./

Learning Report – Applied System Development Life Cycle and Software Testing

**EMPLOYEE MANAGEMENT SYSTEM**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **To be approved By** | **Remarks/Revision Details** |
| 1 |  | Name/PS No | Name/PS No | Module Owner Name | Comments |
| 2 | 25/05/21 | 99004428 | 99004429 | Thiruppathi Seenivasan |  |
| 3 | 25/05/21 | 99004429 | 99004430 | Thiruppathi Seenivasan |  |
| 4 | 25/05/21 | 99004430 | 99004431 | Thiruppathi Seenivasan |  |
| 5 | 25/05/21 | 99004431 | 99004432 | Thiruppathi Seenivasan |  |
| 6 | 25/05/21 | 99004432 | 99004428 | Thiruppathi Seenivasan |  |

**Document History**

Table of Contents

[Table of Figures 3](#_Toc53129062)

[Table of Tables 4](#_Toc53129063)

[ACTIVITY 1: SYSTEM/ SOFTWARE DEVELOPMENT 4](#_Toc53129064)

[**INTRODUCTION** 4](#_Toc53129065)

[**MY PRODUCT: “EMPLOYEE MANAGEMENT SYSTEM ”** 6](#_Toc53129066)

[**SWOT ANALYSIS** 7](#_Toc53129067)

[**REQUIREMENTS** 7](#_Toc53129068)

[**DESIGN** 8](#_Toc53129069)

[HIGH LEVEL DESIGN 8](#_Toc53129070)

[LOW LEVEL DESIGN 11](#_Toc53129071)

[**TEST PLANS** 13](#_Toc53129072)

[**REFERENCES** 15](#_Toc53129073)

[ACTIVITY 2: AGILE METHODOLOGY 15](#_Toc53129074)

[**THEME** 15](#_Toc53129075)

[**EPIC** 15](#_Toc53129076)

[**USER STORY** 16](#_Toc53129077)

[**REFERENCES** 17](#_Toc53129078)

[APPENDIX: 17](#_Toc53129079)

## Table of Figures

[Figure 1 CLASS DIAGRAM(HIGH LEVEL) 10](#_Toc52177314)

[Figure 2 USE CASE DIAGRAM (HIGH LEVEL) 11](#_Toc52177315)

[Figure 3 ACTIVITY DIAGRAM (HIGH LEVEL) 12](#_Toc52177316)

[Figure 4 USE CASE DIAGRAM (LOW LEVEL) 12](#_Toc52177317)

[Figure 5 ACTVITY DIAGRAM (LOW LEVEL) 13](#_Toc52177318)

[Figure 6 BLOCK DIAGRAM 13](#_Toc52177319)

[Figure 7 COMPONENT DIAGRAM (HIGH LEVEL) 22](#_Toc52177320)

[Figure 8 ACTIVITY DIAGRAM (high level) 23](#_Toc52177321)

[Figure 9 ACTIVITY DIAGRAM (LOW LEVEL) 24](https://lnttsgroup.sharepoint.com/sites/GEA/Global%20Engineering%20Academy/GEA%20Insights/Genesis/Shared%20Documents/Submission/MYSORE/2009MYSEMB/Foundation/Applied%20SDLC%20with%20Software%20Testing/99002439/FINAL.docx#_Toc52177322)

[Figure 10- ACTIVITY DIAGRAM (LOW LEVEL) 24](#_Toc52177323)

[Figure 11 TEST PLAN 25](#_Toc52177324)

[Figure 12 GIT 27](#_Toc52177325)

[Figure 13 GIT ISSUES 28](#_Toc52177326)

[Figure 14 GIT COMMITS 1 28](#_Toc52177327)

[Figure 15 GIT COMMIT 2 29](#_Toc52177328)

[Figure 16 GIT 30](#_Toc52177329)

[Figure 17 GIT MAKE 31](#_Toc52177330)

[Figure 18 GIT MAKE 2 31](#_Toc52177331)

[Figure 19 GIT BUILD 32](#_Toc52177332)

[Figure 20 GIT CODE QUALITY 32](#_Toc52177333)

## Table of Tables

[Table 1 AGING 6](#_Toc52177304)

[Table 2 GRADING COST 6](#_Toc52177305)

[Table 3 REQUIREMENTS 8](#_Toc52177306)

[Table 4 HIGH LEVEL TEST PLAN 15](#_Toc52177307)

[Table 5 LOW LEVEL TEST PLAN 16](#_Toc52177308)

[Table 6 USER STORIES 17](#_Toc52177309)

[Table 7 AGING 19](#_Toc52177310)

[Table 8 GRADING COST 19](#_Toc52177311)

[Table 9 REQUIREMENTS 21](#_Toc52177312)

[Table 10 USER STORIES 27](#_Toc52177313)

**ACTIVITY 1: SYSTEM/ SOFTWARE DEVELOPMENT**

**INTRODUCTION**

* Employee management system is an application, developed to help owners of an organization to manage their employees more easily and efficiently. It helps to eliminate the manual process and saves a lot of time and money. This application has several services such as Add/Delete employee data in the record, search, modify and display the data of employee in the record, checking the attendance and number of leaves.
* Conventionally this system is done using papers, file and blinders. This Employee management system saves the time of an organization.
* To implement this project, we require C Programming, make file and Unit Testing using Unity.
* Basic knowledge of structures and functions is required.

**SCOPE**

Without an Employee management system, managing and maintaining the details of an employee is a tedious job for any organization. It will store the details of an employee including their background details, personal information, educational qualifications and other details.

**BENEFITS**

* It is easy to fetch and update data whenever required which will reduce the human work force to maintain them.
* Reduce the traditional File Management Systems.
* Analysis, Adding, Updating and Searching becomes easier.

**MY PRODUCT: EMPLOYEE MANAGEMENT SYSTEM**

**FUNCTIONAL REQUIREMENTS CONSTRAINTS**

Employee management system have some features like-

* Add/Delete of employee record.
* Search for an employee.
* Update employee record.
* Attendance/ Leaves of an employee.
* Display employee record.

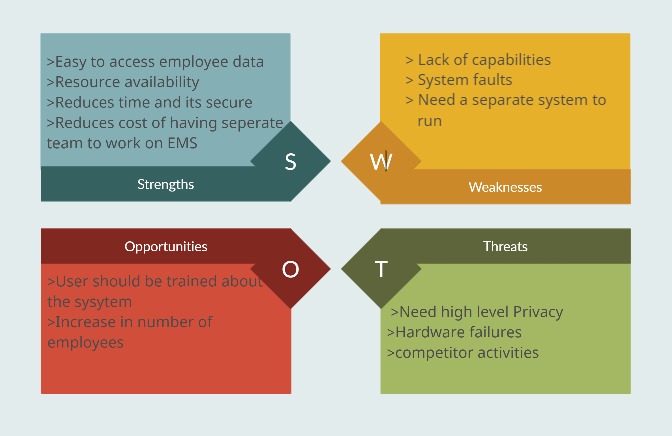
**CONSTRAINTS**

* Every user must be comfortable using computer.
* All operations are in english so user must have basic knowledge of English.

**AGING**

|  |  |
| --- | --- |
| **AGE** | **FEATURE** |
| Olden Days | This system is done using papers , file, record and blinders to store the details of an employee manually. |
| At present | This system is developed using well-designed data base to store the details of an employee. |

**SWOT ANALYSIS**

****

**REQUIREMENTS**

**HIGH LEVEL REQUIREMENTS -**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **DESCRIPTION** | **CATEGORY** | **STATUS** |
| HR01 | New account creation for new employee's (fresher and experienced candidates) and filling the necessary data as well as deleting details if needed. | **Technical** | **IMPLEMENTED** |
| HR02 | To update employee's required information. | **Technical** | **IMPLEMENTED** |
| HR03 | To find the Employees required information. | **Technical** | **IMPLEMENTED** |
| HR04 | To find the list of the employees from Business Units, Roles and Joining Date. | **Technical** | **IMPLEMENTED** |
| HR05 | Employee should be able to view their attendance and leaves report. | **Technical** | **IMPLEMENTED** |
| HR06 | To display the details of the existing employees. | **Technical** | **IMPLEMENTED** |

