ISEM
(An Integrated Solution for Examination Management)

MAIN PROJECT REPORT

Submitted in partial fulfillment of requirement for the award of the degree of Master of Computer Applications of Cochin University of Science and Technology

By

ANGITHA SAJU
(Reg. No: 95580001)

DEPARTMENT OF COMPUTER APPLICATIONS
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY
Kochi- 682022, Kerala.
DECLARATION

I hereby declare that this project work and the report submitted to the department of Computer Applications, Cochin University of Science and Technology in the partial fulfillment of the award of degree of master of Computer Applications is an outcome of my own work.

A copy of the project has been submitted to the organization for which this project was developed.

To the best of my knowledge this project work or parts there , does not form a part of any other project work or thesis on the basis of which a degree or award was conferred on an earlier occasion.

Place: CUSAT
Date: 27th April 2011

ANGITHA SAHU
Reg.No : 95580001
Project Work
(Sixth Semester)

ISEM

(An Integrated Solution for Examination Management)

Bonafide record of work done by

ANGITHA SAJU
Reg.No : 95580001

Submitted in the partial fulfillment of the
Requirements for the award of the degree of
Master of Computer Application
of Cochin University of Science and Technology
April 2011

Project guide

Head of the Department

Internal Examiner

External Examiner

Submitted on:
ACKNOWLEDGEMENTS

I am thankful to god almighty for the blessings in the successful completion of my project “ISEM”.

I have a great pleasure in acknowledging the help given by various individuals throughout the project work. This project is itself an acknowledgement to the inspiration, drive and technical assistance contributed by many individuals.

I express my sincere and heartfelt gratitude to Mr. K.V.Pramod, Head of the Department of Computer Applications, for being helpful and cooperative during the period of the project.

I also express my deep gratitude to the project guide Ms. Malathi for her valuable guidance, timely suggestions and help in the completion of this main project.

I extend my sincere thanks to all the teaching and non-teaching staff for providing the necessary facilities and help. Without the support of anyone of them this project would not have been a reality. I am also thankful to all my friends and to my parents for their incredible support given to me in every aspect.

Sincerely
Angitha Saju
Reg.no-95580001
INDEX

1. SYNOPSIS
2. PREFACE
   2.1 ANALYSIS PHASE
   2.2 DESIGN PHASE
   2.3 DEVELOPMENT PHASE
   2.4 TESTING PHASE
   2.5 IMPLEMENTATION PHASE
3. INTRODUCTION
   3.1 ORGANIZATION PROFILE
   3.2 PROJECT PROFILE
      3.2.1. OBJECTIVES
   3.3 DOMAIN INFORMATION
4. SOFTWARE REQUIREMENT SPECIFICATION
   4.1 USER REQUIREMENTS
   4.2 FUNCTIONAL REQUIREMENTS
   4.3 HARDWARE REQUIREMENTS
   4.5 SOFTWARE REQUIREMENTS
   4.6 NON-FUNCTIONAL REQUIREMENTS
   4.7 DOMAIN REQUIREMENTS
   4.8 SECURITY REQUIREMENTS
5. SYSTEM ANALYSIS
   5.1 INTRODUCTION
5.2 EXISTING SYSTEM
5.3 PROPOSED SYSTEM
5.4 FEASIBILITY STUDY

6. SOFTWARE QUALITY ASSURANCE PLAN

7. SYSTEM DESIGN
   7.1 INPUT DESIGN
   7.2 OUTPUT DESIGN
   7.3 SYSTEM ARCHITECTURE
   7.4 DATABASE DESIGN
   7.5 DATA FLOW DIAGRAM

8. SYSTEM TESTING

9. SYSTEM IMPLEMENTATION

10. MODULAR DESCRIPTION

11. SCREEN IMAGES

12. FUTURE SCOPE

13. CONCLUSION

14. BIBLIOGRAPHY
1. SYNOPSIS

Our purpose is to provide an Examination Evaluation system. This software system will be helpful for the examination valuation of a University. This will manages the examinations valuation for the Post Graduation courses MCA and MBA. It provide hall ticket, arranges the answer papers for 1st, 2nd & 3rd valuation, process marks, publish results and provide mark lists and certificates for the University. By using this system we can search the marks and details of any student of any year. This is more relevant in the case of Universities which have large number of colleges affiliated to them.

It has a way for calculating the remuneration for the evaluators. It will also show the summary of each valuation camp and the passing board.

Details regarding the Project
1. Providing hall ticket.
2. Maintain panel of evaluators and informing them of events and manage their attendance
3. Provide a ’To Do List’ Everyday.
5. Remuneration.
6. Recording the marks and details after 3 valuations.
7. A passing board is organized.
8. Search

Users of the system
1. Administrator.
2. Administrate office authorities
2 PREFACES

2.1 ANALYSIS PHASE

The project ISEM collects information from the MCA, MBA department office of the university about the student, course subjects, internal marks and attendance details. It provides hall ticket for the eligible ones. It organizes the first, second and third valuation.

It also conducts the passing board. Mark list and certificate is also provided by ISEM.

During requirement analysis we carried out the following activities

Requirement gathering: We gathered possible information regarding the system by interacting with the corresponding office authorities, who give all the details which we needed. Which was really an interesting experience?

Analysis of the gathered requirement: A complete analysis of the gathered information was done. With this we get an exact and clear idea of the requirements of the client. Which help us to resolve conflicts, inconsistencies in the gathered requirements.

Conceptual modeling has been done in the analysis phase. In conceptual modeling we had identified all the entities needed for the ISEM. Some of the entities are the student, staff, hall ticket, valuation marks scholars, semester, branch, passing board etc:

2.2 DESIGN PHASE

In the design phase exam details, 3 valuation mark passing board decisions etc. are the primary objective to create a design that satisfies the agreed application requirements. In the design phase of the SDLC process continues to move from the “what” questions of the analysis phase to the “how” questions. The requirements prototype that was developed earlier during the analysis phase is gradually improved and extended to include all the specified functions of the application. The system documentation process also starts in this phase.

In physical design, the entities and attributes that have been identified in the analysis phase, has been implemented. The relationship between the entities and its attribute are defined using primary key and foreign key constraints. The size and the data
types of the attributes have been modified here. Whether the value of an attribute can be NULL or not can be also be specified. Architectural design is thriftier architecture namely.

Presentation layer
Application layer
Data layer

In Presentation layer we have used java. Also Java was used in Application layer. In Data layer we have used SQL server Test plan and the test cases for unit testing and system testing was also done in this phase. The review of all design phase has been reviewed in this phase itself.

2.3 DEVELOPEMENT PHASE

The development phase is the most exciting phase of the SDLC. During this phase computer hardware is purchased and the software is developed. Coding starts in this phase. In this phase, examination and re-examination of the requirements statement is needed to ensure that it is being followed as per customer needs. Any deviations would usually have to be approved either by the project leader or by the customer.

The development phase can be split into two sections, that of prototyping and production ready application creation, prototyping is the stage of the development phase that produces a pseudo complete application, which for all intents and purposes appears to be fully functional.

Developers use this stage to demo the application to the customer as another to be fully functional.

Developers use this stage to demo the application to the customer as another check that the final software answers the problem posed. When they are given the ok from the customer, the final version code is written into this shell to complete the phase.

We have used the prototype design which we had developed in the analysis and design phase in this phase. The coding of presentation layered data layer is done here. We have created stored procedures, functions, and constraints as a part of this phase.
2.4 TESTING PHASE

The testing phase requires organizations to complete various tests to ensure the accuracy of programmed code and the inclusion of expected functionality. Thorough testing is critical to ensuring systems meet organizational and end user requirements.

If organizations use effective project management techniques, they will complete test plans while developing applications, prior to entering the testing phase. Test plans created during initial project phases enhance an organization's ability to create detailed tests. The use of detailed test plans significantly increases the likelihood that testers will identify weaknesses before products are implemented.

Testing groups are comprised of technicians and end users who are responsible for assembling and loading representative test data into a testing environment. Functional tests should ensure that expected functional, security, and internal control features are present and operating properly. Testers then complete integration and end-to-end testing to ensure application and systems meet defined acceptance criteria.

Documenting corrections and modifications is necessary to maintain the integrity of the overall program documentation.

We have done unit testing, functional testing, integration testing, end to end testing and acceptance testing.
2.5 IMPLEMENTATION PHASE

The implementation phase involves installing approved applications into production environments. Primary tasks include announcing the implementation schedule, training end users and installing the product. Additionally organizations should input and verify data, configure and test system and security parameters and conduct post implementation reviews. Management should circulate implementation schedules to all affected parties and schedules parties and should notify users of any implementation responsibilities.

