CUSTOMER CARE REGISTRY USING

CLOUD

## A PROJECT BASED LEARNING REPORT

***Submitted by***

# ANISHA S C (20104009)

**RAKSHANAA R (20104047)**

# YOSHITHA SRI H A (20104064)

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

***in***

**COMPUTER SCIENCE AND ENGINEERING**

|  |  |  |
| --- | --- | --- |
|  | **HINDUSTHAN COLLEGE OF ENGINEERING AND TECHNOLOGY**  Approved by AICTE, New Delhi, Accredited with ‘A’ Grade by NAAC  **(An Autonomous Institution, Affiliated to Anna University, Chennai)**  Valley Campus, Pollachi Highway, Coimbatore – 641 032 |  |

**MAY 2023**

# Hindusthan College of Engineering And Technology

Approved by AICTE, New Delhi, Accredited with ‘A’ Grade by NAAC **(An Autonomous Institution, Affiliated to Anna University, Chennai)** Valley Campus, Pollachi Highway, Coimbatore – 641 032

## BONAFIDE CERTIFICATE

Certified that this project report **“CUSTOMER CARE REGISTRY USING CLOUDS”** is the bonafide work of **RAKSHANAA R (20104047), ANISHA S C (20104009), YOSHITHA SRI H A (20104064),** who carried

out the project work under my supervision.

|  |  |
| --- | --- |
| **SIGNATURE**  **Mr. R.JAYARAJ**  **SUPERVISOR**  Associate Professor  Computer Science and Engineering Hindusthan College of Engineering and Technology, Coimbatore-32 | **SIGNATURE**  **Dr. S. SHANKAR, M.E., Ph.D.,**  **HEAD OF THE DEPARTMENT**  Computer Science and Engineering  Hindusthan College of Engineering  and Technology, Coimbatore-32 |

Submitted for the Autonomous Institution Project Based Learning Viva-Voce conducted on

**INTERNAL EXAMINER EXTERNALEXAMINER**

# DECLARATION

We, hereby jointly declare that the project work entitled **“CUSTOMER CARE REGISTRY USING CLOUD”,** submitted to the Autonomous Institution Project Based Learning Viva Voce - May 2023 in partial fulfillment for the award of the degree of “**BACHELOR OF ENGINEERING IN COMPUTER SCIENCE AND ENGINEERING”**, is the report of the original project work done by us under the guidance of **Mr.R.JAYARAJ** Assistant Professor, Department of Computer Science Engineering, Hindusthan College of Engineering and Technology, Coimbatore.

## NAME SIGNATURE

**ANISHA S C**

## RAKSHANAA R

**YOSHITHA SRI H A**

I certify that the declaration made by the above candidates are true.

Project Guide,

## Mr.R.JAYARAJ

Associate Professor, Department of CSE, Hindusthan College of Engineering and Technology, Coimbatore –32

**ACKNOWLEDGEMENT**

We take this opportunity to express our wholehearted thanks and our profound respect to all those who guided and inspired us in the completion of this project work.

We extend our sincere thanks to the Founder and Chairman of Hindusthan Educational and Charitable Trust **Shri.T.S.R. KHANNAIYANN** and the Managing Trustee **Smt. SARASUWATHI KHANNAIYANN** and Executive Trustee & Secretary **Mrs. PRIYA SATISH PRABHU** for providing essential infrastructure.

We would like to reveal our profound thanks to our respected Principal,

**Dr. J. JAYA, M. Tech, Ph.D.,** who happens to be striving force in all endeavors.

We would like to express our gratitude to the Head of the Department

**Dr. S. SHANKAR, M.E., Ph.D.,** for bringing out the project successfully and for strengthening the ray of hope.

We would like to express our sincere thanks and deep sense of gratitude to our Class Advisor and Project Co-Ordinator/Supervisor **Mr.R.JAYARAJ** Associate Professor, Department of Computer Science and Engineering, for his valuable guidance, suggestions and constant encouragement which paved way for the successful completion of the mini project work.

We express our immense pleasure and thankfulness to all our department faculty members, technical staffs and friends who helped us for the successful completion of this mini project. We express our earnest gratitude to our **Parents** and family members who encouraged us and strengthened us in perilous path, encountered during our task.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **TITLE** | **PAGE NO** |
|  | **ABSTRACT**  **LIST OF FIGURES**  **LIST OF TABLES** | **i**  **ii**  **iii** |
| **1** | **INTRODUCTION** | **1** |
| **2** | **OBJECTIVE** | **2** |
| **3** | **IDEATION PHASE** | **3** |
|  | **3.1 Literature Survey** | 4 |
|  | **3.2 Problem Statement** | 9 |
| **4** | **PROJECT DESIGN PHASE 1** | 10 |
|  | **4.1 Proposed Solution** | 11 |
|  | **4.3 System Architecture** | 14 |
| **5** | **PROJECT DESIGN PHASE 2** | 17 |
|  | **5.1 Solution Requirements** | 18 |
|  | **5.2 Data Flow Diagrams** | 19 |
|  | **5.3 Technology Stack** | 22 |
| **6** | **PROJECT DEVELOPMENT PHASE** | 25 |
|  | **6.1 Project Development 1** | 26 |
|  | **6.2 Project Development 2** | 28 |
| **7** | **Coding and Solutions** | 32 |
| **8** | **Result and ScreenShots** | 47 |
| **9** | **CONCLUSION** | 49 |
| **10** | **FUTURE ENHANCEMENTS** | 50 |
| **11** | |  |  |  | | --- | --- | --- | | **9** | **REFERENCES** |  | | 51 |

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **FIGURE.NO** | **FIGURE NAME** | **PAGE NO** |
| 5.1 | Level 0 Diagram | 19 |
| 5.2 | Level 1 Admin | 19 |
| 5.3 | Level 1 Customer | 20 |
| 5.4 | Level 1 Service Providers | 20 |
| 5.5 | Level 2 Admin | 21 |
| 5.6 | Level 2 Customers | 21 |
| 8.1 | Performance metrics | 47 |
| 8.2 | Employee Count by Department | 47 |
| 8.3 | Attrition Based on Salary Hike Percentage | 48 |
| 8.4 | Attrition Based on Department, Job Role and Education | 48 |

# LIST OF TABLES

|  |  |  |
| --- | --- | --- |
| **TABLE NO** | **TABLE NAME** | **PAGE NO** |
| 4.1 | Proposed Solution | 11 |
| 5.1 | Technologies Used | 22 |
|  |  |  |

# ABSTRACT

Corporate employee attrition analytics refers to the analysis of data related to employee turnover within a company. This involves collecting and analysing data on various factors such as employee demographics, job satisfaction, compensation, work environment, and performance, to determine patterns and trends that contribute to employee turnover.

The primary objective of conducting attrition analytics is to identify the factors that drive employee turnover, develop strategies to reduce attrition, and retain high-performing employees. The analysis of attrition data can also provide insights into the effectiveness of recruitment, training, and talent management processes.

In addition, attrition analytics can help organizations forecast future attrition rates, estimate the cost of employee turnover, and evaluate the impact of turnover on organizational performance. By using predictive analytics, companies can anticipate which employees are likely to leave and take proactive measures to retain them, such as offering career development opportunities or increasing compensation.

Overall, corporate employee attrition analytics is a valuable tool for organizations to understand the drivers of employee turnover, develop strategies to reduce it, and improve overall organizational performance.

**CHAPTER 1**

## 1.INTRODUCTION

The corporate employee attrition analytics project is designed to help organizations understand the reasons for employee turnover and develop strategies to reduce it. This project involves collecting and analyzing data on various factors that influence employee turnover, such as job satisfaction, compensation, work environment, and performance.

The primary goal of the project is to identify the root causes of employee turnover and develop strategies to address them. The project team will use statistical and predictive analytics techniques to analyse the data and identify patterns and trends that contribute to employee turnover.

The project will begin by defining the scope of the analysis, including the specific employee groups and time period to be included. The team will then collect data from various sources, such as HR records, employee surveys, and performance evaluations. The data will be cleaned, processed, and analyzed using statistical software to identify key factors influencing employee turnover.

The team will then develop a predictive model to forecast future attrition rates and estimate the cost of employee turnover. Based on the analysis, the team will make recommendations to the organization for improving employee retention and reducing turnover.

## 

## CHAPTER 2

## 

## 2 . OBJECTIVE

Employee attrition has become a vital problem across the world. It is one of the crucial issues faced by business leaders within companies where they lose the most talented employees. A good employee is always an asset to the organization and their resignation can lead to various problems like financial losses, overall performance, and loss of acquired knowledge.

Furthermore, hiring new employees is far exorbitant, taxing, and time-consuming in comparison to recruiting the existing one. It is very time- consuming to recruit a new employee as it takes him months for training, adjusting to the culture, rules, and environment.

Therefore, upcoming trends and technology using Machine Learning Algorithms must be exploited for the benefit of business organizations. Knowing the reason beforehand for the employee attrition, companies can mitigate this loss. This analysis provides a conclusive review of employee attrition from the data set IBM HR Analytics Employee Attrition Performance.

## CHAPTER 3

## 3.IDEATION PHASE

Hardik P. K. (2016), researched on “a study on employee attrition: with special reference to Kerala IT Industry”. His research examined the relationship between organizational factors and attrition of IT professionals. The result can conclude that the organizational factors played significant role in predicting the variance in turnover intention (attrition) of Kerala IT professionals. Therefore, the HR managers in organizations may take into consideration the problems with organizational factors of their workers to reduce the turnover intention of the skilled employees.

