**PROJECT SYNOPSIS**
**ON**

# STUDENT HELP DESK

# TABLE OF CONTENTS

# INTRODUCTION

Student Help Desk project is developed for the facilities in the campus of college/university/school. This system can be used to automate the workflow of service requests for the various facilities in the campus. Previously, handling the services in campus was very hectic process and also not efficient. The previous systems are developed using old methodologies, most of the are deprecated now for the new systems.

As their work load has increased so various software have been developed in the recent time. This project is also designed keeping all these requirements of the Student Help Desk that are to automate its operation of keeping records and store them in the form of a large and user friendly database further facilitating easy access to the personnel.

# PROBLEM DEFINITION

At present there are many online help desk that maintain their day to day transactions manually. These have a number of students, faculties, staff as a result they need to track of all these clients requirements. At this point Student Help Desk will play an important role in helping the admin to perform all these operation on a single click. This project will handle all the necessary data as well as every minute details of the campus and properly.

## Purpose

The purpose of this document is to specify requirements and to give guidelines for the development of above said project. In particular it gives guidelines on how to prepare the above said project. The old system was suffering from a series of the drawbacks. Since whole system was to be maintained with hands the process of keeping, maintaining and retrieving the information was very tedious and lengthy. The records were never used to be in a systematic order as a result there used to be lot of difficulties in associating any particular transaction with a particular context. If any information was to be found it was required to go through different registers, documents there would never exist anything like report generation. There would be unnecessary consumption of time while entering records and retrieving the records. One more problem was that it was very difficult to find errors while entering records as a result if one record was entered then it was difficult to update these records.

## Scope

As this is generic software it can be used by a wide variety of colleges/universities/schools to automate the process of manually maintaining the records related to the subject of maintaining the entries and updating these records that too in a single click. as the name of the project, STUDENT HELP DESK states that it would be a two way communication software i.e. it could be accessed by both admin and the client in order to view certain information and this creates a plus point fir this software.

# OBJECTIVE

Today's world is computer world because most of work is doing with the help of computer. Dependency on computer is behind the few reasons. We cannot easily manage to store large number of data or information single handle. If we will be need some information or data in urgency then we cannot manage in manually these works are very difficult if we cannot use computer.

As the generic software it can be used by a wide verity of colleges, universities, and many more to automate the process of manually maintaining records related to the subject of maintain the records of each personnel in campus.

This software is basically updating the manual work of the help desk systems. So that organization can manage their record in efficient and organize them.

- The main objective of the project is to develop software that facilitates the data storage, data maintenance and its retrieval for the gym in an igneous way.
- To store the records of the customers, the staff that has the privileges to access, modify and delete any record and finally the service, college/school provides to its staff and students.
- The faculties, staff and students can log in to the system in order to check details of their facilities but they cannot bring any changes in the records.
- To develop easy-to-use software, this handles the student-campus relationship in an effective manner.
- To save manpower.
- It will speed the processing of data and transaction.
- It will provide best security features such as provisions of passwords
- To develop a user friendly system that requires minimal user training.

**System Objective:-**

Today's world is computer world because most of work is doing with the help of computer. Dependency on computer is behind the few reasons. We cannot easily manage to store large number of data or information single handle. If we will be need some information or data in urgency then we cannot manage in manually these works are very difficult if we cannot use computer.

**System Context:-**

This section clearly depicts the environment and boundaries of STUDENT HELP DESK SYSTEM and the entities with which it interacts. It helps us see how the system fits into the existing scheme of things. What the system will do by itself.

**Functional Requirement:-**

This Software must request Username and Password for access to data, only after authentication will allow access to the system. The Software must allow input of products data from administrator and secured access.

**Non-Functional Requirement:-**

In this Software Input error will be returned in red with appropriate message box. System should automatically update after every transaction.

# REQUIREMENTS SEPECIFICATIONS

## HARDWARE AND SOFTWARE SPECIFICATION

### 1) Hardware Requirements

- Pentium IV Processor
- 512 MB RAM
- 40 GB HDD
- Color Monitor
- Keyboard, Mouse

### 2) Software Requirements:-

- WAMP/XAMPP
- Any text editor

# MODULE DISCRIPTION

- **Login:**

The STUDENT HELP DESK SYSTEM first activates the login page. Here the user enters USER NAME AND PASSWORD and the system starts authentication process in which the USER NAME AND PASSWORD is matched with the existing USER NAME AND PASSWORD in the database. If the password matches then it is allowed to the main page else it warns the user for invalid USER NAME AND PASSWORD.