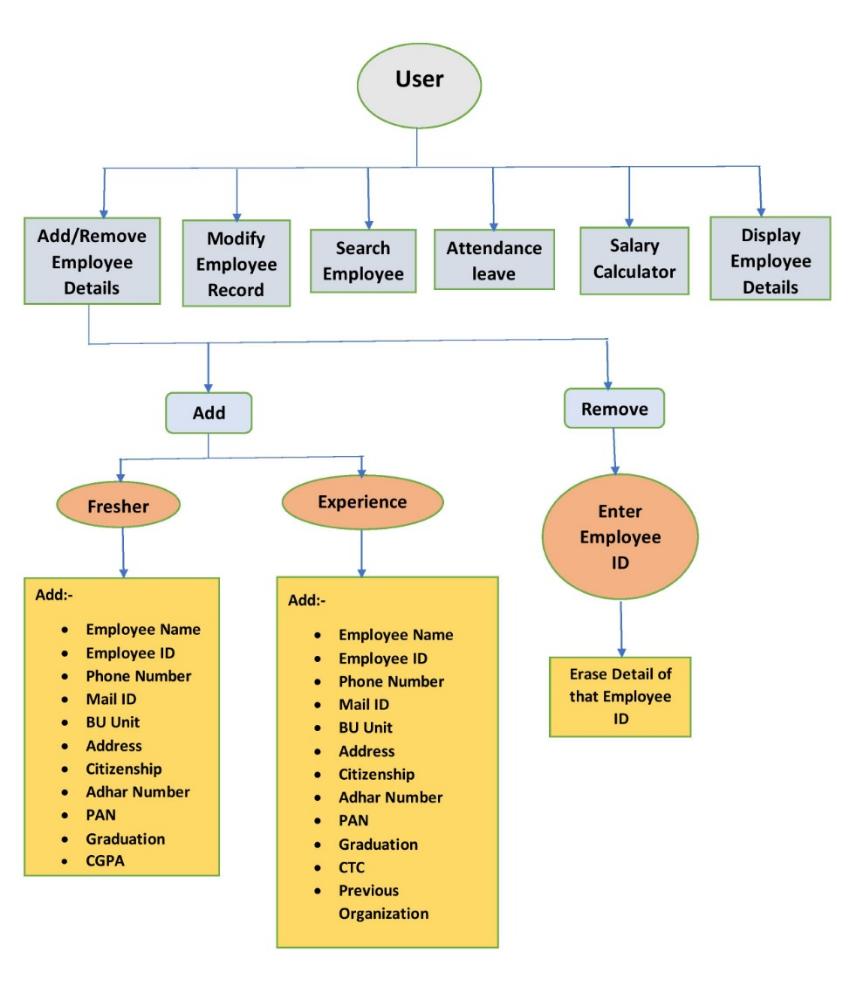
**LOW LEVEL REQUIREMENTS -**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **DESCRIPTION** | **HLR**  **ID** | **STATUS**  **(IMPLEMENTED**  **/FUTURE)** |
| LR01 | 1. User should choose whether the employee is a fresher or an experienced employee. 2. After choosing the status of the employee then his/her personal details can be added (like name, mail id, employee id, Aadhaar, PAN, etc.) 3. User can remove his/her details if needed. | **HR01** | **IMPLEMENTED** |
| LR02 | (1) User can update the specific information By searching employee id, if ID not found then "No Record Found" message should get displayed.  (2) If ID found then Update the required information. | **HR02** | **IMPLEMENTED** |
| LR03 | 1. User should be able to enter the BU to get the list of employees in that BU. 2. User should be able to enter the roles to get the list of employees in that role. 3. User should be able to enter the joining date to get the list of employees who joined on that date. | **HR03** | **IMPLEMENTED** |
| LR04 | 1. User should be able to enter the employee id to get the attendance report. 2. User should be able to enter the employee to get the no. of. leaves of an employee. | **HR04** | **IMPLEMENTED** |
| LR05 | 1. User should be able to enter the employee id to calculate the salary. 2. User should be able to calculate gratuity.   (3) User should be able to calculate PF.  (4) User should be able to calculate Tax. | **HR05** | **IMPLEMENTED** |
| LR06 | (1) User shall be able to enter the employee name and employee id.  (2) User will be able to view employee details. | **HR06** | **IMPLEMENTED** |

