**Chapter 1: Introduction**

* 1. **Overview of the System**

The “Student Information System” has been developed to overcome the problems existing in the manual system. This software is developed to eliminate and reduce the hardships faced by the existing system. Moreover this system is designed for the particular need of the administration to carry out operations in a smooth and effective manner.

This application is developed to speed up the operations as much as possible and to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus, by this it proves that it is user-friendly. Student Information system, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

* 1. **Problem Statement**

The data of students is written on Microsoft excel sheet, taking name, ID, marks each column, attendance on a paper. Time consumption is very huge with lack of security. Any unauthorized person can get access to all the information. There is maximum chance of losing data in the manual system. There is no privacy of student data. We don’t know what action is being performed by whom.

Student’s data can be modified without any permission.

* 1. **Proposed System**

The **Student Information System** is developed to store student details like ID, name, marks obtained by the student during exams, attendance records and everything related to the payment section.

As our system is being developed at administrative end, it will help the administrator to reduce the workload that exists in the manual system and minimizing the probability of losing any data. This system will change the way of viewing the data and help them to do their work in a very less amount of time.

This system will provide user friendly approach and eliminate the hard work faced by the existing (manual) system.

* 1. **Objectives**

The main objective of the project **Student Information System** is to manage the details of the students. This project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the information of the students. It showcases all the details of the student, their courses, address, marks, payment details etc.

**Advantages of using Student Information System**:

* Provides smart management of student’s data.
* It manages the information of all the students.
* Provides the facility of searching student’s information based on their enrollment number.
* It provides editing, adding and updating of records which results in proper resource management of student data.
* More work in less time.
* Accuracy in works.
* Well designed reports.

**Chapter 2: Requirement Analysis**

**2.1. Tools and Technology used**

**Hardware Specifications:**

* Processor : Intel Core i3
* Hard Disk: 1 TB
* RAM: 4 GB DDR3
* RAM Type: DDR4
* Processor Variant: 6006U
* Operating System: Microsoft Windows 10

**Software to be used:**

* Operating System: Microsoft Windows 10
* Front End: Microsoft Visual Studio 2008
* Back End: MySQL Server 2005

**2.2. Feasibility Study**

The objective of feasibility is just not to solve the problem but also to acquire a sense of its scope. The three major areas to consider while determining the feasibility of a project are:

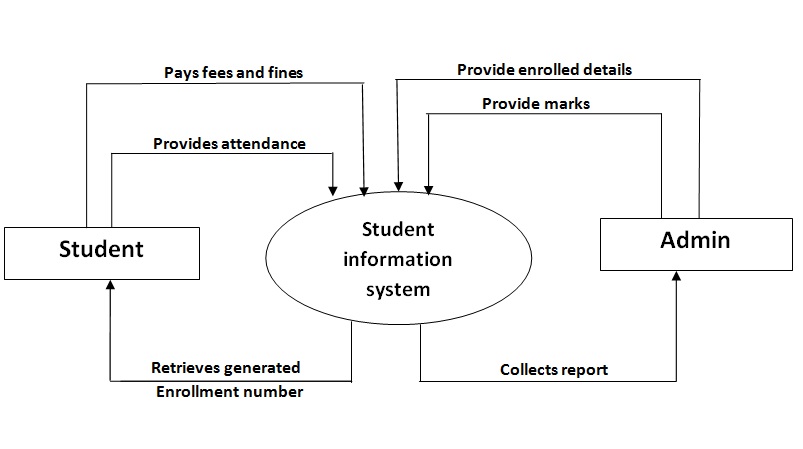
1. Behavioral Feasibility:The system working is quite easy to use and learn due to its simple but attractive interface. User requires no special training for operating the system.
2. Economically Feasibility: The system being developed is economic with respect to University’s point of view. It is cost effective in the sense that has eliminated the paper work completely.

The system is also time effective because the calculations are automated which are made at the end of the month or as per the user requirement.

The result obtained contains minimum errors and are highly accurate as the data is required.

1. Technical Feasibility: The System working is quite easy to use and learn due to its simple but attractive interface. User requires no special training for operating the system.