After the successful authentication the system activated menus. The activity log also prepared for failure and security There are two types of users using this software i.e., admin, student, faculties and staff.

- **Student:***

This module has software configuration that students can access this module. Here students performs certain operations like requesting for facility, feedback, faculty details, etc.

- **Admin:**

Admin can access this module. Here, admin can perform some tasks like, view students, view faculties, facilties, staff and much more.

# PROBLEM FORMULATION

## Introduction:-

Problem introduction or problem starting is the starting point of the software development activity. The objective of this statement is to answer: Exactly *what must the system do*? The software project is initiated by the client's need. In the beginning, these needs are on the minds of various people in the client's organization. The analyst has to identify the requirements by talking to the people and understanding to their needs. It goes without saying that an accurate and through understanding of software requirement are essentials to the success of software development effort. All further development like system analysis. System design and coding will depends on how accurate and well understood the requirements are poorly analyzed and specified software will disappoint the user and will bring brief to the developer. No matter how well designed and well appearances are often deceiving. Chances of misinterpretation are very high, ambiguity is probable and communication gap between customer and developer is bound to bring confusions. Requirements understanding begin with a clear and concise heading stating in sentence the task to be performed. Then requirements are describe in a technical manner precise statement.

# Feasibility study:-

All projects are feasible given unlimited resources and infinite time! Unfortunately, the development of computer based system is more likely to be plagued by a scarcity of resources. It is both necessary and prudent to evaluate the feasibility of the project at the earliest possible time. Months or years of effort, Money loss and untold professional embarrassment can be averted I few better understand the project at its study time.

This type of study determines if an application can and should be developed. Once it has been determining that, application is feasible. After that analyst can go ahead and prepares the project specification, which finalizes project requirements. Feasibility studies are undertaken within tight time constraints.

Technical Feasibility
Operational Feasibility
Economic Feasibility
Legal Feasibility

## 1.Technical Feasibility:-

As we know the technical feasibility is concerned with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may vary considerably, but might include:

- The facility to produce outputs of advertisements, shopping and mailing in a given time for ease of use.
- Response time under certain condition is minimal.

  Ability to process a certain volume of transaction at a particular speed.
  Facility to communicate data to distinct location.
  In examining the technical feasibility, configuration of the system is given more importance than the actual make of hardware. The configuration should give the complete picture about the system's requirements- how many workstations are required, how these units are interconnected so that they could operate and communicate smoothly.

## 2.Operational Feasibility:-

Proposed projects are beneficial only if they can be turned into information system that will meet the financial management requirements of the business/organization. This test of feasibility asks if the system will work when it developed and installed. Are there major barriers to implementation?

Some of the important questions that are useful to test the operational feasibility of a project are given below:

- Is there sufficient support for the project from the implementation? From user? If the present system is well liked and used to the extent that persons will not be able to see reasons for change, there may be resistance.
- Are current business methods acceptable to the user? If they are not, user may welcome a change that will bring about a more operational and useful system.
- Have the user been involved in the planning and development of the Project? If they are involved at the earliest stage of project development, the chances of resistance can be possibly reduced.
- Will the proposed system cause harm? Will it produce poorer result in any case or area?
- Will the performance of staff member fall down after implementation? Issue that
- Appears to be quite minor at the early stage can grow into major problem after Implementation.

## 3. Economical Feasibility:-

Economic analysis is the most frequently used technique for evaluating the effectiveness of the proposed system. More commonly known as cost/benefits analysis, the procedure is to determine the benefits and savings that are expected from the purposed system and compared with costs.

If benefits outweigh cost, a decision is taken to design and implement the system. Otherwise, further justification or alternative of the proposed system will have to be made if it has a chance of being approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle. The analysis part also clears the doubt of economic problems which could be possible in developing the system. As already mentioned that the company has to just pay the developed software cost and not other investment is needed at the time of implementation of the new system as the preliminary requirements already exist in the company.

## 4.Legal Feasibility:-

In the legal feasibility is necessary to check that the software we are going to develop is legally correct which means that the ideas which we have taken for the proposed system will be legally implemented or not so, it is also an important step in feasibility study.

