Table of Contents

[List of Tables iii](#_Toc412115140)

[List of figure iv](#_Toc412115142)

[Acknowledgement vi](#_Toc412115143)

[Abstract vii](#_Toc412115144)

[Keywords/Acronyms viii](#_Toc412115145)

**[Chapter one](#_Toc412115146)** [1](#_Toc412115146)

[1.1 Introduction 1](#_Toc412115147)

[1.2 Back ground 1](#_Toc412115148)

[1.3 Statement of problem 2](#_Toc412115149)

[1.4 Objectives 2](#_Toc412115150)

[1.4.1General objectives 2](#_Toc412115151)

[1.4.2 Specific objectives 3](#_Toc412115152)

[1.5 Scope of the project 3](#_Toc412115153)

[1.6 Significance of the project 4](#_Toc412115154)

[1.6.1 Benefits for the human resource office 4](#_Toc412115155)

[1.6.2 Benefits for the applicant 4](#_Toc412115156)

[1.6.3 Benefits for the employee 4](#_Toc412115157)

[1.6.4 Benefits for the Department 4](#_Toc412115158)

[1.7 Feasibility study of the project 5](#_Toc412115159)

[1.7.1 Economic Feasibility 5](#_Toc412115160)

[1.7.2. Technical Feasibility 6](#_Toc412115161)

[1.7.3. Operational Feasibilities 7](#_Toc412115162)

[1.7.4. Schedule Feasibility 7](#_Toc412115163)

[1.8 Methodology 7](#_Toc412115164)

[1.8.1 Fact finding Techniques 8](#_Toc412115165)

[1.8.2 System analysis and design methods 8](#_Toc412115166)

[1.8.3 Tools Used in the Project 9](#_Toc412115167)

**[Chapter two](#_Toc412115168)** [10](#_Toc412115168)

[System analysis 10](#_Toc412115169)

[2.1. Introduction 10](#_Toc412115170)

[2.2. Existing system description 10](#_Toc412115171)

[2.2.1. Player of the existing system 10](#_Toc412115172)

[2.2.2. Work Flow of the Existing System 11](#_Toc412115173)

[2.2.3 Report generating in the existing system 11](#_Toc412115174)

[2.2.4. Problem of Existing System 11](#_Toc412115175)

[2.2.5 Explanation of the business rule of the organization 12](#_Toc412115176)

[2.2.6 CRC (class responsibility and collaboration) modeling 12](#_Toc412115177)

[2.3. Proposed system description 14](#_Toc412115178)

[2.3.1 Purpose of the new system 14](#_Toc412115179)

[2.3.2 Functional requirements 15](#_Toc412115180)

[2.3.3. Non-functional requirements 16](#_Toc412115181)

[2.4. System model and artifacts 16](#_Toc412115182)

[2.4.1. Use case diagram 16](#_Toc412115183)

[2.4.2 Use case description 18](#_Toc412115184)

[2.5. Dynamic models 27](#_Toc412115185)

[2.5.1. Sequence diagram 27](#_Toc412115186)

[2.5.2. Collaboration diagram 34](#_Toc412115187)

[2.5.3. Activity diagram 37](#_Toc412115188)

[2.6. Static model 41](#_Toc412115189)

[2.6.1. State chart diagram 41](#_Toc412115190)

[2.6.2. Class diagram 44](#_Toc412115191)

**[Chapter three](#_Toc412115192)** [46](#_Toc412115192)

[Design deliverables of the new system 46](#_Toc412115193)

[3.1 Introduction 46](#_Toc412115194)

[3.2 Design goals and objectives 46](#_Toc412115195)

[3.3. System architecture 47](#_Toc412115196)

[3.4 system decomposition 48](#_Toc412115197)

[3.5. Component modeling 51](#_Toc412115198)

[3.6. Deployment diagram 52](#_Toc412115199)

[3.7 user interface prototyping design 53](#_Toc412115200)

[3.8. Design of class diagram (class mapping) 54](#_Toc412115201)

[3.9. Data base design 56](#_Toc412115202)

**[Chapter 4](#_Toc412115203)** [58](#_Toc412115203)

[System implementation 58](#_Toc412115204)

[4.1 user interface 58](#_Toc412115205)

**[Chapter 5](#_Toc412115203)** [63](#_Toc412115203)

[5.1. prototype development 63](#_Toc412115204)

**[Chapter 6](#_Toc412115203)** [86](#_Toc412115203)

[6.1. Conclusions 86](#_Toc412115206)

[6.2. Recommendation 86](#_Toc412115207)

[References 87](#_Toc412115208)

# List of Tables page

# [Table 1.1: Miscellanies Costs......................................................................6](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852777)

[Table 1.2: Software Development Costs………………………………………….…6](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852778)

[Table 1.3: Time Schedule for the Project......................................................7](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.1 CRC (class responsibility collaboration)......................................13](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.2 Main Course of Action about Login.............................................18](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.3 Main Course of Action about employee requisition......................19](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.4 Main Course of Action about post announcement.......................20](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.5 Main Course of Action about registering applicant online............21](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.6Main Course of Action about placement......................................21](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.7: Main Course of Action about Update employee…………………….22](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852780)

[Table 2.8: Main Course of Action about searching……………………….........22](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852793)

[Table 2.9: Main Course of Action about deleting employee…………………..23](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852793)

[Table 2.10Main Course of Action about applies leave.................................24](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852793)

[Table 2.11Main Course of Action about send employee leaves app...........24](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852793)

[Table 2.12Main Course of Action about approve leave application.............25](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852793)

[Table 2.13Main Course of Action about notification.................................26](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852793)

[Table 2.14Main Course of Action about report..........................................26](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852793)

# List of figure page

[Figure2.1 use case diagram...................................................................17](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.2 sequence diagram for login...................................................28](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.3 sequence diagram for employee requisition...........................29](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.4 sequence diagram for post announcement............................30](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.5 sequence diagram for view and register.................................31](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.6 sequence diagram for placement...........................................32](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.7 sequence diagram for apply leave..........................................33](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.8 collaboration diagrams for login ...........................................34](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.9 collaboration diagram for employee requisition.....................35](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.10 collaboration diagram for post announcement....................35](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.11 collaboration diagram for view and register.........................36](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.12 collaboration diagram for placement...................................36](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.13 collaboration diagram for apply leave..................................37](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.14 activity diagrams for login………………………………………...38](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.15 activity diagrams for employee requisition..........................38](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.16 activity diagram for post announcement.............................39](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.17 activity diagrams for read and apply...................................39](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.18 activity diagrams for placement..........................................40](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.19 activity diagram for apply leave..........................................40](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.20 state chart diagram for login..............................................41](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.21 state chart diagram for employee requisition......................42](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.22 state chart diagram for vacancy announcement.................42](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 5.23 state chart diagram for view and register............................43](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.24 state chart diagram for placement......................................43](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.25 state chart diagram for apply leave.....................................44](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 2.26 analysis level of class diagram…………………………………..45](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure3.1.system architecture.........................................................48](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 3.2 system decomposition………………………………………….......50](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure3.3. Component diagram..........................................................51](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure3.4.Deployment diagram..........................................................52](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure3.5. User interface prototype design.........................................53](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure3.6. Design of class diagram (class mapping)...........................55](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure3.7. Data base design diagram.................................................57](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

# User interface page

[Figure4.1 user interface for login..............................................................58](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 4.2 user interfaces for employee requisition....................................59](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 4.3 user interfaces for apply...........................................................59](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 4.4 user interfaces for view registration...........................................60](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 4.5 user interfaces for placement....................................................61](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 4.6user interfaces for leave application.........................................61](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

[Figure 4.7user interfaces for message.......................................................62](file:///C:\Users\%601\Documents\Visual%20Studio%202010\Projects\AudioSignalManipulatingSystem\Final%20documentationV2.docx#_Toc327852577)

# Acknowledgement

Firstly, we are greatly indebted to the following for their contribution to our project: The Almighty God, who gave us the insight and perseverance to accomplish this project. Our Advisor Mr. MERHAWI who’s sounding advice helped us guide this project in the right direction, in which he can enforce us to have a massive knowledge about the project to easily familiarize with the concept of project development. And also for Mr. FILIMON for the solid knowledge-base that enabled us carryout the project.

Secondly we would like to thank Debre Berhan University HRMS personnel officer and the department officers especially Ato. ALEMU ABEBE and W/ro SIRGUT H/MARIAM for their partial willingness of interview, patience in answering to our numerous questions, giving existing activity materials and reading materials that help us to precede our project.

# Abstract

This system document is prepared for employee requriment management system of DBU and concerned with the description of the existing Human Resource management system and different types of models used to model the new system under study. Web based human resource management system mainly provides effective and fast data processing and controlling of personnel. This web based system of managing human resource in our University setting is expected to help various services keep an updated data on the status of their employee personal information. In designing such a system, PHP has been engaged as a development language and MySQL as a backend database with CSS implemented for the interface. Generally, the main goal of online human resource management system is to shorten data-processing time, to reduce errors, to improve the accuracy of input and to provide data reliability of the personnel.

# Keywords/Acronyms

ERMS employee requriment management system

DBU Debre Berhan University

CGPA Cumulative Grade Point Average

UML unified modelling language

DBMS database management system

DB database

GUI graphical user interface

ID identification

UC use case

UI user interface

HTML Hypertext modelling language

SQL structure query language

PHP Hypertext pre-processor

INFO Information

OOSD object oriented system development

OOA Object-oriented analysis

OOD object-oriented design

App \_ID Applicant identification

EMP\_ID employee identification

HR\_ID human resource admin identification

DPT\_ID department identification

CSS cascading style sheet

HTTP hypertext transfer protocol

CRC class responsibility collaboration

BR business rule

# Chapter one

## 1.1 Introduction

In this world of growing technologies everything has been computerized. With large number of work opportunities the Human workforce has increased. Thus there is a need of a system which can handle the data of such a large number of Employees. The aim of this project is to develop aweb based system for DBU human resource management. While DBUhas established italso opened an office which is called human resource management. Currently, the DBU human resource offices process data manually. The manual processing system has many problems .In order to solve this issue; we have developed a web based human resource management system. The completed project solves the problems that had affected the human resource management offices. Since it is online it reduces a lot of costs, time to travel to the offices, work over load and it minimizes the space used to store the data. Online human resource management system enables to register applicants online, search, update employees’ data, and online registration, and placement, online leave application and report generation.

## 1.2 Back ground

DebreBerhan University is officially established in June 3/ 1999 E.C During this time it comprised three faculties. These are natural science, social science, Business and Economics Faculty. At present the university runs undergraduate and post graduate program. It is expected that with the on-going expansion a number of faculties and programs will be opened and the enrolment capacity will increase. The DBU human resource management was established during this time and it is developed from time to time with the development of the university. As it is developed from time to time it increases its capacity by increasing the number of workers in the office. The DBU human resource management is back bone of the university by managing and effective utilization of human resources.

