**GET PERFORMANCE MANAGEMENT SYSTEM**

A

**Major Project Report**

Submitted

In partial fulfillment

For the award of the Degree of

***BACHELOR OF TECHNOLOGY***

***In the Department of Computer science and Engineering***

**Submitted By: Guided By: Submitted To:**

Anam Quazi Mr. Akhilesh Arya Mr. Aaditya Maheshwari

18ETCCS007 Assistant Professor Project In-charge

4th Year CSE Department of CSE Department of CSE



**Department of Computer Science and Engineering**

**TECHNO INDIA NJR INSTITUTE OF TECHNOLOGY** RAJASTHAN TECHNICAL UNIVERSITY

**May 2022**



**CERTIFICATE**

This is to certify that this project report “**GET Performance Management System**” is the confide work of “**Anam Quazi**” who have carried out the project work under my supervision. I approve this project for submission of the Bachelor of Technology in the **Department of Computer Science and Engineering, Techno India NJR Institute of Technology**, affiliated to Rajasthan Technical University, Kota.

**Mr. Aditya Maheshwari**

**Assistant Professor, Project In-charge**

**Department of Computer Science**

#### **ABSTRACT**

#### **Purpose**

***1.1 Introduction***

This Software Requirements Specification provides a complete description of all the functions and specifications of the website GET Performance Management System.

The main objective of GET Performance Management System is to build a high-performance culture for both the individuals and the teams so that they jointly take the responsibility of improving the business processes on a continuous basis and at the same time raise the competence level by upgrading their own skills within a leadership framework.

###### ***1.1.2 Scope***

The scope of performance management should include the following-

1. Provide employees with a better understanding of their role and responsibilities.
2. Increase confidence through recognizing strengths while identifying training needs to improve weaknesses.
3. Improve working relations and communication between supervisors and subordinates.
4. Increase commitment to organizational goals; develop employees into future supervisors.
5. Assist in personnel decisions such as promotions or allocating rewards.
6. Allow time for self-reflection, self-appraisal and personal goal setting.

###### ***1.1.3 Document overview***

The remainder of this document is 8 chapters, the first providing introduction to the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter tells about the conclusion and future enhancements of the project. The final chapter concerns the bibliography.

#### **ACKNOWLEDGEMENT**

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed guide **Mr. Akhilesh Arya (Assistant Professor)** and **Mr. Aditya Maheshwari (Assistant Professor, Project In-charge)**, **TINJRIT** for their valuable guidance, encouragement, and help in completing this work. Their useful suggestions for this whole work and co-operative behavior are sincerely acknowledged. At the end, I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

I also wish to express my indebtedness to my parents as well as my family member whose blessings and support always helped me to face the challenges ahead.

At the end I would like to express my sincere thanks to all my friends and others who helped me directly or indirectly during this project work.

**ANAM QUAZI**

**Place - Udaipur**

**Date - 02/06/2022**

#### **TABLE OF CONTENTS**

**Table of Contents**

[Abstract. 3](#_TOC_250011)

[Acknowledgement 4](#_TOC_250010)

[Table of Contents 5-8](#_TOC_250009)

[List of Tables 9](#_TOC_250008)

[List of Figures 10](#_TOC_250007)

CHAPTER – 1 Introduction 11-12

Introduction 11

* 1. Purpose. 11
  2. Overall Description 11-12

CHAPTER – 2 Software Requirement Specification 13-17

* 1. [Purpose. 13](#_TOC_250006)
  2. [Overall Description. 14](#_TOC_250005)
     1. Functional Requirements definitions 15
     2. [Use Cases. 15](#_TOC_250004)
     3. Non – Functional Requirements. 14
  3. Requirement Specification 15
     1. External interface specifications. 15
     2. Functional Requirements 15
  4. Hardware Specification 16
  5. Software Specification. 16
  6. Hardware and Software Requirements in detail 17

CHAPTER – 3 System Analysis and Design 18-22

3.1. Study & Weaknesses of current system 18

3.2 Requirements of new system 18

* 1. Feasibility Study 19
  2. Features of new System 19
  3. Data flow diagram 20
  4. UML Modeling… 20
     1. Activity Diagram 21
     2. Context Diagram 22

CHAPTER – 4 Data Dictionary 23-28

* 1. [Table Details 23](#_TOC_250003)
  2. E-R Diagram 23-28

CHAPTER - 5 Screen Shots 29-37

* 1. Admin Master 29
  2. Parameter Master 30
  3. Employee Master 32
  4. Department Master 37
  5. Evaluation Master 35
  6. Group Master 37

CHAPTER – 6 Testing 38-43

CHAPTER – 7 Conclusion and Future Enhancements 44-45

* 1. [Limitations 44](#_TOC_250002)
  2. [Future Enhancements 44](#_TOC_250001)
  3. [Conclusion 45](#_TOC_250000)

CHAPTER – 8 Project Timeline… 44

CHAPTER – 9 Bibliography and Dependencies 45

### **List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No.** |
|  | **Use Cases** |  |
| Figure 2.2.2a | Access Home page | 10 |
| Figure 2.2.2b | User login or signup | 10 |
| Figure 2.2.2c | Admin selects, create a new Create parameter/employee/department/evaluation/group | 12 |
| Figure 2.2.2d | Admin, Head, Team Lead selects parameter, employee, date and enter marks for evaluation | 13 |
|  | **Data Flow Diagrams** |  |
| Figure 3.1 | DFD Level 0 | 25 |
| Figure 3.2 | Admin (DFD Level 1) | 26 |
| Figure 3.3 | Head (DFD Level 1) | 27 |
| Figure 3.4 | Team Lead (DFD Level 1) | 28 |
| Figure 3.5 | Employee (DFD Level ) | 29 |
|  | **UML Modeling** |  |
|  | *Activity Diagrams* |  |
| Figure 3.6 | For Admin | 34 |
| Figure 3.7 | For Head | 34 |
| Figure 3.8 | For Team Lead | 35 |
| Figure 3.9 | For Employee |  |
| Figure 3.10 | Context Diagram | 36 |
| Figure 3.11 | E-R Diagram | 45 |

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Table Name** | **Page No.** |
| Table 2.1 | Glossary | 7 |
|  | **Use Cases** |  |
| Table 2.2 | Access Home page | 15 |
| Table 2.3 | User login | 15 |
| Table 2.4 | Create parameter/employee/department/evaluation/group | 16 |
| Table 2.5 | Admin, Head, Team Lead selects parameter, employee, date and enter marks for evaluation | 18 |
|  | **Database Design** |  |
| Table 4.1 | User | 41 |
| Table 4.2 | Parameters | 42 |
| Table 4.3 | Evaluations | 42 |
| Table 4.4 | Departments | 42 |
| Table 4.5 | Groups | 42 |
| Table 4.6 | ParameterGroups | 43 |
| Table 4.7 | DepartmentParameters | 43 |
| Table 4.8 | Role | 44 |
| Table 4.9 | Status | 44 |
| Table 6.1 | Test report with test data | 71 |

**List of Symbols**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Admin | The only user who has the permission to  insert or update category etc. in the database. |
| Entry | Admin stored in the Database |
| Html | Hyper text markup language |
| IEEE | Institute of Electrical and Electronic  Engineers |
| QA | Quality assurance |
| SCMP | Software Configuration Management Plan |
| SDD | Software Design Document |

|  |  |
| --- | --- |
| SQAP | Software Quality Assurance Plan |
| SRS | Software Requirements Specification |

|  |  |
| --- | --- |
| Web Site | A place on the world wide web |

# ***CHAPTER – I***

# ***INTRODUCTION***

### **Introduction**

##### **1.1 Purpose**

###### ***1.1.1 Introduction***

This Software Requirements Specification provides a complete description of all the functions and specifications of the website GET Performance Management System.

The main objective of GET Performance Management System is to build a high-performance culture for both the individuals and the teams so that they jointly take the responsibility of improving the business processes on a continuous basis and at the same time raise the competence level by upgrading their own skills within a leadership framework.

###### ***1.1.2 Scope***

The scope of performance management should include the following-

1. Provide employees with a better understanding of their role and responsibilities.
2. Increase confidence through recognizing strengths while identifying training needs to improve weaknesses.
3. Improve working relations and communication between supervisors and subordinates.
4. Increase commitment to organizational goals; develop employees into future supervisors.
5. Assist in personnel decisions such as promotions or allocating rewards.
6. Allow time for self-reflection, self-appraisal and personal goal setting.

###### ***1.1.2 References***

[IEEE] The applicable IEEE standards are published in “IEEE Standards Collection,” 2001 edition.

[Bruade] The principal source of textbook material is “Software Engineering: An Object- Oriented Perspective” by Eric J. Bruade (Wiley 2001).

[Wiley] ASP.NET Bible by MridulaParihar.

###### ***Document overview***

The remainder of this document is 8 chapters, the first providing introduction of the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter tells about the conclusion and future enhancements of the project. The final chapter concerns with the bibliography.

This document is meant for describing all the features and procedures that were followed while developing the system.

This document specially mentions the details of the project how it was developed, the primary requirement, as well as various features and functionalities of the project and the procedures followed in achieving these objectives.

GET Performance management system can be defined as the development of individuals with competence and commitment, working towards the achievement of shared meaningful objectives within an organization which supports and encourages their achievement.

Performance management is a process which contributes to the effective management of individual and teams in order to achieve high levels of organisational performance.

Employees are the most significant resource of any business, and performance management reflect the organization’s commitment to developing this important resource of human capital. Performance management is a systematic ways of measuring, reviewing and analyzing employee performance over a given period of time, and using the information gathered to plan for the employee’s future with the organization. This periodical, impartial feedback is used to judge employee effectiveness as well as provide necessary development and training to improve the employee’s contributions to the organization’s goals.

This powerful managerial system should directly reflect the overall organization’s goals and objectives. The employee assessment should provide useful feedback about the employee’s contributions or lack of contributions toward these goals.

##### **1.2 Overall description**

**Performance management** is a way of systematically managing people for [innovation](about:blank), goal focus, productivity and satisfaction. It is a goal congruent win- win strategy. Performance management is a holistic, largely participatory and goal congruent process of managing and supervising managers at work. It is understood as a systematic, organized approach to managing and [rewarding performance](about:blank) by generating and sustaining positive employee motivation. Its salient dimensions include performance standards- representing organizational goals and objectives, employee recognition and reward. Performance management is a means of getting better results from the organizations, teams and individuals by understanding and managing performance within the agreed framework of planned goals and competency requirements. It is a process for establishing shared understanding about what is to be achieved and an approach to managing and developing people.

###### ***Functional requirements definitions***

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Nonfunctional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

###### ***Use cases***

This system will be used in four User Modules which are Administrator, Head of Department, Team Lead and Employee. As all of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of all four User Modules have been described below.

1. User can do the following functions in the Administrator Module
   * Create, Edit & Delete Parameter
   * Create, Edit & Delete Department
   * Create, Edit & Delete Employee
   * Create, Edit & Delete Groups
   * Evaluate Employee based on Parameters
   * View Department Structure
   * View Evaluation
   * View Top Employees
   * Find Employee based on Group
2. User can do the following functions in the Head Module
   * Evaluate Employee based on Parameters
   * View Department
   * View Department Structure
   * View Evaluation
   * View Top Employees
   * Find Employee based on Group
3. User can do the following functions in the Team Lead Module
   * Evaluate Employees based on Parameters
   * View Department
   * View Department Structure
   * View Evaluation
   * View Top Employees
   * Find Employee based on Group
4. User can do the following functions in the Employee Module
   * View Department
   * View Department Structure
   * View Top Employees
   * Find Employee based on Group
     1. ***User characteristics***

The user should be familiar with the Employee Management related terminology like Evaluation/Employee List/Department Structure etc.

The user should be familiar with the Internet.

#### **Constraints**

Limited to HTTP/HTTPS.

Limited to Windows OS.

No multilingual support.

# ***CHAPTER – II***

# ***SOFTWARE REQUIREMENT SPECIFICATION***

##### **1.1 Purpose**

###### ***1.1.1 Introduction***

This Software Requirements Specification provides a complete description of all the functions and specifications of the website GET Performance Management System.

The main objective of GET Performance Management System is to build a high-performance culture for both the individuals and the teams so that they jointly take the responsibility of improving the business processes on a continuous basis and at the same time raise the competence level by upgrading their own skills within a leadership framework.

###### ***1.1.2 Scope***

The scope of performance management should include the following-

1. Provide employees with a better understanding of their role and responsibilities.
2. Increase confidence through recognizing strengths while identifying training needs to improve weaknesses.
3. Improve working relations and communication between supervisors and subordinates.
4. Increase commitment to organizational goals; develop employees into future supervisors.
5. Assist in personnel decisions such as promotions or allocating rewards.
6. Allow time for self-reflection, self-appraisal and personal goal setting.

##### **Glossary**

##### ***Table 2.1***

|  |  |
| --- | --- |
| ***TERM*** | ***Definition*** |
| *Admin* | Can monitor sales, view reports, assign roles |
| *Manager* | Can add or remove data |

|  |  |
| --- | --- |
| Salesman | Can view inventory, and sell/reserve items |
| QA | Quality Assurance |
| SCMP | Software Configuration Management Plan |
| SDD | Software Design Document |
| SQAP | Software Quality Assurance Plan |
| SRS | Software Requirement Specification |

###### ***References***

[IEEE] The applicable IEEE standards are published in “IEEE Standards Collection,” 2001 edition.

[Bruade] The principal source of textbook material is “Software Engineering: An Object-Oriented Perspective” by Eric J. Bruade (Wiley 2001).

###### ***Document Overview***

The remainder of this document is 8 chapters, the first providing introduction to the project. It lists all the functions performed by the system. The second chapter consists of software requirements specification. The third chapter provides details about system analysis and design. The fourth chapter gives data dictionary information. The fifth chapter consists of snapshots of the complete project. The sixth chapter gives testing for the project. The seventh chapter tells about the conclusion and future enhancements of the project. The final chapter concerns with the bibliography.

##### **1.2 Overall description**

**Performance management** is a way of systematically managing people for [innovation](about:blank), goal focus, productivity and satisfaction. It is a goal congruent win- win strategy. Performance management is a holistic, largely participatory and goal congruent process of managing and supervising managers at work. It is understood as a systematic, organized approach to managing and [rewarding performance](about:blank) by generating and sustaining positive employee motivation. Its salient dimensions include performance standards- representing organizational goals and objectives, employee recognition and reward. Performance management is a means of getting better results from the organizations, teams and individuals by understanding and managing performance within the agreed framework of planned goals and competency requirements. It is a process for establishing shared understanding about what is to be achieved and an approach to managing and developing people.

###### **Functional requirements definitions**

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Nonfunctional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

###### **Use cases**

This system will be used in four User Modules which are Administrator, Head of Department, Team Lead and Employee. As all of these have different requirements the modules are designed to meet their needs and avoid any type of confusion. The Uses of all four User Modules have been described below.

[1]User can do the following functions in the Administrator Module

* + Create, Edit & Delete Parameter
  + Create, Edit & Delete Department
  + Create, Edit & Delete Employee
  + Create, Edit & Delete Groups
  + Evaluate Employee based on Parameters
  + View Department Structure
  + View Evaluation
  + View Top Employees
  + Find Employee based on Group

[2]User can do the following functions in the Head Module

* + Evaluate Employee based on Parameters
  + View Department
  + View Department Structure
  + View Evaluation
  + View Top Employees
  + Find Employee based on Group

[3]User can do the following functions in the Team Lead Module

* + Evaluate Employees based on Parameters
  + View Department
  + View Department Structure
  + View Evaluation
  + View Top Employees
  + Find Employee based on Group

[4]User can do the following functions in the Employee Module

* + View Department
  + View Department Structure
  + View Top Employees
  + Find Employee based on Group
    1. ***User characteristics***

The user should be familiar with the Employee Management related terminology like Evaluation/Employee List/Department Structure etc.

The user should be familiar with the Internet.

2.2.2a Use Case: Access Home Page

###### Uml_Diagram (7).jpg

**Fig. 2.1 Access Home Page**

Brief Description:

User uses the GET Performance Management System to access the home page.

Initial step-by-step description:

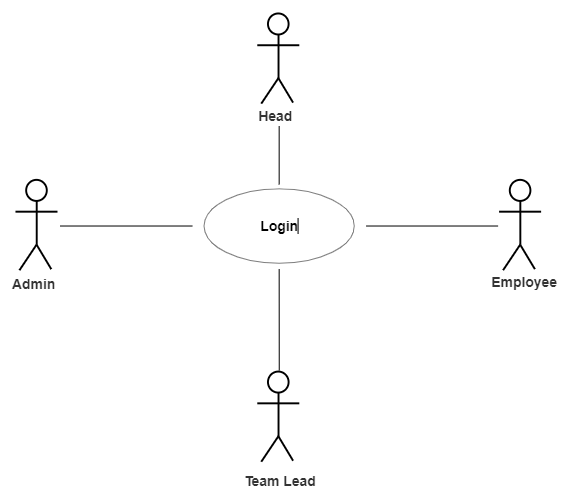
For this use case to be initiated, the user (admin, head, team lead, employee) can use the GET Performance Management System.

1. The user connects to the system using a web browser.

2.The user selects the Home link on the website home page.

3.The system passes the user to the website Home Page.

2.2.2b. Use Case: User Login or Signup



**Fig. 2.2 User Login or Signup**

Brief Description:

The user logs in.

Initial step-by-step description:

1.For this use case to be initiated the user must be on the website Home Page.

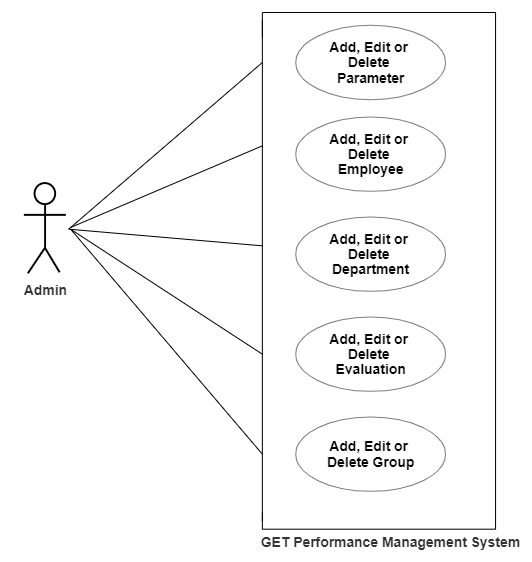
2.The User selects the “Login” link.

3.The User enters email, password and clicks enter.

4.The System retains information in the database.

5.The System returns the user to the website Home Page.