# DATA TABLES

ADD NEWS

| Name Column | Data Type | Constraints |
|---|---|---|
| Sr_no | Int(3) | **(Primary Key)** |
| news | Varchar(100) | Not Null |

ADMINISTRATION

| Name Column | Data Type | Constraints |
|---|---|---|
| Username | Varchar(30) | |

| | | |
|---|---|---|
| Password | Varchar(10) | |

COURSES

| Name Column | Data Type | Constraints |
|---|---|---|
| Course_name | Varchar(30) | |
| Duration | Int(1) | |
| Semester | Int(1) | |
| Seats | Int(3) | |
| Fees | Int(6) | |

Exam_scheme

| Name Column | Data Type | Constraints |
|---|---|---|
| Course_name | Varchar(30) | |
| Semester | varchar(1) | |
| Date | Date | |
| Paper_code | Int(3) | |
| Paper_title | Varchar(100) | |
| Shift | Varchar(10) | |
| Time_from | Time | |
| Time_to | Time | |

STUDENT HELP DESK

Feedback

| Name Column | Data Type | Constraints |
|---|---|---|
| Name | Varchar(30) | |
| Course | varchar(30) | |
| Email | Varchar(30) | |
| Feedback | Varchar(100) | |

Notes

| Name Column | Data Type | Constraints |
|---|---|---|
| course | Varchar(30) | |
| Semester | int(1) | |
| subject_name | Varchar(50) | |
| Notes | Longblob | |

Paper

| Name Column | Data Type | Constraints |
|---|---|---|
| Course | Varchar(30) | |
| Semester | int(1) | |
| paper_name | Varchar(50) | |
| Paper | Longblob | |

STUDENT HELP DESK

Registration

| Name Column | Data Type | Constraints |
|---|---|---|
| Sr_no | int(4) | **PRIMARY KEY** |
| Username | varchar(30) | |
| Password | Varchar(8) | |
| S_name | varchar(30) | |
| F_name | varchar(30) | |
| M_name | varchar(30) | |
| Gender | varchar(6) | |
| Dob | Date | |
| Address | varchar(50) | |
| City | varchar(20) | |
| Pincode | Int(6) | |
| State | varchar(20) | |
| Mobile | varchar(10) | |
| Email | varchar(30) | |
| Photo | varchar(20) | |
| Declaration | varchar(10) | |

Syllabus

| Name Column | Data Type | Constraints |
|---|---|---|
| Course | Varchar(10) | |
| Semester | varchar(1) | |
| Subject_code | Varchar(10) | |
| Subject_name | Varchar(50) | |
| Syllabus | Longblob | |

## PROBLEM SPECIFICATION

The definition of our problem lies in manual system and a fully automated system.

### 1 Manual System:-

The system is very time consuming and lazy. This system is more prone to error and sometimes the approach to various problems is unstructured.

### 2 Technical System:-

With the advent of latest technology if we do not update our system then our business result in losses gradually with time. The technical system contains the tools of latest trend

i.e. computers, printers, FAX, Internet etc the system with the technology are very fast, accurate, user friendly and reliable.

## 3 Need of Student Help Desk System:-

Student Help Desk System software is very needy for various Colleges and Universities. This software helps them maintain day to day entries of the students who are their main focus or those who desire to be.

## 4The Proposed System:-

The proposed system helps them in many ways. It helps them do tracking very easily. Account maintenance also becomes easier. They can keep track of their performance and many more. The software is provided with all the master entries to enter any new service, or faculty or staff to add or modify and delete.

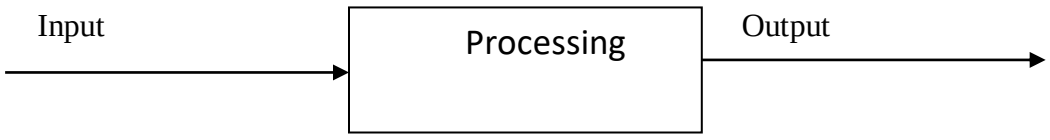
## 5Existing System:-

At present various Student help desk system maintain their day to day transactions manually. These have thousands of students, services, faculties they need to track of all those. So it is very essential to track. To find the proper flow of services is also another risky job and this also applies for in finding the details of the assigned personnel. So proper system is required. They need full pledged software to maintain their day to day transactions..

