



**A
PROJECT
ON
“College Management System”
BY**

Anoop Kumar

Under Guidance
of

Mr. Jayprakash Singh

Submitted to the School of Computer and Information Sciences in
partial fulfillment of the requirements

For the degree of

**Masters
of
Computer Applications**



Indira Gandhi National Open University

**Maidan Garhi
New Delhi – 110068.**

APPENDIX 2: PROFORMA OF PROJECT PROPOSAL



SCHOOL OF COMPUTER AND INFORMATION SCIENCES IGNOU, MAIDAN GARHI,
NEW DELHI – 110 068

PROFORMA FOR SUGGESTIONS OF MCSP60 PROJECT PROPOSAL

(Note: All entries of the proforma of suggestions should be filled in with appropriate and complete information. Incomplete proforma of suggestions in any respect will be summarily rejected.)

Enrolment No.:- 186954970

Study Centre :- Maharaja Agrasen College

Regional Centre :- Noida

RC Code :- 39

E-mail:- Anoopsaini421@gmail.com

Telephone No.:- 9870694481

1. Name and Address of the Student :- Anoop Kumar ,Dallupura (Delhi) 110096
2. Title of the Project :- College Management System
3. Name and Address of the Counsellor :- Mr. Jayprakash Singh , C-59 Block-4,Hanuman Road New Delhi
4. Educational Qualification of the Counsellor - BCA, MCA, Phd(pursuing)
5. Working / Teaching experience of the Counsellor :- Teaching/7.5 year of experience
6. Software used in the Project :- Node JS, Express JS, Visual Studio, SQLite3

Anoop Kumar

Mr. Jayprakash Singh

Date:

Date:

Suggestions for improving the Project:

APPENDIX 3: CERTIFICATE OF AUTHENTICATED WORK

This is to certify that the project report entitled **College Management System** submitted to **Indira Gandhi National Open University** in partial fulfillment of the requirement for the award of the degree of **MASTER OF COMPUTER APPLICATIONS (MCA)** is an original work carried out by Mr. **Anoop Kumar** enrolment no.**186954970** under my guidance. The matter embodied in this project is authentic and is genuine work done by the student and has not been submitted whether to this University or to any other University/Institute for the fulfillment of the requirement of any course of study.

Name : Anoop Kumar

Address: Dallupura Delhi (110096)

Name: Mr. Jayprakash Kumar

Address:- C-59, block-4

Hanuman Road ,New Delhi

APPENDIX 4: ROLES AND RESPONSIBILITIES FORM

Name of the Project:- College management System

| Name of the Team Member | *Role | Tasks and Responsibilities |
|--------------------------------|--|--|
| 1. Anoop Kumar | Development, Testing, Designing and Database | Run an Application on the system and Handle the all Application Functionalities. |

Name and Signature of the **Project Team members:**

1. Anoop Kumar

DETAILS OF PROJECT GUIDE

JAY PRAKASH KUMAR

C - 59, Block - 4,
Hanuman Road,
New Delhi – 110001.
Mb: +91-9999406033
Email: jpkumar541@gmail.com

Experience Summary

- ❑ 7.5 years of total Experience in Computer Science.
- ❑ Presently working with Simplex Solutions as Coordinator/Assessor cum Trainer from Feb 2016 to till.
- ❑ Presently working with Maharaja Agarsen College (IGNOU Centre code -07107) as Guest Faculty of Computer Science from Feb 2016 to till.
- ❑ Worked with Softdot Hi-tech Educational & Training Institute (JAMIA HUMDARD Center Code – 1010 & IGNOU Center code – 0766P) as Trainer cum Faculty from 23rd August 2013 to Feb 2016.
- ❑ Worked as Trainer (Lab/Theory) at AMR IT Consultancy July 2012 to 20 Aug 2013.

Work Summary

- ⇒ 5.5 years of result oriented experience in IT Industry.
- ⇒ Hands-on experience in leading, motivating and mentoring a team
- ⇒ Have created technical material for the trainings and delivered them successfully
- ⇒ Accomplished educator with demonstrated ability to teach, motivate, and instruct
- ⇒ Participants while maintaining high interest and achievement
- ⇒ Self-motivated with strong planning, organizational and leadership skills
- ⇒ Good communication and presentation skills

Technical Expertise

- **Operating System** : Windows, Linux
 - **Database** : Oracle 10g, SQL server.
 - **Language** : Java, C.
 - **Server Side Scripting:** Servlets, JSP, Java Scripts.
 - **Server Tools** : Tomcat 6.5, Glassfish.
 - **Design Pattern** : HTML, CSS.
 - **Frameworks** : Struts, Hibernate, Spring.
-

AMR IT Consultancy

Trainer (July 2012 to July 2013)

Responsibilities:

- To provide training classes on Java/J2EE along with DBMS(Oracle) for BCA, MCA, BTech, B. Sc (IT) and other professional courses.
- Conducting lab for Java/ J2EE, Database.
- Guide students in their projects which he/she prepares for academic propose in java technologies.
- Prepare test materials for internal test.

Partial List of the projects and clients handled:

- Corporate training in various engineering collage.
- Cargo & Courier Service Management – This project was for a cargo company. This project help the client to manage their resources, employee and services (like vehicles, man power and functionality). The technology used in this project was Servlet, Hybernate, Struts, JSP and Oracle 10g for database.
- Human Resource Management – This project was for a man power provider company. This project help the organization to manage employee based activates like Hiring of employee and their record keeping and take care of employee working in various places and various position at client side. The technology used in this project was Servlet, Core, JSP and JDBC (Oracle as database)
- Project guide of more than 20 students for their acedimic projects in various university.

Academic Qualification

- ❑ Pursing Phd in Computer Science from Suresh Gyan Vihar University.
 - ❑ Master of Computer Application (MCA) from IGNOU, Delhi with 60.33% Aggregate.
 - ❑ Bachelor of Computer Application (BCA) from IGNOU, Delhi with 60% Aggregate.
 - ❑ Intermediate in Science (PCM) from Bihar Intermediate Education Council, Patna, with 72% marks
 - ❑ Certificate course in Web Component Development Using Java Technologies (CJEV3) from NIIT Ltd. WITH 75% AGGREGATE.
 - ❑ Hardware (A+) & Networking (N+), MCP from Institute of Network and Technology
-

- **Others** : **Computer Networks, Software Engineering, SAD, Web Programming, Data Structures.**

Professional Experience

Simplex Solutions

Cordinator/Assessor cum Trainer (Feb 2016 to till)

Responsibilities:

- To conduct assessment of all trade under ICT for SDI Scheme.
- To prepare question paper for assessment.
- Uploading result of conducted assessment.
- Laising for assessment in East and North East state.
- To take care of all portal work of SDIS.
- To provide training of web protal work.