Customer service is vital to every company. What would happen if someone called your customer service line and didn’t get satisfaction? Many times customers complain about poor services and bad treatment from businesses.

We’ve heard horror stories where customers have had negative experiences with customer service representatives (CSRs). Most of us would rather pay higher prices because CSR didn’t give them what they were promised. Unfortunately, this leads to lower sales volumes, fewer repeat purchases, and lost revenue. That’s why it’s important to monitor customer service.

Monitoring customer service involves identifying key metrics and measuring their trends over time to determine whether things are improving or deteriorating. The goal is to identify areas where improvements can be made before customers start complaining. The result is better customer experiences overall.

# 3.1 LITERATURE SURVEY

## OVERVIEW

A literature survey for a corporate employee attrition analytics project would involve researching and analyzing existing studies, articles, and publications related to employee turnover and attrition in organizations. The survey would aim to identify key trends, best practices, and insights that can inform the design and implementation of the attrition analytics project.

The survey would start with a search of academic databases, such as JSTOR, EBSCO, and ProQuest, for relevant articles and studies on employee turnover. This would involve reviewing studies on various factors that contribute to employee turnover, such as job satisfaction, organizational culture, compensation, and work-life balance. The survey would also look at research on the impact of turnover on organizational performance, such as productivity, employee morale, and financial performance.

In addition to academic sources, the survey would also include industry publications, such as trade journals and white papers, to gain insights into current trends and best practices in employee retention and attrition management. This would involve reviewing case studies and success stories of organizations that have implemented effective attrition management strategies and analyzing the key factors that contributed to their success.

The survey would also look at the use of data analytics and predictive modelling in employee retention and attrition management. This would involve reviewing studies on the use of data analytics to identify factors that drive turnover, predict employee behaviour, and develop effective retention strategies.

Overall, the literature survey would provide a comprehensive overview of the current state of research and practice in employee retention and attrition management, and help inform the design and implementation of the corporate employee attrition analytics project.

## Advantages of a corporate employee attrition analytics:

**Improved retention:** The project can help organizations identify the root causes of employee turnover and develop strategies to address them. By reducing employee turnover, organizations can improve retention and retain their most valuable employees.

**Cost savings:** High employee turnover can be costly for organizations due to the expenses Assistant d with recruitment, training, and lost productivity. The project can help organizations estimate the cost of employee turnover and develop strategies to reduce it, leading to cost savings.

**Improved organizational performance:** High employee turnover can negatively impact organizational performance, including productivity .employee morale, and financial performance. The project can help organizations identify areas for improvement and develop strategies to improve overall organizational performance.

**Data-driven decision-making:** The project involves collecting and analyzing data on various factors that influence employee turnover. By using data-driven decision-making, organizations can make more informed and effective decisions about retention strategies.

## Disadvantages of a corporate employee attrition analytics:

**Cost and time:** Collecting and analyzing data on employee turnover can be time-consuming and expensive. Organizations may need to allocate resources to conduct the analysis and implement the recommended retention strategies.

**Ethical concerns:** The project may involve collecting sensitive data on employee behaviour and job satisfaction. Organizations must ensure that they adhere to ethical standards and protect employee privacy throughout the project.

**Limited scope:** The project may only focus on specific employee groups or time periods, limiting the scope of the analysis. Organizations must carefully define the scope of the analysis to ensure that it is comprehensive and relevant to their needs.

**Limited predictive accuracy:** Predictive analytics models may not always accurately predict future employee turnover rates, as they are based on historical data and assumptions about future trends. Organizations must be aware of the limitations of predictive analytics and use them as a tool to inform decision-making, rather than as a definitive prediction of future outcomes.

## The algorithm for a corporate employee attrition analytics project would typically involve the following steps:

**Define the problem and objectives:** The first step is to clearly define the problem and objectives of the project. This involves identifying the specific employee groups and time period to be analyzed, as well as the key factors that may be contributing to employee turnover.

**Collect and clean data:** The next step is to collect and clean the relevant data from various sources, such as HR records, employee surveys, and performance evaluations. The data should be cleaned and processed to ensure its quality and accuracy.

**Exploratory data analysis:** Once the data is cleaned, exploratory data analysis techniques can be used to identify patterns and trends in the data. This may involve visualizations, such as scatter plots, histograms, and box plots, to help identify relationships between variables.

**Statistical analysis:** After identifying potential relationships between variables, statistical analysis techniques can be used to test these relationships and identify statistically significant factors that contribute to employee turnover. This may involve techniques such as regression analysis, hypothesis testing, and chi-square tests.

**Predictive modelling:** Based on the results of the statistical analysis, predictive models can be developed to forecast future employee turnover rates and estimate the cost of employee turnover. This may involve techniques such as logistic regression, decision trees, and random forests.

**Interpretation and recommendation:** Once the predictive models are developed, the results can be interpreted and used to develop recommendations for improving employee retention and reducing turnover. This may involve developing retention strategies, such as improving job satisfaction, increasing compensation, and providing opportunities for career development.

**Implementation and monitoring:** Finally, the recommendations can be implemented, and the impact of the retention strategies can be monitored over time. This involves collecting data on employee turnover rates and evaluating the effectiveness of the retention strategies.

## 3.2 PROBLEM STATEMENT

The Existing system includes only few attributes for analysis and also us deals with qualitative observations and simple statistical analysis. The qualitative observations deal with data and can be observed through human senses. They do not involve measurements or number. Due to the increase in IOT and connected device, we now have access to so much of data and along with it an increase needs to manage and understand data.

**Manual tracking and analysis:** Some organizations may still use manual methods to track and analyse employee turnover, such as spreadsheets or paper records. This method can be time-consuming and may not provide a comprehensive view of the factors that contribute to employee turnover.

**Business intelligence tools:** Some organizations use business intelligence tools to analyse data from various sources, including HR data. These tools may provide more advanced analytics and reporting features, but may still not provide the specialized predictive modelling capabilities needed for an employee attrition analytics project.

**Third-party analytics services:** Some organizations may outsource their employee attrition analytics project to third-party analytics services. These services may provide specialized predictive modelling capabilities and expertise in employee turnover analytics, but may be expensive and may not provide the same level of customization and control as an in-house project.

**CHAPTER 4**

**PROJECT DESIGN PHASE I:**

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer’s goal is to produce a model or representation of an entity that will later be built.

Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software. The importance can be stated with a single word “Quality”. Design is the place where quality is fostered in software development.

Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer’s view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective.

From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

## 4.1 PROPOSED SOLUTION

The Existing system includes only few attributes for analysis and also deals with qualitative observations and simple statistical analysis. The qualitative observations deal with data and can be observed through human senses. They do not involve measurements or number. Due to the increase in IOT and connected device,

We now have access to so much of data and along with it an increase needs to manage and understand data. A proposed system for corporate employee attrition analytics would involve using data analytics tools and techniques to analyse and understand the factors that contribute to employee turnover within the organization. The following steps could be taken to develop such a system:

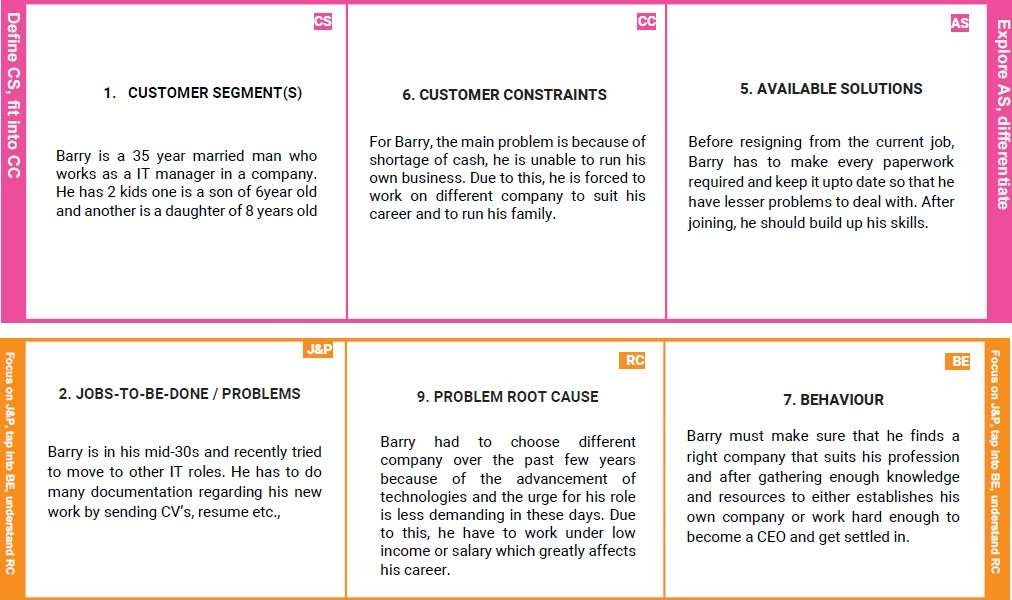
|  |  |  |
| --- | --- | --- |
| **S. No** | **Parameter** | **Description** |
| 1. | Problem Statement  (Problem to be solved) | Employee attrition is a major cost to an |
|  |  | organization and |
|  |  | predicting such |
|  |  | attritions is the most |
|  |  | important requirement |
|  |  | of the Human |
|  |  | Resources department |
|  |  | in many organizations. |
|  |  | In this problem, your |