**2.2.2c Use Case: Create a New Parameter, Employee, Department, Evaluation, Group**



**Fig. 2.3 Admin Selects Create a New Parameter, Employee, Department, Evaluation, Group**

Brief Description:

The Admin chooses to create a new Parameter, Employee, Department, Evaluation, Group.

Initial step-by-step description:

For this use case to be initiated, the Admin must be on the website’s Parameter, Employee, Department, Evaluation, Group page.

1. The Admin selects the “Parameter, Employee, Department, Evaluation, Group” link.

2. The website returns the corresponding page.

3. The Admin fills in the entries.

4. The Admin can choose which fields to be entered or updated in which

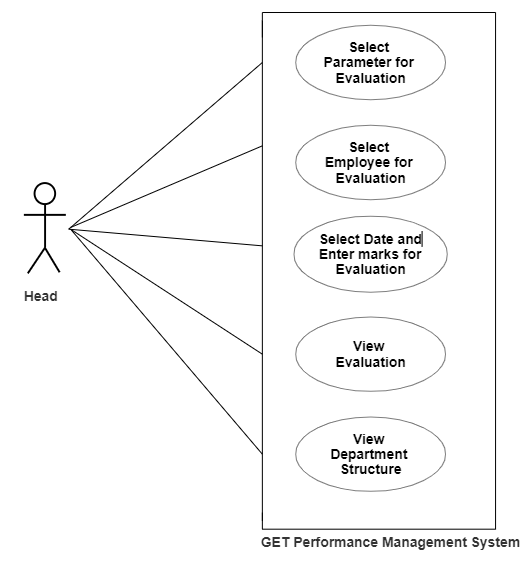
Parameter, Employee, Department, Evaluation, Group.

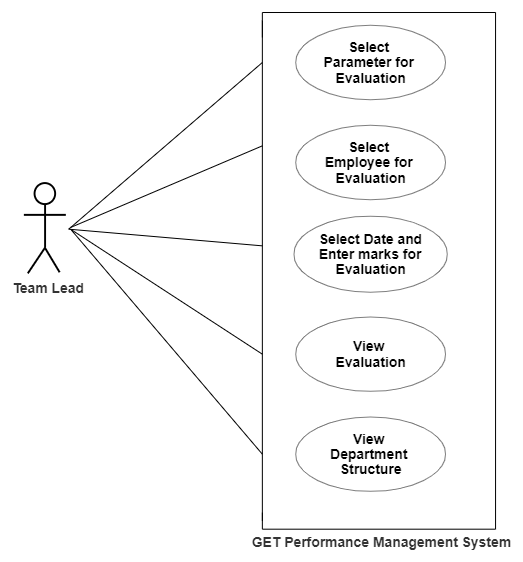
5. The Admin clicks submit.

6. The system adds the data to the Database.

7. The system returns the Admin to the Admin Home Page.

**2.2.2d. Use Case: Admin, Head & Team lead selects parameter, employee, date and enter marks for evaluation**





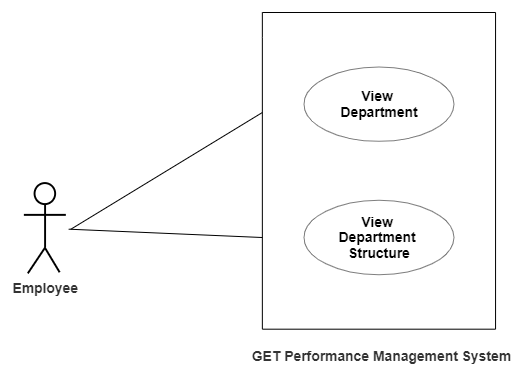


Fig. 2.4 **Admin, Head & Team lead selects parameter, employee, date and enter marks for evaluation**

Brief Description:

Admin, Head & Team lead selects parameter, employee, date and enter marks for evaluation

Initial step-by-step description:

For this use case to be initiated, the admin, head, team lead must be logged in and on “Evaluation” page.

1. The admin, head, team lead selects the “Evaluation” link.

2. If the admin, head or team lead is not logged in, they are directed to the login page.

Once admin, head or team lead gets logged in, they can evaluate the employee performance.

3. The admin, head or team lead selects parameter, employee, date and enter marks.

4. The admin, head or team lead click on “create” to create evaluation.

5. The admin, head or team lead can edit the marks.

6. The list of evaluated employees is displayed on the evaluation home page.

###### ***Non-functional requirements***

There are requirements that are not functional in nature. Specifically, these are the constraints the system must work within.

The web site must be compatible with Microsoft Edge/Google Chrome web browser.

##### **Requirement specifications**

###### ***External interface specifications***

None

* + 1. ***Functional Requirements***

|  |  |
| --- | --- |
| **Use Case Name:** | Access Home Page |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | User is on the home page. |
| **Basic Path** | 1. The user connects to the system using a  web browser.  2. The user selects the link on the  website home page.  3. The system passes the user to the  website Home Page. |
| **Alternate Path** | N/A |
| **Postcondition** | The User is on the Home Page |
| **Exception Path** | If there is a connection failure the website  returns to the wait state |
| **other** |  |

|  |  |
| --- | --- |
| **Use Case Name:** | User Login or Signup |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | User is on the home page. |
| **Basic Path** | 1. The User selects the “Login” link.  2. The User enters email, password and clicks enter.  3. The System retains information in the database.  4. The System returns the user to the website Home Page. |
| **Alternate Path** | N/A |
| **Postcondition** | The User is on the Home Page |
| **Exception Path** | If there is a connection failure the website  returns to the wait state |
| **other** |  |

|  |  |
| --- | --- |
| **Use Case Name:** | Create a New Parameter, Employee, Department, Evaluation, Group |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | The Admin must be logged in and on any of the pages.  (Parameter, Employee, Department, Evaluation, Group) |
| **Basic Path** | 1. The Admin selects the “Parameter, Employee, Department, Evaluation, Group” link.  2. The website returns the corresponding page.  3. The Admin fills in the entries.  4. The Admin can choose which fields to be entered or updated in which Parameter, Employee, Department, Evaluation, Group.  5. The Admin clicks submit.  6. The system adds the data to the  Database.  7. The system returns the Admin to the  Admin Home Page. |
| **Alternate Path** | N/A |
| **Postcondition** | A record is created or updated in the  related Table of the Database. |
| **Exception Path** | 1. If the connection is terminated before the  form is submitted, the fields are cleared  and the website is returned to the wait  state.  2. If the connection is terminated after the  form is submitted, but before the Admin  is returned to the Admin Home Page, the record is created in the Table of the  Database. |
| **other** |  |

|  |  |
| --- | --- |
| **Use Case Name:** | The admin, head or team lead must be logged in and on “Evaluation” page. |
| **Priority** | Essential |
| **Trigger** | Menu selection |
| **Precondition** | User is on the home page. |
| **Basic Path** | 1. 1 The admin, head or team lead selects the “Evaluation” link. 2. If the admin, head or team lead is not logged in, they are directed to login page. Once admin, head or team lead gets logged in, they can evaluate the employee performance. 3. The admin, head or team lead selects parameter, employee, date and enters marks. 4. The admin, head or team lead click on “create” to create evaluation. 5. The admin, head or team lead can edit the marks. 6. The list of evaluated employees is displayed on the evaluation home page. |
| **Alternate Path** | N/A |
| **Postcondition** | The marks allotted to the employee are stored in the database. |
| **Exception Path** | If there is a connection failure the website  returns to the wait state |
| **other** |  |

##### **Hardware Specification**

##### **Client Side:**

* Web Browser: Microsoft Edge/Google Chrome
* Processor: Intel Core i3 & above
* RAM: 4 GB
* Hard Disk: 80 GB

##### **Server Side:**

* Processor: Intel Core i3 & above
* RAM: 2 GB
* Disk space: 4 GB

##### **2.5. Software Specification Client Side:**

* .NET Core 3.1(LTS)
* Microsoft Edge/Google Chrome
* Windows 8, 10

**Web Server:**

* IIS/Kestrel
* Windows 8, 10

##### **Data Base Server:**

* SQL Server 2019

##### **Hardware and Software Requirements in detail Hardware Requirements:**

* Processor: Intel Core i3 & above
* Web Browser: Microsoft Edge/Google Chrome
* RAM: 4 GB
* Hard Disk: 80GB
* Disk space: 4GB

##### **Software Requirements:**

* Microsoft Visual Studio 2019
  + Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It can be used to develop console and graphical user interface applications along with Windows Forms applications, web sites, web applications, and web services in both native-code together with managed code for all platforms supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight.
  + .NET Core 3.1(LTS)
  + C# 8.0
* Web Browser: Microsoft Edge/Google Chrome
* Windows 8,10

# ***CHAPTER – III***

***SYSTEM ANALYSIS AND DESIGN***

### **System Analysis and Design**

##### **Study & Weaknesses of Current System**

##### **Current System**

The Current system for evaluating employee performance is very time consuming and expensive.

The weakness in the current system is described as follows:

* + - Lengthy and complex
    - Become a hindrance in the employee’s progress
    - Contradictory and misleading opinions in the performance management file
    - Partialities and favoritism
    - Flawed conditions and standards
    - Employees may suffer from low self-esteem
    - Demeaning the communication system between employer and employee
    - Time Consuming

**Weaknesses in Current System**

Preparing for performance evaluation is very time consuming for both managers and employees.

Performance evaluation discussions are the most stressful ones for employees and their supervisors.

Needless to say, employees are under more stress than the supervisors, this leads to a decrease in employee morale and might also result in disengagement of employees at work.

Sometimes performance evaluation can be ineffective and inaccurate due to lack of proper metrics and various biases. Many times, employee performance reviews are done based on the memory of those who are doing the review, there isn’t a proper system from where proper employee metrics can be monitored and kept track of.

Performance evaluation is usually just a one-way communication from a manager to their employee. It becomes a highly complex process due to the paperwork involved.

Thus, the current system is in every way ineffective for conducting evaluation in these days when time is more costly than anything and the duration for completion of company goals gets delayed.

##### **Requirements of New System**

* + 1. **User Requirements**

The User requirements for the new system are to make the system fast, flexible, less prone to errors and reduce expenses and save time.

* + - * The New system should be less time-consuming. Time should be scheduled by setting an effective timeline.
      * The New system should include frequent evaluation and reviews to prevent inaccurate evaluation and biases.
      * The New system should be more secure in managing user records and reliable enough to be used in any condition.
      * Finally, it should prove cost effective as compared to the current system.

##### **Feasibility Study**

A key part of the preliminary investigation reviews anticipated costs and benefits and recommends a course of action based on operational, technical, economic, and time factors. The purpose of the study is to determine if the system's request should proceed further.

##### **Does the New System Contribute to the Overall Objectives of the Organization?**

The new system would contribute to the overall objectives of the organization. It would provide a quick, error-free and cost-effective solution to the current process. It would provide a solution to many issues in the current system. As the new system is flexible and scalable it can also be upgraded and extended to meet other complex requirements which may be raised in the future. However, it is up to the organization to upgrade or extend it.

##### **Can the New System be Implemented Using Current Technology?**

The organization has a computer laboratory which has about 50 machines connected by Internet LAN and managed by a server. It would be very easy to set up the system in the current environment as the application is web based and it does not require it to be installed on every machine. The database and IIS are set up on the server and the .NET Core installed on every machine; the system can be started as quickly as required by the management.

##### **Features of the New System.**

The new system has been designed as per the user requirements so as to fulfil almost all of them.

##### **Time Efficiency**

Preparing for performance reviews is very time-consuming for both managers and employees. This can be prevented by using a performance management system that encourages frequent performance reviews than the traditional annual performance reviews. Also, the timeline for evaluation should be specified.

* **Proper evaluation mechanism**

Sometimes performance evaluation can be ineffective and inaccurate due to lack of proper metrics and various biases. Many times, employee performance reviews are done based on the memory of those who are doing the evaluation, there isn’t a proper system from where proper employee metrics can be monitored and kept track of. Using flexible performance evaluation software could help the situation.

##### **Easy to Store and Retrieve Information**

The new system makes it easy to store and retrieve information as required and does not involve storing information on separate sheets or papers. It thus solves data management problems faced in the current system as it has an effective Database Management System.

##### **Cost Effective**

One of the main reasons for the success of the new system is its cost effectiveness. It saves the amount spent on stationery as well as the overall cost of conducting the evaluation.

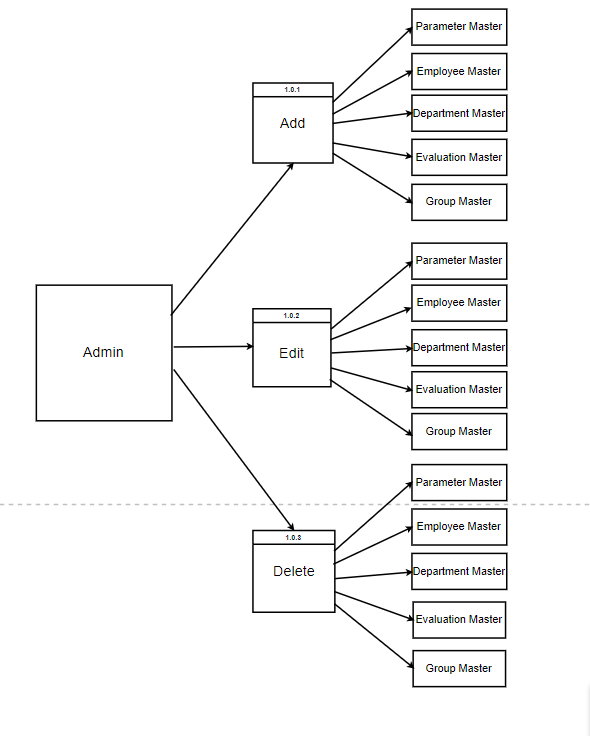
##### **Data Flow Diagram (DFD)**

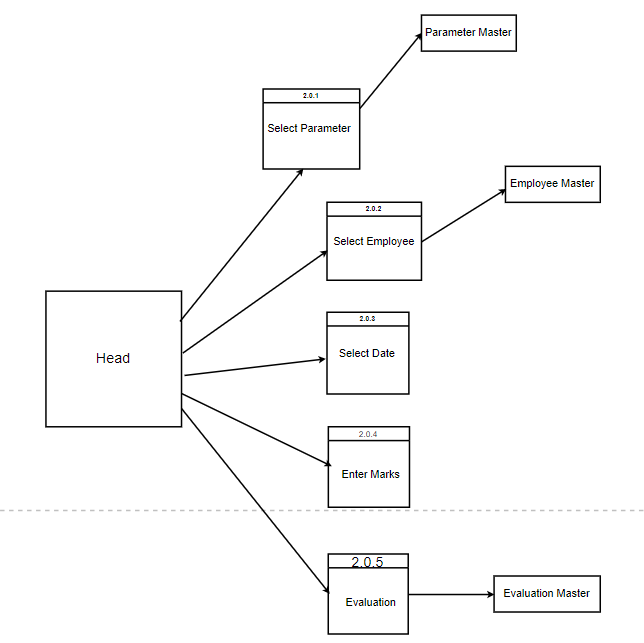
The DFD (also known as *bubble chart*) is a simple graphical formalism that can be used to represent a system in terms of the input data into the system, various processes carried on these data, and the output data generated by the system.

The main reason why the DFD technique is so popular is because the fact that the DFD is a very simple formalism – it is simple to understand and use. A DFD model uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among the functions. Starting with a set of high-level functions that a system performs, a DFD model hierarchy represents various sub-functions.

