A Practicum Report Submitted By

# **Morium Akter**

ID # 16203057

In Partial Fulfillment of the Requirements for the degree of Bachelor of Computer Science and Engineering



# Department of Computer Science and Engineering

College of Engineering and Technology

IUBAT – International University of Business Agriculture and Technology

# Spring 2020

IUBAT IT Society System

Morium Akter

A practicum report submitted in partial fulfillment of the requirements for the degree of Bachelor of Computer Science and Engineering (BCSE)

The practicum has been examined and approved,

Prof Dr Md Abdul Haque Chair and Professor

Dept. of Computer Science and Engineering

IUBAT – International University of Business Agriculture and Technology

Prof Dr Utpal Kanti Das Professor and Coordinator

Dept. of Computer Science and Engineering

IUBAT – International University of Business Agriculture and Technology

Mohammad Sajid Shahriar

Senior Lecturer

Dept. of Computer Science and Engineering

IUBAT – International University of Business Agriculture and Technology

## Letter of Transmittal

11th April, 2020 The Chairman,

Practicum and Placement Board College of Engineering and Technology - CEAT

IUBAT — International University of Business Agriculture and Technology

4 Embankment Drive Road, Sector – 10, Uttara Model Town, Dhaka-1230, Bangladesh

Subject: **Letter of Transmittal.**

Sir,

With due respect, I would like to approach you that it is a great opportunity as well as immense pleasure for me to submit this proposal titled “IUBAT IT Society System ” for the fulfillment of my Practicum course.

It was undoubtedly a splendid opportunity for me to work on this project to actualize my theoretical knowledge and has an enormous exposure with the corporate culture of a renowned company. Now I am looking forward for your kind appraisal regarding this practicum report.

I shall remain deeply grateful to you if you kindly go through this report and evaluate my performance.

Thanking you

Morium Akter

ID# 16203057

Program: BCSE

## Student’s Declaration

I am Morium Akter, student of IUBAT – International University of Business Agriculture and Technology, declaring that this project paper on the stated topic has only been prepared for the fulfillment of practicum defense.

It has not been prepared for any other purpose, reward, or presentation.

……………………….

Morium Akter

ID# 16203057

Program: BCSE

## Supervisor’s Certification

This is to certify that Practicum report on “IUBAT IT Society System.” has been carried out by **Morium Akter** bearing ID# 16203057 of IUBAT – International University of Business Agriculture and Technology as a partial fulfillment of the requirement of practicum defense course. The report has been prepared under my guidance and is a record of the accomplished work carried out successfully. To the best of my knowledge and as per her declaration, no parts of this report has been submitted anywhere for any degree, diploma or certification.

Now she is permitted to submit the report. I wish her success in all her future endeavors.

Practicum Supervisor

Mohammad Sajid Shahriar

Senior Lecturer

Department of Computer Science and Engineering

UBAT- International University of Business Agriculture and Technology

## Departmental Certification

On behalf of the Department of Computer Science and Engineering of International University of Business Agriculture and Technology (IUBAT University) we, the undersigned, certify that this practicum report “IUBAT IT Society System ” for the award of Bachelor of Computer Science and Engineering (BCSE) degree was duly presented by Morium Akter(ID No. 16203057) and accepted by the department.

Mohammad Sajid Shahriar

Senior Lecturer

Prof Dr Utpal Kanti Das

Professor and Coordinator

Prof Dr Md Abdul Haque

Chair and Professor

**Acknowledgements**

At first I want to give thanks to Allah who is the most merciful and the most graceful. I am extremely grateful to Late Prof Dr. M Alimullah Miyan, the Vice-Chancellor for taking me as a student in IUBAT. I also want to thank Prof Dr.Abdur Rab, the present Vice-Chancellor of IUBAT. It is also an immense pleasure for me to thank for assigning me that project.

I am heartily thankful to my supervisor, Mohammad Sajid Shahriar, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the subject.

Furthermore, I would like to thank Prof Dr. Utpal Kanti Das, Mohammad Sajid Shahriar, Md. Rashedul Islam; Arifa-Tur-Rahman because the knowledge and idea obtained from their previous courses helped me a lot to complete this project.

After that I would like to express gratitude to the organization “Xubisoft Ltd.” while I have worked during the last four months as an intern. The members of this organization have given their best to make the project successful.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of the project.

## Dedication

This humble effort, the fruit of my thoughts and study is dedicated to the people who have always been there to encourage and support me and especially to my beloved parents whose affection, love, and prays of day and night make me able to get this project done.

I would also like to dedicate this to my parents, who have inspired me throughout my university.

## Abstract

IUBAT IT Society System is designed to make it easier for keeping record all activities of Society and also generating the report of fund calculation. This project serves three actors Admin, Coordinator and User. All users can upload notice easily using this system. Mentoring course and mentor also include in this system. User can apply for mentor and admin will select or reject those application and System will generate schedule of mentoring course. This system will keep record details of every event and funding calculation. Here the admin can also add new admin to manage this system. The purpose of this project is to reduce time consumption and human effort. This report is for practicum defense. The main purpose of this project is to learn how to conduct a project and work in real field and write it down in a formal and specific way. The secondary objective of this project is to learn how the data collection process is managed and achieved. The report describe how the project has been built. I used php for backend, mysql for database and html & css for frontend.

## Internship Certification

|  |
| --- |
| **Table of Contents** |
|  | **Page No** |
| Title Fly | i |
| Letter of Transmittal | iii |
| Student’s Declaration | iv |
| Supervisor’s Certification | v |
| [Departmental certification](#_bookmark0) | vi |
| [Acknowledgements](#_bookmark1) | vii |
| Dedications | viii |
| Abstract | ix |
| Internship Certification | x |
| **Chapter 01: Project Introduction** |
| 1.1 Project Overview |  |
| 1.2 Objectives |  |
| 1.2.1 Broad Objectives |  |
| 1.2.2 Specific Objectives |  |
| 1.3 Scope of the Project |  |
| 1.4 Benefits of the System |  |
| 1.5 Methodology |  |
| 1.5.1 Data Sources |  |
| 1.6 Software Development Process Model |  |
| 1.6.1 Reason for Choosing |  |
| 1.7 Feasibility Study |  |
| 1.7.1 Technical Feasibility |  |
| 1.7.2 Economic Feasibility |  |
| 1.7.3 Operational Feasibility |  |
| **Chapter 02: Organization Part** |
| 2.1 Organization Overview |  |
| 2.2 Organization Services |  |
| 2.3 Organization Location |  |
| 2.4 Organization Vision |  |
| 2.5 Organization Mission |  |
| 2.6 My position in this Organization |  |
| 2.7 Organization Structure |  |
| **Chapter 03: Requirement Engineering** |
| 3.1 Requirement Analysis |  |
| 3.2 Requirements Engineering |  |
| 3.2.1 User & System Requirements |  |
| 3.3 Functional Requirements |  |
| 3.4 Non-Functional Requirements |  |
| 3.5 Use Case Diagram of the System |  |
| 3.5.1 Use Case Symbols |  |
| 3.5.2 Use case diagram of the project |  |
| **Chapter 04: System Planning** |
| 4.1 Function of the Proposed System |  |
| 4.2 Function Point Estimation |  |
| 4.2.1 Complexity Matrix |  |
| 4.3 Identifying complexity |  |
| 4.3.1 Identifying complexity of transition function |  |
| 4.3.2 Identifying complexity of data function |  |
| 4.3.3 Unadjusted function point contribution |  |
| 4.3.4 Performance and Environmental impact |  |
| 4.4 Processed Based Estimation |  |
| 4.4.1 Effort Distribution |  |
| 4.4.2 Project Schedule Chart |  |
| 4.5 Cost Estimation |  |
| **Chapter 05: Risk Management** |
| 5.1 Risk Management |  |
| 5.2 The RMMM Plan |  |
| **Chapter 06: Analysis Modeling** |
| 6.1 Software analysis pattern |  |
| 6.2 Activity Diagram |  |
| 6.3 Swimlane Diagram |  |
| **Chapter 07: Designing** |
| 7.1 Interface design |  |
| 7.2 Data Flow Diagram |  |
| 7.2.1 DFD of the Project |  |
| 7.3 Entity Relationship Diagram |  |
| 7.4 Database Design |  |
| **Chapter 08: Quality Assurance** |
| 8.1 System Quality Management |  |
| 8.2 System Quality Management Processes |  |
| 8.3 System Testing |  |
| 8.3.1 White Box Testing |  |
| 8.3.2 Black Box Testing |  |
| 8.4 Testing |  |
| **Chapter 09: Conclusion** |
| 9.1 Practicum and Its Value |  |
| 9.2 Conclusion |  |
| 9.3 Limitations |  |
| 9.4 Future Plan |  |
| **References** |  |

|  |  |
| --- | --- |
| **List of Figures** | |
|  | **Page No** |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **List of Tables** | |
|  | **Page No** |
|  |  |

**IUBAT IT Society System.**

**Chapter: 01**

**Project Introduction**

## Project Overview

“IUBAT IT Society System” is safe and very much secure, which is reliable, available and easily accessible to users. There will be three types of users with authorization power for login the system.

System admin will handle the all information in the system. Admin have the permission to access all the Notice information, Member information, Finance information, Event information and Fund information. The ultimate objective of the system is to provide facility to the department and the users for managing this society. Security of this system is very high and the possibility of doing wrong in the calculation is low.

## Objectives

The objective of developing such a computerized system is to reduce the paper work and safe of time for member as well as the student . There by increasing the efficiency and decreasing the work load.

The system provides student the information about notice as well as event and member information. The system must provide the flexibility of generating the required documents on screen as well as on printer when required.

## Broad Objectives

The broad objective of this project is to use my institutional educational experience in the real life working environment by IUBAT IT Society System.

## Specific Objectives

* + - * To make an automated system that can handle update of every Notice information , Mentoring class information , Member information and Fund information.
      * To reduce time wastage that occurred in manual system.
      * To make a system that will be very user friendly.
      * To develop a secure system.

## Scope of the Project

All approved individual can utilize the software with their username, password. Using this software package authority can keep track of Member, Notice, Mentoring-class, Event and Fund information. Software should have the feature of storing every information without any inconvenience.

## Benefits of the System

The following benefits are focusing here:

* User friendly
* Secure the data
* Authority can control the system
* Efficiency rate is high
* Reduce the wastage of time
* Provide better service than manual system
* Calculation will be automatically done
* Reduce the damage or chance of losing data

## Methodology

For this project in data collection phase I collected primary and secondary data. I collected all type of primary and secondary data from IUBAT IT Society those are needed to develop the system . The procedures and processes that I followed to develop this system are clearly described in the Analysis and Design chapter with illustrations.

## Data Sources

For this project in data collection phase I collected two types of data

* + - * Primary Data
      * Secondary Data

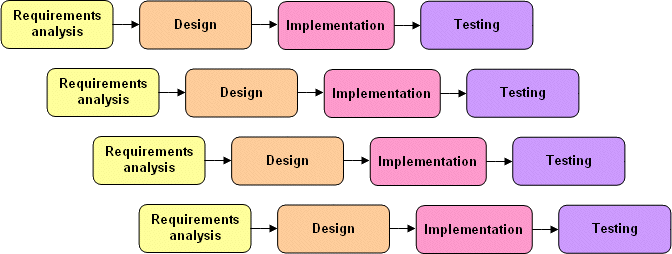
**Primary data** are gathered from IT Society, Student and faculty. The IUBAT’s practical experience, observation and face to face interview with Students, Faculty and our web Administrators Officer helped me to generate the primary data.

**Secondary data** are gathered from internet because I read some documentation of different club management system those have similarity with my system.

## Software Development Process Model

There are many situations in which initial software requirements are reasonably well-defined, but the overall scope of the development effort precludes a purely linear process, In addition, there may be a compelling need to provide a limited set of software functionality to users quickly and then refine and expand on that functionality in later software releases, The

incremental process model is a method of software development where the model is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This model combines the elements of the waterfall model with the iterative philosophy of prototyping.



### Figure 1:Incremental Process Model

## Reason for Choosing

The Waterfall process models provide several benefits, particularly for this type of system .Some points are given below:

* + - * The requirements are very well known, clear and fixed.
      * Product definition is stable.
      * Technology is understood.
      * There are no ambiguous requirements
      * Ample resources with required expertise are available freely

## Feasibility Study

A feasibility study is an analysis of how successfully a project can be completed. Feasibility study is made to see if the project on completion will serve the purpose of the organization

for the amount of work effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability; which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new system is proposed it normally goes through a feasibility study before it is approved for development.

The document provide the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such a Technical, Economic and Operational feasibilities. The following are its features:

## Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must

go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:

* Is it possible to develop the proposed system using the current technical resource?
* If not, can current technical resources be upgraded or added to in a manner that fulfills the request under consideration?
* Is there technology in existence that meets the specifications?

Technical feasibility concerns about the hardware capability, reliability as well as availability also the skills of the team. This study looks at the hardware and software available to perform for the proposed system.

Requirements for System operation:

* Processor: Core i3 or higher.
* Ram: 2GB or higher.
* Hard Disk Space: 500GB or higher.
* System: Windows 7/8/10
* Technology: Server/Local server/MYSQL
* Software: Any Modern browser (Opera/Firefox/Chrome).

## Economic Feasibility

Economic feasibility determines whether the system is economically feasible or not. It also determines what extent a new system is cost effective. We consider whether the company will be able to pay for redesigning and the system will be cost effective or not. The proposed system is within the budget for development. By getting this information, it gives an indication of the system is economically possible for development.

## Operational Feasibility

Operational feasibility determines whether the proposed system can solve the problems as well as concerns about user acceptance, management support and requirements entities that mean it should be user friendly and easy to use. It will help to track down all the information. All behavioral aspects are considered carefully and conclude that the project is behaviorally and operationally feasible.

**Chapter: 02**

**Organization Part**

## Organization Overview

Xubisoft Ltd. is a Web and Android solutions provider at its core with the highly qualified developers having experience of more than 2 years in various and complex designs. Other than our core service like web design and development. Xubisoft Ltd. has satisfied the clients with the services like Mobile app design and development, Software and Mobile Testing, SEO and Social Media Designing & Development. They are customer centric and divert our efforts to act as a one stop solution provider in IT. In every area of our operations they work hard in understanding the Clients requirement and providing the solution.

## Organization Services

* + - Web Design & Development
    - ERP Solutions
    - Software Development
    - Apps Development
    - SEO & Digital Marketing
    - Game Development
    - Graphical Content Design

## Organization Location

House No: 19, Road No: 04, Sector: 04, Uttara, Dhaka, Bangladesh

## Organization Vision

* + - To build a trusted IT Companies in Bangladesh
    - To be the largest Web development Companies in Bangladesh**.**
    - To be the best choice for people when they want to Develop any Apps

## Organization Mission

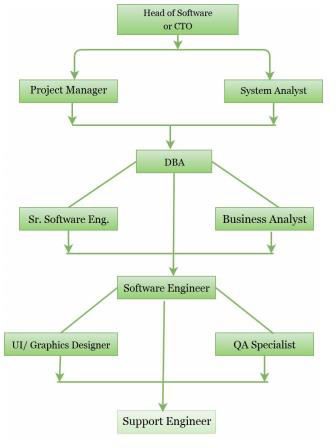
Produce excellent service in the field of IT Service, Software Development, Website Design

& Development, and Apps Development, SEO (Search Engine Optimization) and banking home and abroad, E-commerce and Consultancy with maximum effort driven toward customer satisfaction. Organizations Mission at a glance:

To achieve maximum customer satisfaction over the entire life cycle of our customer solution via our excellence of products and solutions.

