**INTRODUCTION**

The main purpose using computerized system is to avoid manual problems and also documentation storage problem we can’t maintain long period data that’s why we used computerized system to overcome all problem related to college data storing and other areas

College Admission System is a desktop based project that maintains all the activity related to College. This project works on Microsoft Access Database

We can generate report according to date & show all report also; Because of manual system we faced many problems. The maintenance cost of manual system was very high. And they didn’t store historical information and not possible to view all at a time.

**OBJECTIVE**

College Admission System is a desktop based project that is maintain all the activity related to College, it stores all the information of student, College Admission Project provide student activity detail.

Hill fort project show time to time event information related to College.

* To provide student detailed information and course details.
* To make admission procedure fast and easy.
* To calculate rank according to the obtained marks of the student.

**PLATEFORM SPECIFICATION**

**Hardware Requirements**

1GHz + CPU

103MB RAM

**Software Requirement**

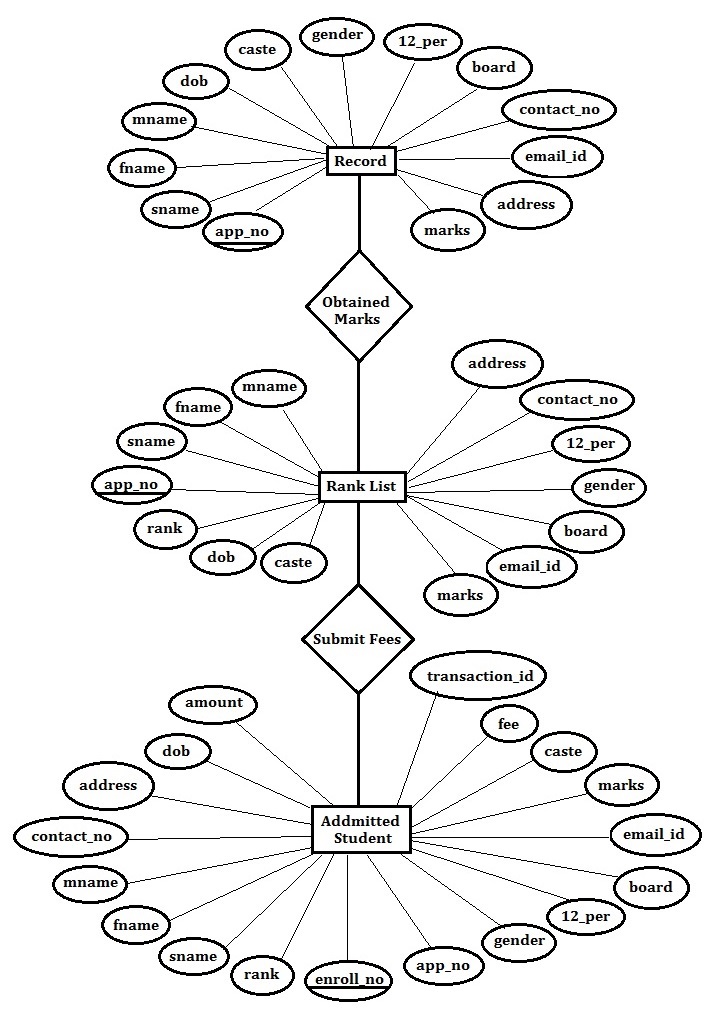
Operating System – Windows 10

Back end – MS Access

Front end – Visual Basic.Net

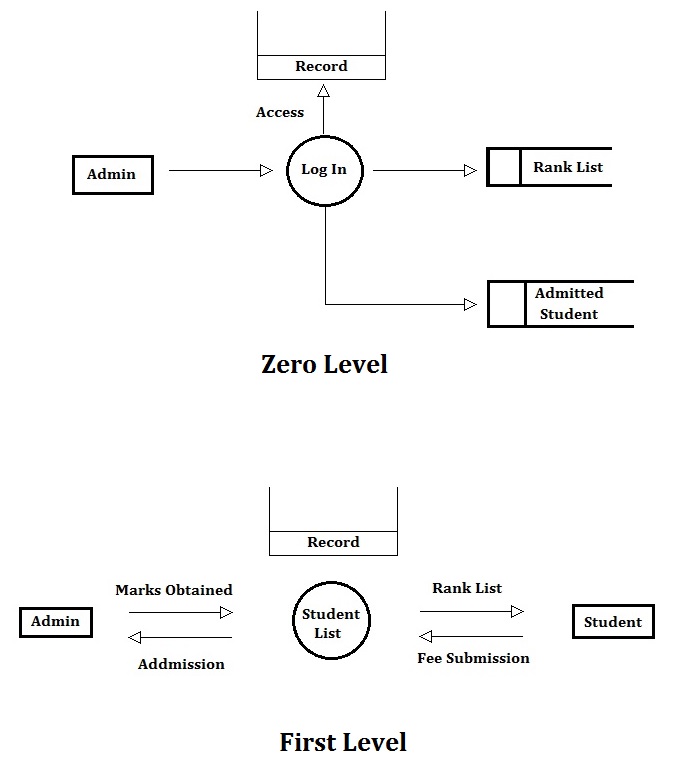
Microsoft Visual Basic 2010

