TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING

Kathmandu Engineering College
Department of Electronics and Computer Engineering

Proposal for Minor Project

On

ONLINE NOTICE BOARD SYSTEM

By
Anisha Agrawal - 67004
Anurag Acharya - 67008
Monika Sharma - 67040

Kathmandu, Nepal

2069

Abstract

This document is a proposal for the "Online Notice Board System" as the Minor Project for the partial fulfillment of the course of Bachelors in Computer Engineering, Third Year Second Part (III/II).

The system will primarily be web-based so as to facilitate easy access to all clients regardless of their location or the type of machine. Apart from this, it will also have supplementary application programs (commonly referred to as "Apps") for today's leading operating platforms like Android™ and Windows 8™. These applications be very user-friendly and will have features like auto-update and push messaging for user convenience.

This project will teach us the practical implementation of object-oriented programming. Moreover, this will also provide us first-hand experience about several important fields of a Computer Engineer, like Web Development and Application Development. It will also help us learn programming languages like C# and Java, web-development scripts like PHP, XAML and CSS and DBMS like MySQL.

Apart from these, we strongly believe going through this project will make us more competent for similar future tasks and enhance our technical and other abilities.

Acknowledgement

First and foremost, we would like to thank **Mrs. Pranita Upadhyaya** for instructing us about the methods of conducting feasibility and requirement studies as well as system designing. We would also like to thank **Mr. Krishna Kandel**, the Project Supervisor and **Mr. Sudeep Shakya**, the Third Year Coordinator for their continuous guidance and support.

Furthermore, we would like to express our deepest gratitude to **Mr. Manish Aryal**, Head of Department of Computer Engineering, for his never-ending support, moral or otherwise.

Finally, we would like to collectively thank any and all people that have helped us in any manner with respect to this proposal.

Table of Contents

Abstract i
Acknowledgement ii
List of Figures v
List of Symbols vi
List of Acronyms / Abbreviations vii
1. Project Introduction 1
2. Problem Statement 1
3. Need 1
4. Aims and Objectives 2
4.1 Primary Aim 2
4.2 Objectives 2
5 Preliminary Requirements Estimate 2
6 Suggested Deliverables 2
7 Process Model 3
8 Visibility Plan 3
9 Feasibility Analysis 3
9.1 Technical Feasibility 3
9.2 Schedule Feasibility 3
9.3 Economic Feasibility 3
9.4 Operational Feasibility 4
9.5 Legal Feasibility 4
9.6 Religious-Cultural Feasibility 4
9.7 Socio-Political Feasibility 4
10 Risk Analysis 5
10.1 Performance Risks 5
10.2 Safety Risks 5
10.3 Security Risks 5
11 Background Theory 5
12 Literature Review 6
13 Application Area 6
14 System Features 6
14.1 File Sharing 6
14.2 Push Messaging 7
14.3 Database and UI Auto-Update 7
15 Requirement Analysis 8
15.1 Major Functional Requirements 8

15.2 External Interface Requirements 8
15.2.1 User Interfaces 8
15.2.2 Hardware Interfaces 8
15.2.3 Software Interfaces 8
15.2.4 Communications Interfaces 8
15.3 Other Nonfunctional Requirements 9
15.3.1 Performance Requirements 9
15.3.2 Safety Requirements 9
15.3.3 Security Requirements 9
15.3.4 Software Quality Attributes 9
16 System Modeling 10
16.1 Block Diagram Representation 10
16.2 Usage Modeling 11
16.2.1 User Profile 11
16.2.2 Use Case Diagram 11
16.2.3 Use Case Analysis 12
16.3 Data Modeling 13
16.3.1 Data Dictionary 13
16.3.2 Data Description 16
16.3.3 Entity Relation (ER) Diagram 19
16.4 Context Diagram 20
16.5 Object Diagrams 21
16.6 Class Diagram 23
17 Cost Estimation 24
18 Gantt Chart 24
19 References 25

List of Figures

Figure 16-1 Block Diagram of the System 10
Figure 16-2 Use Case Diagram 11
Figure 16-3 ER Diagram 19
Figure 16-4 Context Diagram (Level 0 DFD) 20
Figure 16-5 Object Diagram I 21
Figure 16-6 Object Diagram II 22
Figure 16-7 Class Diagram 23
Figure 18-1 Gantt Chart 24