## 1.3Statement of problem

Currently the office is facing many problems due to the use of manual handling system in its day to day activities. The problems are:-

* Difficulty of searching and modifying employee information.
* take more time for simple manual jobs
* Since there are a lot of employee documents in the office it’s hard to manage such huge data manually.
* Lack of accuracy and loss of document
* Since the office use manual system, the mechanism of data handling is unsecured.
* There is a problem in procedures such as leave management when an employee is required to fill in a form which may take several weeks or months to be approved.
* The use of paper work in handling some of these processes could lead to human error, papers may end up in the wrong hands and not forgetting the fact that this is time consuming.

## 1.4 Objectives

### 1.4.**1General objectives**

The general objectives of this project is to implement a web based system for employee requriment management in Debre Berhan university that will bring up a major paradigm shift in the way that employee profile information is handled in a well manner.

### 1.4.2 **Specific objectives**

The specific objectives of our project are:-

* Study and analyse the limitations of the existing system.
* Suggest alternative solutions in order to address the current problem.
* Understand functional and non-functional requirements of the system.
* Develop a system that facilitates fast report generation.
* Increase to the work efficiency of the office.
* Developa system that makes retrieval of data efficient.
* Implementation of the proposed system in efficient way.
* Giving recommendation on further studies to be conducted on the areas of human resource management system.

## 1.5 Scope of the project

The scope of this project will be limited to the following:

* **Announcement process:** the human resource will announce the vacant position to the applicant.
* **Online Registration of Applicants**:-The system registers all the applicant information that is used to hire in the institution.
* **Placement process:** the human resource will place the new employee to their respected positions.
* **Manage the employee profile:** the respected departments will manage the new employee as well as the existing ones.
* **Leave management:** Complete elimination of paperwork in leave management by enabling an employee applies for leave through the system. This will also enable the department to accept/reject leave application through the system.
* **Message:**-the user like employee, department and HR admin can communicate with each other through this system.
* **Report generation:** The departments can generate reports about the employee status or progress to human resource admin order to monitor employees and this can be used for performance appraisals. The reports will have all the information of an employee from educational background, trainings attended, projects done as well as technical skills.

## 1.6 Significance of the project

The significance of the new system includes:

### 1.6.1 Benefits for the human resource office

* Avoiding improper resource consumption like paper, pen…
* Avoiding data loss because of improper data storage
* Enhance security mechanisms to protect employee information.
* Simple process for posting job vacancy.
* To handle the applicant effectively and support the smooth functioning of the business.
* reduces work over load of the office

### 1.6.2 Benefits for the applicant

* Multiple channels to access services from the institution
* Simplified process forregistering online
* Can view posted information’s from anywhere at any time

### 1.6.3 Benefits for the employee

* Can view notification send by the department and HR office from anywhere at any time
* Can be able to fill in leave application form in the appropriate fields online.
* Can also view their personal information through the system.

### 1.6.4 Benefits for the Department

* Facilitates fast and efficient retrieval of data
* Avoiding improper resource consumption like paper, pen, ledger…
* Reduced workload of the office activities such as preparation of reports
* Simplified process for managing employee information

## 1.7 Feasibility study of the project

### 1.7.1 Economic Feasibility

The web based employee requriment management system is economically feasible because:

* The system provide fast and efficient automated environment.
* The system has GUI interface and very less user-training is required to learn it.
* This project is economically feasible because its anticipated benefit is greater than the expected cost.

#### Cost of the Project

##### **Tangible Costs**

The tangible costs acquired in developing the system are:-

* Miscellaneous Cost which includes hardware development cost and other costs.
* Software development cost
  + - * 1. **Miscellaneous Cost**

This cost contains the various types of costs in which we spent for the development of the project or the University covers some of the hardware expenses.

The following table lists the different miscellanies costs that we havespend in the process of the development of the system.

|  |  |  |
| --- | --- | --- |
| Resources | Amount | Price |
| Pen | 12 pen | 60 Birr |
| Printing | 120 pages | 120 Birr |
| Paper | 1 DESTA | 90 Birr |
| Flash disk | 8gb | 450 Birr |
| CDRW | 6 | 60 Birr |
| Laptop | 1 | 12000 Birr |
| Desktop | 1 | By university |
| Total |  | 12860 Birr |
|  | | |

Table 1.1: Miscellanies Costs

* 1. **Software Development Cost**

For this particular project we had used different software but the software is provided by the university.

|  |  |
| --- | --- |
| Type of Software’s | Price |
| Microsoft windows 7/8/XP | by the university |
| Microsoft Office | by the university |
| MySQL fromXampp | Free down load |
| Visual Paradigm for UML 11.0/Microsoft Visio /Rational Rose | Free download |
| Notepad++ | Free download |
| Total | 0.00 Birr |

Table 1.2: Software Development Costs

##### **Intangible Costs**

Those are costs which are uncountable .The intangible costs that are acquired in developing the system are:-Our knowledge that we will spent to develop the system may not be measureable in terms of money.

### 1.7.2. Technical Feasibility

We have technical knowledge about:

* PHP to write the code or implementation with XAMPP.
* MySQL to build the database to store the data.
* Unified Modelling Language (UML) model to do analysing and designing in good manner.

The technical requirement for the human resource management system in order to do their operation by the new computerized system is:

* Training on the new system to know how it operates and how to use the computerized system.

### 1.7.3. Operational Feasibilities

It determines how the system satisfies the organizations need and it also offers Secure, accurate and efficient system to the organization.

The system in which we have developed is also compatible to all operating systems and web browsers.

### 1.7.4. Schedule Feasibility

Within the time duration, we have identified the activities of the project in order to accomplish the project objective within their schedule requirement which is on the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Activities | Submission date | | | |
| **November** | **December** | **January** | **February June** |
| 1 | System proposal |  |  |  |  |
| 2 | Requirement analysis |  |  |  |  |
| 3 | System design |  |  |  |  |
| 4 | Implementation |  |  |  |  |

Table 1.3: Time Schedule for the Project

## 1.8 Methodology

We have used interview and document analysis in order to determine the information, which is used in the existing system and very important to develop the web based system.

### 1.8.1 Fact finding Techniques

**A. Interview** we have gathered information by interviewing the HRMS officers about the existing system. We have prepared some questions to the human resource management system manager **Ato. ALEMU ABEBE and W/roSIRGUT H/MARIAM** to get the necessary information about the existing system

**B. Practical observation**

It helps us to get real information how the organization performs its function and this helps to strength the data that gathered through interview and document analysis.

**C. Document analysis**

This technique provides information on how the existing system works .There for documents related to the existing system of the organization will be assessed.

### 1.8.2 System analysis and design methods

In this project, our team have used object oriented system development methodology (OOSD) for the design.

This technique has several phases some of them are:

#### Object Oriented Analysis (OOA)

During this phase the team uses to model the function of the system (use case modelling), find and identify the business objects, organize the objects and identify the relationship between them and finally model the behaviour of the objects in detail.

#### Object Oriented Design (OOD)

During this phase our team uses Microsoft Visio software to refine the use case model and rational rose for designing the sequence, collaboration, activity diagrams and to model object interactions and behaviour that support the use case scenario.

### 1.8.3 Tools Used in the Project

#### Back End Design Tool

MySQL software of the data base system and PHP language was used in developing and managing the back end of the system.

#### Front End Design Tool

The user interface had been developed using html, java script, CSS since it easily designing the front end and connected in to data base easily.

# Chapter two

## System analysis

### 2.1. Introduction

In this chapter the existing system of DBU ERMS was clearly defined by answering how existing system is working? In what way the employee managed? What are techniques being used to handle personnel file? What are the business rules of the existing system? And what are the problems in the existing system? After studying the existing system it is possible to understand that how the proposed system can solve the existing system problems. Studying the existing system will also use to determine both functional and non-functional requirements.

## 2.2. Existing system description

The existing human resource management system performs the following function with manual system and this leads to less security issues. Because of the manual system recording system is time consuming and boring .This is the result of lack of computerized system or web based system.

### 2.2.1. Player of the existing system

**Personnel officer: -** is a person who is in charge of controlling and following up all the employees’ Activities and departmental communications using acceptable and approved documents.

**Staff workers:-**are those employee works in HRMS office.

**Section/Department: -** A business unit in which all employees are included.

**Applicants: -**they are person who find job.

**Employees: -**they are person who works in the institution (could be lecturers, accountants, technicians)

### 2.2.2. Work Flow of the Existing System

The work flow in the existing system is performed starting from the human resource manager to the employee in every department. The human resource announces vacant position and hired employee and record the specified information about the employee in manual based system. After hiring the employee then send the hired employee data to that of departments who require the employee in letter form. And the applicant departments accept the employee and place them to their tasks. If the employee gets training, done projects the departments update the new information of the employee in manual or paper based system. As we see the existing system all activity are done from the human resource office to every departments in the institution are manual base system.

### 2.2.3 Report generating in the existing system

Different reports are generated in the existing HRMS of DBU. Reports are generated based on the general information about an Employee in the university. This report helps the finance department in predicting budget needed for the coming years and months, helps the Departments to ask for additional employee and so on. Other types of reports are prepared monthly depending on Registered Employee per month and annually.

### 2.2.4. Problem of Existing System

The existing system has many problems. Those problems are:-

* Lack of accurate data ,
* The system has problems related to security.
* It takes long time to search data ,
* Lack of efficiency, effectiveness,
* Since it operable manually it uses large amount of space to store data.
* Lack of portability’s,
* Redundancy of data

### 2.2.5 Explanation of the business rule of the organization

In every organizations or institutions there are rules and policy, which used to govern all activities in specified work flow, control the work flow, and performed in the work environment.

**BR1**: To get employee the departments who needs employee should write an application letter for their vacant position to personnel department

**BR2**: when the vacant position is announced to external applicant on notice board on mass media externally for consecutive 5 to10 work days.

**BR3**: To be employed applicants should bring a clearance letter from previous Employer.

**BR4**: when the human resource hired the new employee to that of departments who needs an employee they must send a letter that has full documents about the new employee.

**BR5**: Access of information depends on the authority of the user.

**BR6**: the employee must have full reason to apply leave application.

**BR7**: if the employee wants to leave from the institution for different reason first they must apply to the department and the department must send their application to the human resource.

**BR8**:if one wants to leave from DBU before he/she fills the form leave form he/she must return all working material to respected department otherwise they will be rejected.

### 2.2.6 CRC (class responsibility and collaboration) modeling

CRC modelling techniques are used to model the existing system that help as build the proposed class diagram by observing how the existing system are collaborate with each activity . A class represents a collection of similar objects, a responsibility is something that a class knows or does, and a collaborator is another class that a class interacts with to fulfil its responsibilities.