* + - To consistently enhance our competitiveness and deliver profitable growth.
    - To practice highest standards of corporate governance and be a financially sound company.
    - To be a partner in nation building and contribute towards Bangladesh economic growth.
    - To encourage ideas, talent and value systems and become the customers of choice.
    - To earn the trust and confidence of all customers, exceeding their expectations.

## Organization Structure:



**Figure 2: Organizational Structure**

**Chapter: 03**

**Requirement Engineering**

## Requirement Analysis

Requirements analysis is a software engineering task that bridges the gap between system engineering and system design. Requirements analysis allows the software engineer to define the software allocation and build the models of the data, functional and behavioral domains that will treat by software. Requirement analysis provides the software designer with a representation of information, function and behavior of the system.

Requirements analysis is the first stage in the software development process. It encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users.

Analysis the requirement is critical to the success of a development project. Requirements must be actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design. Requirements can be functional and nonfunctional.

There are 6 phases of requirement analysis which is described below:

**Requirement Initiation:** I submit a proposal on the project entitled “IUBAT IT Society System”.

**Requirement Elicitation:** Eliciting requirements is the task of communicating with customers and users to determine what their requirements are. This is sometimes also called requirements gathering.

New systems change the environment and relationships between people, so it is important to identify all the users, take into account all their needs and ensure they understand the implications of the new systems.

I also try to understand the user's needs and constraints for the system. I analyze User’s work. In this phase I do mainly two works:

* Analyze the Requirement
* Recording User Requirements

**Requirement Elaboration:** This is the process of collecting user's needs and constraints. How the entities of the system will interact with each other.

**Requirement Negotiation:** This is the process of negotiating with the client about the software cost and other facilities will be provided with the system.

**Requirement Specification:** A software requirements specification (SRS) is a complete description of the behavior of the system to be developed. It includes a set of use cases that describe all of the interactions that the users will have with the software. In this stage I specified the technologies required for the deployment of the system.

**Requirement Verification:** I verify all the requirements of the user whether they need any modification or not.

## Requirements Engineering

Requirements engineering is, as its name suggests, the engineering discipline of establishing user requirements and specifying software systems. There are many definitions of Requirements Engineering; however, they all share the idea that requirements involves finding out what people want from a computer system, and understanding what their needs mean in terms of design. Requirements engineering is closely related to software engineering, which focuses more on the process of designing the system that users want.

* + 1. User requirements
    2. System requirements
    3. Functional requirements

(iv)Non-functional requirements

**User and System Requirement**

User Requirement #1

* Users can login

System requirements for user requirement #1

* + Fill username field
  + Fill password field
  + Click login
  + If information’s are correct then login successful otherwise unsuccessful
  + Login is successful then redirect to the dashboard
  + If any error occurs then redirect to the login page again. And show a Sign up option for create a account
  + After click the Sign up option show some field like: Name, Id, Email, Phone No. User need to fill up those field to create a account.

User Requirement #2

* Admin & User can upload notice & View Notice

System requirements for user requirement #2

* When admin click Notice button from dashboard, admin will see two top-down button : 1-Upload notice 2-View notice.
* After click Post notice, Admin & user will see some field. They can upload notice to fill up those field.
* At the bottom there are a Ok button. By clicking OK button admin & user can post notice.
* By clicking “view notice” , they will view all notice details

User Requirement #3

* Admin can approve and reject user notice

System requirements for user requirement #3

* When user upload notice, those notice will store on requesting notice .
* By clicking requesting notice option, admin can all requesting notice.
* There will be two option for approve and reject. Admin can approve notice by clicking approve option and reject notice by clicking reject option.

User Requirement #4

* Admin can select course for offering

System requirements for user requirement #4

* There are a course option .
* By clicking this option admin will see a field where admin have to insert semester name.
* After that admin will all course .Then admin can select course from those course for offering.

User Requirement #5

* Admin can manage mentoring section

System requirements for user requirement #5

* In mentoring option, there are 3 dropdown. 1-mentoring course, 2-apply section. 3-view mentoring schedule.
* By clicking “mentoring course”, admin can select course for mentoring class.
* By clicking apply option, admin can stop and start apply section
* Into this option , admin will all application for mentor and can select mentor.
* By clicking “view mentoring schedule” option, admin will see mentoring class scheduale.

User Requirement #6

* User can apply for membership and mentor .

System requirements for user requirement #6

* User will see apply button.
* After click apply button, user will see a form where has some field like: Name, Id, CGPA, Semester, Email, Phone No.
* After fill up those field ,At the bottom user click OK button for apply.

User Requirement #7

* User can see mentoring schedule and member details.

System requirements for user requirement #7

* In mentoring section, user will see one option “schedule”.
* By clicking this option, user will see mentoring class schedule.
* In member section , user can see member details.

User Requirement #8

* Admin can manage event

System requirements for user requirement #8

* In event section, there are two dropdown option: 1-add event 2-view details.
* By clicking “add event” option , admin add new event.
* By clicking “new event name” admin can distribute task into member.
* By clicking “view details” , admin will see every event details.

User Requirement #9

* Admin can application for add and deduct amount

System requirements for user requirement #9

* When admin click fund button from dashboard, admin will see three top-down button : 1-Add 2-deductt 3-see fund
* After click a add button, Admin see a form and fill up some field like: from where amount come, amount, purpose. At the bottom, there are a Ok button. After click Ok button this form direct go to coordinator.
* After click a deduct button, Admin see a form and fill up some field like: amount, purpose. At the bottom, there are a Ok button. After click Ok button this form direct go to coordinator.
* Admin see fund details by click on see fund details

User Requirement #10

* Only coordinator can manage transaction

System requirements for user requirement #10

* When coordinator see add form, if coordinator allow the form system atomically add amount on fund
* When coordinator see deduct form, if coordinator allow the form system atomically deduct amount from fund
* Coordinator see fund details by click on see fund details.

**Functional Requirement**

* Automated deduct balance when coordinator allow
* Automated add balance when coordinator allow
* Automated member generate after giving information
* Allow for editing any notice
* Generate required reports
* View notice for user
* Record task details for any event
* View member details
* View mentoring routine
* View Fund details
* Edit Notice
* Update mentoring routine
* Allow for Edit member details.
* Store all course.
* Store offer course in every semester.
* Allow for Edit mentoring details.
* Search semester name.

## Non-Functional Requirements

* System will have secure login system
* Password will be encrypted using password field.
* The system should be consistent and should give the desired results that means system

should be reliable

* The software should be efficient enough to take less memory of the computer system.
* There should not be any performance degradation.
* Show different kind of details.

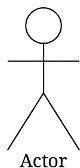
## Use Case Diagram of the System

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the

different use cases and will often be accompanied by other types of diagrams as well. The use case technique is used to capture a system's behavioral requirements by detailing scenario- driven threads through the functional requirements.

## Use Case Symbols

**Actor:** An actor represents a set of roles that users of use case play when interacting with these use cases. Actors can be human or automated systems.

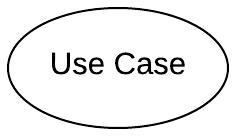


**Use case:** A use case represents a user goal that can be achieved by accessing the system or

software application. A use case is the specification of a set of actions performed by a system,

which yields an observable result that is typically of value for one or more actors of the

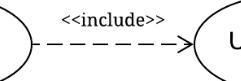
system.



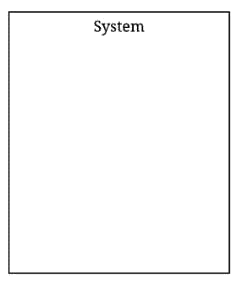
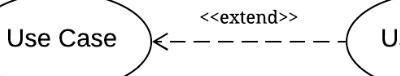
**Association:** Actor and use case can be associated to indicate that the actor participates in that use case. Therefore, an association correspond to a sequence of actions between the actors and use case in achieving the use case.



**Include:** An include relationship specifies how the behavior for the inclusion use case is inserted into the behavior defined for the base use case.

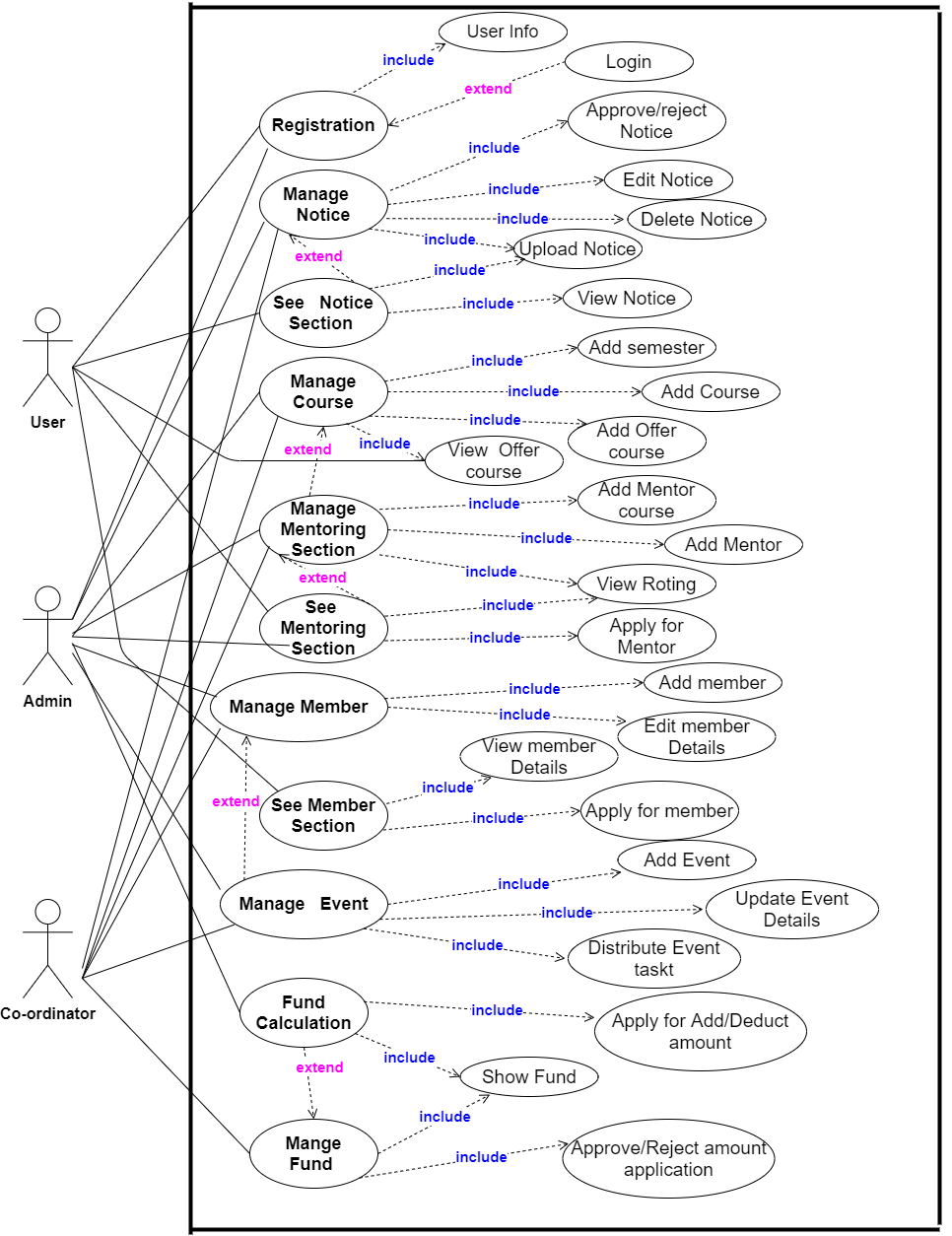


**Extend:** An extend relationship specifies how the behavior of the extension use case can be inserted into the behavior defined for the base use case.



**System:** The use cases of the system are placed inside the system shape, while the actor who interact with the system are put outside the system. The use cases in the system make up the total requiremen

**Use-Case Diagram**

****

**Figure 3: Use-case diagram**

**Chapter: 04**

**System Planning**

Project planning and scheduling chapter shows the functions of the project “Development of IUBAT IT Society System ”. The function point estimation, effort distribution and project schedule chart are also shown in this chapter.

## Function of the Proposed System

Table I: Function of Proposed System

|  |  |
| --- | --- |
| Login into the System | F1 |
| Upload Notice | F2 |
| Approve Notice | F3 |
| View notice | F4 |
| Add Semester | F5 |
| Add Offer Course | F6 |
| Add Mentoring course | F7 |
| Add Mentor | F8 |
| View mentoring routine | F9 |
| Add Member Details | F10 |
| Add Event | F11 |
| Distribute Event task | F12 |
| View event details | F13 |
| Manage Fund | F14 |

## Function Point Estimation

The task of counting function points should be included as part of the overall project plan. This is counting function points should be scheduled and planned. The first function point count should be developed to provide sizing used for estimating.

## Transactional Functions

* External Inputs [EI]
* External Outputs [EO]
* External Queries [EQ]

## Data Functions

* Internal Logical Files [ILF]
* External interface files [EIF]

Also FETs, DET, RET and FTR have been applied for the analysis of data functions and transactional functions.