List of Symbols

S. No.	Symbol	Meaning	Remarks
1.	™	Trade Mark	
2.	®	Registered	
3.	C#	C-Sharp	

List of Acronyms / Abbreviations

S. No.	Acronym	Full Form
1.	PHP	Hypertext Preprocessor
2.	CSS	Cascading Style Sheet
3.	XAML	Extensive Application Markup Language
4.	OS	Operating System
5.	SQL	Structured Query Language
6.	SDLC	Software/System Development Life Cycle
7.	LSM	Linear Sequential Model
8.	IDE	Integrated/Interactive Development Environment
9.	MIC	Microsoft Innovation Centre
10.	PC	Personal Computer
11.	UI	User Interface
12.	IP	Internet Protocol
13.	DFD	Data Flow Diagram
14.	ER	Entity Relation(ship)
15.	app	Application Software

1. Project Introduction

The Online Notice Board System is intended for colleges and institutions where information and file sharing on regular basis plays vital role in the performance. The proposed system will act as an online notice board which will make use of the modern communication methodologies and techniques for information flow. The system is planned to consist of various useful features for the said purpose.

The proposed system aims to create a platform for issuing notice, sharing information and files between the members of the institution. Different users shall have different level of access to the content. In the context of a college, there shall be four users of the software – administrator, student, teachers and guest. The administrator shall be able to issue notice, upload files, and view students' activities on the software. The teacher shall be able to issue notice, upload files (on a general basis or to a particular classroom), and view result and students' activities. The student shall be able to upload/download files, submit to faculty/teacher and view result. A guest will simply be able to view public notices.

2. Problem Statement

In today's world, everything is digitalized and paper is being used less and less every day. How often has it happened that we miss some important notice because we have to go to a wall and read the notice there? There are dedicated file hosting sites and clouds used by some institutions, but there is a definite need for a dedicated noticeboard system. The proposed system is such a system.

3. Need

Almost all leading institutions, excepting a few, currently lack an electronic noticeboard system. Though some have taken the aid of third-party websites like Facebook® to interact, it comes at the cost of mixing one's social life with professional. Keeping this in mind, educational institutes will find this software extremely useful.

The "Online Notice Board System" is a web- based software, with supplementary application software, that aims to aid the institutes by providing such digital noticeboard.

4. Aims and Objectives

4.1 Primary Aim

The primary aim of the Online Noticeboard Software project is to create a fully functional digital noticeboard system which will efficiently handle all assigned tasks.

It may be able to, in due course, heavily minimize, if not eradicate, the conventional, physical noticeboards. With the help of the supplementary applications, users will be able to receive real-time notifications of any and all notices posted by another user with privileges as such.

4.2 Objectives

- To develop supplementary apps for the said noticeboard.
- To create a user-friendly interface.
- To develop and manage a proper database system to ensure data safety and proper management.
- To allocate various levels of users and have proper authentication.
- To prepare proper and detailed system documentation.

5 Preliminary Requirements Estimate

- Programming Languages and scripts
    - Web Application: PHP and CSS
    - Windows 8 App: C# and XAML
    - Android App: Java
- Operating System
    - Web Application: Any current OS
    - Windows 8 App: Windows® 8, Windows® 8 Phone
    - Android App: Android™ OS
- Database – MySQL

6 Suggested Deliverables

- A web-based online noticeboard system
- A Windows 8™ App
- An Android™ App
- Proper system documentation

7 Process Model

The Incremental model has been deemed the best suited for the SDLC process and shall be followed. Using this model will have the benefit over the LSM in the sense that it will inherit the quality of the LSM but exclude the rigidity. Furthermore, the feedback process of Process Iteration model will also be available in the incremental model. Another contributing factor in choosing this model is the short period of time available to make a working initial version of the system.

8 Visibility Plan

- The group members will be in contact timely and will focus on one area at a time as all are beginners.
- The phases will be followed sequentially with equal contribution of group members in each phase.
- The group members will be in contact through regular meetings and social platforms.

9 Feasibility Analysis

9.1 Technical Feasibility

The software is to be developed using C#, Java, PHP, CSS, XAML and MySQL, which are all readily available. Also, the team members have sufficient programming and related knowledge which will enable us to learn and adapt to these specific languages and platforms easily. Thus we can see that the project is technically feasible.

9.2 Schedule Feasibility

The development process is planned to reach designing phase till the end of the semester, which gives us a windows of roughly six months. Although this may be a tight fit for a perfect, final system, as the Incremental Model is being followed in the SDLC, this is enough time to develop a working first version of the software.