Maharaja Agarsen College (IGNOU Centre code -07107)

IT Faculty (Feb 2016 to till)

Responsibilities:

- To provide training classes on Java, C, Data Structure for BCA, MCA, CIT.
- To provide theory classes on SAD, Software Engineering, and Web Programming for BCA, MCA.
- Conducting lab for Java/ J2EE, Database, C Programming.
- Guide students in their projects which he/she prepares for academic propose in java technologies.
- Guide the student for their Assignment.

Softdot Hi-tech Educational & Training Institute

(IGNOU Center code – 0766P)

Trainer/Faculty (Aug 2013 to Jan 2016)

Responsibilities:

- To provide training classes on Java/J2EE along with DBMS(Oracle) for BCA, MCA, B.Tech, B. Sc (IT) students and other students who came for training purpose.
 - Conducting lab for Java/ J2EE, Database.
 - Guide students in their projects which he/she prepares for academic propose in java technologies.
 - Prepare test materials for test which is conducted by company to check student growth.
 - Conducting Seminars on different java concepts for students.
 - Organizing quiz competition for student on technical and current affairs questions.
-
-

Personal Profile

| | | |
|-------------------|---|---|
| Date of Birth | : | Mar-05-1983 |
| Father's Name | : | Late Shri Rajendra Prasad |
| Permanent Address | : | Village – Akabarpur, Post – Asthwan, P.S. - Asthawan Dist. – Nalanda (Bihar) Pin - 803107 |
| Nationality | : | Indian |
| Sex | : | Male |
| Hobbies | : | Music |
| Strengths | : | Positive Attitude, Co-Operative and Practical. |

Date:

(Jay Prakash Kumar)

PAN CARD OF PROJECT GUIDE



CERTIFICATE



VOTER ID



पता : SJE1713154

59, ब्लॉक-सी, टाइप -टी
कवाटर , हनुमान रोड,
दिल्ली

Address :
59, BLOCK -C, TYPE-II QTRS,
HANUMAN ROAD., DELHI

Date : 10/09/2013

40-नई दिल्ली विधानसभा निर्वाचन क्षेत्र
के निर्वाचन रजिस्ट्रार कार्यालय के
पत्राचार के अंगुली

Facsimile Signature of the Electoral
Registration Officer for 40-NEW DELHI
Assembly Constituency

यह कार्ड केवल यह अधिकृत रूप
से ही प्रयोग किया जा सकता है।
यदि यह कार्ड किसी अन्य व्यक्ति
के पास है, तो इसे तुरंत नष्ट कर दिया
जाएगा।
In case of change in address, mention this card
number in the relevant Form for including your name
in the roll at the polling station and send it to the
Electoral Registration Officer, New Delhi.

INDEX

1. INTRODUCTION

2. ANALYSIS

SYSTEM ANALYSIS

SYSTEM SPECIFICATIONS

3. DESIGN APPROACH

INTRODUCTION TO DESIGN
SRS

UML DIAGRAMS

DATA FLOW DIAGRAMS

E-R DIAGRAMS

4. PROJECT MODULES

5. IMPLEMENTATION

CONCEPTS AND TECHNIQUES

TESTING TEST CASES

6. CONCLUSION

7. FUTURE ENHANCEMENTS

8. BIBLIOGRAPHY

Introduction

Objective: - The main objective of college management system is to automate all functionalities of a college or university. Using this system you can manage all college management work like Department, Student data, section of College, instructor and student Admission details. Using this college management system you can view or update data and information about students and staff easily. This system helps in managing the activity like student admission, student ID, student studying year and semester.

The College Management System can be used to store student information, Department Information, instructor information and section information etc. Using this system you can add new student and their student name, total credits, Department name details. You can submit Students details and can check details anytime. You can create Student, Departments, Instructors and Section submits in this system. You can also add new student in the system and can check details of the student easily.

Project Overview:-

Departments Module: - In This Module We Have Showing the Department Name, Building, And Budget and also add the new Department and submit.

Student Module: - In this module we have showing the list of the students like- Student id, Name, Total credits, Instructor ID and Department name and also add the new student details and submit.

Section Module: - In this module we have showing the list of the Section like- id, Semester, Year and also add the new student details and submit.

Instructor Module: - In this module we have showing the list of the Instructor like- Name, Department name, Salary, and ID. And also add the new instructor and create the instructor.

Technology To be Used:

- **Software Platform:**
 - Visual Studio Code
 - OS:-Window 10 Professional
 - Editor:- Vs Code
 - Browser:- Google Chrome
- **Front-end:**
 - Express JS, Node JS
- **Back-end: SQLite3**
- **Hardware Platform**
 - RAM:- 4 GB
 - HDD:-25 GB disk space

Analysis

1. Existing System

College Management system currently uses a manual system for the management and maintenance of critical information. The current system requires numerous paper forms, with data stores spread throughout the management infrastructure. Often information (on forms) is incomplete, or does not follow management standards. Forms are often lost in transit between departments requiring a comprehensive auditing process to ensure that no vital information is lost. Multiple copies of the same information exist in the college management system and may lead to inconsistencies in data in various data stores.

2. Proposed System

The College Management System is designed for any sheet to replace their existing manual, Web based system. The new system is to control the following information; Student Information, Department Information, Section information, instructor information etc. These services are to be provided in an efficient, cost effective manner, with the goal of reducing the time and resources currently required for such tasks.

3. Objective of the System

The main objective of college management system is to automate all functionalities of a college or university. Using this system you can manage all college management work like Department, Student data, section of College, instructor and student Admission details. Using this college management system you can view or update data and information about students and staff easily. This system helps in managing the activity like student admission, student ID, student studying year and semester.

Design

1.1 Introduction:

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization. Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and

maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

SRS (Software Requirement Specification)

1.2 Introduction: This section is written to provide general information about our product "College Management System".

The main objective of our product is to maintain information about students, Departments and other activities like Student name, Year, student ID, Department Budget, Instructor and salary etc. The information is stored for decision making in the future for a business process within an organization. This is a desktop application.

1.3 Purpose: This document describes the software requirements specification (SRS) for the Collage Management System that provides the access and management of information of different modules in a collage-like Students, Instructor, Department and section. Our project is based on a database, which stores and maintains the information of different modules within the system. The advantage of the management system is to avoid entries in hard copies and it saves the burden of hard copies of data. The system is a Desktop Application and GUI for this system is developed in Node JS.

Departments Module: - In This Module We Have Showing the Department Name, Building, And Budget and also add the new Department and submit.

Student Module: - In this module we have showing the list of the students like- Student id, Name, Total credits, Instructor ID and Department name and also add the new student details and submit.

Section Module: - In this module we have showing the list of the Section like- id, Semester, Year and also add the new student details and submit.

Instructor Module: - In this module we have showing the list of the Instructor like- Name, Department name, Salary, and ID. And also add the new instructor and create the instructor.