**Table 2.1** CRC (class responsibility collaboration)

|  |  |
| --- | --- |
| **announcement** | |
| Salary  quantity  end date  Application for vacant position  job no  date | Application |

|  |  |
| --- | --- |
| **pressonnel officer** | |
| Post announcement  Give job no  Record announce date | announcement |

|  |  |
| --- | --- |
| **Application** | |
| Name of applicant  address  CGPA  Marital status  Age  Gender | Applicant |

|  |  |
| --- | --- |
| **Department officer** | |
| Manage employee  Employee id  Status  Manage employee  education level  position | employee |

|  |  |
| --- | --- |
| **Employee** | |
| Name  Id  Status  Education level  position  apply for leave | Leave application |

|  |  |
| --- | --- |
| **Leave application** | |
| Reason  leave status | Approve leave application |

## 2.3. Proposed system description

The proposed system is designed to eliminate all the drawbacks of the existing human resource management system. The system shall be responsible for posting announcement, hiring new employees, placing the new employee to their respected position, and managing information about employees to their personal profile. And also the system shall incorporate leave management all the way from application to acceptance/rejection of leave requests as well as all in generating reports.All these features include the ability to add user, update (edit), and retrieve through search results. It also contains a report generation system.

### 2.3.1 Purpose of the new system

The system have provided

* **Security:** since the system requires verification of logon form, sensitive information’s will not be accessed or modified by unauthorized users.
* **Efficient retrieval of employee files:** since the system record each and every employee’s information on the data base, retrieval of employee files from the database at any time is a very easy process.
* **Efficient way of employee’s data management**:-Since the system uses database system there is no loose of data .The employee information is highly secured, the search and update of employee is simple.
* **Give online information registration**:-The system gives online information about vacant position for the applicant from the institution, so to know the criteria that must be full fill to register and also can register online at a time.
* **Give’s Notification information**:-the system gives notification through posting in the website and email notifications

### 2.3.2 Functional requirements

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Statements of services the system should provide how the system should react to particular inputs and how the system should behave in particular situations.

* **Posting announcements:** the system post vacancy announcements and notice for the applicants in order to make them informed.
* **Register applicant online:** The system registers applicants’ who wants to be hired in the institution with appropriate information. Without coming to the office they can be able to register online by using the system.
* **Placing new employee:** the system is able to recorded the new employed applicants and able to place to their respected position.
* **Manage employee information:** The system is able to search, delete and update the hired employee information when it is needed. And also the department can assign project team for every employee in his department.
* **Leave Application/Approval**
* Leave application- The user is able to fill in leave application form in the appropriate fields.
* Leave approval- The department are able to approve leave applications based on the reasons stated.
* **Message:** The system is can able to communicate the users.
* **Report generation**
  + Report generation- the system is able to generate a report for each employee based on the information in the database.

### 2.3.3. Non-functional requirements

Non-functional requirements are requirement, which has no essential for the system, but it can support and give more quality for the system.

1. **Users interface requirement**

* User interface should be menu driven and attractive.
* The interface should be user friendly.
* The system should support error-handling mechanism that display graphic approach and the system guide the user what will be the next action.

1. **Security Requirement**

* The system support user name and password to authentic.
* The system has different privilege to protect intruding.

1. **Performance: the system is responsibly fast in order to access the required crime information’s**
2. **Error handling requirement**

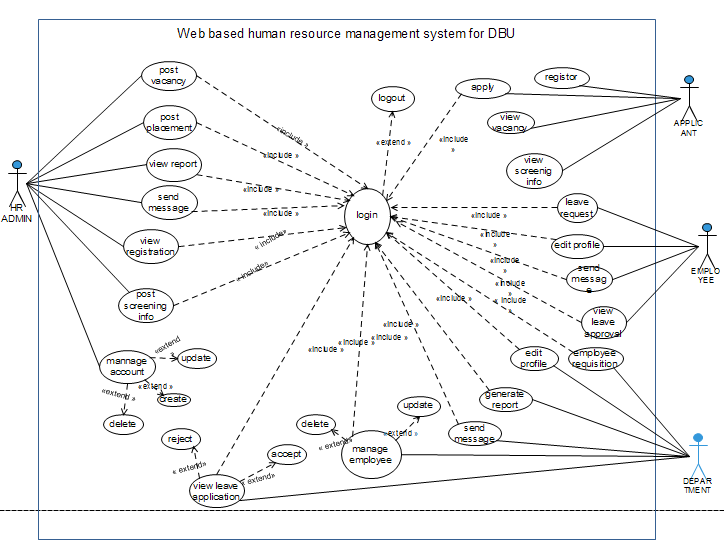
The system have error handling mechanisms that is, as errors occur it will not stop functioning rather provide error manages and should guide the user through what to do next.

## 2.4. System model and artifacts

### 2.4.1. Use case diagram

A use case defines a goal-oriented set of interactions between external users and the system under consideration or development. Thus a Use Case Scenario is a description that illustrates, step by step, how a user is intending to use a system, essentially capturing the system behaviour from the user's point of view.  
In order to create relevant use cases for the system, the following actors for the system have been identified:

* applicant
* Employee (could be lecturers)
* Department officer
* Human Resource Admin



**Figure2.1** use case diagram

### 2.4.2 Use case description

**Use case Name: Login**

**UC\_ID: UC\_01**

**Actor: users** (human resource administrator, department, employee)

**Description:** this use case is used to ensure security for login into the system

**Precondition:** the user must have at least correct username and password.

**Post condition**: the main page will be displayed then user gets access to its privilege and after finishing his/her work he can logout

**Flows of event (main course of action):**

**Table 2.2**Main Course of Action about Login

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** User has to open the system.  **Step3:** by selects account type user fills his or her username and password.  **Step4:** click login button.  **Step6:** the User gets access the system. | **Step2:** The System displays the login interface and allows the user for the user name and password.  **Step5:**display login successful  **Step7:** displays its main window.  **Step8:** Use case ends. |

**Alternative course of action**

If the user is not authorized

**A.6**the system gives a confirmation message that is wrong username or password

**A.7** the system displays a message to enable the user to try to login by entering correct user name and password.

**Use case Name: employee requisition**

**UC\_ID: UC\_02**

**Actor: department officer**

**Description:** This use case enables to submit employment requisition form for section/Dept. recruitment.

**Precondition:** the user must login to the system**.**

**Post condition**: requisition is saved to the database and successful message should be display for the user

**Table 2.3**Main Course of Action about employee requisition

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** click requisition page from menu  **Step3:** fill all information needed to the vacant position.  **Step 4:**click apply button | **Step2**: display requisition form  **Step5:** display successfully posted message.  **Step7:** finish requisition and the use case end |

**Alternative course of action**

1. **1**If the user doesn’t have access privilege to use the system, user is not authenticated and is denied access to the system.

**Use case Name: post announcement**

**UC\_ID: UC\_03**

**Actor: human resource administrator**

**Description:**After section/department asks the employment requisition the human resource administrator should be posted vacancy announcement to get the worker.

**Precondition:** the user must login to the system and the department should apply for employee.

**Post condition**: Record is successfully added to the database message should be display for the user.

**Table 2.4**Main Course of Action about post announcement

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** click post announcement page from menu  **Step3:** fill all information needed to the vacant position.  **Step 4:**click post button | **Step2**: display post announcement form  **Step5:** display successfully posted message.  **Step7:** finish posting announcement and the use case end |

**Alternative course of action**

1. **1**If the user doesn’t have access privilege to use the system; user is not authenticated and is denied access to the system.

**Name: register applicant online**

**UC\_ID: UC\_04**

**Actor: applicant**

**Description:** this use case is done by the applicant in order to register for applying to the vacant position.

**Preconditions:** he/she must have an internet connection, and must have necessary information based on the announcement requirements.

**Post conditions:** the applicant must be registered successfully.

Exceptional condition: if there is no internet connection there is no registration.

**Table 2.5**Main Course of Action about registering applicant online

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step 1:** the applicant opens the home page of the website.  **Step 3:**Select online applicant registration page.  **Step 5**: fill all his/her information required.  **Step 5**:click apply button | **Step2 :** the system display the home page  **Step4:** the system display the applicant registration form  **Step6**: the system display register successfully message |

**Alternative course of action**

**A6**: The system display register failed message if there is unfilled information in the form.

**Use case Name: placement**

**UC\_ID: UC\_07**

**Actor: human resource administrator**

**Description:** the human resource administrator will record the new employed applicant’s information and place them to their position.

**Precondition:** the human resource must login to record and place the new employed applicant information.

**Post condition**: display registration and placement.

**Table 2.6**Main Course of Action about placement

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** click the placement and registration page from menu  **Step3:** fill the all employee information and place to their position.  **Step 4:**click submit button | **Step2:** display placement and registration page  **Step5:**display save successful message  **Step7:** finish placement and the use case end |

**Name: Manage employee**

**UC\_ID: UC\_08**

**Actor: department**

**Description:** This use case is done by the department when they need update search and delete employee profile information

**Preconditions:** The department must login to the system to manage employee.

**Post conditions:** The employee information will be managed (searched, updated) by department**.**

**About updating employee information:**

T**able 2.7:** Main Course of Action about Update employee

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step 1:** the department click manage employee link  **Step3:** the department enter the id number of the employee to update  **Step 5**: the department fill the new updated information of the employee.  **Step 6**:click submit button | **Step2:** the system display the manage employee page  **Step4**: the system display the update form with the registered employee information  **Step7**: display update successful page |

**Alternate course action:**

**A8**: the system display error message if the employee information doesn’t fill accurately.

**About searching employee information**

**Table 2.8**: Main Course of Action about searching

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step 1:** the department click manage employee link  **Step 3:** the department enters the id number of the employee to search and click search button. | **Step2 :** the system display mange employee page  **Step4:** the system display the employee information  **Step8**: finished searching and stop the use case |

**Alternate course action:**

**A4:** the system displays fill again message to department if the entered id is incorrect.

**About deleting employee information**

**Table 2.9:** Main Course of Action about deleting employee

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step 1:** the department officer click manage employee link  **Step 3:** the department officer enters the id number of the employee to search and click search button.  **Step 5**:the administrator enter delete button | **Step2 :** the system display mange employee page  **Step4:** the system display the employee information  **Step6**: the system display delete successful message and stop the use case |

**Alternate course action:**

**A.4:** the system displays fill again message to department officer if the entered id is incorrect.

**A.6:** the system display deletion failed if the entered data are not available

**Use case Name: apply leave**

**UC\_ID: UC\_09**

**Actor: employee**

**Description:** This use case is done by the employee when they want to leave from the institution.

**Precondition:** the user must login to the system and must have full reason to apply leave.

**Post condition**: The employee application will be saved to the database.

**Table 2.10**Main Course of Action about apply leave

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** click leave application page from menu  **Step3:** fill all information needed to apply leave.  **Step 4:**click apply button | **Step2**: display leave application form  **Step5:** display successfully applied message.  **Step7:** finish applying and the use case end |

**Use case Name: send employee leave**

**UC\_ID: UC\_10**

**Actor: department**

**Description:** This use case is done by the department to apply employee leave application to the human resource admin.