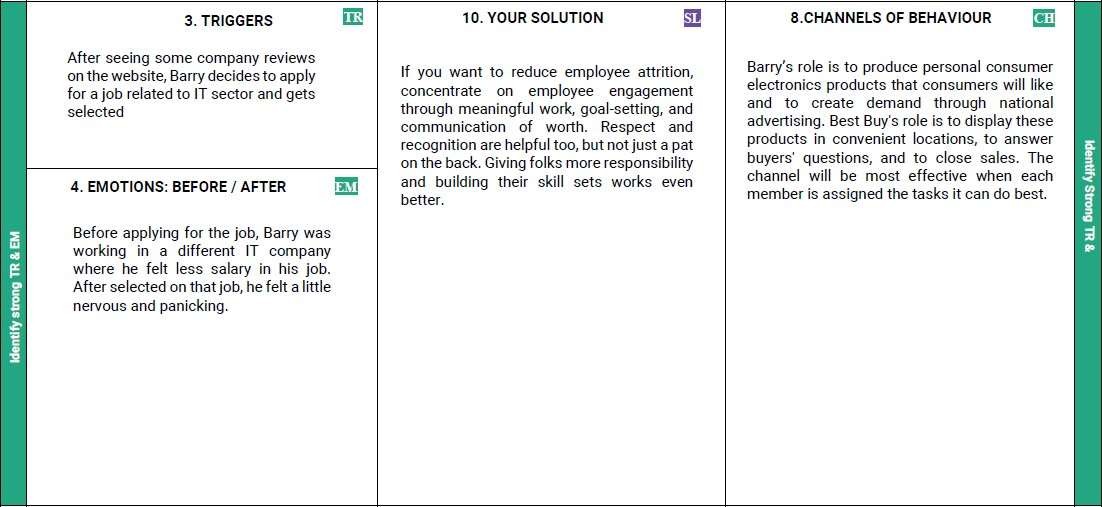
|  |  |  |
| --- | --- | --- |
|  |  | task is to predict the attrition rate of employees of an  organization. |
| 2. | Idea / Solution  description | An employee listening perspective will answer |
|  |  | the question of why. We |
|  |  | can look at what |
|  |  | employees who left |
|  |  | were telling us about |
|  |  | the workplace, work |
|  |  | relationships, and their |
|  |  | sense of connection to |
|  |  | the organization in the |
|  |  | months before they left. |
|  |  | The comparison of |
|  |  | engagement survey data |
|  |  | to termination data can |
|  |  | reveal areas of the |
|  |  | employee experience in |
|  |  | need of improvement. |

|  |  |  |
| --- | --- | --- |
| 3. | Novelty / Uniqueness | A focus on attrition drivers for top talent is particularly important  for many organizations. In these cases, there is often more latitude for interventions; employees in these positions often have unique experience, are highly skilled, If, for example to become caregivers – either for children or elderly relatives – the intervention may be offering a flexible work schedule, the option to work from home, or other changes in the way they work that will allow them to balance caregiving duties with work. |

**TABLE 4.1**

# 4.2 PROBLEM SOLUTION FIT





**4.3 SYSTEM ARCHITECTURE**

There are three fundamental components in CRM:

Operational - automation of basic business processes (marketing, sales, service)  
Analytical - analysis of customer data and behavior using business intelligence

Collaborative - communicating with clients

The Deliverable shall include the architectural diagram as below and the information as per the table 5.1 & table 5.2.

## Corporate Employee Attrition Analytics

## Guidelines:

The first step in any analytics project is to clearly define the problem that you are trying to solve.

The next step is to collect data about your employees.

Once you have collected the data, you need to pre-process it to prepare it for analysis

The next step is to perform exploratory data analysis (EDA) to gain insights into the data.

Feature engineering involves creating new features from the existing data that can improve the predictive power of your model. For example, you may want to create a feature that measures employee satisfaction based on survey responses.

Once you have pre-processed the data and performed EDA, you can build a predictive model. This can involve using machine learning algorithms such as logistic regression, decision trees, or random forests to predict employee attrition.

After building the model, you need to evaluate its performance using metrics such as accuracy, precision, recall, and F1-score.

Once you have developed and evaluated the model, you can deploy it in a production environment.

Finally, it is important to monitor the performance of the model over time and update it as needed to ensure that it continues to provide accurate prediction

**CHAPTER 5**

**PROJECT DESIGN PHASE II:**

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer’s goal is to produce a model or representation of an entity that will later be built.

Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software. The importance can be stated with a single word “Quality”. Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality.

Design is the only way that we can accurately translate a customer’s view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

**5.1 SOLUTION REQUIREMENTS**

**HARDWARE REQUIREMENTS: **

P4 2.8GB processor and above. 

Ram 512 MB and above. 

HDD 20 GB Hard Disk and above.

**SOFTWARE REQUIREMENTS:**

Microsoft .Net framework 2.0

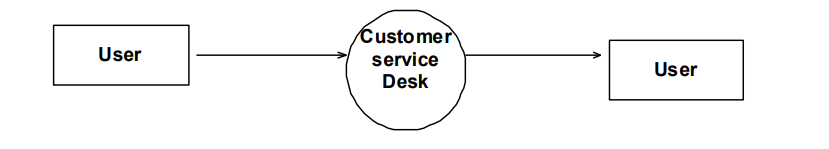
Microsoft ASP.Net.

AJAX Tool kit.

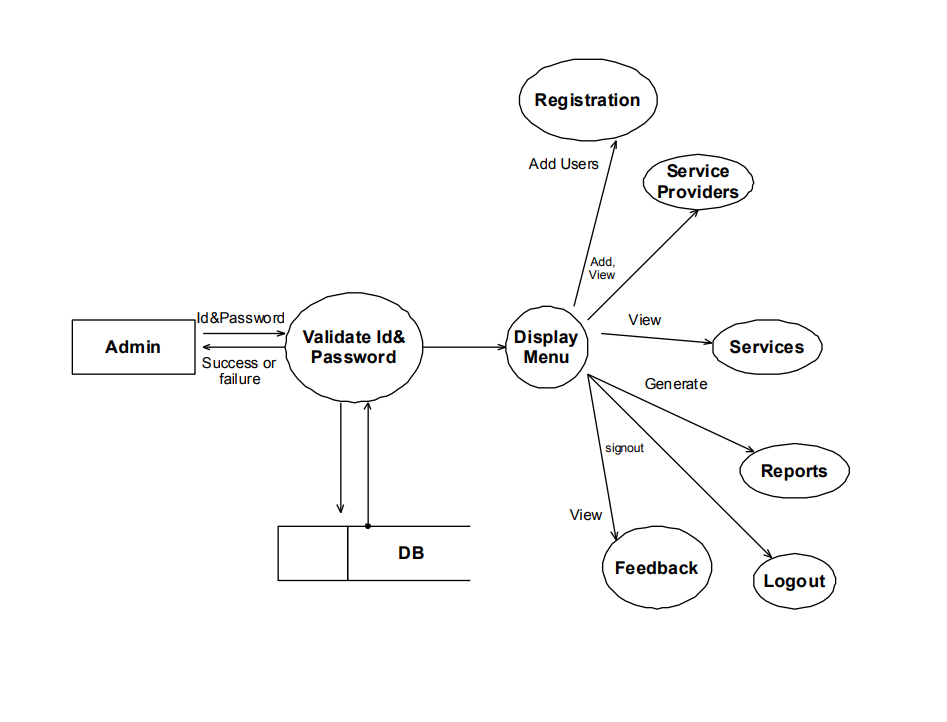
Microsoft C#.Net language.