# SYSTEM ARCHETECTURE DESIGN

## 1 DEFINING A SYSTEM:-

Collection of component, which are interconnected, and work together to realize some objective, from a system. There are three components in every system, namely input, processing and output



# SYSTEM DEVELOPMENT LIFE CYCLE:-

The **System development life cycle (SDLC)**, or **Software development process** in systems engineering, information systems and software engineering, is a process of creating or altering information systems, and the models and methodologies that people use to develop these systems. In software engineering, the SDLC concept underpins many kinds of software development methodologies. These methodologies form the framework for planning and controlling the creation of an information system the software development process.

Broadly, following are the different activities to be considered while defining the system development life cycle for the said project:

Problem Definition
System Analysis
Study of existing system
Drawback of the existing system
Proposed system
System Requirement study
Data flow analysis
Feasibility study
System design
Input Design (Database & Forms)
Updating
Query /Report design
Administration
Testing
Implementation
Maintenance

## 1.2 SYSTEM ANALYSIS:-

**Systems analysis** is the study of sets of interacting entities, including computer systems analysis. This field is closely related to requirements analysis or operations research. It is also "an explicit formal inquiry carried out to help someone (referred to as the decision maker) identify a better course of action and make a better decision than he might otherwise have made.

System development can generally be thought of having two major components: systems analysis and systems design. In System Analysis more emphasis is given to understanding the details of an existing system or a proposed one and then deciding

whether the proposed system is desirable or not and whether the existing system needs improvements. Thus, system analysis is the process of investigating a system, identifying problems, and using the information to recommend improvement to the system.

## 1.3 SYSTEM DESIGN:-

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering. If the broader topic of product development "blends the perspective of marketing, design, and manufacturing into a single approach to product development," then design is the act of taking the marketing information and creating the design of the product to be manufactured. Systems design is therefore the process of defining and developing systems to satisfy specified requirements of the user. Until the 1990s systems design had a crucial and respected role in the data processing industry. In the 1990s standardization of hardware and software resulted in the ability to build modular systems. The increasing importance of software running on generic platforms has enhanced the discipline of software engineering.

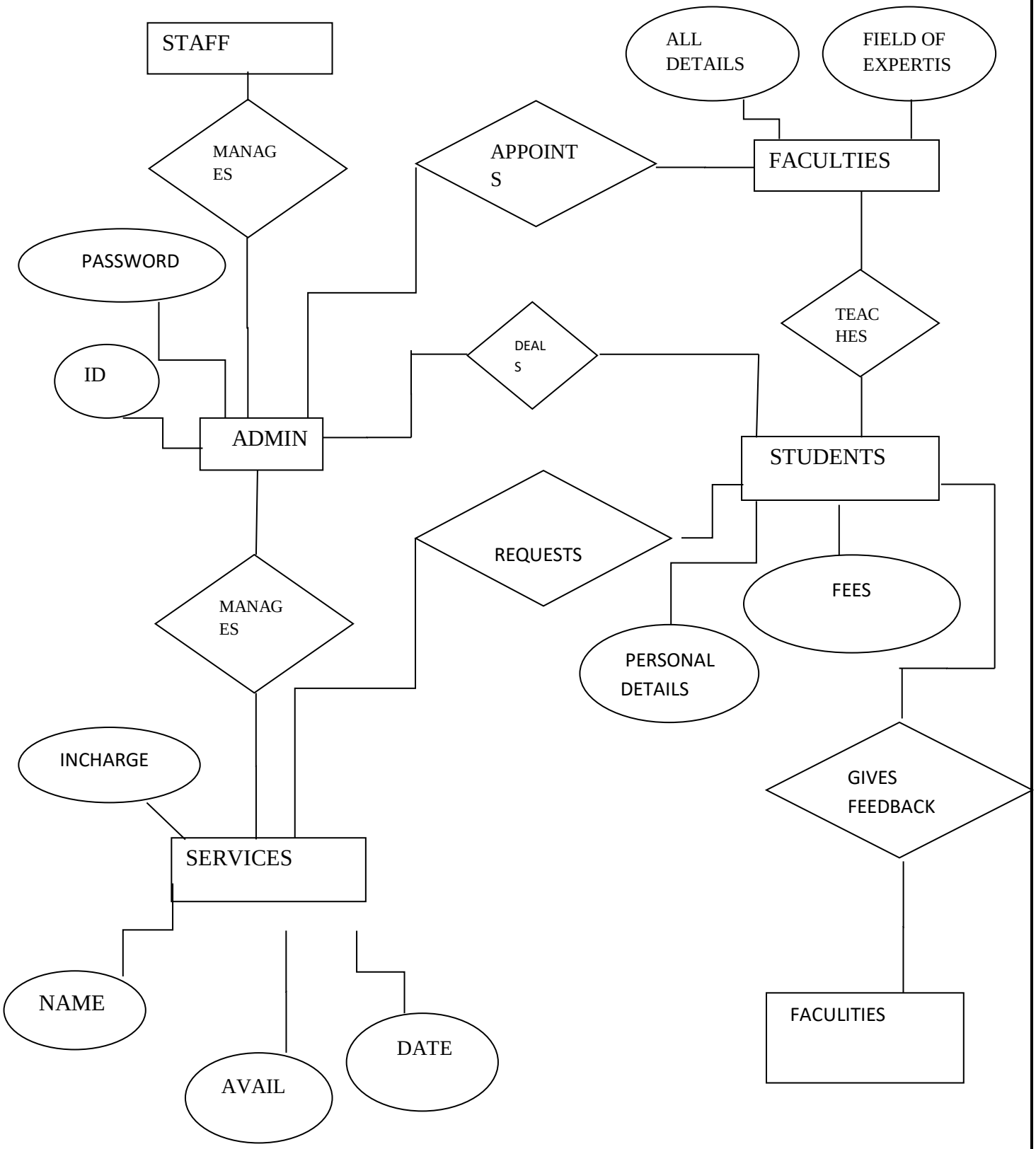
Object-oriented analysis and design methods are becoming the most widely used methods for computer systems design. The UML has become the standard language in object-oriented analysis and design. It is widely used for modeling software systems and is increasingly used for high designing non-software systems and organizations

