**VIRTUAL CLASSROOM**

**INTRODUCTION**

The virtual classroom is an innovative method of teaching used by the teachers to create an interactive platform to teach students remotely. Here the communication is performed in live video and live audio. This method of teaching has lots of advantages than the traditional classroom method. Nowadays due to the advancement in technologies like smartphones, high-speed internet and the cost-efficient internet packs paved the way for greater innovation such as virtual classrooms, virtual office meetings, etc. One of the greatest instructional benefits of video conferencing includes improved communication skills, and presentation skills among students. Instead of just reading textbooks and other printed materials. In Virtual Classroom System the students and teacher need not attend the classes physically, the teaching sessions are carried out through online. The clients were provided with Login ID and passwords at the time of registration that helps them to get connected to the server. Here the student has the freedom of choosing the class he wants to attend. in addition, we also add college management thing in the application. With this application, students can be informed about the college notice, exam, time table and anything else. Teachers can upload study material and Students can also download and use it. This application can develop the bond between the students and the teacher.

**2.1 SYSTEM ANALYSIS**

System analysis is a general term that refers to an orderly, structured process for identify and solving a problem. The system analysis process is calling the life cycle methodology, since it relates to four significant phases in the life cycle of all business information system: study, design, development and operation. The definition of system analysis includes not only the process but also the process of putting together to form a new system. A system analyst is an individual who performs system analysis during any, or all, of the life cycle phases of a business information system. The system analyst not analyses business information system problems, but also synthesizes new to solve those problem or to meet other information needs.

The various techniques used in the study of the present system are:

* Observation
* Interviews
* Site visits
* Discussion

**EXISTING SYSTEM**

Virtual classroom is a teaching and e-learning platform where participants can interact, communicate, view and discuss presentations, and engage with learning resources while working in groups, all in an online environment.

The medium is often through a video conferencing application that allows multiple users to be connected at the same time through the Internet either connecting all nodes together or connecting everyone with one central node, i.e. allowing users from virtually anywhere to participate.

**PROPOSED SYSTEM**

We are introducing a system which can also manage students, share materials and also account for students participents in various classes. Students can also access materials any time they want without time restriction. So, students from across world, from diffrent occupations can also participate in also same classes.In Virtual Classroom System the students and teacher need not attend the classes physically, the teaching sessions are carried out through online. The clients were provided with Login ID and passwords at the time of registration that helps them to get connected to the server. Here the student has the freedom of choosing the class he wants to attend. in addition, we also add college management thing in the application. With this application, students can be informed about the college notice, exam, time table and anything else. Teachers can upload study material and Students can also download and use it. This application can develop the bond between the students and the teacher.

**2.5 System Requirement Specification**

Requirement analysis is a software engineering task that bridges the gap between system level software designs. We have done the requirement analysis in order to understand the problem faced in our objectives. The emphasis in requirement analysis is on identifying from the system, not how the system will achieve this goal

**2.6 Hardware and Software specification**

**Hardware Specification**

The selection of hardware is very important in the existence and proper working of any of the software. When selecting hardware, the size and capacity requirements are also important. The hardware must suit all application developments**.**

* Processor : i3 or above.
* System Bus : 32Bit or 64Bit
* RAM : 4 GB or Above
* HDD : 500 GB or Above
* Monitor : 14” LCD or Above
* Key Board : 108 Keys
* Mouse : Any Type of mouse
* Mobile : Android supported mobile phone

**Software specification**

One of the most difficult tasks is selecting software, once the system requirement is find out then we have to determine whether a particular software package fits for those system requirements. This section summarizes the application requirement.

* Operating System : Windows 10 Any 32 bit or 64 bit platform
* Front End : Android, Python
* Back End : MySQL Sever
* IDE : Eclipse or Android studio

: Python 3.6 or above

: PyCharm

. You can also get reports about customer services and efficiency of our attendees.

**4.1 MODULE DESCRIPTION**

**ADMIN**

* Login
* Manage HOD
* View teachers
* View students
* Post notices
* Register departments

**HOD**

* Register
* Login
* Register and Manage teachers
* Manage students
* Post department information
* Set time table

**TEACHERS**

* Login
* Add internal exam details
* View students
* Upload study material
* View time table
* Host video Conferencing
* Share meeting link

**STUDENTS**

* Register
* Login
* View department information
* View time table
* Download study materials
* View notices
* View exam details
* View internal exam details
* View meeting link and join

**2.7 Feasibility Study**

Feasibility is defined as the practical extent to which a project can be performed successfully. The objective of feasibility study is to establish the reasons for developing the software that is acceptable to the users, adaptable to changes and conformable to the established standards. Various types of feasibility that are commonly considered include:

* Economic feasibility
* Technical feasibility
* Operational feasibility
* Behavioral feasibility

**Economic Feasibility**

Economic feasibility determines whether the proposed system is capable of generating financial gains for an organization. It involves cost incurred on the software development team, estimated cost of hardware and cost of performing feasibility study and so on. The proposed system is economically feasible since the cost incurred for the development of the system produces long term gains. The cost of hardware and software for the class of applications of Advanced ATM system is less. Hence the proposed system is economically feasible.

**Technical Feasibility**

Technical feasibility assess the current resources (includes hardware and software) and technology which are required to accomplish the user requirements in the system within the allocated time and budget. It is concerned with the existing computer system (hardware and software) and to what extent it can support the proposed system. The proposed system requires .net platform only.

**Operational Feasibility**

The proposed project would be beneficial to organization satisfying the objectives when developed and installed. One of the main problems faced during development of a new system is getting the acceptance from the user. There is support from the management of organization towards the development of the project. All the operational aspects are considered carefully. Thus the project is operationally feasible.

**Behavioral Feasibility**

In behavioral feasibility the management considers that the proposed system will fulfil the requirements of the proposed system that is whether the proposed system covers all the jobs that where done manually and whether it has considerable improvements. Understanding theadvantages and efficiency of the proposed system the management has decided to develop a registration purpose.