Microsoft SQL Server 2005 o HTML

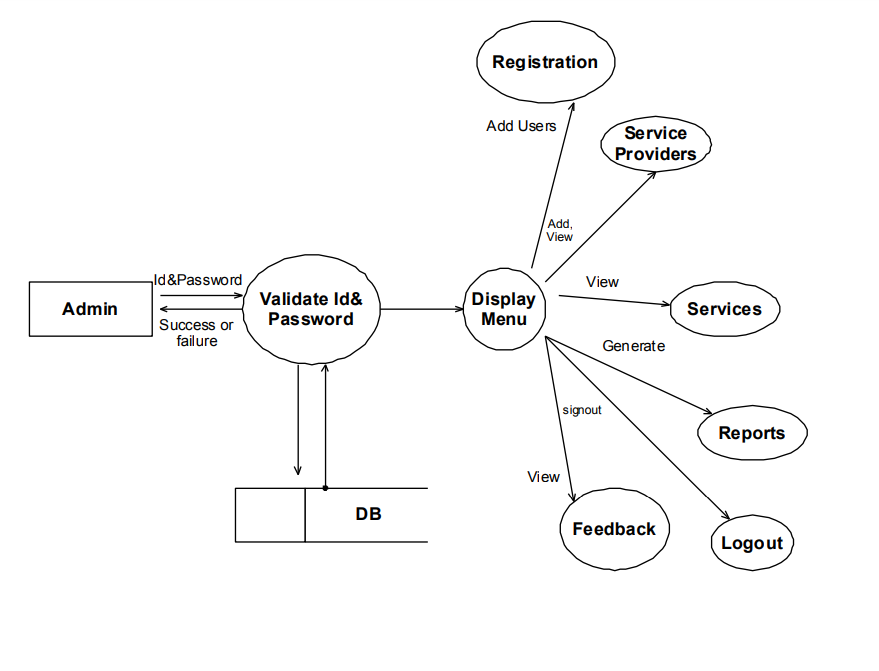
## 5.2 DATA FLOW DIAGRAMS



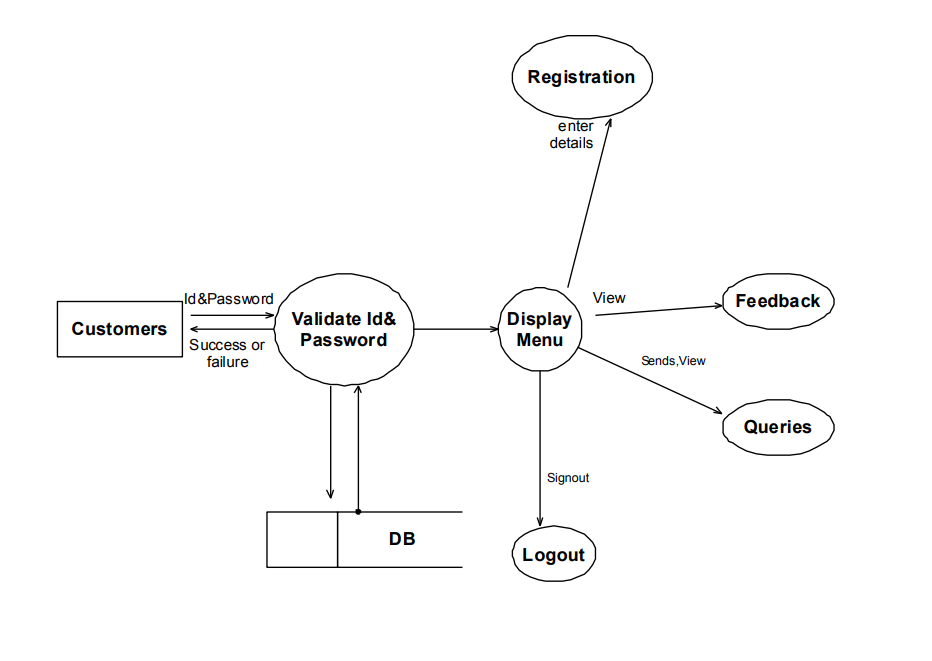
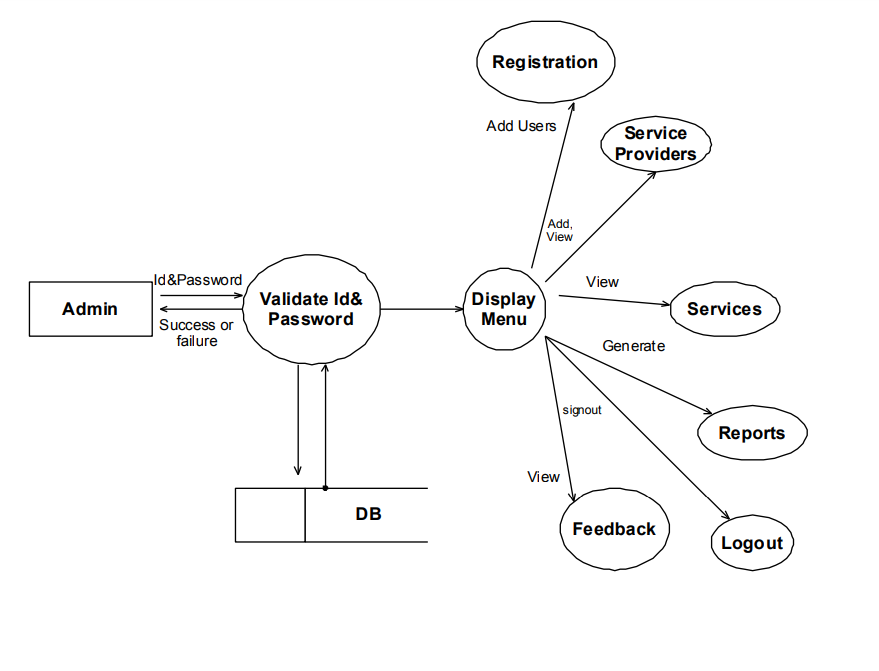
**Fig 5.1 Level 0 Diagram**



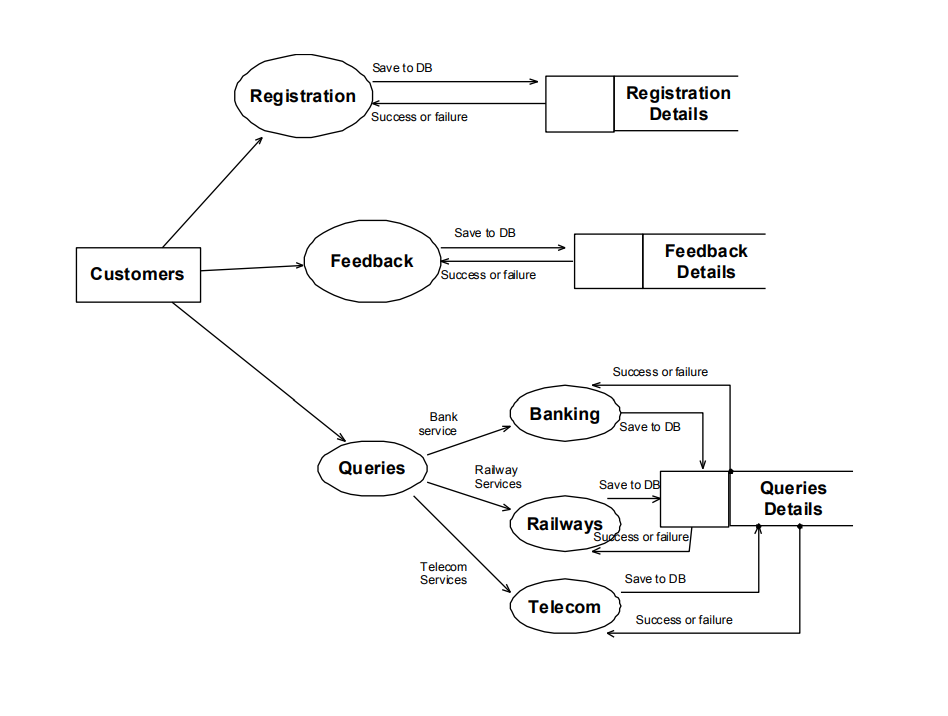
**Fig 5.2 Level 1 for admin**



**Fig 5.3 Level 1 for customers**

**Fig 5.4 Level 1 for service providers**

## Fig 5.5 Level 2 for admin

 **Fig 5.6 Level 2 for Customers**

## 5.3 TECHNOLOGY STACK

# Functional requirement

Following are the functional requirements of the proposed solution

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story/Sub- Task)** |
| FR -1 | User Registration | Registration through Form Registration through Gmail Registration through LinkedIn |
| FR -2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR -3 | Account Creation | Create an account in the Profile Dashboard |
| FR -4 | Input Credentials | Uploading your dataset Analyzing the attrition rate using dashboard |
| FR -5 | Processing Methods | Using IBM Cognos Analytics Dashboard  Using prediction algorithm to find  attrition rate |
| FR -6 | Output Credentials | Using the Dashboard and Algorithm they know about the employee |

|  |  |  |
| --- | --- | --- |
|  |  | attrition and way to reduce the  employee attrition |
| FR -7 | Report Preparation | Record the outcome of the  algorithm in document. |

## Non-Functional Requirement

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement (Epic)** | **Description** |
| NFR -1 | Usability | The user can be able to interact with the system user friendly. The system is build with a simple modules and  algorithms. |
| NFR -2 | Security | Access permissions for the particular system information may only be changed by the system’s  data administrator. The user’s data must be having an high security measures, |
| NFR -3 | Reliability | The database update process must roll back all related updates when any update fails. The dataset will not be modified by anyone only the user can be able to modify the dataset. |

|  |  |  |
| --- | --- | --- |
| NFR -  4 | Performance | The performance of the dashboard  flexible to every user’s. The front- page load time must be no more than 2 seconds for users that access  the website using an LTE mobile connection. |
| NFR -5 | Availability | New module deployment mustn’t  impact frontpage, dashboard and check out pages availability and mustn’t take longer than one hour. The rest of the pages that may experience problems must display a notification with a timer showing when the system is going to be up again. |
| NFR -6 | Scalability | The website attendance limit must  be scalable enough to support 2000,000 users at a time. The dashboard is scalable for the companies when their employee’s dataset is used for analysis. The model can successfully predict the futuristic approach and suggests preventive measures. |

**Table 5.1**

**CHAPTER 6**

**6.1 PROJECT DEVELOPMENT 1**

**CLIENT APPLICATION DEVELOPMENT**

Client applications are the closest to a traditional style of application in Windows-based programming. These are the types of applications that display windows or forms on the desktop, enabling a user to perform a task.

Client applications include applications such as word processors and spreadsheets, as well as custom business applications such as data-entry tools, reporting tools, and so on. Client applications usually employ windows, menus, buttons, and other GUI elements, and they likely access local resources such as the file system and peripherals such as printers.

Another kind of client application is the traditional ActiveX control (now replaced by the managed Windows Forms control) deployed over the Internet as a Web page. This application is much like other client applications: it is executed natively, has access to local resources, and includes graphical elements.