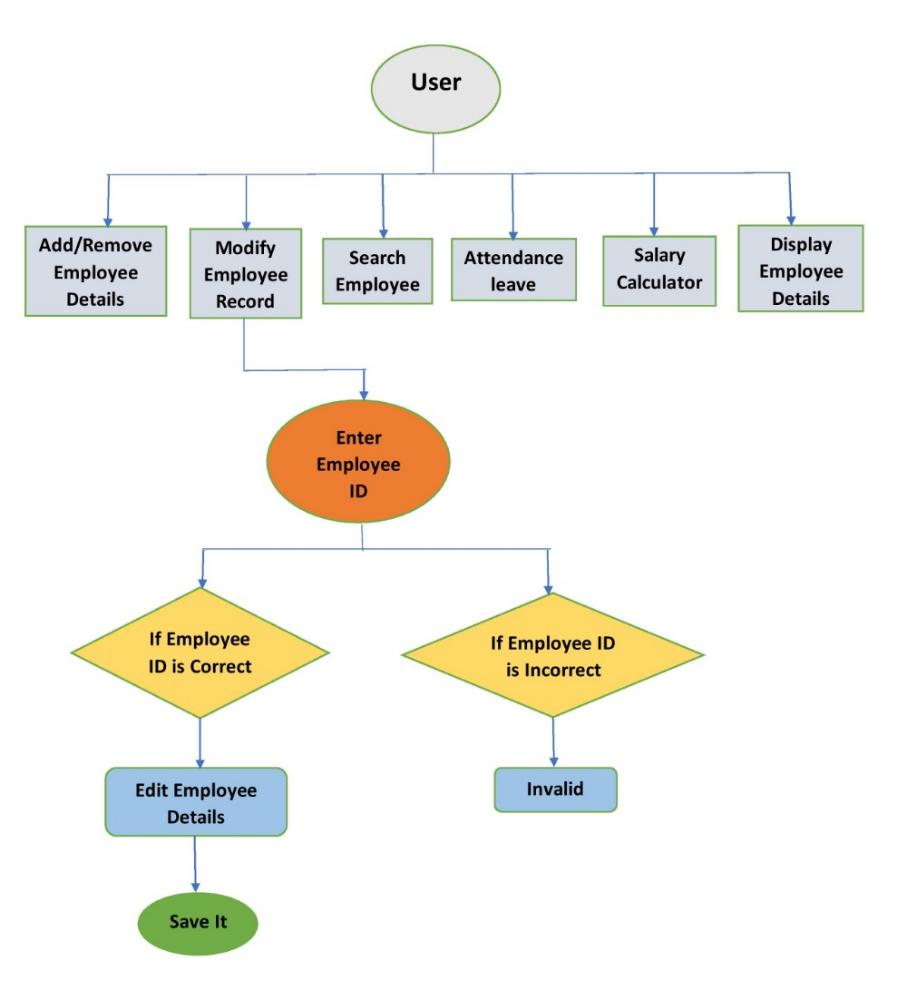
**DESIGN**

**HIGH LEVEL DESIGNS**

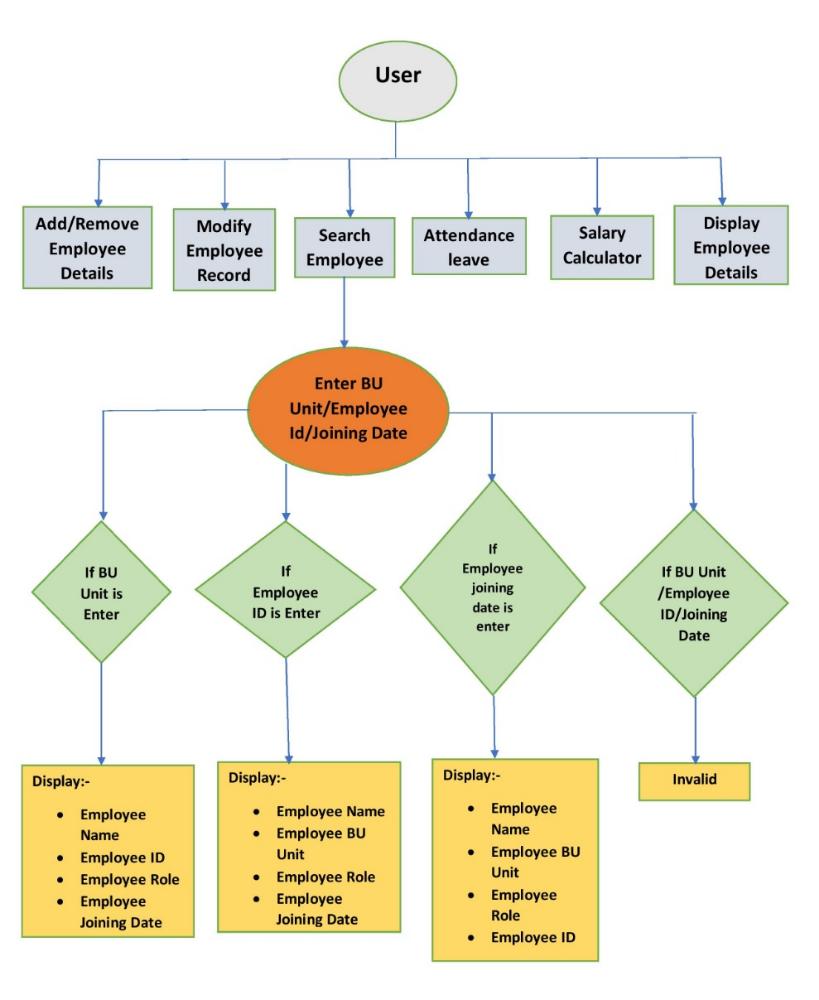
**FOR ADD/REMOVE –**



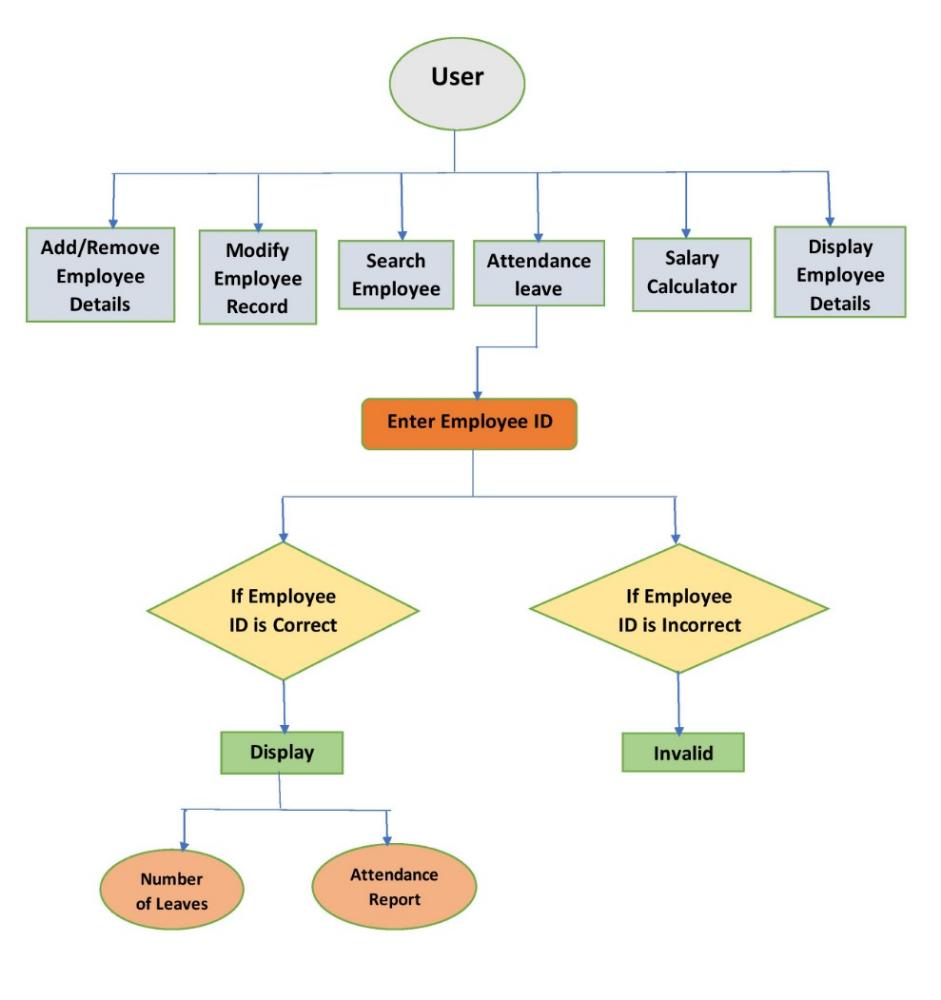
**FOR MODIFY –**



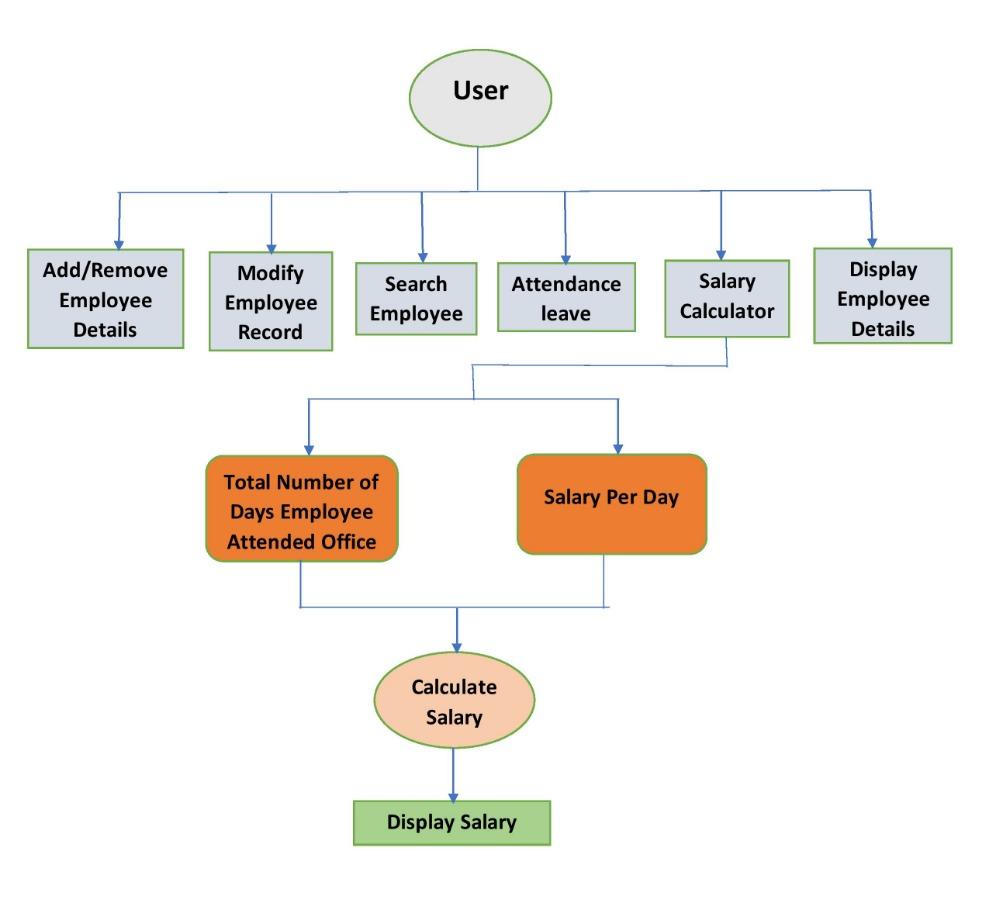
**FOR SEARCH –**



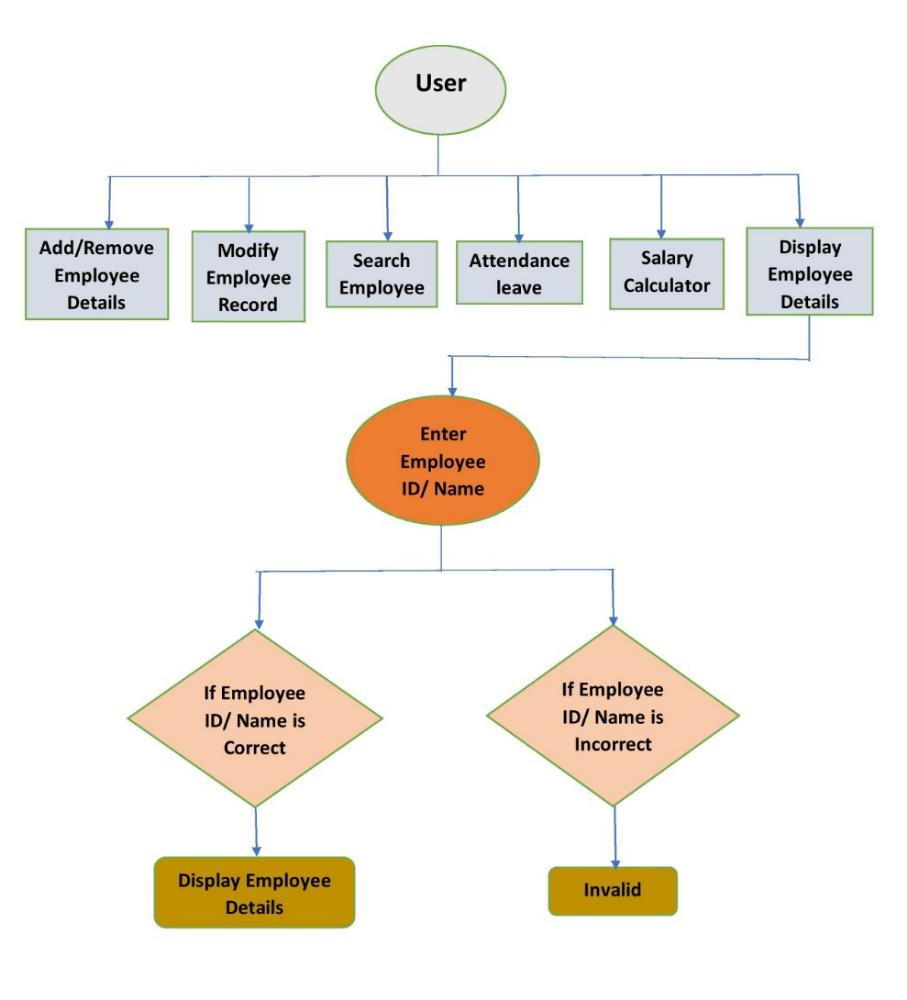
**FOR ATTENDANCE/LEAVE –**



**FOR SALARYCALCULATOR –**



**FOR DISPLAY –**

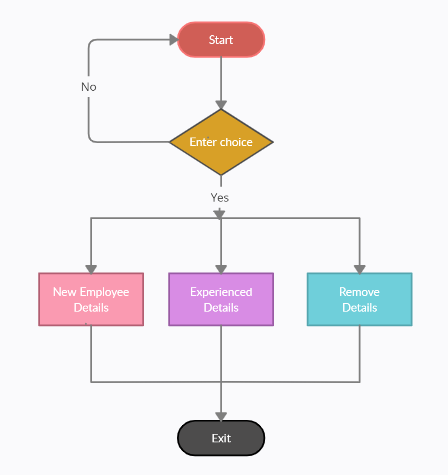


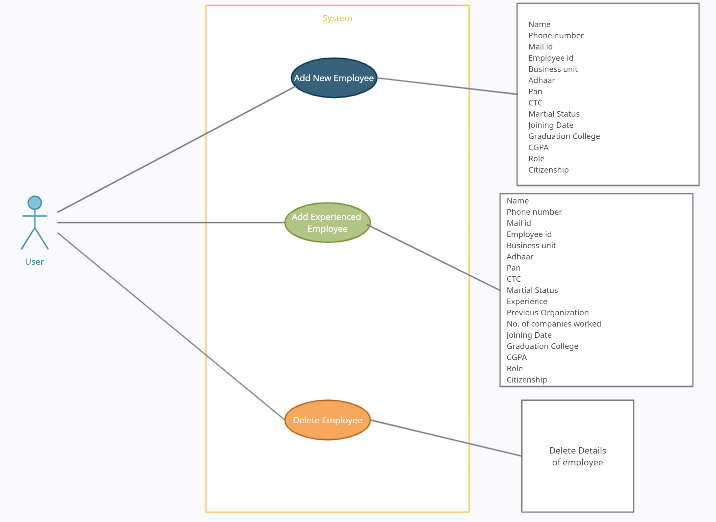
**LOW LEVEL DESIGNS**

**FOR ADD/REMOVE –**

**BEHAVIOURAL DIAGRAMS:**

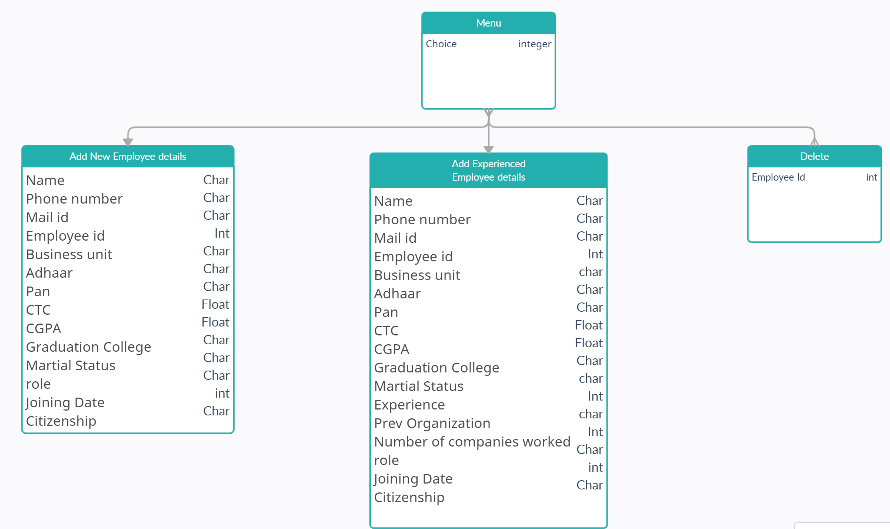
**FLOWCHART**

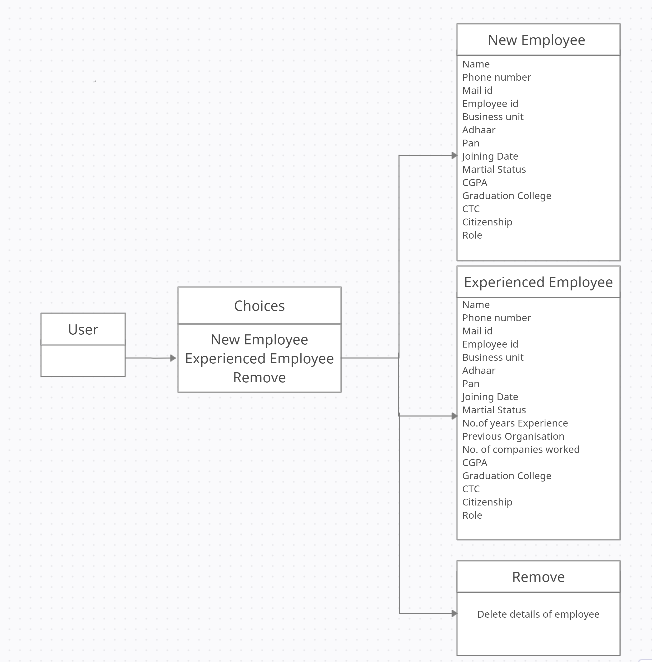


**USECASE** 

**STRUCTURAL DIAGRAMS:**

**COMPOSITE**

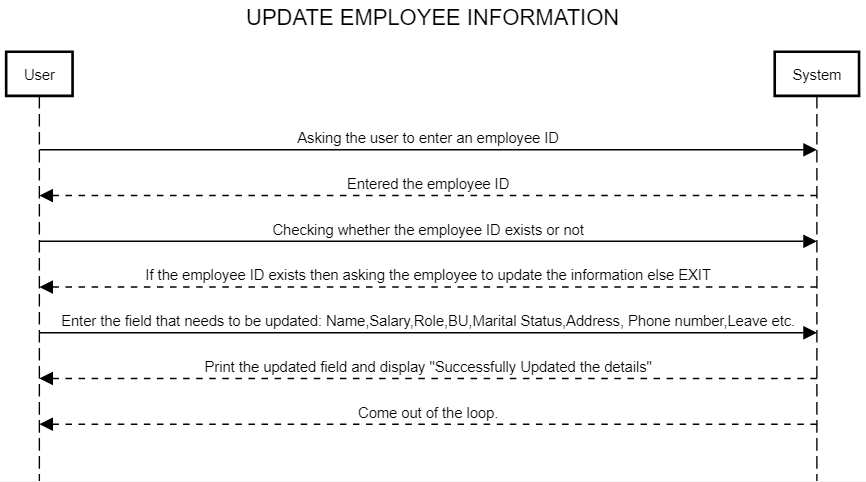


**OBJECT**

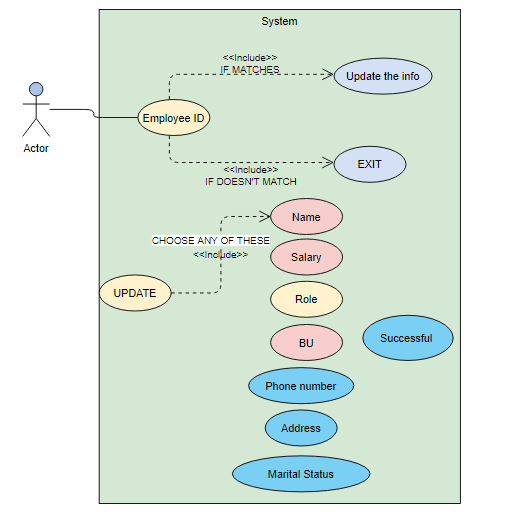
**FOR UPDATE –**

**BEHAVIOURAL DIAGRAMS:**

**SEQUENCE**

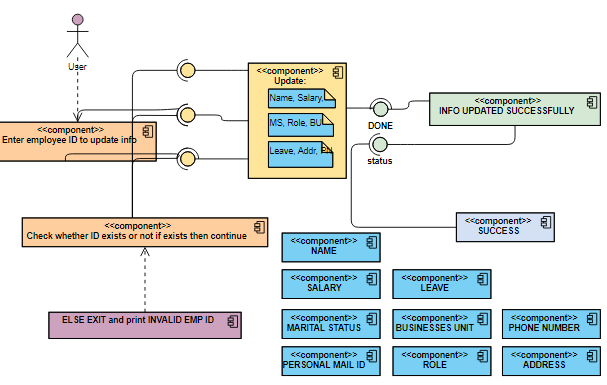


**USECASE**

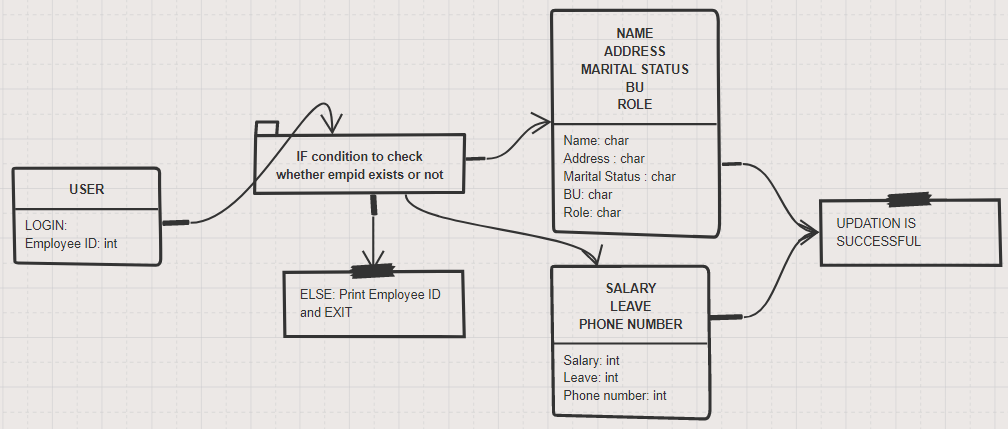


**STRUCTURAL DIAGRAMS:**

**COMPONENT**



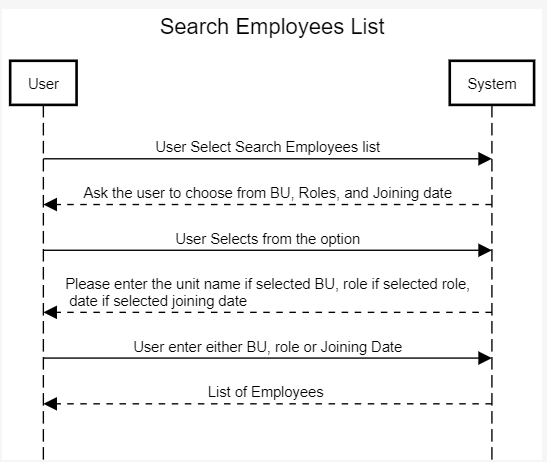
**PROFILE**



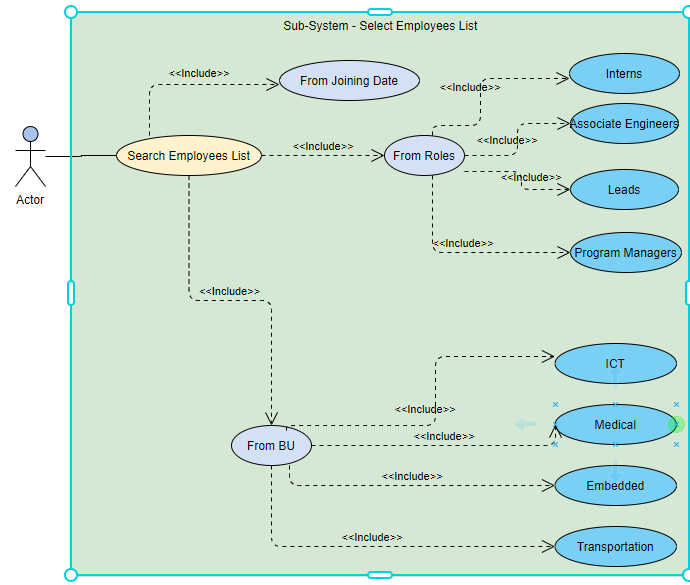
**FOR SEARCH –**

**BEHAVIOURAL DIAGRAMS:**

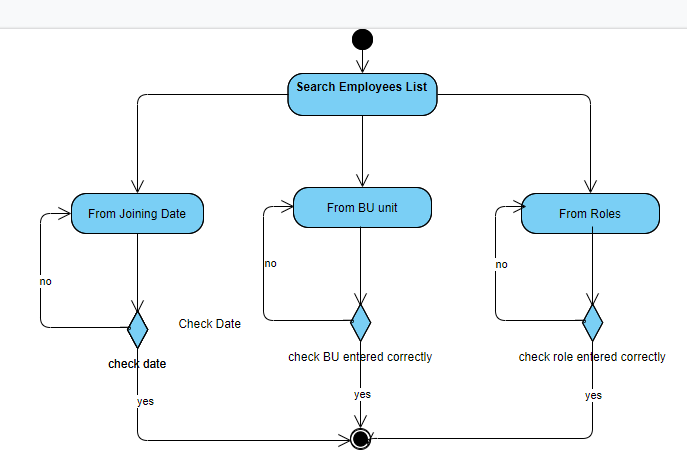
**SEQUENCE**



**USECASE**



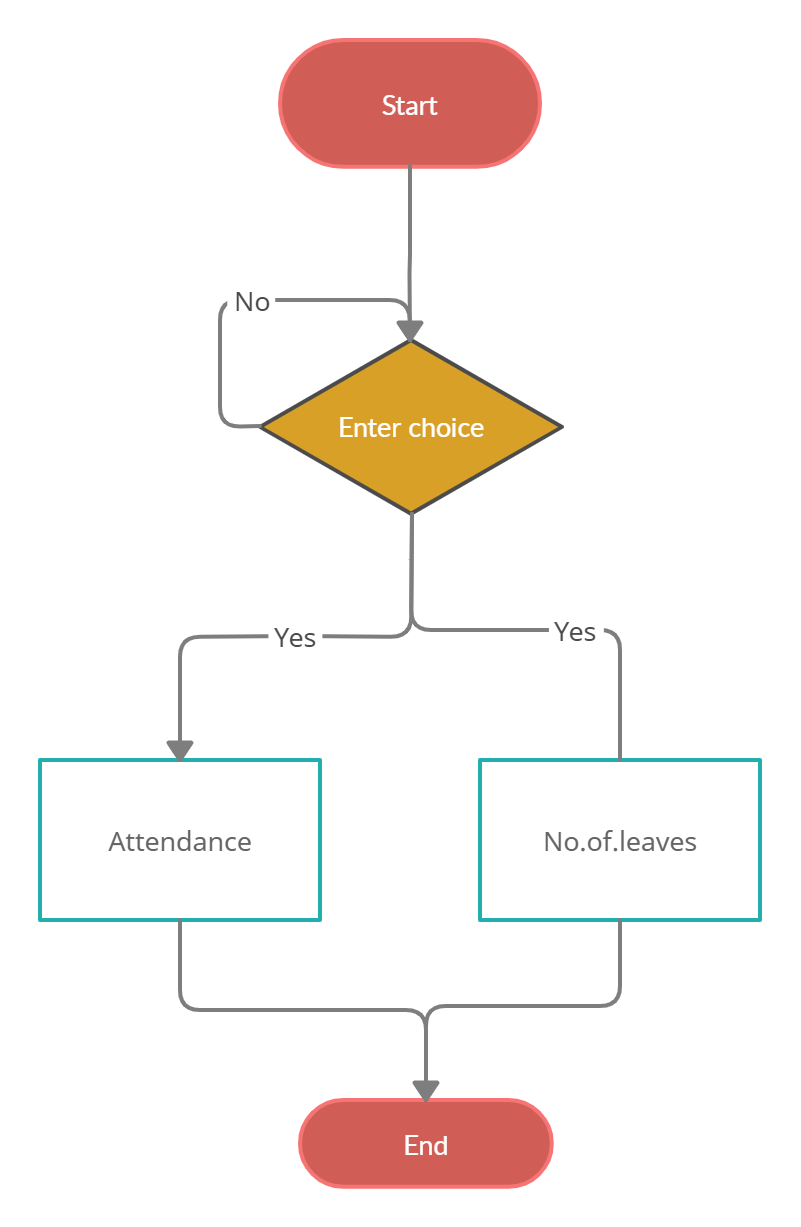
**ACTIVITY**



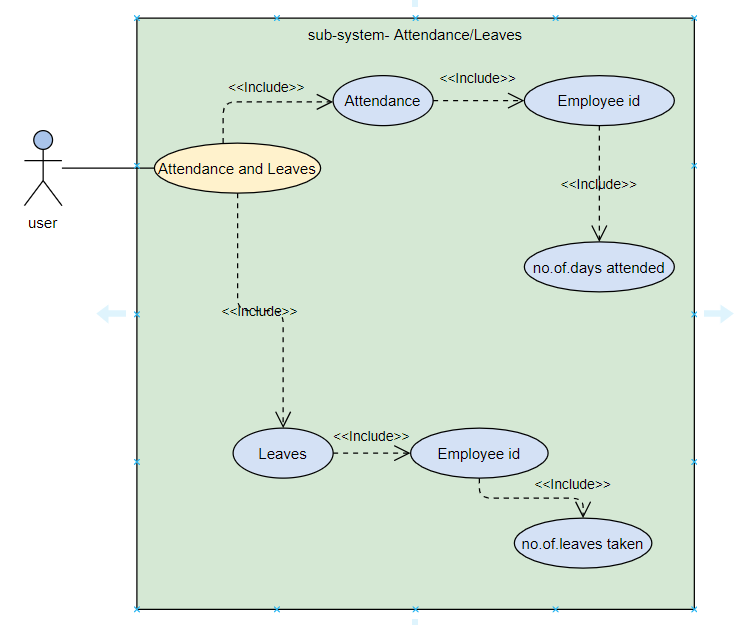
**FOR ATTENDANCE/LEAVE –**

**BEHAVIOURAL DIAGRAMS:**

**FLOW CHART**

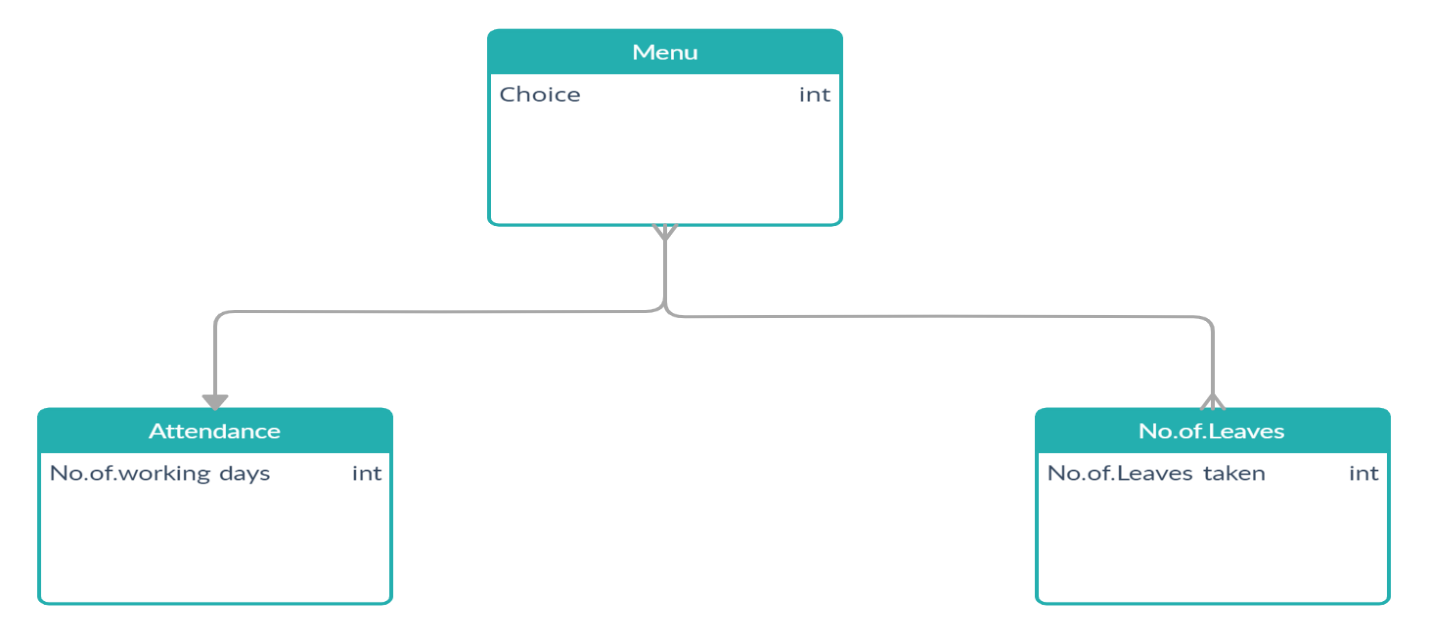


**USECASE**



**STRUCTURAL DIAGRAMS:**

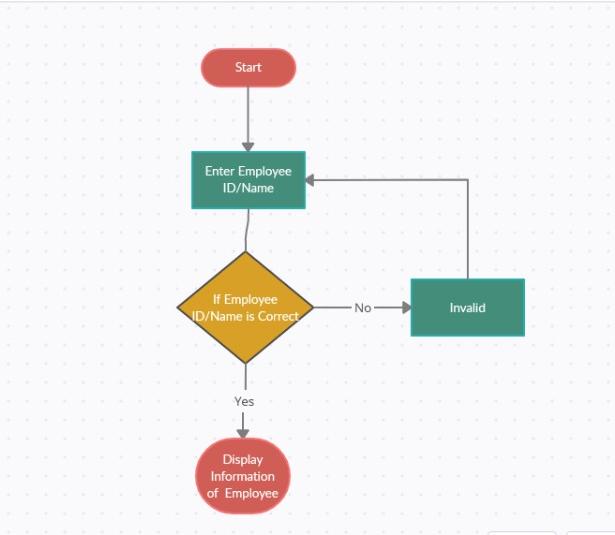
**COMPONENT**



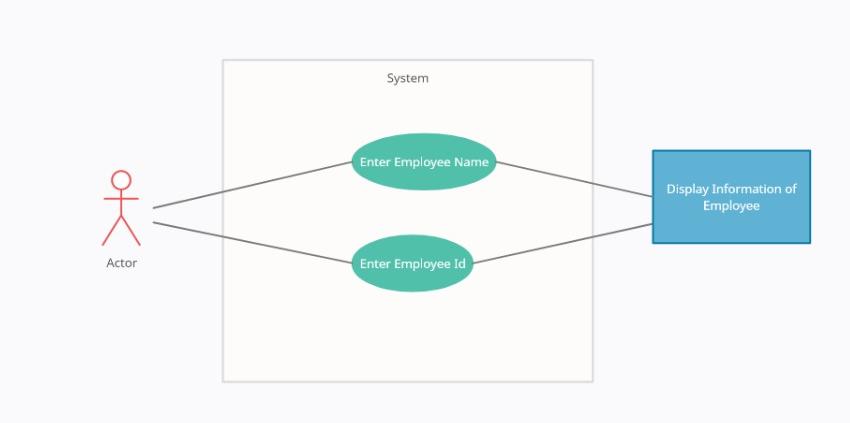
**FOR DISPLAY –**

**BEHAVIOURAL DIAGRAMS:**

**FLOWCHART**

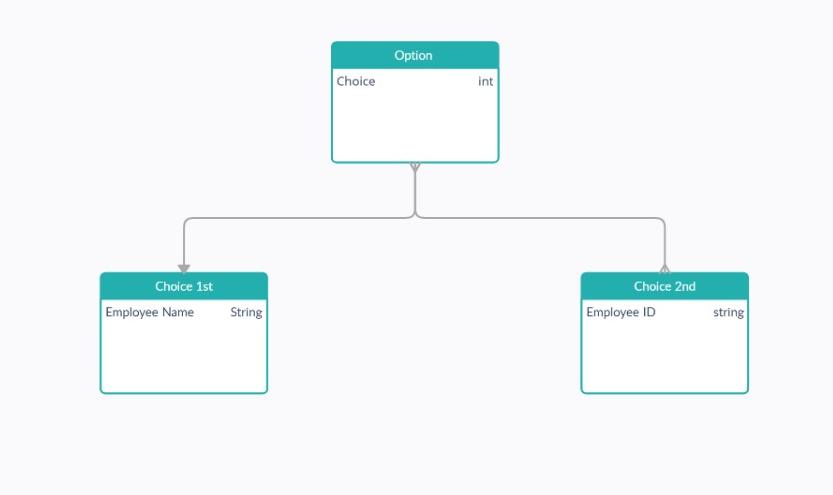


**USECASE**



**STRUCTURAL DIAGRAMS:**

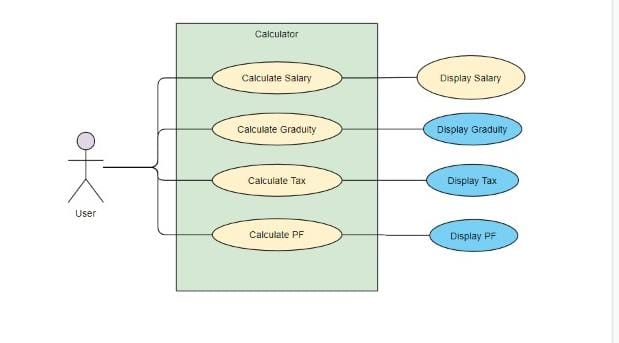
**COMPONENT**

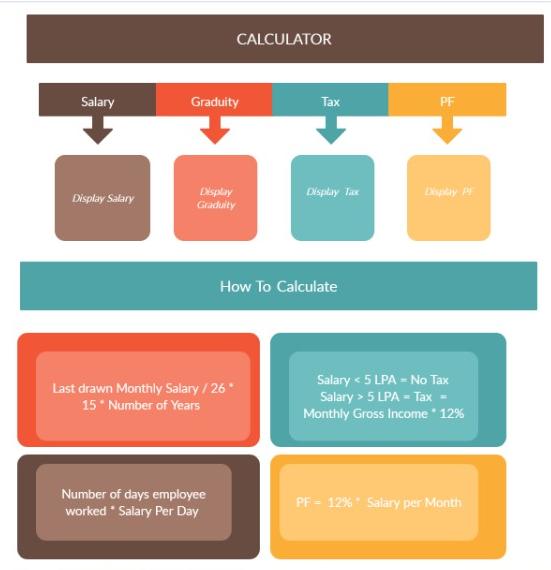


**FOR SALARY CALCULATOR –**

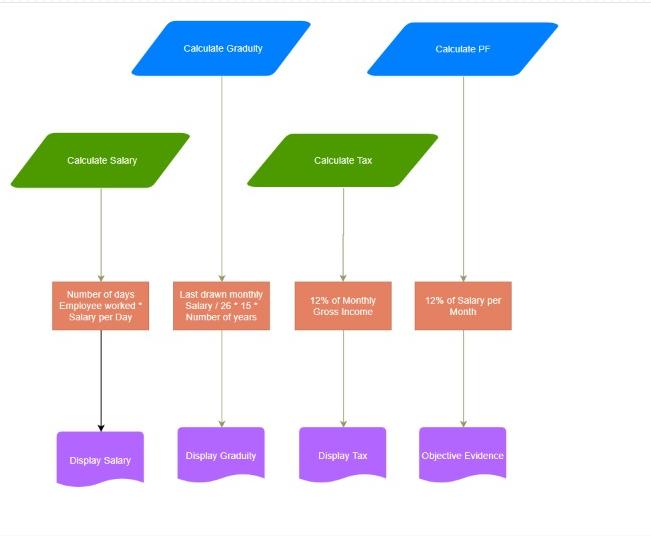
**BEHAVIOURAL DIAGRAMS:**

**USECASE**





**ACTIVITY DIAGRAM**



**BLOCK DIAGRAM**



**TEST PLAN**

**HIGH LEVEL TEST PLAN:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TEST ID** | **DESCRIPTION** | **EXP I/P** | **EXP O/P** | **ACTUAL O/P** | **TYPE OF TEST** |
| HL\_01 | Check if the user has entered correct details | (1) name (2) employee id (3) BU (4) email (5) mobile no | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT**  **BASED** |