## Complexity Matrix

Table II: Complexity Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| **EI** | **1-4 DETs** | **5-15 DETs** | **16 or more DETS** |
| **1 FTR** | **Low** | **Low** | **Average** |
| **2 FTRs** | **Low** | **Average** | **High** |
| **3 or More FTRs** | **Average** | **High** | **High** |

Table III: Complexity Matrix 2

|  |  |  |  |
| --- | --- | --- | --- |
| **EO/EQ** | **1-5 DETs** | **6-19 DETs** | **20 or more DETS** |
| **1 FTR** | **Low** | **Low** | **Average** |
| **2 to 3 FTRs** | **Low** | **Average** | **High** |
| **4 or More FTRs** | **Average** | **High** | **High** |

Table IV: Complexity Matrix for UFP

|  |  |  |
| --- | --- | --- |
| **Complexity** | **Transaction Function Type** | **Transaction Function Type** |
|  | EI/EQ | EO |
| **Low** | 3 | 4 |
| **Average** | 4 | 5 |
| **High** | 6 | 7 |

Table V: Complexity Matrix 3

|  |  |  |  |
| --- | --- | --- | --- |
| **ILF/EIF** | **1-19 DETs** | **20-50 DETs** | **51 or more DETS** |
| **1 RET** | **Low** | **Low** | **Average** |
| **2 to 5 RETs** | **Low** | **Average** | **High** |
| **6 or More RETs** | **Average** | **High** | **High** |

Table VI: Complexity Matrix for UFP 2

|  |  |  |
| --- | --- | --- |
| **Complexity** | **Transaction Function Type** | **Transaction Function Type** |
|  | **ILF** | **EIF** |
| **Low** | **7** | **5** |
| **Average** | **10** | **7** |
| **High** | **15** | **10** |

## Identifying complexity

* + 1. **Identifying complexity of transition function**

Table VII: Identifying complexity of transition function

|  |  |  |  |
| --- | --- | --- | --- |
| **Transition function** | **Fields/ File involve** | **FTRs** | **DETs** |
| 1. Login (EI) | Fields – user\_id, user name, phone\_number, email  File Name – users | 1 | 4 |
| 2.Upload Notice (EI) | Fields – n\_id ,User\_name, n\_title, n\_des, file\_name, uploaded\_on,status  File Name – notice,users,file | 3 | 7 |
| 3. View Notice(EO) | Fields – n\_id ,User\_name, n\_title, n\_des, file\_name, uploaded\_on,status  File Name – notice,users,file | 3 | 7 |
| 4.Update Brand (EI) | Fields – n\_id ,User\_name, n\_title, n\_des, file\_name, uploaded\_on,status  File Name – notice,users,file | 3 | 7 |
| 5. Approve Notice (EI) | Fields – n\_id ,User\_name, n\_title, n\_des, file\_name, uploaded\_on,status  File Name – notice,users,file | 3 | 7 |
| 6.Search notice (EQ) | Fields – n\_id ,User\_name, n\_title, n\_des, file\_name, uploaded\_on,status  File Name – notice,users,file | 3 | 7 |
| 7.Add Semester (EI) | Fields – s\_id, semester  File Name – semester | 1 | 2 |
| 8.View Semester (EO) | Fields – s\_id, semester  File Name – semester | 1 | 2 |
| 9.Add Course (EI) | Fields - c\_id, course\_code, course\_name, credit  File Name – course | 1 | 4 |
| 10.View Course (EO) | Fields - c\_id, course\_code, course\_name, credit  File Name – course, offer\_course, mentoring | 1 | 4 |
| 11.Search Course (EQ) | Fields - c\_id, course\_code, course\_name, credit  File Name – course, offer\_course, mentoring | 1 | 4 |
| 12. Add Offer Course (EI) | Fields – id, s\_id, course\_id, course\_code.  File Name – offer\_course, course, semester. | 3 | 4 |
| 13. View Offer Course (EQ) | Fields – id, s\_id, course\_id, course\_code.  File Name – offer\_course, course, semester. | 3 | 4 |
| 14. Add Mentoring Course (EI) | Fields – id, s\_id, course\_id, course\_code.  File Name – mentoring, offer\_course, semester. | 2 | 4 |
| 15. View Mentoring Course (EO) | Fields – id, s\_id, course\_id, course\_code.  File Name – offer\_course, course, semester. | 2 | 4 |
| 16. Add Mentor (EI) | Fields – m\_id, UserID, username,email,phone\_number, cgpa, Course\_Code, s\_id, status, room, starttime, endtime, day.  File Name – mentor, users, mentoring. | 3 | 13 |
| 17. View mentor (EO) | Fields – m\_id, UserID, username, email, phone\_number, cgpa, Course\_Code, s\_id, status, room, starttime, endtime, day.  File Name – mentor, users, mentoring. | 3 | 13 |
| 18. Add Member (EI) | Fields – id, userID, username, email, phone\_number, cgpa, designation, dpt, status.  File Name – member, user, designation | 3 | 9 |
| 19. View member (EO) | Fields – id, userID, username, email, phone\_number, cgpa, designation, dpt, status.  File Name – member, user, designation | 3 | 9 |
| 20.Update Member (EI) | Fields – id, userID, username, email, phone\_number, cgpa, designation, dpt, status.  File Name – member, user, designation | 3 | 9 |
| 21.Add Event (EI) | Fields - e\_id, event\_title, date  File Name – event\_title | 1 | 3 |
| 22.Distribute Event Task (EI) | Fields – task\_id,e\_id, task\_title, m\_name, amount, date  File Name- event\_task, event\_title,member. | 3 | 6 |
| 23. View Event Details (EO) | Fields – task\_id,e\_id, task\_title, m\_name, amount, date  File Name- event\_task, event\_title,member. | 3 | 6 |
| 24. Add Amount (EI) | Fields – a\_id, name,purpose, amount, date, status.  File Name – add\_amount | 1 | 6 |
| 25. Deduct Amount (EI) | Fields – d\_id, e\_id, name,purpose, amount,totalamount,date, status.  File Name – deduct\_amount,member. | 2 | 8 |
| 26. View Transaction(EO) | Fields – t\_id,a\_id,d\_id, type,amount,totalamount  File Name – transaction, add\_amount,deduct\_amount. | 3 | 6 |

**Identifying complexity of data function**

Table VIII: Identifying complexity of data function

|  |  |  |  |
| --- | --- | --- | --- |
| **Data function** | **Fields/File involve** | **RETs** | **DETs** |
| 1. users (ILF) | **Fields –** user\_id, user name, phone\_number, email. | 1 | 5 |
| 2.notice(ILF) | **Fields –** id, username,type, **n\_title, n\_des, file\_name, image, uploaded\_on, Status.** | 2 | 9 |
| 3.course (ILF) | **Fields – id, course\_code, course\_name, credit.** | 1 | 4 |
| 4. semester(ILF) | **Fields – id,**semester | 1 | 2 |
| 5. mentoring (ILF) | **Fields –** id, course\_id,s\_id,course\_code | 3 | 4 |
| 6. mentor (ILF) | **Fields- id,ID\_no,username, email, phone\_number, cgpa, Course\_Code, s\_id, status, room,**  **Starttime, endtime, day.** | 2 | 13 |
| 7. member (ILF) | **Fields – id,ID\_no, username, email, phone\_number, cgpa,**  **Designation, dpt,status** | 3 | 9 |
| 8. designation(ILF) | **Fields –** id, designation, dpt, staus | 1 | 4 |
| 9. event\_title (ILF) | **Fields – e\_id, event\_title,date** | 1 | 3 |
| 10.event\_task(ILF) | **Fields – task\_id, e\_id, task\_title, m\_name, amount, date** | 3 | 6 |
| 11.file (ILF) | **Fields – id, file\_name,n\_id** | 1 | 2 |
| 12. add\_amount(ILF) | **Fields – a\_id, name, purpose, amount, date,status** | 2 | 5 |
| 13. deduct\_amount(ILF) | **Fields – d\_id, e\_id, name, purpose, amount, totalamount, date,status** | 3 | 8 |
| 14. transaction(ILF) | **Fields – t\_id,a\_id, d\_id, type, amount, Total\_amount** | 2 | 6 |

## Unadjusted function point contribution

Table IX: Unadjusted function point contribution of transition function

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Transition function** | **FTRs** | **DETs** | **Complexity** | **UFP** |
| 1. Login (EI) | 1 | 4 | Low | 3 |
| 2.Upload Notice (EI) | 3 | 7 | High | 6 |
| 3. View Notice(EO) | 3 | 7 | Average | 5 |
| 4.Update Brand (EI) | 3 | 7 | High | 6 |
| 5. Approve Notice (EI) | 3 | 7 | High | 6 |
| 6.Search notice (EQ) | 3 | 7 | Average | 4 |
| 7.Add Semester (EI) | 1 | 2 | Low | 3 |
| 8.View Semester (EO) | 1 | 2 | Low | 4 |
| 9.Add Course (EI) | 1 | 4 | Low | 3 |
| 10.View Course (EO) | 1 | 4 | Low | 4 |
| 11.Search Course (EQ) | 1 | 4 | Low | 3 |
| 12. Add Offer Course (EI) | 3 | 4 | Average | 4 |
| 13. View Offer Course (EQ) | 3 | 4 | Low | 3 |
| 14. Add Mentoring Course (EI) | 2 | 4 | Low | 3 |
| 15. View Mentoring Course (EO) | 2 | 4 | Low | 4 |
|  |  |  |  |  |
| 16. Add Mentor (EI) | 3 | 13 | High | 6 |
| 17. View mentor (EO) | 3 | 13 | Average | 5 |
| 18. Add Member (EI) | 3 | 9 | High | 6 |
| 19. View member (EO) | 3 | 9 | Average | 5 |
| 20.Update Member (EI) | 3 | 9 | High | 6 |
| 21.Add Event (EI) | 1 | 3 | Low | 3 |
| 22.Distribute Event Task (EI) | 3 | 6 | High | 6 |
| 23. View Event Details (EO) | 3 | 6 | Average | 5 |
| 24. Add Amount (EI) | 1 | 6 | Low | 3 |
| 25. Deduct Amount (EI) | 2 | 8 | Average | 4 |
| 26. View Transaction(EO) | 3 | 6 | Average | 5 |
| Total | | | | 86 |

Table X: Unadjusted function point contribution for data functions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data function** | **RETs** | **DETs** | **Complexity** | **UFP** |
| 1. users (ILF) | 1 | 5 | Low | 7 |
| 2.notice(ILF) | 2 | 9 | Low | 7 |
| 3.course (ILF) | 1 | 4 | Low | 7 |
| 4. semester(ILF) | 1 | 2 | Low | 7 |
| 5. mentoring (ILF) | 3 | 4 | Low | 7 |
| 6. mentor (ILF) | 2 | 13 | Low | 7 |
| 7. member (ILF) | 3 | 9 | Low | 7 |
| 8. designation(ILF) | 1 | 4 | Low | 7 |
| 9. event\_title (ILF | 1 | 3 | Low | 7 |
| 10.event\_task(ILF) | 3 | 6 | Low | 7 |
| 11. add\_amount(ILF) | 2 | 5 | Low | 7 |
| 12. deduct\_amount(ILF) | 3 | 8 | Low | 7 |
| 13. transaction(ILF) | 2 | 6 | Low | 7 |
| Total | | | | 91 |

## Performance and Environmental impact

Table XI: Performance and environmental impact

|  |  |
| --- | --- |
| **GSC (General System Characteristics)** | **TDI** |
| 1. Data Communications | 4 |
| 2. Distributed Data Processing | 1 |
| 3. Performance | 4 |
| 4. Heavily Used Configuration | 2 |
| 5. Transaction Rate | 2 |
| 6. On-Line Data Entry | 1 |
| 7. End-user Efficiency | 2 |
| 8. Online Update | 1 |
| 9. Complex Processing | 2 |
| 10. Reusability | 2 |
| 11. Installation Ease | 4 |
| 12. Operational Ease | 3 |
| 13. Multiple Sites | 3 |
| 14. Facilitate Change | 1 |
| Total Degree of Influence (TDI) (Range 0 to 70 -> influence size ± 35%) | 32 |

Value Adjustment Factor (VAF) = (0.65 + (0.01\*TDI))

= (0.65 + (0.01\*32))

= 0.97

UFP = UFP (Data Function) + UFP (Transition Function)

= 91+86

=177

Adjusted Function Point (AFP) =UFP\*VAF

= 177\*0.97

= 171.69

Efforts for Project

= AFP \* Productivity

= 171.69 × 15.5

= 2661.2per hour

One person work 8 hour per day

=2661.2 /8 = 332.65 days

Approximate 333 days

In a group, there are 4 members

=333/4 =83.25 days

= Approximately 84 days

In a month , 25 days are working days.

=84/25=3.36 month

## Processed Based Estimation

**Table XII: Processed based Estimation**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Function Name** | **RG**  **(Client Visit))** | **Data Analysis** | **Risk Analysis** | **Engineering** | | **Developing** | | **Testing** | **Implementation**  **And Support** | **Total** |
| **Planning** | **Design** | **Coding** | **Documentation** |
| **F1** | **0.2** | **0.9** | **0.1** | **0.1** | **0.6** | **1.82** | **0.6** | **0.9** | **0.4** | **6.62** |
| **F2** | **0.3** | **1.0** | **0.21** | **0.21** | **0.72** | **2.1** | **0.72** | **1.32** | **0.52** | **7.1** |
| **F3** | **0.2** | **0.8** | **0.1** | **0.2** | **0.6** | **1.9** | **0.6** | **0.9** | **0.4** | **5.7** |
| **F4** | **0.2** | **0.7** | **0.1** | **0.1** | **0.5** | **1.5** | **0.5** | **0.7** | **0.3** | **5.6** |
| **F5** | **0.2** | **0.7** | **0.02** | **0.1** | **0.4** | **1.5** | **0.4** | **0.7** | **0.2** | **4.82** |
| **F6** | **0.4** | **1.1** | **0.1** | **0.2** | **0.7** | **2.2** | **0.7** | **1.31** | **0.5** | **6.51** |
| **F7** | **0.3** | **0.8** | **0.1** | **0.2** | **0.5** | **1.6** | **0.5** | **0.8** | **0.3** | **4.6** |
| **F8** | **0.4** | **1.1** | **0.11** | **0.3** | **0.7** | **2.3** | **0.7** | **1.5** | **0.5** | **6.91** |
| **F9** | **0.4** | **0.9** | **0.1** | **0.2** | **0.5** | **1.6** | **0.5** | **0.9** | **0.3** | **4.9** |
| **F10** | **0.2** | **1.2** | **0.2** | **0.3** | **0.8** | **2.3** | **0.8** | **1.5** | **0.6** | **8.1** |
| **F11** | **0.3** | **0.7** | **0.1** | **0.1** | **0.5** | **1.4** | **0.5** | **0.8** | **0.3** | **4.2** |
| **F12** | **0.4** | **1.0** | **0.1** | **0.1** | **0.7** | **2.1** | **0.7** | **1.2** | **0.5** | **7.1** |
| **F13** | **0.4** | **0.8** | **0.1** | **0.1** | **0.5** | **1.4** | **0.5** | **1.0** | **0.3** | **4.6** |
| **F14** | **0.5** | **1.5** | **0.32** | **0.33** | **1.4** | **2.7** | **1.4** | **2.41** | **1.03** | **11.19** |
|  |  |  |  |  |  |  |  |  | **Total =** | **87** |
| **Total** | **4.4** | **13.2** | **1.76** | **2.64** | **8.8** | **26.4** | **8.8** | **15.84** | **6.16** |  |
| **Percentage** | **5%** | **15%** | **2%** | **3%** | **10%** | **30%** | **10%** | **18%** | **7%** | **100%** |

## Effort Distribution

The software project estimation technique leads to estimate of work units required to complete the software development In this project, 40% of full software development has been allocated to Coding/Developing, 35% has been allocated to analysis and design and the remaining 25% has been allocated to software testing and support.

Figure 4: Effort Based Estimation

A detailed view of the effort distribution chart is illustrated bellow:

In the Pie chart we can see that, among 35% of Analysis & Design we spend 5% for client

visit and collect all requirement gathering, 15% for analysis the data and information, 2% for risk analysis, 3% for planning and 10% for designing.

Among 40% of Developing, 30% is used for coding and 10% is used for Documentation. And finally 18% is used for testing and last 7% is used for Implementing and giving the support to the client.