**Precondition:** the user must login to the system and must accept employee leave accept.

**Post condition**: The department application will be saved to the database.

**Table 2.11**Main Course of Action about send employee leave application

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** click leave application page from menu  **Step3:** fill all information needed to apply employee leave.  **Step 4:**click apply button | **Step2**: display leave application form  **Step5:** display successfully applied message.  **Step7:** finish applying and the use case end |

**Alternative course of action**

**A.1**If the user doesn’t have access privilege to use the system; user is not authenticated and is denied access to the system.

**Use case Name: approve leave application**

**UC\_ID: UC\_11**

**Actor: human resource administrator**

**Description:** after the department applies the leave application of employee to the human resource admin the human resource admin will view and approve their application.

**Precondition:** the user must login to the system

**Post condition**: application isviewed and display successfully approved message to the user.

**Table 2.12**Main Course of Action about approve leave application

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** click view leave application page from menu  **Step3:**read the reason for applying leave  **Step4:**check leave status  **Step5:** approve the leave application. | **Step2**: display leave application page  **Step6:** finish approving and the use case end |

**Alternative course of action**

**A.1** If the user doesn’t have access privilege to use the system, user is not authenticated and is denied access to the system.

**Use case Name: notification**

**UC\_ID: UC\_12**

**Actor: human resource administrator**

**Description:** human resource admin after viewing leave application he/she notify the result for the leave application to the employee.

**Precondition:** the actor must login to notify the result and the employee must have an email address

**Post condition**: The applicant employee must be notified

**Table 2.13**Main Course of Action about notification

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** click notification page from menu  **Step3:** fill the notification  **Step 4:**click send button | **Step2:** display notification page **Step5:**display sent successful message  **Step6:** finish notification and the use case end |

**Use case Name: generate report**

**UC\_ID: UC\_13**

**Actor: human resource administrator**

**Description:** human resource admin will generate report to the necessary activities

**Precondition:** the user must login to the system

**Post condition**: application is viewed and display successfully approved message to the user.

**Table 2.14**Main Course of Action about report

|  |  |
| --- | --- |
| **Actor action** | **System response** |
| **Step1:** User clicks the generate report menu.  **Step3:**The user fill the form  **Step4:** The user clicks on report type menu item | **Step2**: The system display report form  **Step6:** The system displays successful message and the use case end |

## 2.5. Dynamic models

### 2.5.1. Sequence diagram

A Sequence diagram is an [interaction diagram](http://en.wikipedia.org/wiki/Interaction_diagram) that shows how processes operate with one another and in what order. It is a construct of a [Message Sequence Chart](http://en.wikipedia.org/wiki/Message_Sequence_Chart). A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.



**Figure 2.2** sequence diagram for login



**Figure 2.3** sequence diagram for employee requisition



**Figure 2.4** sequence diagram for post announcement

**Figure 2.5** sequence diagram for view and register



**Figure 2.6**sequence diagram for placement



**Figure 2.7** sequence diagram for apply leave

### 2.5.2. Collaboration diagram

A Collaboration diagram is easily represented by modelling objects in a system and representing the associations between the objects as links. The interaction between the objects is denoted by arrows. To identify the sequence of invocation of these objects, a number is placed next to each of these arrows.



**Figure 2.8**collaboration diagram for login



**Figure 2.9**collaboration diagram for employee requisition



**Figure 2.10**collaboration diagram for post announcement



**Figure 2.11**collaboration diagram for view and register



**Figure 2.12**collaboration diagram for placement



**Figure 2.13**collaboration diagram for apply leave

### 2.5.3. Activity diagram

Activity diagrams are graphical representations of [workflows](http://en.wikipedia.org/wiki/Workflow) of stepwise activities and actions with support for choice, iteration and concurrency. In the [Unified Modelling Language](http://en.wikipedia.org/wiki/Unified_Modeling_Language), activity diagrams are intended to model both computational and organisational processes (i.e. workflows). Activity diagrams show the overall flow of control.



**Figure 2.14**activity diagrams for login



**Figure 2.15**activity diagrams for employee requisition



**Figure 2.16**activity diagram for post announcement



**Figure 2.17**activity diagrams for read and apply



**Figure 2.18**activity diagrams for placement



**Figure 2.19** activity diagram for apply leave

## 2.6. Static model

### 2.6.1. State chart diagram

State chart diagram is used to describe the states of different objects in its life cycle. So the emphasis is given on the state changes upon some internal or external events. These states of objects are important to analyze and implement them accurately. State chart diagrams are very important for describing the states. States can be identified as the condition of objects when a particular event occurs. Before drawing a State chart diagram we must have clarified the following points:

* Identify important objects to be analyzed.
* Identify the states.
* Identify the events.



**Figure 2.20**state chartdiagram for login



**Figure 5.21**state chartdiagram for employee requisition



**Figure 2.22**state chart diagram for vacancy announcement



**Figure 5.23**state chart diagram for view and register



**Figure 2.24**state chart diagram for placement



**Figure 2.25** state chart diagram for apply leave

### 2.6.2. Class diagram

The class diagram is the main building block of [object oriented](http://en.wikipedia.org/wiki/Object_oriented) modeling. It is used both for general [conceptual modeling](http://en.wikipedia.org/wiki/Conceptual_model) of the systematic of the application, and for detailed modeling translating the models into [programming code](http://en.wikipedia.org/wiki/Programming_code). Class diagrams can also be used for [data modeling](http://en.wikipedia.org/wiki/Data_modeling).The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed.

In the diagram, classes are represented with boxes which contain three parts:

* The top part contains the name of the class. It is printed in bold and centered, and the first letter is capitalized.
* The middle part contains the attributes of the class. They are left-aligned and the first letter is lowercase.
* The bottom part contains the methods the class can execute. They are also left-aligned and the first letter is lowercase.

##### **Figure 2.26**analysis level of class diagram

# Chapter three

## Design deliverables of the new system

### 3.1 Introduction

System design is the transformation of the analysis model into a system design model. System design is the first part to get into the solution domain in a software development. This chapter focuses on transforming the analysis model into the design model that takes into account the non-functional requirements and constraints described in the problem of the statement and requirement analysis sections discussed earlier.

The purpose of designing is to show the direction how the system is built and to obtain clear and enough information needed to drive the actual implementation of the system. It is based on understanding of the model the software built on. The objectives of design are to model the system with high quality. Implementing of high quality system depend on the nature of design created by the designer. If one wants to change to the system after it has been put in to operation depends on the quality of the system design. So if the system is design clearly, it will be easy to make changes to it.

### 3.2 Design goals and objectives

The objectives of design are to model the system with high quality. The design goals are derived from non-functional requirements that means non-functional requirement is the description of the feature characteristics and attribute of the system as well as any constraints that may limit the boundary of the proposed solution.

Design goals describe the qualities of the system that the developers should consider.

* **Security:** the system should be secured, i.e., not allow other users or unauthorized users to access data that has no the right to access it.
* **Modifiability:** the system should be modifiable for further modification and enhancement of the application.
* **Performance:** DBU HRMS**shouldrespond fast with high throughput, i.e. Itshould perform retrieving employee file, registering new applicants, updating employee files and generating report as quickly as possible.**
* **Fault Tolerance:** DBU HRM system should be fault tolerant to loss of connectivity with the service.
* **End User Criteria:** - Thesystem should have simple and understandable graphical user Interface such as forms and buttons, which have descriptive names. It should give reliable response for each request at least before the session expires.
* **Reliability**: **it should be able to carry on invalid user inputs, fault tolerant, reliable and available.**
* **Cost:The system should be developed, deployed, administered and maintained with minimum cost as possible.**

### 3.3. System architecture

In this project the team uses a three tier architecture which has three layers. These three layers are the Application or Presentation layer, the business layer and the data access layer. Application or presentation layer is the form which provides the user interface to either programmer or end user. The business layer is the class which the team uses to write the function which works as a mediator to transfer data from application layer or presentation layer to data layer. This layer also has a property layer which is a class where variables are declared corresponding to the fields of the database which can be required for the application and make the properties so that the team can get or set the data using these properties into the variables. The third tire is the data access layer which is also a class to get or set data to the database queries back and forth. This layer only interacts with the database. The database queries or stored procedures will be written here to access the data from the database or to perform any operation to the database.

**Process** Application, Controller)

(Application, Controller)

**Domain** (Business)

(Business)

**Interface** (User interface, System Interface)

**()**

(User interface, System Interface)

**Persistence** (Data)

(Data)

**System**

(Infrastructure, Platform)

Data sources

**Figure3.1**.system architecture

### 3.4system decomposition

The system is divided in to modules. There are four modules in this system with their function. Those modules are

* **Registration module**
* Register applicant
* **Placement module**
* Add new employee
* **Login page**
  + Password
  + User name
  + User Type
* Only HR admin, department and employee have their own page on the DBU human resource management system page. They will perform those activity listed on each page.
* **Notification module**
* Posting notification information through
  + Email
* **Manage account**
* This module helps the HR admin to add new user accounts, update and inactive existed account.
* **Employee management**
* This module helps the department officer to view, identify and organize (manage) all employees under his department.
* **recruitments**
* This module helps HR admin to post vacancy announcement. This module also helps applicants to view posted information’s.
* **Generate report**
* This module helps the HR admin to prepare monthly employee reports to the finance and also the department prepare employee information report.
* **Leave management**
* This module helps the employee to apply leave to the department to approve the employee leave application.



**Figure 3.2**system decomposition

### 

### 3.5. Component modeling

By this Diagram, components of the system will be wired showing that there is relation among components; management of the system, database and operations performed on databases such security issue. This in some extent shows which component or objects will be accessed by whom and what type of security infrastructures it is using.

**Figure3.3.** Component diagram

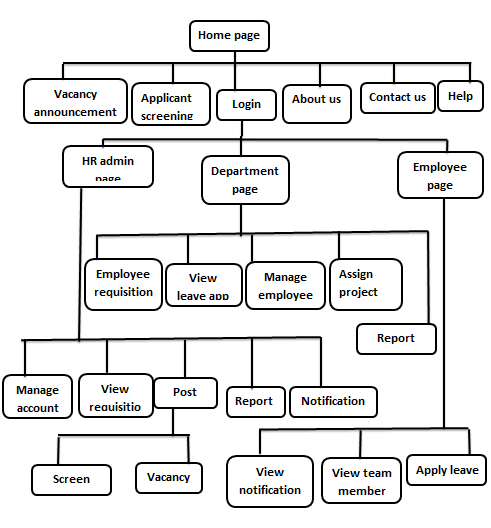
### 3.6. Deployment diagram

Deployment diagrams model the physical architecture of a system, and it shows the relationships between the software and hardware components in the system and the physical distribution of the processing.