| HL\_02 | Check if the user details have been removed or not | employee id | Calls the function and check the input fields is removed or not | **SUCCESS** | **REQUIREMENT**  **BASED** |
| HL\_03 | Check if the user details entered in modification is valid or not | (1) Employee ID (2) Search function (3) Field to be modified | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT**  **BASED** |
| HL\_04 | Check modification of information | (1) Employee id (2) Field to be modified | Calls the function and check the input field is updated or not | **SUCCESS** | **REQUIREMENT**  **BASED** |
| HL\_05 | Check whether roles, BU, and joining date exist or not | (1) Employee ID (2) BU (3) Joining Date | Calls the function and check the input field is correct or not | **SUCCESS** | **REQUIREMENT**  **BASED** |
| HL\_06 | Check attendance and leave whether they have entered the correct employee id or no | (1) Employee ID | Calls the function and check the input field is correct or not | **SUCCESS** | **TECHNICAL** |
| HL\_07 | Check if the name or unique id is present in our employee list or not | (1) Employee ID (2) Name | Calls the function and check the input field is correct or not | **SUCCESS** | **REQUIREMENT BASED** |
| HL\_08 | Check if salary calculator, gratutity calculator, PF calculator and tax calculator are working correctly or not | (1) Monthly salary | Correct salary, gratutiy, PF and test values | **SUCCESS** | **TECHNICAL** |

**LOW LEVEL TEST PLAN:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TEST ID** | **HL\_ID** | **DESCRIPTION** | **EXP I/P** | **EXP O/P** | **ACTUAL O/P** | **TYPE OF TEST** |
| LL\_01 | HL\_01, HL\_07 | Check if the user has entered correct name | Name | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_02 | HL\_01, HL\_06, HL\_07 | Check if the user has entered correct employee id | Employee id | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_03 | HL\_01 | Check if the user has entered correct BU | BU | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_04 | HL\_01 | Check if the user has entered correct email | Email | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_05 | HL\_01 | Check if the user has entered the correct mobile no. | Mobile no. | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_06 | HL\_02 | Check if the user is removed or not | Employee id | Calls the function and check the input fields is removed or not | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_07 | HL\_03 | Check if employee id exist | Employee id | Calls the function and check the input fields | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_08 | HL\_04 | Check if the information is updated or not | (1) Employee id (2) Field to be modified | Calls the function and check the input fields is updated or not | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_09 | HL\_05 | Check if the role has entered correctly or not | Role | Calls the function and check the input fields is correct or not | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_10 | HL\_05 | Check if the BU has entered correctly or not | BU | Calls the function and check the input fields is  correct or not | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_11 | HL\_05 | Check if the date has entered correctly or not | Date | Calls the function and check the input fields is correct or not | **SUCCESS** | **REQUIREMENT BASED** |
| LL\_12 | HL\_08 | Check if salary calculator are working correctly or not | **(**1) Monthly salary | Correct salary | **SUCCESS** | **TECHNICAL** |
| LL\_13 | HL\_08 | Check if gratuity calculator are working correctly or not | **(**1) Monthly salary | Correct gratuity | **SUCCESS** | **TECHNICAL** |
| LL\_14 | HL\_08 | Check if PF calculator are working correctly or not | **(**1) Monthly salary | Correct PF | **SUCCESS** | **TECHNICAL** |
| LL\_15 | HL\_08 | Check if tax calculator are working correctly or not  Employee id | **(**1) Monthly salary | Correct TAX | **SUCCESS** | **TECHNICAL** |

**REFERENCES**

1. <https://tallyfy.com/uml-diagram/>
2. https://creately.com/blog/diagrams/uml-diagram-types-examples/

**ACTIVITY 2: AGILE METHODOLOGY**

**THEME**

**AGILE MODEL –**

* Agile is a software development. This model includes acceptance of change, mid course feedback and correction

**Features of Agile:**

* Individuals and interactions over processes and tools
* Working software over comprehensive documentation
* Customer Collaboration over contract negotiation
* Responding to change over following a plan

**USER STORIES-**

## **Scrum Master: Shriya Prashant Naik (99004428)**

**Questions:**

* (1) What have I done since the last Daily Scrum?
* (2) What do I plan to do by the next Daily Scrum?
* (3) What are my blocks?

Date: 21/05/21