In the past, developers created such applications using C/C++ in conjunction with the Microsoft Foundation Classes (MFC) or with a rapid application development (RAD) environment such as Microsoft® Visual Basic®.

The .NET Framework incorporates aspects of these existing products into a single, consistent development environment that drastically simplifies the development of client applications.

The Windows Forms classes contained in the .NET Framework are designed to be used for GUI development. You can easily create command windows, buttons, menus, toolbars, and other screen elements with the flexibility necessary to accommodate shifting business needs.

82 For example, the .NET Framework provides simple properties to adjust visual attributes associated with forms.

In some cases the underlying operating system does not support changing these attributes directly, and in these cases the .NET Framework automatically recreates the forms. This is one of many ways in which the .NET Framework integrates the developer interface, making coding simpler and more consistent.

Unlike ActiveX controls, Windows Forms controls have semi-trusted access to a user's computer. This means that binary or natively executing code can access some of the resources on the user's system (such as GUI elements and limited file access) without being able to access or compromise other resources. Because of code access security, many applications that once needed to be installed on a user's system can now be safely deployed through the Web.

Your applications can implement the features of a local application while being deployed like a Web page. 6.2 ASP.NET Server Application Development Server-side applications in the managed world are implemented through runtime hosts. Unmanaged applications host the common language runtime, which allows your custom managed code to control the behavior of the server.

This model provides you with all the features of the common language runtime and class library while gaining the performance and scalability of the host server. The following illustration shows a basic network schema with managed code running in different server environments. Servers such as IIS and SQL Server can perform standard operations while your application logic executes through the managed code.

**.NET FRAMEWORK CLASS LIBRARY**

The .NET Framework class library is a collection of reusable types that tightly integrate with the common language runtime. The class library is object oriented, providing types from which your own managed code can derive functionality.

This not only makes the .NET Framework types easy to use, but also reduces the time associated with learning new features of the .NET Framework. In addition, thirdparty components can integrate seamlessly with classes in the .NET Framework. For example, the .NET Framework collection classes implement a set of interfaces that you can use to develop your own collection classes.

Your collection classes will blend seamlessly with the classes in the .NET Framework. As you would expect from an object-oriented class library, the .NET Framework types enable you to accomplish a range of common programming tasks, including tasks such as string management, data collection, database connectivity, and file access. In addition to these common tasks, the class library 81 includes types that support a variety of specialized development scenarios.

**6.2 PROJECT DEVELOPMENT 2**

ASP.NET is the hosting environment that enables developers to use the .NET Framework to target Web-based applications. However, ASP.NET is more than just a runtime host; it is a complete architecture for developing Web sites and Internet-distributed objects using managed code.

Both Web Forms and XML Web services use IIS and ASP.NET as the publishing mechanism for applications, and both have a collection of supporting classes in the .NET Framework. XML Web services, an important evolution in Web-based technology, are distributed, server-side application components similar to common Web sites. However, unlike Web-based applications, XML 83 Web services components have no UI and are not targeted for browsers such as Internet Explorer and Netscape Navigator.

Instead, XML Web services consist of reusable software components designed to be consumed by other applications, such as traditional client applications, Web-based applications, or even other XML Web services. As a result, XML Web services technology is rapidly moving application development and deployment into the highly distributed environment of the Internet.

If you have used earlier versions of ASP technology, you will immediately notice the improvements that ASP.NET and Web Forms offers. For example, you can develop Web Forms pages in any language that supports the .NET Framework. In addition, your code no longer needs to share the same file with your HTTP text (although it can continue to do so if you prefer). Web Forms pages execute in native machine language because, like any other managed application, they take full advantage of the runtime.

In contrast, unmanaged ASP pages are always scripted and interpreted. ASP.NET pages are faster, more functional, and easier to develop than unmanaged ASP pages because they interact with the runtime like any managed application.

The .NET Framework also provides a collection of classes and tools to aid in development and consumption of XML Web services applications. XML Web services are built on standards such as SOAP (a remote procedure-call protocol), XML (an extensible data format), and WSDL ( the Web Services Description Language).

The .NET Framework is built on these standards to promote interoperability with non-Microsoft solutions. For example, the Web Services Description Language tool included with the .NET Framework SDK can query an XML Web service published on the Web, parse its WSDL description, and produce C# or Visual Basic source code that your application can use to become a client of the XML Web service.

The source code can create classes derived from classes in the class library that handle all the underlying communication using SOAP and XML parsing. Although you can use the class library to consume XML Web services directly, the Web Services Description Language tool and the other tools contained in the SDK facilitate your development efforts with the .NET Framework.

If you develop and publish your own XML Web service, the .NET Framework provides a set of classes that conform to all the underlying communication standards, such as SOAP, WSDL, and XML. Using those classes enables you to focus on the logic of your service, without concerning yourself with the communications infrastructure required by distributed software development. Finally, like Web Forms pages in the managed environment, your XML Web service will run with the speed of native machine language using the scalable communication of IIS.

**ACTIVE SERVER PAGES.NET**

ASP.NET is a programming framework built on the common language runtime that can be used on a server to build powerful Web applications. ASP.NET offers several important advantages over previous Web development models:

**Enhanced Performance**  ASP.NET is compiled common language runtime code running on the server. Unlike its interpreted predecessors, ASP.NET can take advantage of early binding, just-intime compilation, native optimization, and caching services right out of the box. This amounts to dramatically better performance before you ever write a line of code.

**World-Class Tool Support.** The ASP.NET framework is complemented by a rich toolbox and designer in the Visual Studio integrated development environment. WYSIWYG editing, drag-and-drop server controls, and automatic deployment are just a few of the features this powerful tool provides. 

**Power and Flexibility.** Because ASP.NET is based on the common language runtime, the power and flexibility of that entire platform is available to Web application developers. The .NET Framework class library, Messaging, and Data Access solutions are all seamlessly accessible from the Web. ASP.NET is also language-independent, so you can choose the language that best applies to your application or partition your application across many languages.

Further, common language runtime interoperability guarantees that your existing investment in COM-based development is preserved when migrating to ASP.NET.



**Simplicity.** ASP.NET makes it easy to perform common tasks, from simple form submission and client authentication to deployment and site configuration. For example, the ASP.NET page framework allows you to build user interfaces that cleanly separate application logic from presentation code and to handle events in a simple, Visual Basic - like forms processing model. Additionally, the common language runtime simplifies

development, with managed code services such as automatic reference counting and garbage collection.

 **Manageability.** ASP.NET employs a text-based, hierarchical configuration system, which simplifies applying settings to your server environment and Web applications. Because configuration information is stored as plain text, new settings may be applied without the aid of local administration tools. This "zero local administration" philosophy extends to deploying ASP.NET Framework applications as well. An ASP.NET Framework application is deployed to a server simply by copying the necessary files to the server. No server restart is required, even to deploy or replace running compiled code

# **Scalability and Availability.** ASP.NET has been designed with scalability in mind, with features specifically tailored to improve performance in clustered and multiprocessor environments. Further, processes are closely monitored and managed by the ASP.NET runtime, so that if one misbehaves (leaks, deadlocks), a new process can be created in its place, which helps keep your application constantly available to handle requests.

# **Customizability and Extensibility.** ASP.NET delivers a well-factored architecture that allows developers to "plug-in" their code at the appropriate level. In fact, it is possible to extend or replace any subcomponent of the ASP.NET runtime with your own custom-written component.

# **Implementing custom authentication** or state services has never been easier.  Security. With built in Windows authentication and per-application configuration, you can be assured that your applications are secure.

# CHAPTER 7

# CODING & SOLUTIONING

**7.1 FEATURE CODE 1**

from google.colab import drive drive. Mount('/content/drive') #GENERAL

import pandas as pd import numpy as np import seaborn as sns

import matplotlib.pyplot as plt

#FEATURE ENGINEERING

from sklearn.preprocessing import LabelEncoder from imblearn.over\_sampling import SMOTE

from sklearn.model\_selection import train\_test\_split from sklearn.preprocessing import StandardScaler #MODEL SELECTION

from sklearn.model\_selection import KFold

from sklearn.model\_selection import cross\_val\_score from sklearn.model\_selection import GridSearchCV #MODEL from sklearn.linear\_model import LogisticRegression from

sklearn.ensemble import RandomForestClassifier from sklearn.svm import SVC

from sklearn.tree import DecisionTreeClassifier

#MODEL SCORES

from sklearn.metrics import confusion\_matrix , accuracy\_score,classification\_report

#FEATURE IMPORTANCE

from sklearn.inspection import permutation\_importance

path = '/content/drive/MyDrive/Colab Notebooks/HR-Employee- Attrition.csv' df =pd.read\_csv(path)

df

df.shape df.info()

df.select\_dtypes('int64' ,'float64').columns

cat\_cols = df.select\_dtypes('object').columns cat\_cols df.describe().T df

for cat in cat\_cols:

print(cat ,'-> ' , df[cat].unique()) print() print("All columns Unique values count") for col in df:

print(col, len(df[col].unique()), sep=': ')

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot(df['Attrition'] ,color ='b' ,hue =df['Gender']) plt.title('Attrition by Gender')

plt.subplot(1,2,2)

plt.pie(df['Attrition'].value\_counts() ,colors =['r' ,'c'] ,explode

=[0,0.1]

,autopct = '%.2f' ,labels =['No' ,'Yes']) plt.title('Attrition') plt.figure(figsize =(16 ,4)) plt.subplot(1,3,1) sns.distplot(df['Age']