Purpose of this document is to retrieve and analyze the ideas that define the product and requirements that the user needs. This document describes the details of our product, its parameter, and its goals. This SRS document describes the target, audience, user interface of product and Software/Hardware requirements of our product. This document also describes the problem we have faced during the designing and implementation of the product and also describes how we have solved this problem and make our product more efficient. The management system saves the human power and time cost to perform the same task. The data in the database can be saved for a long time and can be used for different purposes in the future. In management systems, there is a minor chance of losing the data. This document also defines how customers and users see our product and understand the functionality of the product. This document will help the developers/designers in case of maintenance of the software product.

1.4 Key Focus and Abbreviations

When writing this SRS for College Management System the following terminologies are used:

CMS: College Management System.

Node JS: Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux. **GUI:** Graphical User Interface.

DBMS: Database Management System.

SQLite3: Structured Query Language, used to create the database.

NPM: Node Package Manager.

Express JS: It is a Framework of Node JS.

CLI: Command Line Interface

Development Server: Running the application.

1.5 Problem Statement

Manual College systems were paper-based and difficult to maintain, expensive, more manpower required and unable to handle large records, the previous system was not efficient, not effective and there were issues of redundancy and consistency.

1.6 Proposed System

In this project, the system is proposed by understanding the issues in the existing system. In this management system the problems are solved that were in the previous system by shifting on a computerized system of the modern age. The database is used to store the data at the backend of the system. The graphical interface GUI is developed in HTML,JS, CSS. In a certain way get the data from the user and store it into the database. Reports of stored data are generated through Crystal reports. The system that is proposed provides consistent and redundancy free data in storage and should be more efficient.

1.7 The main modules which are focused on this project:

- 1- Student Module**
- 2- Section module**
- 3- Department module**
- 4- Instructor Module**
- 5- Create Department**
- 6- Create student**
- 7- Create section**

1.8 Project Scope

As Colleges are growing day by day more and more, and also increasing the complexity of storing information of students and related to the college system, they face many related issues: students name, id, salary details of instructor, etc.

This project is based on the educational institute system where this application gives maximum services in a single software product that is used by teacher and system administration. This project is based on a desktop application that is sharing information on different departments in a college.

1. Specific Requirements:

1.2 External interfaces:

1.2.1 Tools/Platforms Requirements

Front End: Visual Studio Code

Backend: SQLite3

2.1 Hardware Specification

CPU: Intel Dual Core

Total RAM: 4GB

Hard Disk: 500 GB

3.1 Software Specification:

Front End: Express JS, HTML, Java Script, CSS

Backend: SQLite3

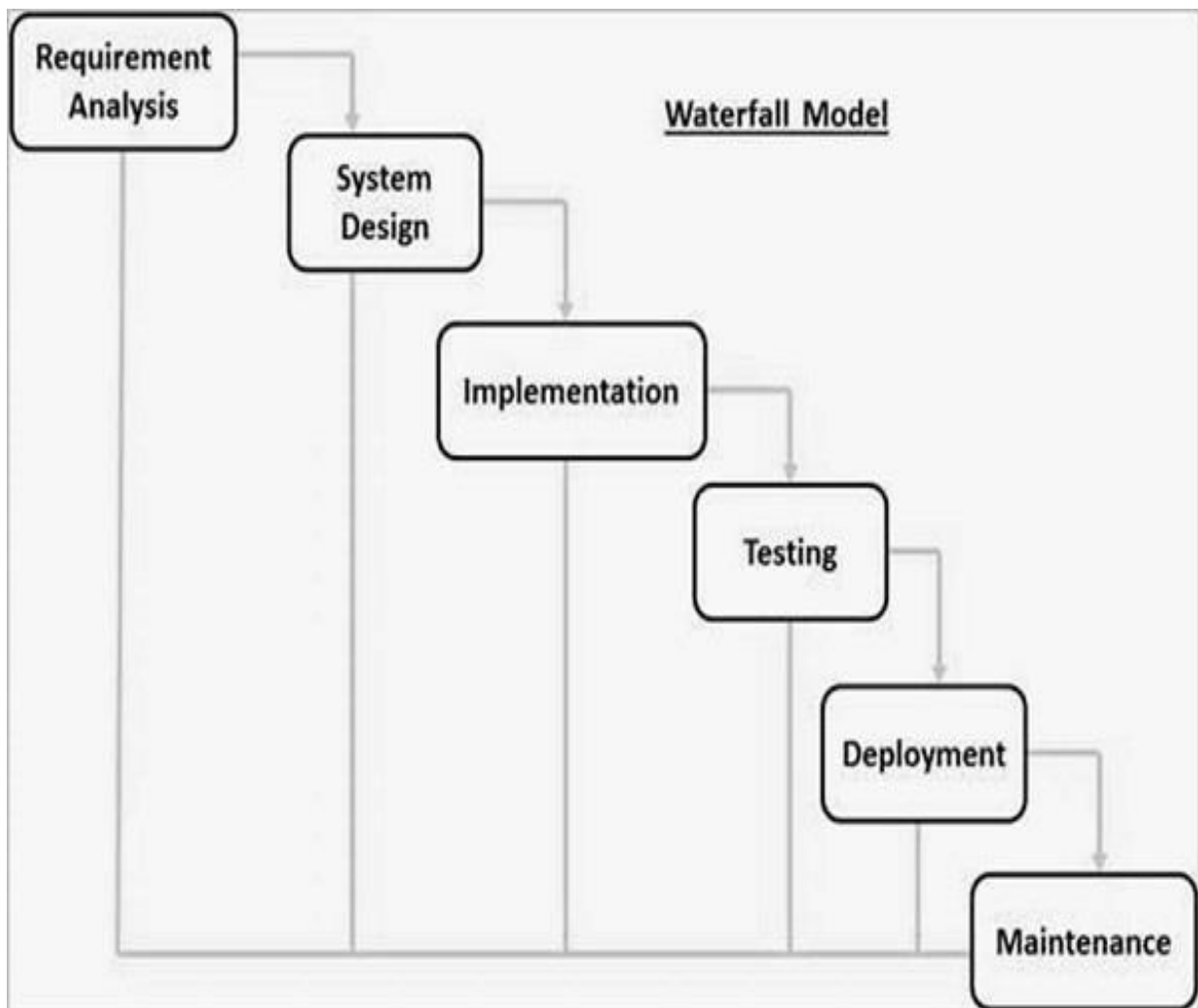
Operating System: Windows 10

4.1 Project Planning:

Waterfall Model:- The Waterfall Model was the first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases. The Waterfall model is the earliest SDLC approach that was used for software development. The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

Waterfall Model Design

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.



Waterfall Model (SDLC approach)

UML Diagrams:

Actor:

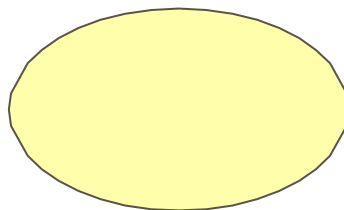
A coherent set of roles that users of use cases play when interacting with the use `cases.



Use case:

A description of sequence of actions, including variants, that a system

Performs that yields an observable result of value of an actor.

