1. Proposal

**1. Introduction/Overview of Mini Application/System**

*Problem Statement:* Managing employee records manually is time-consuming and prone to errors. An automated Employee Record System simplifies this process, ensuring accurate and efficient record-keeping.

*Objectives:*

* Develop an automated system to maintain and manage employee records.
* Provide quick access to employee information.
* Ensure data accuracy and integrity.

*Target Users:*

* HR departments in small to medium-sized organizations.
* Managers who need quick access to employee information.

*Scope of Application:* The system will cover the following functionalities:

* Adding, updating, and deleting employee records.
* Viewing employee details.
* Searching for employees by various criteria (e.g., name, ID, department).

*Development Methodology:* We will use the Waterfall model, starting with requirement analysis, followed by system design, implementation, testing, and maintenance.

*Milestones:*

* Requirement Analysis: 1 week
* System Design: 2 weeks
* Implementation: 4 weeks
* Testing: 2 weeks
* Deployment: 1 week

**2. Analysis**

*Choice of Tool:* The C programming language will be used for developing the system, due to its efficiency and widespread use in system-level programming.

*Feasibility Study:*

* **Technical Feasibility:** The tools required (C compiler, development environment) are readily available.
* **Operational Feasibility:** The system will streamline the HR process, reducing manual effort and errors.
* **Economic Feasibility:** The development cost is low, as it mainly involves programming resources and minimal hardware requirements.

**3. Design**

*Flowchart Explanation:* The system flowchart will depict the sequence of operations, including adding, updating, viewing, and deleting employee records. It will also show the validation checks for input data.

*Development Methodology:* We will follow the structured programming approach, breaking down the system into manageable modules and functions.

**4. Conclusion and Recommendations**

*Potential Contribution of Animation:* While animations are not necessary for this system, simple visual indicators (e.g., loading animations) can enhance the user experience.

*Limitations:*

* Limited to small to medium-sized organizations due to data handling capacity.
* No built-in backup and recovery system.

*Future Enhancements:*

* Implementing a database for larger data handling.
* Adding role-based access control for improved security.
* Integrating with other HR systems.

**5. References**

* [Your reference materials, books, websites, etc.]

**6. Appendix**

* [Any additional information, diagrams, etc.]

1. User Function

list of potential user-defined functions for the Employee Record System:

1. **void addEmployee()**: This function will prompt the user to input new employee details and add the record to the system.
2. **void updateEmployee()**: This function will allow the user to update existing employee details based on the employee ID.
3. **void deleteEmployee()**: This function will enable the user to delete an employee record from the system based on the employee ID.
4. **void viewEmployee()**: This function will display the details of a specific employee based on the employee ID.
5. **void listEmployees()**: This function will list all the employees in the system.
6. **int searchEmployee()**: This function will search for an employee based on a given criterion (e.g., name, ID, department) and return the index or ID if found.
7. **void saveToFile()**: This function will save the current employee records to a file.
8. **void loadFromFile()**: This function will load employee records from a file at the start of the program.
9. **void displayMenu()**: This function will display the main menu and handle user input to navigate the system.
10. **int validateInput()**: This function will handle input validation to ensure correct data entry for employee records.

These functions cover the main operations for managing employee records in the system, ensuring a modular and structured approach to programming.

1. Storyboard

**Features List**

**Main Menu**

* **Add Employee**: Allows the user to input new employee details and add the record to the system.
* **Update Employee**: Enables the user to update existing employee details based on the employee ID.
* **Delete Employee**: Allows the user to delete an employee record from the system based on the employee ID.
* **View Employee**: Displays the details of a specific employee based on the employee ID.
* **List Employees**: Lists all the employees in the system.
* **Search Employee**: Searches for an employee based on various criteria (e.g., name, ID, department).
* **Save to File**: Saves the current employee records to a file.
* **Load from File**: Loads employee records from a file at the start of the program.
* **Exit**: Exits the application.

**Input Validation**

* Ensures all required fields (e.g., Name, ID, Department, Position) are filled in correctly.
* Checks for duplicate employee IDs.

**User Interface**

* Simple and intuitive text-based interface for ease of use.

**Error Handling**

* Displays appropriate error messages for invalid inputs or actions.

1. Pseudocode & flowchart