,color ='m') plt.title('Age') plt.subplot(1 , 3 ,2)

sns.stripplot(x = 'Gender' ,y = 'Age' ,data = df ,palette="Set2") plt.title('Gender vs Age')

plt.subplot(1,3,3)

sns.countplot('Gender' ,data = df ,color ='c') plt.title('Gender') plt.tight\_layout()

plt.figure(figsize = (14 , 13))

plt.subplot(2 ,1,1)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r') plt.title('JOB ROLE')

plt.subplot(2,1,2)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r'

,hue=df['Attrition'])

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot('Department' ,data = df ,hue

='Attrition',palette='gist\_rainbow\_r') plt.subplot(1,2,2) plt.pie(df['Department'].value counts() ,autopct ='%.2f' ,colors = ['r'

,'c','g'],labels =['Research & Development’, ‘Sales', 'Human Resources'] ,explode=[0 ,0.1,0])

#HANDLING CATEGORICAL OUTPUT VARIABLE

df['Attrition'].replace({'Yes':1 ,'No':0} ,inplace = True) df['Attrition'].head()

plt.figure(figsize =(14 ,10)) plt.subplot(2,2,1) sns.countplot(df['JobSatisfaction'] ,hue =df['Attrition']

,palette='Accent\_r') plt.subplot(2,2,2) sns.countplot(df['EnvironmentSatisfaction'] ,hue

=df['Attrition'],palette='Accent') plt.subplot(2,2,3) sns.countplot(df['JobInvolvement'] ,hue =df['Attrition']

,palette='brg\_r') plt.subplot(2,2,4) sns.countplot(df['PerformanceRating'] ,hue

=df['Attrition'],palette='twilight\_r') plt.figure(figsize =(20 ,8))

sns.boxplot(x ='JobRole', y = 'MonthlyIncome' ,data = df ,hue

='Attrition',color ='red')

lt.figure(figsize =(12,10)) plt.subplot(2,1,1)

sns.boxplot(x = 'MaritalStatus' ,y ='RelationshipSatisfaction' ,data = df ,hue= 'Attrition', color = 'g') plt.subplot(2,1,2) sns.boxplot(df['JobLevel'],df['MonthlyIncome'] ,hue = df['Attrition'],palette='Reds\_r')

print(col, len(df[col].unique()), sep=': ')

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot(df['Attrition'] ,color ='b' ,hue =df['Gender']) plt.title('Attrition by Gender')

plt.subplot(1,2,2)

plt.pie(df['Attrition'].value\_counts() ,colors =['r' ,'c'] ,explode

=[0,0.1]

,autopct = '%.2f' ,labels =['No' ,'Yes']) plt.title('Attrition') plt.figure(figsize =(16 ,4)) plt.subplot(1,3,1) sns.distplot(df['Age']

,color ='m') plt.title('Age') plt.subplot(1 , 3 ,2)

sns.stripplot(x = 'Gender' ,y = 'Age' ,data = df ,palette="Set2") plt.title('Gender vs Age')

plt.subplot(1,3,3)

sns.countplot('Gender' ,data = df ,color ='c') plt.title('Gender') plt.tight\_layout()

plt.figure(figsize = (14 , 13))

plt.subplot(2 ,1,1)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r') plt.title('JOB ROLE')

plt.subplot(2,1,2)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r'

,hue=df['Attrition'])

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot('Department' ,data = df ,hue

='Attrition',palette='gist\_rainbow\_r') plt.subplot(1,2,2) plt.pie(df['Department'].value counts() ,autopct ='%.2f' ,colors = ['r'

,'c','g'],labels =['Research & Development’, ‘Sales', 'Human Resources'] ,explode=[0 ,0.1,0])

print(col, len(df[col].unique()), sep=': ')

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot(df['Attrition'] ,color ='b' ,hue =df['Gender']) plt.title('Attrition by Gender')

plt.subplot(1,2,2)

plt.pie(df['Attrition'].value\_counts() ,colors =['r' ,'c'] ,explode

=[0,0.1]

,autopct = '%.2f' ,labels =['No' ,'Yes']) plt.title('Attrition') plt.figure(figsize =(16 ,4)) plt.subplot(1,3,1) sns.distplot(df['Age']

,color ='m') plt.title('Age') plt.subplot(1 , 3 ,2)

sns.stripplot(x = 'Gender' ,y = 'Age' ,data = df ,palette="Set2") plt.title('Gender vs Age')

plt.subplot(1,3,3)

sns.countplot('Gender' ,data = df ,color ='c') plt.title('Gender')

plt.tight\_layout()

plt.figure(figsize = (14 , 13))

plt.subplot(2 ,1,1)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r') plt.title('JOB ROLE')

plt.subplot(2,1,2)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r'

,hue=df['Attrition'])

,autopct = '%.2f' ,labels =['No' ,'Yes']) plt.title('Attrition') plt.figure(figsize =(16 ,4)) plt.subplot(1,3,1) sns.distplot(df['Age']

,color ='m') plt.title('Age') plt.subplot(1 , 3 ,2)

sns.stripplot(x = 'Gender' ,y = 'Age' ,data = df ,palette="Set2") plt.title('Gender vs Age')

plt.subplot(1,3,3)

sns.countplot('Gender' ,data = df ,color ='c') plt.title('Gender') plt.tight\_layout()

plt.figure(figsize = (14 , 13))

plt.subplot(2 ,1,1)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r') plt.title('JOB ROLE')

plt.subplot(2,1,2)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r'

,hue=df['Attrition'])

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot('Department' ,data = df ,hue

='Attrition',palette='gist\_rainbow\_r') plt.subplot(1,2,2) plt.pie(df['Department'].value counts() ,autopct ='%.2f' ,colors = ['r'

,'c','g'],labels =['Research & Development’, ‘Sales', 'Human Resources'] ,explode=[0 ,0.1,0])

print(col, len(df[col].unique()), sep=': ')

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot(df['Attrition'] ,color ='b' ,hue =df['Gender']) plt.title('Attrition by Gender')

plt.subplot(1,2,2)

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot('Department' ,data = df ,hue

='Attrition',palette='gist\_rainbow\_r') plt.subplot(1,2,2) plt.pie(df['Department'].value counts() ,autopct ='%.2f' ,colors = ['r'

,autopct = '%.2f' ,labels =['No' ,'Yes']) plt.title('Attrition') plt.figure(figsize =(16 ,4)) plt.subplot(1,3,1) sns.distplot(df['Age']

,color ='m') plt.title('Age') plt.subplot(1 , 3 ,2)

sns.stripplot(x = 'Gender' ,y = 'Age' ,data = df ,palette="Set2") plt.title('Gender vs Age')

plt.subplot(1,3,3)

sns.countplot('Gender' ,data = df ,color ='c') plt.title('Gender') plt.tight\_layout()

plt.figure(figsize = (14 , 13))

plt.subplot(2 ,1,1)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r') plt.title('JOB ROLE')

plt.subplot(2,1,2)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r'

,hue=df['Attrition'])

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot('Department' ,data = df ,hue

='Attrition',palette='gist\_rainbow\_r') plt.subplot(1,2,2) plt.pie(df['Department'].value counts() ,autopct ='%.2f' ,colors = ['r'

,'c','g'],labels =['Research & Development’, ‘Sales', 'Human Resources'] ,explode=[0 ,0.1,0])

print(col, len(df[col].unique()), sep=': ')

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot(df['Attrition'] ,color ='b' ,hue =df['Gender']) plt.title('Attrition by Gender')

plt.subplot(1,2,2)

,'c','g'],labels =['Research & Development’, ‘Sales', 'Human

Resources'] ,explode=[0 ,0.1,0])

col = ['YearsInCurrentRole' ,'YearsSinceLastPromotion'

,'YearsWithCurrManager','YearsAtCompany'] plt.figure(figsize = (10 ,10))

for i,c in enumerate(col): plt.subplot(2 ,2,i+1) sns.distplot(df[c]

,color ='b') plt.figure(figsize = (16 ,16))

sns.heatmap(df.corr() ,cmap = 'ocean' , cbar = True , annot = True) no\_use = []

for col in df.columns:

if(len(df[col].unique()) ==1): no\_use.append(col) no\_use

df.drop(columns = no\_use , axis = 1 , inplace = True) df.columns y\_n\_type = [] others =[]

,autopct = '%.2f' ,labels =['No' ,'Yes']) plt.title('Attrition') plt.figure(figsize =(16 ,4)) plt.subplot(1,3,1) sns.distplot(df['Age']

,color ='m') plt.title('Age') plt.subplot(1 , 3 ,2)

sns.stripplot(x = 'Gender' ,y = 'Age' ,data = df ,palette="Set2") plt.title('Gender vs Age')

plt.subplot(1,3,3)

sns.countplot('Gender' ,data = df ,color ='c') plt.title('Gender') plt.tight\_layout()

plt.figure(figsize = (14 , 13))

plt.subplot(2 ,1,1)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r') plt.title('JOB ROLE')

plt.subplot(2,1,2)

sns.countplot(y= 'JobRole' ,data = df ,palette='winter\_r'

,hue=df['Attrition'])

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot('Department' ,data = df ,hue

='Attrition',palette='gist\_rainbow\_r') plt.subplot(1,2,2) plt.pie(df['Department'].value counts() ,autopct ='%.2f' ,colors = ['r'