#### Kanamarlapudi Kavya Lakshmi Naga Pavani (99004429):

(1) Gave an insight of high and low level test plan as well as well as high and low level requirements to the team members, contributed to the introduction part with introduction for Employee Management System, features used in the project, benefits and ageing. (2) Plan by next daily scrum is to work on UML diagrams and the code of the sub-feature assigned. (3) Attendance & Leave.

#### Vanukuri Meher Thanmaiee (99004431):

(1) Gave an insight of high and low level requirements as well as test plan to the team members, contributed to the introduction part with SWOT analysis, 4W’s and 1H as well as low level design for the feature: Add/Remove Employee Details. (2) Plan by the next daily scrum is to work on the code for adding employee (fresher and experienced) details. (3) Add/Remove Employee Details.

#### Shriya Prashant Naik (99004428):

(1) Gave an insight of high and low level requirements and test plan, contributed to the requirements part by collecting all the ideas regarding every feature and dividing, making table on high level and low level requirements as well as low level design for the feature: Update Employee Details. (2) Plan by the next daily scrum will be to work on the code for updating details (check whether the employee id gets searched or not). (3) Update Employee Details.

#### Yukti Vinodkumar Patel (99004432):

(1) Gave an insight of high and low level requirements and test plan, contributed to the requirements part by collecting all the ideas regarding every feature and dividing, making table on high level and low level requirements as well as low level design for the feature: Search Employee List. (2) Plan by the next scrum is to complete 1 remaining low level design and to work on code to check whether user have entered correct data or not & search for the record by date. (3) Search Employee List.

#### Shivangi Na (99004430):

(1) Gave an insight of high and low level requirements and test plan, contributed to the architecture part of high level design for all the features and low level design for the feature: Display. (2) Plan by the next scrum is to complete remaining high level and low level designs for salary calculator and display employee details. (3) Display Employee details, Salary Calculator.

Date: 22/05/21

