1. **INTRODUCTION**

An effective employee management system is used to manage the work process and organizational responsibilities of human resources and other departments. It helps managers and employees to work together and accurately monitor, access, manage, and efficiently utilize the working hours for better business growth.

every organization has different employee management needs, therefore we design exclusive employee management systems that are adapted to your managerial requirements.

This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of human resources for your future goals.

**1.1PROJECT SUMMARY**

The Employee Management System project aims to develop a software to manage employee data and records more efficiently than a manual system. It will allow users to enter, edit, update and view employee information like personal details, skills, attendance, salary and leaves.

**1.2 PURPOSE**

These systems centralize your employee data to streamline HR operations and functions and make decision-making much easier. From helping you onboard employees to managing payroll activities, these tools help manage an organization's HR operations

**1.3 HTML, CSS, ANDROID**

**Front-End**

• XML

**Back – End**

• JAVA

**Database**

• Firebase

**Development Tools**

• Android Studio

**Version Control**

• Git And Git hub

**2.PROJECT MANAGEMENT**

* 1. **PROJECT PLANNING AND SCHEDULING**

**2.1.1 PROJECT DEVELOPMENT APPROACH**

Following the software engineering standard as specified for Software Engineering.

We are using the Iterative Waterfall Model for the development of the system.

This process model is explained in brief below.

**· Justification**

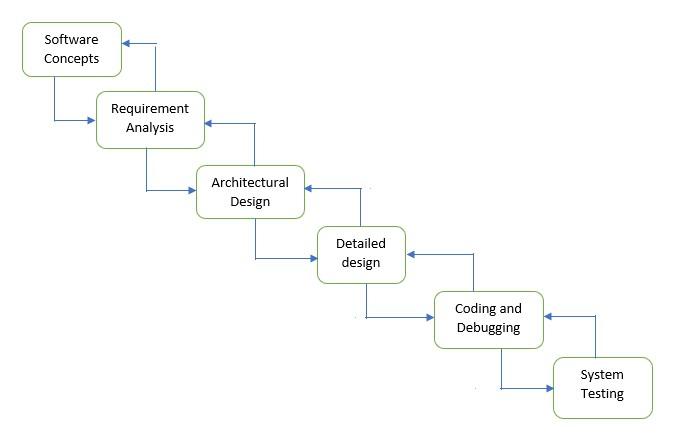
In the Software Development Life Cycle, there are different stages for requirement collection, feasibility study, requirement determination, design, coding and implementation and then testing and debugging so we can first identify requirements and we can do the feasibility study.

Thus it is beneficial to first identify the requirements and then through feasibility study we can analyze these requirements and determine them for implementation.

Then after gathering all necessary requirements we can easily design them and then the implementation becomes very easy and faster.

The client requirements were quite fluctuating and that enforces us to choose a model that allows us to move back to any previous phase of the development life cycle, make changes over there, & again get it implemented in the next phase.

This repeats until the satisfactory level is reached.



**2.1.2 PROJECT PLAN**

The road to successful project development is the well planned strategy for the best and optimal use of resources available. The step wise plan for the project is as follows:

1. Understand the system firstly.

2. Once familiar with the system, start to work on it.

3. First of all, design a page using various controls from the toolbox.

4. Create a relationship among all these tales, wherever needed.

5. Generate classes that cover the coding part and help with the insertion operations.

6. Approval by testing

** Milestones and Deliverables**

Milestone and Deliverables are the important task for the project scheduling because if milestone is achieved in the specified time than I will increase customer faith and Deliverables means that software is delivered with all the requirements specified by the user

** Roles and Responsibilities**

As the project development was under a team organization of three phones, all the phases are divided into parts and each module of it was assign to each person in the team. Each person has to complete his task within a specifications defined and then finally integration of the whole work was done All of us worked together on same phases and we also divided the phases into sub phases so that our work was more efficient, effective and less time consuming As a result we wore able to achieve our defined system with all the specifications given to us.

** Group Dependencies**

A project Guide provides the technical leadership and is designated as the chief programmer. The chief programmer partitions the task into small activities and assigns us. He also verifies and integrates the products developed by us and we worked under the constant supervision of the project Guide.

**2.1.3 SCHEDULE REPRESENTATION**

Scheduling the project task is an important project planning activity. It involves deciding which tasks would be taken up when. Based on the planned duration of required tests and collection of resources to complete those tasks projected completion date is calculated.