,'c','g'],labels =['Research & Development’, ‘Sales', 'Human Resources'] ,explode=[0 ,0.1,0])

print(col, len(df[col].unique()), sep=': ')

plt.figure(figsize =(14,5)) plt.subplot(1,2,1) sns.countplot(df['Attrition'] ,color ='b' ,hue =df['Gender']) plt.title('Attrition by Gender')

plt.subplot(1,2,2)

for col in df.select\_dtypes('object').columns: if(len(df[col].unique()) ==2): y\_n\_type.append(col) y\_n\_type

df['Gender'].replace({'Male':1 ,'Female':0} ,inplace = True)

df['OverTime'].replace({'Yes':1 ,'No':0} ,inplace = True) others = df.select\_dtypes('object').columns others

le = LabelEncoder() for col in others:

df[col] = le.fit\_transform(df[col]) df.select\_dtypes('object').columns x = df.drop('Attrition' ,axis =1) y = df['Attrition']

print(x.shape ,y.shape) sns.countplot(df['Attrition']) (df.Attrition.value\_counts()/1470)\*100

sns.countplot(df['JobInvolvement'] ,hue =df['Attrition']

,palette='brg\_r') plt.subplot(2,2,4) sns.countplot(df['PerformanceRating'] ,hue

=df['Attrition'],palette='twilight\_r') plt.figure(figsize =(20 ,8))

sns.boxplot(x ='JobRole', y = 'MonthlyIncome' ,data = df ,hue

='Attrition',color ='red')

lt.figure(figsize =(12,10)) plt.subplot(2,1,1)

sns.boxplot(x = 'MaritalStatus' ,y ='RelationshipSatisfaction' ,data = df ,hue= 'Attrition', color = 'g') plt.subplot(2,1,2) sns.boxplot(df['JobLevel'],df['MonthlyIncome'] ,hue = df['Attrition'],palette='Reds\_r')

col = ['YearsInCurrentRole' ,'YearsSinceLastPromotion'

smote = SMOTE(sampling\_strategy='minority') x ,y = smote.fit\_resample(x ,y) print(x.shape ,y.shape) y.value\_counts()

sns.countplot(y ,palette='viridis') plt.title('Now Class is Balanced') x\_train , x\_test , y\_train ,y\_test = train\_test\_split(x , y, test\_size=0.2

, random\_state= 52) print(x\_train.shape)

#scaling the data

sc = StandardScaler()

x\_train = sc.fit\_transform(x\_train) x\_test = sc.transform(x\_test) x\_train

k = KFold(n\_splits = 5) lr\_model = LogisticRegression() lr\_score = cross\_val\_score(lr\_model , x\_train , y\_train ,cv = k

,scoring ='neg\_mean\_squared\_error') lr\_score.mean() rf\_model = RandomForestClassifier()

rf\_score = cross\_val\_score(rf\_model , x\_train , y\_train ,cv = k

,scoring ='neg\_mean\_squared\_error') rf\_score.mean() svm\_model = SVC()

svm\_score = cross\_val\_score(svm\_model , x\_train , y\_train ,cv = k

,scoring ='neg\_mean\_squared\_error') svm\_score.mean() dt\_model = DecisionTreeClassifier()

sns.countplot(df['JobInvolvement'] ,hue =df['Attrition']

,palette='brg\_r') plt.subplot(2,2,4) sns.countplot(df['PerformanceRating'] ,hue

=df['Attrition'],palette='twilight\_r') plt.figure(figsize =(20 ,8))

sns.boxplot(x ='JobRole', y = 'MonthlyIncome' ,data = df ,hue

='Attrition',color ='red')

lt.figure(figsize =(12,10)) plt.subplot(2,1,1)

sns.boxplot(x = 'MaritalStatus' ,y ='RelationshipSatisfaction' ,data =

df ,hue= 'Attrition', color = 'g') plt.subplot(2,1,2) sns.boxplot(df['JobLevel'],df['MonthlyIncome'] ,hue = df['Attrition'],palette='Reds\_r')

col = ['YearsInCurrentRole' ,'YearsSinceLastPromotion'

dt\_score = cross\_val\_score(dt\_model , x\_train , y\_train ,cv = k

,scoring ='neg\_mean\_squared\_error') dt\_score.mean() plt.figure(figsize = (14 , 6)) plt.subplot(1,2,1)

x = ['Logistic Regression','Random Forest' ,'Support Vector'

,'Decision Tree'] y = [lr\_score.mean() , rf\_score.mean()

,svm\_score.mean() , dt\_score.mean()] plt.title('Neg Mean square error for Models') sns.barplot(y,x,palette="viridis") plt.subplot(1,2,2)

plt.plot(x ,y,marker = 'o' ,color = 'r',mfc ='b' ,ms =8 ) plt.title('Neg Mean square error')

#we obtained less less -ve mena sq error for SVC and random forest #lets try building model with both of them

model\_params ={ 'RandomForestClassifier':

{

'model':RandomForestClassifier(), 'param':

{

'n\_estimators':[10 ,50 ,100,130], 'criterion':['gini' ,'entropy'], 'max\_depth':range(4,8,1),

'max\_features':['auto' ,'log2']

}

}, 'SVC':

{

model':SVC(), 'param':

{ 'C':[1,20],

'gamma':[1,0.1],

'kernel':['rbf']

}

}

}

scores =[]

for model\_name , mp in model\_params.items():

model\_sel = GridSearchCV(estimator= mp['model'] ,param\_grid= mp['param']

,cv = 4 ,return\_train\_score=False) model\_sel.fit(x\_train,y\_train) scores.append({

'model':model\_name, 'best\_score':model\_sel.best\_score\_, 'best\_params':model\_sel.best\_params\_

})

scores

svm\_model = SVC(C=20 ,gamma=0.1 ,kernel='rbf') svm\_model.fit(x\_train ,y\_train)

ytest\_pred = svm\_model.predict(x\_test)

ytrain\_pred = svm\_model.predict(x\_train) accuracy\_score(y\_test

,ytest\_pred)

sns.countplot(df['JobInvolvement'] ,hue =df['Attrition']

,palette='brg\_r') plt.subplot(2,2,4) sns.countplot(df['PerformanceRating'] ,hue

=df['Attrition'],palette='twilight\_r') plt.figure(figsize =(20 ,8))

sns.boxplot(x ='JobRole', y = 'MonthlyIncome' ,data = df ,hue

='Attrition',color ='red')

lt.figure(figsize =(12,10)) plt.subplot(2,1,1)

sns.boxplot(x = 'MaritalStatus' ,y ='RelationshipSatisfaction' ,data = df ,hue= 'Attrition', color = 'g') plt.subplot(2,1,2) sns.boxplot(df['JobLevel'],df['MonthlyIncome'] ,hue = df['Attrition'],palette='Reds\_r')

col = ['YearsInCurrentRole' ,'YearsSinceLastPromotion'

print(classification\_report(y\_test , ytest\_pred))

print(classification\_report(y\_train , ytrain\_pred)) sns.heatmap(confusion\_matrix(y\_test ,ytest\_pred) ,annot = True

,cmap ='ocean')

sns.heatmap(confusion\_matrix(y\_train ,ytrain\_pred) ,annot = True

,cmap='Spectral\_r')

from sklearn.inspection import permutation\_importance perm\_importance = permutation\_importance(svm\_model, x\_test, y\_test) perm\_importance

perm\_importance.importances\_mean df.columns

cols = ['Age', 'BusinessTravel', 'DailyRate', 'Department', 'DistanceFromHome', 'Education', 'EducationField', 'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate', 'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction', 'MaritalStatus',"MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked', 'OverTime', 'PercentSalaryHike',

'PerformanceRating',

'RelationshipSatisfaction', 'StockOptionLevel', 'TotalWorkingYears',

'TrainingTimesLastYear',

'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', 'YearsWithCurrManager']

features = np.array(cols) plt.figure(figsize = (14 ,10)) sorted\_idx = perm\_importance.importances\_mean.argsort() sns.barplot( perm\_importance.importances\_mean[sorted\_idx]

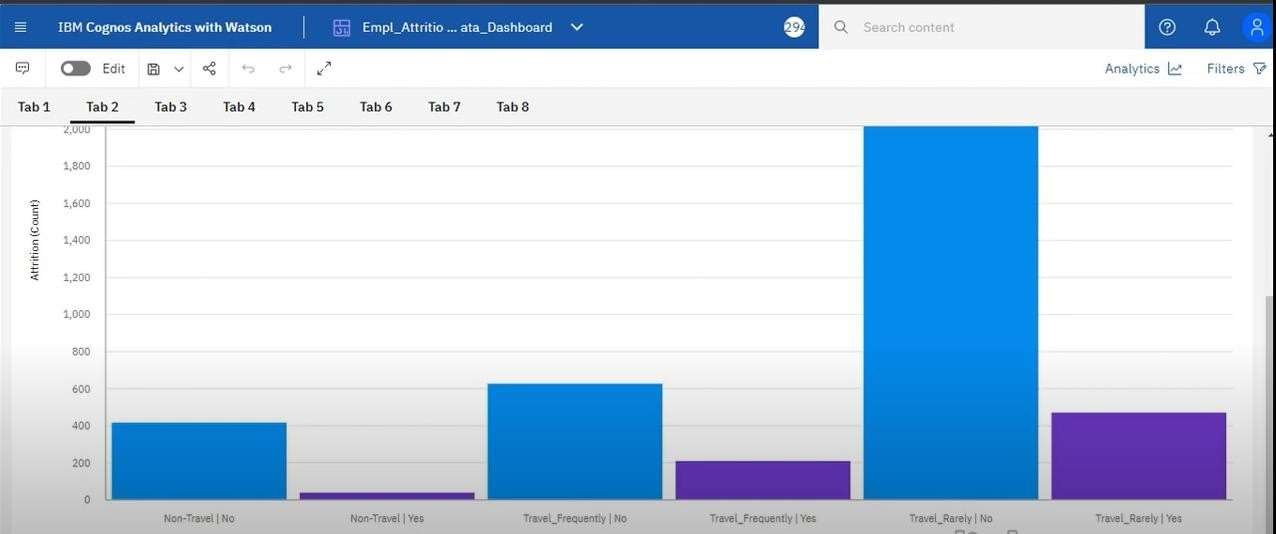
,features[sorted\_idx]