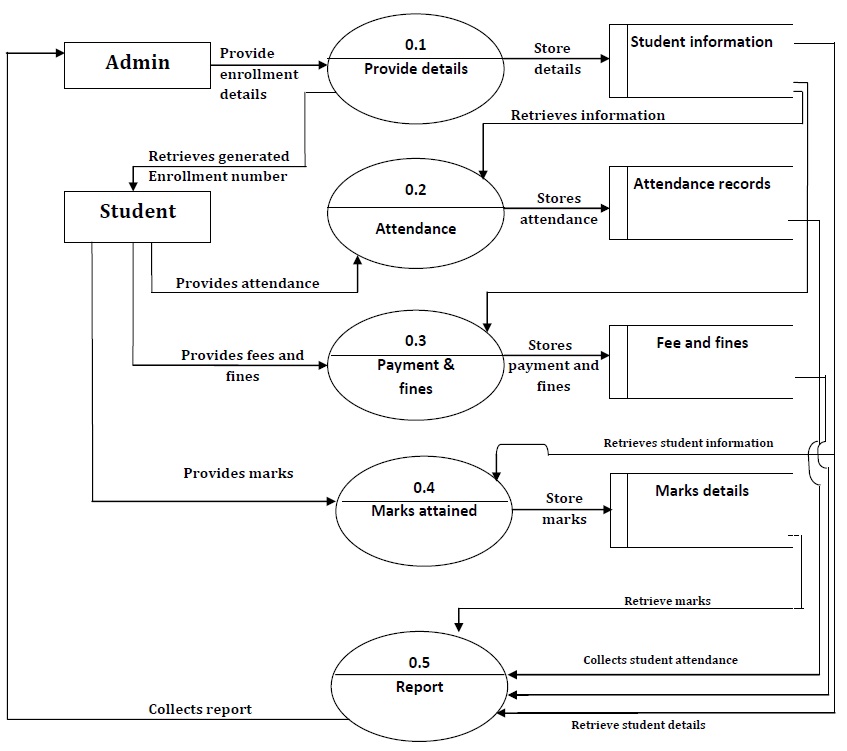
**Chapter 3: System Design**

****

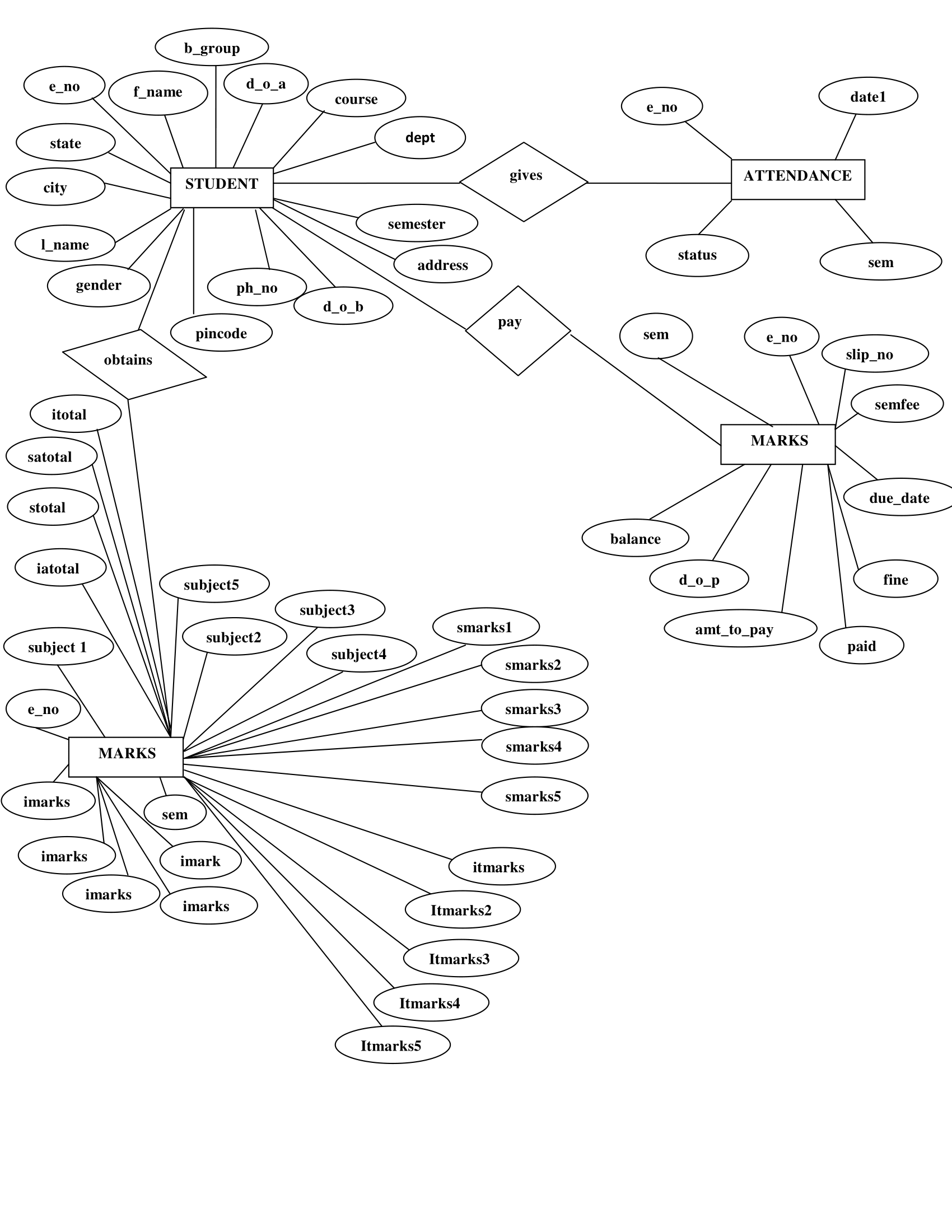
**3.1. Data Flow Diagrams**

**3.1.1. Zero level Data Flow (context) Diagram**

**3.1.2. First level Data Flow Diagram**

****

**3.2. ER Diagram**

****

**3.3. Data Dictionary**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Description** | **Source table** | **Constraint** |
| **address** | **Student's Address** | **Student** |  |
| **amt\_to\_pay** | **Amout to be paid** | **Payment** |  |
| **b\_group** | **Blood group** | **Student** |  |
| **balance** | **Payment balance** | **Payment** |  |
| **city** | **City where student lives** | **Student** |  |
| **course** | **Course taken by student** | **Student** |  |
| **d\_o\_a** | **Date of admission** | **Student** |  |
| **d\_o\_b** | **Date of birth** | **Student** |  |
| **d\_o\_p** | **Date of payment** | **Payment** |  |
| **date1** | **Date of Attendance** | **Attendance** |  |
| **dept** | **Department of the student** | **Student** |  |
| **due\_date** | **Due date** | **Payment** |  |
| **e\_no** | **Enrollment number** | **Student** | **primary key** |
| **f\_name** | **First Name of the student** | **Student** |  |
| **fine** | **Fines** | **Payment** |  |
| **gender** | **Gender of the student** | **Student** |  |
| **iatotal** | **Internal exam total marks achieved** | **Marks** |  |
| **imarks1** | **Internal exam subject 1 marks** | **Marks** |  |
| **imarks2** | **Internal exam subject 2 marks** | **Marks** |  |
| **imarks3** | **Internal exam subject 3 marks** | **Marks** |  |
| **imarks4** | **Internal exam subject 4 marks** | **Marks** |  |
| **imarks5** | **Internal exam subject 5 marks** | **Marks** |  |
| **itmarks1** | **Internal exam subject 1 total marks** | **Marks** |  |
| **itmarks2** | **Internal exam subject 2 total marks** | **Marks** |  |
| **itmarks3** | **Internal exam subject 3 total marks** | **Marks** |  |