# ENTITY RELATION DIAGRAMS:-

The Entity Relation Model or Entity Relation Diagram (ERD) is a data model or diagram for high-level description of conceptual data model, and it provides a graphical notation for representing such data models in the form of entity relationship diagrams. Such models are typically used in the first stage of Management information system design; they are used for example, to describe information needs and/ or the type of information that is to be stored in the Database during the requirement analysis. The data modeling technique, however, can be used to describe any ontology ( i.e an overview and classification of used term and their relationships) for a certain universe of discourse (i.e area of interest ).

In the case of design a Management Information System that is based on a database, the conceptual data model is, a later stage( usually called logical design), mapped to a logical data model such as, relational data model; this is turn in mapped to a physical model during physical design. Note that sometimes, both of the phases are referred a “physical design”. There are number of convention for entity-relation diagrams (ERDs). The classical notation is describe in the remainder of this article, and mainly related to the conceptual modeling. There is a range of notation more typically employed in physical and logical database design.

STUDENT HELP DESK



# DATA FLOW DIAGRAM

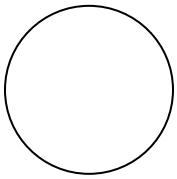
The data flow diagram shows the flow of data within any system. It is an important tool for designing phase of software engineering. Larry Constantine first developed it. It represents graphical view of flow of data. It's also known as BUBBLE CHART. The purpose of DFD is major transformation that will become in system design symbols used in DFD.

In the DFD, four symbols are used and they are as follows.

1. A square defines a source (originator) or destination of system data.

2. An arrow identifies data flow-data in motion. It is 2a pipeline through which information flows.

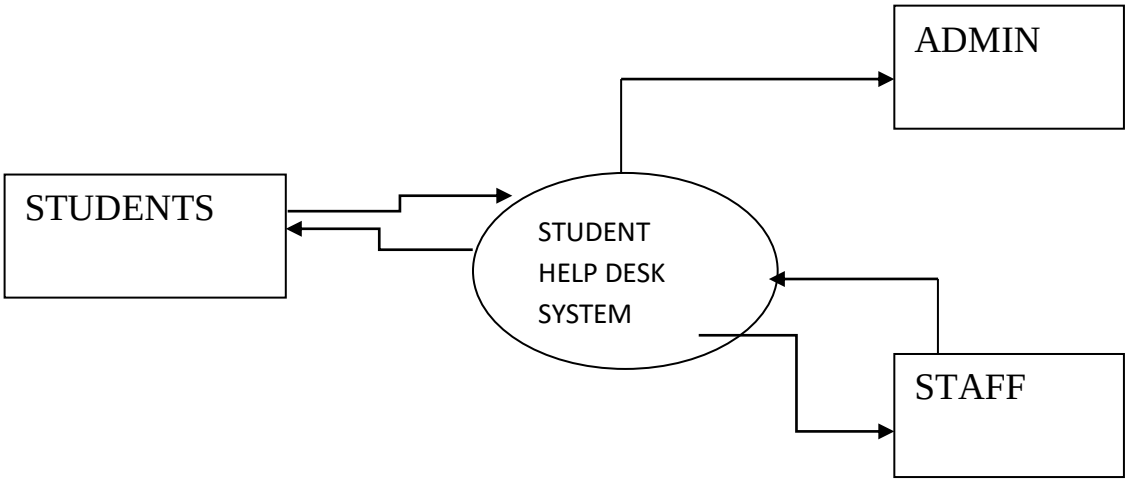
3. A circle or a "bubble "(Some people use an oval bubble) represents a process that transfers informing data flows into outgoing data flows.