**2.7. FRONT END**

**PYTHON**

[Python](https://www.geeksforgeeks.org/python-programming-language/) is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.

Python is a programming language that lets you work quickly and integrate systems more efficiently.

There are two major Python versions- **Python 2 and Python 3**. Both are quite different.

**HTML**

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

**PyCharm**

**PyCharm** is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [computer programming](https://en.wikipedia.org/wiki/Computer_programming), specifically for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) language. It is developed by the Czech company [JetBrains](https://en.wikipedia.org/wiki/JetBrains).[[6]](https://en.wikipedia.org/wiki/PyCharm) It provides code analysis, a graphical debugger, an integrated unit tester, integration with [version control systems](https://en.wikipedia.org/wiki/Revision_control) (VCSes), and supports web development with [Django](https://en.wikipedia.org/wiki/Django_(web_framework)) as well as [Data Science](https://en.wikipedia.org/wiki/Data_science) with [Anaconda](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)).[[7]](https://en.wikipedia.org/wiki/PyCharm)

PyCharm is [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), with [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux) versions. The Community Edition is released under the [Apache License](https://en.wikipedia.org/wiki/Apache_License),[[8]](https://en.wikipedia.org/wiki/PyCharm) and there is also Professional Edition with extra features – released under a [proprietary license](https://en.wikipedia.org/wiki/Proprietary_software).

**Features**

* Coding assistance and [analysis](https://en.wikipedia.org/wiki/Code_analysis), with [code completion](https://en.wikipedia.org/wiki/Autocomplete), syntax and error highlighting, [linter integration](https://en.wikipedia.org/wiki/Lint_(software)), and quick fixes
* Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages
* Python [refactoring](https://en.wikipedia.org/wiki/Refactoring): includes rename, extract method, introduce variable, introduce constant, pull up, push down and others
* Support for web frameworks: [Django](https://en.wikipedia.org/wiki/Django_(web_framework)), [web2py](https://en.wikipedia.org/wiki/Web2py) and [Flask](https://en.wikipedia.org/wiki/Flask_(web_framework)) [professional editional only]
* Integrated Python [debugger](https://en.wikipedia.org/wiki/Debugger)
* Integrated [unit testing](https://en.wikipedia.org/wiki/Unit_testing), with line-by-line [code coverage](https://en.wikipedia.org/wiki/Code_coverage)
* [Google App Engine](https://en.wikipedia.org/wiki/Google_App_Engine) Python development [professional edition only]
* Version control integration: unified user interface for [Mercurial](https://en.wikipedia.org/wiki/Mercurial), [Git](https://en.wikipedia.org/wiki/Git_(software)), [Subversion](https://en.wikipedia.org/wiki/Apache_Subversion), [Perforce](https://en.wikipedia.org/wiki/Perforce) and [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) with change lists and merge
* Support for scientific tools like matplotlib, numpy and scipy [professional edition only]

It competes mainly with a number of other Python-oriented IDEs, including [Eclipse](https://en.wikipedia.org/wiki/Eclipse_(software))'s PyDev, and the more broadly focused [Komodo IDE](https://en.wikipedia.org/wiki/Komodo_IDE).

**ANDROID**

The android system supports background processing, provides a rich user interface library, supports 2-D and 3-D graphics using the OpenGL-ES (short OpenGL) standard and grants access to the file system as well as an embedded SQLite database.

Android Software Development Kit (Android SDK) contains the necessary tools to create, compile and package Android applications. Most of these tools are command line based. The primary way to develop Android application is based on Java programming language.

Most of Android‘s configuration files are based on XML. In this case these editors allow you to switch between the XML representation of file and a structured user interface for entering the data.

**Android Eclipse**

A complete set of tools for developers who want to create Eclipse plug-ins or Rich Client Applications. It includes a complete SDK, developer tools and source code, an XML editor and the Eclipse Communication Framework.

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming through the use of plugins,

**Features**

* Org .eclipse .cvs
* Org.eclipse.eqinox.p2.user.ui
* Org .eclipse. help
* Org .eclipse .jdt
* Org .eclipse .pde
* Org .eclipse .platform
* Org .eclipse .rcp

**Eclipse IDE**

Google provides an integrated development environment (IDEs) to develop new applications. The *Android Developer Tools* (ADT) are based on the Eclipse IDE. ADT is a set of components (plug-ins), which extend the Eclipse IDE with Android development capabilities. Eclipse is an integrated development environment (IDE).it contains a base workspace and an extensible plug-in system for customizing the environment. It is written mostly in java. Eclipse can be used to develop applications. Eclipse sometimes performs multiple commands within a single connection to the server. This may cause problems with servers that are servers that are running server scripts in response to certain commands.Eclipse IDE contains all required functionality to create, compile, debug and deploy Android applications. This also allows the developer to create and start virtual Android devices for testing. Both tools provide specialized editors for Android specific files. Most of Android's configuration files are based on XML. In this case these editors allow you to switch between the XML representation of the file and a structured user interface for entering the data. Eclipse uses plug-ins to provide all the functionality within and on top of the runtime system.

The plug-in architecture supports writing any desired extension to the environment, such as for configuration management. Java and CVS support is provided in the Eclipse SDK, with support for other version control systems provided by third-party plug-ins. The Eclipse SDK includes the Eclipse Java development tools (JDT), offering an IDE with a built-in incremental Java compiler and a full model of the Java source files. This allows for advanced refactoring techniques and code analysis. The IDE also makes use of a workspace, in this case a set of metadata over a flat filespace allowing external file modifications as long as the corresponding workspace "resource" is refreshed afterwards. Eclipse implements use the graphical control elements of the Java toolkit called SWT, whereas most Java applications use the Java standard Abstract Window Toolkit (AWT) or Swing. Eclipse's user interface also uses an intermediate graphical user interface layer called JFace, which simplifies the construction of applications based on SWT. Eclipse was made to run on Wayland during a GSoC-Project in 2014.