**3. SYSTEM REQUIREMENTS STUDY**

**3.1 USER CHARACTERISTICS**

The System contains only two Users

1. Admin

2. Users

 **Admin**

Admin is the main user responsible for some important tasks of the system which are necessary for proper working of system. The tasks only performed by Admin are

•Manage Android Application

** Users**

Users can perform basic tasks like

•Login to the Application

•View Profile

•View Working Hours

•View Tasks

•Add Leave

•View Salary-slips

•View Holidays

•Logout to the Application

**3.2 HARDWARE**

* **Hardware Requirements**

|  |  |
| --- | --- |
| **Processor** | Intel core i5, core i7 , Ryzen 3, |
| **Ram** | 8.00GB Ram |
| **Keyboard** | Standard |
| **Mouse** | Standard |
| **Hard Disk** | **F** Free Space enough to install and run software cleanly. |

* **Software Requirements**

|  |  |
| --- | --- |
| **Operating System** | Windows 11,10 ,8 |
| **Tools\*** | Android Studio |

**4. SYSTEM ANALYSIS**

**4.1 STUDY OF CURRENT SYSTEM**

Notes works with current systems and leverages existing technology.

Centralizes the mountains of data to learning and automates routing administrative functionality.

This package has education's most flexible and interactive scheduling function, thus meeting the communication and information needs of the entire Institute community in real time.

Institute Management system is the total management system imagined the first truly scalable institute management package with power revolutionize the way the Institutes are run.

The software is more than just another technology solution. It is an education system that will improve the way Institute is managed IMs is a policy driven process, which requires customization as per our client's process.

**4.2PROBLEM & WEAKNESS OF CURRENT SYSTEM**

Current system is a policy driven process, which requires customization as per our client’s process.

The design is an easy process to understand, handle or customize the product by creating institute specific rules.

With experiences of various intuit we deliver the best processing modules with great comfort level of our user’s it’s a very high level of security and functional models this is the best End easy to use software for any kind of institute.

These systems vary in size, scope and capability, from packages that are implemented in relatively small organizations to cover student records alone, to enterprise-wide solutions that aim to cover most aspects of running large multi-campus organizations with significant local responsibility.

Many systems can be scaled to different levels of functionality by purchasing add-on ‘modules’ and can typically be configured by their home institutions to meet local needs. Doesn’t provide user friendliness at the pick level or the flow of the system is very awkward the system is an online website so if a User wants to integrate it with their own software for Intranet website certain modification to its usability has to be done.

**4.3 FEASIBILITY STUDY**

**User-friendly**

Feasibility study is carried out whenever there is a complex problem or opportunity it is a fact a preliminary investigation which emphasizes the Lock undertaken to determine the possibility or probability of either in he existing system or developing a completely new system. It helps to obtain an overview of the problem and to get rough assessment of whether feasible solutions exist.

Tiera is essential to avoid committing large resources to a project and the repent UN it later.

Need for feasibility study:

The feasibility study is needed to answer the question

•Whether a new system is too installed or not?

•Determine the potential of the existing system.

•Improve the existing system.

•Know what should be embedded in the new systems.

•Define the problems and objectives involved in a projects

•Avoid costly repairs at a later larger stage when the system is implemented.

•Avoid the hardware approach Le getting a computer first and then deciding How to use it.

**Method:**

To conduct a detailed feasibility study, firstly an expert committee called Committee is appointed. This committee generally consists of systems analyst, representatives from the departments we likely to benefit from the prima and chairman who is generally a key person in the organization

The committee will look into

•Technical feasibility

•Economic feasibility

•Operational feasibility of the project.

Technical feasibility

The technical feasibility should ask questions related so:

1.Adequacy of available technology.

2.Adequacy of hardware

3.Available of computer

4.Operating time and support facilities, etc.

Technical feasibility determines whether available and how it can be integrated within the system. Technical evaluation must also assess whether the existing system can be upgraded to use the new technology and whether the Educational Institute information system has expertise to use it.

The technical feasibility in the proposed system deals with the technology used in the system. It deals with the hardware and software used in the system whether they are of latest technology or not. It happens that after a system is prepared a new technology arises and the user wants the system based on that technology. Thus it is important to check the system to be technically feasible.

•Available through internet.

