**1. INTRODUCTION**

* 1. **PURPOSE**

The purpose of this project is to build an online system that allows analysis, edit and manage departments of company and its employee.

* 1. **INTENDED AUDIENCE AND READING SUGGESTIONS**

This project is a prototype for the employee management system and it is restricted within the company premises. This project is useful for the management team.

* 1. **PROJECT SCOPE**

The purpose of the online management system is to ease employee management and to create a convenient and easy-to-use application for managers, trying to analisys company situation. The system is based on a relational database with its employee and departments management. I hope to provide a comfortable user experience.

**2. OVERALL DESCRIPTION**

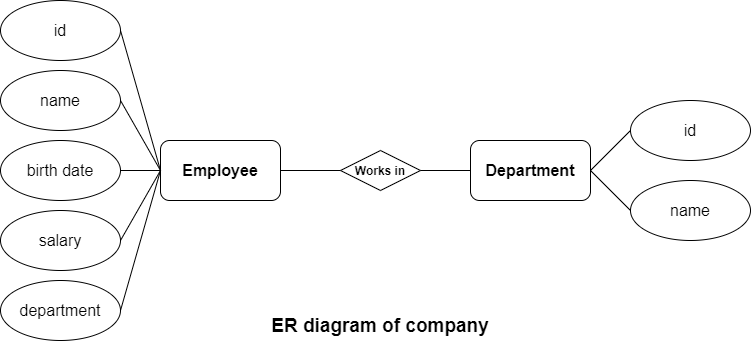
**2.1 PRODUCT PERSPECTIVE**

The online management system stores the following information:

* Employee details
* Department details
* Relationship between employee and department

**2.2 PRODUCT FEATURES**

The major features of employee/departments database system as shown in below [entity–relationship model](https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model" \t "_blank)



The diagram shows the layout of employee/departmen database system – entity–relationship model

**2.3 USER CLASS AND CHARACTERISTICS**

Users of the system should be able to retrieve information about company. The system will support one type of user and it is manager. The manager should be able to do the following functions:

* display a list of departments and the average salary (calculated automatically) for these departments
* display a list of employees in the departments with an indication of the salary for each employee and a search field to search for employees born on a certain date or in the period between dates
* change (add / edit / delete) the above data

**3. SYSTEM FEATURES**

**CLIENT/SERVER SYSTEM**

Client/server refers primarily to an architecture or logical division of responsibilities, the client is the application (also known as the front-end), and the server is the DBMS.

A client/server system is a distributed system in which,

* Some pages are client pages and others are server pages.
* All the data resides at the server pages.
* All applications execute at the client pages.

**4. EXTERNAL INTERFACE REQUIREMENTS**

**4.1 USER INTERFACES**

* Front-end software: Vb.net version
* Back-end software: SQL+

**4.2 HARDWARE INTERFACES**

* Windows.
* A browser which supports CGI, HTML & Javascript.

**4.3 SOFTWARE INTERFACES**

Following are the software used for the flight management online application. <<*Include the software details as per your project*>>

|  |  |
| --- | --- |
| **Software used** | **Description** |
| Operating system | We have chosen Windows operating system for its best support and user-friendliness. |
| Database | To save the flight records, passengers records we have chosen SQL+ database. |
| VB.Net | To implement the project we have chosen Vb.Net language for its more interactive support. |

**4.4 COMMUNICATION INTERFACES**

This project supports all types of web browsers. We are using simple electronic forms for the reservation forms, ticket booking etc.

**5. NONFUNCTIONAL REQUIREMENTS**

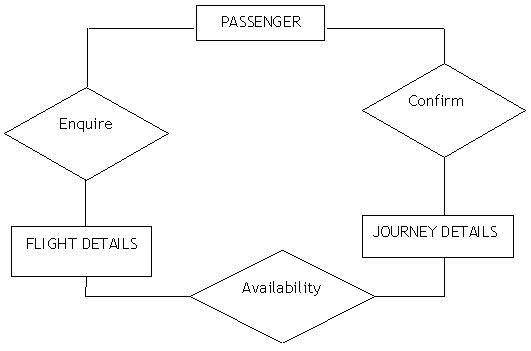
**5.1 PERFORMANCE REQUIREMENTS**

The steps involved to perform the implementation of airline database are as listed below.

**A) E-R DIAGRAM**

The E-R Diagram constitutes a technique for representing the logical structure of a database in a pictorial manner. This analysis is then used to organize data as a relation, normalizing relation and finally obtaining a relation database.

* **ENTITIES:**Which specify distinct real-world items in an application.
* **PROPERTIES/ATTRIBUTES:** Which specify properties of an entity and relationships.
* **RELATIONSHIPS:** Which connect entities and represent meaningful dependencies between them.

[](https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database/attachment/ad-er-diagram-of-airline-database)

*the diagram shows the ER diagram of airline database*

**B) NORMALIZATION:**

The basic objective of normalization is to reduce redundancy which means that information is to be stored only once. Storing information several times leads to wastage of storage space and increase in the total size of the data stored.

If a database is not properly designed it can give rise to modification anomalies. Modification anomalies arise when data is added to, changed or deleted from a database table. Similarly, in traditional databases as well as improperly designed relational databases, data redundancy can be a problem. These can be eliminated by normalizing a database.

Normalization is the process of breaking down a table into smaller tables. So that each table deals with a single theme. There are three different kinds of modifications of anomalies and formulated the first, second and third normal forms (3NF) is considered sufficient for most practical purposes. It should be considered only after a thorough analysis and complete understanding of its implications.

**5.2 SAFETY REQUIREMENTS**

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

**5.3 SECURITY REQUIREMENTS**

Security systems need database storage just like many other applications. However, the special requirements of the security market mean that vendors must choose their database partner carefully.

**5.4 SOFTWARE QUALITY ATTRIBUTES**

* **AVAILABILITY:** The flight should be available on the specified date and specified time as many customers are doing advance reservations.
* **CORRECTNESS:** The flight should reach start from correct start terminal and should reach the correct destination.
* **MAINTAINABILITY:** The administrators and flight in chargers should maintain correct schedules of flights.
* **USABILITY:** The flight schedules should satisfy a maximum number of customers needs.