### Kanamarlapudi Kavya Lakshmi Naga Pavani (99004429):

(1) Worked on low level design for attendance and leave, making algorithms for attendance part of the code. (2) Plan by the next scrum is to work and try to complete the code for attendance as well as leave. (3) Was facing issues while designing low level design, so took reference, discussed with the team members and completed it

#### Vanukuri Meher Thanmaiee (99004431):

(1) Modified low level design for add/remove employee information and updated it in the Github repo, started with the code and finished coding part for adding employee (fresher’s & experienced) details and the code is working as expected. (2) Plan by the next scrum is to work on delete employee details part and will check the working of the whole code. (3) Found difficulty in the number of lines in the code and managed it by adding new features.

#### Shriya Prashant Naik (99004428):

(1) Worked on the code to check whether the employee id exists or not, made a search function to compare the provided employee id with the existing ones also added workflows to the Github repo. (2) Plan by the next scrum is to work on the update function and to check whether it’s working fine. (3) Facing issues with the search logic, explored more on it and completed that part.

#### Yukti Vinodkumar Patel (99004432):

(1) Worked on makefile and code for searching employee list by date of joining. (2) Plan by the next scrum is to write the code for searching employee list with the help of BU and roles. (3) Facing issues with VScode where it was made compatible for Embedded C, now changed the settings and made it compatible for C language.

#### Shivangi Na (99004430):

(1) Modified high level designs and worked and updated on low level design for display in Github repo. (2) Plan by the next scrum is to work on low level design for salary calculator and the code for display. (3) Facing issues with UML diagrams- explored and completed it, searching for salary calculator’s part formulas i.e. for insurance, PF calculator etc.

Date: 23/05/21

#### Kanamarlapudi Kavya Lakshmi Naga Pavani (99004429):

(1) Learn how to create main file and completed the code for attendance and leave. (2) Plan by the next scrum is to work and try to complete the test file for attendance as well as leave. (3) Was facing issues in the code, compile time errors, fixed those by discussing with the team members.