**2.8.2 ABOUT THE BACK END**

**Database Severs** A database server is used to store data in a database. Users can access the data and manipulate it. There are many type of databases. The most popular among them is the Relational Database Management

System (RDBMS).

**RDBMS**

RDBMS is a type of database management system that stores data in the form of related tables. Relational database are powerful because they require few assumptions about how data is related or how it will be extracted from the database. As a result, the same database can be viewed in many different ways. An important feature of relational systems is that a single database can be spread across several tables. This differs from flat-file database, in which each database is self-contained in a single table.

**My SQL**

MySQL is an open source relational database and it includes advanced data types. MySQL operates using client/server architecture in which the server runs on the machine containing the database and client connect to the server over the network. MySQL run on all platforms supported by MySQL and provides the most direct means of interacting with the server, so it’s the logical client to begin with.

* You need to have the MySQL software installed.
* You need a MySQL account so that you can connect to the server.
* You need a database to work with.

The required software includes the MySQL clients and a MySQL clients and a MySQL server. The client program must be located on the machine where you will work. The server can be located on our machine although that is not required. As long as you have permission to connect to it the server can be located anywhere. In addition to the MySQL software you will need a MySQL account so that the server will allow you to connect and create us sample database and its table.

Microsoft SQL Server 2008 is a full-featured relational database management system (RDBMS) that offers a variety of administrative tools to ease the burdens of database development, maintenance and administration. In this article, we’ll cover six of the more frequently used tool: Enterprise Manager, Query analyzer, SQL Profiler, Service Manager, Data Transformation Services and Books Online. Let’s take a brief look at each:

**Enterprise Manager** is the main administrative console for SQL Server installations. It provides you with a graphical “birds-eye” view of all of the SQL Server installation on your network. You can perform high-level administrative functions that affect one or more servers, schedule common maintenance tasks or create and modify the structure of individual databases.

**Query Analyzer**offers a quick method for performing queries against any of your SQL Server databases. It’s a great way to quickly pull information out of a database in response to a user request, test queries before implementing them in other applications, create/modify stored procedures and execute administrative tasks.

**SQL Profiler** provides a window into the inner workings of your database. You can monitor many different event types and observe database performance in real time. SQL Profiler allows you to capture and replay system “traces” that log various activities. It’s a great tool for optimizing databases with performance issues or troubleshooting particular problems.

**Service Manager** is used to control the MS SQL Server (the main SQL Server process), MSDTC (Microsoft Distributed Transaction Coordinator) and SQLSeverAgent processes. An icon for this service Manager to start, stop or pause any one of these services.

**Data Transformation Services (DTS)** provide an extremely flexible method for importing and exporting data between a Microsoft SQL Server installation and a large variety of other formats. The most commonly used DTS application is the “Import and Export Data” wizard found in the SQL Server program group.

**2.8 REQUIREMENTS MODELING**

**2.8.1 Data Flow diagram**

Data flow diagram is used to define the flow of the system and resources such as information. Data flow diagram are a way of expressing system requirement in a graphical manner. DFD represents one of the most ingenious tools used for structured analysis. DFD is also known as bubble chart. It has the purpose of clarifying system Requirements and identifying major transformations that will become programs in system design.

In the normal convention, logical DFD can be completed using only 4 notations

* To construct a dataflow diagram, we use
* Squares representing external entities, which are sources or destinations of data.
* Rounded rectangles representing processes, which take data as input, do something to it, and output it.
* Arrows representing the data flows which can either, be electronic data or physical items.
* Open-ended rectangle representing data store including electronic store such as database or XML files and physical stores such as or filling cabinets or stack of paper.

**PROCESS :**

A process transforms incoming data flow into outgoing data flow.

**DATA STORE :**

Data store are repositories of data in the system. They are sometimes also referred to as files.

**DATA FLOW :**

Dataflow are pipelines through which packets of information flow. Label the arrow with the name of the data that moves through it.

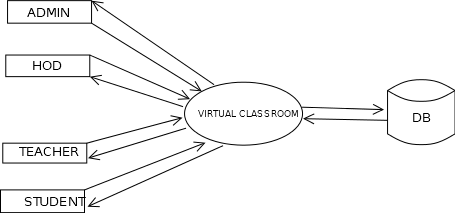
**EXTERNAL ENTITY :**

External entities are objects outside the system with which the system communicates. External entities are sources and destinations of the systems inputs and outputs.