| **itmarks4** | **Internal exam subject 4 total marks** | **Marks** |  |
| **itmarks5** | **Internal exam subject 5 total marks** | **Marks** |  |
| **itotal** | **Internal exam all subject total marks** | **Marks** |  |
| **l\_name** | **Last Name of the student** | **Student** |  |
| **paid** | **Payment paid** | **Payment** |  |
| **ph\_no** | **Phone number of the student** | **Student** |  |
| **pincode** | **Pincode** | **Student** |  |
| **satotal** | **Semester exam total marks achieved** | **Marks** |  |
| **sem** | **Semester** | **Attendance** |  |
| **sem\_fee** | **Semester fees** | **Payment** |  |
| **semester** | **Semester** | **Student** |  |
| **slip\_no** | **Slip number** | **Payment** | **primary key** |
| **smarks1** | **Semester exam subject 1 marks** | **Marks** |  |
| **smarks2** | **Semester exam subject 2 marks** | **Marks** |  |
| **smarks3** | **Semester exam subject 3 marks** | **Marks** |  |
| **smarks4** | **Semester exam subject 4 marks** | **Marks** |  |
| **smarks5** | **Semester exam subject 5 marks** | **Marks** |  |
| **state** | **State where student lives** | **Student** |  |
| **status** | **Student present or absent** | **Attendance** |  |
| **stmarks1** | **Semester exam subject 1 total marks** | **Marks** |  |
| **stmarks2** | **Semester exam subject 2 total marks** | **Marks** |  |
| **stmarks3** | **Semester exam subject 3 total marks** | **Marks** |  |
| **stmarks4** | **Semester exam subject 4 total marks** | **Marks** |  |
| **stmarks5** | **Semester exam subject 5 total marks** | **Marks** |  |
| **stotal** | **Semester exam all subject total marks** | **Marks** |  |
| **subject1** | **Name of subject 1** | **Marks** |  |
| **subject2** | **Name of subject 2** | **Marks** |  |
| **subject3** | **Name of subject 3** | **Marks** |  |
| **subject4** | **Name of subject 4** | **Marks** |  |
| **subject5** | **Name of subject 5** | **Marks** |  |