9.3 Economic Feasibility

The program uses programming languages whose IDE are freeware. One exception to this is the Visual Studio 2012, which is needed for the supplementary app development. But this software is provided free of cost by

the MIC Nepal. Further costs for this project are the costs of online domain, space and database and registering and uploading the apps in the respective market, which is expected to be covered by the college. The remaining cost is that of training the developer team in the particular language and/or platform, which is minimal. So the project is economically feasible.

9.4 Operational Feasibility

The software requires very little specific environment to run. Only the apps require their environment to run, i.e. Windows 8™ and Android™. As a staggering majority of the PCs in the world are based on Windows™ OS and many mobile devices operate on Android™, this cannot really be considered a need. The software will be extremely user-friendly, removing the need for specifically trained employees. Similarly, the cost of buying the rights and the maintenance cost will not be very high for the client. So the software is feasible for operation.

9.5 Legal Feasibility

The developers will obviously use no illegal means or methodologies in the development process of the system. The software will be built and operated abiding by the Cyber and other applicable laws prevailing in the country enforced by the Government of Nepal. The user will be held responsible for only the data they enter to the system. In case of international users, they will be subjected to the applicable laws in that country. So the software has no legal barrages.

9.6 Religious-Cultural Feasibility

This system will never ask the user of their religion or cultural origin and ergo will not act in any way whatsoever that may hurt the sentiments of any cultural or religious group. The product development or operation will never undergo any process that might be unacceptable to a specific religion or culture. The software itself will be generic and impartial. So no religious or cultural issues should disrupt the system.

9.7 Socio-Political Feasibility

This software, being a simple notice board system, will by no means cause any alarms or questions in the society nor will it challenge any existing social conventions. Further, the software will not contradict or interfere in any way with the political happenings. The software will not be used as a means of campaign or promotion or a specific political or social organization. So the software is socially and politically feasible.

10 Risk Analysis

10.1 Performance Risks

The software performance may be hampered due to various factors like unresponsive UI and failure to execute given command(s). One of the factors for these can be a large-sized front end, which can be corrected by making a lightweight UI. The other reason for this can be poor internet connection or client-side faults, and are outside the control of the developers.

The use of a middleware also hampers the performance so it is advisable to design the system such that front and back end can communicate directly.

10.2 Safety Risks

Another risk with the system is the safety of the user data. The user data may be lost or corrupted if not for proper management and storage. This can be negotiated by using reliable databases in the back end.

10.3 Security Risks

A major risk in the system is the security of user data as well as their personal information. Firstly, the user authentication will have to be very stringent. The communication between UI and database should be secure. The database itself should be secure and measures like hashing and password protection should be used. The user's private information should never be accessible without the express permission of the user in question.

11 Background Theory

In today's world, everything is digitalized and paper is being used less and less every day. How often has it happened that we miss some important notice because we have to go to a wall and read the notice there?

The Online Notice Board System is mainly intended for colleges and institutions where information and file sharing on regular basis plays vital role in the performance. The proposed system will act as an online notice board which will make use of the modern communication methodologies and techniques for information flow. The system is planned to consist of various useful features for the said purpose.

The "Online Notice Board System" is a web- based software, with supplementary application software, that aims to aid the institutes by providing such digital noticeboard.

12 Literature Review

Almost all leading institutions, excepting a few, currently lack an electronic noticeboard system. Though some have taken the aid of third-party websites like Facebook® to interact, it comes at the cost of mixing one's social life with professional. Keeping this in mind, educational institutes will find this software extremely useful. There are dedicated file hosting sites and clouds used by some institutions, but there is a definite need for a dedicated noticeboard system. The proposed system is such a system.

13 Application Area

The proposed system is currently mainly intended to be applied in educational institutes as an online noticeboard and file sharing system. Further applications of this system may include any type of office or business organizations where computers with working internet connection is available (which is pretty much every organization.)

14 System Features

14.1 File Sharing

Description:

This will allow primary users to share files. These files may range from simple text or image notices to files of other formats. The pre-loaded notices and files could be accessed without an internet connection.

Stimulus/Response Sequences:

Files can be downloaded by clicking a single download button which triggers the download event. Any file to be uploaded can be browsed and then uploaded using another button event.