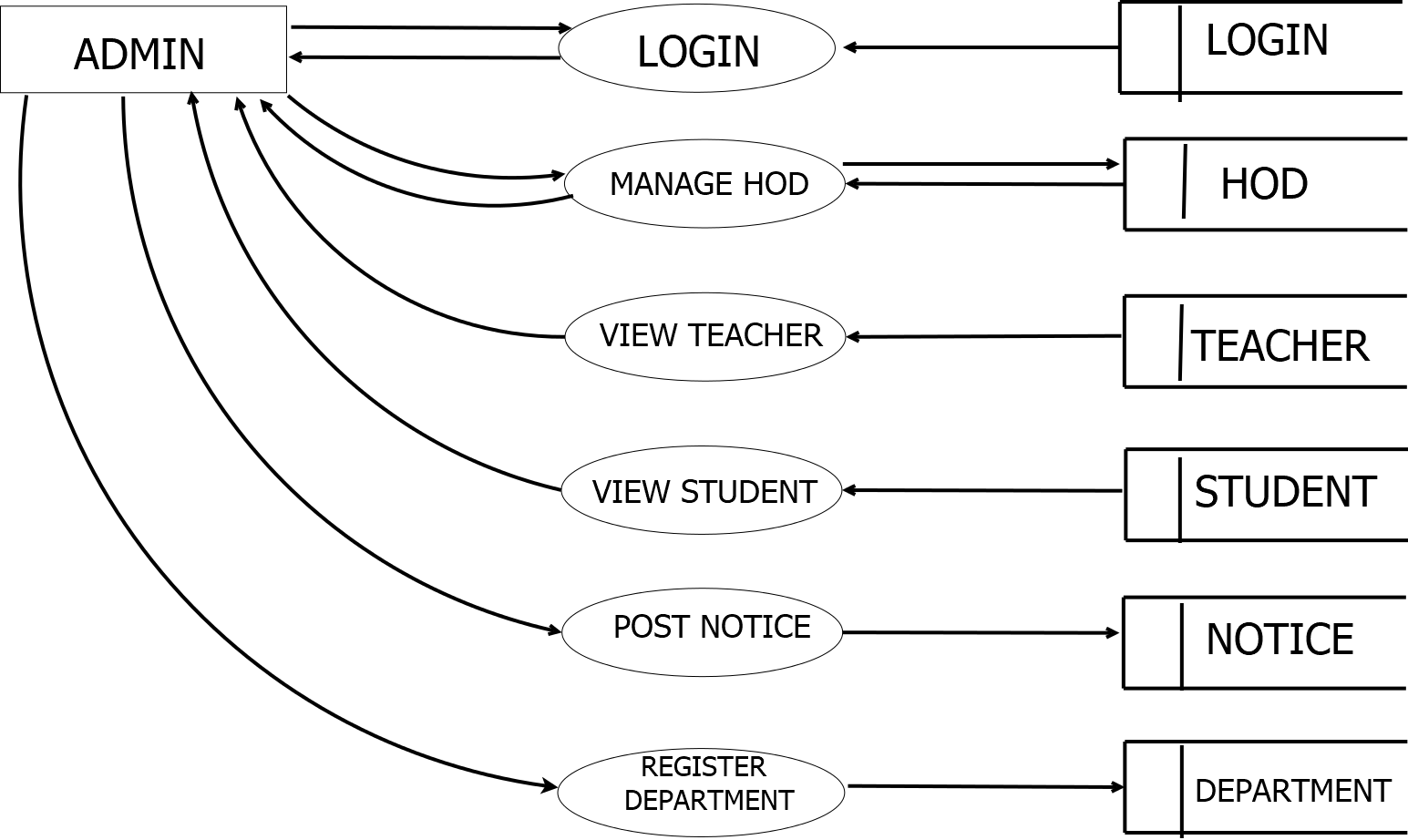
**DATA FLOW DIAGRAM LAYERS :**

Data flow diagrams in several nested layers. A single process node on a high level diagram can be expanded to show a more detailed data flow diagram. Draw the content diagram first, followed by various layers of data flow diagrams.