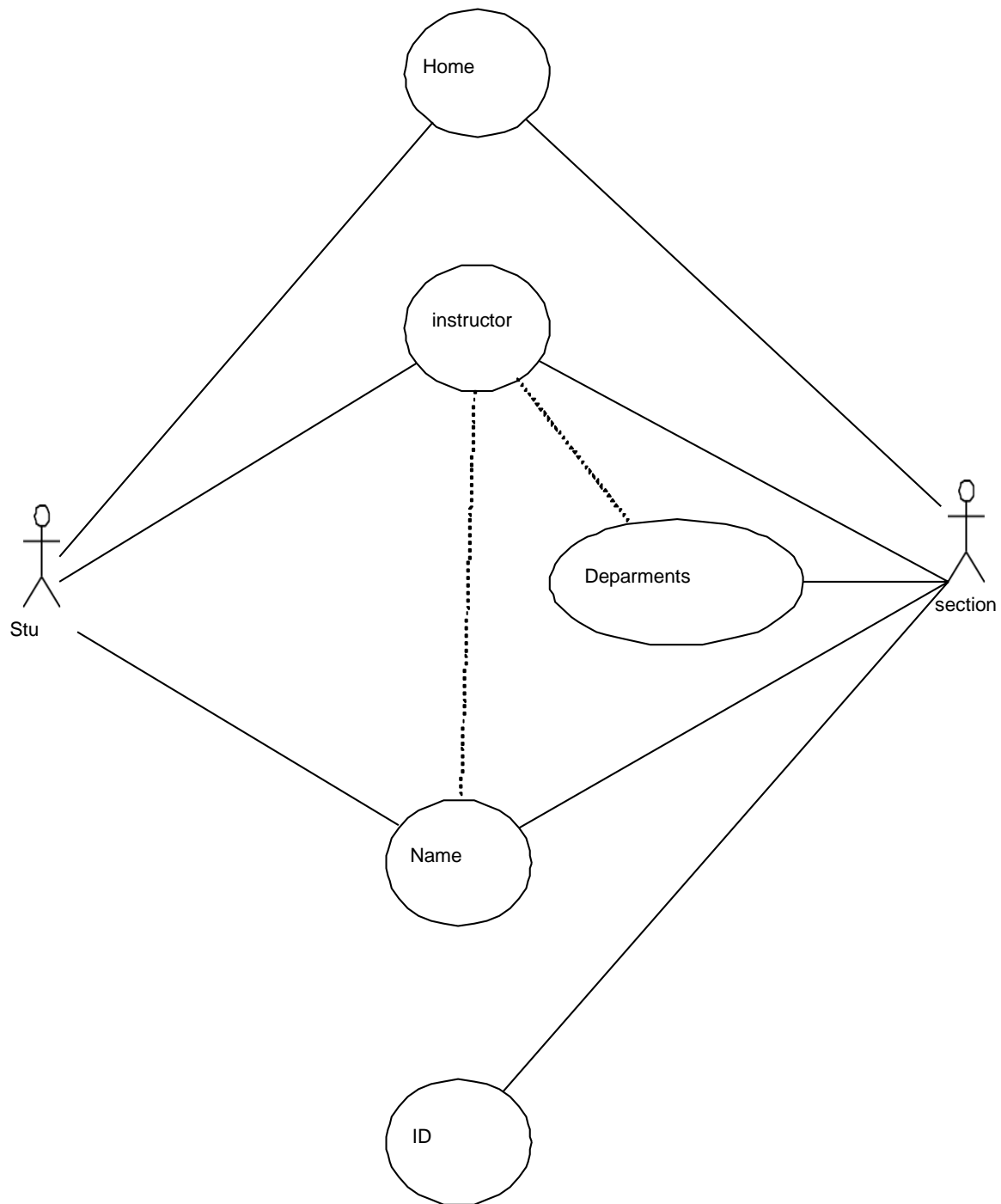


UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

There are various kinds of methods in software design: They are as follows

- Use case Diagram
- Sequence Diagram
- Collaboration Diagram

Use Case Diagram



Project Planning:-

Pert Chart: The pert chart showing the software development plan is shown below:

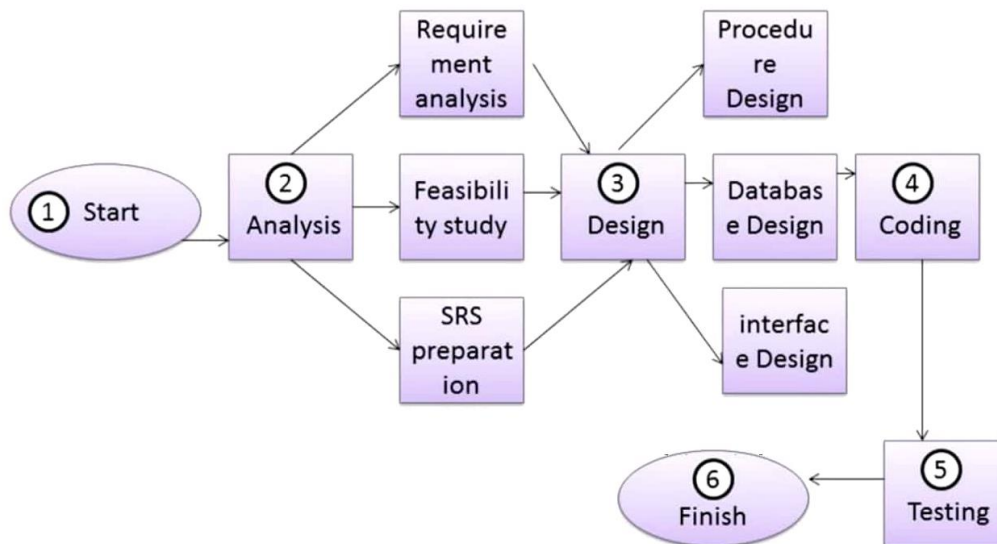


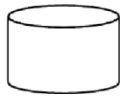


Fig: Pert Chart

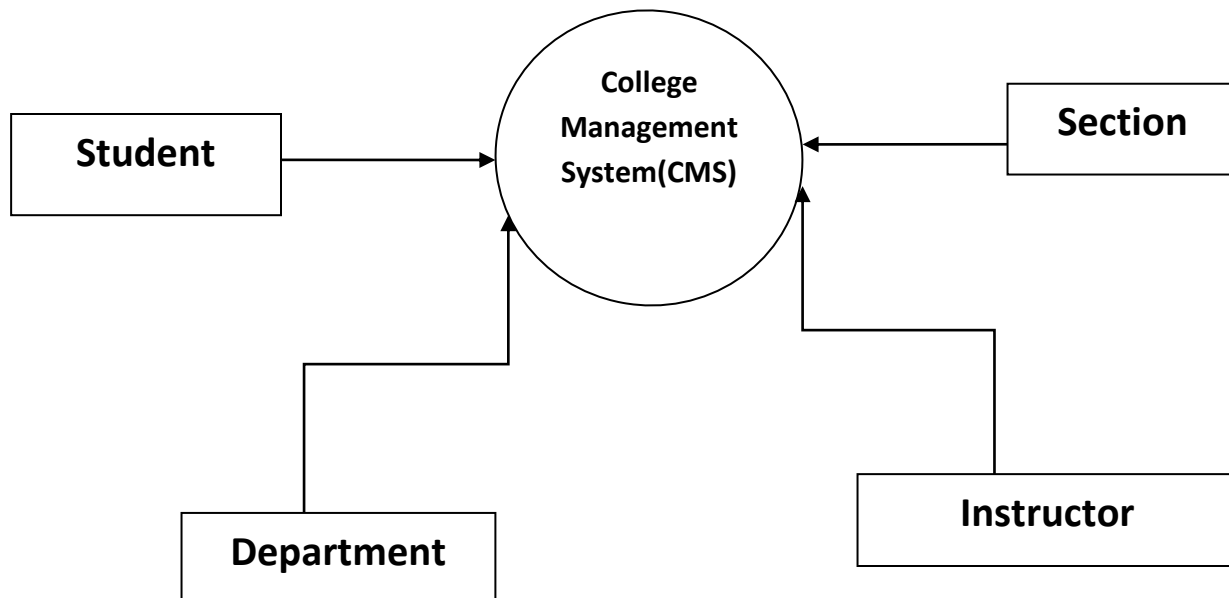
Data Flow Diagram: The Data flow Diagram shows the flow of data. The DFD can be up to several levels. The 0 level DFD states the flow of data in the system as seen from the outward in each module. The first level DFD show more detail, about the single process of the 0 level DFD. The second level DFD can show even more details and so on.

It is generally made of symbols given below:

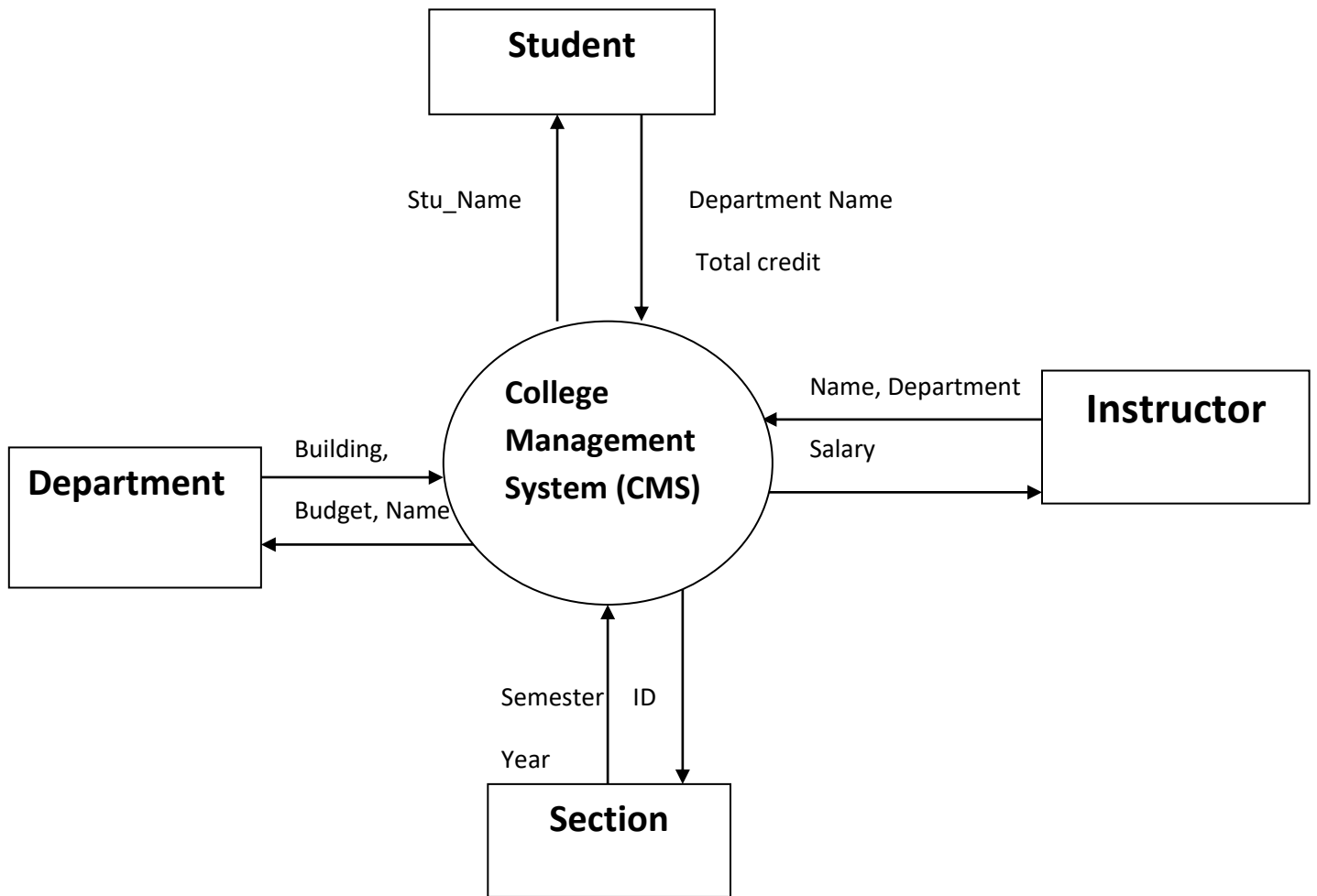
- (1) A **square** shows the Entity: - 
- (2) A **Circle** shows the Process: - 
- (3) An **Cylindrical Shape** shows the data store: --