**ER-Model**

****

**Figure - 1**

**DATA FLOW DIAGRAM (DFD)**

****

**Figure - 2**

**MODULE DESIGN**

**ADMIN**

The administrator logs in using the admin login. In this module two operations are done. During login the user id and Password is verified.

**INPUT DESIGN**

The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy.

**OBJECTIVES**

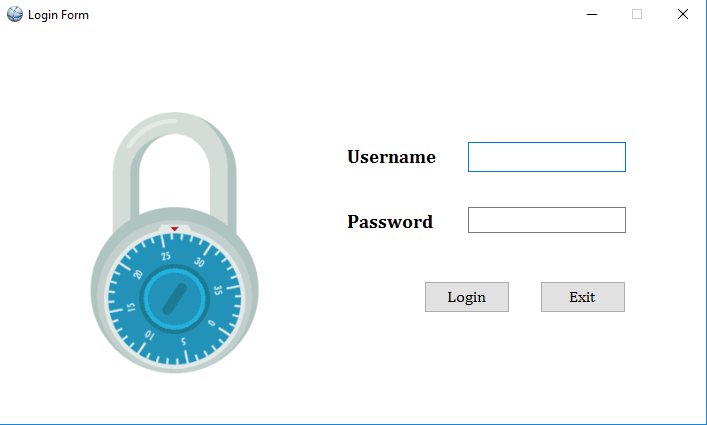
Input design is the process of converting a user-oriented description of the input into a computer based system. This design is important to avoid errors in the data input process and show the correct decision to the management for getting correct information from the computerized system. It is achieved by creating user friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

**SYSTEM DESIGN**

**INTRODUCTION**

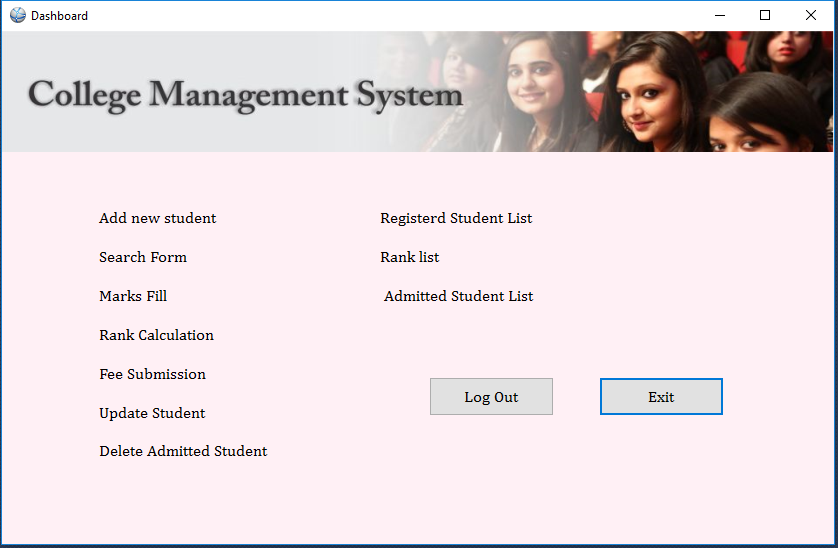
Design the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. THE term “design” is defined as “the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization”. It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless if the development paradigm that is used. The system design develops the architectural detail required to build a system or product. As in the case of any systematic approach, this software too has undergone the best possible design phase fine tuning all efficiency, performance and accuracy levels. The design phase is a transition from a user oriented document to a document to the programmers or database personnel.