•It is easy to use

•Any One can use no need technology knowledge

•Provide better Functionality

•Provide better GUL

**Operational feasibility**

Operational feasibility covers two aspects. one is de technical performance aspect and other is the acceptance within the In the system operational feasibility check, whether the user who is going use the system is able to work with the system with which the system is code and also the mind of the user going to use the system. If the user does not underfund or is able to work on the system further development is of waste .

•Removes manual work

•No extra programming or other skills are required

•Faster work

**Economic feasibility**

Economic feasibility looks at the financial aspects of the project Economic feasibility concerns with the returns from the investments in s project, It determines whether it is worthwhile to invest the money in the proposed system.

It is not worthwhile spending a lot of money on a project for no return. To carry out an economic feasibility for a system, it is necessary to place actual money value against any purchases or activities needed to implement the project.

The system plans to acquire the necessary hardware and software requires for the system and there is no hindrance whether economical or otherwise towards its purchase.

**4.4 REQUIREMENT VALIDATION**

Requirement validation is a crucial step in the development process of any project, including an Employee Management System (EMS). Here's a comprehensive guide on how to validate the requirements for an EMS project:

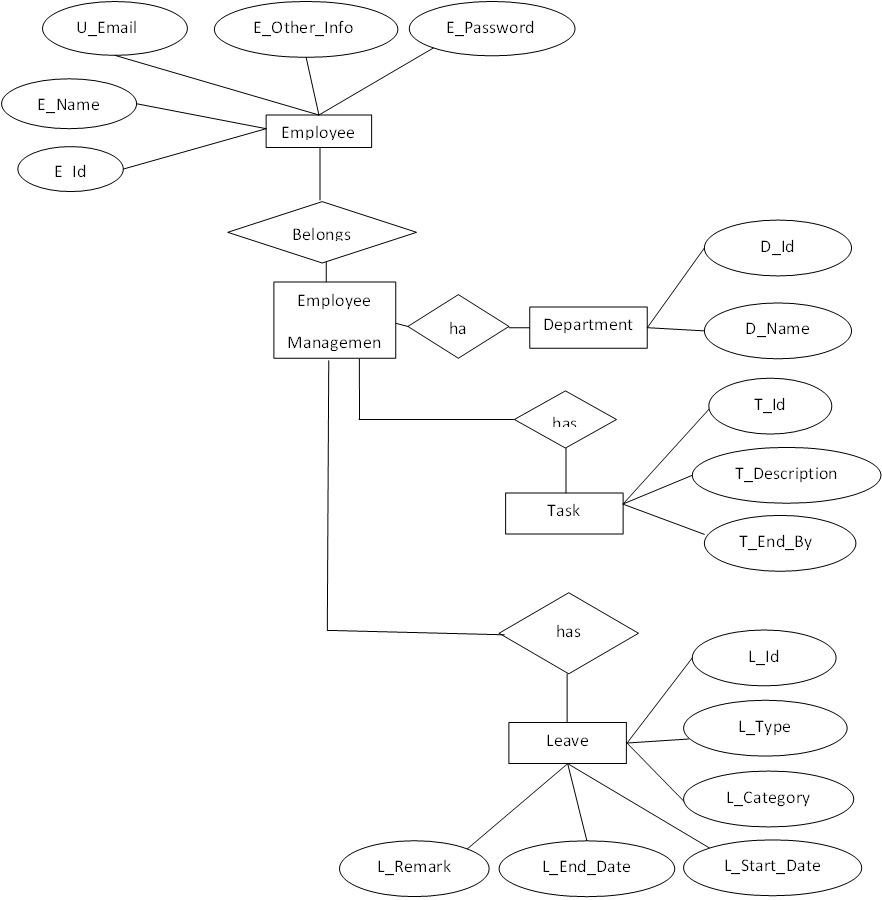
**Gather Requirements:** Before you can validate requirements, you need to have a clear understanding of what the stakeholders expect from the EMS. This involves gathering requirements through interviews, surveys, meetings, and any other means of communication with stakeholders.

**Document Requirements:** Once you have gathered requirements, document them systematically. Use techniques such as use cases, user stories, or requirement specifications to ensure clarity and completeness.