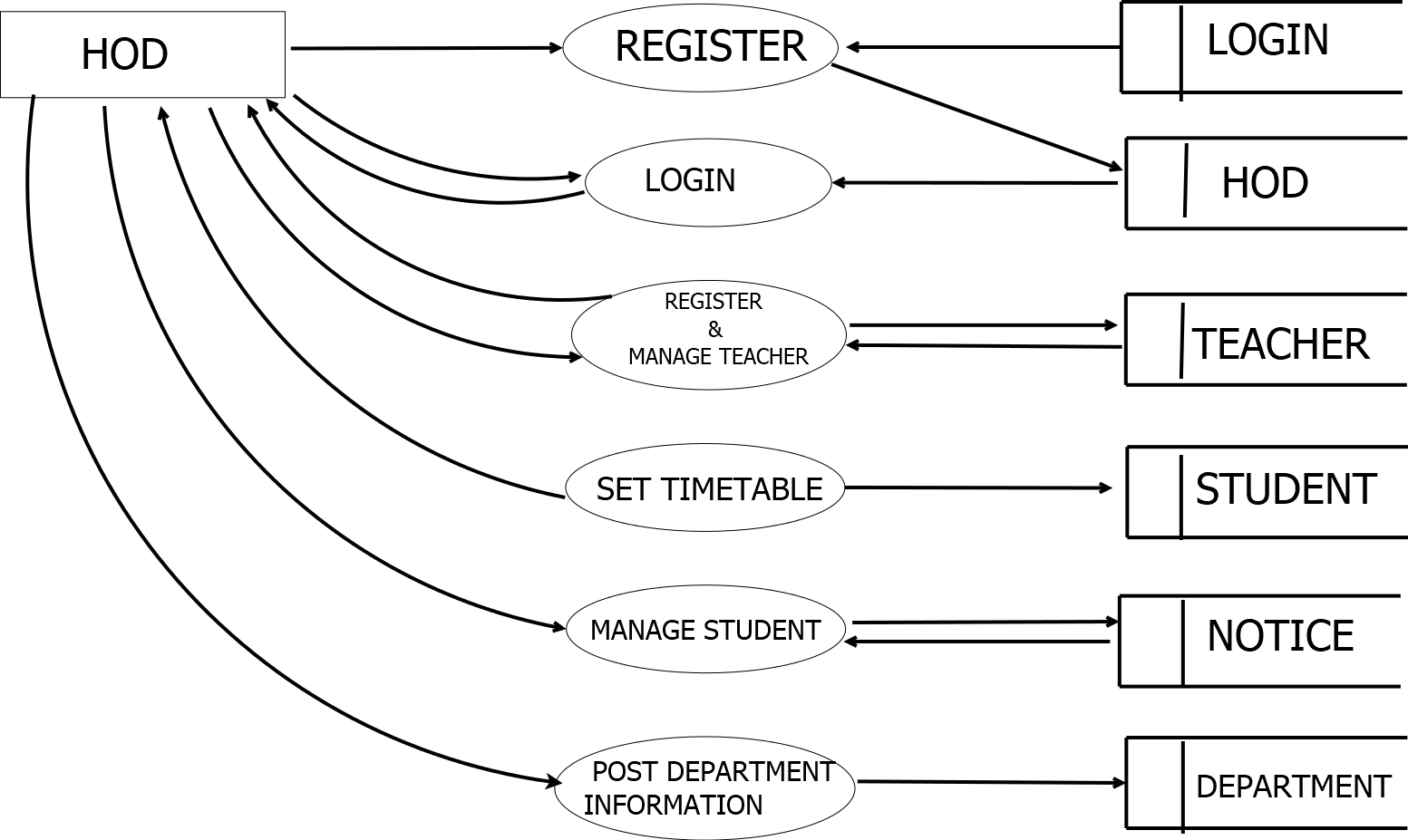
**Level-0**

****

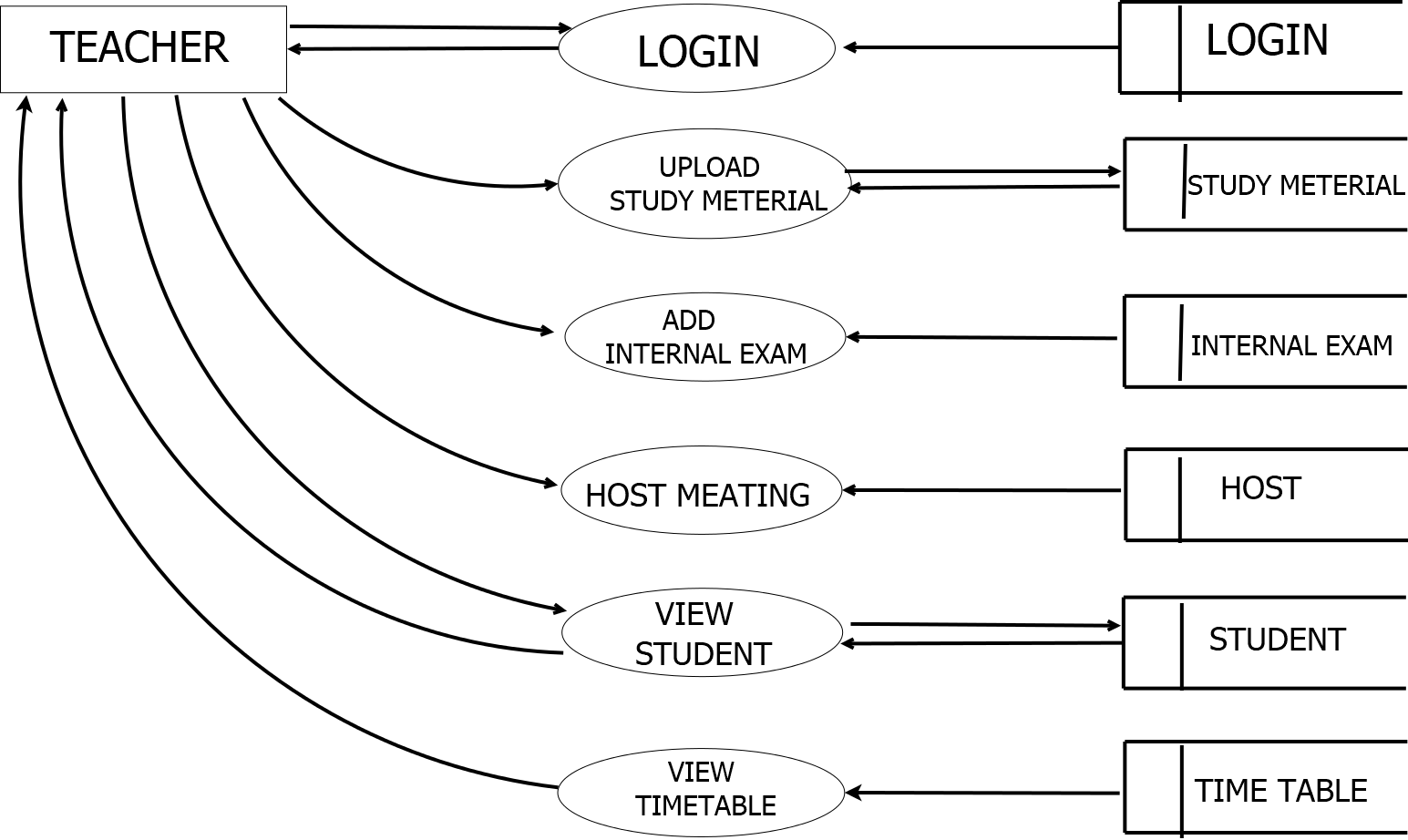
**Level-1: Admin**



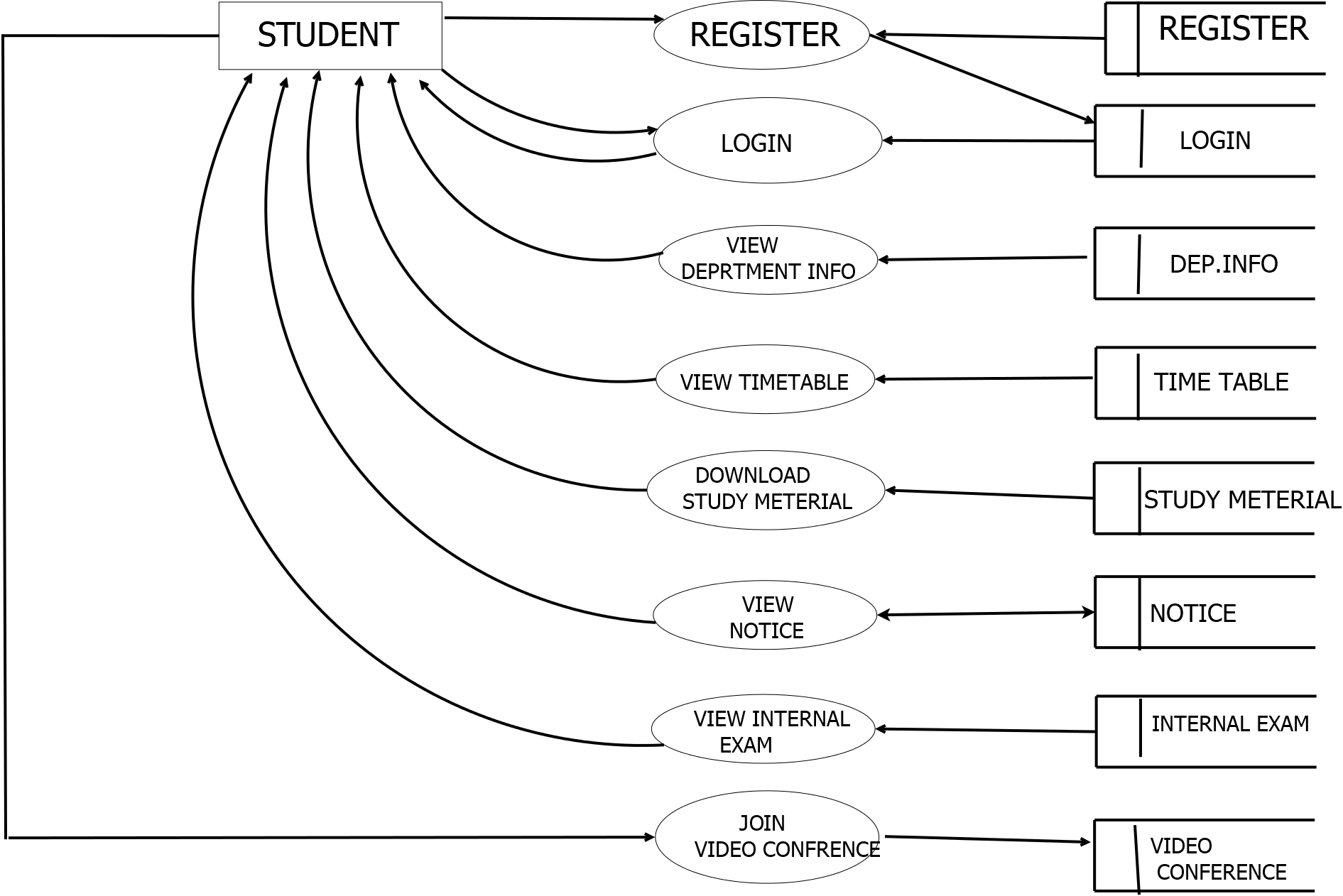
**Level-2: HOD**



**Level-2: Teacher**



**Level-3: Student**



**2.8.2 Entity Relationship diagram**

An entity relationship diagram is a data modeling technique that creates a graphical representation of the entities and the relationship between entities within an information system. An ER model is an abstract way to describe a database. Describing a database usually starts with a relational database, which stores data in tables. Some of the data in these tables point to data in the other tables.