Functional Requirements:

REQ-1: Access to the internet
REQ-2: Required user authentication

14.2 Push Messaging

Description and Priority:

The system shall notify the primary users with recent notices and events with an automated message in their mobile phones or computers.

Stimulus/Response Sequences:

A change in the value in the database, caused by a user uploading a notice from his client side, will trigger an event which in turns triggers the push messaging event across all users who are set to receive push message for that particular notice and/or uploading user in question.

Functional Requirements:

REQ-1: Access to the Internet
REQ-2: Push Messaging Supporting Device
REQ-3: Proper user authentication

14.3 Database and UI Auto-Update

Description and Priority:

The system shall auto update its database. Recently posted notices and events as well as prioritized files shall be focused while outdated notices and files shall be cleaned up to make memory space available for new posts.

Stimulus/Response Sequences:

Any notice uploaded by any of the users triggers an event to append it to the database and update the UI accordingly across all the clients with users of the same class.

Functional Requirements:

REQ-1: Access to the internet
REQ-2: Proper User Authentication
REQ-3: An easy and flexible database management system

15 Requirement Analysis

15.1 Major Functional Requirements

The system should meet the following functional requirements:

- The system should be able to manage and store notices and files.
- The system should provide appropriate UI for notice and file upload and download.
- The system should be able to recognize and authenticate several levels of users.
- The system should be able to manage and maintain a proper database.
- The system should be easily operable and user friendly.
- The apps should be able to work seamlessly with the web server.
- The apps should be able to provide push messaging in real time, whenever available.

15.2 External Interface Requirements

15.2.1 User Interfaces

The system will start with a login page to identify the user level and the features will be available accordingly. The application will have easily accessible buttons for all major activities of the software. Key features and recent events will be highlighted to attract user attention.

15.2.2 Hardware Interfaces

The system will interact with the hardware resources of the system on which it is running. While any system will support the core software, a device with push messaging facility is recommended for Windows 8™ or Android™ app.

15.2.3 Software Interfaces

Only Windows 8™ and Android™ OS will support this software's apps. It will have the most interaction with an external database server. It may also be integrated with other systems like Employee Management System and/or Student Management System.

15.2.4 Communications Interfaces

An active internet connection will be required for the functioning of the software. In case of the mobile apps, an internet connection will then allow the software to connect to the online database. The system will use the Hyper Text Transfer Protocol (HTTP) to transmit data.

15.3 Other Nonfunctional Requirements

15.3.1 Performance Requirements

The system in itself does not require anything specific for basic operation, but the complete software with all its components running may have some performance requirements. Except viewing pre-loaded notices and files (in mobile apps), an internet connection is needed for the features of the system to become available.

15.3.2 Safety Requirements

Major attention should be given to the safety and security of the data and information that are stored in the software. The database must be trustworthy and non-leakage to ensure no data loss occurs.

15.3.3 Security Requirements

User authentication must be absolute and non-by-passable. No user should be able to access the software without providing proper authentication. In case of guest users, only public notices and events should be visible. Also, the IP address of the client machine can be recorded for future follow up of any security issues that may arise.

15.3.4 Software Quality Attributes

Several additional qualities and characteristics of the system will be important to the client and/or the developers, like correctness, maintainability, portability, testability and usability. For correctness, proper care and attention should be given during the design and coding from both developers and customer (should correct some false and unwanted features) side. Usability is achieved by developing the product as user friendly as possible. Similarly, maintainability and testability plays vital role in the long life of the software.