Since we have done a web application, the product will be uploaded into the server. We found everything works satisfactorily and meets all the requirements.
INTRODUCTION
3. INTRODUCTION

3.1 ORGANIZATION PROFILE

Cochin University of Science & Technology (CUSAT) is a government owned autonomous university in Kochi (Cochin), Kerala, India. Founded in 1971, the university consists of three campuses, two in Kochi and one in Kuttanad, Alappuzha, about 66 km inland. The university awards degrees in various fields of engineering and allied subjects at the undergraduate, postgraduate and doctoral levels. Nearly 2,000 students engage yearly in various areas of undergraduate and postgraduate study in this university. The institute was poised to become Indian Institute of Engineering, Science and Technology, Kochi (IIEST Kochi), but the proposal was rejected by the state government in a surprising turn of events.

Originally known as University of Cochin, the University came into being in 1971 through an Act of the Legislature of Government of Kerala that was the result of a concerted campaign for postgraduate education in the State of Kerala. The university was reorganized into Cochin University of Science and Technology (CUSAT) in February 1986. The process of reorientation resulted in redefining the objectives as the promotion of undergraduate and postgraduate studies and advanced research in applied science, technology, industry, commerce, management and social sciences. Admissions are based on an All India Entrance Examination known as Common Admission Test (CAT) conducted by the University which includes different papers for admissions to various undergraduate and postgraduate courses. Various Departmental Admission Tests (DAT) are also conducted for some postgraduate Courses.

Department of Computer Application

Computer Applications is one of the thrust areas in science and technology. In appreciation of its growing importance in business and visualizing the career prospects, the University established the Department of Computer Applications to facilitate research and human resource development in the subject. The M.C.A. programme was started (1994) with a model curriculum prepared jointly by ISTE and the Department of Electronics with minor modifications. The Syllabus is updated periodically, based on the current trends and requirements of the industry. The Department has a proud alumnus, most of them being placed in much reputed international firms like IBM, WIPRO, INFOSYS, TCS, CTS etc. The Research activities of the Department include the subject areas Fuzzy sets and Logic Applications in Artificial Intelligence, Simulation,
Cryptography & Coding Theory, Algorithm, Pattern recognition, Internet-Marketing, E-commerce and Internet Technology, Networking and Mobile Communication and Software Engineering. Apart from this, the Department has taken up a challenging research project funded by AICTE, Computer Assisted Classical Music. The Department of Computer Applications is also doing consultancy work for public and private sector undertakings. The Department has an excellent library with more than 3000 books and various national and international journals. The Department has a well-equipped laboratory, which is being constantly updated with the latest computers.

### 3.2 PROJECT PROFILE

<table>
<thead>
<tr>
<th>Title</th>
<th>ISEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Application</td>
</tr>
<tr>
<td>Project objective</td>
<td>This project is a system for Examination Valuation of MBA &amp; MCA in CUSAT Administrative office.</td>
</tr>
<tr>
<td>Organization</td>
<td>Department of Computer Application, Cochin University of Science and technology, Kochi-22</td>
</tr>
<tr>
<td>Duration</td>
<td>4 months</td>
</tr>
<tr>
<td>Project Guide</td>
<td>Dr Malathi., Lecturer, Department of Computer Application, Cusat</td>
</tr>
</tbody>
</table>

#### 3.2.1 OBJECTIVES

This software system will be used for the examination valuation of a University. This will evaluate the examination papers for MCA and MBA. It contains the details of the valuation camp and 1st, 2nd & 3rd valuation. By using this system we can search the marks and details of any student of any year. Remuneration which is given to the evaluators can also be calculated.

The system will be designed to maximize the productivity of the examination system by providing tools to assist in calculating moderation and remuneration of the staffs in the valuation camp according to their attendance. A good system invariably has good software that is a collection of computer programs, procedures, rules and associated documentation and data. The study of what is needed and evolving the best possible method to implement them is a very important step towards the goal of efficient software development. The same format of application form can be used by candidates appearing for all the examinations under this University. Cost of printing, maintaining and distributing different types of forms saved.
The major objectives of the system are

- Reduction/Elimination of repetition of work
- Decrease the work load
- Minimized processing time
- Error free information
- Efficient mark details
- Accurate remuneration details

3.3. Domain Information

Operating system : Windows Xp
Programming Languages : JAVA, MySQL
Development Environment : Netbeans IDE, WAMP SERVER

WINDOWS XP

Windows XP is a line of proprietary operating systems developed by Microsoft for use on general-purpose computer systems, including home and business desktops, notebook computers, and media centers. The letters "XP" stand for experience. Codenamed "Whistler" after Whistler, British Columbia, as many Microsoft employees skied at the Whistler-Blackcomb ski resort during its development, Windows XP is the successor to both Windows 2000 and Windows Me, and is the first consumer-oriented operating system produced by Microsoft to be built on the Windows NT kernel and architecture. Windows XP was first released on October 25, 2001, and over 400 million copies are in use, according to a January 2006 estimate by an IDC analyst. It is succeeded by Windows Vista, which was released to volume license customers on November 8, 2006, and worldwide to the general public on January 30, 2007.

Windows XP is known for its improved stability and efficiency over previous versions of Microsoft Windows. It presents a significantly redesigned graphical user interface, a change Microsoft promoted as more user-friendly than previous versions of Windows. New software management capabilities were introduced to avoid the "DLL hell" that plagued older consumer versions of Windows. It is also the first version of Windows to use product activation to combat software piracy, a restriction that did not sit well with some users and privacy advocates. Windows XP has also been criticized by some users for security vulnerabilities, tight integration of applications such as Internet Explorer and Windows Media Player, and for aspects of its user interface.
Windows XP had been in development since early 1999, when Microsoft started working on Windows Neptune, an operating system intended to be the "Home Edition" equivalent to Windows 2000 Professional. It was eventually cancelled and became Whistler, which later became Windows XP. Many ideas from Neptune and Odyssey (another cancelled Windows version) were used in Wind.

**JAVA**

Java is the first programming language designed from ground up with network programming in mind. The core API for Java includes classes and interfaces that provide uniform access to a diverse set of network protocols. As the Internet and network programming has evolved, Java has maintained its cadence. New APIs and toolkit have expanded the available options for the Java network programmer.

Java is both a programming language and an environment for executing programs written in Java language. Unlike traditional compilers, which convert source code into machine level instructions, the Java compiler translates Java source code into instructions that are interpreted by the runtime Java Virtual Machine. So unlike language like C and C++, Java is an interpreted language.

**Java Environment:**

The Java environment is composed of several separate entities.

**Java Language:**

This is a language that follows object-oriented concept used to create executable contents such as applications and applets. But Java is not pure object oriented language, it does not support multiple inheritance & Operator overloading.

**Java Runtime Environment:**

The runtime environment used to execute the code. It is made up of the java language and java virtual machine. It is portable and it is platform neutral.

**Java tools:**

It is used by the developers to create java code. They include java compiler, java interpreter, classes, libraries and applet viewer.

**Java Application:**
Applications are programs written in java to carry out certain tasks on stand alone local computer. Execution of a stand-alone program involves two steps. 
Compiling the source code into byte code using javac. 
Executing byte code program using java interpreter

**Java Applets:**
Java applets are pieces of java code that are embedded in HTML document using the applet tag. When the browser encounters such code it automatically download it and execute it.

**Java Virtual Machine:**
It is a specification to which java codes must be written. All java code is to be compiled in this nonexistent virtual machine. Writing the code that compiles in JVM ensures platform independence.

**Advantages of Java**

**Java is Robust:**
Robust programs are those reliable programs that are unlikely to fail even under the most unlikely conditions. Many languages like C do not have this feature because they are relaxed in terms of type checking in terms of programming errors. Java is strict about type declaration and does not allow automatic typecasting. Also it uses a pointer model that does not overwrite memory or corrupt data.

**Java is secure:**
Java allows creation of virus-free, tamper free systems to be created. It ensures security in the following ways.

- Pointers and memory allocations are removed during compile time.
- The interpreter verifies all byte codes before executing.
- All java applets are treated as entrusted code executing in trusted environment.

Because Java was written to support distributed applications over the computer networks, it can be used with a variety of CPU and operating system architectures. To achieve this goal a compiler was created that produces architecture-neutral object files from Java code.
Java is Portable:

Java byte code will be executed on any computer that has Java Runtime Environment. The portability is achieved in the following ways.

- Java primitive data types and the behavior of arithmetic operations on these data types are explicitly specified.
- The java libraries include portable interfaces for each platform on which the runtime environment is available.
- The entire java system itself is portable.

Java is small:

Because java was designed to run on small computers, java system is relatively small for a programming language. It can run efficiently on PCs with 4MB RAM or more. The java interpreter takes up only a few hundred-kilo bytes.

Java is garbage collected:

Java programs don’t have to worry about memory management. The java system has a built in program called the garbage collector, which scans the memory and automatically frees the memory chunks that are not in use.