An entity may be defined as a thing which is recognized as being capable of an independent existent and this can be uniquely identified. An entity is an abstraction from the complicity of a domain. When we speak of an entity, we normally speak of some aspect of the real world which can be distinguished from other aspects of the real world.

* Three basic elements in ER model includes :
* Entities are the things about which we seek information
* Attributes are the data we collect about the entities
* Relationships provide the structure needed to draw information from multiple entities

E-R Diagram Symbols

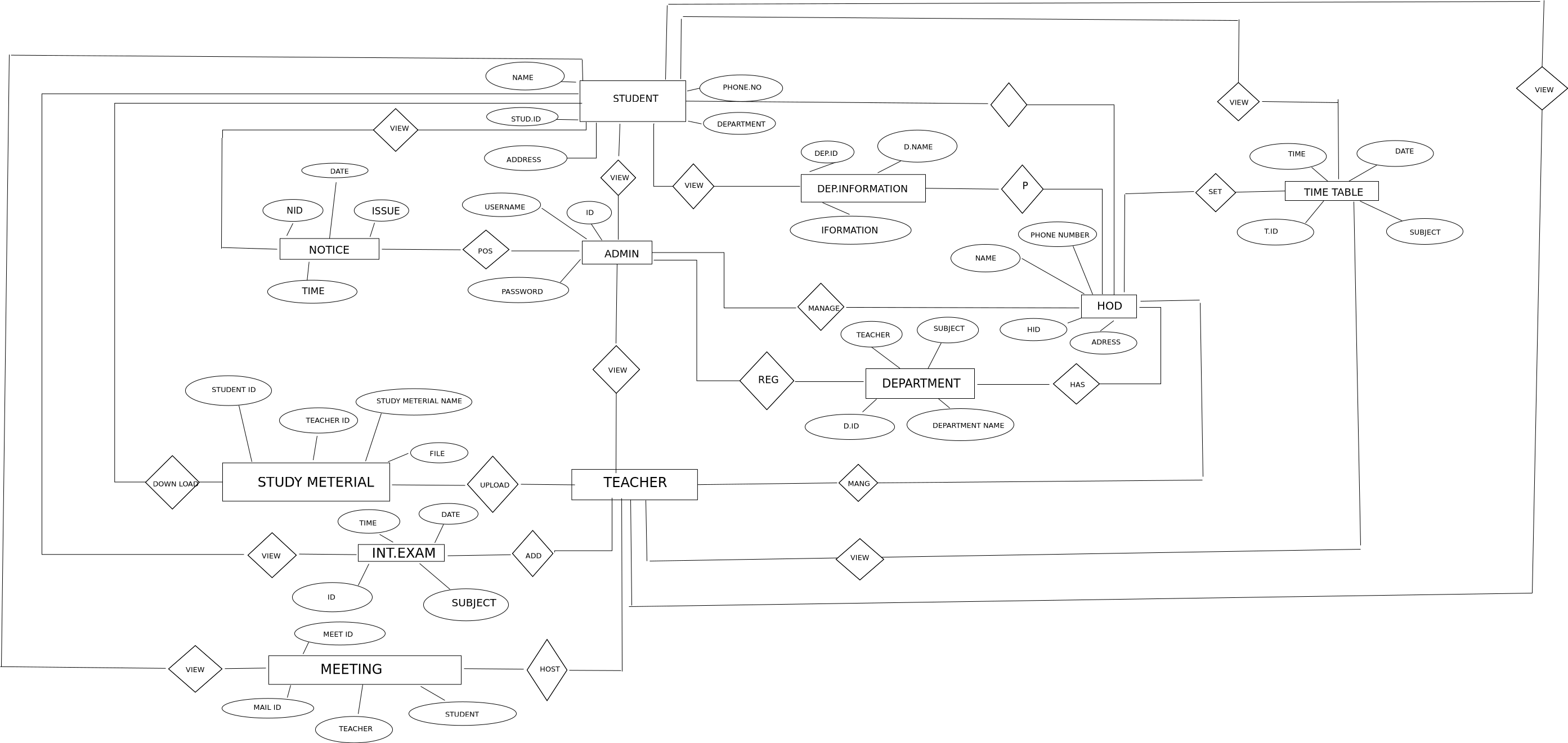
. ENTITY :