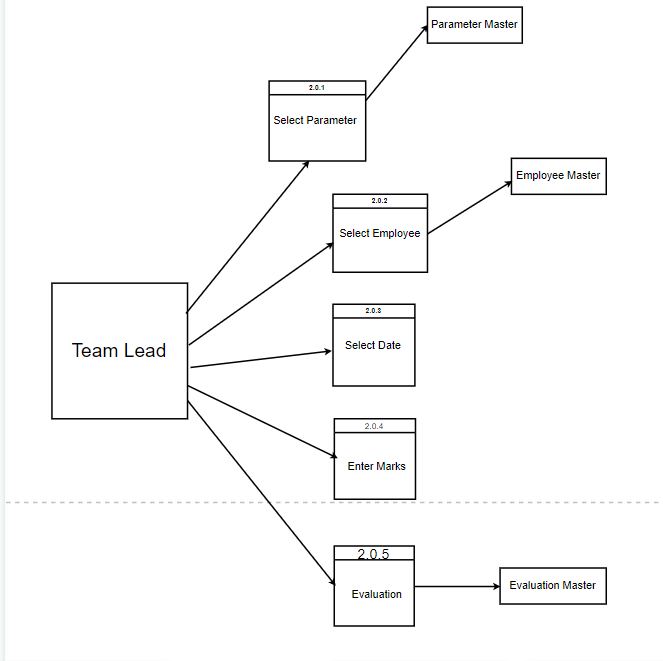
##### Screenshot 2022-06-01 193218.png

**Fig 3.1 DFD (Level 0 )**

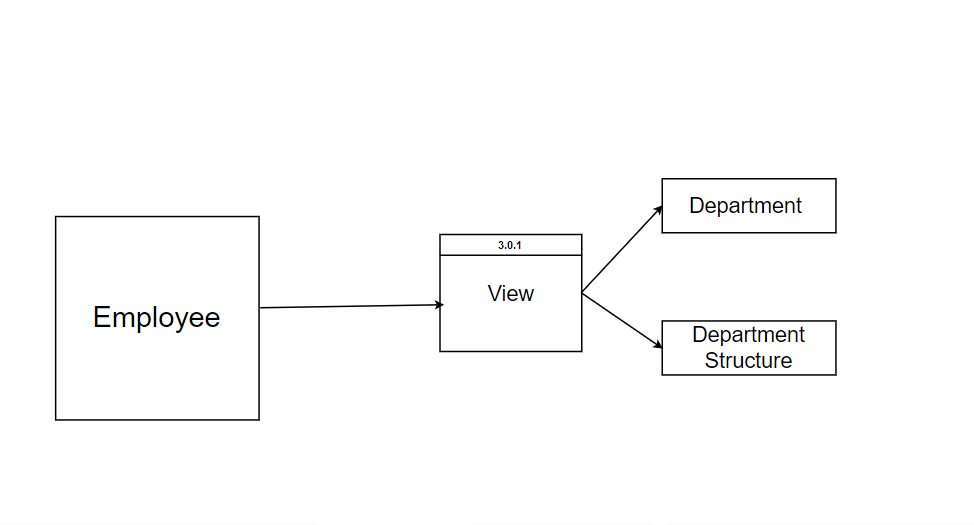
**Fig. 3.2 Admin (DFD Level 1)**



##### **Fig. 3.3 Head (DFD Level 1)**



##### **Fig. 3.5 Team Lead (DFD Level 1)**

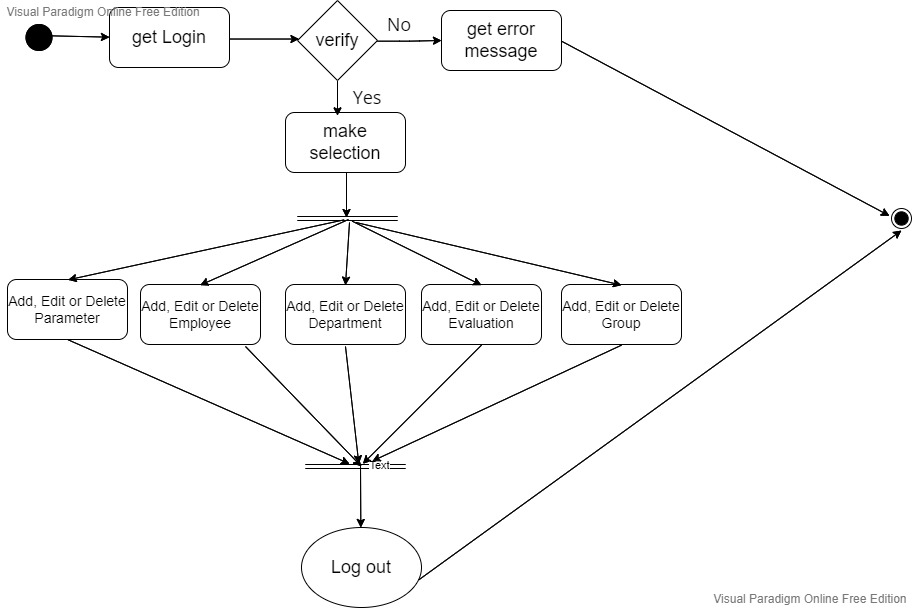


##### **Fig. 3.5 Employee (DFD Level 1)**

**UML Modeling**

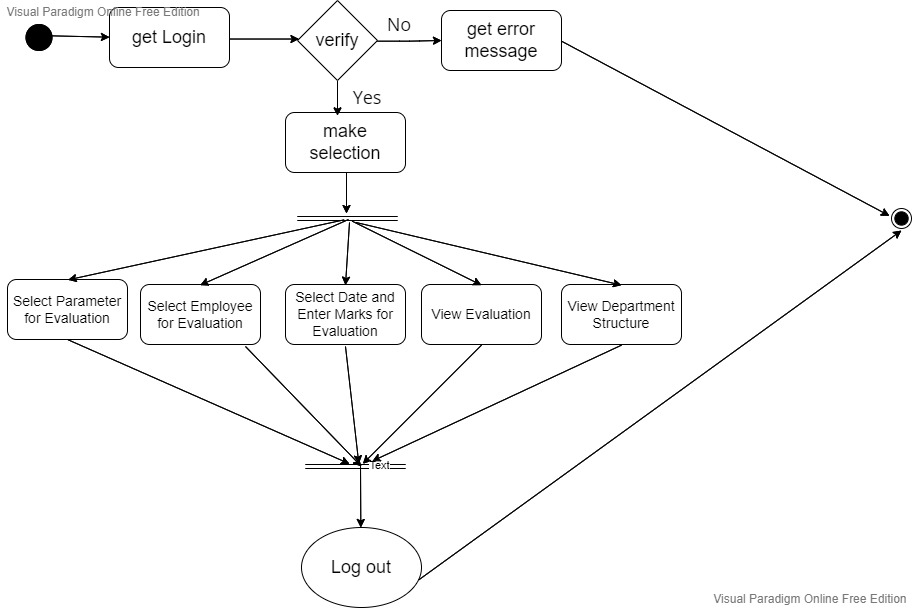
##### **3.6.1 Activity Diagram**

**3.6.1a. Activity Diagram for Admin**



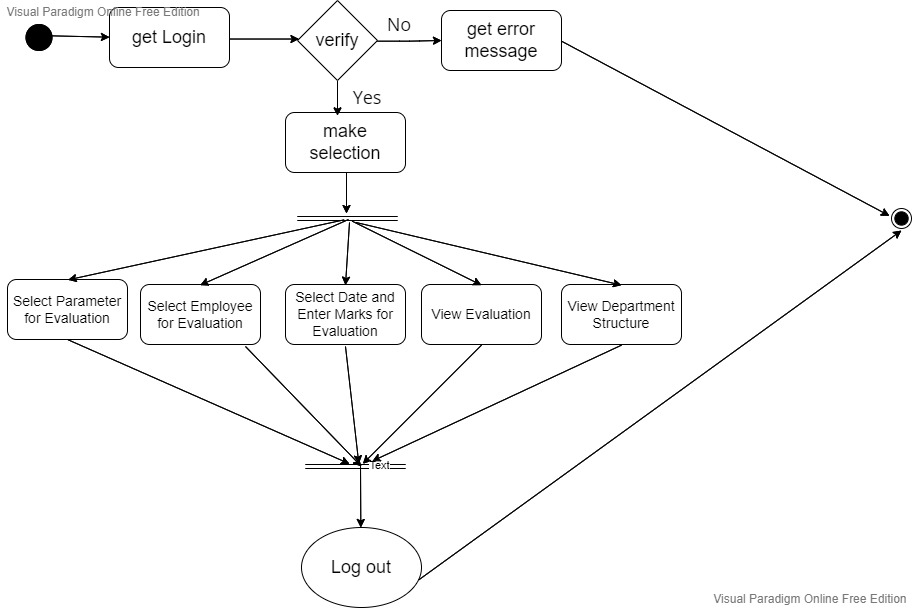
**Fig. 3.6 Activity Diagram for Admin**

##### **3.6.1b. Activity Diagram for Head**



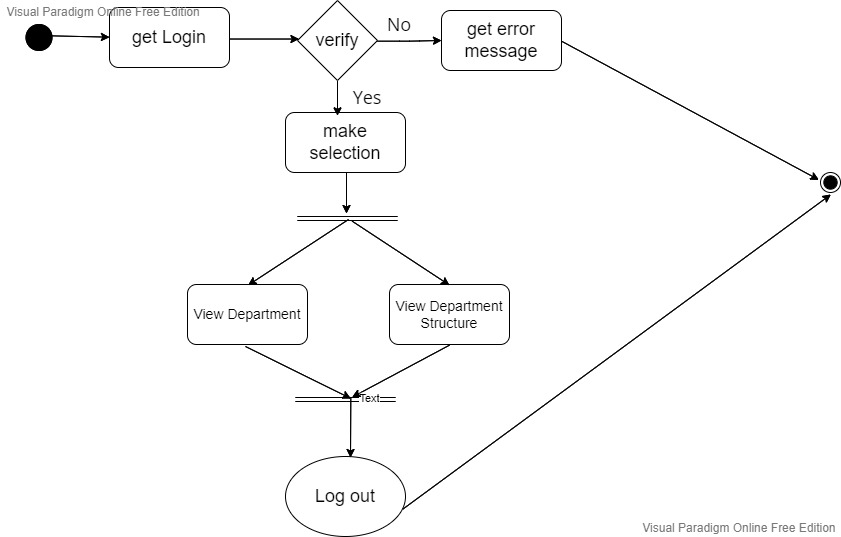
##### **Fig. 3.7 Activity Diagram for Head**

##### **3.6.1c. Activity Diagram for Team Lead**



**Fig. 3.8 Activity Diagram for Team Lead**

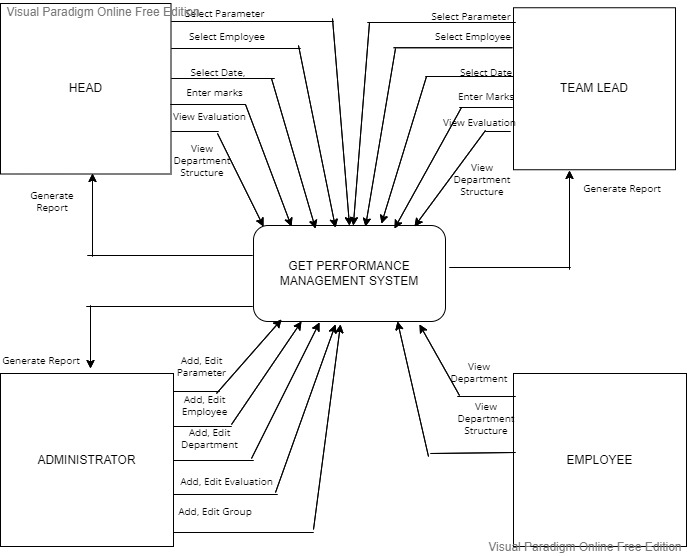
##### **3.6.1d. Activity Diagram for Employee**



##### **Fig. 3.9 Activity Diagram for Employee**

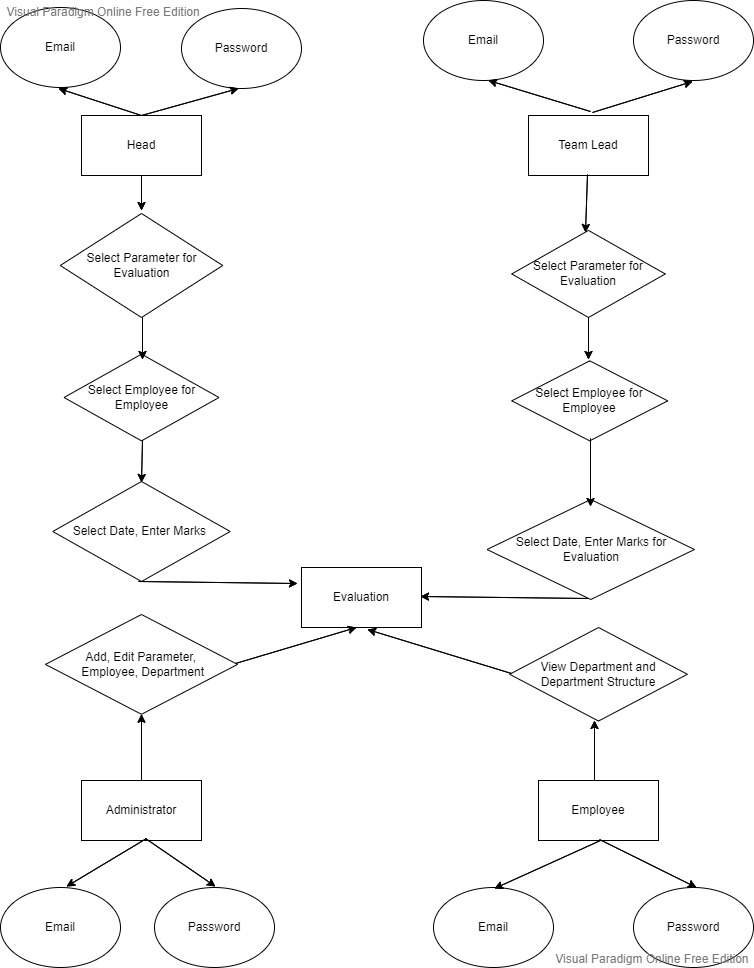
**Context Diagram**

The context diagram is a top-level view of an information system that shows the boundaries and scope. It describes the main objective of the system and the entities involved.



**Fig. 3.9 Context Diagram**

##### **E-R Diagram**



**Fig. 3.9 E-R Diagram**

# ***CHAPTER – IV***

# ***DATA DICTIONARY***

### **Data Dictionary**

A data dictionary is a catalog-a-repository of the elements in a system. As the name suggests, their elements center on data and the way they are structured to meet user requirements and organizational needs. In a data dictionary you will find a list of all the elements composing the data flowing through a system. The major elements are data flows, data stores and processes. The data dictionary stores details and descriptions of these elements.

If analysis want to know characters are in a data item by what other names it is referenced in the system, or where it is referenced in the system, or where it is issued in the system, they should be able to find the answers in issued in the system, they should be able to find the answer in properly developed data dictionary.

The Dictionary contains two types of description for the data following through the system.

##### **Data Elements**

The most fundamental data is the elements. They are building blocks for all other data in the system. Data elements are also alternatively known as fields, data item or elementary item.

##### **Data Structure**

A data structure is a set of items that are related to one another and describe the component in the system.

### **Table Details**

##### **Table 4.1. User**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| Id | Primary Key | Not null | 4 | int |
| Name | Name of user |  | MAX | nvarchar |
| Surname | Surname of user |  | MAX | nvarchar |
| Email | Email Id of user |  | MAX | nvarchar |
| Password | Password of user |  | MAX | nvarchar |
| RoleId | Foreign Key |  | 4 | int |
| StateId | Foreign Key |  | 4 | int |
| SupervisorId | Foreign Key |  | 4 | int |
| DepartmentId | Foreign Key |  | 4 | int |

**Table 4.2. Parameters**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| Id | Primary Key | Not Null | 4 | int |
| Name |  |  | MAX | nvarchar |
| Coefficient | Coefficient | Not Null | 53 | float |
| Mark\_1\_Description | Description of Mark 1 |  | MAX | nvarchar |
| Mark\_2\_Description | Description of Mark 2 |  | MAX | nvarchar |
| Mark\_3\_Description | Description of Mark 3 |  | MAX | nvarchar |
| Mark\_4\_Description | Description of Mark 4 |  | MAX | nvarchar |
| Mark\_5\_Description | Description of Mark 5 |  | MAX | nvarchar |

##### **Table 4.3. Evaluations**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| Id | Primary Key | Not Null | 4 | int |
| Date | Date of evaluation | Not Null | 7 | datetime2 |
| AssessorId | Foreign Key |  | MAX | nvarchar |
| UserId | Foreign Key |  | MAX | nvarchar |
| ParameterId | Foreign Key |  | MAX | nvarchar |
| Mark | Marks allotted | Not Null | 53 | float |

**Table 4.4. Departments**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| Id | Primary key | Not Null | 4 | int |
| Name | Name of Department |  | MAX | nvarchar |

**Table 4.5. Groups**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| Id | Primary key | Not Null | 4 | int |
| Name | Name of the  group |  | MAX | nvarchar |
| DepartmentId | Foreign Key | Not Null | 4 | int |

##### **Table 4.6. ParametersGroup**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| ParameterId | Composite Key | Not Null | 4 | int |
| GroupId | Composite Key | Not Null | 4 | int |
| Mark | Marks Allotted | Not Null | 53 | float |

**Table 4.7. DepartmentParameters**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| DepartmentId | Primary Key | Not Null | 4 | int |
| ParameterId | Primary Key | Not Null | 4 | int |
| Mark | Marks Allotted | Not Null | 53 | float |

**Table 4.8**. **Role**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| Id | Primary Key | Not Null | 4 | int |
| Name | Name of Role |  | MAX | nvarchar |

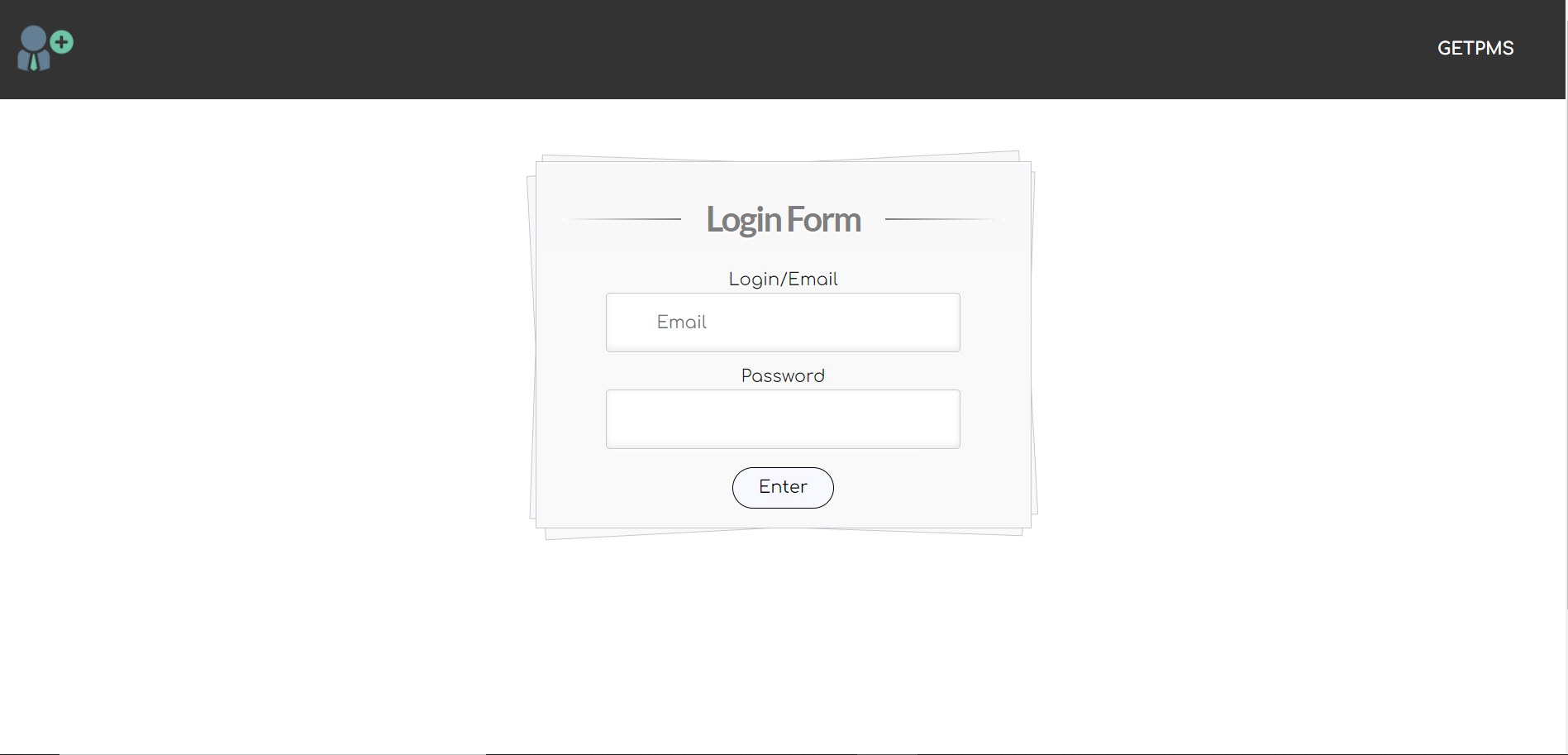
**Table 4.9. Status**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Constraints** | **Size** | **Data Type** |
| Id | Primary Key | Not Null | 4 | int |
| Name | Name of Status |  | MAX | nvarchar |