16 System Modeling

16.1 Block Diagram Representation

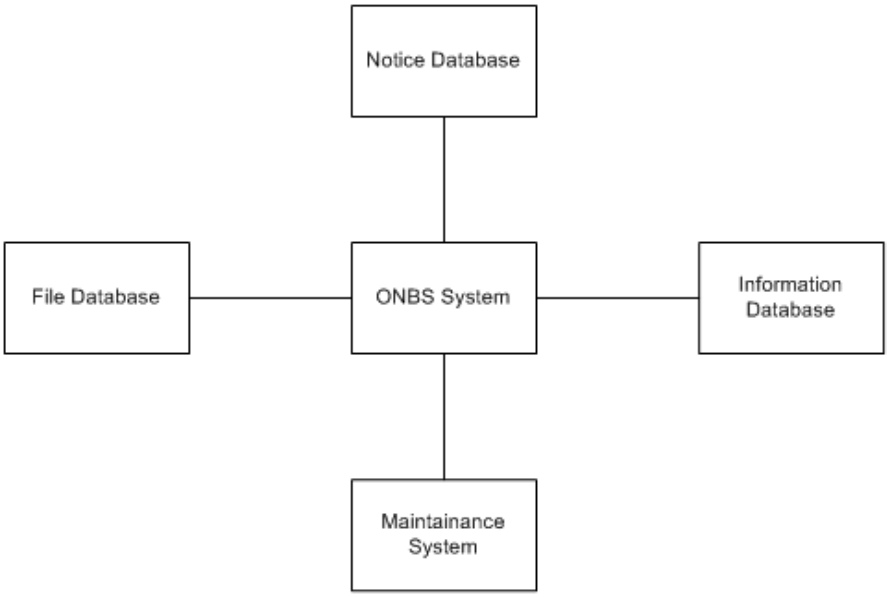


Figure 16-1 Block Diagram of the System

16.2 Usage Modeling

16.2.1 User Profile

User	Type
Guest	Public User
Student	Restricted User
Teacher	Manager
Admin	Administrator (Super User)

Table 16-1 User Profile of the System

16.2.2 Use Case Diagram

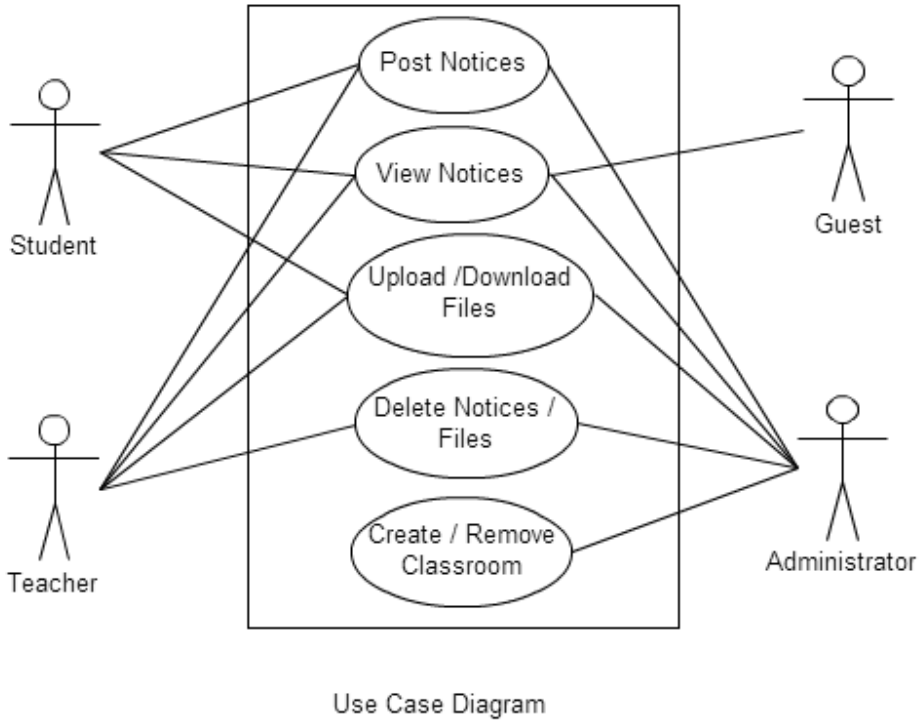


Figure 16-2 Use Case Diagram

16.2.3 Use Case Analysis

Actor	Action	Pre-condition	Post Condition	Alternative Action
Student	Login	Should have a valid username and password	Will get access to the Online Notice Board System	Reset username and password
				User Registration
	Post Notice		the notice will be displayed and pushed to the mobile system	
	Upload File	the size of the file should not overflow	file will be uploaded to the database	Prompt message will be displayed
	View Notice			
Teacher	Login	Should have a valid username and password	Will get access to the Online Notice Board System	Reset username and password
				User Registration
	Post Notice		the notice will be displayed and pushed to the mobile system	
	Upload File	the size of the file should not overflow	file will be uploaded to the database	Prompt message will be displayed
	View Notice			
Admin	Login	Should have a valid username and password	Will get access to the Online Notice Board System	Reset username and password
				User Registration
	Post Notice		the notice will be displayed and pushed to the mobile system	
	Upload File	the size of the file should not overflow	file will be uploaded to the database	Prompt message will be displayed
	View Notice			
	Monitor work			
Guest	View Notice			