Figure 5: Effort Distribution PIE Chart

## Project Schedule Chart

Total system development is a combination of set of tasks. These set of tasks should done sequentially and timely. Project schedule works as the guideline of the system developer. The following is the schedule chart of this project:

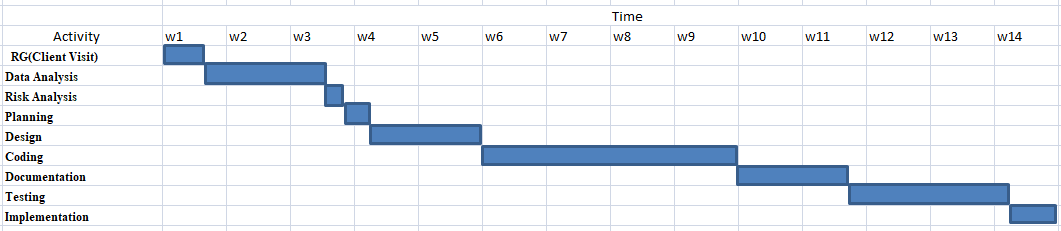


Figure 6: Project Schedule Chart

## Cost Estimation

The approximation of the cost of a program is cost estimation. In this project, there are few factors to analyze and calculate the cost. Given below,

* Personnel costs
* Software costs
* Hardware costs
* Other costs

## Personnel cost:

Number of days in a year = 365

Number of government holidays in a year =24 Number of weekly holidays in a year =52

Total number of working days to develop the project = 365-(52+24) = 289 days Total number of working days per months to develop the project = 289/12

=24.083 days

Organization working hours per day = 8 hours

Organization working hours per month=24.083\*8 = 192.664hours

Total working hour in 2.3 months = 24.083\*8\*3.36 = Approximately 647hours

Table XIII: Personnel Cost

|  |  |  |
| --- | --- | --- |
| **Position** | **Salary/month** | **Salary/hour** |
| System Analyst | 35000 | 182.29 |
| Designer | 25000 | 130.2 |
| Coder | 20000 | 104.17 |
| Tester | 20000 | 104.17 |

Total salary of Analyst in 3.36 months = TK 87\*182.29= TK 15859 Total Salary of Designer in3.36 months= TK 180\*130.2= TK 23436 Total Salary of Coder in 3.36 months= TK 290\*104.17= TK 30209

Total Salary of Tester in 3.36 months= TK 90\*104.17= TK 9375

Table XIV: Personnel Cost 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Designation** | **Person** | **Working Hour** | **Salary** | **Total**  **Salary** |
| System Analyst | One | 87 | 15859 | **78879 BDT** |
| Designer | One | 180 | 23436 |
| Coder | One | 290 | 30209 |
| Tester | One | 90 | 9375 |

## Hardware cost

Cost of a Computer = 35000

Computer life = 3 years

Computer Usage = 14 weeks = 3.36 months

Computer Cost = (35000/36) \* 3.36 \* 8= 26133.33 BDT

Table XV: Hardware Cost

|  |  |
| --- | --- |
| **Hardware** | **Cost** |
| Computer | 26133.33 |
| Modem | 1600 |
| Printer | 5000 |
| **Total** | **32733.33 BDT** |

Table XVI: Software Cost

|  |  |
| --- | --- |
| **Name** | **Amount** |
| Windows 10 | 8500 |
| MS Office 2016 | Free |
| XAMPP | Free |
| MySQL | Free |
| Notepad++ | Free |
| Total | **8500 BDT** |

Table XVII: Estimation of Other Cost

|  |  |
| --- | --- |
| **Name** | **Price (BDT)** |
| Transport | 2000 |
| House Rent | 10000 |
| Other | 1500 |
| Total | **13500 BDT** |

**Account Table**

Table XVIII: Accounts Table

|  |  |  |
| --- | --- | --- |
| Particulars | | TK |
| Salary- | |  |
|        | System Analyst Designer  Coder Tester | 15859  23436  30209 |
| 9375 |
|  | | **78879 BDT** |
| Hardware Cost – | |  |
|  | Computer | 26133.33 |
|  | Modem | 1600 |
|  | Printer | 5000 |
|  | | **32733.33 BDT** |
| Software Cost –   * Windows 10 * MS office 2016 * Xampp * MySQL * Notepad++ | | **8500**  **Free**  **Free**  **Free**  **Free** |
|  | | **8500 BDT** |
| Other Costs- | |  |
|  Transport | | **2000** |
|  House Rent | | **10000** |
|  Other | | **1500** |
|  | | **13500 BDT** |
| Total Cost | | **133612 BDT** |

**Chapter 05**

**Risk Management**

A risk is a serious problem that might or might not happen. It is necessary to analyze the potential risks in a project. If the risks of a software project are not properly analyzed and estimated, many problems can plague the software project. Anyone developing ant type of system encounter with it and it has to be managed.

## Risk Management

Risk analysis and management are a series of steps that help a software team understand and manage uncertainty. Many problems can plague of software project. A risk is a potential problem; it might happen, it might not. But regardless of the outcome, it’s a really good idea to identify it, assess its probability of occurrence, and estimate its impact, and establish a contingency plan should the problem actually occur. Risk analysis and management are a series of steps that help a software them to understand and manage uncertainty. To establish a risk management model the following phases are followed:

**Identification:** Risk identification is the process of detecting potential risks or hazards through data collection. A range of data collection and manipulation tools and techniques exists. The team is using both automated and manual techniques to collect data and begin to characterize potential risks to Web resources. Web crawling is one effective way to collect information about the state of Web pages and sites.

**Classification:** Risk classification is the process of developing a structured model to categorize risk and fitting observable risk attributes and events into the model. The team combines quantitative and qualitative methods to characterize and classify the risks to Web pages, Websites, and the hosting servers.

**Assessment:** Risk assessment is the process of defining relevant risk scenarios or sequences of events that could result in damage or loss and the probability of these events. Rosenthal describe the characteristics of a generic standard for risk assessment as "transparent, coherent, consistent, complete, comprehensive, impartial, uniform, balanced, defensible, sustainable, flexible, and accompanied by suitable and sufficient guidance.

**Analysis:** Risk analysis determines the potential impact of risk patterns or scenarios, the possible extent of loss, and the direct and indirect costs of recovery. This step identifies vulnerabilities considers the willingness of the organization to accept risk given potential consequences, and develops mitigation responses.

**Implementation:** Risk management implementation defines policies, procedures, and mechanisms to manage and respond to identifiable risks. The implemented program should balance the value of assets and the direct and indirect costs of preventing or recovering from damage or loss.

To take comprehensive care of a web-based system we must consider the following points:

* Hardware and software environment including any upgrades to the operating system and Web server, the installation of security patches, the removal of insecure services, use of firewalls, etc.
* Administrative procedures such as contracting with reputable service providers, renewing domain name registration, etc.
* Network configuration and maintenance including load balancing, traffic management, and usage monitoring.
* Backup and archiving policies and procedures including the choice of backup media, media replacement interval, number of backups made and storage location.
* Physical location of the server and its vulnerability to fire, flood, earthquake, electric power anomalies, power interruption, temperature fluctuations, theft, and vandalism.

There are different categories of risks that should be considered in any software project. The following categories of risks have been considered in this software project:

**Project risks:** These risks threaten the project plan. If these risks become real, it is likely that the project schedule will slip and that costs will increase. Project risks identify potential budgetary, schedule, personnel, resource, customer and requirement problems and their impact on the software project.

**Technical risks:** These risks threaten the quality and timeliness of the software to be produced. If a technical risk becomes a reality, implementation may become difficult or impossible. Technical risks identify potential design, implementation, interface, verification and maintenance problems. Moreover, specification ambiguity, technical uncertainty, technical obsolescence are also risk factors.

**Business risks:** These risks threaten the viability of the software to be built. The business risks can be market risks, building a system that no one really wants. Strategic risks, building a system that no longer fits into the overall business strategy for the company. Management risks, losing the support of senior management due to a change in focus or a change in people. Budget risks, losing budgetary or personnel commitment.

## The RMMM Plan

**Risk Mitigation:** Proactive planning for risk avoidance.

**Risk Monitoring:** Assessing whether predicted risks occur or not, ensuring preventive steps are being properly applied, collect information for future risk analysis, attempt to determine which risks caused which problem.

**Risk Management:** Actions to be taken in the event that mitigation steps have failed and the risk has become a live problem.

**Type of Impact:** Catastrophic (1), Marginal (2), Tolerable (3), Critical (4).

**Type of Probability:** very low (<10%), low (10–25%), moderate (25–50%), high (50– 75%),

very high (>75%).

**Project Risks:** Threaten the project plan. In my system, the bellow mentioned projects risks I needed manage.

Tabel XIX: Project Risk (P01)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Project Risk (P01)** | | **Date: 22-03-2020** | | |
| **Name** | | Changes the requirements | | |
| **Probability** | | Low (25%) | | |
| **Impact** | | Marginal (2) | | |
| **Description** | | Customer may change their requirements. | | |
| **Mitigation and Monitoring** | | Requirements are redefined by the company due to time or business needs. Meeting will be held with the company  regularly. This insures that the product we  are producing solves a problem | | |
|  |  | | |
| **Management** | Emergency meeting between both parties to identify new requirements and goals. | | |
| **Status** | There were changed some requirement and we done those successfully | |  |

Tabel XX: Project Risk (P02)

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Risk (P02)** | **Date: 10-03-2020** | | |
| **Name** | Poor Quality Documentation | |  |
|  |  |
| **Probability** | Low (15%) | |  |
| **Impact** | Catastrophic (1) | |  |
|  |  | |
| **Description** | Poor quality documentation of the members. | | |
| **Mitigation and Monitoring** | Meeting will documentation | be held routinely to offer suggestions and topics. | |
| The progress on documentation will also  have a monitor in each meeting. | | |
|  |  | |
| **Management** | The addition of new topics or removal of unnecessary topics into the documentation will assigned to responsible person. | | |
|  |  | | |
| **Status** | Monitoring it. | | |

**Technical Risks:** threaten product quality and the timeliness of the schedule. As this is my practicum project, therefore these types of risks need to be take care of properly.

Tabel XXI: Technical Risk (TR01)

|  |  |
| --- | --- |
| **Technical Risk (TR01)** | **Date: 25-02-2020** |
| **Name** | Computer Crash |
| **Probability** | Moderate (25-40%) |
| **Impact** | Catastrophic (1) |
| **Description** | Computer may crash due to several reasons |
| **Mitigation and Monitoring** | We should take proper follow up of computers. We also take regular data backup every day, We can use IPS to stop  unexpected shutdown. |
| **Management** | If our computer has been crashed then we will restore backup. |
| **Status** | We have not encountered such issue yet |

Tabel XXII: Technical Risk (TR02)

|  |  |
| --- | --- |
| **Technical Risk (TR02)** | **Date: 17-02-2020** |
| **Name** | Poor Training Skill in Team Members. |
| **Probability** | Moderate (30%) |
| **Impact** | Catastrophic (1) |
| **Description** | Poor Training Skill in Team Members to Train the Client. |
| **Mitigation and Monitoring** | The training team should have a clear knowledge about the entire functionality of the software. System analyst need to  ensure and monitor it while training session start. |
| **Management** | We should arrange a meeting with the train team and come to a point to solve this problem. |
| **Status** | We have not encountered such issue yet |

**Business Risk:** Threaten the viability of the software to be built (market risks, strategic risks, management risks, budget risks). As I am developing it as my practicum project by myself, classic business risks won’t be encountered here. The Probability of all type of Business Risks is therefore, determined as Low.

Tabel XXIII: Business Risk (B01)

|  |  |
| --- | --- |
| **Business Risk (B01)** | **Date: 09-02-2020** |
| **Name** | Insufficient Budget |
| **Probability** | Low (10%) |
| **Impact** | Marginal (2) |
| **Description** | If the budget is low project may not complete. |
| **Mitigation and Monitoring** | The project needs streaming server that is costly to set-up. We find several alternative streaming services to reduce the budget  risk. |
| **Management** | Refinement in project goal. A new plan for regulate the budget. |
| **Status** | We have not encountered such issue yet |

Tabel XXIV: Business Risk (B02)

|  |  |
| --- | --- |
| **Business Risk (B02)** | **Date: 05-02-2020** |
| **Name** | End Users Accept System |
| **Probability** | Low (10%) |
| **Impact** | Critical (4) |
| **Description** | The system fails to gain user’s faith. |
| **Mitigation and Monitoring** | In order to prevent this from happening, the software will develop with the end user in mind. The user-interface will design in a  way to make use of the program convenient and pleasurable. |
| **Management** | Training the users to familiarize them with the new system. Releasing patches/bug fixes for greater user satisfaction. |
| **Status** | The risk has not been arisen yet. |

Tabel XXV: Business Risk (B03)

|  |  |
| --- | --- |
| **Business Risk (B04)** | **Date: 30-01-2020** |
| **Name** | Late delivery of the project |
| **Probability** | Very Low (05%) |
| **Impact** | Catastrophic (1) |
| **Description** | The project may take more time to complete what was estimated. |
| **Mitigation and Monitoring** | Steps have been taken to ensure a timely delivery by determining the scope of project. |
| **Management** | The only course of action available would be to request an extension to the deadline from customer. |
| **Status** | My project is completed in time. |

**Chapter: 06**

**Analysis Modeling**

## Software analysis pattern

Software analysis patterns or analysis patterns in software engineering are conceptual models, which capture an abstraction of a situation that can often be encountered in modelling. An analysis pattern can be represented as a group of related, generic objects (meta-classes) with stereotypical attributes (data definitions), behaviors (method signatures), and expected interactions defined in a domain-neutral manner. By analysis modeling developer can finalize the specifications of the system. It is mandatory -

* + - To describe what the customer require
    - To establish a basis for the creation of a software design
    - To define a set of requirements that can be validated once the software is built.

## Activity Diagram

Activity diagram is an important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity. The diagrams describe the state of activities by showing the sequence of activities performed.