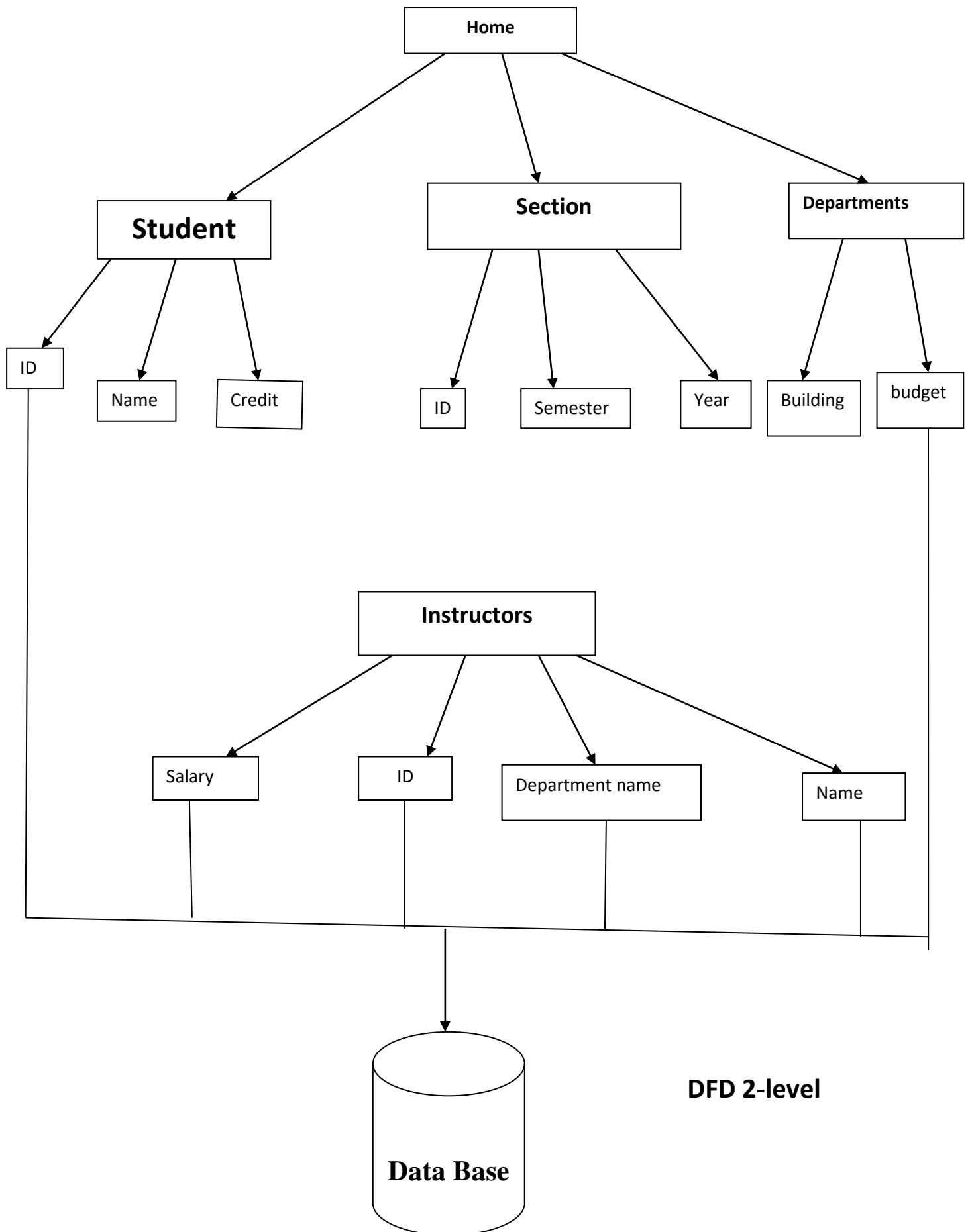
(4) An arrow shows the data How:-



Context Level DFD (0-Level)



DFD 1-level



DFD 2-level

ER-Diagram

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design. For the database designer, the utility of the ER model is:

- 1- It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
- 2- It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- 3- In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

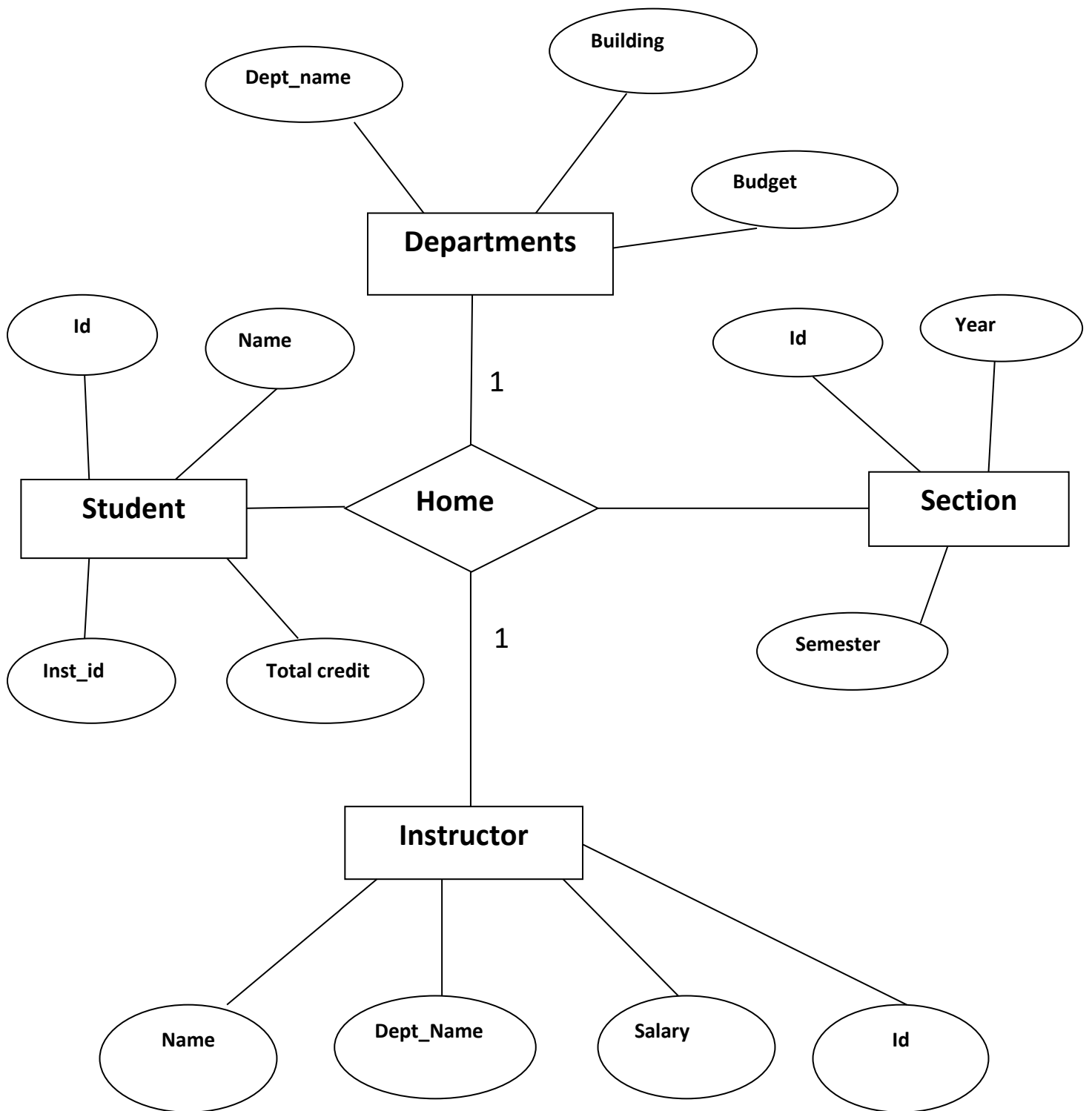
E-R Notation:

Entities are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.

Relationships are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs.