Table 16-2 Use Case Analysis

16.3 Data Modeling

16.3.1 Data Dictionary

Faculty Room

Attribute	dataType	defaultValue	required	auto_increment	primary_key
roomID	alphanumeric	NONE	YES	YES	YES
roomName	alphanumeric	NONE	YES	NO	NO
NOticeList	alphanumeric	NONE	YES	NO	NO
fileList	alphanumeric	NONE	YES	NO	NO
teacherList	alphanumeric	NONE	YES	NO	NO

Admin

Attribute	dataType	defaultValue	required	auto_increment	primary_key
adminID	alphanumeric	NONE	YES	YES	YES
fullName	varchar	NONE	YES	NO	NO
userName	char	NONE	YES	NO	NO
password	md5	NONE	YES	NO	NO
email	char	mail@keckist.edu.np	YES	NO	NO
phoneNO	alphanumeric	NONE	NO	NO	NO

Class Room

Attribute	dataType	defaultValue	required	auto_increment	primary_key
roomID	alphanumeric	NONE	YES	YES	YES
roomName	alphanumeric	NONE	YES	NO	NO
NOticeList	alphanumeric	NONE	YES	NO	NO
fileList	alphanumeric	NONE	YES	NO	NO
teacherList	alphanumeric	NONE	YES	NO	NO

File

Attribute	dataType	defaultValue	required	auto_increment	primary_key
fileID	alphanumeric	NONE	YES	YES	YES
uploadDate	timestamp	NONE	YES	NO	NO
expireDate	timestamp	NONE	YES	NO	NO
expireOrNOt	boolean	NONE	YES	NO	NO
uploader	alphanumeric	NONE	YES	NO	NO
destination	alphanumeric	NONE	YES	NO	NO
fileFormat	char	NONE	YES	NO	NO

Notice

Attribute	dataType	defaultValue	required	auto_increment	primary_key
NOticeID	alphanumeric	NONE	YES	YES	YES
issueDate	timestamp	NONE	YES	NO	NO
expireDate	timestamp	NONE	YES	NO	NO
expireOrNOt	boolean	NONE	YES	NO	NO
issuer	alphanumeric	NONE	YES	NO	NO
destination	alphanumeric	NONE	YES	NO	NO
NOticeFormat	char	NONE	YES	NO	NO

Student

Attribute	dataType	defaultValue	required	auto_increment	primary_key
studentID	alphanumeric	NONE	YES	YES	YES
fullName	varchar	NONE	YES	NO	NO
userName	char	NONE	YES	NO	NO
password	md5	NONE	YES	NO	NO
profile	varchar	NONE	NO	NO	NO
NoticeList	alphanumeric	NONE	NO	NO	NO
fileList	alphanumeric	NONE	NO	NO	NO
classList	alphanumeric	NONE	NO	NO	NO
email	char	mail@keckist.edu.np	YES	NO	NO
phoneNO	alphanumeric	NONE	NO	NO	NO

Teacher

Attribute	dataType	defaultValue	required	auto_increment	primary_key
TeacherID	alphanumeric	NONE	YES	YES	YES
fullName	varchar	NONE	YES	NO	NO
userName	char	NONE	YES	NO	NO
password	md5	NONE	YES	NO	NO
post	char	Lecturer	YES	NO	NO
profile	varchar	NONE	NO	NO	NO
NoticeList	alphanumeric	NONE	NO	NO	NO
fileList	alphanumeric	NONE	NO	NO	NO
classList	alphanumeric	NONE	NO	NO	NO
facultyID	alphanumeric	NONE	YES	NO	NO
email	char	mail@keckist.edu.np	YES	NO	NO
phoneNO	alphanumeric	NONE	NO	NO	NO

Corridor

Attribute	dataType	defaultValue	required	auto_increment	primary_key
NoticeList	alp	NONE	YES	NO	NO

16.3.2 Data Description

FacultyRoom

Attribute	description	example
roomID	provides unique identification symbol to room	EX
roomName	provides name to the room	ElectronicsSixtySeven
noticeList	list of notices to be displayed in the room	067A020212R, 068C010110B
fileList	list of files to be displayed in the room	067A022212R, 068C041213B
teacherList	list of teachers associated with the room	[EXP001, EXF021, CTP014]