# ***CHAPTER – V***

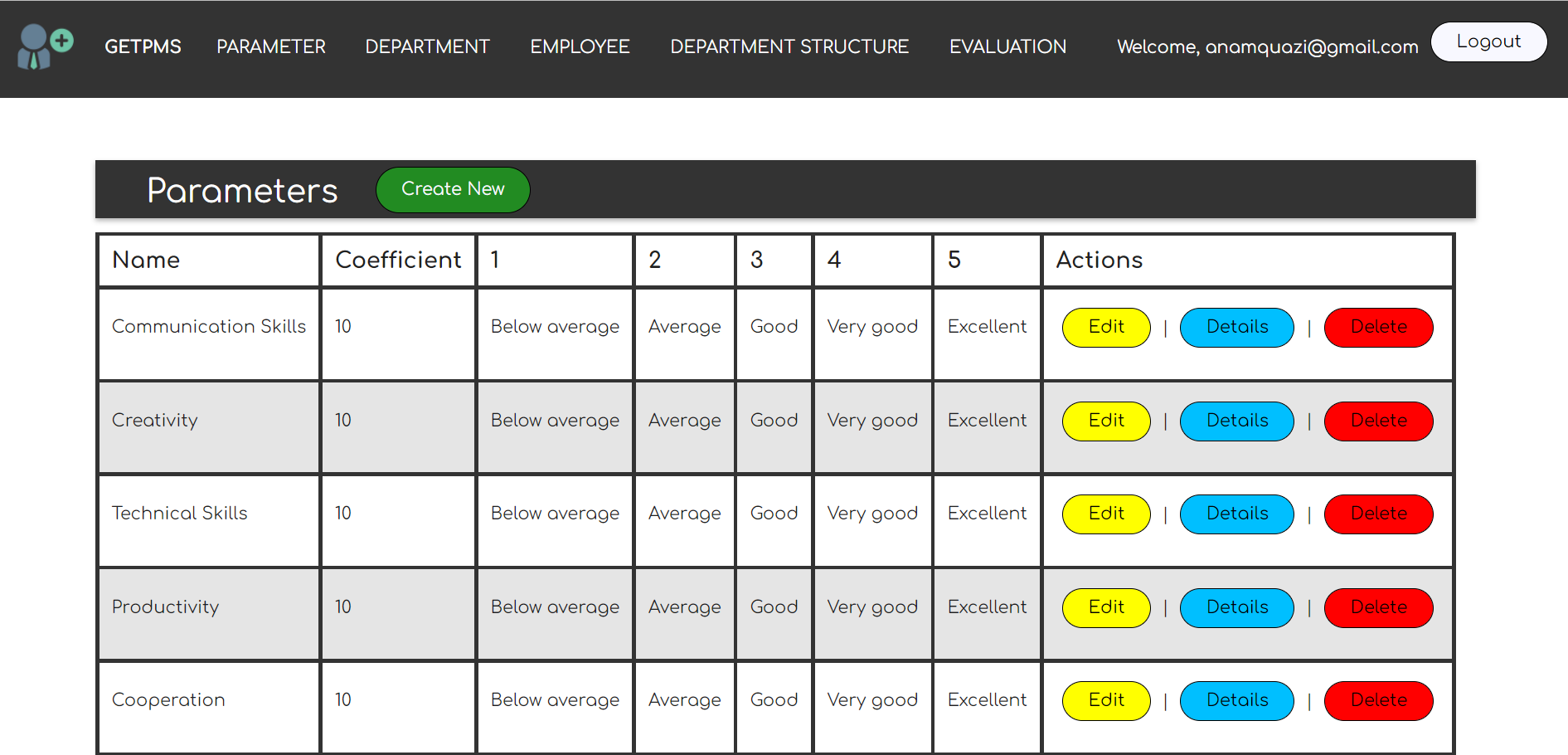
# ***SCREENSHOTS***

### **5.1. Login Form**

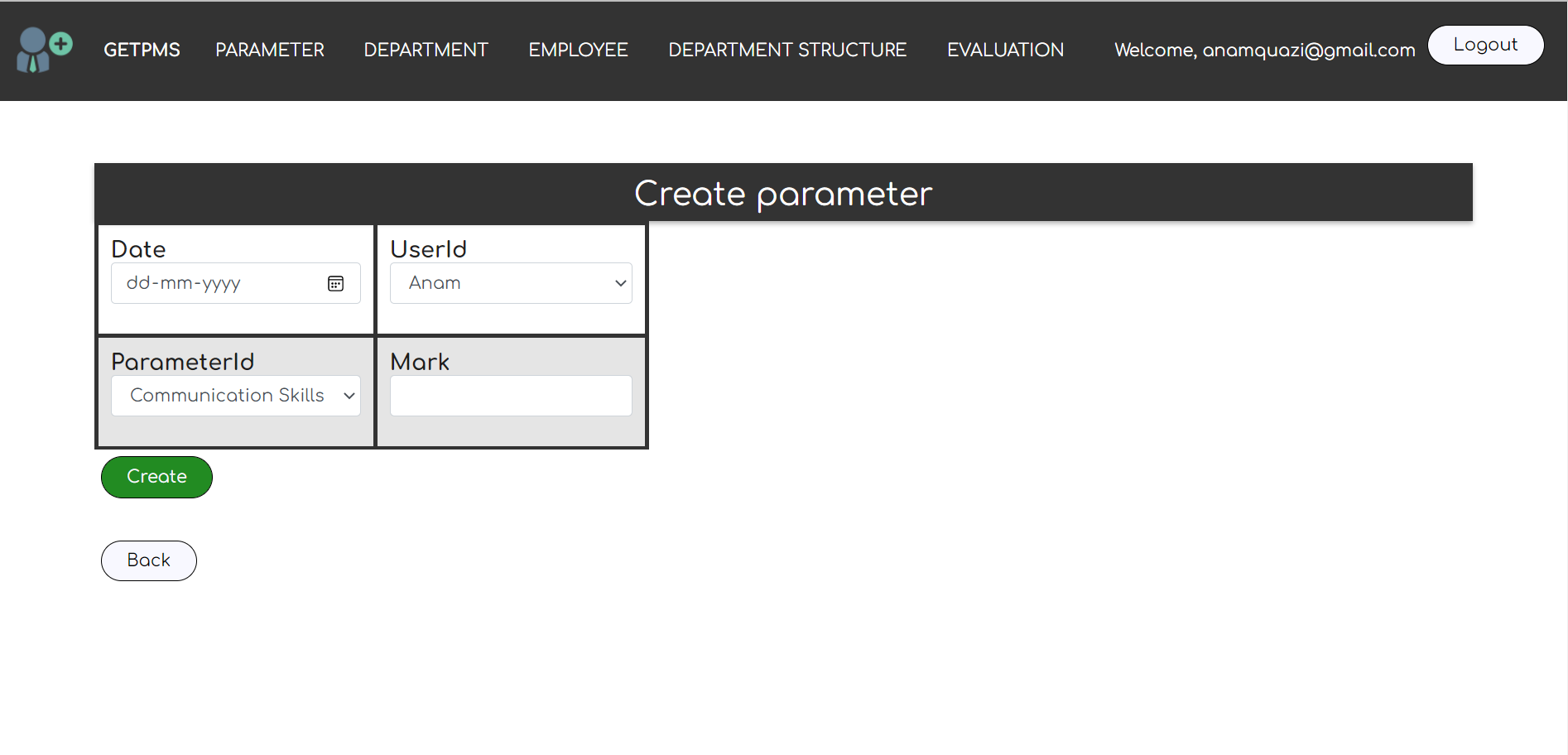


### **5.1.1. Admin Login**

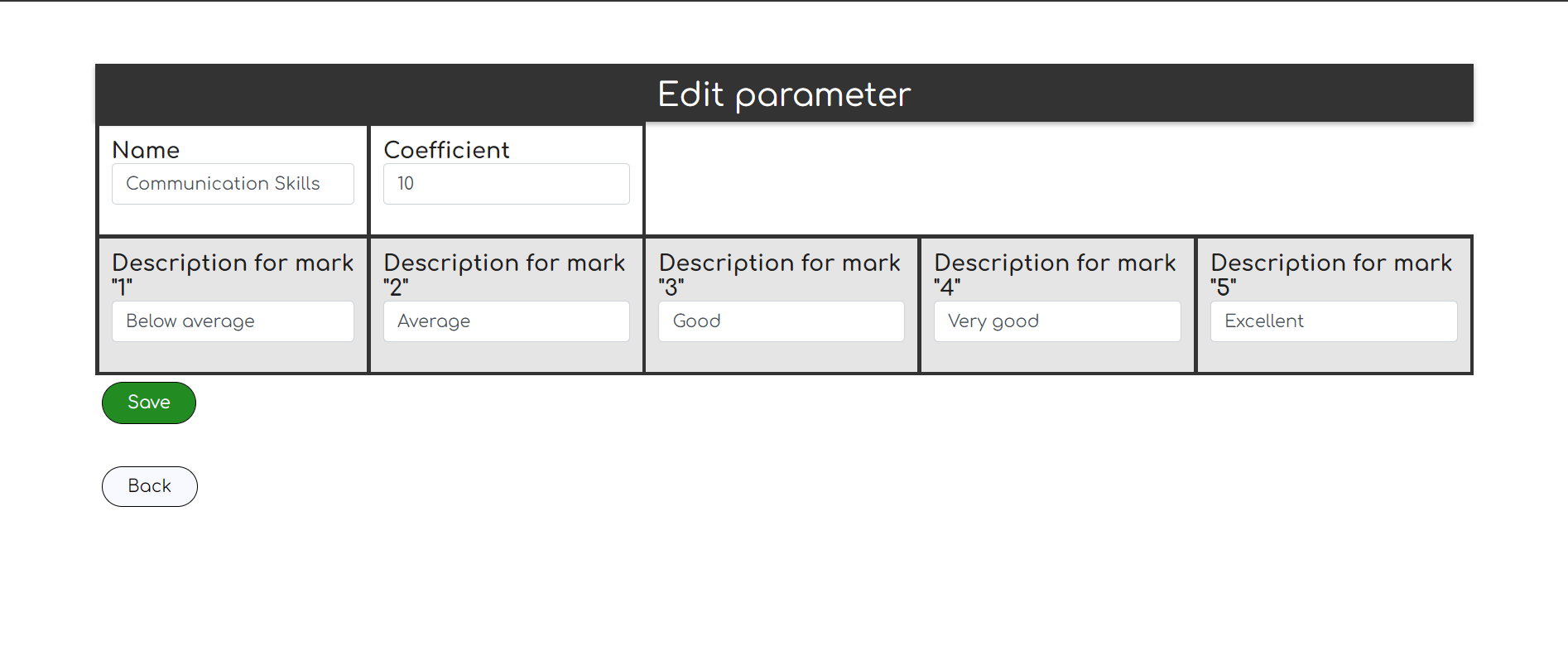
### **5.2. Parameter Home Page**



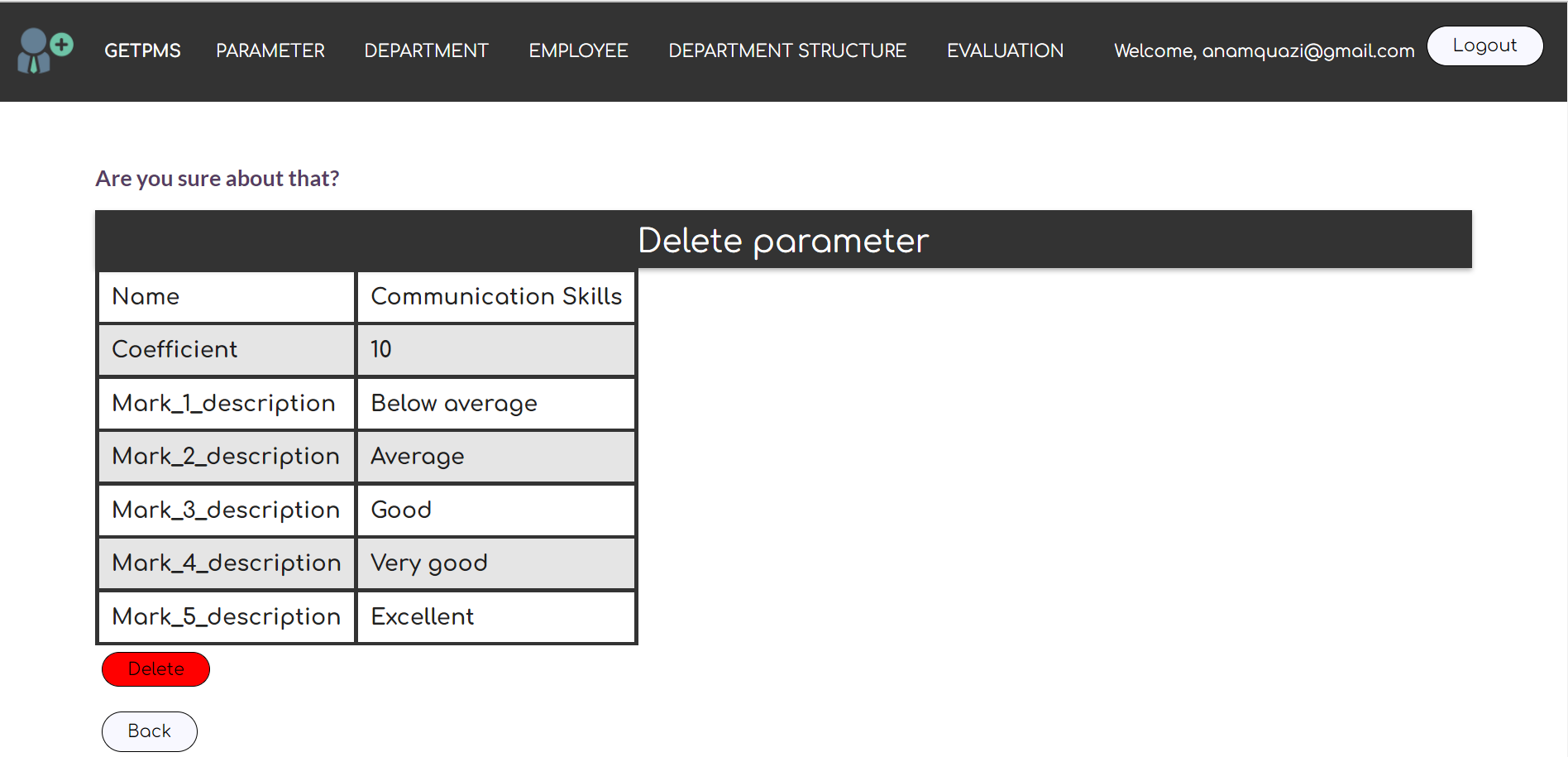
### **5.2.1. Create Parameter**

****

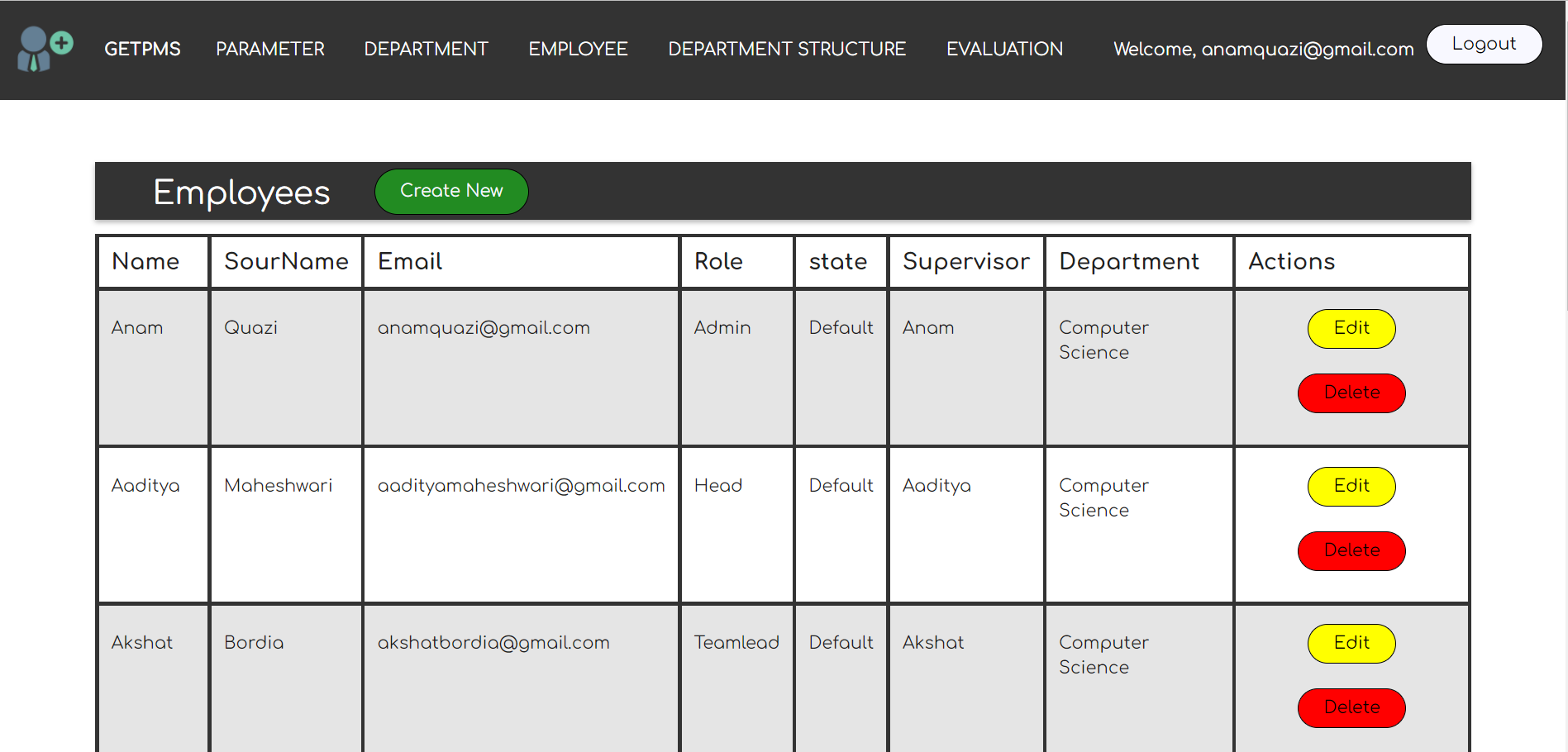