**Figure3.4**.Deployment diagram

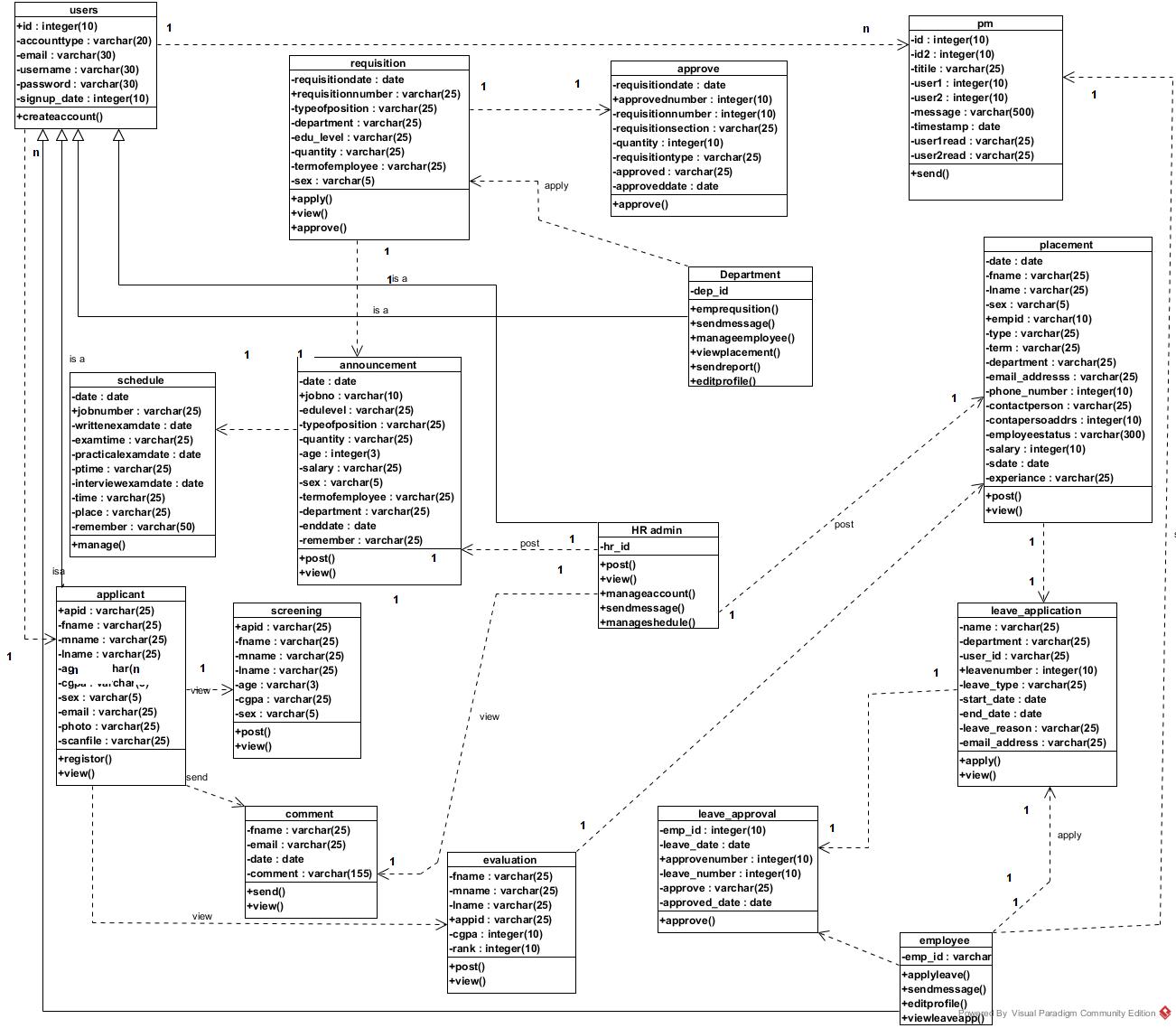
### 3.7 user interface prototyping design



**Figure3.5.** User interface prototype design

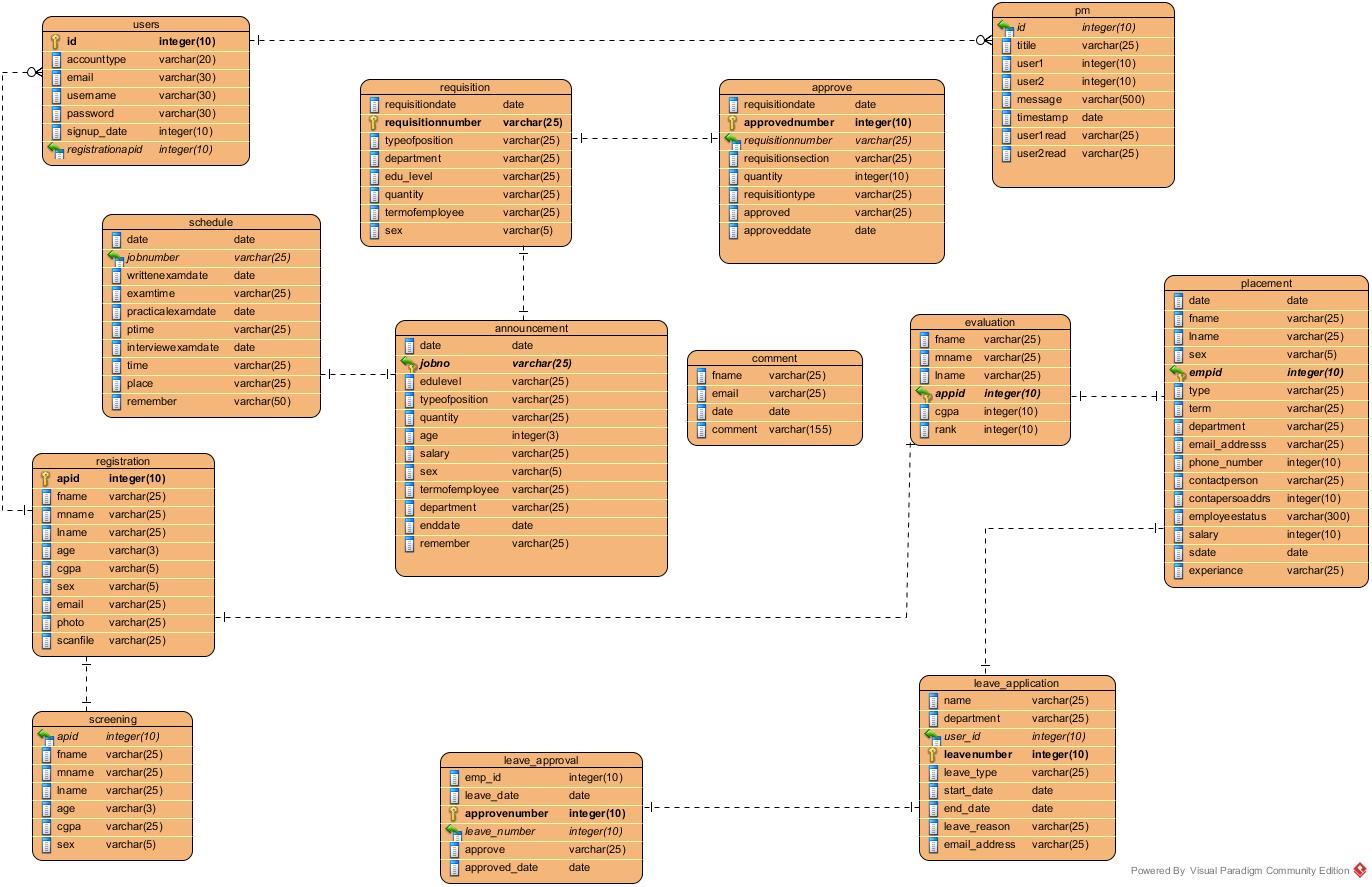
### 3.8. Design of class diagram (class mapping)

The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The classes diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages. The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as a *structural diagram*.

**Figure3.6.** Design of class diagram (class mapping)

### 3.9. Data base design

Database design is the process of producing a detailed [data model](http://en.wikipedia.org/wiki/Data_model) of a [database](http://en.wikipedia.org/wiki/Database). This [logical data model](http://en.wikipedia.org/wiki/Logical_data_model) contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a [Data Definition Language](http://en.wikipedia.org/wiki/Data_Definition_Language), which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity. The term database design can be used to describe many different parts of the design of an overall [database system](http://en.wikipedia.org/wiki/Database_system). Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data.

  
**Figure 3.7.**data base design diagram

# Chapter 4

# System implementation

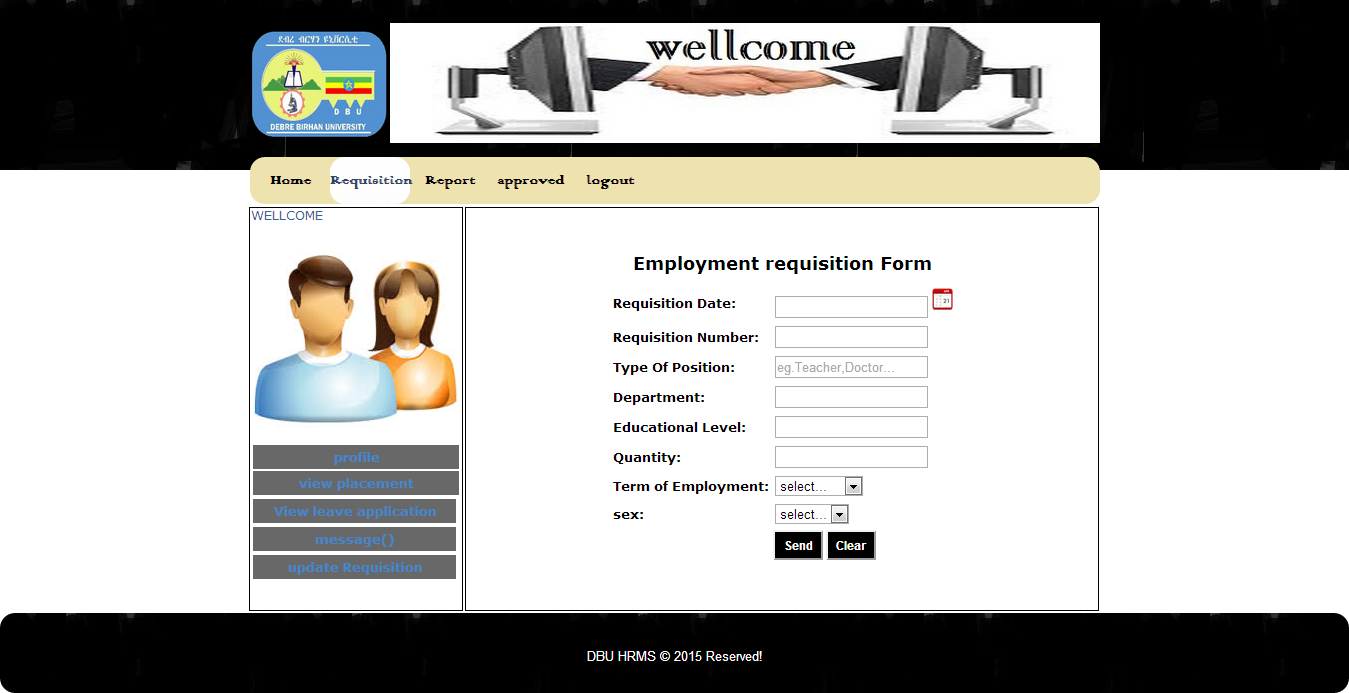
## 4.1. User Interface

In this system, users will interact with the system through the following user interfaces.