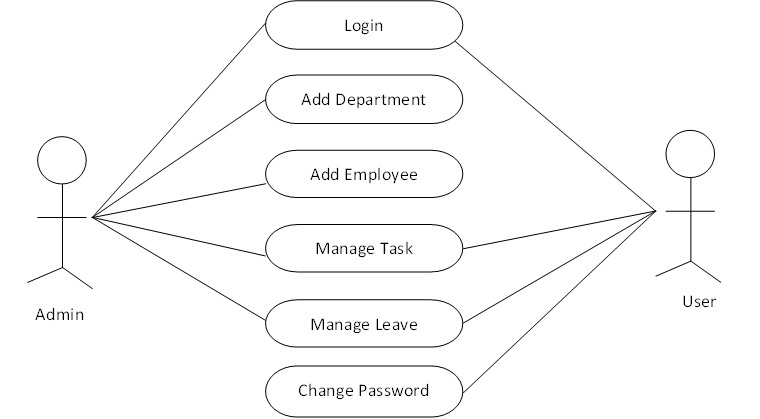
**Review with Stakeholders:** Conduct a thorough review of the documented requirements with all stakeholders. This includes HR managers, employees, IT staff, and any other parties involved. Ensure that everyone understands the requirements and that they accurately represent their needs.

**4.5 FUNCTION SYSTEM**

**4.5.1 E-R DIAGRAM**



**4.5.2 USE CASE DIAGRAM**



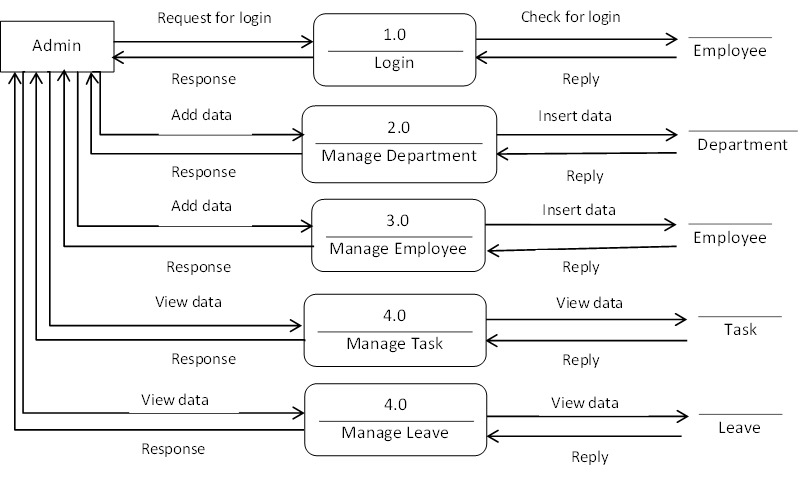
**4.5.3 DATA FLOW DIAGRAM**

**0-Level Data Flow Diagram**

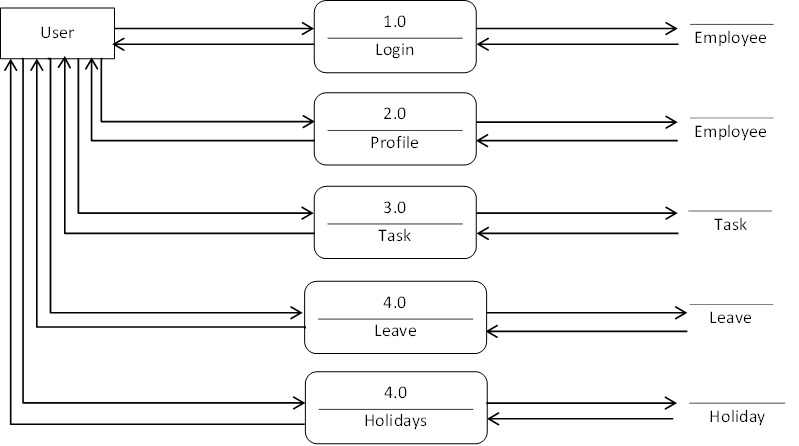


**1-Level Data Flow Diagram**

**Admin side**



**User side**



**4.6 DATA DICTIONARY**

**Employee Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr No.** | **Field Name** | **Field Data type** | **Constrains** | **Description** |
| 1 | E\_No | Int(10) | Primary Key | Hold Employee No |
| 2 | E\_Id | Varchar(15) | Unique Key | Hold Employee Id |
| 3 | E\_Full\_Name | Varchar(100) | Not null | Hold Employee Name |
| 4 | E\_ Father\_Name | Varchar(100) | Not null | Hold Employee Father Name |
| 5 | E\_ Mother\_Name | Varchar(100) | Not null | Hold Employee Mother Name |
| 6 | E\_Gender | Varchar(10) | Not null | Hold Employee Gender |
| 7 | E\_DOB | Date | Not null | Hold Employee DOB |
| 8 | E\_Blood\_Group | Varchar(10) | Not null | Hold Employee Blood Group |
| 9 | E\_Age | Varchar(100) | Not null | Hold Order User Email Id |
| 10 | E\_Adhar\_No | Varchar(12) | Unique Key | Hold Employee Adhar No |
| 11 | E\_Mobile\_No | Varchar(15) | Unique Key | Hold Employee Mobile No |
| 12 | E\_Email | Varchar(100) | Unique Key | Hold Employee Email |
| 13 | E\_Address | Varchar(250) | Not null | Hold Employee Address |
| 14 | E\_City | Varchar(50) | Not null | Hold Employee City |