Attributes when included, are listed inside the entity rectangle. Attributes which are identifiers are underlined. Attribute names should be singular nouns.

Existence is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required.



ACTIVITY DIAGRAM: - is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The basic purpose of activity diagrams is to capture the dynamic behavior of the system.. It is also called object-oriented flowchart. This UML diagram focuses on the execution and flow of the behavior of a system instead of implementation. Activity diagrams consist of activities that are made up of actions that apply to behavioral modeling technology.

Activities

It is a behavior that is divided into one or more actions. Activities are a network of nodes connected by edges. There can be action nodes, control nodes, or object nodes. Action nodes represent some action. Control nodes represent the control flow of an activity. Object nodes are used to describe objects used inside an activity. Edges are used to show a path or a flow of execution. Activities start at an initial node and terminate at a final node.

Activity Diagram Notations

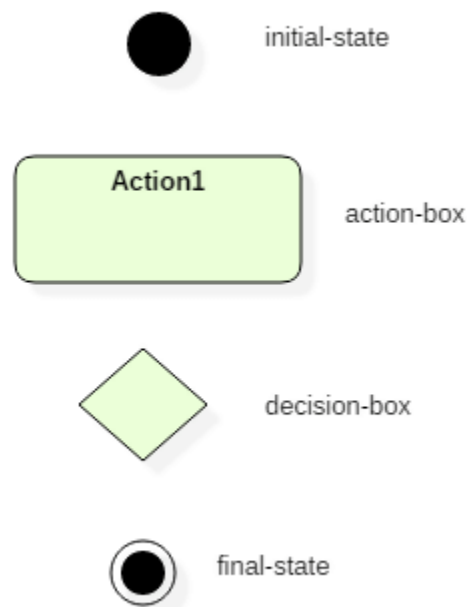
Activity diagrams symbols can be generated by using the following notations:

Initial states: The starting stage before an activity takes place is depicted as the initial state

Final states: The state which the system reaches when a specific process ends is known as a Final State

State or an activity box:

Decision box: It is a diamond shape box which represents a decision with alternate paths. It represents the flow of control.



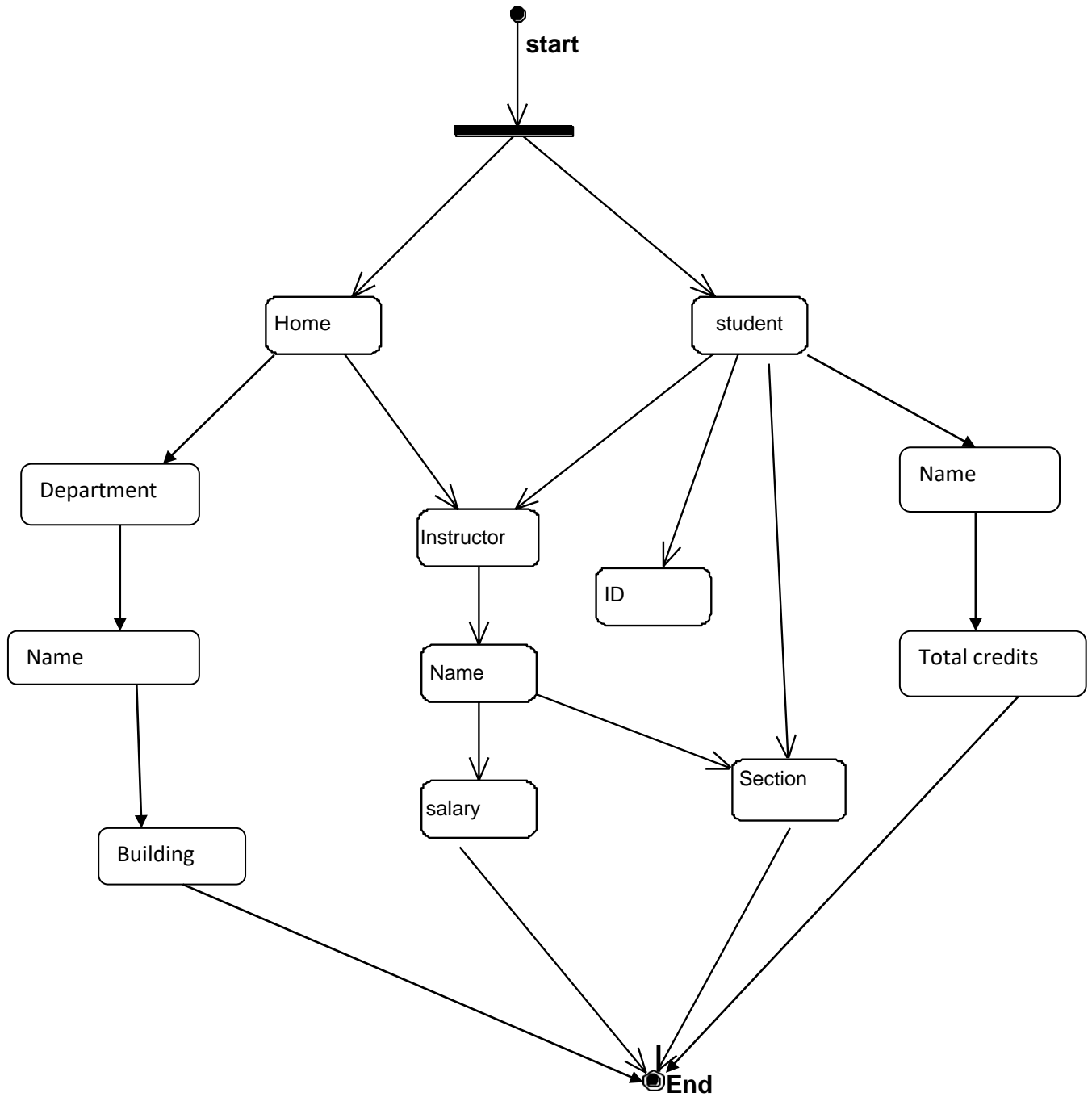
Why use Activity Diagrams?

Activity diagram in UML allows you to create an event as an activity which contains a collection of nodes joined by edges. An activity can be attached to any modeling element to model its behavior. Activity diagrams are used to model-

- Use cases
- Classes
- Interfaces
- Components
- Collaborations

Activity diagrams are used to model processes and workflows. The essence of a useful activity diagram is focused on communicating a specific aspect of a system's dynamic behavior. Activity diagrams capture the dynamic elements of a system.

Activity Diagram



Class Diagram:

The class diagram depicts a static view of an application. It represents the types of objects residing in the system and the relationships between them. A class consists of its objects, and also it may inherit from other classes. A class diagram is used to visualize, describe, document various different aspects of the system, and also construct executable software code.

It shows the attributes, classes, functions, and relationships to give an overview of the software system. It constitutes class names, attributes, and functions in a separate compartment that helps in software development. Since it is a collection of classes, interfaces, associations, collaborations, and constraints, it is termed as a structural diagram.