## Activity Diagram for Notice Upload from Admin & Coordinator

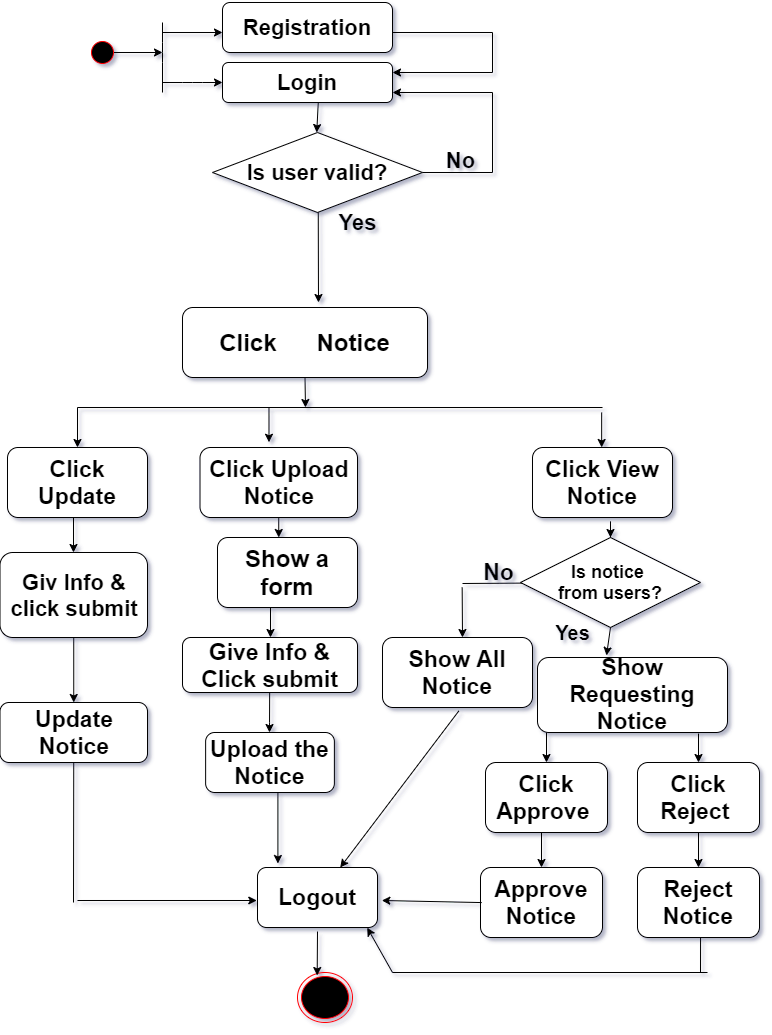
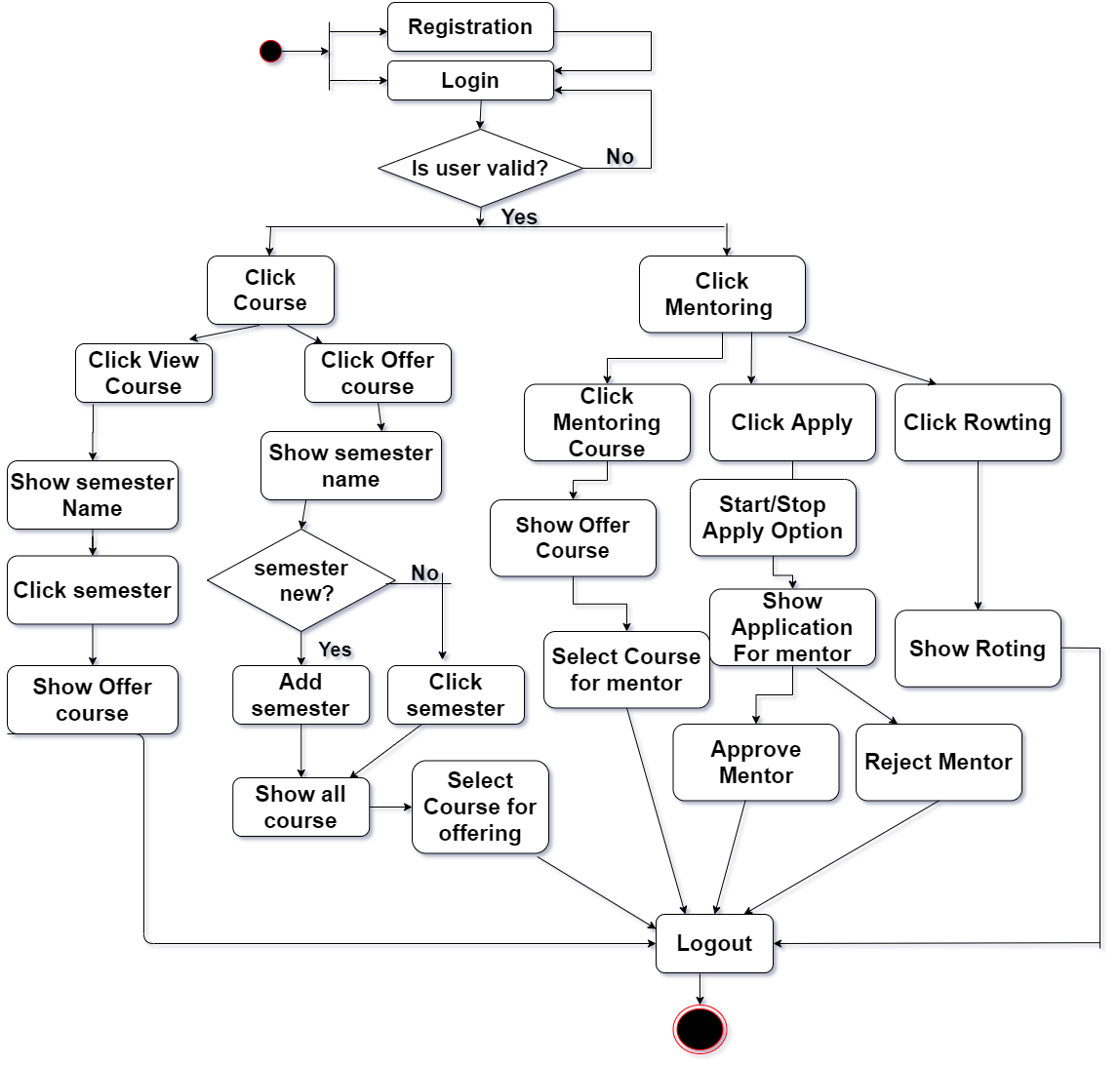
****

Figure 4: Activity Diagram for Notice Upload from Admin & Coordinator

Activity Diagram for Course & Mentoring from Admin & Coordinator **** Figure 5: Activity Diagram for Course & Mentoring from Admin & Coordinator

## Activity Diagram for member, event & fund from Admin

## C:\Users\Morium\Downloads\Actiity Diagram for event member & fund.png

## Figure 6: Activity Diagram for member, event & fund from Admin

Activity Diagram for member, event & fund from Coordinator

## C:\Users\Morium\Downloads\Actiity Diagram for event member & fundCo.png

## Figure 7 : Activity Diagram for member, event & fund from Coordinator

## Activity Diagram for user

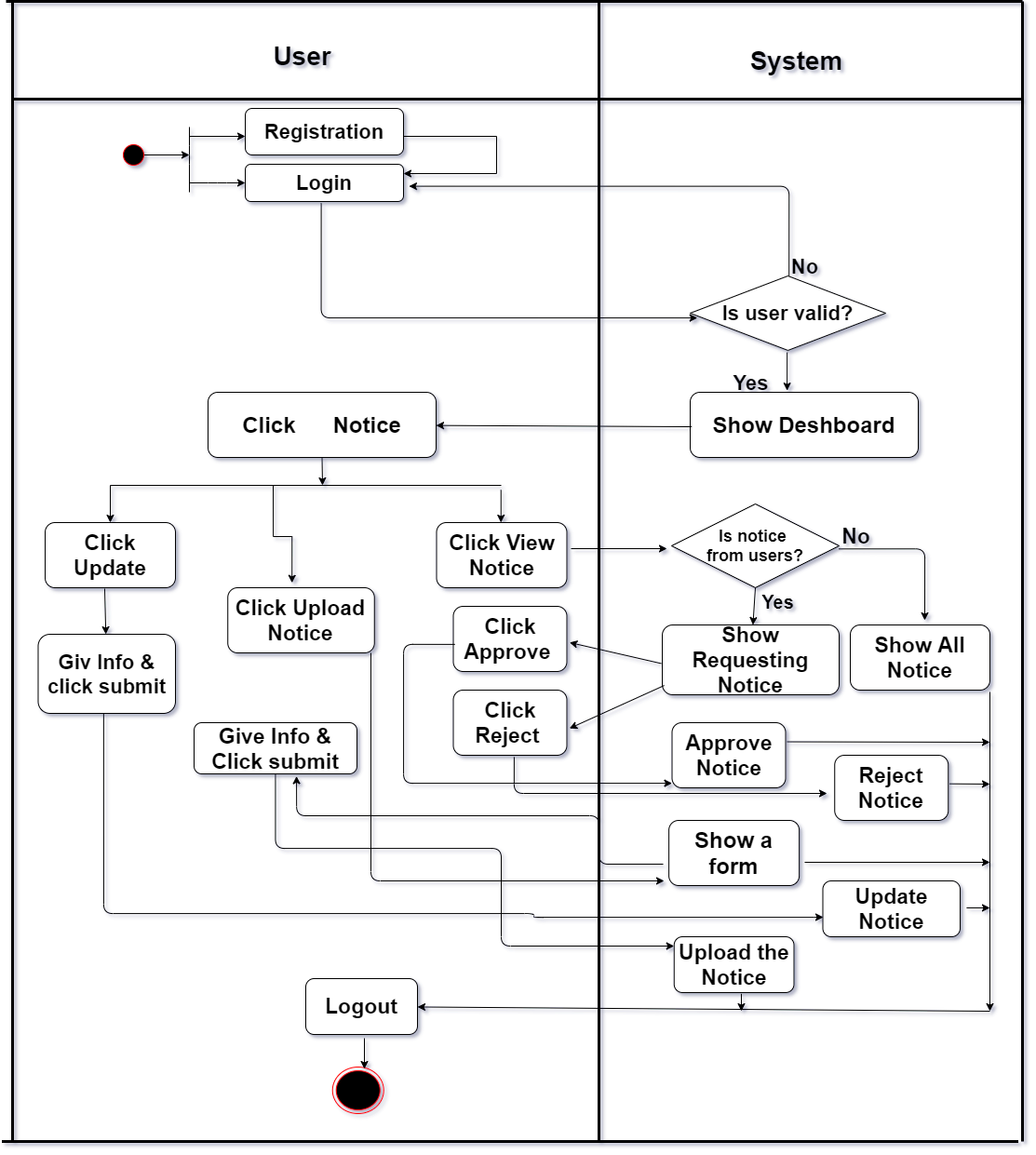
## C:\Users\Morium\Downloads\User activity last.png

## Figure 8: User activity diagram

## Swimlane Diagram

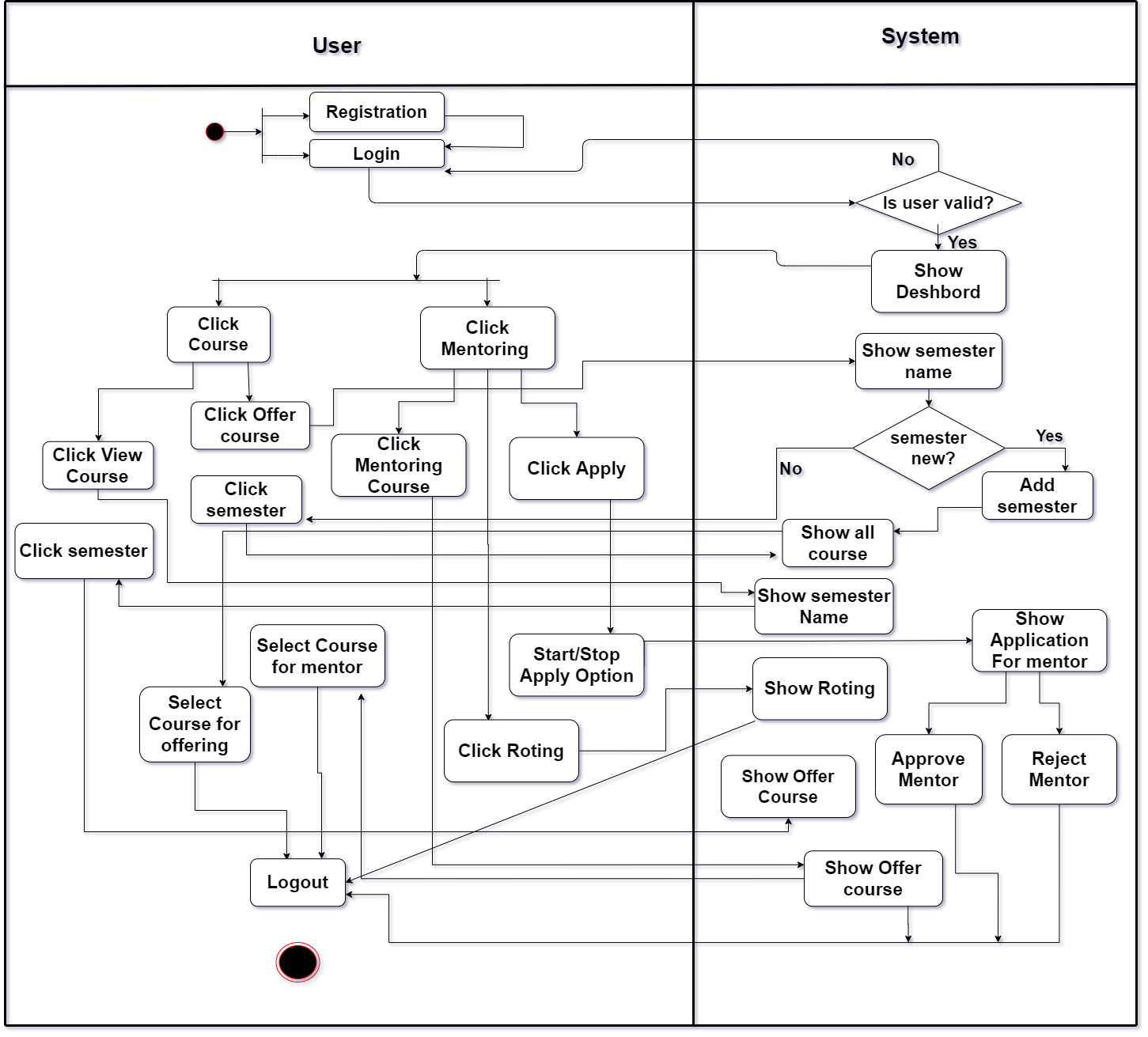
A Swimlane diagram is a type of flowchart that delineates who does what in a process. Using the metaphor of lanes in a pool, a Swimlane diagram provides clarity and accountability by placing process steps within the horizontal or vertical “Swimlanes” of a particular employee, work group or department. It shows connections, communication and handoffs between these lanes, and it can serve to highlight waste, redundancy and inefficiency in a process.

## Swim lane Diagram for Notice Upload from Admin & Coordinator

****

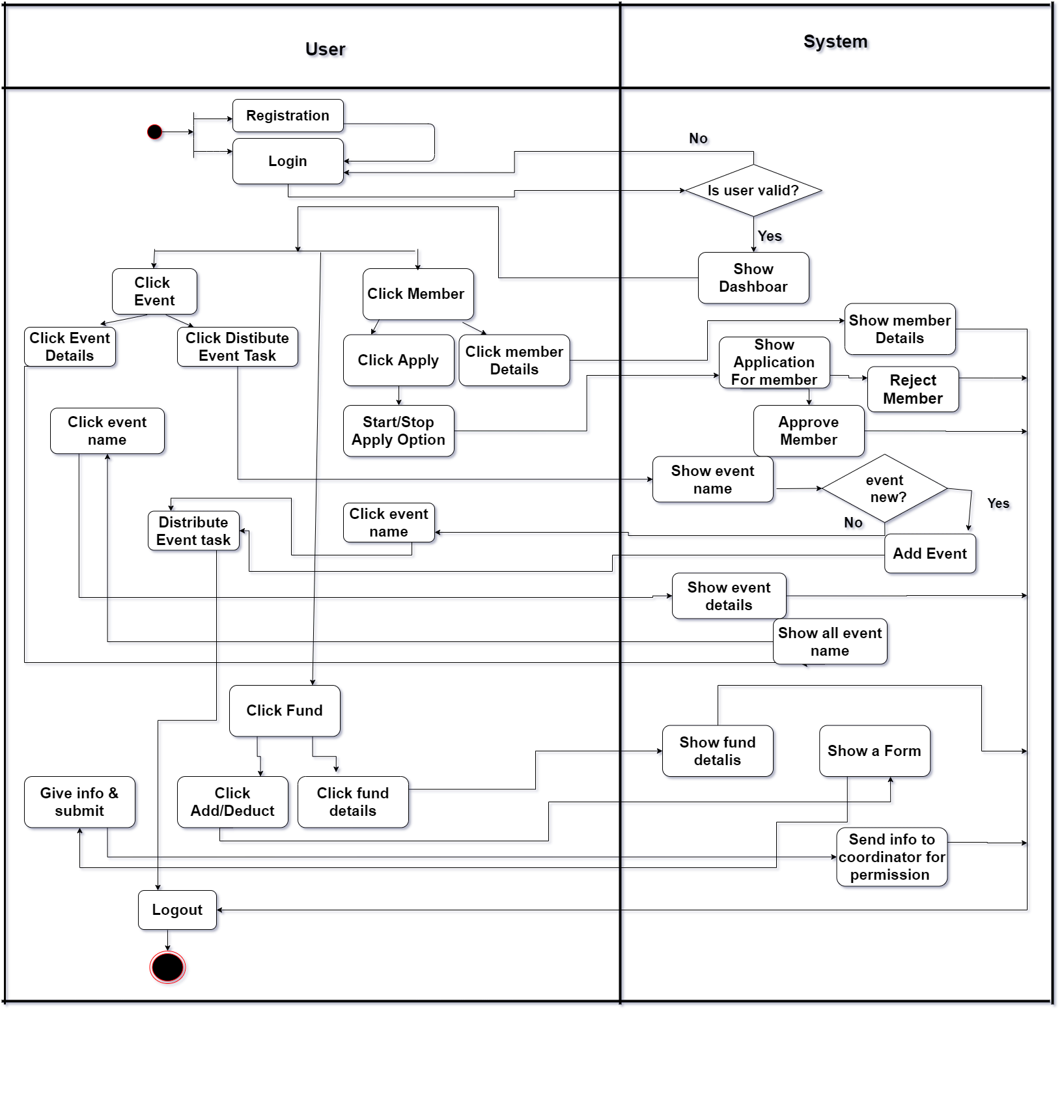
## Figure 9: Swim lane Diagram for Notice Upload from Admin & Coordinator