)

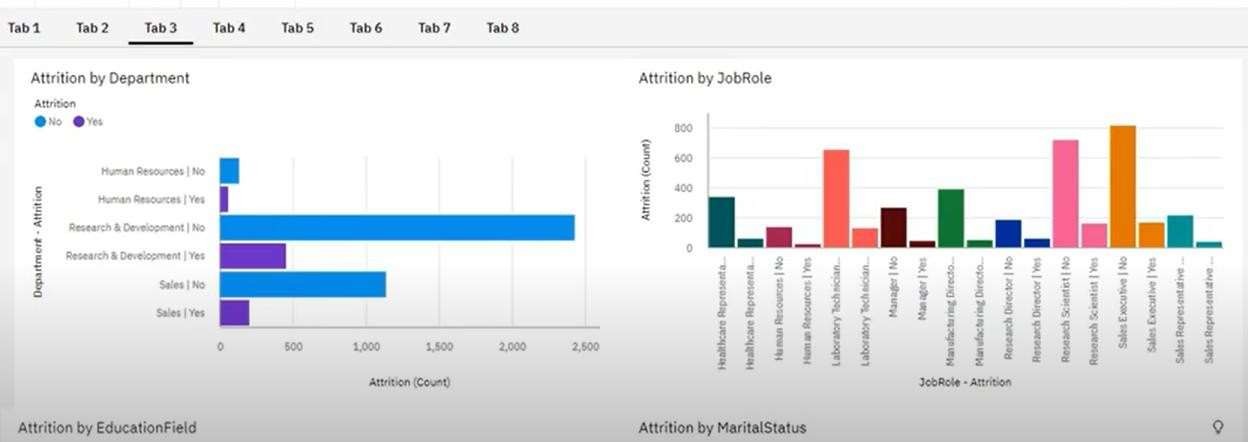
plt.xlabel("Permutation Importance") plt.title('FEATURE IMPORTANCE')

# CHAPTER 8

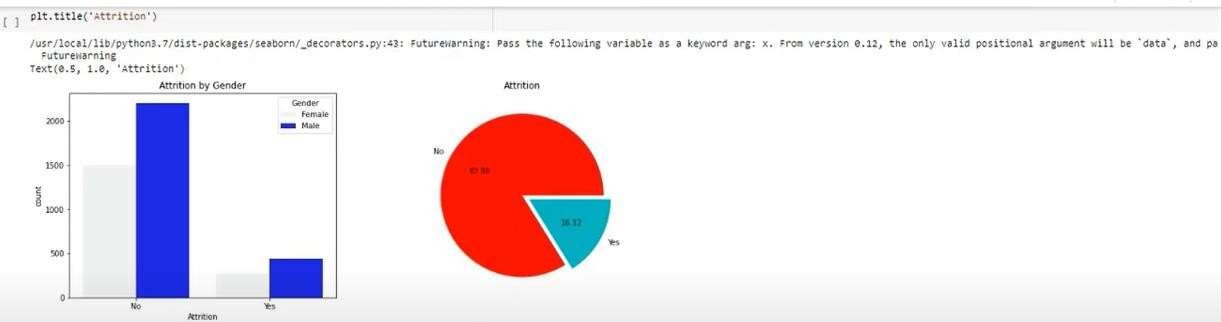
# RESULTS & SCREENSHOTS



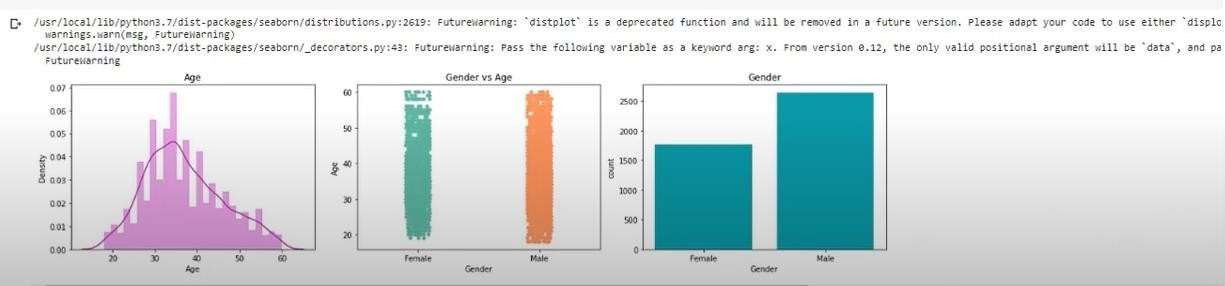
## Fig 8.1 Performance metrics



**Fig. 8.2 Employee Count by Department**



**Fig 8.3 Attrition Based on Salary Hike Percentage**



# Fig 8.4 Attrition Based on Department, Job Role and Education

**CHAPTER 9**

**CONCLUSION**

Employees as well as organizations must be clear with their expectations regarding the job profile. Any sort of mismatch leads to discrepancy and employees may fail to perform at their job. This eventually leads to attrition. Organizations should state the requirements and expectations unambiguously. This helps candidates decide upon to accept the job position or not. This eventually avoids further conflicts in the employment terms.

The attrition rate of the company is relatively high and should be addressed to retain key employees. The majority of employees who left the company cited job dissatisfaction as the main reason for their departure.

Improving job satisfaction should be a top priority for the company. A significant number of employees who left the company cited a poor work-life balance as a reason for their departure. The company should focus on providing more flexible work arrangements to help employees achieve a better work-life balance. The analysis revealed that there is a correlation between compensation and benefits and employee retention. The company should ensure that they are providing competitive compensation packages and benefits to their employees.

**CHAPTER 10**

**FUTURE ENHANCEMENT**

Research findings suggest that attrition reasons in IT organizations primarily revolve around professional growth and challenges in the organization. Although economic factors happen to the most influential factor, professionals may settle for second best criteria of their preference that is career growth and supportive work policies in the organization.

On the other hand, candidates who aspire to have a better job than the one in hand are more interested in securing the next job. Young talent wants to work on latest technology and functional domain. IT professionals who are young career makers are less influenced by Brand name or geographical area. Most of the IT professionals look for challenging role and position in the organization.

Candidates as well as senior professionals believe that challenging work motivate them to maintain the interest in the work life. Employees as well as organizations must be clear with their expectations regarding the job profile. Any sort of mismatch leads to discrepancy and employees may fail to perform at their job. This eventually leads to attrition.

Organizations should state the requirements and expectations unambiguously. This helps candidates decide upon to accept the job position or not. This eventually avoids further conflicts in the terms. Further this research can make more detailed conclusions over “mapping of candidates’ expectations with organizations’ requirement” by collecting the data focusing on all the steps of recruitment and selection process.

# 8.REFERENCES

[1]. From Big Data to Deep Data to support people analytics for employee attrition prediction, Nesrine Ben Yahia, Hlel Jihen, Ricardo Colomo- Palacio (2021)

[2]. Machine Learning Approach for Employee Attrition Analysis.Dr. R. S. Kamath | Dr. S. S. Jamsandekar | Dr. P. G. Naik, Published in International Journal of Trend in Scientific Research and Development (ijtsrd), (March 2019)

[3]. Investigation of early career teacher attrition (ECT) and the impact of

induction programs in Western Australia, Janine E. Wyatt, MichaelO’Neill (2021)

1. <https://www.researchgate.net/publication/323981576_Establishing_User-centric_Cloud_Service_Registries>
2. https://servicealliancegroup.com/blog/why-its-important-to-monitor-customer-service/?utm\_source=rss&utm\_medium=rss&utm\_campaign=why-its-important-to-monitor-customer-service#:~:text=Monitoring%20customer%20service%20involves%20identifying,is%20better%20customer%20experiences%20overall.
3. <https://www.researchgate.net/figure/Service-registry-architecture_fig1_323981576>
4. <https://www.sciencedirect.com/topics/computer-science/registry-architecture>
5. <https://microservices.io/patterns/service-registry.html>
6. E. Shafie, A. Al-Ajlan, K. Aldrawiesh, A. Bajahzar and Y. Al-Saawy, "Online registration system," 2011 3rd International Conference on Advanced Computer Control, Harbin, China, 2011, pp. 174-179, doi: 10.1109/ICACC.2011.6016392.
7. N. Goyal and H. Singh, "A Design of Customer Service Request Desk to Improve the Efficiency using Robotics Process Automation," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), Solan, India, 2021, pp. 21-24, doi: 10.1109/ISPCC53510.2021.9609338.