**Login form**



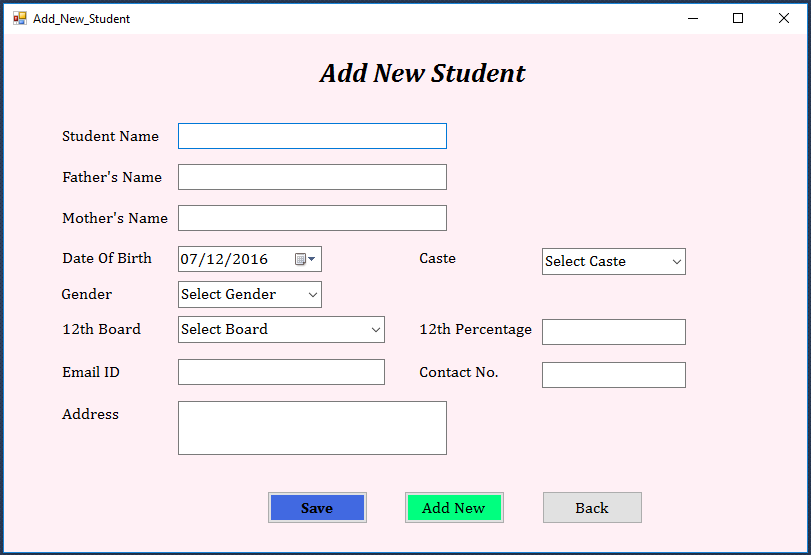
**Figure - 3**

**Dashboard Form**



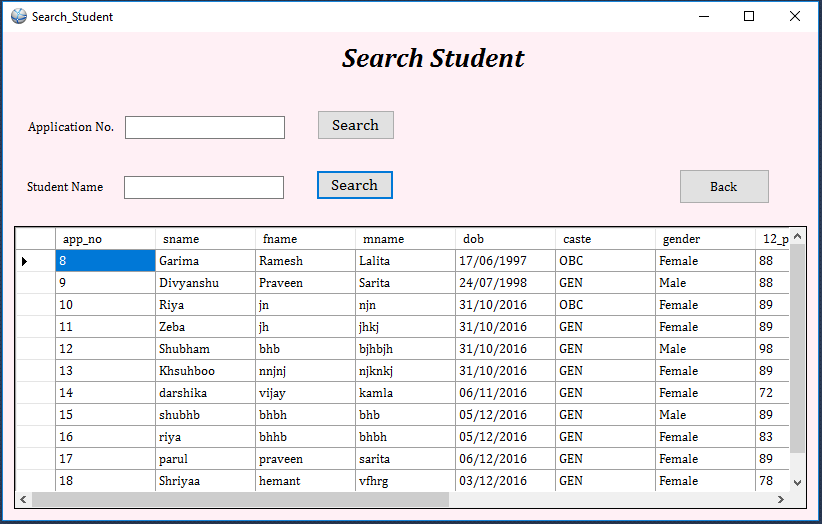
**Figure - 4**

**Add New Student Form**



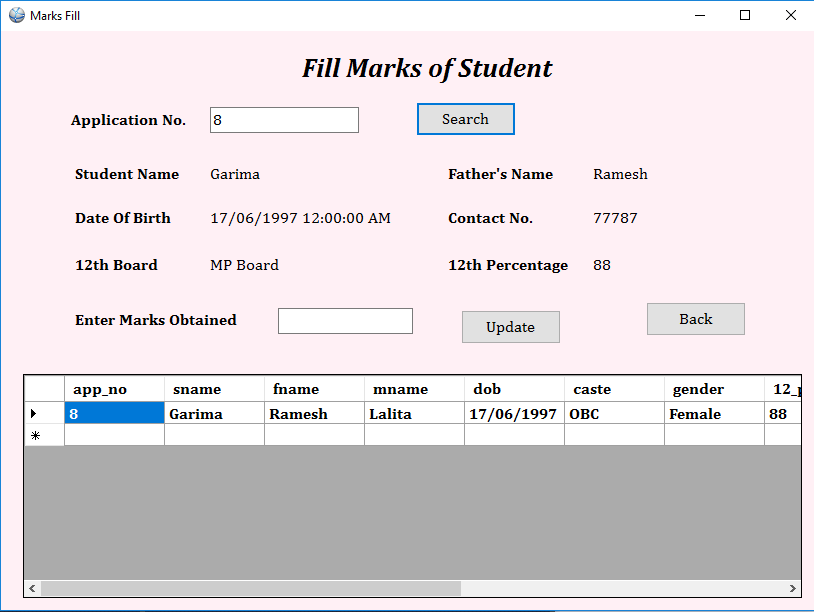
**Figure - 5**

**Search Registered Student Form**



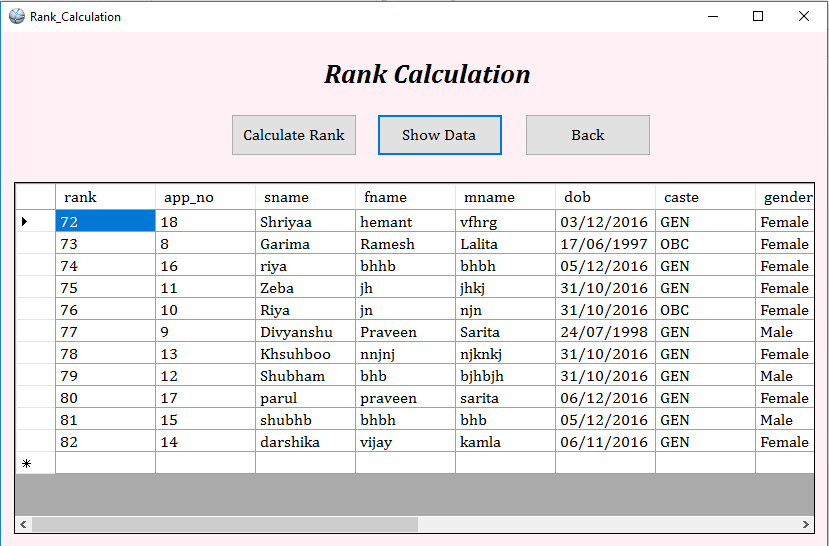
**Figure - 6**

**Marks Fill Form**



**Figure - 7**

**Rank Calculation Form**



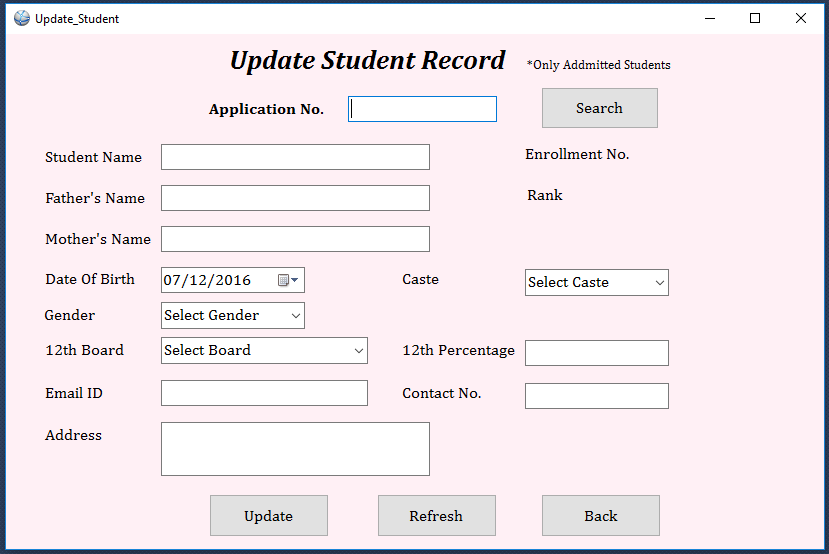
**Figure - 8**

**Fee Submission Form**

****

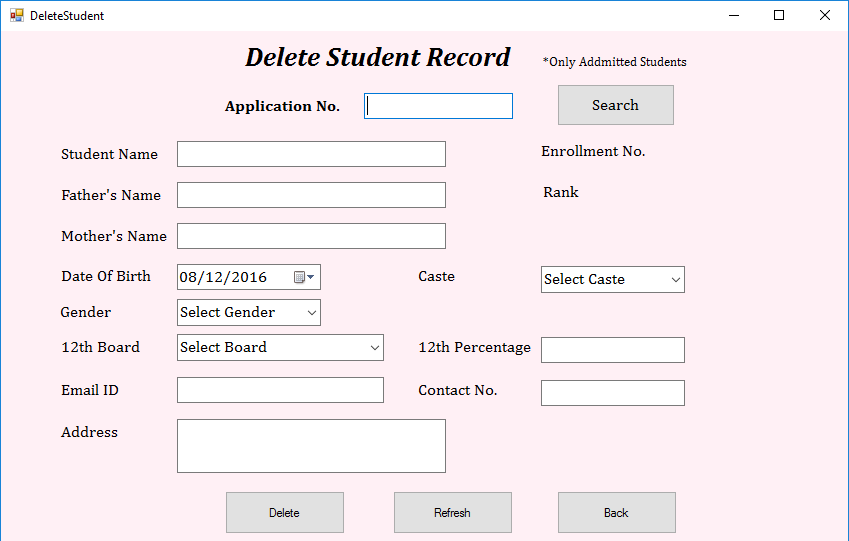
**Figure - 9**

**Update Student Form**

****

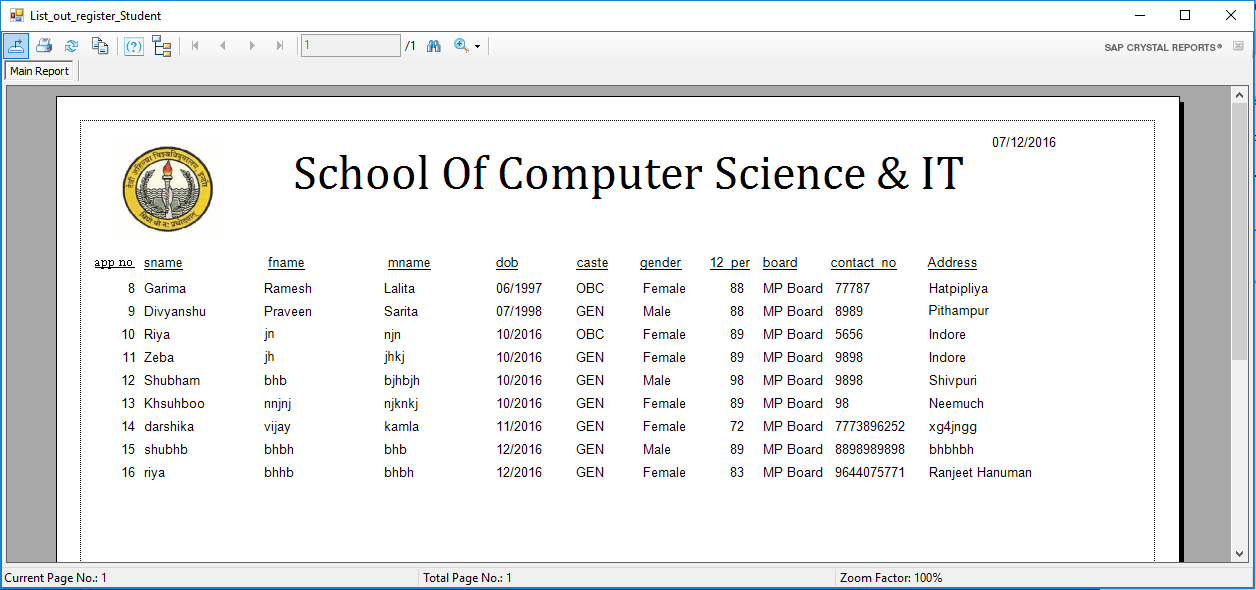
**Figure - 10**

**Delete Admitted Student Form**

****

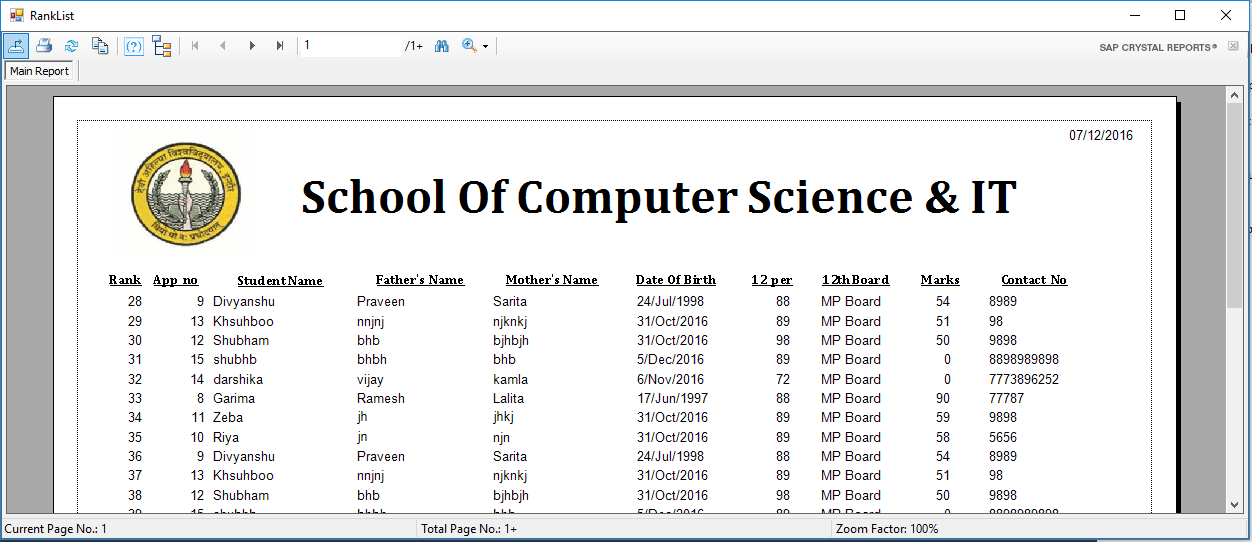
**Figure - 11**

**Report Of Registered Student**

****

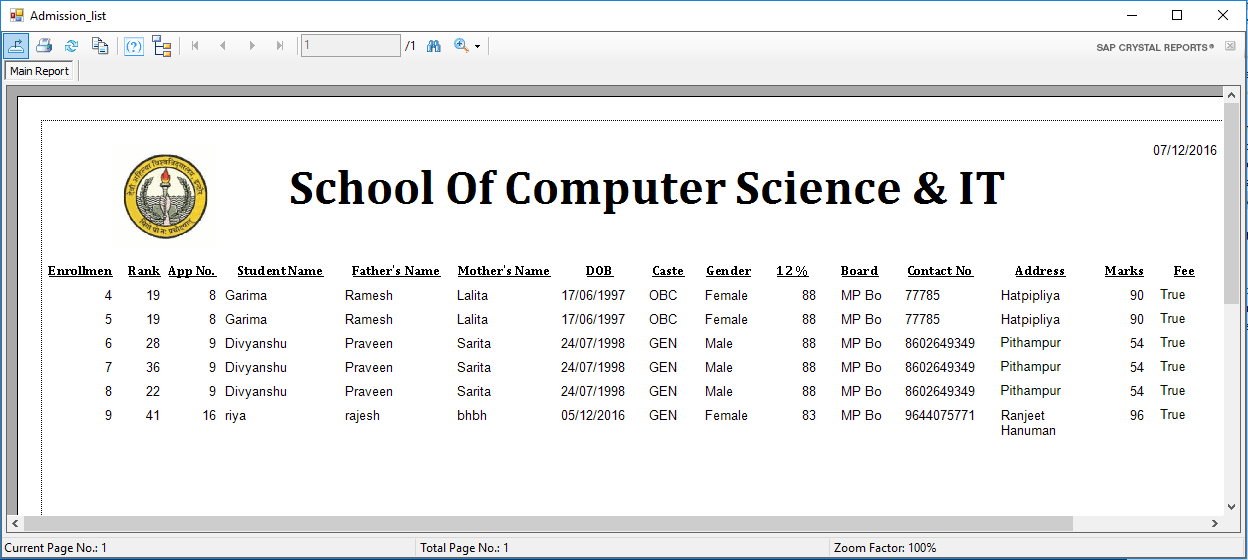
**Figure - 12**

**Report Of Rank List**

****

**Figure - 13**

**Report of Admitted Student**

****

**Figure - 14**

**SYSTEM IMPLEMENTATION**

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