| 15 | E\_State | Varchar(50) | Not Null | Hold Employee State |
| 16 | E\_Country | Varchar(50) | Not Null | Hold Employee Country |
| 17 | E\_Pincode | Varchar(50) | Not Null | Hold Employee Pincode |
| 18 | E\_Emergency\_No | Varchar(50) | Not Null | Hold Employee Emergency No |
| 19 | E\_Department\_Id | Varchar(15) | Foreign Key | Hold Employee Department Id |
| 20 | E\_Joining\_Date | Date | Not Null | Hold Employee Joining Date |
| 21 | E\_Company\_Detail | Varchar(100) | Not Null | Hold Employee Company Detail |
| 22 | E\_Job\_Type | Varchar(50) | Not Null | Hold Employee Job Type |
| 23 | E\_Password | Varchar(50) | Not Null | Hold Employee Password |

**Employee Department**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr No.** | **Field Name** | **Field Data type** | **Constrains** | **Description** |
| 1 | Dept\_No | Int(10) | Primary Key | Hold Employee Department |
| 2 | Dept\_Id | Varchar(15) | Unique Key | Hold Employee Id |
| 3 | Dept\_Name | Varchar(100) | Not null | Hold Employee Name |

**Employee Leave**

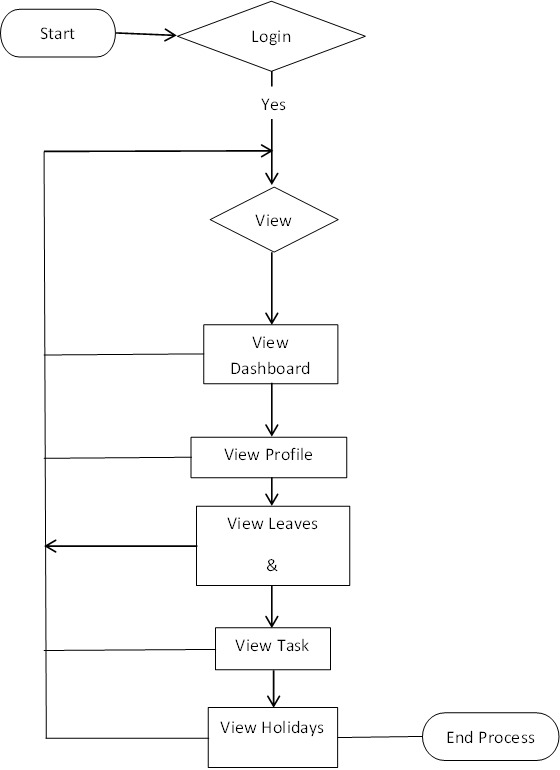
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr No.** | **Field Name** | **Field Data type** | **Constrains** | **Description** |
| 1 | L\_No | Int(10) | Primary Key | Hold Leave No |
| 2 | E\_Id | Varchar(15) | Foreign Key | Hold Leave Id |
| 3 | L\_Type | Varchar(100) | Not null | Hold Leave Type |
| 4 | L\_Category | Varchar(100) | Not null | Hold Leave Category |
| 5 | L\_Start\_Date | Date | Not null | Hold Leave Start Date |
| 6 | L\_End \_Date | Date | Not null | Hold Leave End Date |
| 7 | L\_Description | Varchar(100) | Not null | Hold Leave Description |
| 8 | L\_Status | Varchar(50) | Not null | Hold Leave Status |