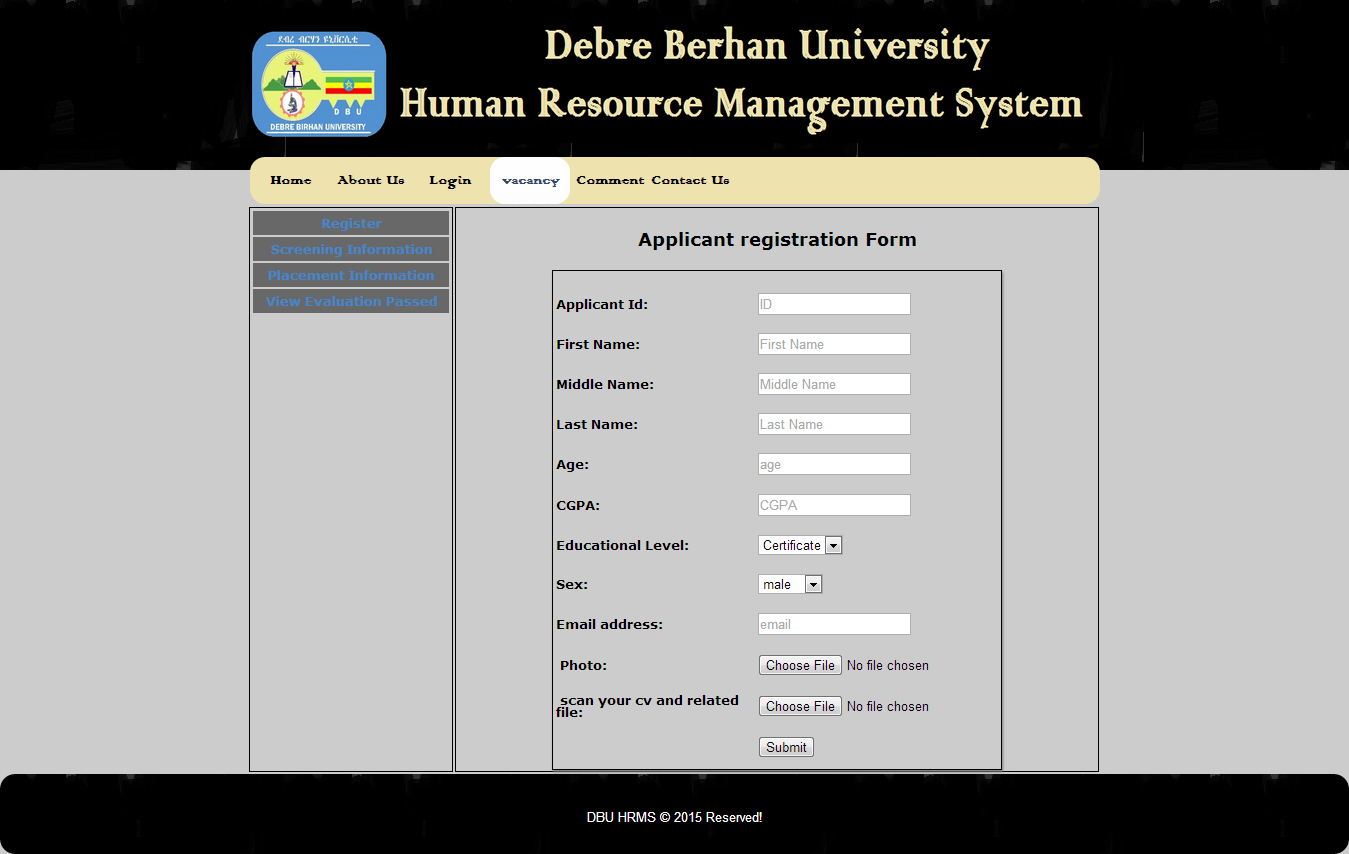
**Home Page**: This form appears on the site in which the system deployed is opened and contains some links which lead the user to other page according to his/her privilege, and if the user is authorized user or has an account, he/she will directly go to the page that he/she wants by entering correct category, User ID and password.

## C:\Users\admin\Desktop\home.png

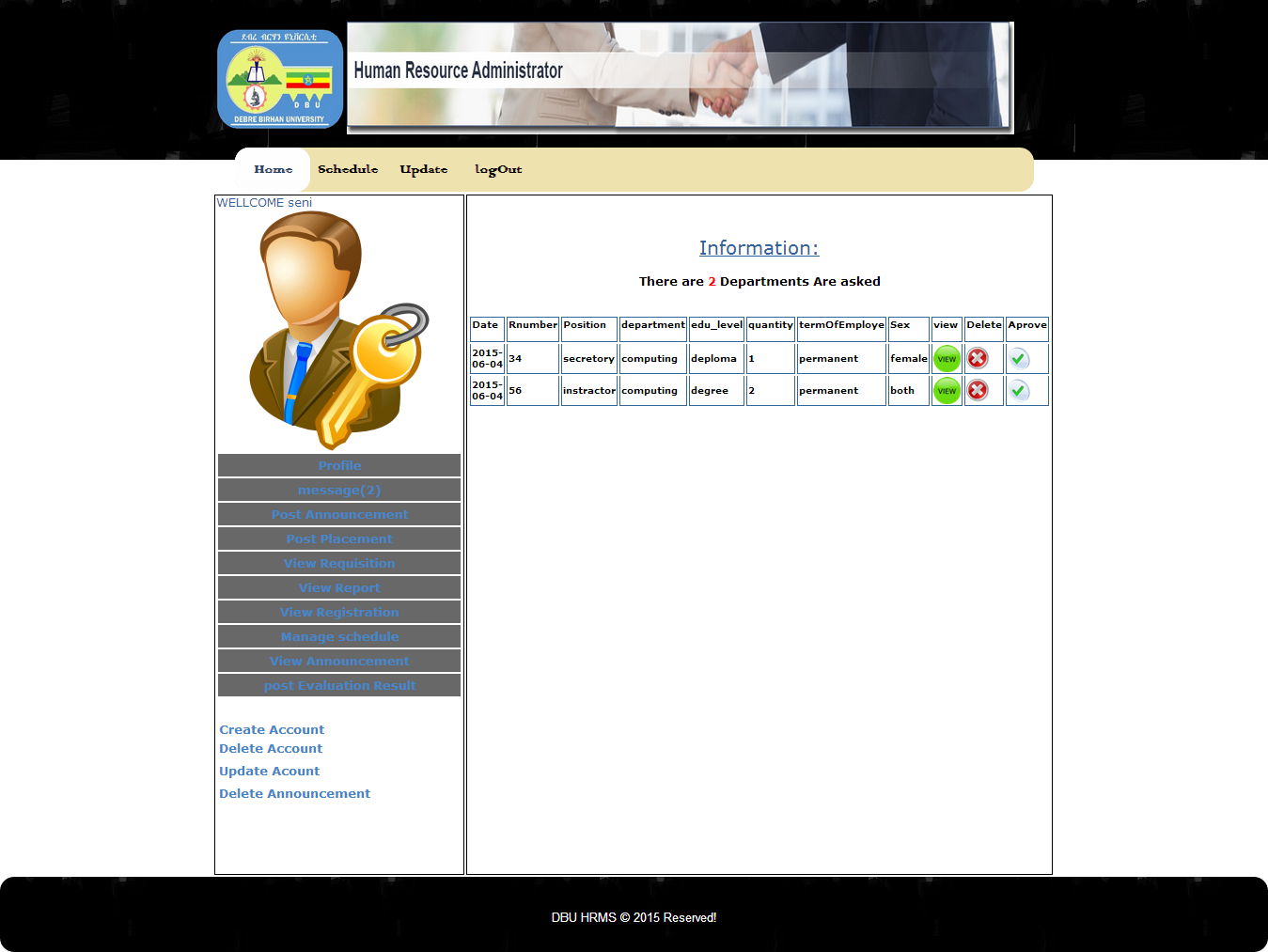
**Fig 4.1: User Interface Design for Login**



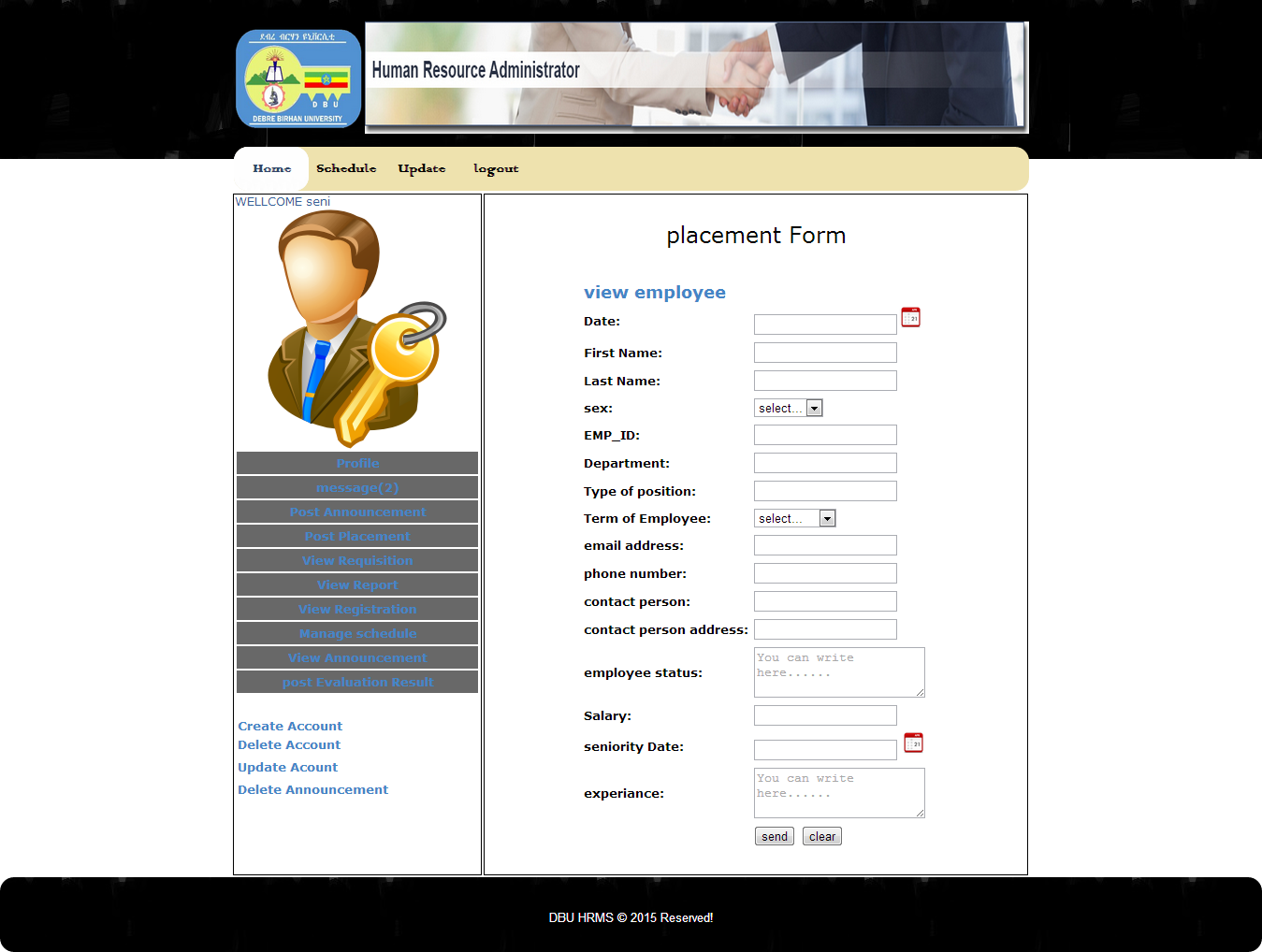
**Fig 4.2: User Interface Design for employee requisition**



**Fig 4.3: User Interface Design for apply**



**Fig 4.4: User Interface Design for view applicant registration**



**Fig 4.5: User Interface Design for employee placement**



**Fig 4.6: User Interface Design for leave application**



**Fig 4.7: User Interface Design for message**

# Chapter five

# 5.1. PROTOTYPE DEVELOPMENT

Prototype development can be defined it is the sample code of the given project. These are some of the sample codes that we have done it.