**SYSTEM TESTING**

Software Testing is the process of executing software in a controlled manner, in order to answer the question-Does the software behave as specified? Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an association specification. Software testing is just one kind of verification, which also uses techniques such reviews, analysis, inspection, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Validation: Are we doing the right job right?

Testing is a process of execution a program with intend of finding an undiscovered error. A successful test is one that uncovers an undiscovered error. A successful test is one that uncovers an undiscovered error. If a testing is conducted successfully according to the objectives as stated above, it would uncovered errors in the software also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met. There are three ways to test program.

* For correctness
* For implementation efficiency
* For computational complexity

Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

**TEST PLAN**

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The test plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual of the programs, insuring that each performs the function for which it was designed. There is an independent test group (ITG)

Which is to remove the inherent problems associated with letting

The builder to test the thing that has been built. The specific objective of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated with in test plan.

The levels of testing include:

* Unit testing
* Integration testing
* Data validation testing
* Output testing

**UNIT TESTING**

Unit testing focuses verification effort on the smallest unit of software design- the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors with in the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithms execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, on error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other test are moot.

Selective testing of execution paths is an essential task during the unit test.

Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur. Boundary testing is a last task of unit testing step. Software often fails at its boundaries.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some clause in the internal logic of the modules were found and were rectified.

**INTEGRATION TESTING**

Integration testing is systematic technique for constructing the program structure while at the same time conducting test to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expense of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop.

After unit testing in Sell-Soft System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

**VALIDATION TESTING OR SYSTEM TESTING**

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System testing.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

**OUTPUT TESTING OR USER ACCEPTANCE TESTING**

The system considered is tested for user interface; here it should satisfy the firm’s need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points

* Input Screen Designs
* Output Screen Designs
* Display message to guide the user and the like.

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

**BIBLIOGRAPHY**

**Reference Book**

1. Steven Holster (2006) ’visual basic 6 black book’ Pearson
2. Database System Concepts, A. Silberschatz, H. F. Korth, and S. Sudershan, 5th Edition, McGraw Hill
3. Sql, Pl/Sql - The Programming Language of Oracle, 4/E by Ivan Bayross
4. System Analysis and Design: Awad, EM, Galgotia Publications Pvt. Ltd
5. Practical Standards for Microsoft® Visual Basic® .NET
6. Programming Microsoft® .NET
7. Microsoft® Visual Basic® .NET Language Reference
8. Programming Microsoft® Visual Basic® 6.0
9. Programming Windows®, Fifth Edition
10. Programming Collaborative Web Applications with Microsoft® Exchange 2000 Server
11. [www.youtube.com](http://www.youtube.com)
12. [www.google.com](http://www.google.com)
13. [www.tutorialspoint.com](http://www.tutorialspoint.com)
14. [www.programmingknowledge.com](http://www.programmingknowledge.com)
15. www.coderspoint.com