Java is dynamic:

Fundamentally distributed computer environments must be dynamic. Java is capable of dynamic linking new libraries, methods and instance variables as it goes without breaking and without concern.

Java Swing:

The swing classes eliminate Java’s biggest weakness: Its relatively primitive user interface toolkit. Swing provides many new components and containers that allow us to build sophisticated user interfaces, far beyond what was possible with AWT. The old components have been greatly improved, and there are many new components, like trees, tables, and even text editors. It also adds several completely new features to Java’s user interface capabilities: drag-and-drop, undo, and the ability to develop our own “Look and Feel”, or the ability to choose between several standard looks. The swing components are all “lightweight”, and therefore provide more uniform behavior across platforms, making it easier to test our software.
**Reason for Using Java:**

It is required to explore systems running different operating system. In order to do so, there should be some way to connect to bridge those operating systems so that all the differences between them are solved and the functionalities are achieved. Also the functions performed in one system should be able to transfer to another and the result should be able to reflect there properly. Java serves as a bridge between these Operating systems. Also java is widely considered to be the best in developing network applications.

The communication happens between Java Virtual Machines running on the systems. When the client wants to perform the functionalities in another system and see the result, a method in the remote system is invoked from the client. The corresponding method in the remote system performs the job and sends the results to the client that is reflected in its interface.

**NETBEANS IDE**

The NetBeans Platform allows applications to be developed from a set of modular software components called modules. A module is a Java archive file that contains Java classes written to interact with the NetBeans Open APIs and a manifest file that identifies it as a module

**MySQL**

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Sun Microsystems, Inc.

The MySQL Web site (http://www.mysql.com/) provides the latest information about MySQL software.

- **MySQL is a database management system.**

  A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.
• **MySQL is a relational database management system**

  A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of “MySQL” stands for “Structured Query Language.” SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, “SQL-92” refers to the standard released in 1992, “SQL:1999” refers to the standard released in 1999, and “SQL:2003” refers to the current version of the standard. We use the phrase “the SQL standard” to mean the current version of the SQL Standard at any time.

• **MySQL software is Open Source.**

  Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), to define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us. See the MySQL Licensing Overview for more information.

• **The MySQL Database Server is very fast, reliable, and easy to use**

  If that is what you are looking for, you should give it a try. MySQL Server also has a practical set of features developed in close cooperation with our users. You can find a performance comparison of MySQL Server with other database managers on our benchmark page.

  MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

• **MySQL Server works in client/server or embedded systems.**

  The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backend, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

  We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.
• A large amount of contributed MySQL software is available.

It is very likely that your favorite application or language supports the MySQL Database Server.

The official way to pronounce “MySQL” is “My Ess Que Ell” (not “my sequel”), but we do not mind if you pronounce it as “my sequel” or in some other localized way.
SOFTWARE REQUIREMENTS SPECIFICATION
4. Software Requirement Specification

4.1 User Requirements

Our purpose is to provide an Examination Evaluation system. This software system will be helpful for the examination valuation of a University. This will manages the examinations valuation for the Post Graduation courses MCA and MBA. It provide hall ticket, arranges the answer papers for $1^{st}$, $2^{nd}$ & $3^{rd}$ valuation, process marks, publish results and provide mark lists and certificates for the University. By using this system we can search the marks and details of any student of any year.

It has a way for calculating the remuneration for the evaluators. It will also show the summary of each valuation camp and the passing board.

4.2 Functional Requirements

- $1^{st}$ valuation is conducted
- $2^{nd}$ valuation is conducted
- $3^{rd}$ valuation is conducted
- Details of evaluators and valuations are recorded
- provide remuneration by checking the attendance of evaluators and number of papers evaluated
- DA is considered here.
- Check whether evaluator had collected the money or not.
- Note a summary about the works during the time of valuation.
- Recording the marks and details after 3 valuations.

4.3 Hardware Requirements

Client Machine:

<table>
<thead>
<tr>
<th>Processor</th>
<th>Core 2 Duo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monitor</td>
<td>SVG Colour Monitor</td>
</tr>
<tr>
<td>RAM</td>
<td>256 MB</td>
</tr>
<tr>
<td>FREE SPACE</td>
<td>1 GB</td>
</tr>
</tbody>
</table>

Server Machine:
4.4 Software Requirements

Client Machine:

Operating System : Windows xp
Development Kit    : Net beans IDE 6.9.1
Dependencies       : JavaScript
Browser            : Mozilla Firefox
Language           : Java

Server Machine:

Back End           : MYSQL
Web Server         : Apache Tomcat 6.0.26

4.5 Non-functional Requirements

User constraints are the non functional requirements. It must contain the language and the operating system to be used in the developing of the software. This software system is used for managing the examination valuation. If the valuation is computerized then it can provide better outcomes. All the details of the valuation can be viewed at any time.

4.6 Domain requirements

Domain requirements are derived from the application domain of the system rather than from the specific needs of the system users. May be new functional requirements, constrain existing requirements or set out how particular computation must take place.

4.7 Security Requirements

An id and password is given to the authority for keeping the marks and details secured.
SYSTEM ANALYSIS
5. SYSTEM ANALYSIS

5.1 INTRODUCTION

Analysis can be defined as “breaking up of any whole so as to find out their nature, function etc. a statement of these findings”. It defines design as “to make preliminary sketches of; to sketch a pattern or outline for plan. To plan and carry out especially by artistic arrangement or in a skillful way”. System analysis and design can be characterized as a set of techniques and processes, a community of interests, a culture and an intellectual orientation.

The various tasks in the system analysis include the following:

- Understanding application.
- Planning.
- Scheduling.
- Developing candidate solution.
- Performing trade studies.
- Recommending alternative solutions.
- Supervising, installing and maintaining the system.

System analysis involves the study of an application area to fully understand the problem being posed. Activities are focused on developing a comprehensive knowledge of the existing system, its strengths and weaknesses and the reasons for the need to restructure, replace, or automate the existing system. Risk assessments and risk containment plan, cost estimation and plans for the remainder of the development are results usually prepared by the system analyst as a by-product of system analysis.


System Planning and Initial Investigation

The most critical phase during system analysis is planning. To launch a system investigation, we need a clear cut plan detailing the steps to be taken. Initial investigation has the objective of whether the user’s request has potential merit.

Information Gathering

A key part of system analysis is gathering information of the present system about the factors affecting the system, the drawbacks and merits of the system, present activities and the process of the system etc. The traditional tools used for information gathering are interview, questionnaire and on-site observation. They help in evaluating the existing system.

Need for computerization

The drawbacks found in the existing system above speak the name for computerization. Almost all the above drawbacks can be removed to a great extent if not completely, through computerization. Hence computerization is essential to

- Minimize human efforts
- Reduce error drastically
- Speedup the process
- Remove redundant data
- Improve efficiency
- Reduce manpower

Applying analysis tools for structured analysis:
The traditional tools for information gathering have many drawbacks. Because of these drawbacks, the analyst needs to focus on functions rather than physical implementation.

Therefore, structured tools, such as Dataflow Diagrams, Data Dictionary and Structured English are used along with the traditional tools. The main objectives of structured tools are to study user affected areas, model new logic systems, select hardware and software etc.

**Feasibility Study**

It describes and evaluates the proposed system. Cost estimate also taken in to consideration hardware, personal facility and supply costs for final evaluation.

**5.2 EXISTING SYSTEM**

In the existing system only we can see the details of some information about the student mark in a university. The existing system has more workload for the authorized person, but in the case of Proposed the existing system only we can see the details of particular information about the student mark, camp details, and no more information about their 1\textsuperscript{st} valuator, 3\textsuperscript{rd} valuator, their remuneration details etc.. Also the existing system has the following demerits

- More man power.

- Time consuming.

- Co
nsumes large volume of pare work.
• Ne
eds manual calculations.
• No
direct role for the higher officials.
• Da
damage of machines due to lack of attention.

To avoid all these limitations and make the working more accurately the system needs to be computerized.

5.3 PROPOSED SYSTEM

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work. The system provides The existing system has several disadvantages and many more difficulties to work well. The proposed system tries to eliminate or reduce these difficulties up to some extent. The proposed system will help the user to reduce the workload and mental conflict. The proposed system helps the user to work user friendly and he can easily do his jobs without time lagging.

Advantages

The system is very simple in design and to implement. The system requires very low system resources and the system will work in almost all configurations. It has got following features
• En
  sure data accuracy’s.
• Pro
per control of the higher officials.

- Reduce the damages of the machines.
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- Better service.
- User friendliness and interactive.