**3.4. Table Structure**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Datatype** | **Size** | **Constraint** |
| **e\_no** | **int** | **11** | **primary key** |
| **f\_name** | **varchar** | **30** |  |
| **l\_name** | **varchar** | **30** |  |
| **gender** | **varchar** | **7** |  |
| **d\_o\_a** | **date** | **8** |  |
| **course** | **varchar** | **70** |  |
| **dept** | **varchar** | **90** |  |
| **semester** | **int** | **11** |  |
| **d\_o\_b** | **date** | **8** |  |
| **address** | **varchar** | **100** |  |
| **ph\_no** | **bigint** | **25** |  |
| **b\_group** | **varchar** | **10** |  |
| **pincode** | **int** | **6** |  |
| **city** | **varchar** | **40** |  |
| **state** | **varchar** | **40** |  |

**Table. 3.4.1 Student**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Datatype** | **Size** | **Constraint** |
| **e\_no** | **int** | **11** | **foreign key** |
| **sem** | **int** | **11** |  |
| **date1** | **date** | **8** |  |
| **status** | **varchar** | **11** |  |

**Table. 3.4.2 Attendance**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Datatype** | **Size** | **Constraint** |
| **e\_no** | **int** | **11** | **foreign key** |
| **sem** | **int** | **2** |  |
| **subject1** | **varchar** | **100** |  |
| **subject2** | **varchar** | **100** |  |
| **subject3** | **varchar** | **100** |  |
| **subject4** | **varchar** | **100** |  |
| **subject5** | **varchar** | **100** |  |
| **imarks1** | **int** | **4** |  |
| **imarks2** | **int** | **4** |  |
| **imarks3** | **int** | **4** |  |
| **imarks4** | **int** | **4** |  |
| **imarks5** | **int** | **4** |  |
| **itmarks1** | **int** | **4** |  |
| **itmarks2** | **int** | **4** |  |
| **itmarks3** | **int** | **4** |  |
| **itmarks4** | **int** | **4** |  |
| **itmarks5** | **int** | **4** |  |
| **itotal** | **int** | **4** |  |
| **iatotal** | **int** | **4** |  |
| **smarks1** | **int** | **4** |  |
| **smarks2** | **int** | **4** |  |
| **smarks3** | **int** | **4** |  |
| **smarks4** | **int** | **4** |  |
| **smarks5** | **int** | **4** |  |
| **stmarks1** | **int** | **4** |  |
| **stmarks2** | **int** | **4** |  |
| **stmarks3** | **int** | **4** |  |
| **stmarks4** | **int** | **4** |  |
| **stmarks5** | **int** | **4** |  |
| **stotal** | **int** | **4** |  |
| **satotal** | **int** | **4** |  |