**//Sample for login**

<?php

error\_reporting(0);

include('config.php')

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>DBU HRM Login</title>

<?php

if(isset($\_SESSION['username']))

{

unset($\_SESSION['username'], $\_SESSION['userid']);

header("location:login.php");

?>

<?php

}

else

{

$ousername = '';

if(isset($\_POST['username'], $\_POST['password']))

{

if(get\_magic\_quotes\_gpc())

{

$ousername = stripslashes($\_POST['username']);

$username = mysql\_real\_escape\_string(stripslashes($\_POST['username']));

$password = stripslashes($\_POST['password']);

}

else

{

$username = mysql\_real\_escape\_string($\_POST['username']);

$password = $\_POST['password'];

}

$req = mysql\_query('select password,accounttype,id from users where username="'.$username.'"');

$dn = mysql\_fetch\_array($req);

if($dn['password']==$password and $dn['accounttype']=='admin' and mysql\_num\_rows($req)>0)

{

$form = false;

$\_SESSION['username'] = $\_POST['username'];

$\_SESSION['userid'] = $dn['id'];

header('location:admin.php');

}

else if($dn['password']==$password and $dn['accounttype']=='department' and mysql\_num\_rows($req)>0)

{

$form = false;

$\_SESSION['username'] = $\_POST['username'];

$\_SESSION['userid'] = $dn['id'];

header('location:department.php');

}

else if($dn['password']==$password and $dn['accounttype']=='employee' and mysql\_num\_rows($req)>0)

{

$form = false;

$\_SESSION['username'] = $\_POST['username'];

$\_SESSION['userid'] = $dn['id'];

header('location:employee.php');

}

else

{

$form = true;

$message = 'The username or password is incorrect.';

}

}

else

{

$form = true;

}

if($form)

{

if(isset($message))

{

echo $message;

}

?>

?>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="style.css" rel="stylesheet" type="text/css" media="screen" />

<link href="menu.css" rel="stylesheet" type="text/css" media="screen" />

<link href="themes/8/js-image-slider.css" rel="stylesheet" type="text/css" />

<script src="themes/8/js-image-slider.js" type="text/javascript"></script>

</head>

<body>

<table align="center">

<!--Header-->

<tr>

<td width="850px" colspan="3" height="150px">

<p><a href="index.php"><imgsrc="image/logo.png" align="left" width="140px" height="120px"></a>

</p>

<p><imgsrc="image/dbuhrm.png" align="left" width="710px" height="120px">

</p>

</td>

</tr><!--End of Header-->

<!--Main menus-->

<tr>

<td colspan="3">

<div id="Menus">

<ul>

<li><a href="index.php">Home</a></li>

<li><a href="about.php">About Us</a></li>

<li class="active"><a href="login.php">Login</a></li>

<li><a href="vacancy.php">vacancy</a></li>

<li><a href="comment.php">Comment</a></li>

<li><a href="contact.php">Contact Us</a></li>

</ul>

</div>

</td>

</tr>

<!--End main menus-->

<!--Slide show-->

<tr>

<td colspan="3">

<div id="sliderFrame">

<div id="slider">

<imgsrc="images/46.jpg" />

<imgsrc="images/47.jpg" />

<imgsrc="images/stud.jpg" />

<imgsrc="images/36.jpg" />

<imgsrc="images/hum.jpg" />

</div>

</div>

</td>

</tr>

<!--End of slide show-->

<tr>

<td width="200px" height="400px" valign="top" id="insides">

<table>

<tr>

<thbgcolor="686868" width="200px" height="20px" align="center"><a href="memeber.php"><strong><font size ="2.5">Register</font></strong></a></th>

</tr>

<tr>

<thbgcolor="686868" width="200px" height="20px" align="center"><a href="Screening.php"><strong><font size="2">Screening Information</font></strong></a></th>

</tr>

<tr>

<thbgcolor="686868" width="200px" height="20px" align="center"><a href="evaluationresult.php"><strong><font size="2">Placement Information</font></strong></a></th>

</tr>

<tr>

<thbgcolor="686868" width="200px" height="20px" align="center"><a href="evaluationresult.php"><strong><font size="2">View Evaluation Passed</font></strong></a></th>

</tr>

</table>

</td>

<td valign="top" id="insides">

<br><br>

<br><br>

<form action="login.php" method="post" onsubmit='return formValidation();'>

<table style="border:1px solid #000000;box-shadow:1px 2px 3px gray;" width="450px" height="350px" >

<tr>

<thbgcolor="#686868" colspan="2"<td align="right"><center><font color="black" size="4">login form</font></center><a href="index.php"><imgsrc="image/close\_icon.gif" title="Close"></a></td></tr></th>

</tr>

<tr>

<center>

<td>

<p align="center"><imgsrc="image/key\_lock-animated.gif" title=" you must Login" width="80"></p>

</td>

</center>

<tr>

<td><font color="black" size="2">&nbsp;Username:</font></td><td><input type="text" name="username" value="<?php echo htmlentities($ousername, ENT\_QUOTES, 'UTF-8'); ?>" /></td>

</tr>

<tr>

<td><font color="black" size="2">&nbsp;Password:</font></td><td><input type="password" name="password" placeholder="Password" id="pass" required x-moz-errormessage="Enter Your Passord" ></td>

</tr>

<tr>

<td>&nbsp;</td>

<td><input type="submit" value="Login" name='login'/></td></tr>

<tr>

<td align="center"><a href="forgotp.php">Forget Your Passord?</a></td>

</tr>

</table>

</form>

<!--PHP Login-->

</center>

</td>

</tr>

</table>

<div id="sample"><br><br><font face="arial" color="white" size="2"><p align="center">DBU HRMS &copy; 2015 Reserved!</div>

<?php

}

}

?>

</body>

</html>

## //sample code for manage employee

<?php

error\_reporting(0);

include('config.php')

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Derhan Berhan University HRM...</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="style.css" rel="stylesheet" type="text/css" media="screen" />

<link href="menu.css" rel="stylesheet" type="text/css" media="screen" />

<link href="themes/8/js-image-slider.css" rel="stylesheet" type="text/css" />

<script src="themes/8/js-image-slider.js" type="text/javascript"></script>

<script>

functionisdelete()

{

var d = confirm('Are you sure you want to Delete !!');

if(!d)

{

alert(window.location='adelete.php');

}

else

{

return false;

}

}

</script>

</head>

<body>

<table align="center">

<!--Header-->

<tr>

<td width="850px" colspan="3" height="150px">

<p><a href="index.php"><imgsrc="image/logo.png" align="left" width="140px" height="120px"></a>

<imgsrc="image/section.png" width="710px" height="120px">

</p></td>

</tr>

<!--End of Header-->

<!--Main menus-->

<tr>

<td colspan="3">

<div id="Menus">

<ul>

<li class="active" ><a href="department.php">Home</a></li>

<li><a href="requisition.php">Requisition</a></li>

<li><a href="report.php">Report</a></li>

<li><a href="viewapprove.php">approved</a></li>

<li><a href="signout.php">logout</a></li>

</ul>

</div>

</td>

</tr>

<!--End main menus-->

<!--Slide show-->

<!--End of slide show-->

<tr>

<td width="200px" height="400px" valign="top" id="insides">

<table>

<tr><td><imgsrc="image/dep.jpg"></td><h3>WELLCOME</h><?php

if(isset($\_SESSION['user\_id'])){echo ''.htmlentities($\_SESSION['user\_id'],ENT\_QUOTES,'UTF-8');}

?><tr><br>

<tr>

<thbgcolor="#686868" width="200px" height="20px" align="center"><a href="edit\_infos2.php"><strong><font size="2">profile</font></strong></th>

</tr>

<tr>

<thbgcolor="#686868" width="200px" height="20px" align="center"><a href="placementview.php"><strong><font size="2">view placement</font></strong></th>

</tr>

</table>

<table>

<thbgcolor="#686868" width="200px" height="20px" align="center"><a href="viewleaveapp.php"><strong><font size="2">View leave application</font></strong></th>

</tr>

</table>

<table>

<tr>

<thbgcolor="#686868" width="200px" height="20px" align="center"><a href="list\_pm2.php"><strong><font size="2">message(<?php echo $nb\_new\_pm?>)</font></strong></th>

</tr>

</table>

<table>

<tr>

<thbgcolor="#686868" width="200px" height="20px" align="center"><a href="updaterequ.php"><strong><font size="2">update Requisition</font></strong></th>

</tr>

</table>

<br><br><br>

</td>

<td valign="top" id="insides">

<br><br>

<?php

$ctrl = $\_REQUEST['key'];

$query="SELECT \* FROM placement where empid='{$ctrl}'";

$result=mysql\_query($query);

$count=mysql\_num\_rows($result);

if(!$result){

die("user not registered!".mysql\_error());

}

if($count==1){

while($row=mysql\_fetch\_array($result)){

$r0=$row[0];

$r1=$row[1];

$r2=$row[2];

$r3=$row[3];

$r4=$row[4];

$r5=$row[5];

$r6=$row[6];

$r7=$row[7];

$r8=$row[8];

$r9=$row[9];

$r10=$row[10];

$r11=$row[11];

$r12=$row[12];

$r13=$row[13];

$r14=$row[14];

$r15=$row[15];

}

?>

<form id="form1" method="POST" action="updateemp.php" onsubmit='return formValidation()'>

<table>

<tr>

<tr><td>first Name:</td><td><input type='text' name='name' value="<?php echo "$r1"?>"></td></tr>