- It provides facility to enter the students details.
- It provides a facility to enter the details of staff.
- Add, edit the internal mark of students by subject wise and student wise.
- Add the external mark details after the three valuations.
- It provides mark list for each student in each semester.
- It provide certificate.
- It shows the details of passing board.
calculates the remuneration details.

5.4 FEASIBILITY STUDY

Preliminary investigation examines project feasibility, the likelihood the system organization. All projects are feasible given unlimited resources and infinite time. Unfortunately the development of a computer based system or product is more likely plagued by a scarcity of resources and difficult delivery dates. It is both necessary and prudent to evaluate the feasibility of the project at the earliest possible time. Feasibility and risk analysis are related in many ways.

Feasibility study is a test of system proposal according to its work ability, impact on the organization, ability to meet the users need, and effective use of resources. A feasibility study is conducted to identify the best system that meets all the requirements. This entails an identification description, an evaluation of proposed systems and the selection of the best system for job. The requirement of the system is specified with a set of constraints such as system objectives and description of the outputs. Then the feasibility of the proposed is evaluated to generate the above results. Depending of the results of the initial investigation, the study expanded to more detailed feasibility study. During this study, the problem definition is crystallized and aspects of the problem to be included in the system are determined. Consequently, costs and benefits are estimated with greater accuracy in this stage.

Three key considerations are involved in the feasibility analysis. They are

- Technical Feasibility
- Behavioral Feasibility

Technical Feasibility
There are a number of technical issues, which are generally raised during the feasibility stage of the investigation. A study of function, performance and constraints that may affect the ability to achieve an acceptable system.

The considerations that are normally associated technical feasibility include:

1. Development risk.
2. Resource availability.
3. Technology.

Technical feasibility centers around the existing computer system (Hardware, Software etc.) and to what extend it can support the proposed addition. This involves financial considerations to accommodate technical enhancements. If the budget is a serious constraint, then the project is judged not feasible.

Operational Feasibility

Proposed projects are beneficial only if they can be turned into information systems that will meet the operating requirements of the organization. This test of feasibility asks if the system will work when it is developed and installed. This project satisfies all the operational conditions.

Legal Feasibility

A determination of any infringement, violation, or liability that could result from development of the system. Legal feasibility encompasses a broad range of concerns that include contracts, liability, infringement, and myriad other traps frequently unknown to technical staff.

Alternatives

An evaluation of alternatives approaches to the development of the system or product. The degree to which alternatives are considered often limited by cost and time
constraints; however a legitimate but unsponsored variation should not be buried. The feasibility study may be documented as a separate report to upper management and included as an appendix to the system specification. The feasibility study is reviewed first by project management and upper management. The study should result in a “go/no go” decision.
6. SOFTWARE QUALITY ASSURANCE PLAN

The project isem collects information from the department office of the university about the student, internal marks, staff details, fee details, etc: Then it provide hall ticket for the eligible candidates. After that it conducts 1st, 2nd, and 3rd valuations. In the 2nd
valuation it organizes a camp for paper valuation. At last it conducts a passing board. Also isem provide mark list of each semester and certificate.

**Purpose**

The purpose of this plan is to define the software quality assurance (SQA) for record keeping system, SQA tasks and responsibilities; provide reference documents and guidelines to perform the SQA activities; provide the standards, practices and conventions used in carrying out SQA activities; and provide the tools, techniques and methodologies to support SQA activities.

**Scope**

The plan establishes the SQA activities performed throughout the life cycle of the record keeping system. The goal of SQA program is to verify that all the software and documentation to be delivered.

**Reference Documents**

- Software engineering, Roger S Pressman
- IEEE guide for software quality assurance planning, IEEE std 730.1-1995
- IEEE standard for software quality assurance plans, IEEE std730-1998

**Overview of the document**

The rest of the document is organized as follows:

**Management**: A description of each major element of the institution and a description of the SQA tasks and their relationships.

**Documentation**: Identification of the documents related to development, verification, validation, use and maintenance of the software.

**SQAP Requirements**: This section defines the SQA review, reporting, and auditing procedures used to ensure that software deliverables are developed in accordance with this plan and the project’s requirements.

**Training**: This section describes the training program for the developer.
**SOAP REQUIREMENTS**: This section defines the SQA review, reporting, and auditing procedures used to ensure that deliverables are developed in accordance with this plan and the projects requirements.

**MANAGEMENT**

**ORGANIZATION**

This tool is developed as an individual project as part of partial fulfillment of requirements in masters degree in computer application. Since there is only 2 members involved in the development team, it will be the sole responsibility of the developer to review the product’s usability, efficiency, reliability, and accuracy. The project guide will however conduct inspections and walkthrough on regular basis. Her suggestion will be used in places where quality decisions need to out-weight development schedule decisions.

**ROLES**

Project guide: Mrs Malathy (faculty)
Developers: Angitha Saju, Nishi Muhammad Ali, Sini mol V R

The responsibilities of the developer are as follows:

- Develop the requirement specification for the project.
- Develop the design plan and test plan for testing the application.
- Implement and test the application and deliver the application along with the necessary documentation.
- Give a formal presentation to the project guide and director of the institution on completion of the analysis, design and testing phases. The project guide and director of institution reviews the developer’s work and provides feedback / suggestions.

**SQA IMPLEMENTATION IN DIFFERENT PHASES**

Quality assurance plan will be implemented through all the software life cycles of the tool’s development process, until the releases of the software product. The following are the quality assurance tasks for each phase of the software development.
**Requirements phases**: when the SRS is being developed, the developer has to ensure that it elucidates functionality of the product and to keep refining the SRS until the requirements are clearly stated and understood.

**Specification and Design phase**: Due to the great importance for accuracy and completeness in these documents, weekly reviews shall be conducted between the developer and the project guide to identify any defects and rectify them.

**Implementation phase**: The developer shall do code reviews when the construction phase of the Tool begins.

**Software testing phase**: The developer shall test each case. The final product shall be verified with the functionality of the software as specified in the Software Requirement Specification (SRS) for the Tool.

Through all these phases of the software development, the following shall also be conducted to improve the software quality:

- Develop and generate SQAP: Generate a finalized SQAP plan
- Communication and Feedback: The developer is encouraged to freely express disagreements, suggestions and opinions about all aspects of the weekly process of software development.
- Internal audits and evaluations: The project guide is expected to do audits and evaluations at the end of each phase in the project.

**DOCUMENTATION**

In addition to this document, the essential documentation will include:

**The Software Requirements Specification (SRS)**

- Prescribes each of the essential requirements (functions, performances, design constraints and attributes) of the software and external interfaces.
- Objectively verifies achievement of each requirement by a prescribed method (e.g. Inspection, analysis, demonstration or test)
- Facilitates traceability of requirements specification to product delivery.
- Gives estimates of the cost/effort for developing the product including a project plan.

**The Software Design Description (SDD)**
➢ Depicts how the software will be structured
➢ Describes the components and sub-components of the software design, including various packages, if any.
➢ Gives a sample data flow diagram, showing the key interactions in the application.

Software Test Plan
Describes the test cases that will be employed to test the product.

SQA PROGRAM REQUIREMENTS

Standards
➢ Coding standards- JAVA
➢ Coding Documents standards- JAVA Documentation
➢ Test Standards- IEEE Standard for software test documentation

Metrics
➢ LOC – lines of code is used to measure the size of software

Software Documentation Audit
Quality assurance for this project will include at least one review of all current work products in each stage of development (Requirement, Design, and Implementation). The reviews will assure that the established project processes and procedures are being followed effectively and exposures and risks to the current project plan are identified and addressed. The review process includes:
➢ A formal presentation at the end of each development phase (Requirement, Design and Implementation). All current work products are presented to the committee members for review.
➢ A managerial review by the advisor periodically to ensure the work generated is in compliance with project requirements.
➢ Reviews by the committee after each presentation.

Requirements Traceability
The SRS will be used to check off the deliverables. The Project Review will ensure that each of the requirements mentioned in the SRS is met by the deliverables.
Software Development Process.

The software development process involves three stages: 1) Requirement phase, 2) Design phase (this phase involves the development of the product prototype) and 3) Implementation and testing phase. During each phase, the project guide and the committee will review the deliverable documents. The developer would incorporate modifications suggested by the committee. This would ensure quality of the software product.

**Testing and Quality Check**

Testing will be carried out in accordance with the Software Testing Plan (STP). Testing documentation will be sufficient to demonstrate that testing objectives and software requirements have been met. Test results will be documented and discussed in the final phase of the project.
7. SYSTEM DESIGN

Design is the first in the development phase for any engineered product or system. System design is the process of evaluating alternate solutions, evaluating the choice following up the specification for the chosen alternative. System design work follows logical system analysis. The objective of the system design is to improve the
existing system or design a new system as the case may be and implement the system with improved facilities.