**Table. 3.4.2 Marks**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Datatype** | **Size** | **Constraint** |
| **slip\_no** | **int** | **11** | **primary key** |
| **e\_no** | **int** | **11** | **foreign key** |
| **sem** | **int** | **11** |  |
| **sem\_fee** | **decimal** | **8,2** |  |
| **due\_date** | **date** | **8** |  |
| **d\_o\_p** | **date** | **8** |  |
| **amt\_to\_pay** | **decimal** | **8,2** |  |
| **paid** | **decimal** | **8,2** |  |
| **fine** | **decimal** | **8,2** |  |
| **balance** | **decimal** | **8,2** |  |

**Table. 3.4.2 Payment**

**Chapter 4: Testing and Implementation**

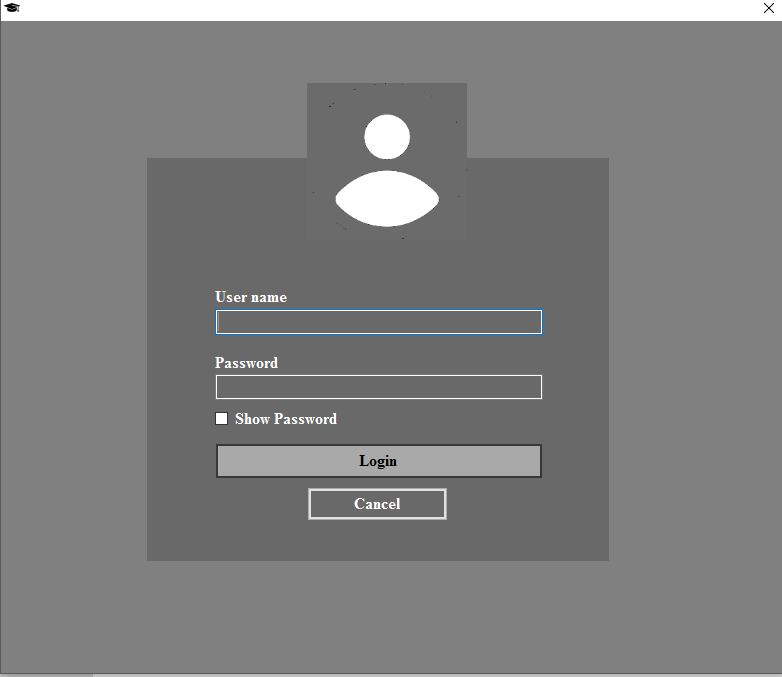
Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is defect free.

Software testing has 3 different stages of the software development lifecycle where testing is conducted. The 3 different stages are

1. Unit testing: A level of the software testing process where individual units of software are tested. The purpose is to validate that each unit of the software performs as designed.
2. Integration Testing: A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units.
3. System Testing: A level of the software testing process where a complete, integrated system is tested. The purpose of this test is to evaluate the system’s compliance with the specified requirements.