## Swim lane Diagram for Course & Mentoring from Admin & Coordinator

****

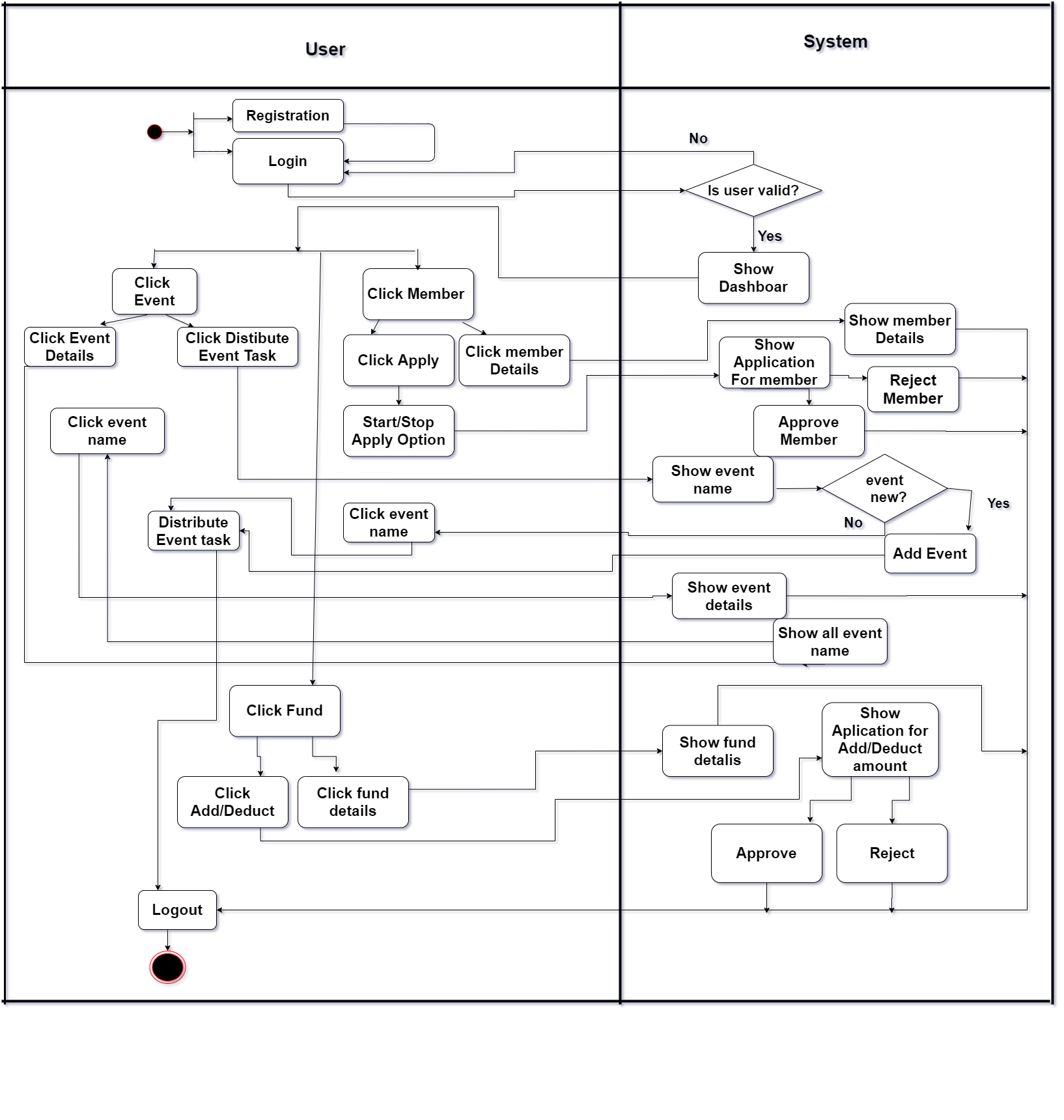
## Figure 10: Swim lane Diagram for Course & Mentoring from Admin & Coordinator

## Swimelane Diagram for member, event & fund from Admin

****

## Figure 11: Swimelane Diagram for member, event & fund from Admin

**Swimelane Diagram for member, event & fund from Coordinator**



## Figure 12: Swimelane Diagram for member, event & fund from Adm

Swimelane Diagram for user

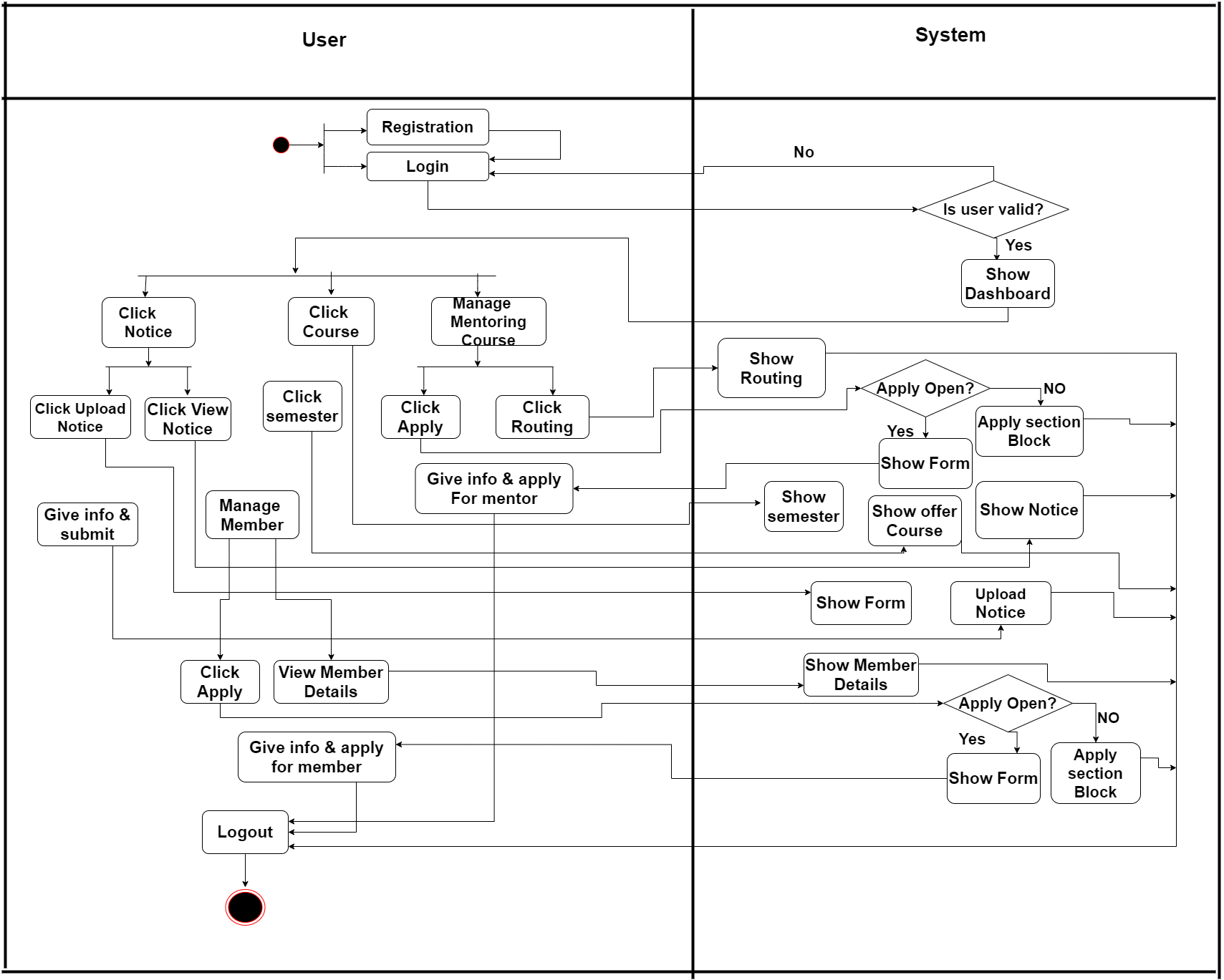


Figure 13: Swimelane Diagram for user

**Chapter: 07**

**Designing**

## 

## Figure 14 :First interface

## 

## Figure 15: Login interface

## 

## Figure 16: Admin Login

## 

## Figure 17; Upload Notice

## 

## Figure 18; View Notice

## 

## Figure 19: Requesting Notice for approve/reject

## 

## Figure 20: Offer course interface

## 

## Figure 21: View offer course interface

## 

## Figure 22: Offer mentoring courser

## 

## Figure 23:View member details

## 

## Figure 24e: Event Details interface

## 

## Figure 25: Fund deduct form

## Data Flow Diagram

A data flow diagram is a graphical representation that depicts information flow and the transforms that are applied as data move from input to output. It is known as data flow graph or bubble chart.

The DFD may be used to represent a system or software at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail. Therefore, the DFD provides a mechanism for functional modeling as well as information flow modeling.

A level 0 DFD, which is also known as fundamental system model or a Context model, represents the entire software or system element into as a single bubble with input and output data indicated by in Coming and outgoing arrows respectively. Then bubble of context model should be decomposed into several levels.

In DFD, there are four symbols:

* A square defines a source or destination that is external entity of system data.
* An arrow identifies data flow that data is motion. It is pipeline through which information flows.
* A circle or a bubble represents a process that transforms incoming data flow(s) into outgoing data flow(s).
* An open rectangle is a data store or a temporary repository of data.