Computer software design, like engineering design approaches in other disciplines, changes continually as new methods, better analysis and broader understanding evolved. Using one of the design methods, the design steps reduce a data design, architectural design and procedural design.

### 7.1 INPUT DESIGN

Input design is the part of the overall system design that requires very careful attention and is the most expensive phase. It is the point of contact for the users with the computer system and so it itself is prone to error. If data going into the system is incorrect then processing and output will magnify these errors. Objective during input design are as follows:

- Produce cost effective method input.
- Achieve high-level accuracy.
- Ensure that input is free of ambiguity.

The input type involves converting the user-originated inputs into a computer-based format. The aim of the computer design is to make the data entry easier, logical error free. It helps us to filler errors in the input data that otherwise entered into the database might have brought in a lot of inconsistency.

Alert for wrong entries such as primary key duplication, letters in numeric data, wrong data format, range exceed have been provided in the application. Upon this, a well-documented instruction set and navigation maps have been also provided for the non-frequent and first-time users to familiarize them with our web site.

Maximum care has been taken to ensure that users type in only minimum data into the system, as all he or she will have to do is to move and click the mouse or strike a key to select the desired data at the desired position.

The screens are designed in such a way that the user can find the needed components like options, actions etc. with ease of use. The needed columns where interaction is needed like Labels, Push Buttons etc. are also simple. The related data
columns are clubbed together as groups, this is made so that the user can understand the related data easily and the data entered suits the purpose.

The input design is the link between the information system and the user. It comprises developing specification and procedures for data preparation and those steps that are necessary to put input data into a usable form for processing data entry. Instructing the computer to read data from a written or a printed document can achieve the activity of putting data into the computer for processing or it can occur by having people key data directly into the system. The design of inputs focuses on controlling the amount of inputs required, controlling errors, avoiding delay, avoiding extra steps and keeping the process simple.

The main input designs used in this system are:

- Home page
- Add details such as staff details, student details, valuation marks etc.
- Search form for marks, camp details, passing board etc.
- To Do List form
- Login page
- Remuneration form

7.2 OUTPUT DESIGN

An inevitable activity in the system design is the proper design of output in a form acceptable to the user. Outputs from the system are required primarily to communicate the result of processing to users.

Outputs also provide a permanent copy of the results for later consultation. An intelligible output will improve system relationship with the user and help in the decision-making process.

The various types of outputs required by most systems are:

- **External Outputs**: Whose destination is outside the organization and which require special attention
- **Internal Outputs**: Whose destination is within the organization and which require careful design because they are user’s main interface with the computer?
- **Operational Outputs**: Whose use is purely within the computer department?
- **Interactive Outputs**: Which involve the user in communicating with the computer?
The approach to output design is very dependent on the type of output and nature of data. Special attention has to be made to data editing. The choice of appropriate output medium is also an important task. The selection may be affected by the following kinds of consideration.

- Response time.
- Location of users.
- Cost.
- Software/hardware.
- Suitability of the device for application concerned.

The output design must be specified and documented, data items have to be accurately defined and arranged for clarity and easy comprehension. The other two objectives that were taken care of were:

The interpretation of the results of the computer part of the system to users in a form that they can understand meets their requirements.

The output Design specification is made in such a way that it is unambiguous, comprehensive and capable of being translated into a programming language.

The main output designs are the following

- To view staff details form
- Mark details form.
- Mark list form.
- Course certificate form.
- To view camp details form
- To view passing board details.
- Staff details form
- Details of students.

7.3. SYSTEM ARCHITECTURE

7.3.1 Architectural Design

![User input and Output diagram]
7.3.2 Decomposition Description

Input: - The input of this system are the following
1. Internal marks of the student.
2. Semester fee paid details.
3. Staff details.
4. Camp attendance and number of papers checked by each evaluator.
5. Passing board details.

Database:
These inputs are stored in the database and effectively retrieved and produced as result.

Output: The output of the system is the following
1. Students total marks and grade in each semester.
2. Certificate after completing the course.
3. Camp details for each semester.
4. Remuneration details.

7.3.3 DATA DESIGN

The major data values that are considerable in this project are listed below
1. Students external mark details of each semester.
2. Passing board details.
3. Camp details.
4. Hall ticket details.

7.4 DATABASE DESIGN
The objectives of database design are to provide effective auxiliary storage and to contribute to the overall efficiency of the computer program component of the information system. It can be defined as a representation of an information system in a computer. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the user. In database design, several specific objectives are considered:

- Controlled redundancy
- Ease of learning and use
- Data independence
- More information at low cost
- Accuracy and integrity
- Recovery from failure
- Privacy and security
- Performance

The schema is the view that helps the dbms decide what data In storage should act upon as requested application program. The subschema is concerned with a relatively small part of schema. In database design several views of data must be considered along with the persons who use them. The logical views is what the data look like regardless of how they are stored, the physical view is the way data exists in physical storage. It deals with how data are stored, accessed or related to other data in storage. The logical view is the users view. A field, which is a member of any candidate key, is called a prime field. Any field of an n-set that is not prime field is referred to as a non prime field. Every n-set has a primary key which is an arbitrary choice of one of the candidate keys of the n-set. It is usually a requirement that the primary key is fully defined or in other words, that undefined value is not permitted in any of the component field of the primary key.

Normalization is a formal process of developing data structures in a manner that eliminates redundancy and promotes integrity. The process of normalization is concerned with the transformation of the conceptual schema into a compute represent table form. There are four steps to data normalization, which are called the normal forms

1. **The first normal form**: Elimination of repeating groups
2. **The second normal form**: Elimination of redundant data
3. **The third normal form**: Eliminates columns that are not depending on the key
4. **The fourth normal form**: Isolate independent relation
An entity is something of interest to the user about which to collect or stored data. It is also called a data aggregate because it represents a number of data elements. Data entities are explained by the use of several terms: attribute, value key, an instance of an entity. Each attribute takes on a value for a specific occurrence of an entity key is a unique identifier of the entity. A candidate key $k$ for an $n$-set is the set of one or more fields of the $n$-set with the following two purposes: the candidate key $k$ uniquely identifies each $n$ vector that is no two $n$-vectors have the same value for $k$.

**DATABASE DESIGN**

1. **Table Name: Camp_Details**
   For entering the details of the second valuation camp.
   Date_of_begin, Branch : Primary key
Staff_id: Foreign key

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staff_name</td>
<td>varchar</td>
<td>30</td>
<td>Not null</td>
<td>Faculty members name</td>
</tr>
<tr>
<td>Staff_id</td>
<td>Int</td>
<td>11</td>
<td>Foreign key</td>
<td>Id of the faculty</td>
</tr>
<tr>
<td>Informed_not</td>
<td>varchar</td>
<td>10</td>
<td>Not null</td>
<td>Informed about the camp details.</td>
</tr>
<tr>
<td>Date_of_begin</td>
<td>date</td>
<td>10</td>
<td>Primary key</td>
<td>Date when the camp started.</td>
</tr>
<tr>
<td>Subject_to_value</td>
<td>varchar</td>
<td>20</td>
<td>Not null</td>
<td>Subject given for the evaluator.</td>
</tr>
<tr>
<td>End_date</td>
<td>Date</td>
<td>10</td>
<td>Not null</td>
<td>Date when the camp ended.</td>
</tr>
<tr>
<td>Summary</td>
<td>varchar</td>
<td>200</td>
<td>Not null</td>
<td>Summary of the particular camp.</td>
</tr>
<tr>
<td>Branch</td>
<td>varchar</td>
<td>10</td>
<td>Primary key</td>
<td>Branch for which the camp conducted.</td>
</tr>
<tr>
<td>Semester</td>
<td>varchar</td>
<td>10</td>
<td>Not null</td>
<td>Semester for which the camp conducted.</td>
</tr>
<tr>
<td>attendance</td>
<td>int</td>
<td>10</td>
<td>Not null</td>
<td>Total no. of attendance in the camp.</td>
</tr>
</tbody>
</table>

2. Table Name: Hall ticket
For entering the details of providing the hall ticket.

Reg_no : Primary key

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reg_no</td>
<td>Int</td>
<td>11</td>
<td>Primary key</td>
<td>Registration no. of a student</td>
</tr>
<tr>
<td>Attendance</td>
<td>Double</td>
<td>11</td>
<td>Not null</td>
<td>Attendance percentage of a student</td>
</tr>
<tr>
<td>Fee_paid</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Whether the student had paid the fees</td>
</tr>
<tr>
<td>Condonation_paid</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Whether the student had paid the condonation</td>
</tr>
<tr>
<td>Semester</td>
<td>Varchar</td>
<td>23</td>
<td>Not null</td>
<td>Semester for which the camp conducted.</td>
</tr>
<tr>
<td>Year</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Year which student writing the exam.</td>
</tr>
</tbody>
</table>