**Employee Working Hours**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr No.** | **Field Name** | **Field Data type** | **Constrains** | **Description** |
| 1 | Wh\_No | Int(10) | Primary Key | Hold Working Hour No |
| 2 | Wh\_Date | Date | Not Null | Hold Working Hour Date |
| 3 | Wh\_Done | Varchar(500) | Not null | Hold Working Hour Done |
| 4 | Wh\_Tangible\_Outcome | Varchar(500) | Not null | Hold Working Hour Tangible Outcome |

**5. SYSTEM DESIGN**

**5.1 FLOW CHART**



**6. TESTING**

**6.1 TEST PLAN STRATEGIES**

Testing is important phase of the website development. After implementing the website and before delivering it to the client, it is necessary to verify that whether the code written is working properly or not. By testing the website, we can detect the logical errors in the code and can be able to correct them so that the client can get a perfect working website.

We have tested our website of Assets Management by inputting various valid and invalid data. We have checked all the input conditions required to store valid data. For this we have tried all the normal conditions as well as extreme conditions.

We have performed the process of testing at during the implementation phase as well as after the completion of implementation. We tested our website after the completion of each module by entering all kind of data and corrected almost all the incorrect working of the website we observed.

**6.2 STRATEGIES**

**1 Unit Testing :** Unit tests focus on testing individual components or units of code in isolation. For Android development, you can use JUnit along with Mockito or Robolectric for mocking Android components.

**2 Integration Testing:** Integration tests verify the interaction between different modules or components within the application. Espresso is a popular framework for writing integration tests for Android applications. It allows you to simulate user interactions and assert UI behavior..

**3.UI Testing:** UI testing ensures that the user interface behaves as expected. Apart from Espresso, you can use frameworks like UI Automator or Google's Firebase Test Lab for running UI tests across different devices and screen sizes.

**4.Functional Testing:** Functional tests validate the functionality of the application as a whole. These tests focus on user scenarios and cover end-to-end flows. Tools like Appium, Calabash, or Robot Framework can be used for functional testing of Android applications.

**5.Performance Testing:** Performance testing evaluates the responsiveness, stability, and resource usage of the application under various conditions. Tools like Android Profiler, Traceview, or third-party services like Firebase Performance Monitoring can help in identifying performance bottlenecks.

**6. Compatibility Testing:** Ensure your application works across different versions of Android OS, screen sizes, resolutions, and device configurations. Emulators and real devices can be used for compatibility testing.

**7. Security Testing:** Security testing is essential to identify vulnerabilities in the application that could be exploited by malicious users. Tools like OWASP ZAP or IBM Application Security on Cloud can help in conducting security assessments.

**8. Usability Testing:** Usability testing involves gathering feedback from real users to evaluate the user experience of the application. This can be done through alpha/beta testing with real users, usability testing sessions, or feedback mechanisms within the app.

**9. Localization and Internationalization Testing:** If your application is targeted for global audiences, it's essential to test for localization (translation accuracy) and internationalization (support for different languages, date/time formats, etc.).

**10. Automated Build and Deployment Testing:** Automate the build and deployment process to ensure that new changes do not introduce regressions. Continuous Integration (CI) tools like Jenkins, Travis CI, or CircleCI can be integrated into your development workflow for automated testing and deployment.

**11. Exploratory Testing:** While automated tests are essential, manual exploratory testing by QA engineers can uncover issues that automated tests might miss. Encourage your QA team to explore the application thoroughly and report any anomalies they encounter.

**12. Robust Error Handling:** Test how the application behaves under unexpected conditions such as network errors, device orientation changes, low memory situations, etc.

**6.3 TESTING METHODS**

Testing methods for Android applications encompass a variety of approaches to ensure the quality and reliability of the software. Here are some common testing methods used in Android app development:

1. **Manual Testing:** Manual testing involves human testers executing test cases and interacting with the application as an end-user would. Testers validate features, functionalities, and UI elements manually, exploring the application for defects and inconsistencies.
2. **Automated Testing:** Automated testing involves using software tools to execute test cases and compare actual outcomes with expected outcomes. Automated testing can be further divided into several subtypes:

* Unit Testing: Tests individual units or components of the application, such as methods or classes, in isolation from the rest of the system.
* Integration Testing: Verifies interactions between different components or modules of the application to ensure they work together correctly.
* UI Testing: Validates the behavior of the user interface, including user interactions, screen navigation, and UI elements' functionality.
* End-to-End Testing (E2E): Tests the entire application from start to finish, simulating real user scenarios and interactions.