#### Vanukuri Meher Thanmaiee (99004431):

(1) Completed 90% of the code and the test files for creating test cases for the feature add/remove employee details. (2) Plan by the next scrum is to work and modify some parts of the test file. (3) Found difficulty in building logic for checking whether the email id is valid or invalid, and fixed it by discussing with the team members.

#### Shriya Prashant Naik (99004428):

(1) Completed the coding part for update function and checked it working also completed 80% of the task assigned. (2) Plan by the next scrum is to work on test file for update employee details. (3) Facing issues with the update function, explored more on it and discussed it with the team members.

#### Yukti Vinodkumar Patel (99004432):

(1) Completed 60% of the coding part as well as searching employee list with BU & roles and test file. (2) Plan by the next scrum is to complete the whole code and to modify things if necessary. (3) No blocks.

#### Shivangi Na (99004430):

(1) Created low level design for salary calculator, completed 80% of the code for display. (2) Plan by the next scrum is to work on the code for salary calculator and complete the test file. (3) Facing issues with building algorithm for display employee details, fixed it by exploring more on the internet.

Date: 24/05/21

#### Kanamarlapudi Kavya Lakshmi Naga Pavani (99004429):

(1)Worked on the code for attendance and leave also created test file for the same. (2) Plan by the next scrum is to upload the code to the Github repo. (3) Was facing errors in the code, run time error, fixed those by discussing with the team members.

#### Vanukuri Meher Thanmaiee (99004431):

(1) Completed 98% of the code and the test files for creating test cases for the feature add/remove employee details. (2) Plan by the next scrum is to complete and upload the code, work on main file and make changes in the Low level design. (3) Found difficulty in test cases, discussed with the team members and fixed those.

#### Shriya Prashant Naik (99004428):

(1) Completed the coding part for update function and checked its working also completed the test file. (2) Plan by the next scrum is to work on main file. (3) Facing issues with the test cases, fixed those by tracking errors at every line.

#### Yukti Vinodkumar Patel (99004432):

(1) Completed 100% of the coding part as well as searching employee list with BU & roles and test file. (2) Plan by the next scrum is to work on the main file. (3) Facing issues with the search function of date of joining.

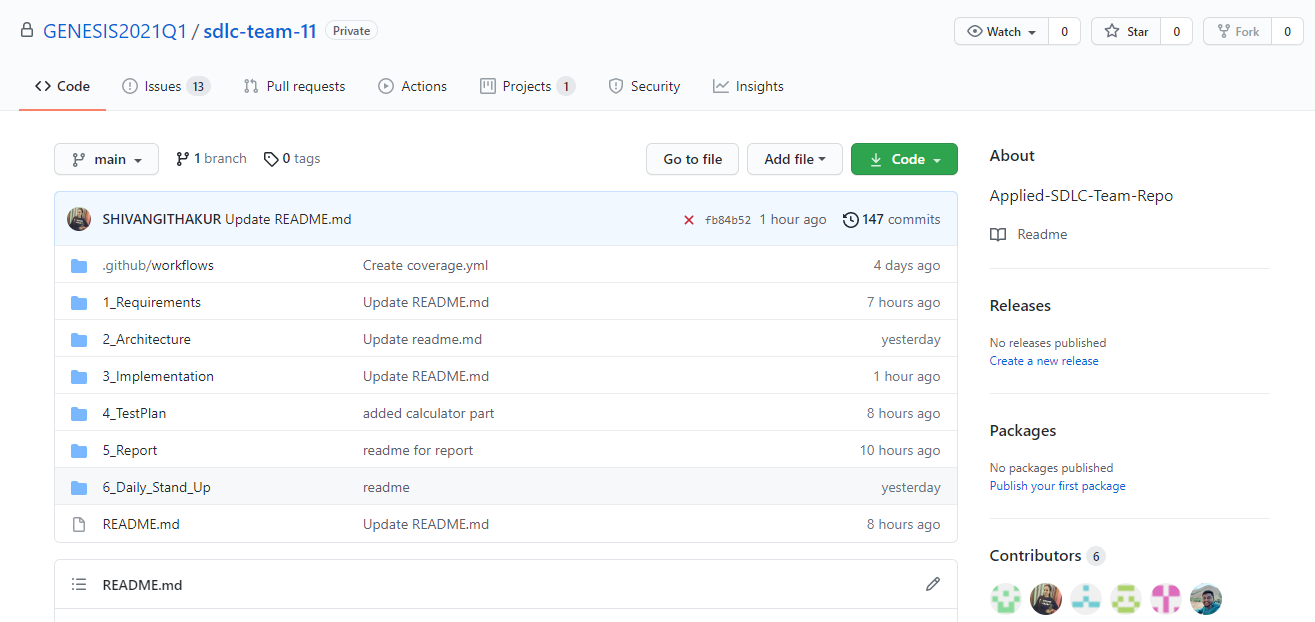
#### Shivangi Na (99004430):

(1) Completed 100% code for display. (2) Plan by the next scrum is to work on the code for salary calculator and complete the test file. (3) Facing errors for display employee details, fixed it by discussing it with the team members.

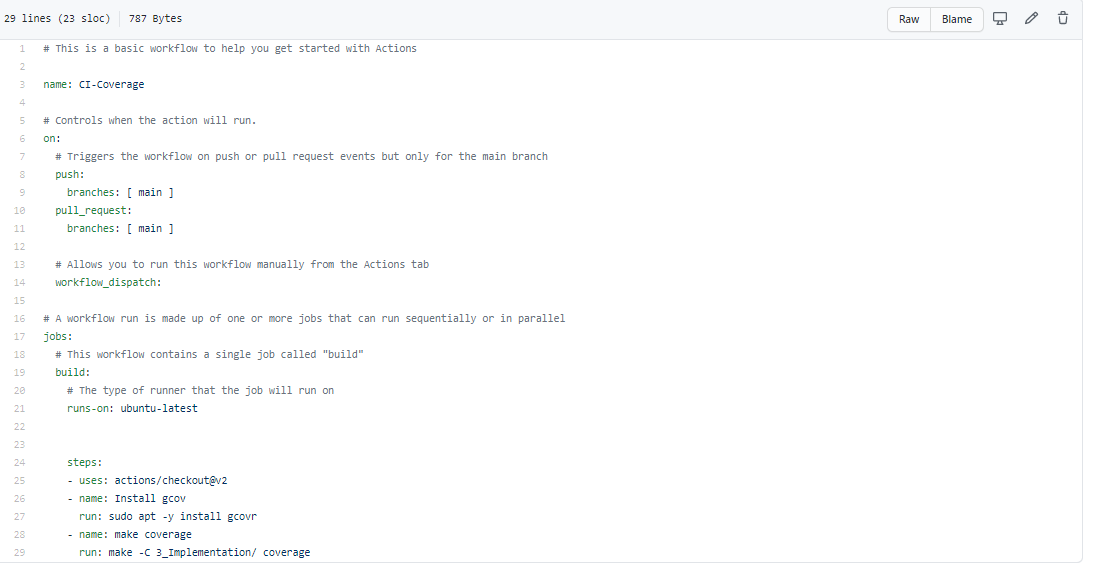
**REFERENCES**

* 1. <https://productivehut.com/v-model-agile-waterfall-spiral/>
  2. <https://www.guru99.com/software-development-life-cycle-tutorial.html#9>

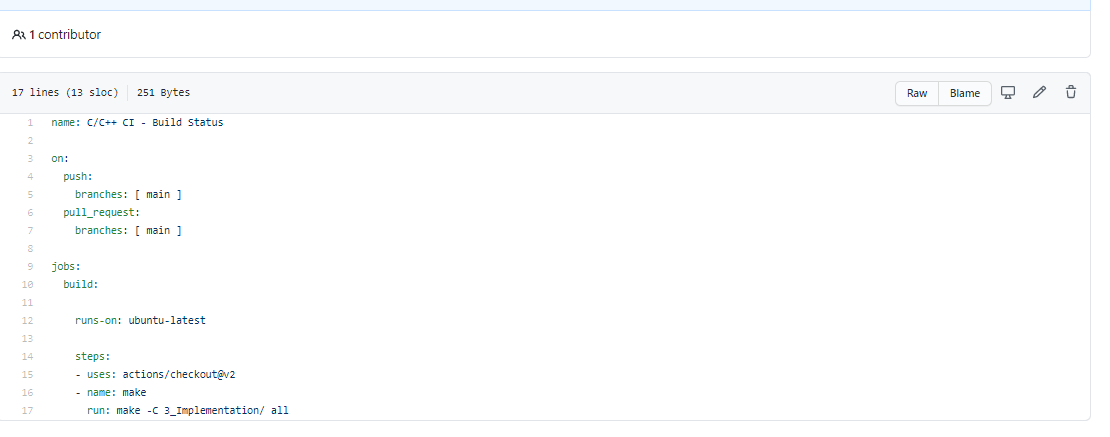
**GIT :**



**GIT CODE QUALITY :**



**GIT BUILD :**



**GIT ISSUES :**