Admin

Attribute	description	example
adminID	provides unique identificaton symbol	adm001
fullName	full name of the person	Dakshina Shrestha
userName	user name to login	dakshina_kec
password	password to login	PBCfMJQ2h74SsHa.uqR903JHNk8kOCn/
email	email id of the person	mail@keckist.edu.np
phoneNo	phone number of the person	9841931178

ClassRoom

Attribute	description	example
roomID	provides unique identification number to the room	067BCTA
roomName	provides name to the room	ComputerSixtySevenA
noticeList	list of notices to be displayed in the room	067A020212R, 068C010110B
fileList	list of files to be displayed in the room	067A022212R, 068C041213B
teacherList	list of teachers associated with the room	[EXP001, EXF021, CTP014]
studentList	list of students belonging to the room	067bct001. 067bct002

File

Attribute	dataType	defaultValue
fileID	provides unique identification symbol to file	067C121212R
uploadDate	date of upload of file	12-12-12
expireDate	date of expiry of file	12-12-13
expireOrNot	if the file expires or not	yes
uploader	id of the uploader of file	067bct001
destination	id of the destination of file	068BCTB
fileFormat	image or text	pdf

Notice

Attribute	description	example
noticeID	provides unique identification symbol to notice	067C121212R
issueDate	date of issue of notice	12-12-12
expireDate	date of expiry of notice	12-12-13
expireOrNot	if the notice expires or not	yes
issuer	id of the issuer of notice	067bct001
destination	id of the destination of notice	EX
noticeFormat	image or text	text

Student

attribute	description	example
studentID	provides a unique identification symbol for the student	067BCT004
fullName	full name of the student	Anurag Acharya
userName		anurag_ach
password		PBCfMJQ2h74SsHa.uqR903JHNk8kOCn/
profile	a short description of the person	Psycho
noticeList	a list of all notices the person has uploaded	067A020212R, 068C010110B
fileList	a list of all files the person has uploaded	067A022212R, 068C041213B
classList	a list of all classes the person belongs to	067BCTA
email	mail id of the person	anurag.acharya@hotmail.com
phoneNo	phone number of the person	9841465878

Teacher

Attribute	description	example
TeacherID	provides unique identification symbol	bct001
fullName	full name of the teacher	Manish Aryal
userName	username for login	manish_aryal
password	password for login	PBCfMJQ2h74SsHa.uqR903JHNk8kOCn/
post	post of the teacher	hod
profile	a short description of the person	(Not Available)
noticeList	alphanumeric	067A020212R, 068C010110B
fileList	alphanumeric	067A022212R, 068C041213B
classList	alphanumeric	[067BCTA, 069BCEC]
facultyID	alphanumeric	EX
email	char	mail@keckist.edu.np
phoneNo	alphanumeric	01-12345678

Corridor

Attribute	description	example
noticeList	list of notices to be displayed	067C121212P