4. An open rectangle is a data store-data at rest, or a temporary Repository of data.

## Context Level Data Flow Diagram:-

This level shows the overall context of the system and its operating environment and shows the whole system as just one process. Online book store is shown as one process in the context diagram; which is also known as zero level DFD, shown below. The context diagram plays important role in understanding the system and determining the boundaries. The main process can be broken into sub-processes and system can be studied with more detail; this is where 1st level DFD comes into play.

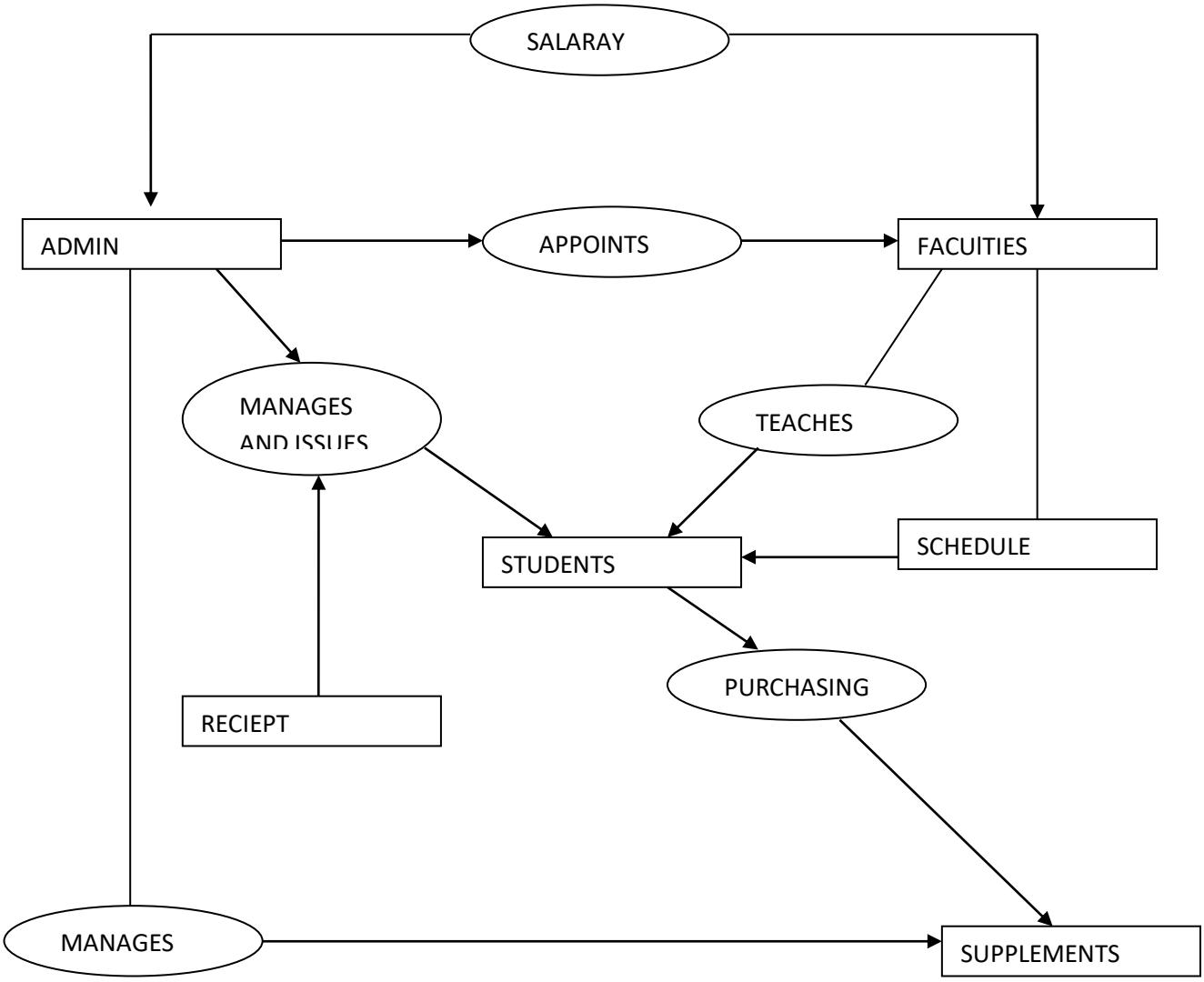



Zero Level Data Flow Diagram

## First Level DFD:-

This level (level 1) shows all processes at the first level of numbering, data stores, external entities and the data flows between them. The purpose of this level is to show the major high-level processes of the system and their interrelation. A process model will have one, and only one, level-1 diagram. A level-1 diagram must be balanced with its parent context level diagram, i.e. there must be the same external entities and the same data flows, these can be broken down to more detail in the level 1.

STUDENT HELP DESK




One level data flow diagram

# Project Planning & Project Scheduling

## PERT CHART

A project plan needs to be created to ensure the timely completion of the project. As part of project analysis, we break the project down to a number of stages and use a Gantt chart and PERT chart to describe specific tasks and status.

The Work Breakdown Structure of our proposed system "E-Commerce" is shown below:

| ID | | Task Name | Duration | Start | Finish | Predecessors | Resource Names |
|---|---|---|---|---|---|---|---|
| 1 | | **Project Initiation** | 2 days | Mon 6/8/18 | Tue 7/8/18 | | |
| 2 | | Draft Project Plan | 2 days | Mon 6/8/18 | Tue 7/8/15 | | |
| 3 | | **Analysis Phase** | 7 days | Wed 8/8/18 | Thu 16/8/18 | | |