## DFD of the Project

Context level diagram

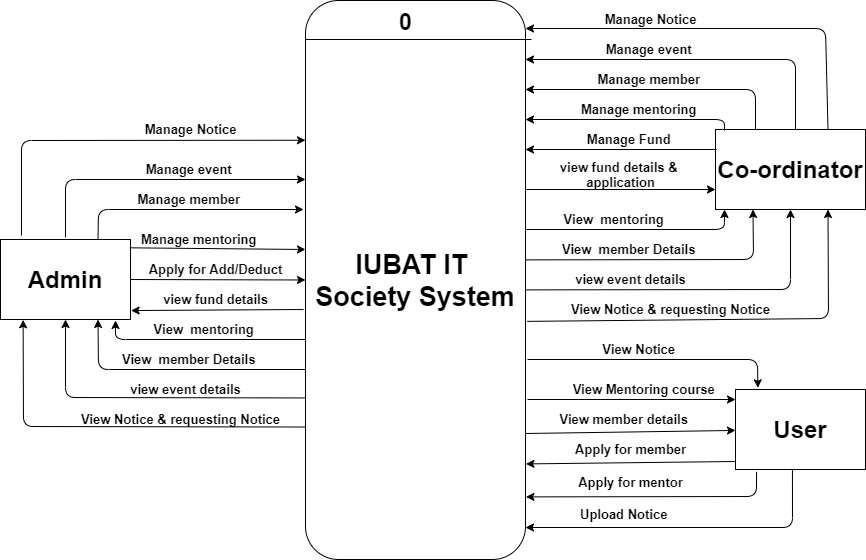


Figure 26: Context Level Diagram

Level 1 DFD

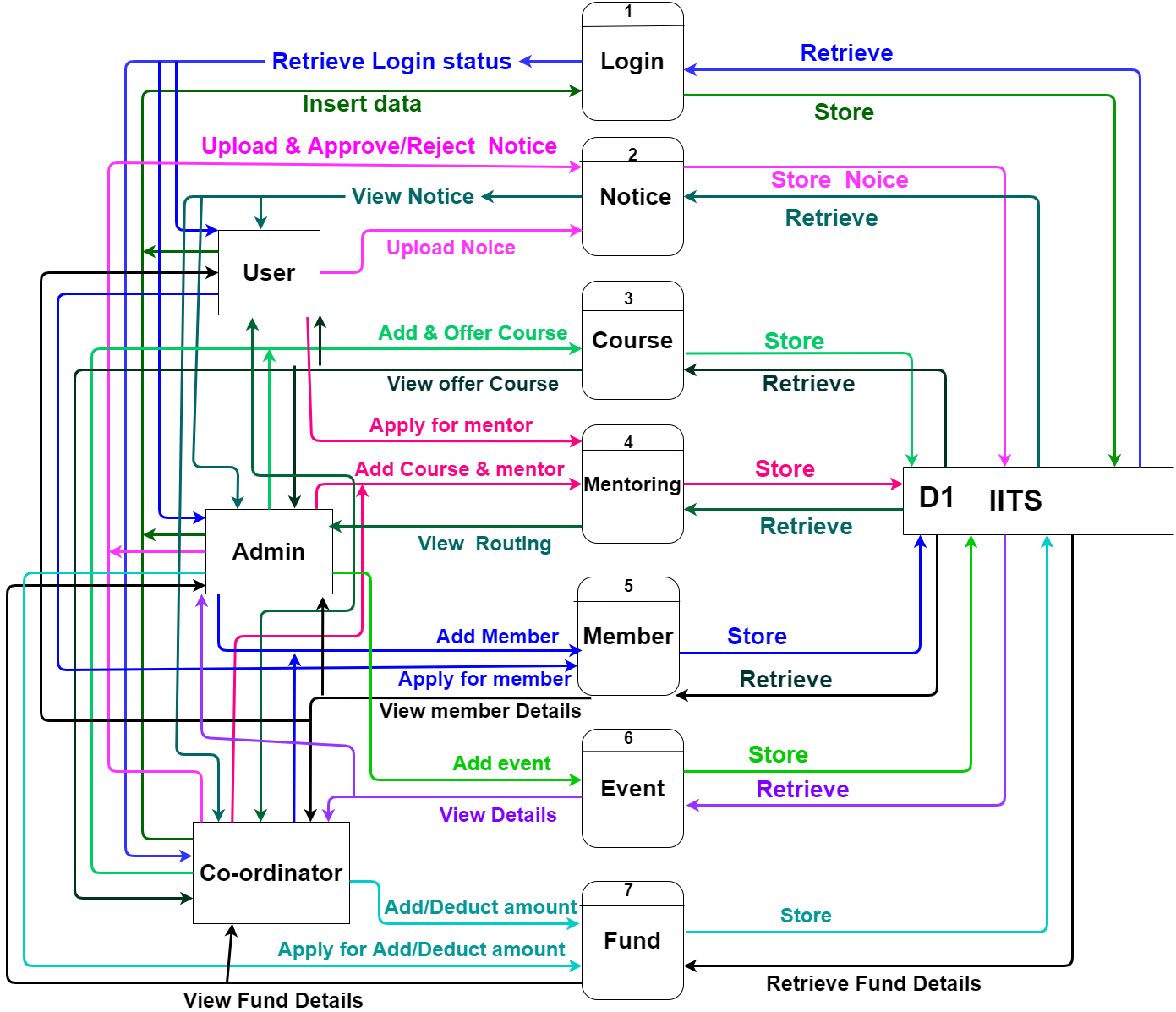


Figure 27: Level 1 DFD

Level 2 process 1

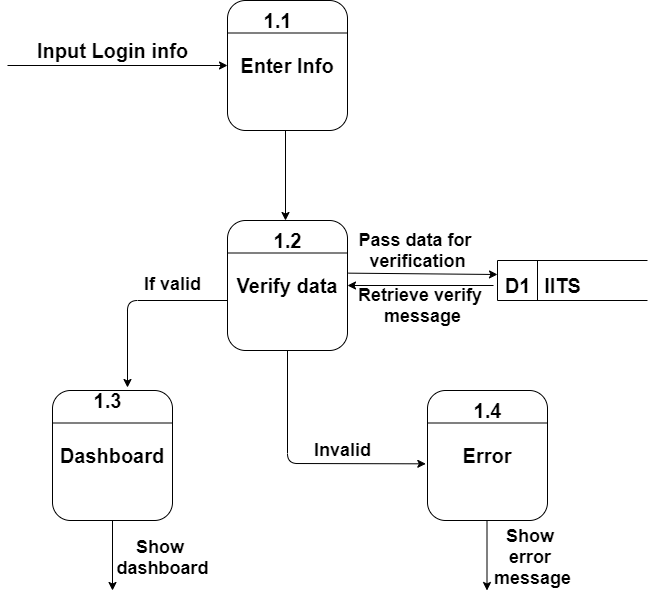


Figure 28: Level 2 process 1

Level 2 process 2

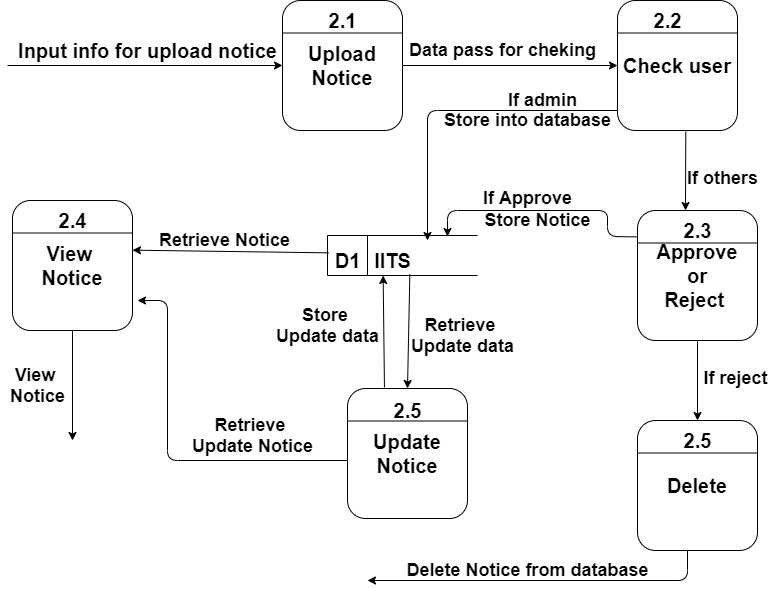


Figure 29: Level 2 process 2

Level 2 process 3

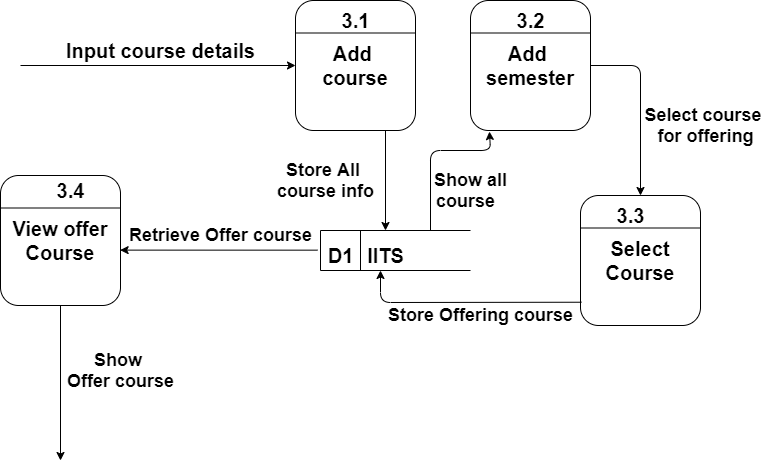


Figure 30: Level 2 process 3.

Level 2 process 4

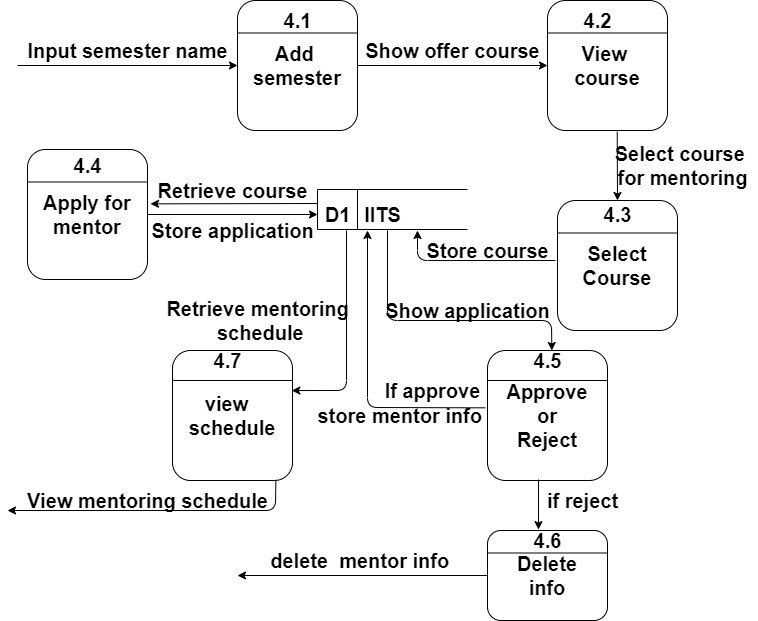


Figure 31: Level 2 process 4

Level 2 process 5

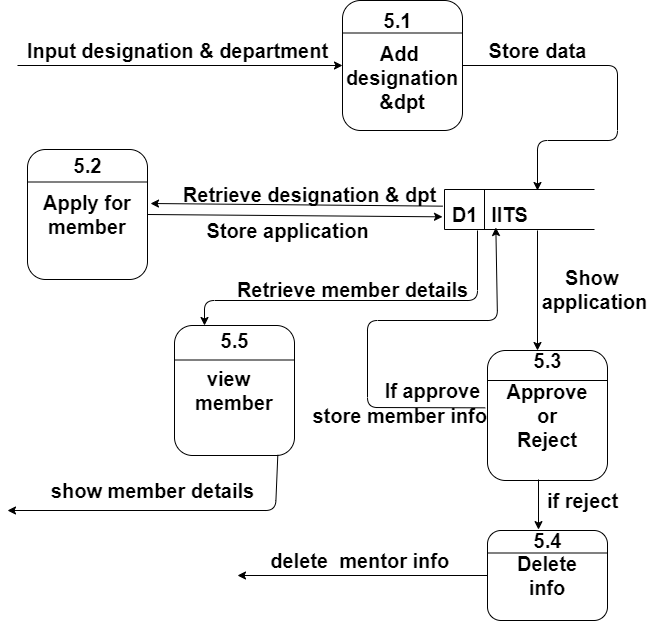


Figure 32 : Level 2 process 5

Level 2 process 6

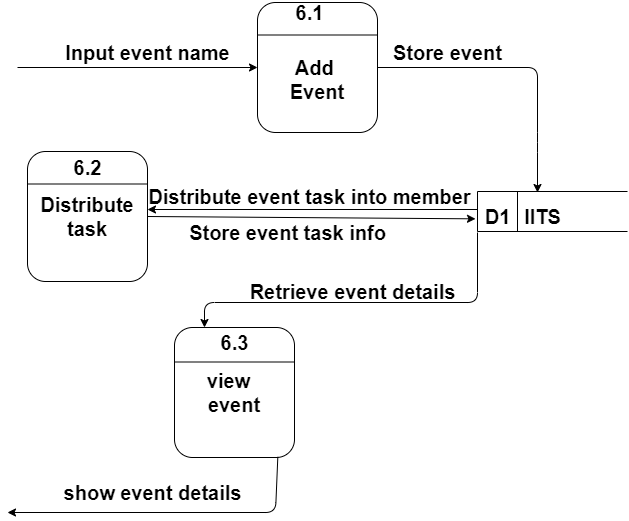


Figure 33: Level 2 process 6

Level 2 process 7

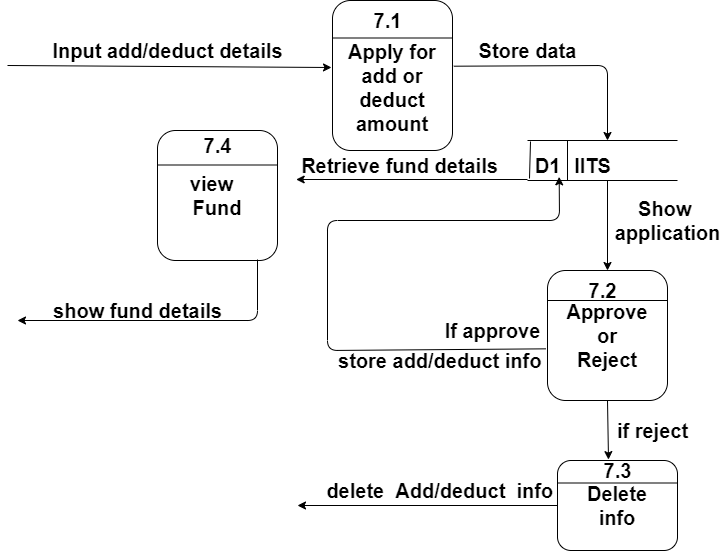


Figure 34: Level 2 process 7

## Entity Relationship Diagram

The entity relationship (ER) diagram of the project “Development of Point of Sales System (POS) for Xubisoft Ltd.” illustrates how entities of this system relate to each other within the system. It illustrates the logical structure of the database used for this project.

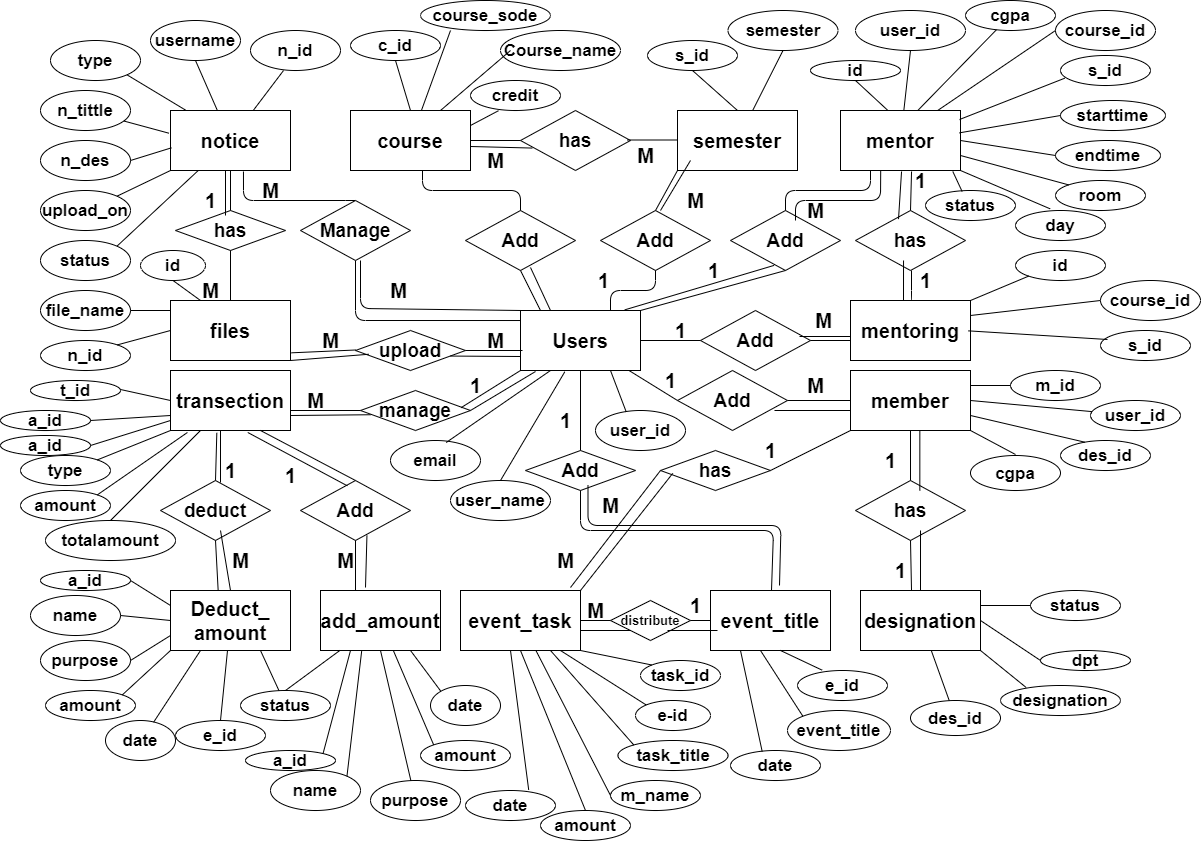
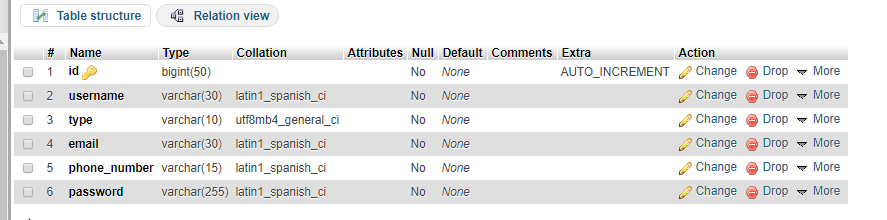


Figure 35: Entity Relationship Diagram

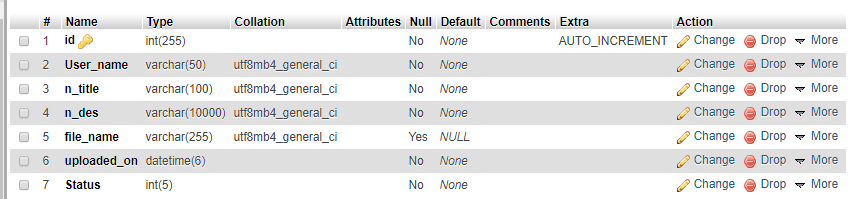
## Database Design

A database is an organized collection of data. Database design is the process of producing a detailed data model of database.