**3. Black Box Testing:** Black box testing focuses on validating the functionality of the application without knowing its internal implementation details. Testers interact with the application through its user interface and verify whether it behaves as expected.

**4. White Box Testing:** White box testing, also known as structural or glass box testing, examines the internal structure and code implementation of the application. Testers have access to the application's source code and design test cases based on its internal logic and structure.

**5. Grey Box Testing:** Grey box testing combines elements of both black box and white box testing. Testers have partial knowledge of the application's internal workings and use this knowledge to design comprehensive test cases.

**6. Regression Testing:** Regression testing ensures that recent code changes or updates have not adversely affected existing functionalities. It involves re-running previously executed test cases to detect any regression issues.

**7. Performance Testing**: Performance testing evaluates the application's responsiveness, stability, scalability, and resource usage under various conditions, such as different network speeds, device configurations, or user loads.

**8. Security Testing:** Security testing identifies vulnerabilities in the application that could be exploited by malicious actors. It includes techniques such as penetration testing, vulnerability scanning, and code review to ensure the application's security posture.

**9. Usability Testing:** Usability testing assesses the application's user experience, including ease of use, intuitiveness, accessibility, and overall user satisfaction. Testers gather feedback from real users to identify areas for improvement.

**10. Compatibility Testing:** Compatibility testing verifies that the application functions correctly across different devices, operating system versions, screen sizes, and orientations. It ensures a consistent user experience across a wide range of platforms and configurations.

**6.4 TESTING CASES**

Testing cases for an Employee Management System (EMS) should cover a wide range of scenarios to ensure that the system functions as intended, is user-friendly, and meets the requirements of stakeholders. Here are some examples of testing cases for an EMS**:**

**User Authentication and Authorization:** Verify that users can log in with valid credentials. Ensure that invalid login attempts are appropriately handled (e.g., error messages). Test different user roles (e.g., admin, HR manager, regular employee) and their respective permissions.

**Employee Information Management:** Add a new employee to the system and verify that the information is correctly stored. Edit employee details (e.g., contact information, department) and ensure that changes are updated accurately. Delete an employee and confirm that their data is removed from the system.

**Attendance and Leave Management:** Log employee attendance for various scenarios (e.g., regular workdays, holidays, sick leave). Submit leave requests for different types of leave (e.g., vacation, sick, maternity) and verify approval workflows. Check leave balances before and after leave requests are processed.

**Payroll Management:** Calculate employee salaries based on different pay structures (e.g., hourly, salary, commission). Verify deductions for taxes, benefits, and other withholdings. Confirm that paychecks are generated accurately and on time.

**Performance Evaluation:** Create performance evaluation forms with different criteria and weightings. Conduct performance reviews for employees and assess how ratings are calculated. Verify that performance feedback is saved and accessible to relevant stakeholders.

**7. SCREEN SHOTS**

**8. LIMITATION & FUTURE ENHANCEMENT**

**8.1 Limitations**

The major limitations of the project are as follows:

• Due to the constraint of resources and time, the size of the project could not be increased.

• The project has been developed through utilizing the records of the employees and other information available at certain organization. The requirements gathered through various sources might not be properly reflected in the requirements analysis and the design documents due to limited knowledge and time.

• Due to the small scope the project may not represent the whole spectrum of the human resource management.

• Due to very little knowledge of the team members about ASP.NET language and MySQL database, the project may not have been developed as envisioned.

**9. CONCLUSION AND DISCUSSION**

An application for employee management systems helps your organization improve workforce productivity and boost overall well-being by tracking and monitoring the daily working activities of every employee. In conclusion of employee management system blog, Desk Track is one of the best application for workforce management. It keeps track of every activity done by an employee during his working hours.

As a result, if you are searching for the best employee productivity monitoring software then Desk Track has some impressive features that can be the best fit for your organization.

**10. REFERENCES**

**10.1 WEB REFERENCES**

* URL: <https://firebase.google.com/>
* URL: https://stackoverflow.com/
* URL: https://developer.android.com/