3. **Table Name : Passing Board**
   For entering the Passing board details
   Date, sem : Primary key
   Members_id: Foreign key
### Table Name: Members

For entering the details of the faculty members.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>members_id</td>
<td>Int</td>
<td>10</td>
<td>Foreign key</td>
<td>Faculty members id</td>
</tr>
<tr>
<td>Date</td>
<td>date</td>
<td>11</td>
<td>Primary key</td>
<td>Date in which passing board conducted</td>
</tr>
<tr>
<td>Sem</td>
<td>Int</td>
<td>10</td>
<td>Not null</td>
<td>Semester for which the passing board conducted</td>
</tr>
<tr>
<td>Branch</td>
<td>varchar</td>
<td>10</td>
<td>Primary key</td>
<td>Branch for which the passing board conducted</td>
</tr>
<tr>
<td>Summary</td>
<td>mediumtext</td>
<td>200</td>
<td>Not null</td>
<td>Summary of the particular passing board</td>
</tr>
<tr>
<td>max_marks_student</td>
<td>float</td>
<td>10</td>
<td>Not null</td>
<td>Maximum marks given for each student by the passing board</td>
</tr>
<tr>
<td>subject1</td>
<td>varchar</td>
<td>20</td>
<td>Not null</td>
<td>Subject to which marks are given</td>
</tr>
<tr>
<td>mark1</td>
<td>float</td>
<td>10</td>
<td>Not null</td>
<td>Marks given to a particular subject</td>
</tr>
<tr>
<td>special_sub</td>
<td>varchar</td>
<td>20</td>
<td>Not null</td>
<td>Subject to which special marks are given</td>
</tr>
<tr>
<td>special_marks</td>
<td>float</td>
<td>10</td>
<td>Not null</td>
<td>Special marks given to each subject</td>
</tr>
</tbody>
</table>

### Table Name: Remuneration

For entering the details of the remuneration given to the evaluators.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staff_id</td>
<td>Primary key</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### 5. Table Name: Staff
For entering the staff details.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staff_id</td>
<td>Int</td>
<td>11</td>
<td>Primary key</td>
<td>ID no. given to the faculty.</td>
</tr>
<tr>
<td>Staff_name</td>
<td>Varchar</td>
<td>11</td>
<td>Not null</td>
<td>Name of the faculty.</td>
</tr>
<tr>
<td>SCollege</td>
<td>Varchar</td>
<td>11</td>
<td>Not null</td>
<td>College of the staff</td>
</tr>
<tr>
<td>Designation</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Designation of the staff.</td>
</tr>
<tr>
<td>Email_id</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Email-id of the faculty.</td>
</tr>
<tr>
<td>Experience</td>
<td>Int</td>
<td>10</td>
<td>Not null</td>
<td>Experience of the faculty.</td>
</tr>
</tbody>
</table>

### 6. Table Name: Student
For entering the details of the student.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reg_no</td>
<td>Int</td>
<td>11</td>
<td>Primary key</td>
<td>ID no. given to the student.</td>
</tr>
<tr>
<td>Name</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Name of the student.</td>
</tr>
<tr>
<td>Father_name</td>
<td>Varchar</td>
<td>15</td>
<td>Not null</td>
<td>Father's name.</td>
</tr>
<tr>
<td>Mother_name</td>
<td>Varchar</td>
<td>15</td>
<td>Not null</td>
<td>Mother's name.</td>
</tr>
<tr>
<td>Address</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Address of the student.</td>
</tr>
<tr>
<td>Phone_number</td>
<td>Varchar</td>
<td>11</td>
<td>Not null</td>
<td>Phone number of the student.</td>
</tr>
<tr>
<td>Email_id</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Email-id of the student.</td>
</tr>
<tr>
<td>Student_id</td>
<td>Int</td>
<td>11</td>
<td>Primary key</td>
<td>ID no. given to the student.</td>
</tr>
</tbody>
</table>
1. **Table Name: Student**  
   For entering the details of a student
   
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reg_no</td>
<td>Int</td>
<td>11</td>
<td>Primary key</td>
<td>Registration no. of a student</td>
</tr>
<tr>
<td>Name</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Name of the student.</td>
</tr>
<tr>
<td>Year_of_admission</td>
<td>Varchar</td>
<td>21</td>
<td>Not null</td>
<td>Admission year of the student.</td>
</tr>
<tr>
<td>College</td>
<td>Varchar</td>
<td>40</td>
<td>Not null</td>
<td>Student’s college.</td>
</tr>
<tr>
<td>Regular</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Whether the student is a regular candidate.</td>
</tr>
<tr>
<td>Course</td>
<td>Varchar</td>
<td>10</td>
<td>Not null</td>
<td>Course of the student.</td>
</tr>
<tr>
<td>Branch</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Branch of the student.</td>
</tr>
</tbody>
</table>

7. **Table Name: Syllabus**  
   For entering the details of the course subjects.
   
   Year, Code_Subject : Primary key
   
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Course of the student.</td>
</tr>
<tr>
<td>Branch</td>
<td>Varchar</td>
<td>41</td>
<td>Not null</td>
<td>Branch of the student.</td>
</tr>
<tr>
<td>Semester</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Semester for the subjects.</td>
</tr>
<tr>
<td>Year</td>
<td>Varchar</td>
<td>10</td>
<td>Primary key</td>
<td>Year for the subjects.</td>
</tr>
<tr>
<td>Code_Subject</td>
<td>Varchar</td>
<td>40</td>
<td>Primary key</td>
<td>Code and subject.</td>
</tr>
<tr>
<td>Credits</td>
<td>Varchar</td>
<td>40</td>
<td>Not null</td>
<td>Credits of each subject.</td>
</tr>
</tbody>
</table>

8. **Table Name: Valuation_marks**  
   For entering the valuation marks.
   
   Reg_no, Code_Subject : Primary key
   
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Course of the student.</td>
</tr>
<tr>
<td>Branch</td>
<td>Varchar</td>
<td>41</td>
<td>Not null</td>
<td>Branch of the student.</td>
</tr>
<tr>
<td>Semester</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Semester for the subjects.</td>
</tr>
<tr>
<td>Year</td>
<td>Varchar</td>
<td>10</td>
<td>Primary key</td>
<td>Year for the subjects.</td>
</tr>
<tr>
<td>Code_Subject</td>
<td>Varchar</td>
<td>40</td>
<td>Primary key</td>
<td>Code and subject.</td>
</tr>
<tr>
<td>Credits</td>
<td>Varchar</td>
<td>40</td>
<td>Not null</td>
<td>Credits of each subject.</td>
</tr>
<tr>
<td>Field Name</td>
<td>Data Type</td>
<td>Length</td>
<td>Constraints</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------</td>
<td>--------</td>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Reg_no</td>
<td>int</td>
<td>11</td>
<td>Primary key</td>
<td>Registration no. of a student</td>
</tr>
<tr>
<td>Code_Subject</td>
<td>varchar</td>
<td>40</td>
<td>Primary key</td>
<td>Code and subject.</td>
</tr>
<tr>
<td>Attended_Exam</td>
<td>varchar</td>
<td>10</td>
<td>Not null</td>
<td>Whether he had attended the exam.</td>
</tr>
<tr>
<td>Internal_mark</td>
<td>double</td>
<td>10</td>
<td>Not null</td>
<td>Internal mark for a subject.</td>
</tr>
<tr>
<td>First_valuation_mark</td>
<td>double</td>
<td>10</td>
<td>Not null</td>
<td>First valuation mark.</td>
</tr>
<tr>
<td>first_evaluators_id</td>
<td>int</td>
<td>10</td>
<td>Not null</td>
<td>Id of the first evaluator.</td>
</tr>
<tr>
<td>Second_valuation_mark</td>
<td>double</td>
<td>20</td>
<td>Not null</td>
<td>Second valuation mark.</td>
</tr>
<tr>
<td>Third_valuation_mark</td>
<td>double</td>
<td>10</td>
<td>Not null</td>
<td>Third valuation mark.</td>
</tr>
<tr>
<td>Third_evaluators_id</td>
<td>int</td>
<td>20</td>
<td>Not null</td>
<td>Id of the third evaluator.</td>
</tr>
<tr>
<td>External_mark</td>
<td>double</td>
<td>10</td>
<td>Not null</td>
<td>Final external mark.</td>
</tr>
<tr>
<td>Final_mark</td>
<td>double</td>
<td>10</td>
<td>Not null</td>
<td>Total mark.</td>
</tr>
<tr>
<td>Semester</td>
<td>Varchar</td>
<td>20</td>
<td>Not null</td>
<td>Semester for the subjects.</td>
</tr>
<tr>
<td>moderation_given</td>
<td>char</td>
<td>3</td>
<td>Not null</td>
<td>Moderation given to each subject.</td>
</tr>
</tbody>
</table>