| 4 | | Plan User Interviews | 2 days | Wed 8/8/18 | Thu 9/8/18 | | |
| 5 | | Schedule users Interviews | 3 days | Sat 11/8/18 | Sun 12/8/18 | | |
| 6 | | Conducting users Interviews | 2 days | Mon 13/8/18 | Wed 15/8/18 | | |
| 7 | | **System Design** | 16 days | Sat 18/8/18 | Sat 8/9/18 | 6 | |
| 8 | | Modules Design | 10 days | Sat 18/8/18 | Tue 28/8/18 | | |
| 9 | | Data Structure Design | 3 days | Sat 1/9/18 | Mon 3/9/18 | 8 | |
| 10 | | User Interface Design | 3 days | Sat 15/9/18 | Tue 18/9/18 | | |
| 11 | | Coding Phase | 34 days | Wed | Sun | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | 19/9/18 | 4/11/18 | | |
| 12 | | **Testing Phase** | 9 days | Sun 4/11/18 | Mon 12/11/18 | | |
| 13 | | Integration Testing | 5 days | Sun 4/11/18 | Thu 8/11/18 | | |
| 14 | | System Level Testing | 4 days | Fri 9/11/18 | Mon 12/11/18 | | |
| 15 | | Implementation | 4 days | Tue 13/11/18 | Sat 17/11/18 | | |
| 16 | | Post-Implementation Review | 2 days | Sun 18/11/18 | Mon 19/11/18 | | |

## FUTURE SCOPE OF THE PROJECT

The proposed system helps them in many ways. It helps them in managing all the aspects related to student helpin an easy and effective manner. The software is provided with all the master entries to enter any new service, student, staff , or to add or modify and delete.

As this is generic software it can be used by a wide variety of outlets (Colleges and Universities/Schools) to automate the process of manually maintaining the records related to the subject of maintaining the student and facilities flows.

In future it can be modify, so that it can be done online. In order to meet these facility this is the major change which can be done in future regarding this project.

# References

## Reference Books

**Head First PHP & MYSQL- By Lynn Beighley & Michael Morrison**

**The Complete Reference- PHP- Steven Holzner**

## Reference websites

- **www.scribed.com**
- **www.w3schools.com**
- **www.javatpoints.edu**