**4.1. Unit Testing**

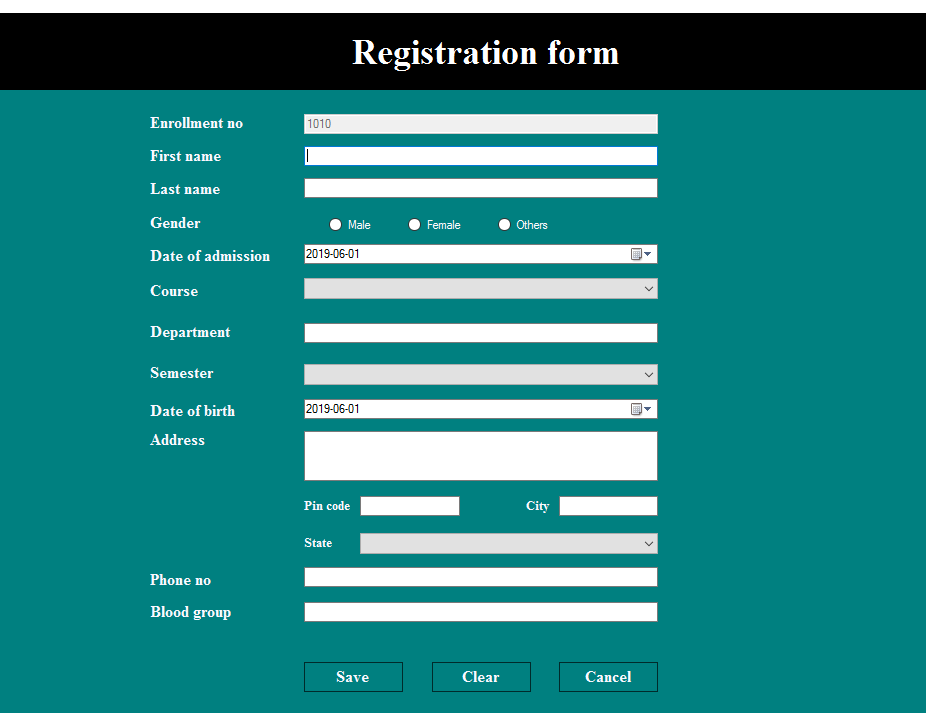
**4.1.1. Test for login screen**

****

**Fig. 4.1 Login Form**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sl.no** | **Objective** | **Steps to Perform** | **Test Value** | **Expected Results** | **Test Result** |
| 1. | To Check if the  ‘Login’ button works  properly | Click ‘Login’  Button | Username: Admin;  Password: 123456 | Incorrect  Username or  password | Pass |
| 2. | Username: admin;  Password: 123456 | Incorrect  Username or  password | Pass |
| 3. | Username: admin;  Password: 123 | Incorrect  Username or  password | Pass |
| 4. | Username: ADMIN;  Password: 123 | Incorrect  Username or  password | Pass |
| 5. | Username: ADMIN;  Password: 123456 | Enter homepage | Pass |
| 6. | To check if the checkbox  shows text on password  field upon checking it | Check ‘Show password’  while entering text in  password field | Password: 123456 | Password: ‘123456’  visible | Pass |
| 7. | To check if the checkbox  hides text on password  field upon unchecking it | Uncheck ‘Show password’  while entering text in  password field | Password: 123456 | Password  hidden | Pass |
| 8. | To Check if ‘Cancel’  Button works | Click ‘Cancel’ button | Click ‘Cancel’ | Software close | Pass |