<tr>

<tr><td>last Name:</td><td><input type='text' name='lname' value="<?php echo "$r2"?>"></td></tr>

<tr>

<td><font color="black" size="2">sex:</font></td>

<td><select id="sex" name="sex"value="<?php echo "$r3"?>" >

<option>select...</option>

<option>female</option>

<option>male</option>

</select>

</td></tr>

<tr>

<tr><td>Emp\_id:</td><td><input type='text' name='empid' value="<?php echo "$r4"?>"></td></tr>

<tr>

<tr><td>Type of position:</td><td><input type='text' name='type' value="<?php echo "$r5"?>"></td></tr>

<tr>

<td><font color="black" size="2" value="<?php echo "$r6"?>">Term of Employee:</font></td>

<td><select id="term" name="term">

<option>select...</option>

<option>permanent</option>

<option>temporary</option>

<option>contract</option>

</select>

</td>

</tr>

<tr><td>department:</td><td><input type='text' name='department' value="<?php echo "$r7"?>"></td></tr>

<tr><td>email address:</td><td ><input type='text' name='email' value="<?php echo "$r8"?>"></td></tr>

<tr>

<tr><td>phone number</td><td><input type='text' name='number' value="<?php echo "$r9"?>"></td></tr>

<tr>

<tr><td>contact person</td><td><input type='text' name='contact' value="<?php echo "$r10"?>"></td></tr>

<tr><td>contact person address :</td><td><input type='text' name='address' value="<?php echo "$r11"?>"></tr></td>

<tr><td>employee status:</td><td><input type='text' name='status' value="<?php echo "$r12"?>"></tr></td>

<tr>

<tr><td>salary</td><td><input type='text' name='salary' value="<?php echo "$r13"?>"></td></tr>

<tr>

<tr><td>seniority date</td><td><input type='text' name='date' value="<?php echo "$r14"?>"></td></tr>

<tr>

<tr><td>experiance</td><td><input type='text' name='expe' value="<?php echo "$r15"?>"></td></tr>

<tr><td colspan=2 align='center'><input type='submit' name='update' value='Save Changes' class="button\_example"></tr></td>

</table>

<?php

}

?>

<?php

if(isset($\_POST['update']))

{

$name1=$\_POST['name'];

$name2=$\_POST['lname'];

$sex=$\_POST['sex'];

$empid=$\_POST['empid'];

$type=$\_POST['type'];

$term=$\_POST['term'];

$department=$\_POST['department'];

$email=$\_POST['email'];

$phone=$\_POST['number'];

$contact=$\_POST['contact'];

$addresss=$\_POST['address'];

$status=$\_POST['status'];

$salary=$\_POST['salary'];

$date=$\_POST['date'];

$expe=$\_POST['expe'];

$update = mysql\_query("update placement set fname='$name',

lname='$name1',

sex='$sex',

empid='$empid',

type='$type',

term='$term',

department='$department',

email\_address='$email',

phone\_number='$phone',

contactperson='$contact',

contapersoaddres='$address',

employeestatus='$status',

salarys='$salary',

sdate='$date',

experiance='$expe',

WHERE empid='{$empid}'") or die(mysql\_error());

echo "<script>window.location='placementview.php';</script>";

}

?>

</form>

</div>

</td>

</tr>

</table>

<div id="sample"><br><br><font face="arial" color="white" size="2"><p align="center">DBU HRMS &copy; 2015 Reserved!</div>

</body>

</html>

//sample code for send message

<?php

error\_reporting(0);

include('config.php')

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>DBU HRM ...</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="style.css" rel="stylesheet" type="text/css" media="screen" />

<link href="menu.css" rel="stylesheet" type="text/css" media="screen" />

</head>

<body>

<table align="center">

<!--Header-->

<tr>

<td width="850px" colspan="3" height="150px">

<p><a href="employee.php"><imgsrc="image/logo.png" align="left" width="140px" height="120px"></a>

</p>

<p><imgsrc="image/emp.png" align="left" width="710px" height="120px">

</p>

</td>

</tr>

<!--End of Header-->

<!--Main menus-->

<tr>

<td colspan="3">

<div id="Menus">

<ul>

<li class="active"><a href="employee.php">Home</a></li>

<li><a href="applyleav.php">leave</a></li>

<li><a href="viewleaveapprove.php">approved</a></li>

<li><a href="signout.php">logout</a></li>

</ul>

</div>

</td>

</tr>

<!--End main menus-->

<!--Slide show-->

<!--End of slide show-->

<tr>

<td width="200px" height="400px" valign="top" id="insides">

<table>

<tr><td><imgsrc="image/p.png"></td><td><h3>WELCOME</h3><?php if(isset($\_SESSION['username'])){echo ' '.htmlentities($\_SESSION['username'], ENT\_QUOTES, 'UTF-8');} ?></td></tr>

<?php

if(isset($\_SESSION['username']))

{

$nb\_new\_pm = mysql\_fetch\_array(mysql\_query('select count(\*) as nb\_new\_pm from pm where ((user1="'.$\_SESSION['userid'].'" and user1read="no") or (user2="'.$\_SESSION['userid'].'" and user2read="no")) and id2="1"'));

$nb\_new\_pm = $nb\_new\_pm['nb\_new\_pm'];

?>

<tr>

<thbgcolor="#686868" width="200px" height="20px" align="center"><a href="edit\_infos.php"><strong><font size ="2">profile</font></strong></a></th>

</tr>

<tr>

<thbgcolor="#686868" width="200px" height="20px" align="center"><a href="list\_pm.php"><strong><font size ="2">message(<?php echo $nb\_new\_pm?>)</font></strong></a></th>

</tr>

</table>

<br><br>

</td>

<td valign="top" id="insides">

<br><br>

<table><tr><td>

<td width="350" valign="top" height="700">

<?php

//We check if the user is logged

if(isset($\_SESSION['username']))

{

//We list his messages in a table

//Two queries are executes, one for the unread messages and another for read messages

$req1 = mysql\_query('select m1.id, m1.title, m1.timestamp, count(m2.id) as reps, users.id as userid, users.username from pm as m1, pm as m2,users where ((m1.user1="'.$\_SESSION['userid'].'" and m1.user1read="no" and users.id=m1.user2) or (m1.user2="'.$\_SESSION['userid'].'" and m1.user2read="no" and users.id=m1.user1)) and m1.id2="1" and m2.id=m1.id group by m1.id order by m1.id desc');

$req2 = mysql\_query('select m1.id, m1.title, m1.timestamp, count(m2.id) as reps, users.id as userid, users.username from pm as m1, pm as m2,users where ((m1.user1="'.$\_SESSION['userid'].'" and m1.user1read="yes" and users.id=m1.user2) or (m1.user2="'.$\_SESSION['userid'].'" and m1.user2read="yes" and users.id=m1.user1)) and m1.id2="1" and m2.id=m1.id group by m1.id order by m1.id desc');

?>

:<br />

<a href="new\_pm.php" >CREATE NEW REPORT</a><br />

<h3>Unread REPORTS(<?php echo intval(mysql\_num\_rows($req1)); ?>):</h3>

<table>

<tr>

<th>Title</th>

<th>Nb. Replies</th>

<th>Participant</th>

<th>Date of creation</th>

</tr>

<?php

//We display the list of unread messages

while($dn1 = mysql\_fetch\_array($req1))

{

?>

<tr>

<td ><a href="read\_pm.php?id=<?php echo $dn1['id']; ?>"><?php echo htmlentities($dn1['title'], ENT\_QUOTES, 'UTF-8'); ?></a></td>

<td><?php echo $dn1['reps']-1; ?></td>

<td><a href="profile.php?id=<?php echo $dn1['userid']; ?>"><?php echo htmlentities($dn1['username'], ENT\_QUOTES, 'UTF-8'); ?></a></td>

<td><?php echo date('Y/m/d H:i:s' ,$dn1['timestamp']); ?></td>

</tr>

<?php

}

//If there is no unread message we notice it

if(intval(mysql\_num\_rows($req1))==0)

{

?>

<tr>

<td colspan="4">You have no unread report.</td>

</tr>

<?php

}

?>

</table>

<br />

<h3>Read reports(<?php echo intval(mysql\_num\_rows($req2)); ?>):</h3>

<table>

<tr>

<th>Title</th>

<th>Nb. Replies</th>

<th>Participant</th>

<th>Date or creation</th>

</tr>

<?php

//We display the list of read messages

while($dn2 = mysql\_fetch\_array($req2))

{

?>

<tr>

<td ><a href="read\_pm.php?id=<?php echo $dn2['id']; ?>"><?php echo htmlentities($dn2['title'], ENT\_QUOTES, 'UTF-8'); ?></a></td>

<td><?php echo $dn2['reps']-1; ?></td>

<td><a href="info.php?id=<?php echo $dn2['userid']; ?>"><?php echo htmlentities($dn2['username'], ENT\_QUOTES, 'UTF-8'); ?></a></td>

<td><?php echo date('Y/m/d H:i:s' ,$dn2['timestamp']); ?></td>

</tr>

<?php

}

//If there is no read message we notice it

if(intval(mysql\_num\_rows($req2))==0)

{

?>

<tr>

<td colspan="4" >You have no read report.</td>

</tr>

<?php

}

?>

</table>

<?php

}

else

{

echo 'You must be logged to access this page.';

}

?>

<a href="<?phpecho $url\_home; ?>">Go Home</a>

<?php

}

?>

</td></tr></table>

</td>

</tr>

</table>

<div id="sample"><br><br><font face="arial" color="white" size="2"><p align="center">DBU HRMS &copy; 2015 Reserved!</div>

</body>

</html>

**Chapter six**

### 1. Conclusions

Human resource Management System allows DBU human resource office and department to store employee’s detail information properly. This project is also able to provide reports about employee detail information from the department to human resource office and from the human resource office to the finance office. And alsoHR admin is able to postvacancy announcement and screen and view posted announcement to the user. The implementation of the system in the organization has considerably reduce manual data entry, time and also provide readily calculated reports.

### 2. Recommendation

The system that we have developed involves web based human resource management system for Debre Berhan university that means it’s a huge system so it is very difficult to include all functionality of the HRM office so that we only concerned on the recruitments and placement, employee profile management and employee leave managementsubsystems because of limited development capacity and time. Therefore, we recommend the following features need to be included in any further revision and extension attempt.

* Include online applicant examination.
* Include Payroll and Benefits

**Therefore, others who are interested to develop a new system on human resource management system theycan get some initial idea about the system. By focusing on the limitation and functional areas of the system they can also develop a better human resource management system.**

# References

* Thesis on employee profile management by Robert kimbeya ,Zambia
* Ediwin B. Flippo, Personnel Management, 1994.6th edition .McGraw will book company.
* Dianl Arthur. Managing human resource in small and midsized companies. 1986, 2nd edition, amacom.
* Trevor Bolton, An introduction to human Resource Management, 2002 3rd edition.
* William P.ANTHONY, K.MICHELE KACMAR PAMAL. L.PERREWE’ to human Resource management a strategic Approach, 4th edition.