**5.2.2. Edit Parameter**

****

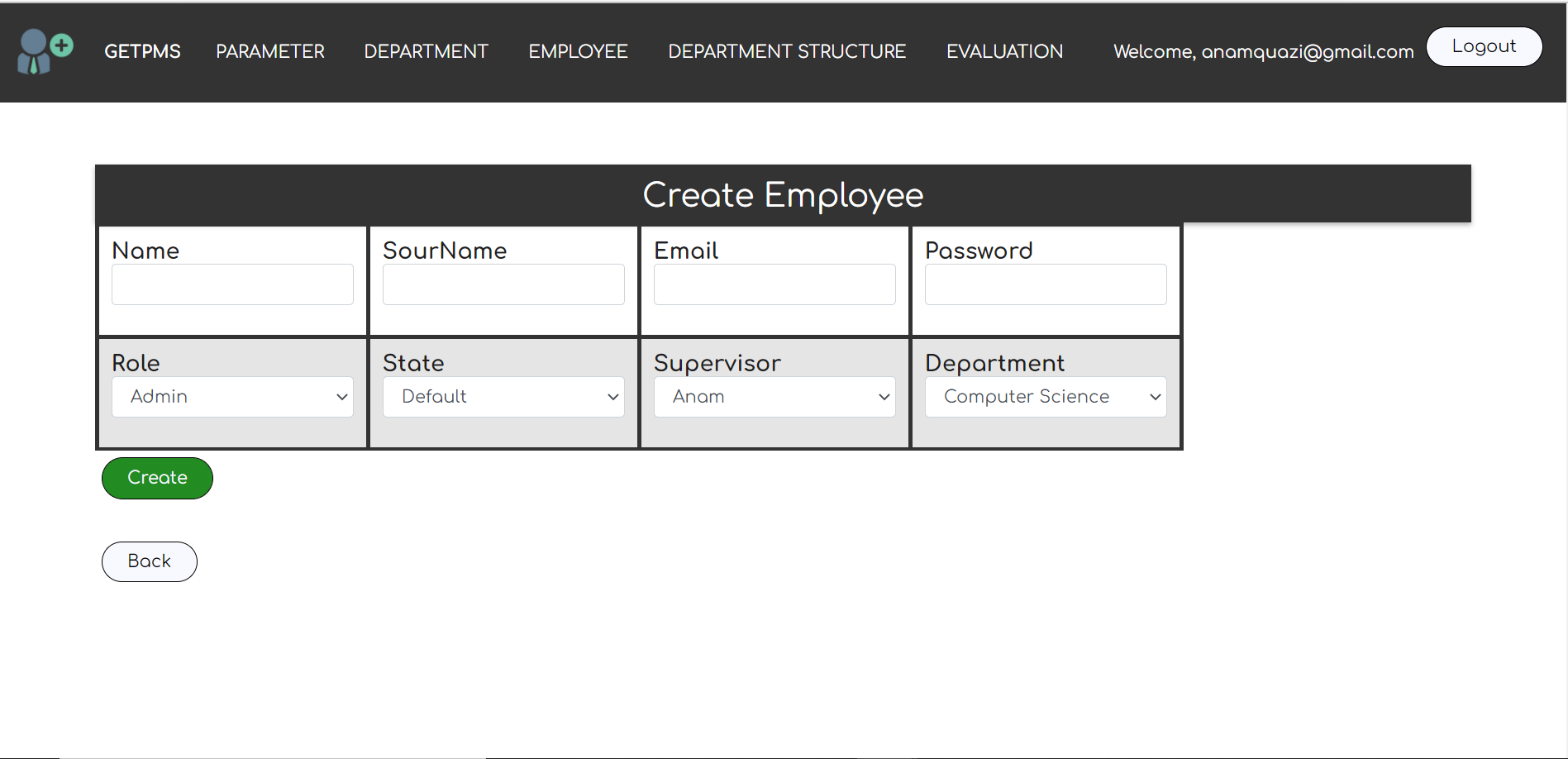
**5.2.3. Delete Parameter**

****

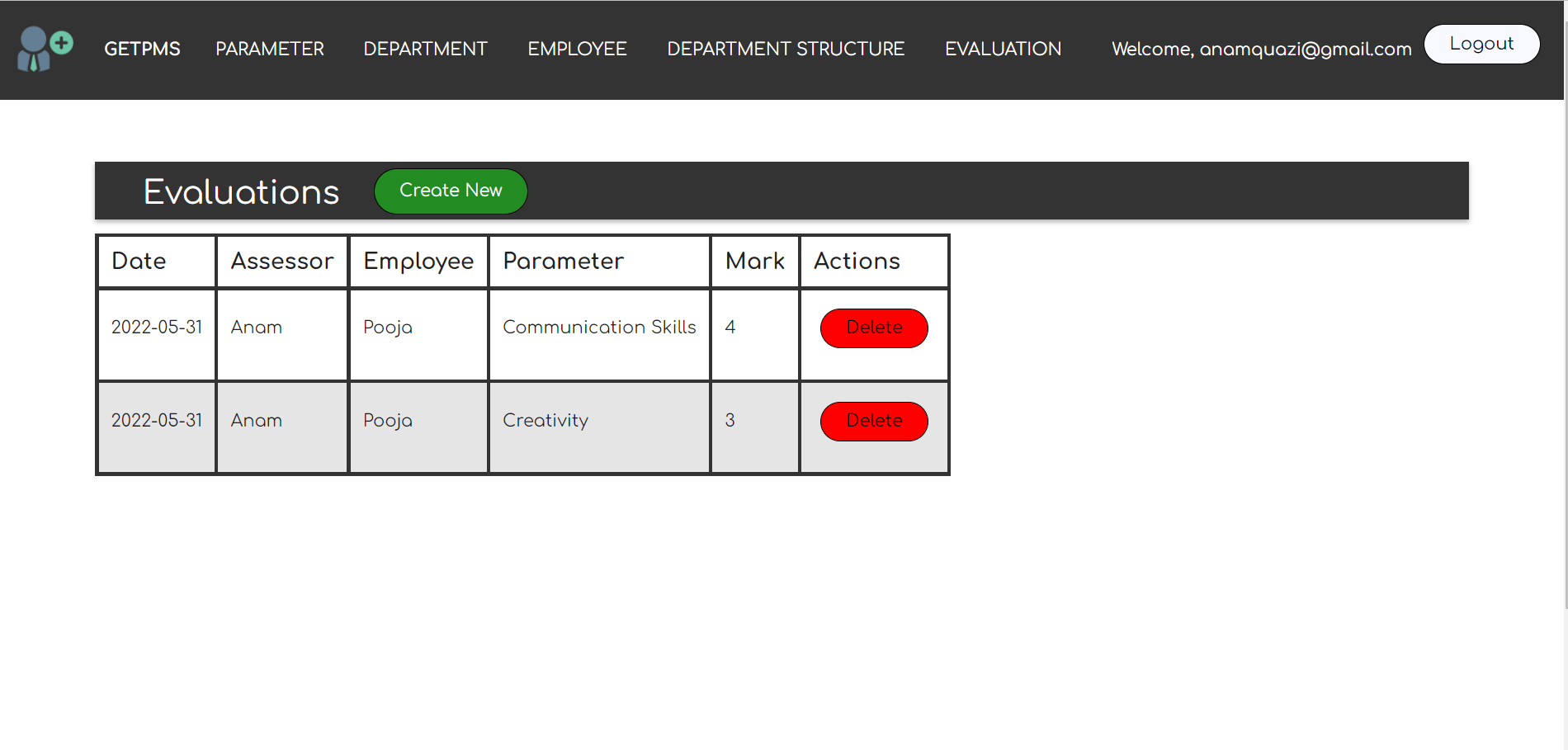
**5.3. Employee Home Page**

****

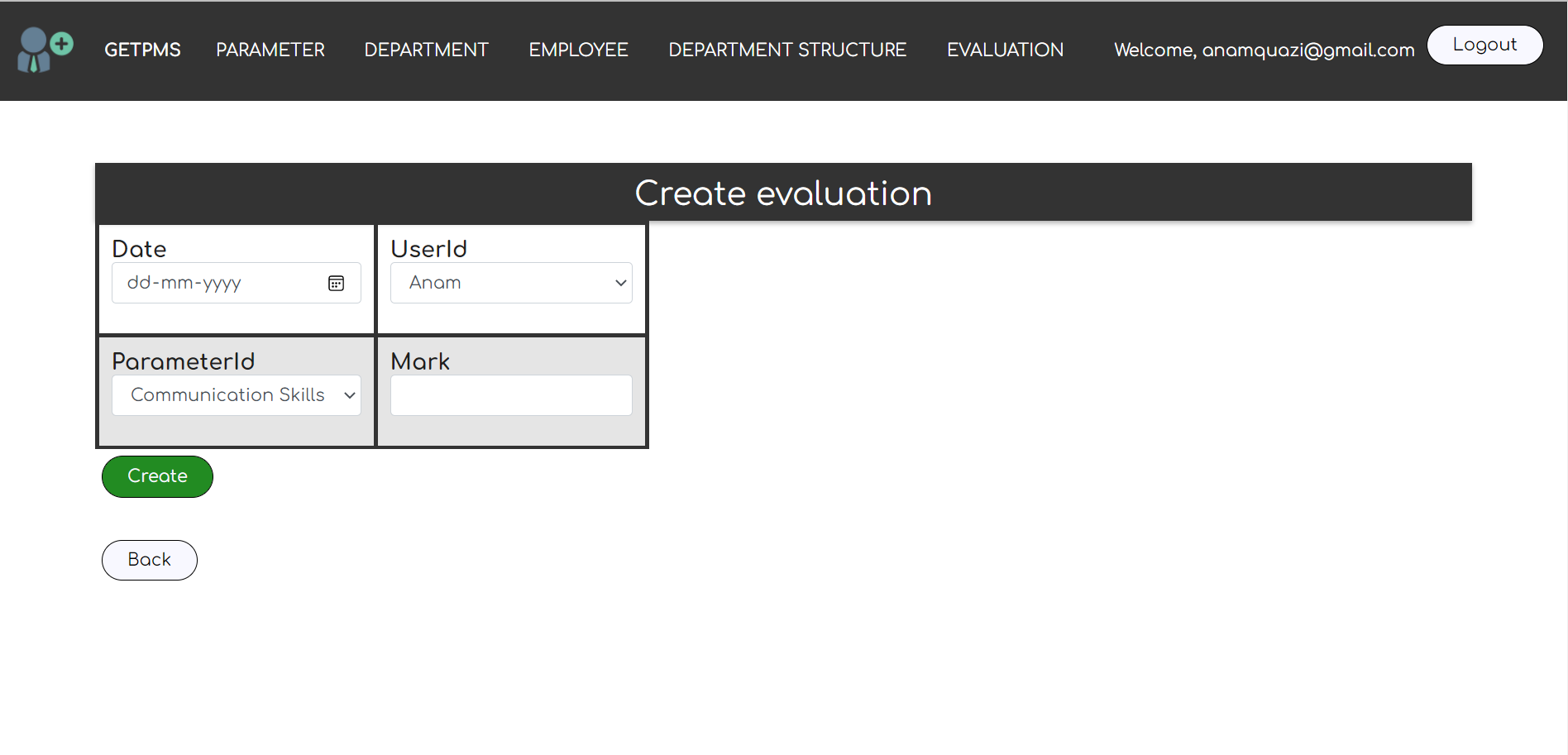
**5.3.1. Create Employee**

****

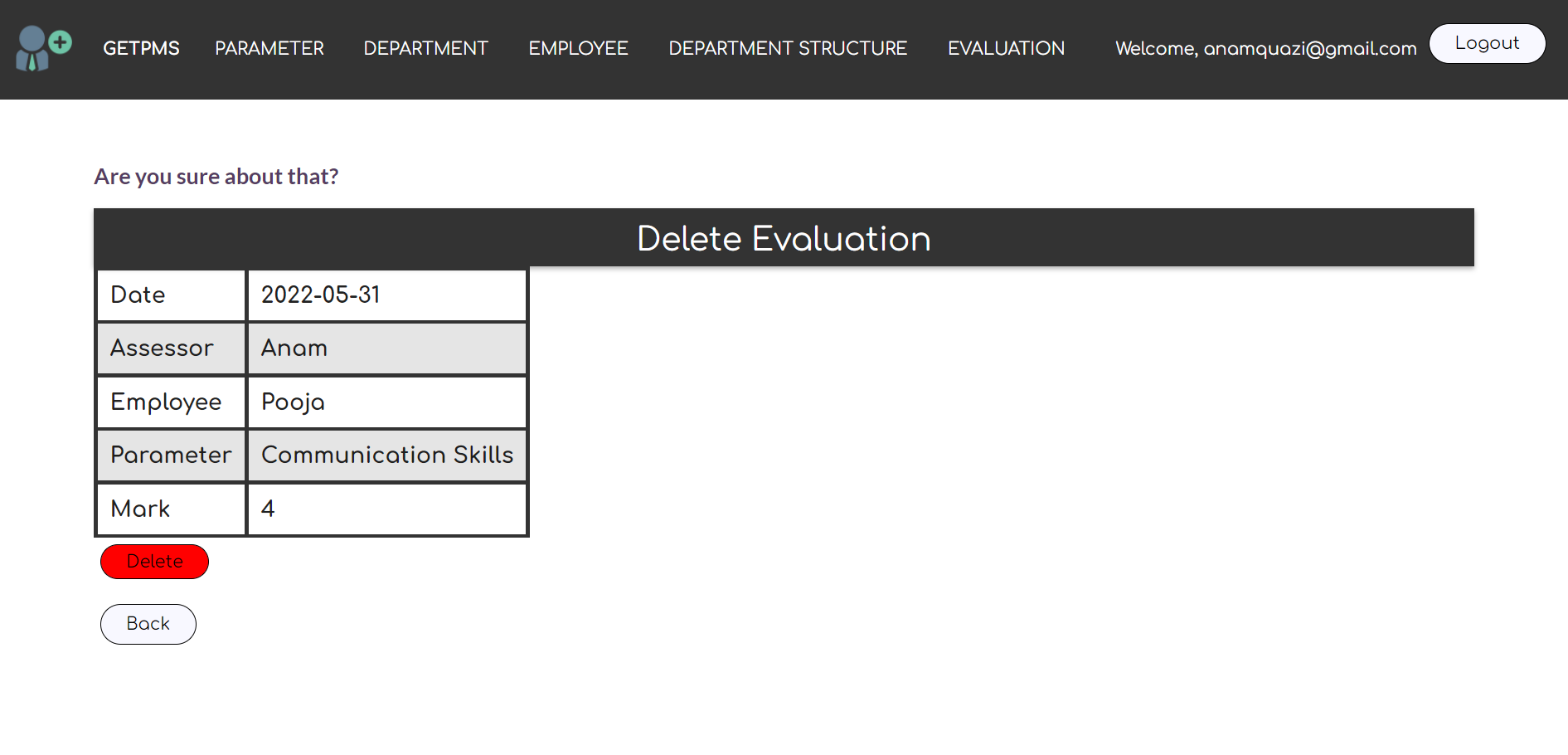
**5.4 Evaluation Home Page**

****

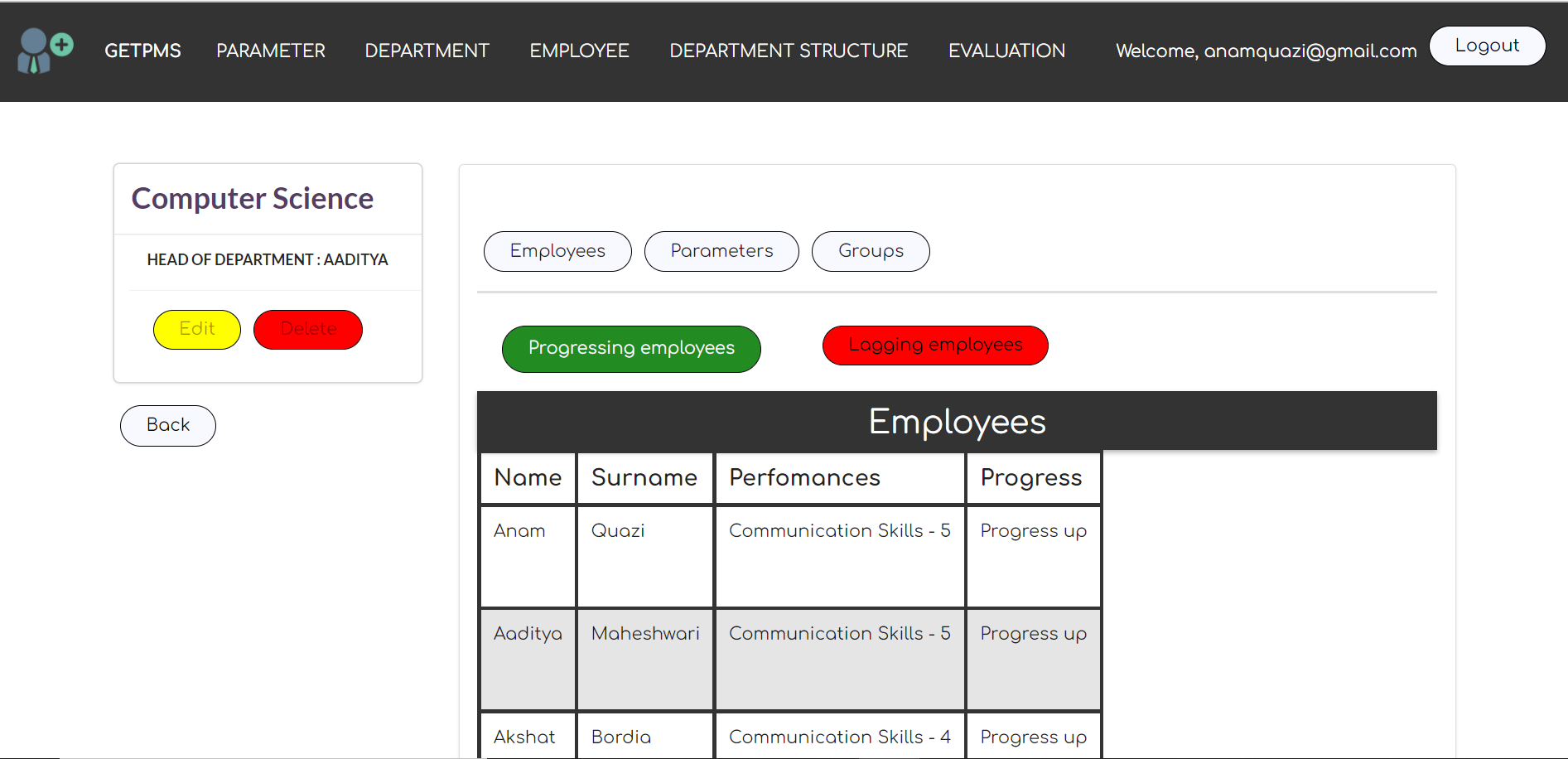