9. Table Name: To Do List
   For entering each day tasks for the evaluators.
   Date: Primary key

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>Primary key</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field Name</td>
<td>Data Type</td>
<td>Length</td>
<td>Constraints</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>--------</td>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Todolist</td>
<td>Varchar</td>
<td>350</td>
<td>Not null</td>
<td>List of works for camp evaluators.</td>
</tr>
<tr>
<td>Date</td>
<td>Date</td>
<td>10</td>
<td>Primary key</td>
<td>Date of the particular to do list.</td>
</tr>
</tbody>
</table>

**10. Table Name: Login**
For entering username and password.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Varchar</td>
<td>13</td>
<td>Not null</td>
<td>For username.</td>
</tr>
<tr>
<td>Password</td>
<td>Varchar</td>
<td>13</td>
<td>Not null</td>
<td>For password..</td>
</tr>
</tbody>
</table>

**11. Table Name: Camp_attendance**
For entering the camp attendance of the evaluators.
Staff_id, Date : Primary key

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Length</th>
<th>Constraints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staff_id</td>
<td>Int</td>
<td>11</td>
<td>Primary key, foreign key</td>
<td>ID no. given to the faculty.</td>
</tr>
<tr>
<td>Date</td>
<td>Date</td>
<td>10</td>
<td>Primary key, foreign key</td>
<td>Date of the particular to do list.</td>
</tr>
<tr>
<td>attendance</td>
<td>int</td>
<td>10</td>
<td>Not null</td>
<td>For entering each days attendance</td>
</tr>
</tbody>
</table>

**7.5 DATA FLOW DIAGRAM**
A data flow diagram (DFD) is a structured analysis and design tool that can be used for flow charting in place, or in association with, information oriented and process-oriented system flowcharts. A DFD is a network that describes the flow of data and the processes that change, or transform data throughout a system. This network is constructed by using a set of symbols. They
Dataflow diagrams are divided into two. Basic DFD symbols are described below:

- **A directed line** represents a data flow, that is, a data stream.
- **Circle, or a “bubble”,** represents a process that transforms data streams.
- **This rectangle** represents a data store.
- **This rectangle** represents a data source or destination.

Data dictionary refers to the picture of the organization data. Data dictionary can be a manual file describing data items or a computerized system.

The complete set of data definitions can provide the basis of the data dictionary. It facilitates cross-referencing and assessment of the implications of changes. Once data items have been defined at the lowest level then the data structure required by the particular system under consideration can be identified.

The terms normally sued to define structure are:

- **Group Items**: Combinations of data items.
- **Records**: Collection of data items or group items.
- **Files**: Collection of records that are gathered together to allow processing of data about several entities.
- **Data Base**: Collection of records that are inter-related for purposes of processing to meet user requirements.

Dataflow diagrams are divided into two.

1. **Logical DFD**
2. **Physical DFD**

**Logical DFD** is an implementation independent view of a system, focusing on the flow of data between processes without regard for the specific devices, storage locations or people in the system. In logical DFD the physical characteristics will not be listed.

**Physical DFD** is an implementation dependant view of the current system, showing what tasks are carried out and how they are performed. Physical characteristics include names of people form and document names or numbers, names of departments, master and transaction file locations and names of procedures.
Context Level:

User -> Evaluation System -> User
Level 1:

1. Generate Eligibility List
   - Hall ticket
   - ADM office file
   - Mark list, Hall ticket

2. Valuation System
   - Attendance
   - Papers
   - Internal marks
   - Marks

3. Record Keeping

Level 2:

1. Providing hall ticket
Level 3:

1. Inform Panel of evaluators
2. Eligibility list creation

3. Prepare Remuneration

4. Recording details

4.1 Providing mark list & certificates

ADM Office file

User

1.2. Prepare To Do list

3.1 1st, 2nd, 3rd valuations

4. Conducting Passing Board

To Do list

User
8. SYSTEM TESTING
**Testing, Training and Documentation**

Testing is the stage of implementing, which is aimed at earning system running accurately and efficiently. The purpose of the system testing is to identify and correct errors in the new system. The performance factors like turnaround time, back up, file protection and human factors are some of the performance criteria for system testing. A system is tested for online response, volume of transactions, recovery from failure and usability.

Effective testing early in the process translates directly into long-term cost savings from a reduced number of errors. Back up files are need when the system is failure or down. The usability test verifies the user-friendly nature of the system. Accurate and complete documentation is necessary for the user-friendly nature of the system.

System testing is designed to uncover weaknesses that are not found in the earlier tests. This includes forced system failure and validation of the total system, as its users in the operational environment will implement it. Generally it begins with low volume of transactions based on live data. The volume is increased until the maximum level for each transaction type is reached. The total system is tested for recovery and fallback after various major failures to ensure that no data are lost during the emergency. All this is done with the old system still in operation. After the candidate system passes the test, the old system is discontinued.

System testing involves unit testing, integration testing, acceptance testing. Careful planning and scheduling are required to ensure that modules will be available for integration into the evolving software product when needed. A test plan has the following steps:

- Prepare test plan
- Specify conditions for user acceptance testing
- Prepare test data for program testing
- Prepare test data for transaction path testing
- Plan user training
- Compile/assemble programs
- Prepare job performance aids.
System testing is the stage of implementation that is aimed at ensuring that the system works accurately and efficiently before live operation commences. The system on a whole was tested for the following:

- Validation of inputs
- Referential integrity test
- Sequential tests
- Consistency of the application

System testing, asks a logical assumption that if all the parts of the system are correct, the system will be successfully achieved. The objective of testing is to discover errors. To fulfill these objectives a series of tests were planned and executed.

The logical design and the physical design should be thoroughly and continually examined on paper to ensure that they will work when implementation should be a confirmation that all is correct and an opportunity to show the users that the system works.

### 8.1 Unit Testing

Here, each individual program was tested using the test data. The outputs as per the requirements were found satisfactory. Thus it was possible to conclude that every program in the software was functionally correct. The interrelated modules were also tested in an exhaustive that will make the whole software work properly.

- Module interface is tested to ensure that information properly flows into and put of the program under test.
- Local data structures are examined to ensure that data stored temporarily maintains its integrity during all steps in algorithm execution.
- Boundary condition is tested to ensure that the module operates properly at boundaries established to limit or restrict processing.

- All independent paths through the control structures are executed to ensure that all statements in the module have been executed at least once.
- Error handling paths are also tested.

This test focuses verification effort on the smallest unit of software design, the module. Here, the module interfaces, local data structure, boundary conditions, and all independent paths and last but not the least, all error handling paths were verified by
inputting false data. Tests of data flow across each module interface of this software were done before any other test was initiated.

A **unit testing** focuses on the verification effort on the smallest unit of the software design. Using the unit test plan, prepared in the design phase of the system, important control paths had tested to uncover the errors within these modules. This testing was carried out doing the coding itself. In this testing step, each module is going to be working satisfactorily as the expected output from the module.

### 8.2 Integrated Testing

The individual programs are combined together to form modules. Integrated tests were performed on each of the modules and again the validity was checked. After that, all modules were brought under a single module and the integrity test was found to be successful.

This system was validated in such a way that even the slightest deviation in inputting the data will invoke error messages and provide guidelines regarding the input. Before the software is being released, the developers do testing by implementing the commercial security package for security. This ensures that the software works properly. These tests can also be performed

- Top down integration
- Bottom up integration

It is systematic technical for constructing the program structure while at the same time conducting test to uncover errors associated with the interface. The objective is to take unit tested module and build the program structure that has been detected by design.

All modules are combined in this testing step, and the entire program is tested as a whole. If a set of errors are encountered correction is difficult because the isolation of causes is complicated by vastness of the entire program.

Using integrated system test plan prepared in the design phase of the system developed as a guide, he integration was carried up.
8.3 Validation Testing

Validation testing is done to ensure complete assembly of the error-free software. Validation can be termed successful only if it functions in manner that is reasonably expected by the customer.

Under validation is alpha and beta testing. Alpha testing is where the end user tests the system rather than the developer, but in a controlled environment. The software is used on a natural setting with the developer monitoring the user using the system. The developer records the errors and usage problems encountered by the user.

The sales person conducts beta testing at one more sites. The developer is not present during these tests. Hence, beta test can be said as the live application of the software on an environment that cannot be controlled by the developer.

The sales person takes down the problems encountered during beta testing and reports these to the developer at regular intervals. The developer makes suitable modifications to the software henceforth.

The first step in system testing is to develop a plan that tests all the aspects of the system. Completeness, correctness, reliability and maintainability of the software are to be tested for the best quality assurance-an assurance that the system meets the specification and requirements for its intended user and performance. System testing is most useful practical process of executing a program with explicit intention of finding errors that makes the program fail. The following phases were developed.