. ATTRIBUTE :

. RELATION :

. CONECTION :

ER DIAGRAM



**3. SYSTEM DESIGN**

**3.1. Normalization**

Normalization is the process of organizing the fields and tables of a relational database to minimize redundancy. Normalization usually involves dividing large tables into smaller tables and defining relationships between them. The objective is to isolate data so that additions, deletions and notification of a field can made in just one table and then propagated through the rest of the database using the defined relationship.

**3.1.1 First Normal Form**

First normal form achieved when all repeating data group are removed so that the record is of fixed lengths. The repeating group are treated as an additional group structure. A common data item interconnects the original record and the new record.

**3.1.2 Second Normal Form**

Second normal form is achieved when a record is in first normal form and each item in the record is fully depend on the primary key for identification to achieve second normal form, every data items in a record that is not depend on the primary key of the record should be removed and used to form a separate a relation.

**3.1.3 Third Normal Form**

It is achieved when transitive dependencies are removed from the second normal form.

* **DATA DESIGN**

**ADD YOUR DATABASE DESIGN**

**SYSTEM DESIGN**

System design is the second phase of the system life cycle. The detailed design of the system selected in the study phase is accomplished in the design phase. The principal activity performed during this phase includes allocation of function between computer programs equipment and manual tasks and database design and test requirement definition.

In the course of design phase, the performance specification is expanded into the design specification. The user oriented baseline prepared in study phase becomes a baseline document, oriented to the needs of the programmers and other professional who will actually develop the system. A design phase report is prepared after the completion of design phase activities and the review is held with the user organization in order to determine whether or not the computer based business information system project is ready to the development phase.

**INPUT DESIGN**

Input design is a part of the overall system design, which requires very careful attention. Often the collection of input data is the most expensive part of the system, in terms of both the equipment used and people involved. If the data going into the system is incorrect, then the processing and output will magnify the errors. Thus the clear objectives of input design are:

* To produce a cost-effective method of input.
* To achieve the highest possible level of accuracy.
* To ensure that the input is acceptable to and understood by the user.

**OUTPUT DESIGN**

The output design is done so that the result of processing could be committed to the user and to provide a hard copy of these results and evaluations for later consultations. Effective output design will improve the clarity and performance of outputs. Output design phase of the system is concerned with the convergence of information’s to the end user-friendly manner. The output design should be efficient, intelligible so that system relationship with the end user is improved and thereby enhancing the process of decision making. Outputs from the computer systems are required primarily to communicate the results of the processing to the users. They are also used to provide a permanent copy of these results of processing to the users. They are also used to provide a permanent copy of these results for late consultation. There are various types of output required by most systems, the main ones are:

* External outputs, whose destination is outside the organization and which require special attention because they project the image of the organization.
* Internal outputs, whose destination is within the organization and which require careful design because they are the user’s main interface with the computer.
* Operational outputs, whose use is purely within the computer department.
* Turn around outputs, to which the data will be added before they are returned to the computer for further processing.

ADD YOUR I/O DESIGN

* **REQUIREMENT SPECIFICATION**

This document fully and formally describes th**e** requirements of the proposed project system. It sets out the functional and nonfunctional requirements (Non-functional requirements impose constraints on the design or implementation such as performance engineering requirements, quality standards, or design constraints.) and includes a description of the user interface and training requirements. SRS is a document that completely describes what the proposed should do without describing how the software will do it. The basic purpose of SRS is to bridge the communication gap between the parties involved in the development of the software. Another important purpose of developing an SRS is helping the user to understand their own needs. The main advantage of SRS is it provides reference for validation of the final product.

* **FUNCTIONAL REQUIREMENTS**
* **REGISTRATION**

Admin register Hod and Hod registere Teachers , Students and Teachers can register through android application and login themselves.

* **ADMIN MANAGEMENT**

Admin can manage Hod ,Admin can take action like Approve and Reject against Hod.

* **HOD MANAGEMENT**

Admin can manage Hod likely Hod can manage Teachers take action like Approve or

Reject.

* **TEACHER MANAGEMENT**

Teachers can manage students ,Teachers can allows students to get in the class.

* **NON-FUNCTIONAL REQUIREMENTS**
* **Usability**: This application be used by Smartphone users. This is really user friendly since it is simple and understandable. The simple user interfaces make students relaxed. The mobile application interface is made extremely simple and easy to use.
* **Maintainability**: Maintenance and future enhancement is possible for this software. In future, more functions and features can be added with ease.
* **Security**: Security can be implemented through proper authentication. Login module checks the login activities in the system using the username and password. Unauthorized accesses will not be possible in the system.
* **Availability** Interested users can register themselves by downloading the application and they can login by using their username and password using android mobile. Only requirement is that they need an internet connection. Users can use the applications by login to the web application. Their registration is done by the themselves. Mobile application can be made available at the play store.