**5.4.1. Create Evaluation**

****

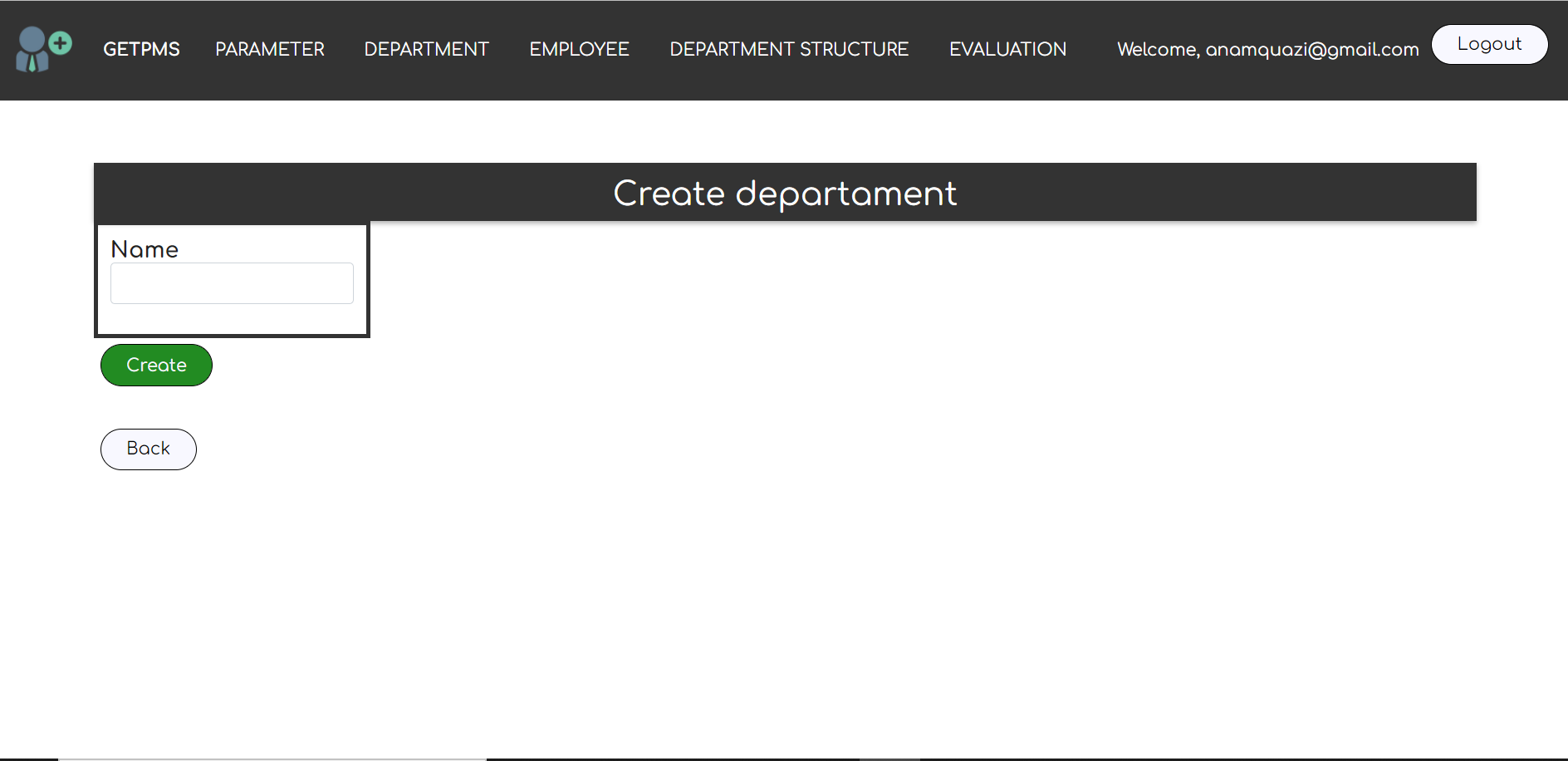
**5.4.1. Delete Evaluation**

****

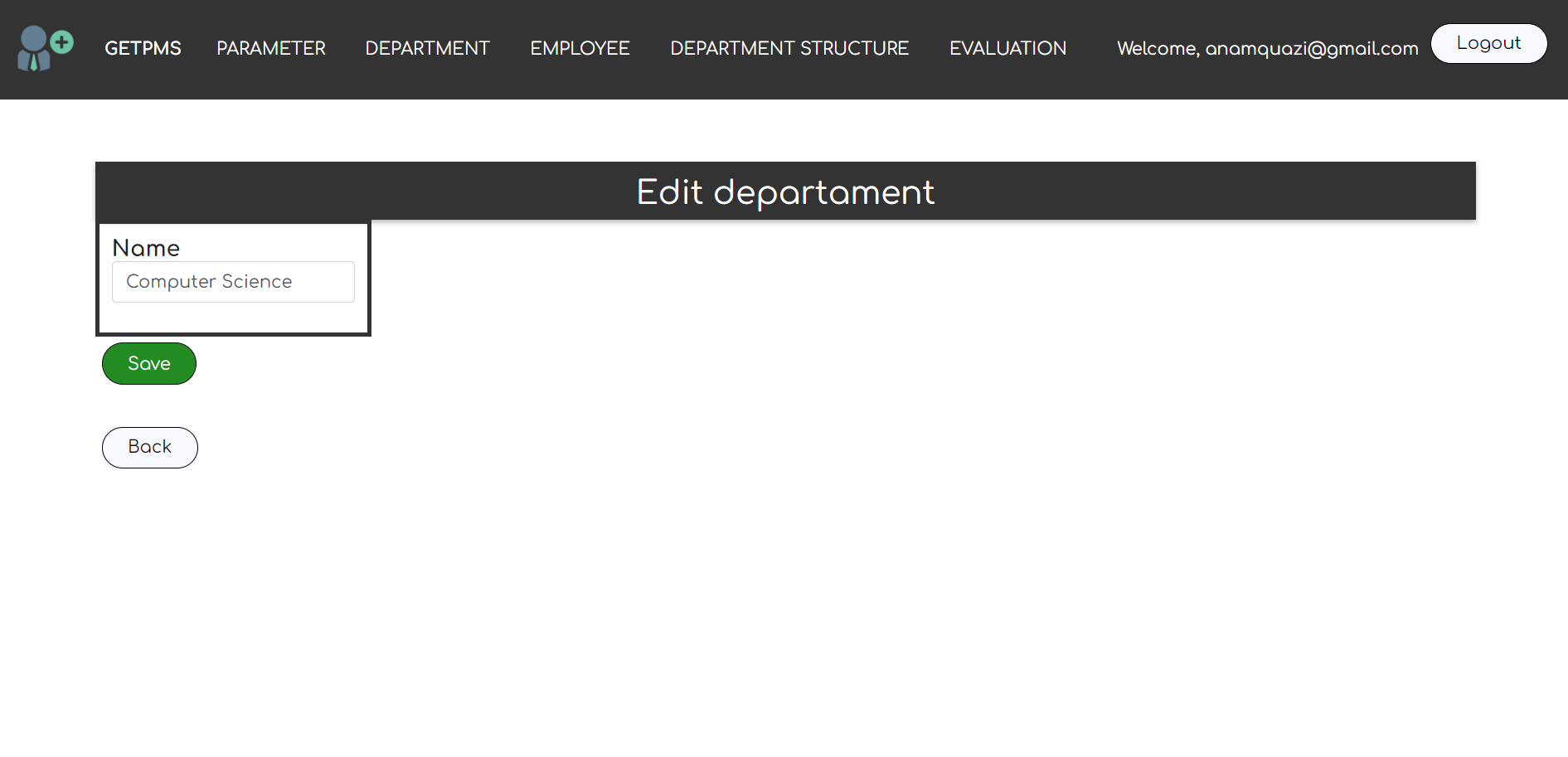
**5.5. Department Home Page**

****

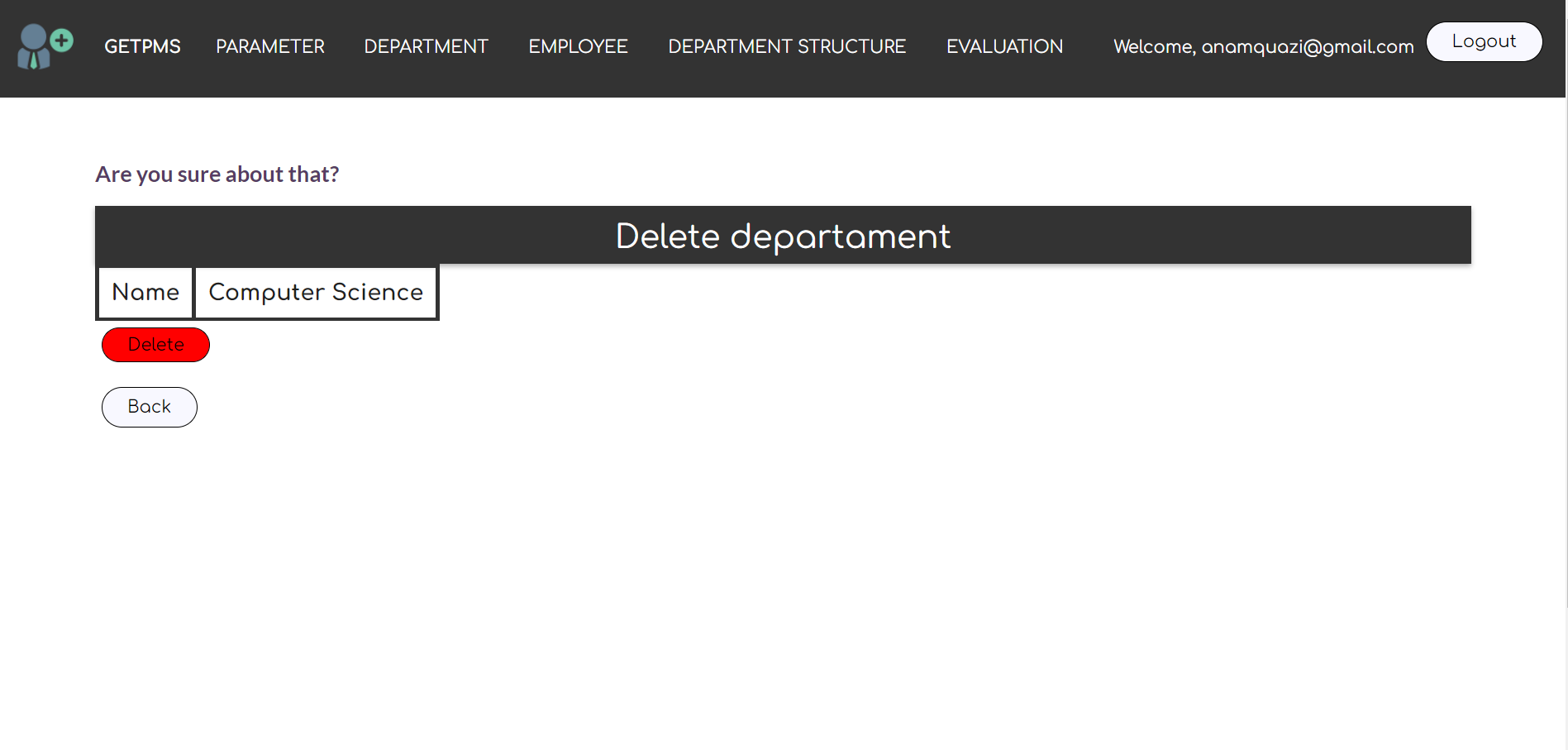
**5.5.1. Create Department**

****

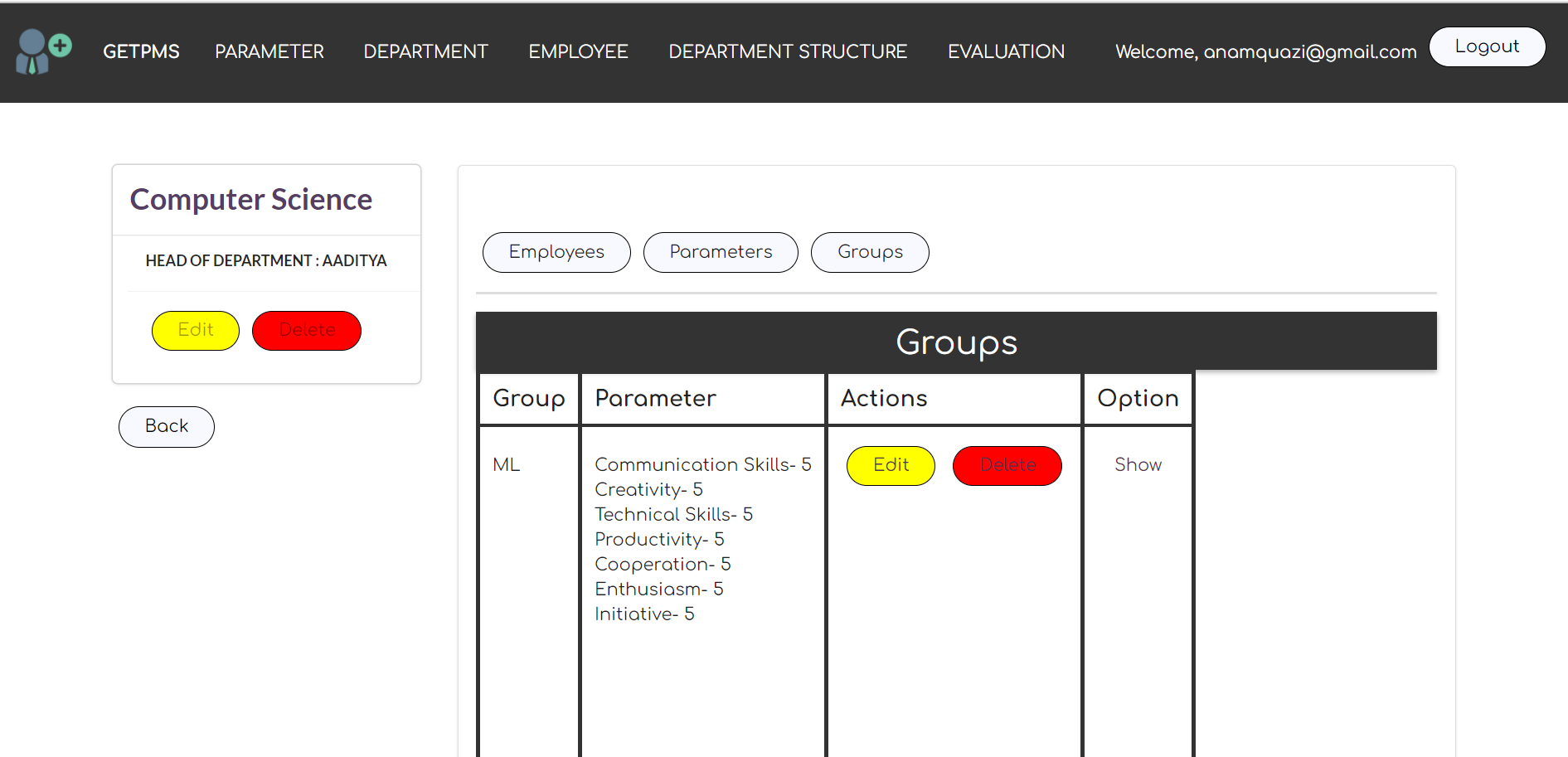
**5.5.2. Edit Department**

****

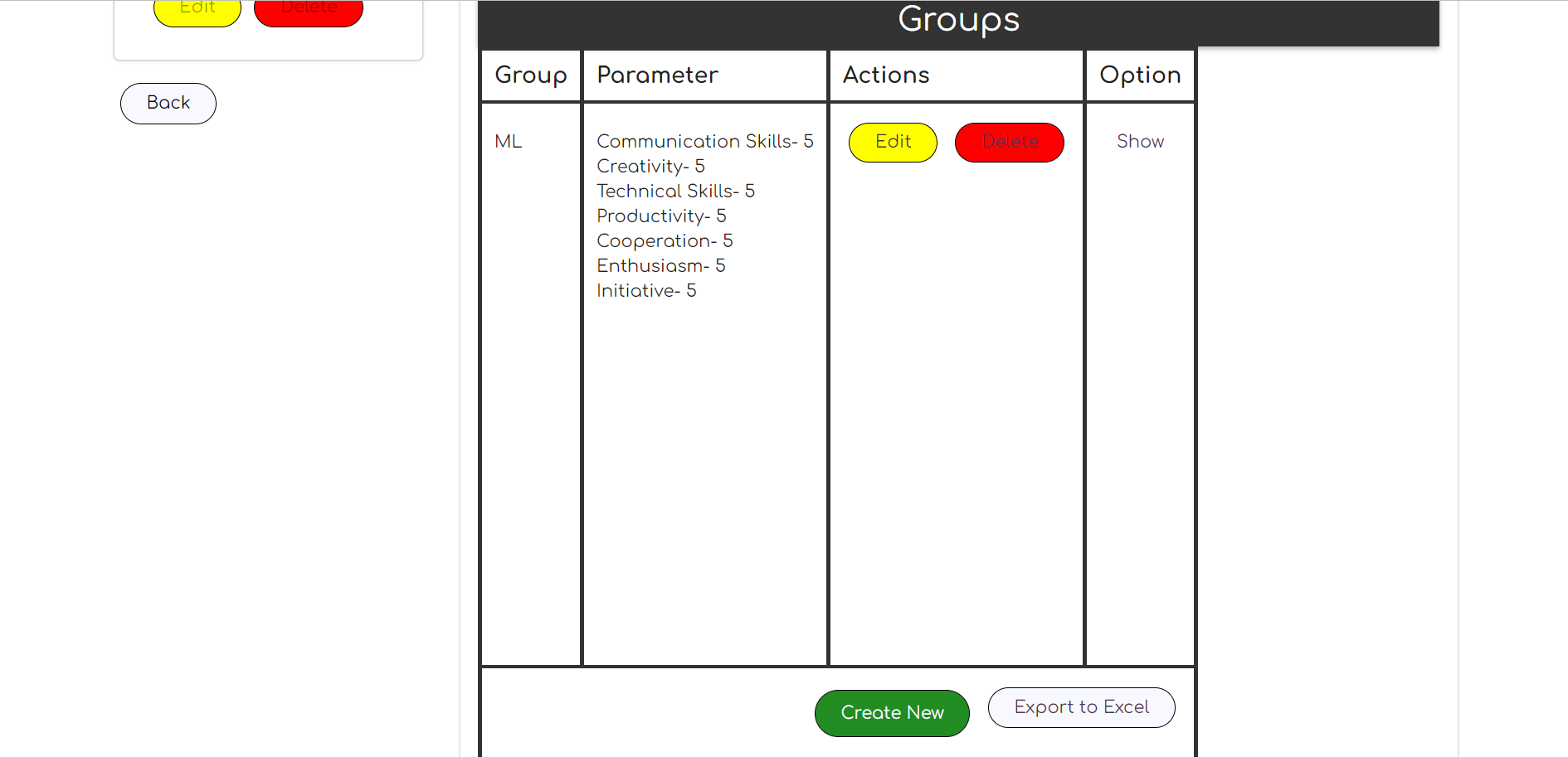
**5.5.3. Delete Department**

****

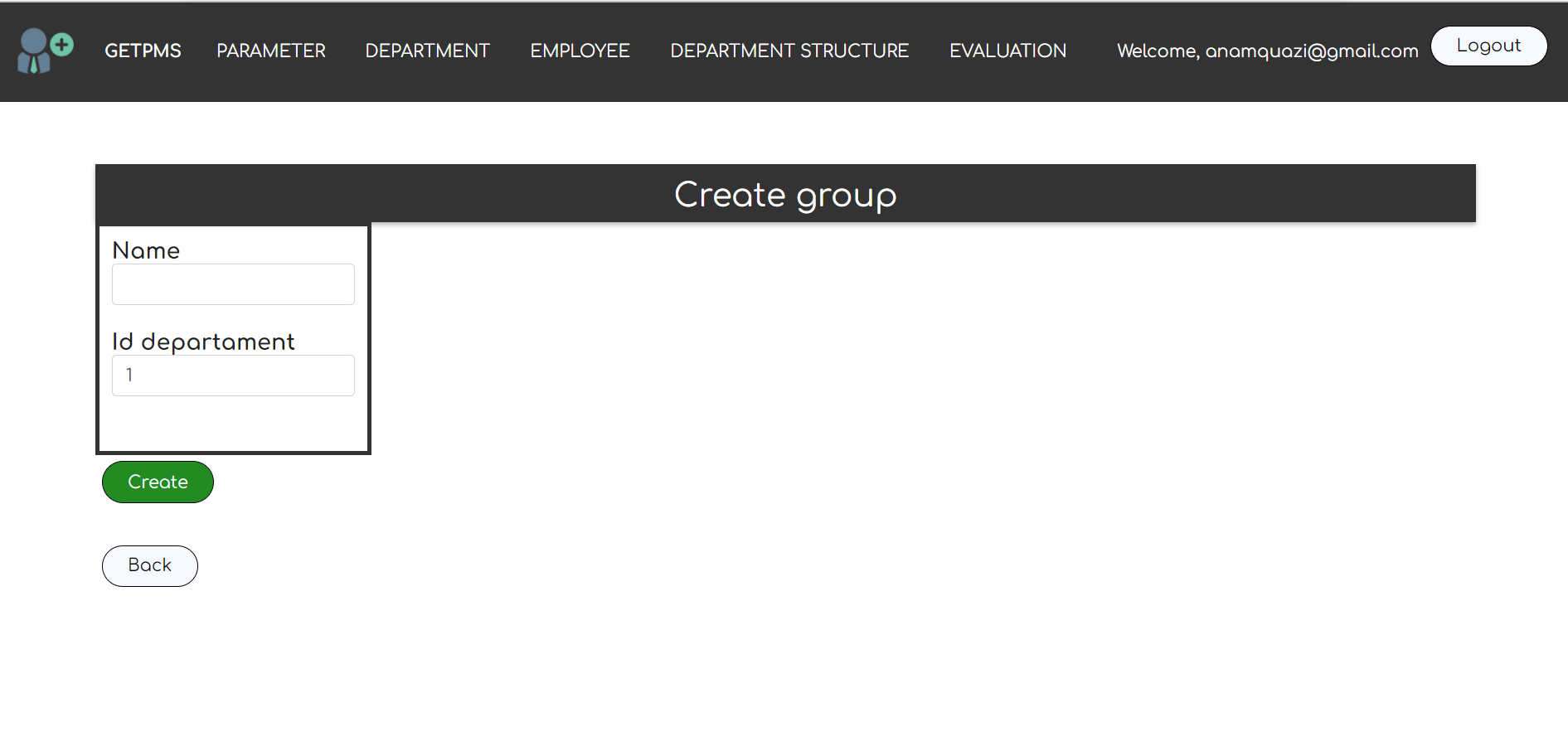
