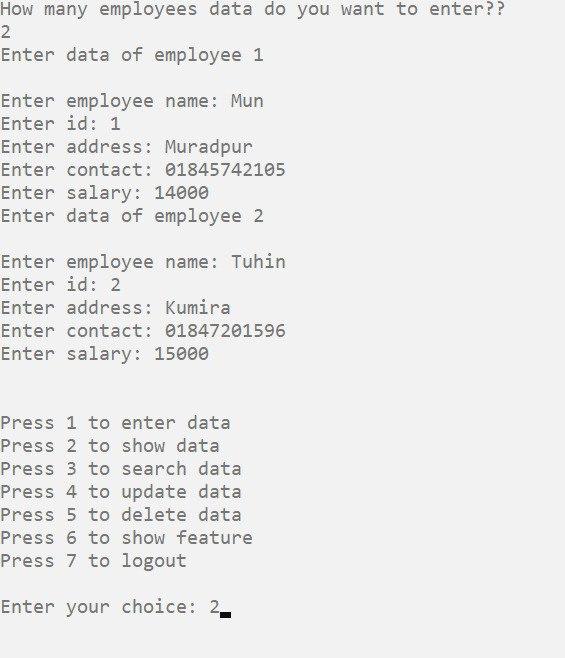
**Welcome Page :**

****

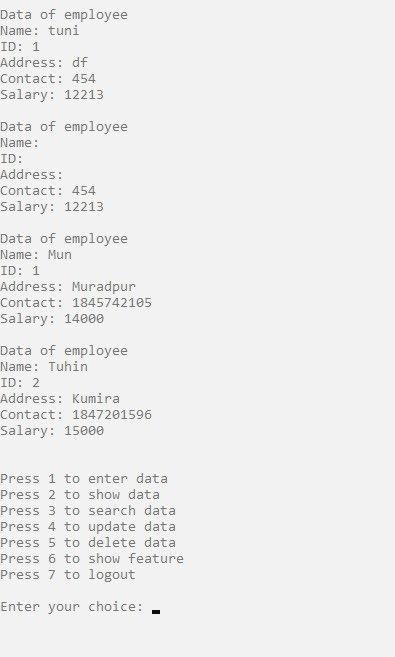
**Menu Page :**

****

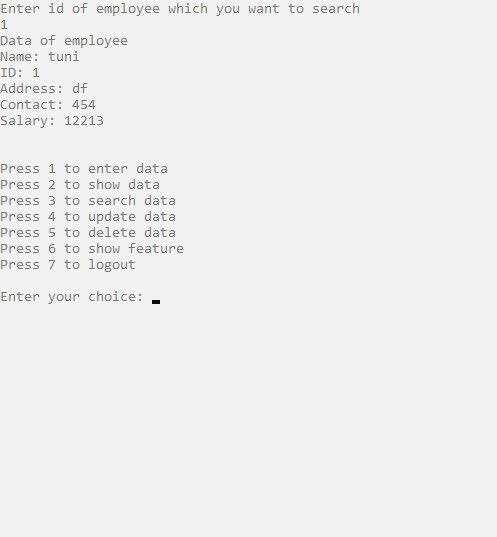
**Enter data page :**

****

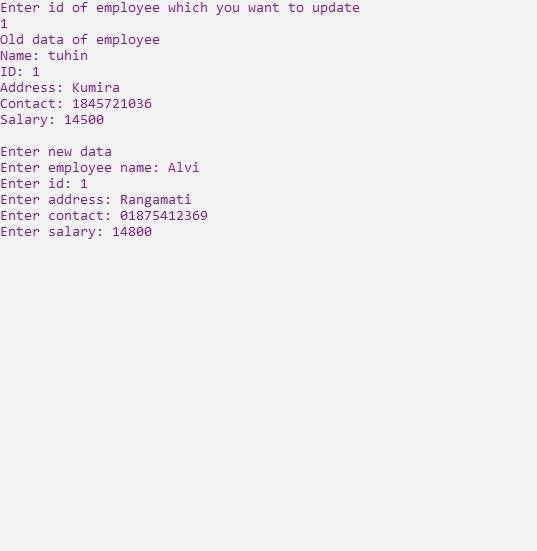
**Show data page :**

****

**Search data page :**

****

**Update data page :**

****

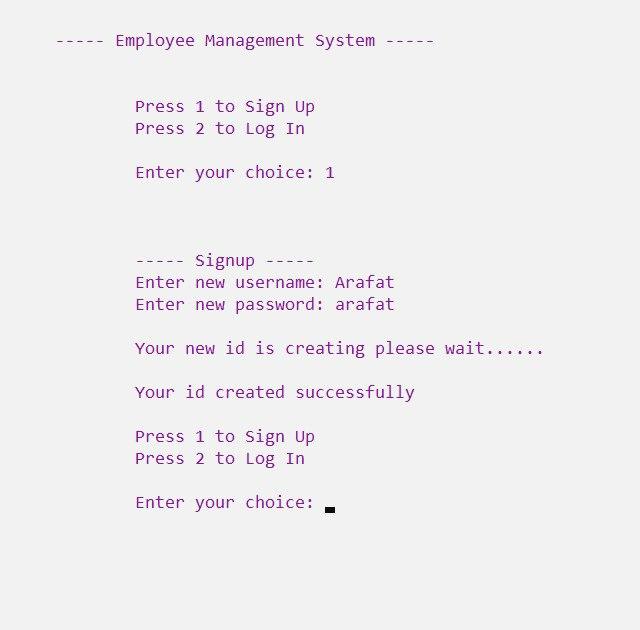
### **Delete data page :**

### 

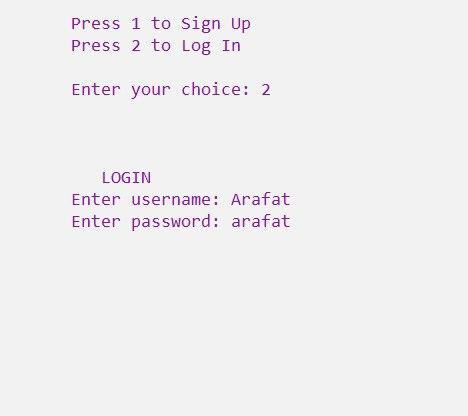
### **Feature page :**

### 

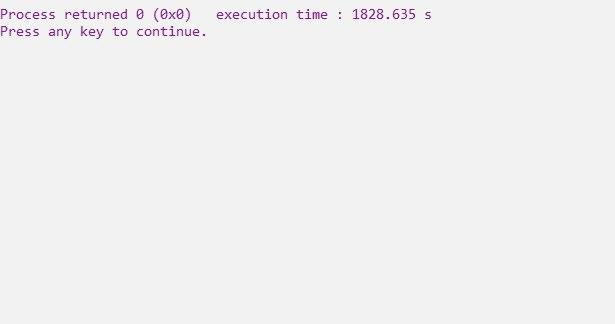
### **Sign-up page :**



### **Login with new data :**



### **Log out page :**



### **Conclusion :**

The Employee Management System provides essential functionalities for managing employee data through a console-based interface. It supports multiple users with individual data management and ensures data persistence through file handling. The future enhancements aim to improve user experience, security, and scalability. The current implementation serves as a robust foundation for further development and expansion.