**4. SYSTEM IMPLEMENTATION**

**4.1 Introduction**

System implementation is the final stage of software development life cycle. For the successful implementation and cooperation of new systems users must be selected, educated and trained. Unless the users are not trained, the system will become complex it will become feel as a burden for them. A software implementation method is a systematically structured approach to effectively integrate software based service or component into the workflow of an organizational structure or an individual end-user. A software implementation method is a blueprint to get users and/or organizations running with a specific software product. The method is a set of rules and views to cope up with the most common issues that occur when implementing a software product: business alignment from the organizational view and acceptance from human view. It is stated that the implementation of software consumes up to 1/3 of the budget of a software purchase. The complexity of implementing product differs on several issues. Examples are: the number of end users that will use the product, the effects that the implementation has on changes of tasks and responsibilities for the end user, the culture and the integrity of the organization where the software is going to be used and the budget get available for acquiring the software.

The implementation stage of the system being by preparing a plan for implementation of the system. According to this plan, activities are to be carried out, discussions are to be made regarding the equipment to be required, resources and additional facilities required implementing the system. The most critical stage in achieving a successful system is by giving users confidence that the system will work based on their requirements and be effective. This method also offers the greatest securities since the old system can take over if the errors are found or inability to handle certain transactions while using the new system.

The implementation involves the following formalities:

* Careful planning
* Investigation of the system and constraints
* Design the methods to achieve the changes
* Training the staffs in the changed phase
* Evaluation of the changeover method

**SYSTEM TESTING**

Testing the process of executing website is controlled manner in order to answer the suggestion does the website behave be specified?” Testing is often used in association with the terms verification and validation .Verification is checking or testing of items, including websites, for conformance and consistency with associated specifications.

**Unit Testing**

Unit testing is a method by which individual unit of source code, set of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they are fit to use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming, a unit could be an entire module, but it is more commonly an individual function or procedure.

**Module Level Testing**

In module level testing error will be found at each individual unit, it encourages the programmer to find and rectify the errors with affecting other modules.

**Integration Testing**

Integration testing identifies problems that occur when unit is combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining unit. You know that any errors discovered when combining units likely related to the interface between units .This method reduce the number of possibilities to a far simpler level of analysis.

**Validation Testing**

It ensures that the product has been build according to the requirements and design specifications. Validation ensures that “you built the right product”. In this majority of validations done during the data entry in the operation where there is a maximum possibility of entering wrong data. We perform validation testing in “Advanced ATM” application system.

**Recovery Testing**

Recovery Testing is the activity of testing how well an application is able to recover from crashes, hardware failures and other similar problems. Recovery testing is the forced failure of the software in a variety of ways to verify that recovery is properly performed.

**Output Testing**

After performing the validation testing, the next step is output testing of Advanced ATM application system. Since no system would be termed as useful until it produce the required output in the specified format.

**Source code Testing**

This examines the logic of the system. If we are getting the output that is required by the user, then we can say that the logic is perfect.

**IMPLEMENTATION**

Implementation of the system refers to the final installing of the package in its real environment , to the satisfaction of the indeed users and the operation of the system. It is the process of converting a new or revised system design to operation. It is the key stage in achieving successful new system. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to new system. It must therefore be carefully planned and controlled. Proper guidance should be imparted to the users so that he is comfortable in using the application.

**5.1 Implementation Plan**

The transformation from theoretical designs to working system is done in this stage. Developed package of system is tested with simple data, accurate error identification and then through proposed change from the user etc. a dress rehearsal working of system is done, so as the system is scrutinized, for pointing out errors and modifications required if any keeping in mind the expectations and specifications from the system.

**Education And Training**

The expectations from the system are made achieved by the people who will be involved to be confident of their role in the new system. The complexity of the system is directly proportional to the amount of training and education given for the user .Education is different from the training, as the user through education can be a part of development of the system. Education has the capability to make training more interesting and important contributions in the system changes.

Training just means to give user specific skills in order to meet their new job requirements. The role of system analyst in training will make it more understandable and effective. Training provides a better overview of new system and its present objectives.

**Training Of Application Software**

Awareness about the new system is made to the users through training, and with the underlying philosophy of the system (screen design, flow, error types during inputs, validation checks etc.) application use the system, as the users of the system may be of at different levels of hierarchy.

**5.2 Post Implementation Review**

System performance v/s expected requirements are evaluated. The implementation problems if any is taken seriously and taken care of along with admiring the achievements, failures etc. The works done here are used to improve the efficiency and user friendliness of the system.

**Security**

System security is a branch of technology known as information security as applied to computers and networks. The objective of system security includes protection of information and property from theft, corruption, or natural disaster, while allowing the information and property to remain accessible and productive to its intended users. The term system security, means the collective processes and mechanisms by which sensitive and valuable information and services are protected from publication, tempering or collapse by unauthorized activities or untrustworthy individuals and unplanned events respectively. The technologies of system security are based on logic. As security is not necessarily the primary goal of most computer applications, designing a program with security in mind often imposes restrictions on that program’s behavior.

**Maintenance**

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited.

There are three types of maintenance:

* Corrective(Fixing bugs/errors)
* Adaptive(Updates due to environment changes)
* Perfective(Enhancements, requirements changes

Maintenance is enigma of the system development. The definition of the software maintenance can be given describing four activities that are undertaken after the program is released for use.

The maintenance activity occurs since it is unreasonable to assume that software testing will uncover all in a large system. The second activity that contributes the definition of maintenance occurs since rapid changes are encountered in every aspects of computing. The third activity involves recommendation for new capabilities, modification to the existing functions and general enhancements when the software is used. The fourth maintenance activity occurs when software is changed to improve future maintainability or reliability.

**CONCLUSION**

ADD CONCLUTION ABOUT PROJECT

FUTURE ENHANCEMENT

ADD FUTURE ENHANCEMENT IN DETAILS

**APPENDIX**

**ADD SAMPLE SCREENSHOT**

**ADD SAMPLE CODE**