8.4 Module Testing

Each individual programs module is tested for any possible errors. They were also tested for specifications, i.e. to see whether they are working as per what the program should do and how it should perform under various conditions.

Concurrence Testing

Since the system is a multi-user it was tested for concurrence problems. The system worked perfectly since the table locking and other security measures were taken with care by the database itself.

Login Testing
The login process was tested with both authorization and unauthorized login. Access was denied for invalid login ids and incorrect passwords. Passwords were changed and addition and deletions of users were done. The results were as expected.

**Insert /Update Testing**

The insertion and updation were tired on all tables in the database. Checking was done to see whether the corresponding entries were made in the different tables when a new record was created. Updation of non-existing records and duplicate values were tired. The system was found to check and disallow invalid attempts.

**Display Testing**

The display procedures were tested since the displayed is of much importance. The data was input in the different modules and it was checked whether the information is properly displayed in the other dependent modules. The consistency of the display and attractiveness of the display were also tested. The following tests were also conducted over the system developed:

- **Integration:** These test the integration between browsers and servers, applications and data, hardware and software.
- **Usability:** These test the overall usability of a web page or a web application, including appearance clarity and navigation.
- **Security:** These test the adequacy and correctness of security controls including access control and authorizations.
- **Performance:** These test the performance of the web applications under load.
- **Verification of code:** This validate that the code used in building the web application has been used in a correct manner.

Comments and suggestions from the observations during the test run were later considered. Special care was given to user interface comments. The list of possible values, the attractiveness and user-friendliness of the software was well appreciated.

**User Training**

A well-designed system, if not operated and used properly could fail. Training the users is important, as if not done well enough could prevent the successful implementation of an information system. Through the systems development life cycle the user has been involved. By this stage the analyst should possess an accurate idea of
the users they need to be trained. They must know what their roles will be, how they can use the system and what the system will do and will not do. Both system operators and users need training. During their training, they need to be given a trouble-shooting list that identifies possible problems and identifies remedies for the problem. They should be advised of the common malfunctions that may arise and how to solve them.

**Operational Documentation**

Once the implementation plan is decided, it is essential that the user of the system is made familiar and comfortable with the environment. Education involves right atmosphere & motivating the user. A documentation providing the whole operations of the system is being developed.

The system is developed in such a way that the user can work with it in a well consistent way. The system is developed user friendly so that the user can work the system from the tips given in the application itself. Useful tips and guidance is given inside the application itself to help the user. Users have to be made aware that what can be achieved with the new system and how it increases the performance of the system.
SYSTEM IMPLEMENTATION

9. SYSTEM IMPLEMENTATION
IMPLEMENTATION

Implementation is the process of converting a new or a revised system design into an operational one. Conversion means changing from one system to another. The objective is to put the system into operation while holding costs, risks and personnel irritation to minimum. It involves

- Creating computer compatible files.
- Training the operating staff.
- Installing terminals and hardware.

A critical aspect of conversion is not disrupting the functioning of the organization. The implementation plan is a function of line management at least as far as key decisions or alternative plans are concerned. The implementation plan was to convert the existing clerical files to the computer. The implementation plan listed all sub tasks so that individuals in the organization may be assigned specific responsibilities.

The installation of the new system that is bound to replace the current one may require a major revision of computer facilities swell as completely new after space. Space planning took in to account the space occupied by the people, space by equipment and the moment of people and equipment in the working investment. After conducting, the initial testing the system is loaded in the client officer’s computer. Some of the user employees in this case are selected. These users are trained first and they run the system.

MAINTANANCE

This phase occurs as a result of deploying the whole system at the end users organization. They will perform the beta testing at the end users and inform to the developers about any needed modification to the application .the customer records all the problems that are encountered during the beta testing and reports these to the developer at regular intervals.

As result of problems reported during implementation, the software product to the entire customer base

Types of changes that can be encountered during the maintenance phase

Corrective maintenance:
Even with the best quality assurance activities, it is likely that the customer will uncover defects in the software. Corrective maintenance changes the software to correct the defects.

**Adaptive maintenance:**
Over time, the original environment (CPU, operating system, business rules, external product characteristics) for which the software was developed is likely to change. Adaptive maintenance results in modification to the software to accommodate changes to its external environment.

**Enhancement maintenance:**
As software is used, the customer/user will recognize additional functions that will provide the benefit. Perfect maintenance extends the software beyond its original functional requirements.

**Preventive maintenance:**
Computer software deteriorates due to change, and because of this preventive maintenance often called software reengineering, must be conducted to enable the software to serve the needs of its end users. Preventive maintenance makes changes to computer programs so that they can be more easily corrected, adapted and enhanced.
10. MODULAR DESCRIPTION
Those modules of the entire project which I have exercised

Module2
   1 Conduct a valuation camp, details of the 1\textsuperscript{st}, 2\textsuperscript{nd} & 3\textsuperscript{rd} valuation.
   2 Remuneration.
   3 After 3 valuations recording the marks, details, etc.

1) **Conducting 1\textsuperscript{st}, 2\textsuperscript{nd} and 3\textsuperscript{rd} valuation**

We get the details of the student and staff from the department. The first valuation is mainly done by corresponding faculty of corresponding college. First valuation can do by both guest and permanent faculty. After the first valuation the papers are gone for the second valuation.

In the second valuation camp is conducted. The staff’s are informed about the camp through mails. During camp, there will be attendance taken. According to this attendance and the number of papers remuneration is calculated. There is a rate for each paper. There is a To Do List for the camp evaluators. The teachers from same institute are not allowed to check the papers from their institution. Also 1\textsuperscript{st} valuator is not allowed check paper in 2\textsuperscript{nd} valuation.

If there is a mark difference between 1\textsuperscript{st} and 2\textsuperscript{nd} valuation then the papers go for third valuation. For MCA it is a 5 marks difference and MBA 7 marks.

2) **Remuneration**

Remuneration is given to the valuators according to the number of papers checked. It is to be noted that when the remuneration is given it is marked or not.

3) **Recording the details**

In recording details the passing board details are added to the valuation marks. According to the each passing board, the rules are entered and this marks is added to the external marks.
1) LOGIN PAGE
ISEM

AN INTEGRATED SOLUTION FOR EXAMINATION MANAGEMENT

LOGIN ID

PASSWORD

LOGIN  RESET

2) MAIN PAGE
3) ENTERING REMUNERATION DETAILS
4) TO GET REMUNERATION DETAILS

[Image of a computer screen displaying a form titled "REMUNERATION DETAILS" with fields for Staff_ID and Date, and buttons for Back and Search]
5) REMUNERATION DETAILS

6) FOR ENTERING THE STAFF DETAILS
7) FOR ENTERING STUDENT DETAILS
8) FOR ENTERING THE COURSE SUBJECTS
9) FOR ENTERING THE VALUATION DETAILS
10) FOR ENTERING THE CAMP DETAILS
12. Future Scope

In this system we can also add more features for providing the certificates of other graduation and post graduation courses. It can be applied for any other
Universities. Also official transcripts of marks with photographs of all candidates are made available on the Website within seconds after the announcement of final semester results. The manual system took nearly two months to prepare an official transcript of a candidate. Similarly, mark lists corresponding to each semester are also made available on the internet in a printable format immediately after the respective results are announced.

It can also make more useful for companies, who are coming for recruitments. Recruiting agencies can verify the genuineness of degrees through Internet. They need enter only Register Number (available on the degree certificates) to get the name, photograph, date of birth, semester wise mark details, whether the candidate had failed in any exam, percentage score in each paper etc. on-line.
13. CONCLUSION

The purpose of “ISEM” is to provide an Examination Evaluation in a University. This will manages the examinations valuation for the Post Graduation courses MCA and MBA. It provide hall ticket, arranges the answer papers for 1st, 2nd & 3rd
valuation, process marks, publish results and provide mark lists and certificates for the University. By using this system we can search the marks and details of any student of any year. This is more relevant in the case of Universities which have large number of colleges affiliated to them. The same format of application form can be used by candidates appearing for all the examinations under this University. Cost of printing, maintaining and distributing different types of forms saved. This will make the communication between candidates and university office effective.

Hall Tickets of about 5000 candidates can be generated with a single day's data entry work in all semesters/years of a Course except the first. It has a way for calculating the remuneration for the evaluators. It will also show the summary of each valuation camp and the passing board.
BIBLIOGRAPHY

References

- Analysis & Design of Information System
  Author: James A. Senn
- Database System Concepts- Abraham Silberschatz
- Software Engineering- Ian Sommerville
• Software Engineering concepts-Richard Fairley
• System Analysis And Design-Elias Awad
• Core Java-Cay S.Horstmann,Gary Cornell

Websites

THANK YOU