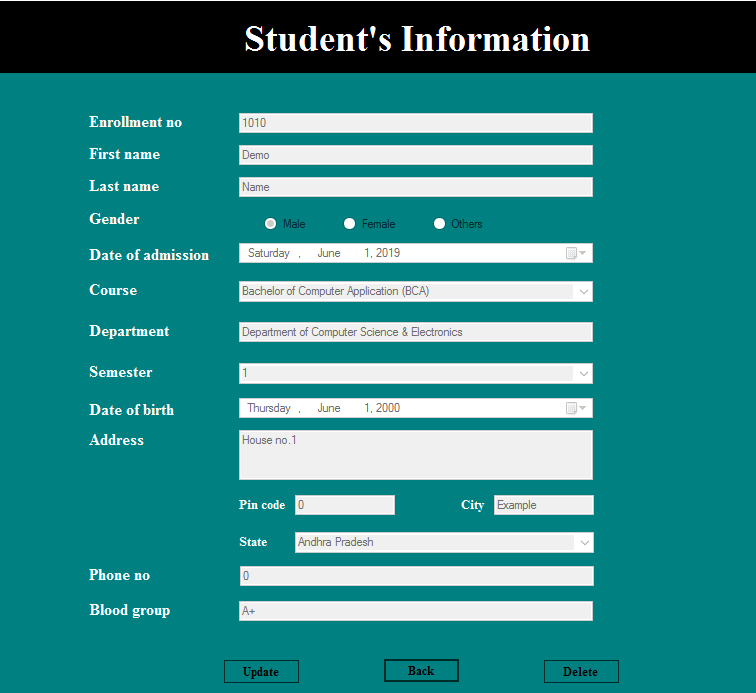
**4.1.2. Test for Registration form**



**Fig. 4.2 Registration form**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sl.no** | **Objective** | **Steps to Perform** | **Test Value** | **Expected Results** | **Test Result** |
| 1. | To check if the  ‘Save’ button works | Enter relevant details  and click ‘Save’ button | Enter all relevant  details | Message box  “Save” | Pass |
| 2. | Leave few textbox  blank and click ‘Save’  button | Leave few  textbox blank | Message box  “Enter Records  Properly” | Pass |
| 3. | To check if the  ‘Clear’ button works | Fill up the textbox  and click ‘clear’ | Fill up textboxes  with random  values | All textboxes  cleared | Pass |
| 4. | To check if the  ‘Cancel’ button  works | Click ‘Cancel’ | Nil | Display home  page | Pass |

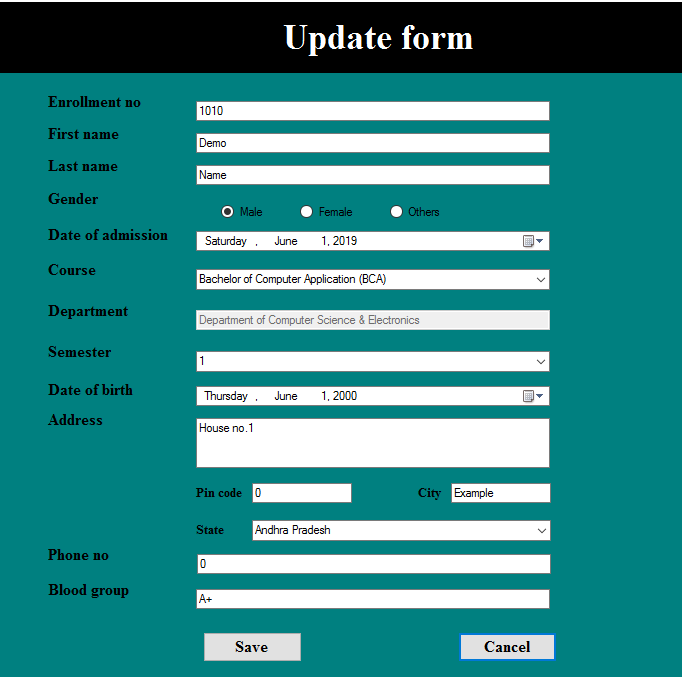
**4.1.3. Test for Search form**

****

**Fig. 4.3 Search form**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sl.no** | **Objective** | **Steps to Perform** | **Test value** | **Expected Results** | **Test Result** |
| **1.** | **To check if ‘Update’**  **button works** | **Click ‘Update’ button**  **and enter a value** | **1001** | **‘Update form’ is displayed**  **and Search form is hidden** | **Pass** |
| **2.** | **To check if ‘Back’**  **Button works** | **Click ‘Back’ button** | **Nil** | **Home page is**  **displayed** | **Pass** |
| **3.** | **To check**  **‘Delete’ button** | **Click ‘Delete’ button** | **Nil** | **Message “Record is**  **Deleted successfully”** | **Pass** |

**4.1.4. Update form**



**Fig. 4.4 Update form**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sl.no** | **Objective** | **Steps to Perform** | **Test value** | **Expected Results** | **Test Result** |
| **1.** | **To check ‘Save’ button** | **Change any values**  **and click ‘save’** | **Semester=2** | **Message box**  **‘Record updated**  **successfully’** | **Pass** |
| **2.** | **Leave any field empty** | **Nil** | **Message box ‘Field**  **cannot be empty’** | **Pass** |
| **3.** | **To check ‘Cancel’ button** | **Click ‘Cancel’** | **Nil** | **Display ‘Home’ page** | **Pass** |