**5.6. Group Home Page**

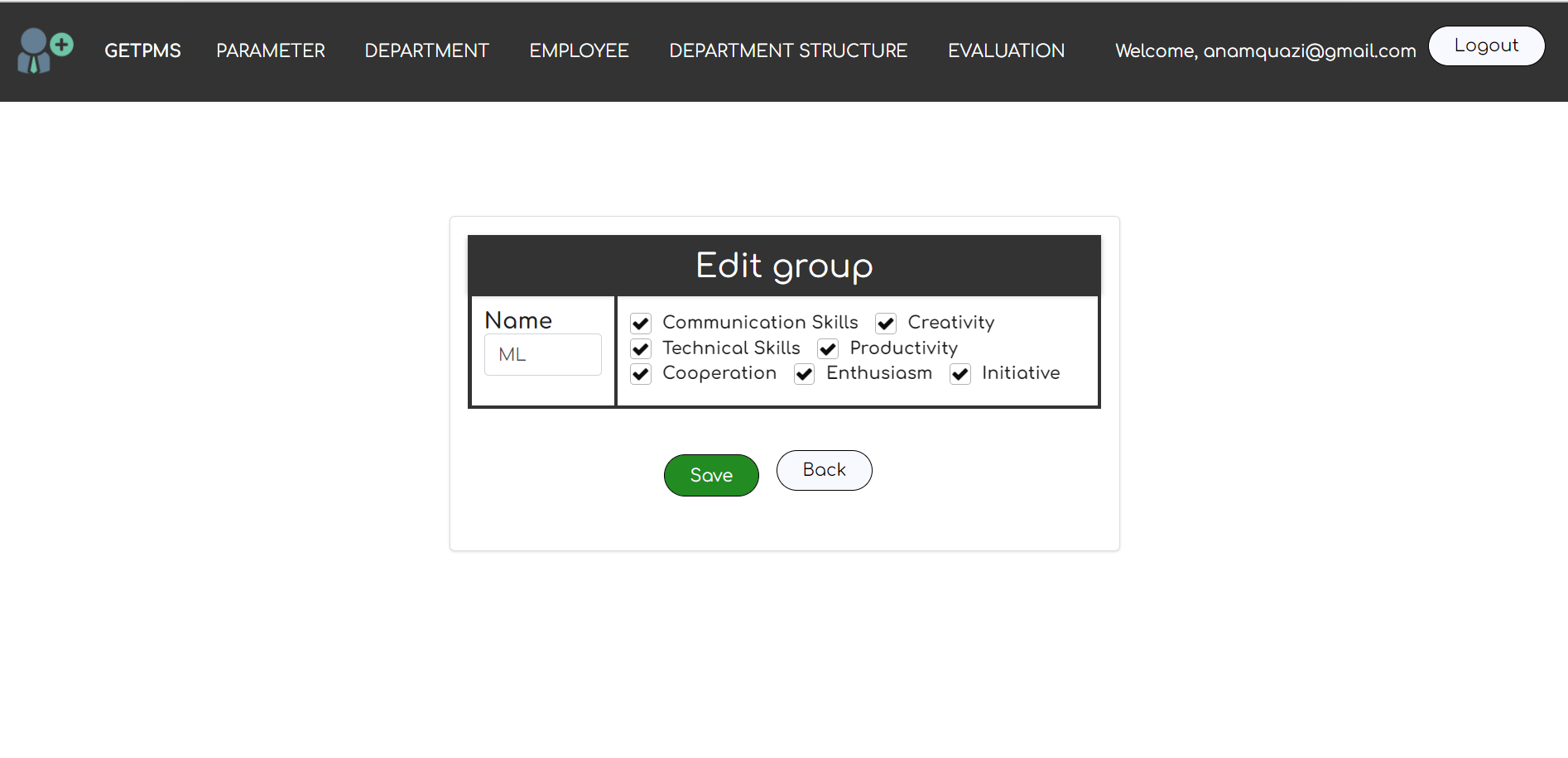
****

**5.6.2. Export to Excel**

****

**5.6.2. Create Group**

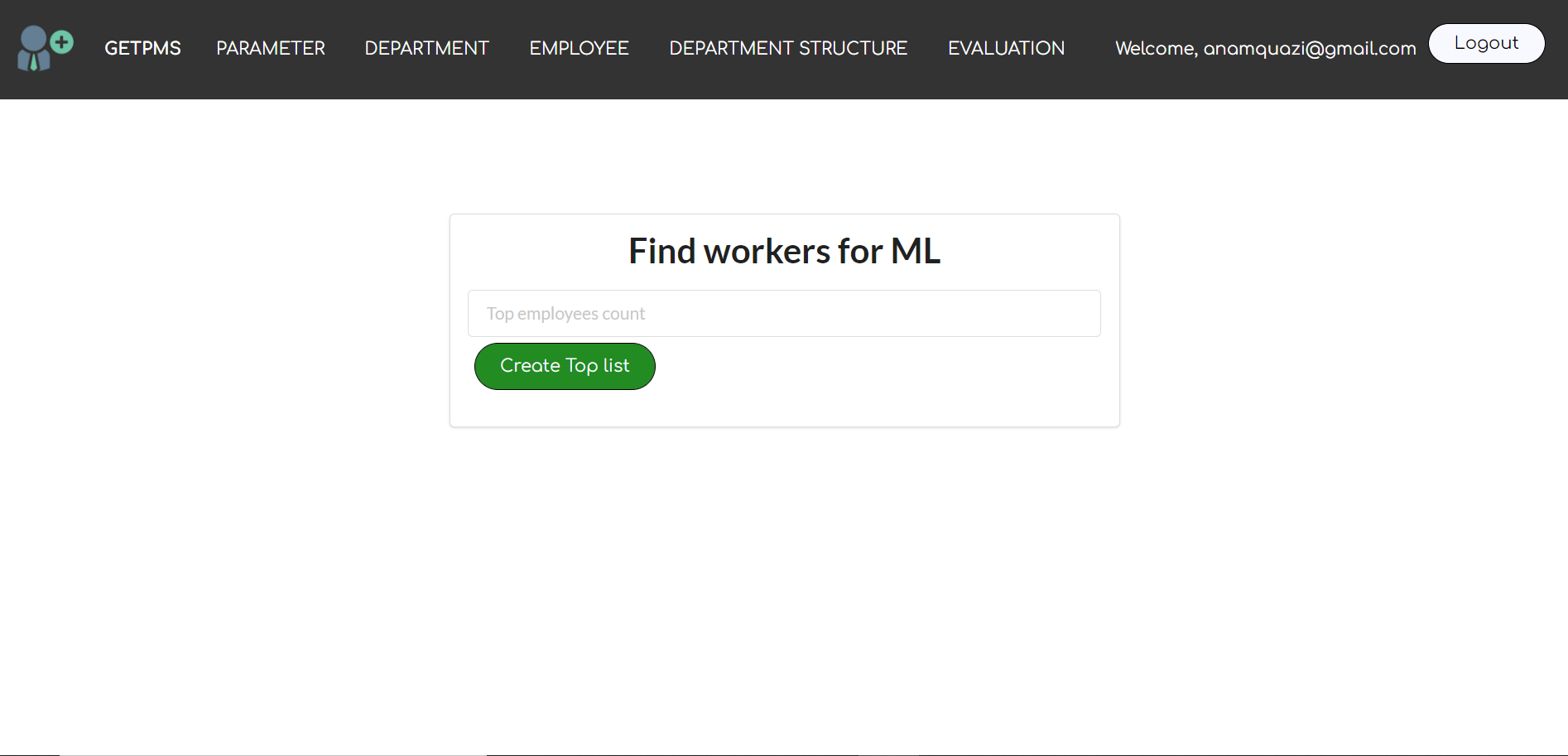
****

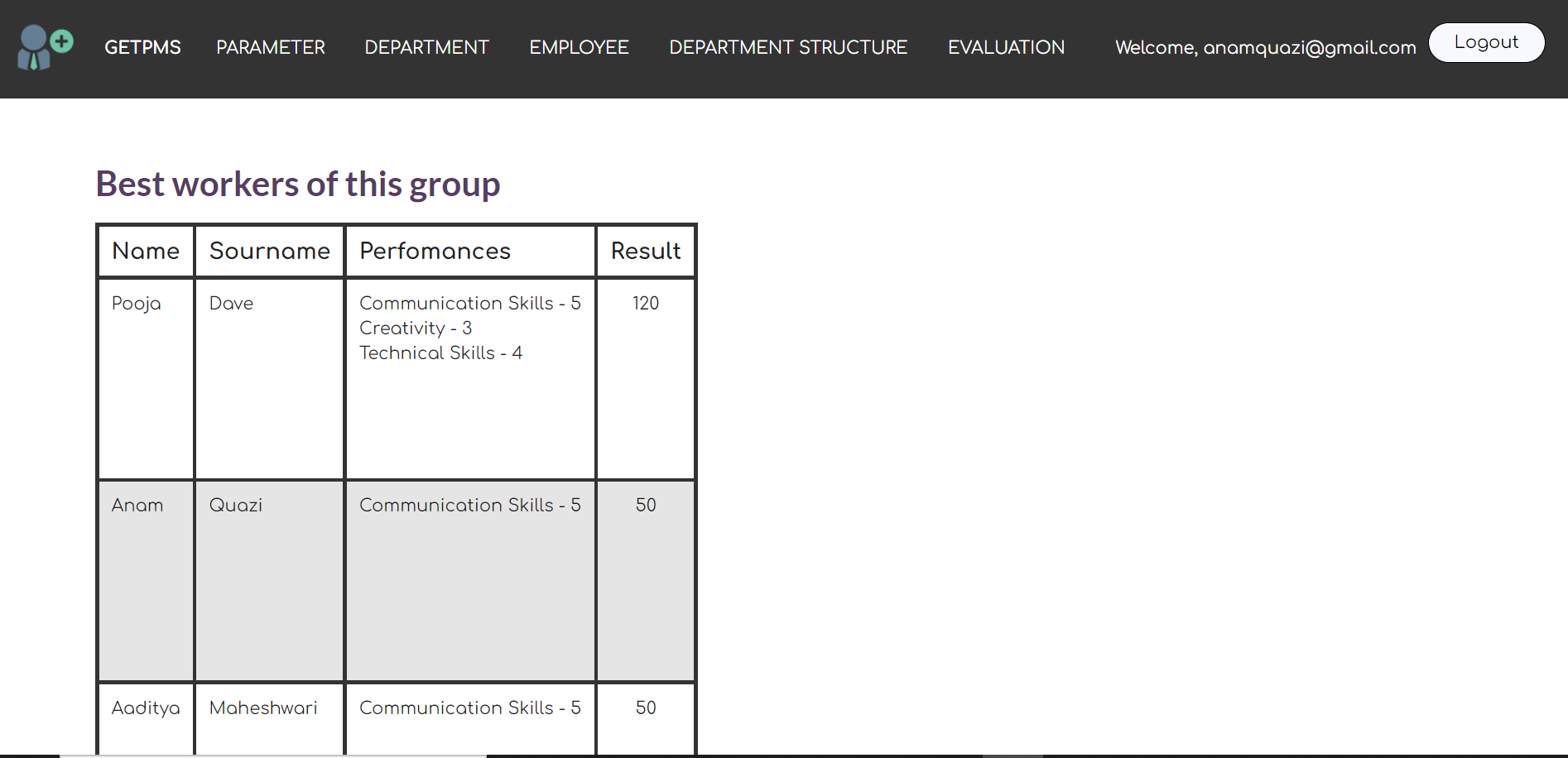
**5.63. Edit Group**

**5.6.4. Delete Group**

****

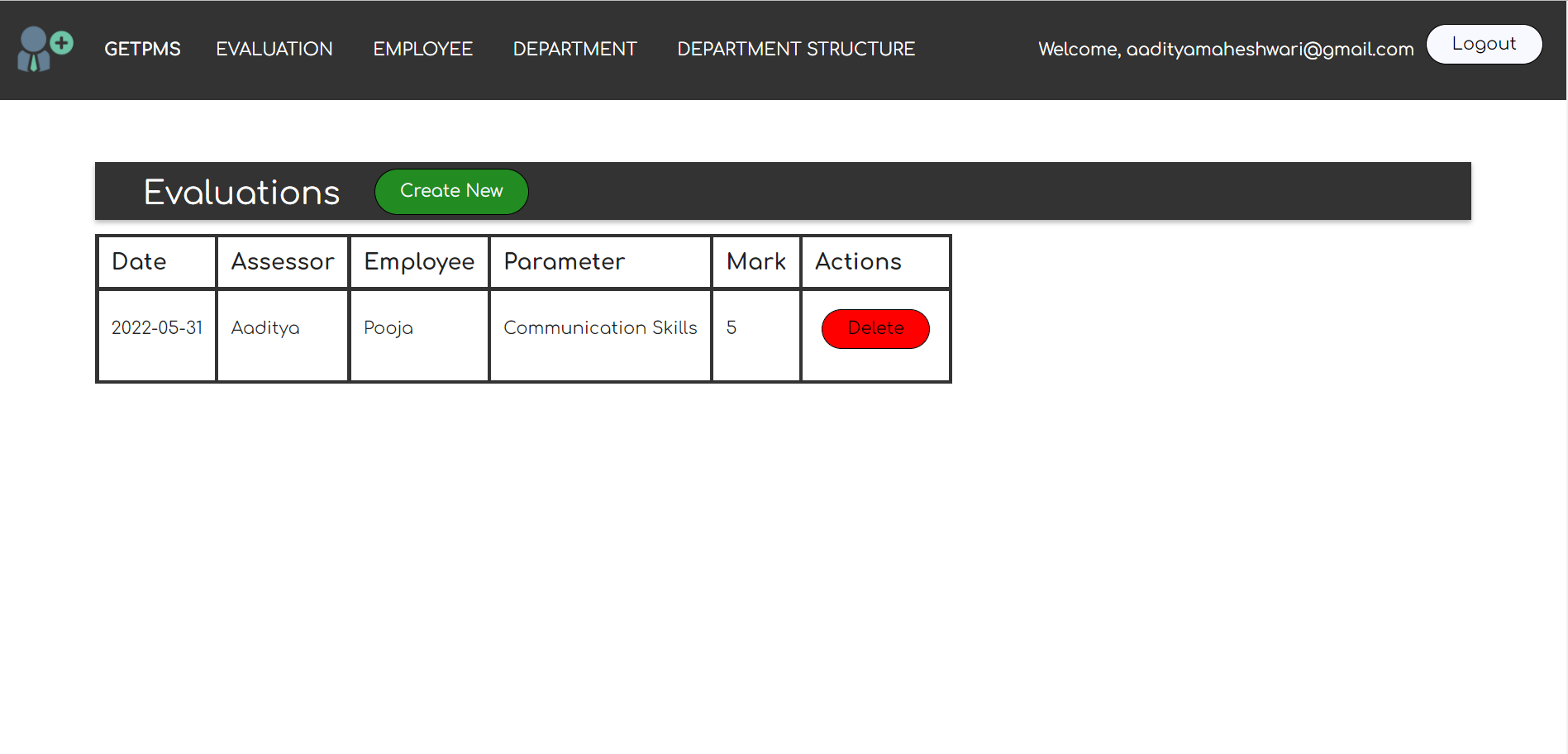
**5.6.5. Find Top Employees for each Group**