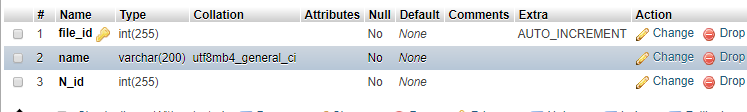
Database table for “Users”



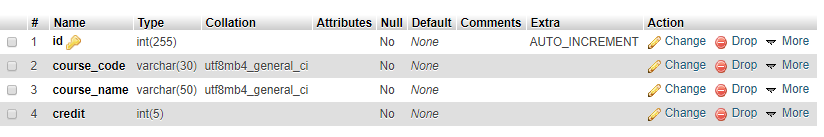
Database table for “notice”



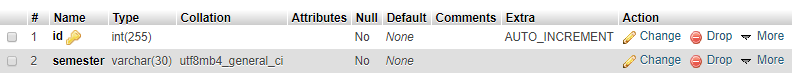
Database table for “Files”



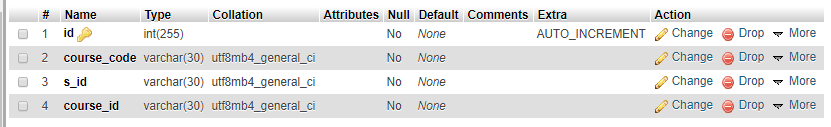
Database table for “course”



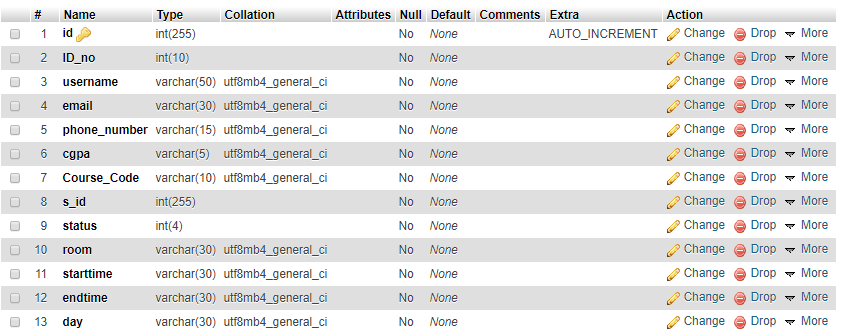
Database table for “semester”



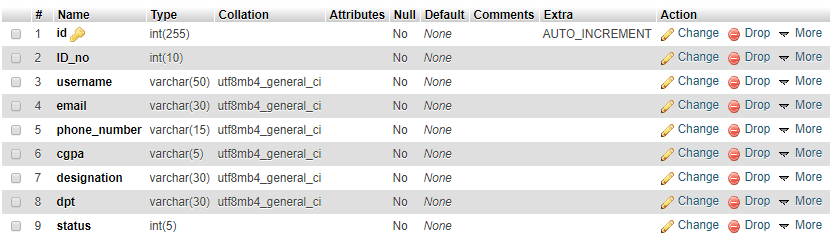
Database table for “mentoring”



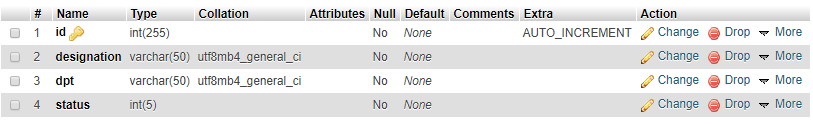
Database table for “mentor”



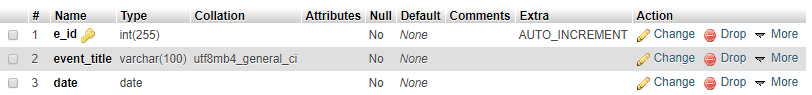
Database table for “member”



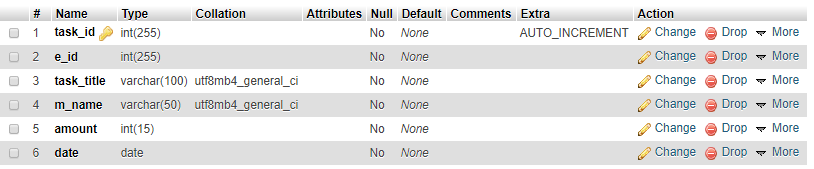
Database table for “designation”



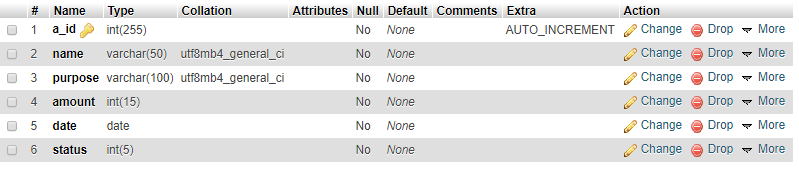
Database table for “event\_title”



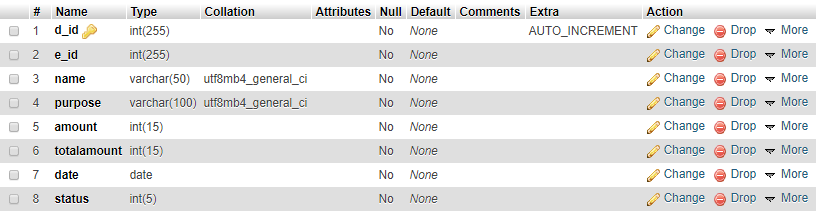
Database table for “event\_task”



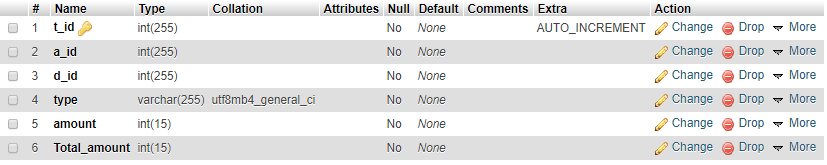
Database table for “add\_amount”



Database table for “deduct\_amount”



Database table for “transaction”



Chapter: 08

Quality Assurance

## System Quality Management

System quality management for the project “IUBAT IT Society System.” ensures that the required level of quality is achieved by submitting improvements to the product development process. The quality of this software is measured and the process is improved until the proper quality level is achieved. It is measured by a number of variables. The aim of this system quality management is to manage the quality of this “IUBAT IT Society System.” and of its development process.

## System Quality Management Processes

System quality management of the project “IUBAT IT Society System.” involves the application of specific quality processes and checking that these planned processes have been followed. The processes involved in the System quality management of this software. They are:

* + - Quality Planning
    - Quality Assurance
    - Quality Control

## Quality Planning

Quality planning is performed by first selecting the applicable procedures and standards for this project “IUBAT IT Society System.” and then modifying these as required. All the important factors that contribute the customer requirements are explained in this development.

## Quality Assurance

Quality assurance established organizational procedures and standards for the quality of the project “IUBAT IT Society System.”. It involved evaluating overall project performance to ensure that the project can fulfill the relevant quality standards.

## Quality Control

Quality control for the project “IUBAT IT Society System.” ensured that the software development team implemented procedures and standards. Project result is monitored to ensure that improve while identifying ways the overall quality according to quality standards.

## System Testing

Testing is performed in the project “IUBAT IT Society System.” with the aim of finding errors. These testing results are given in this testing chapter. These testing are

done by evaluating the software against requirements gathered from users and system specifications. There are various types of testing available to test software. In this chapter two types of testing are included. They are:

1. White Box Testing
2. Black Box Testing

## White Box Testing

White box testing is used in the project “IUBAT IT Society System.” to test this systems internal coding and infrastructure. It focuses primarily on strengthening security, the flow of inputs and outputs through the application, and improving design and usability of the system.

## Black Box Testing

In the project “IUBAT IT Society System.” the black box testing techniques are used to test the functionality of the system without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This testing is done based on the software requirements and specifications.

## Testing

Testing scenarios of the project “IUBAT IT Society System.” are given below:

Table XXVI: System Testing Scenario 1

|  |  |
| --- | --- |
| **Testing scenario No: 1** | |
| Scenario | Admin, Staff Login testing scenario of the system |
| Input’s | User name, password of admin or staff for login |
| Desired Output’s | When enter user name, password then get access level define. |
| Actual Output’s | For login the system work correctly |
| Verdict | Getting result from Desired Output’s and Actual Output’s decided this system is successful for login. |

Table XXVII: System Testing Scenario 2

|  |  |
| --- | --- |
| **Testing scenario No: 2** | |
| Scenario | Admin & User can upload notice and view notice testing scenario of the system. |
| Input’s | Information of notice will be stored into the database. |
| Desired Output’s | When enter all basic information correctly, those information will be stored and display details in the system. |
| Actual Output’s | All the notice stored and displayed in the system. |
| Verdict | Getting result from Desired Output’s and Actual Output’s decided this system  is successfully stored and displayed. |

Table XXVIII: System Testing Scenario 3

|  |  |
| --- | --- |
| **Testing scenario No: 3** | |
| Scenario | Admin can approve and reject notice testing scenario of the system. |
| Input’s | Information of notice by user. |
| Desired Output’s | When enter all basic information correctly, those information will be stored. Admin can only view those notice in the system. |
| Actual Output’s | All the notice stored and admin can only see those in the system. |
| Verdict | The process is worked correctly and successfully. |

Table XXIX: System Testing Scenario 4

|  |  |
| --- | --- |
| **Testing scenario No: 4** | |
| Scenario | Admin can add course and select course for offering testing scenario of the  system. |
| Input’s | Course details will be stored into the database. |
| Desired Output’s | When enter all basic information correctly, display offer course in the system. |
| Actual Output’s | Select course for offering and Offering course info displays the system work correctly. |
| Verdict | Getting result from Desired Output’s and Actual Output’s decided this system  is successfully added. |

Table XXX: System Testing Scenario 5

|  |  |
| --- | --- |
| **Testing scenario No: 5** | |
| Scenario | Admin can Add mentoring course and mentor testing scenario of the system. |
| Input’s | Course and mentor details by the admin. |
| Desired Output’s | When enter all basic information correctly, those information will be stored.  Mentoring schedule will be generate in the system. |
| Actual Output’s | Mentor details stored and schedule generated the system works correctly. |
| Verdict | The process is worked correctly and successfully. |

Table XXXI: System Testing Scenario 6

|  |  |
| --- | --- |
| **Testing scenario No: 6** | |
| Scenario | User can apply for mentor and member testing scenario of the system. |
| Input’s | Mentor and member information will be stored into the database. |
| Desired Output’s | When enter all basic information correctly, display mentor and member details in the system.. |
| Actual Output’s | For add mentor and member information displays the system work correctly. |
| Verdict | Getting result from Desired Output’s and Actual Output’s decided this system  is successfully added. |

Table XXXII: System Testing Scenario 7

|  |  |
| --- | --- |
| **Testing scenario No: 7** | |
| Scenario | Admin can update and delete the mentor and member information testing scenario of the system. |
| Input’s | Mentor and member basic information updated and deleted by the admin. |
| Desired Output’s | When changed basic information by admin, display updated mentor and member details in the system. |
| Actual Output’s | For update and delete mentor and member information displays the system works correctly. |
| Verdict | The process is worked correctly and successfully.. |

Table XXXIII: System Testing Scenario 8

|  |  |
| --- | --- |
| **Testing scenario No: 8** | |
| Scenario | Admin can update and delete the Notice testing scenario of the system. |
| Input’s | Notice basic information updated and deleted by the admin. |
| Desired Output’s | When changed basic information by admin, display updated notice in the system. |
| Actual Output’s | For update and delete notice displays the system works correctly. |
| Verdict | The process is worked correctly and successfully.. |

Table XXXIII: System Testing Scenario 9

|  |  |
| --- | --- |
| **Testing scenario No: 9** | |
| Scenario | Admin can add event and distribute task testing scenario of the system. |
| Input’s | Event information stored into database. |
| Desired Output’s | When enter all basic information correctly, those information will be stored and display event details in the system. |
| Actual Output’s | All the event details stored and displayed in the system. |
| Verdict | The process is worked correctly and successfully.. |

Table XXXIII: System Testing Scenario 10

|  |  |
| --- | --- |
| **Testing scenario No:10** | |
| Scenario | Admin can send add and deduct amount form to coordinator testing scenario of the system. |
| Input’s | Add and deduct amount information. |
| Desired Output’s | Add and deduct amount information will send to coordinator. |
| Actual Output’s | For Add and deduct amount information displays into coordinator panel the system work correctly. |
| Verdict | The process is worked correctly and successfully.. |

Table XXXI: System Testing Scenario 11

|  |  |
| --- | --- |
| **Testing scenario No:11** | |
| Scenario | Coordinator can approve and reject add/deduct amount application testing scenario of the system. |
| Input’s | add/deduct amount information will be stored into the database. |
| Desired Output’s | When coordinator approve/reject those application, display Fund details in the system.. |
| Actual Output’s | Coordinator approve/reject application add/deduct information displays the system work correctly. |
| Verdict | Getting result from Desired Output’s and Actual Output’s decided this system  is successfully added. |

**Chapter: 09**

**Conclusion**

## Practicum & Its Value

## Practicum is the transition from engineering college study life to a real world workplace through hands on

## experience of engineering practice. It helps to development of deeper skills in the selected research area

## and also increases employability.

## Meeting with different types of people and encountering situations gives practical

## orientation to life. There are many more upright issues of practicum, which only the person

## experiencing it can sense and believe.

## Student of College of Engineering and Technology (CEAT) at IUBAT go for this practicum

## program carrying 6 credit hours weight, which goes for a semester long and usually after the

## completion of the course work.

## Experience:

## Working in this project was a big opportunity for me. I have learned a lot about the technical and

## learning Environment of Xubisoft Ltd where I have done my internship.

## The following indicator will indicate some of my technical issue which I have learnt and

## implemented in this project.

## How to collect requirements for clients.

## How to do the analysis of a project.

## How to insert multiple file at a time & show them.

## How to create application form.

## How to select course.

## How to distribute task into member.

## Implemented Add & Deduct calculation into project.

## Conclusion

The project “IUBAT IT Society System .” is the system that can get various information in one place. With time, new features are added. Overall, the system performs well, and while it does not include all of the features that may have been desired, it lives up to initial expectations. Overall, this system will effectively save the time of the admin and the users and it will give many benefits those who are going to use this system.

## Limitations

* It is not check user type.
* The system can not send any email confirmation.
* User can not download any file from this project.

## Future Plan

* The system will send message through email to user.
* I will make download option for downloading file.
* I will make notifications of live comments.
* I will make new services and newer versions of this software

\

# References

## Books

1. Kendall, E. & Kendall (1999), System Analysis and Design. 4th Ed. New Delhi: Prentice Hall.

1. Silberschattz, Abraham, Korth, Henry F., &Sudrashan S. (2002). Database System Concepts.

4th ed. Boston: McGraw Hill.

1. Pressman, Roger S. (2004). Software Engineering: A Practitioner’s Approach. 5th ed. Boston: McGraw Hill.

## Websites

1. Longstreet, David. (2005). Fundamentals of Function Point Analysis. [Online] Available at:<http://www.softwaremetrics.com/fpafund.htm>
2. How to calculate function points. (2019, Jan 10). Available at: <http://stackoverflow.com/questions/34473698/how-to-calculate-function-points>
3. Tutorials Point. (2015). Database Management System. (2019, Jan 20). Available at: [https://www.tutorialspoint.com/dbms/dbms\_tutorial.pdf](http://www.tutorialspoint.com/dbms/dbms_tutorial.pdf)
4. Mamčenko. J, Introduction to Dta Modeling and MSAccess. (2019, Feb 07). Available

at:<http://gama.vtu.lt/biblioteka/Information_Resources/i_part_of_information_re> sources.pdf

1. Risk Management (2018).

Retrieved from Risk Management: https:/[/www.c](http://www.castsoftware.com/research)a[stsoftware.com/research](http://www.castsoftware.com/research) labs/riskmanagement-in-software-development-and-software-engineering-projects

1. Guru99 (2019)*. What is WHITE Box Testing? Techniques, Example &Types.*Available at: [https://www.guru99.com/white](http://www.guru99.com/white-box-testing.html)-[box-testing.html.](http://www.guru99.com/white-box-testing.html) (Accessed 11 April, 2019).