Purpose of Class Diagrams

The main purpose of class diagrams is to build a static view of an application. It is the only diagram that is widely used for construction, and it can be mapped with object-oriented languages. It is one of the most popular UML diagrams. Following are the purpose of class diagrams given below:

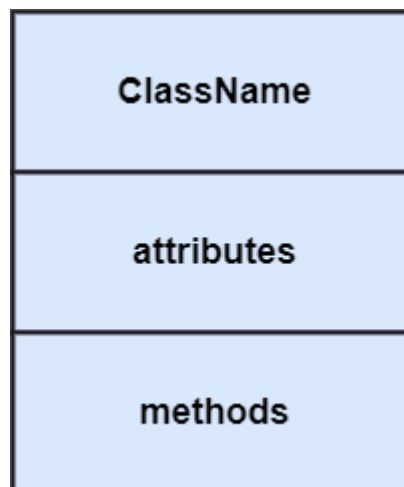
- It analyses and designs a static view of an application.
- It describes the major responsibilities of a system.
- It is a base for component and deployment diagrams.
- It incorporates forward and reverse engineering.

Benefits of Class Diagrams

1. It can represent the object model for complex systems.
2. It reduces the maintenance time by providing an overview of how an application is structured before coding.

3. It provides a general schematic of an application for better understanding.
4. It represents a detailed chart by highlighting the desired code, which is to be programmed.
5. It is helpful for the stakeholders and the developers.

Example:



Relationships

Dependency: A dependency is a semantic relationship between two or more classes where a change in one class cause changes in another class. It forms a weaker relationship.

Generalization: A generalization is a relationship between a parent class (super class) and a child class (subclass). In this, the child class is inherited from the parent class.

Association: It describes a static or physical connection between two or more objects. It depicts how many objects are there in the relationship.

Multiplicity: It defines a specific range of allowable instances of attributes. Incase if a range is not specified, one is considered as a default multiplicity.

Aggregation: An aggregation is a subset of association, which represents has a relationship. It is more specific then association. It defines a part-whole or part-of relationship. In this kind of relationship, the child class can exist independently of its parent class.

Composition: The composition is a subset of aggregation. It portrays the dependency between the parent and its child, which means if one part is deleted, then the other part also gets discarded. It represents a whole-part relationship.

Add new Stu

Student

-Id

-Name

-Total credits

-Instructor id

+Submit ()

can see

Instructors

-Name

-Department name

-Salary

-Id

+Submit ()

Section

-Id

-Semester

-Year

+Submit ()

Add New Dept

Department

-Department Name

-Building

-Budget

+Submit ()

Data Structure:-

Departments: - In This Module We Have Showing the Department Name, Building, And Budget and also add the new Department and submit.

Table 1: Departments

| Field Name | Data Type | Size | Constraints |
|------------|-----------|------|-------------|
| Dept_name | Text | 100 | Primary Key |
| Building | Text | 50 | Primary Key |
| Budget | Integer | 50 | Primary Key |

Student : - In this module we have showing the list of the students like- Student id, Name, Total credits, Instructor ID and Department name and also add the new student details and submit.

Table 2: Student

| Field Name | Data Type | Size | Constraints |
|---------------|-----------|------|-------------|
| Id | Integer | 50 | Primary Key |
| Name | Text | 100 | Primary Key |
| Total Credits | Integer | 50 | Primary Key |
| Inst Id | Integer | 50 | Primary Key |
| Dept Name | Text | 50 | Primary Key |

Section:- In this module we have showing the list of the Section like- id, Semester, Year and also add the new student details and submit.

Table 3: Section

| Field Name | Data Type | Size | Constraints |
|------------|-----------|------|-------------|
| Id | Integer | 100 | Primary Key |
| Semester | Integer | 50 | Primary Key |
| year | Integer | 50 | Primary Key |

Instructor : - In this module we have showing the list of the Instructor like- Name, Department name, Salary, and ID. And also add the new instructor and create the instructor.

Table 4: instructor

| Field Name | Data Type | Size | Constraints |
|------------|-----------|------|-------------|
| Name | Text | 100 | Primary Key |
| Deptname | Text | 50 | Primary Key |
| Salary | Integer | 50 | Primary Key |
| Id | Integer | 50 | Primary Key |

Modules And Process Logic

Modules in This Project:-

- Departments
- Student
- Section
- Instructors

Brief Description On the Modules:

CMS this is our DBMS project of a college Management System. Modules Used in my System is given Below:-

Departments: - In This Module We Have Showing the Department Name, Building, And Budget and also add the new Department and submit.

Student : - In this module we have showing the list of the students like- Student id, Name, Total credits, Instructor ID and Department name and also add the new student details and submit.

Section:- In this module we have showing the list of the Section like- id, Semester, Year and also add the new student details and submit.

Instructor: - In this module we have showing the list of the Instructor like- Name, Department name, Salary, and ID. And also add the new instructor and create the instructor.

4.5 Testing

Software design is a critical element of software quality assurance and represents the ultimate reviews of specifications, design and code generation. Once the source code has been generated then the software must be tested to uncover as many errors as possible before delivery to the consumer.

Unit Testing:- In the unit testing interfaces, local data structures, boundary conditions, independent paths, error-handling paths are testing. Test cases should be design to uncover errors due to erroneous computations, incorrect comparisons or improper control flow. For this Purpose basis path and loop testing is done.

System Testing:- Finally, we arrive at system testing where the software and the other systems elements are tested as a whole System testing verifies that all elements mesh properly and that overall system function/performance is achieved. Ultimately software is incorporated with other system elements and a series of system integration and validation tests are conducted.

5 Security Implementation:- In the Software Security there are two levels of security. The 1st level of security, which is provided by the FRONT END and 2nd level of security provided by the DATABASE which is being used.

Front End: We have implemented **in build html5 tags** for all the clients' side validations. The form is not submitted until users fills in correct data. It is extremely useful to restrict mistake by user.

Back End: We have used SQLite3 as the backend. SQLite3 server provides efficient/effective solution for major database tech.

- a. Large database and space management
- b. Many concurrent database users.
- c. High transactions processing requirement
- d. High Availability
- e. Industry accepted standards
- f. Manageable security
- g. portability

CONCLUSION

The main objective of our product is to maintain information about students, Departments and other activities like Student name, Year, student ID, Department Budget, Instructor and salary etc. The information is stored for decision making in the future for a business process within an organization. This is a desktop application.

Future Enhancement:

- Online examination module would be introduced to conduct online examination.
- Scheduling of the staff. i.e. , time table setting of the staff
- Further, the faculty can upload the videos of their lectures on to this site and students who had missed those classes can view those videos.

BIBLIOGRAPHY

Websites:

- <https://nodejs.org/en/>
- <https://reactjs.org/>
- <https://expressjs.com/>
- <https://www.tutorialspoint.com/index.htm>
- <https://www.sqlite.org/index.html>
- <https://www.javascript.com/>
- <https://www.youtube.com/>