****

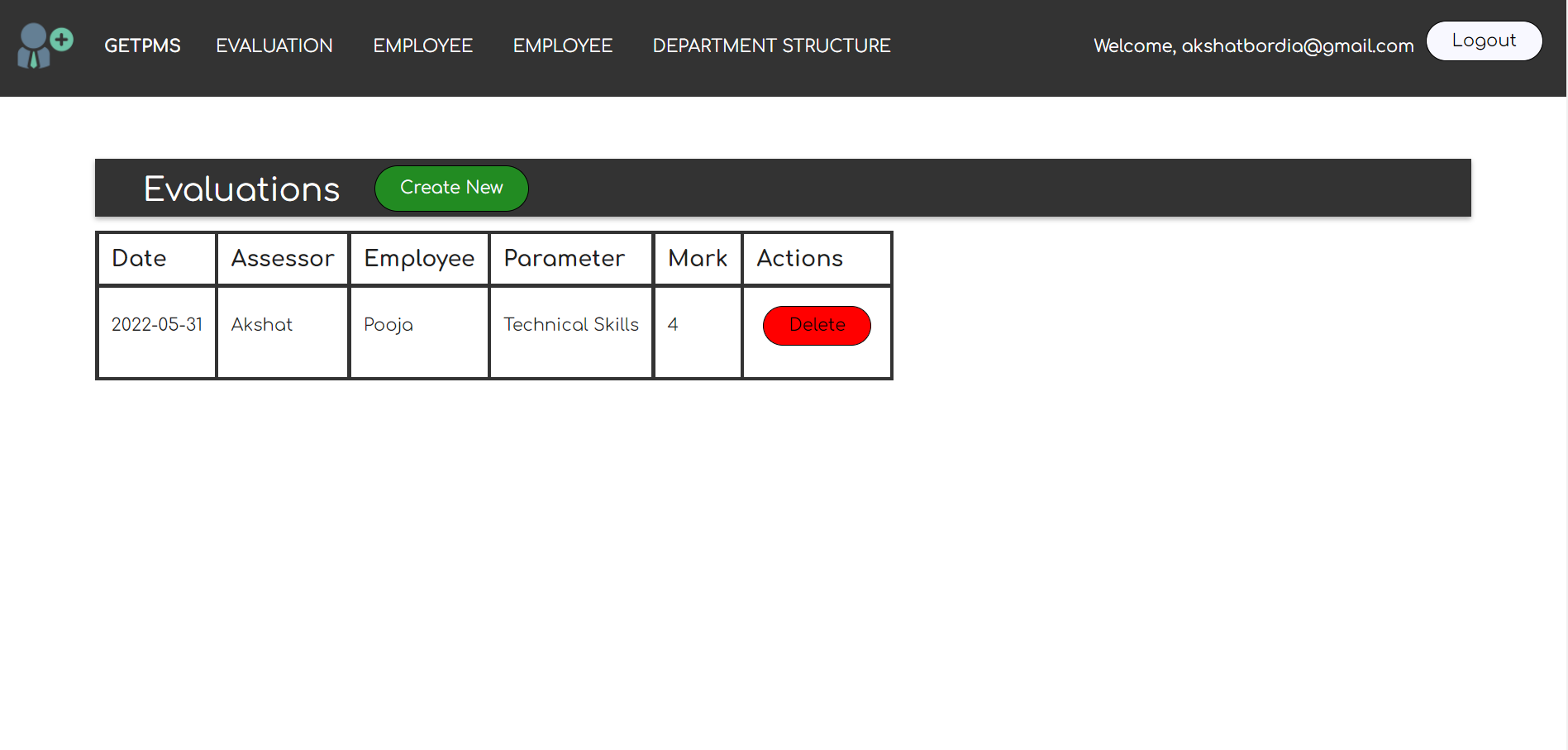
****

**5.7. Department Structure Home Page**

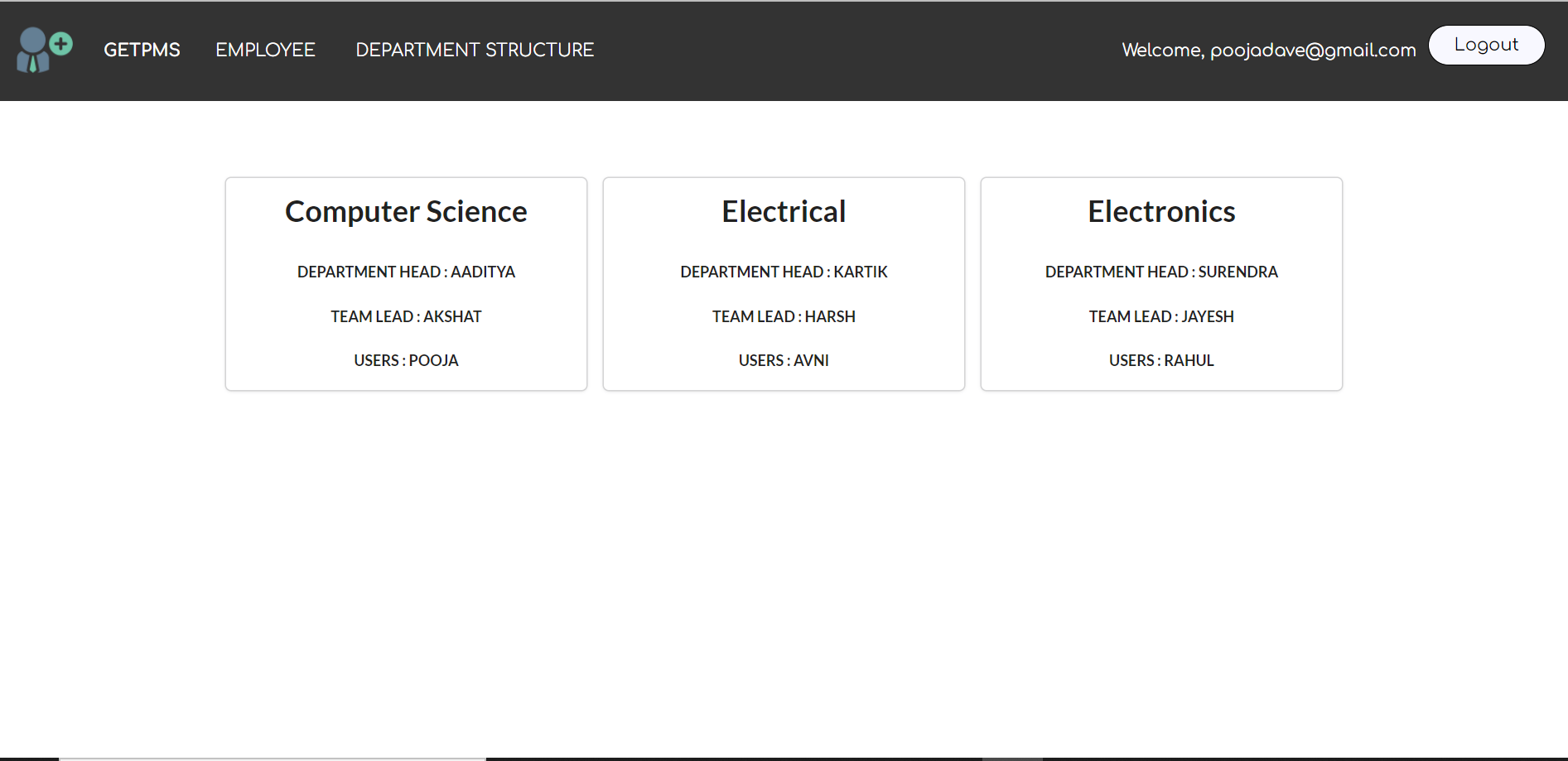
**5.1.2. Head Login**

****

**5.1.3. Team Lead Login**

****

**5.1.4. Employee Login**

****

# ***CHAPTER – VI***

# ***TESTING***

### **Testing**

##### **Testing Methodology**

Companies rely on software more than ever to provide and manage information with strategic and operational importance and to provide key decision support. Rising customer expectations for fault-free, requirements-exact software have increased awareness of the importance of software testing as a critical activity.

We begin the testing process by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verifies that the application meets the requirements specified in the system requirements document and is bug free. At the end of each testing day, we prepare a summary of completed and failed tests. Applications are not allowed to launch until all identified problems are fixed. A report is prepared at the end of testing to show exactly what was tested and to list the final outcomes.

Our software testing methodology is applied in three distinct phases: unit testing, system testing, and acceptance were testing.

**Unit Testing**:

The programmers conduct unit testing during the development phase. Programmers can test their specific functionality individually or with other units. However, unit testing is designed to test small pieces of functionality rather than the system as a whole. This allows the programmers to conduct the first round of testing to eliminate bugs before they reach the testing staff. In unit testing the analyst tests the programs making up a system.

For this reason, unit testing is sometimes called program testing. Unit testing gives stress on the modules independently of one another, to find errors. This helps the tester in detecting errors in coding and logic that are contained within that module alone. The errors resulting from the interaction between modules are initially avoided.

For example, a hotel information system consists of modules to handle reservations; guest checking and checkout; restaurant, room service and miscellaneous charges; convention activities; and accounts receivable billing. For each, it provides the ability to enter, modify or retrieve data and respond to different types of inquiries or print reports. The test cases needed for unit testing should exercise each condition and option.

Unit testing can be performed from the bottom up, starting with smallest and lowest-level modules and proceeding one at a time. For each module in bottom-up testing a short program is used to execute the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

**System Testing:**

The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together.

The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results with the expected results.

The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user who must verify the system and give the go-ahead.

During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way, users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results.

A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferable to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by people other than those who write the original programs. Using people who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software.

The system is tested as a complete, integrated system. System testing first occurs in the development environment but eventually is conducted in the production environment. Functionality and performance testing are designed to catch bugs in the system, unexpected results, or other ways in which the system does not meet the stated requirements.

The testers create detailed scenarios to test the strength and limits of the system, trying to break it if possible. Editorial reviews not only correct typographical and grammatical errors, but also improve the system’s overall usability by ensuring that on-screen language is clear and helpful to users. Accessibility reviews ensure that the system is accessible to users with disabilities.

System testing consists of the following five steps:

1. Program testing
2. String testing
3. System testing
4. System documentation
5. User acceptance testing

##### **Program Testing**

A program represents the logical elements of a system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. It is the responsibility of a programmer to have an error free program. At the time of testing the system, there exist two types of errors that should be checked. These errors are syntax and logic.

A syntax error is a program statement that violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted key words are common syntax errors. These errors are shown through error messages generated by the computer. A logic error, on the other hand, deals with incorrect data fields out of range items, and invalid combinations.

Since the logical errors are not detected by compiler, the programmer must examine the output carefully to detect them. When a program is tested, the actual output is compared with the expected output. When there is a discrepancy, the sequence of the instructions must be traced to determine the problem. The process is facilitated by breaking the program down into self-contained portions, each of which can be checked at certain key points.

##### **String Testing**

Programs are invariably related to one another and interact in a total system. Each program is tested to see whether it conforms to related programs in the system. Each part of the system is tested against the entire module with both test and live data before the whole system is ready to be tested.

##### **System Testing**

System testing is designed to uncover weaknesses that were not found in earlier tests. This includes forced system failure and validation of the total system as it will be implemented by its user in the operational environment. Under this testing, generally we

Take low volumes of transactions based on live data. This volume is increased until the maximum level for each transaction type is reached.

The total system is also tested for recovery and fallback after various major failures to ensure that no data is lost during the emergency.

All this is done with the old system still in operation. When we see that the proposed system is successful in the test, the old system is discontinued.

##### **System Documentation**

All design and test documentation should be well prepared and kept in the library for future reference. The library is the central location for maintenance of the new system.

##### **User Acceptance Testing**

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system's procedures operate to system specifications and that the integrity of important data is maintained. Performance of an acceptance test is actually the user's show. User motivation is very important for the successful performance of the system. After that a comprehensive test report is prepared. This report shows the system's tolerance, performance range, error rate and accuracy.

**Table 6.1 Test Report with test data**

|  |  |  |
| --- | --- | --- |
| **TEST REPORT WITH TEST DATA**  (To be filled by System Analyst/Programmer) | | |
| **Project Name: GET Performance Management System** | | |
| **S No.** | **Testing Parameter** | **Observations** |
| A. | INTERFACE TESTING   1. User-friendliness 2. Consistent menus | ok  ok |
| B. | CONTROL FLOW TESTING   1. IF-THEN-ELSE 2. FOR-EACH | ok  ok |
| C. | VALIDATION TESTING   1. Check for improper or inconsistent typing 2. Check for erroneous initialization or default values 3. Check for incorrect variable names 4. Check for inconsistent Data Types 5. Check for relational/arithmetic operators | okok okokok |
| D. | DATA INTEGRITY/SECURITY TESTING   1. Data Insertion/ Deletion/ Updating 2. Boundary condition(Underflow, Overflow Exception) 3. Check for unauthorized access of data 4. Check for data availability | ok  ok  ok  ok |

|  |  |  |
| --- | --- | --- |
| E. | EFFICIENCY TESTING   1. Throughput of the system 2. Response time of the system 3. Online disk storage required by the system 4. Primary memory required by the system | ok  ok  ok  ok |
| F. | ERROR HANDLING ROUTINES   1. Error description are intelligent/understandable 2. Error recovery is smooth 3. All error handling routines are tested and executed at least once | ok  ok  ok |

# ***CHAPTER – VII***

***CONCLUSION AND FUTURE ENHANCEMENTS***

### **Limitations**

The new system has been designed to meet almost all of the user requirements but it too has certain limitations some of which can be enhanced in future enhancements or updates.

**7.1.1 Risk of Internal Competition**

Under this system, employees compete with each other for job status, position and pay. This could amount to backstabbing, failure among team members to communicate efficiently and strong employee rivalry. It could lead to dysfunction of the department and/or team, resulting in failure to achieve performance standards.

**7.1.2. Favoritism**

Managers and supervisors tend to trust and depend on one employee more than the others. This employee could be the foreman or the team leader. This employee is entrusted with the responsibility of explaining new job roles and duties to other employees. It leads to distrust among the group members. It causes team fraction and adversely affects employee morale and satisfaction.

**7.1.3. Expensive and Time-Consuming**

Performance management systems are costly, requiring a lot of administrative work, patience and time. Performance management demands equipping employees with the "right" skills and knowledge. This means conducting extensive training, retraining and career development workshops for every division and employee level. This turns out to be a costly process. Further, projects are lost as employees are being trained with new improved skills.

**7.1.4. Manager's Dilemma**

The manager is unable to perform his tasks efficiently because he spends too much time supervising employees about their job functions. He is faced with value-based management systems. It becomes challenging and tough to decide value and performance indicators for measurement. It is not possible to have common indicators as each job has different job requirements. Managers are faced with information overload.

**7.1.5. Convoluted and Bureaucratic**

The company ends up hiring and training new personnel. Performance management creates new organizational layers. The employee population increases. Now, instead of one team doing a project, two teams are doing it. This actually affects the financial structure of the organization.

### **Future Enhancements**

Enhancements are the perquisite for development of a system. Every existing system has proposed enhancements which make it better and easier to use and more secure. The enhancements that have been proposed for this system are listed here.

##### **7.2.1. Improving the UI.**

The existing system needs to improve the user interface and make it more aesthetically appealing. It can be done through adding images, different color themes, animations and responsiveness.

##### **7.2.2. Facility to set Evaluation timeline.**

The new system will allow the administrator to set the timeline for evaluation of employees. For e.g., Evaluation can be done on a monthly basis and it needs to be completed before end of each month.

##### **7.2.3. Enable multi-OS support**

The new system will allow multi-OS support. It will allow the user to install, setup and access the system in different operating systems such as Mac and Linux.

##### **Reminders and Updates**

The new system will allow the administrator to send reminders for pending evaluations. It will also allow employees to view updates about training, discussion, meetings and events in the organization.

### **Conclusion**

Performance Management is not only an evaluation process of a person’s performance with reward as an outcome of it. Rather, its intent is to align and improve the performance of an individual to meet the overall organizational goal.

The supervisor and employee should come together to have a free flow of discussion on strengths, weaknesses, career aspiration and competency gaps, this reciprocation and involvement is important for performance management to yield any benefits.

`In all, the performance management system should help an employee in achieving the results efficiently and effectively thereby also fulfilling the organizational goals. Constant learning and improving based on performance management allows for more effective decision making followed by leading and influencing larger teams to achieve their goals.

# ***CHAPTER-VIII***

# ***BIBLIOGRAPHY***

### **Bibliography**

* **List of useful Websites**
  + - * [Developer tools, technical documentation and coding examples | Microsoft Docs](about:blank)
      * [W3Schools Online Web Tutorials](about:blank)
      * [.NET | Free. Cross-platform. Open Source. (microsoft.com)](about:blank)
* **List of useful Books**
* Parallel Programming with C# and .NET Core: Developing Multithreaded Applications Using C# and .NET Core 3.1 from Scratch by Rishabh Verma
  + - * C# 3.0: The Complete Reference by Herbert Schildt
      * Programming ASP.NET Core by Dino Esposito