16.3.3 Entity Relation (ER) Diagram

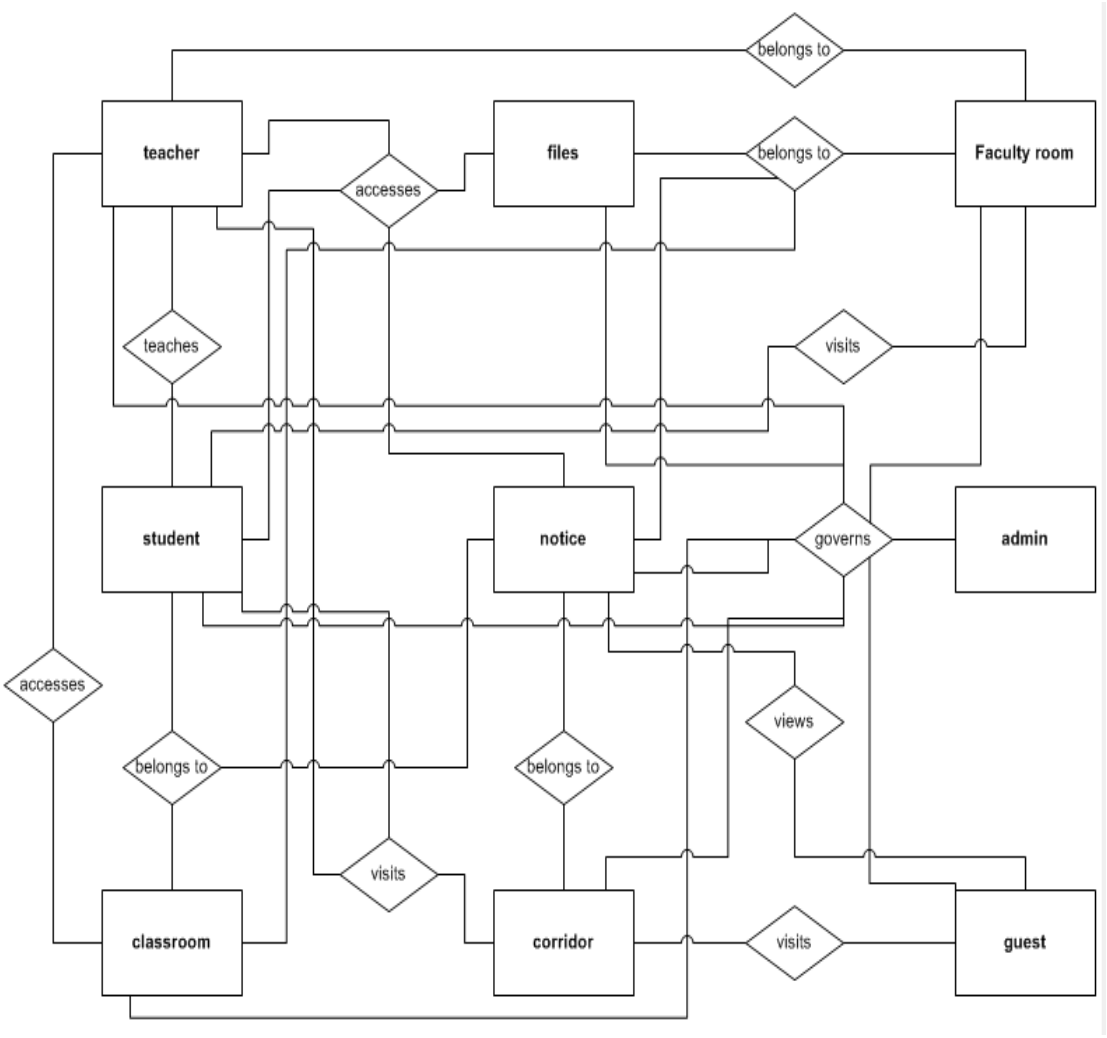


Figure 16-3 ER Diagram

16.4 Context Diagram

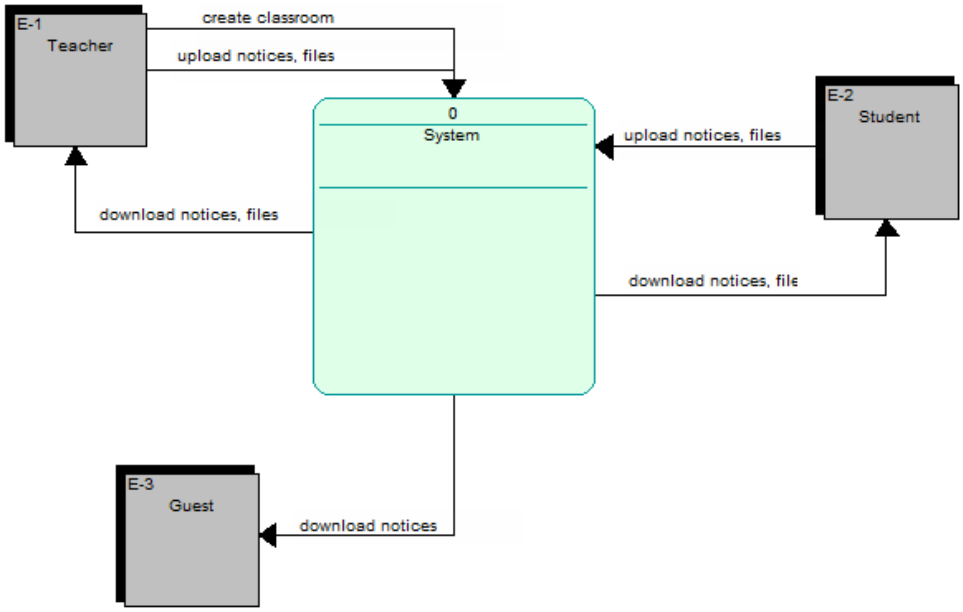


Figure 16-4 Context Diagram (Level 0 DFD)

16.5 Object Diagrams

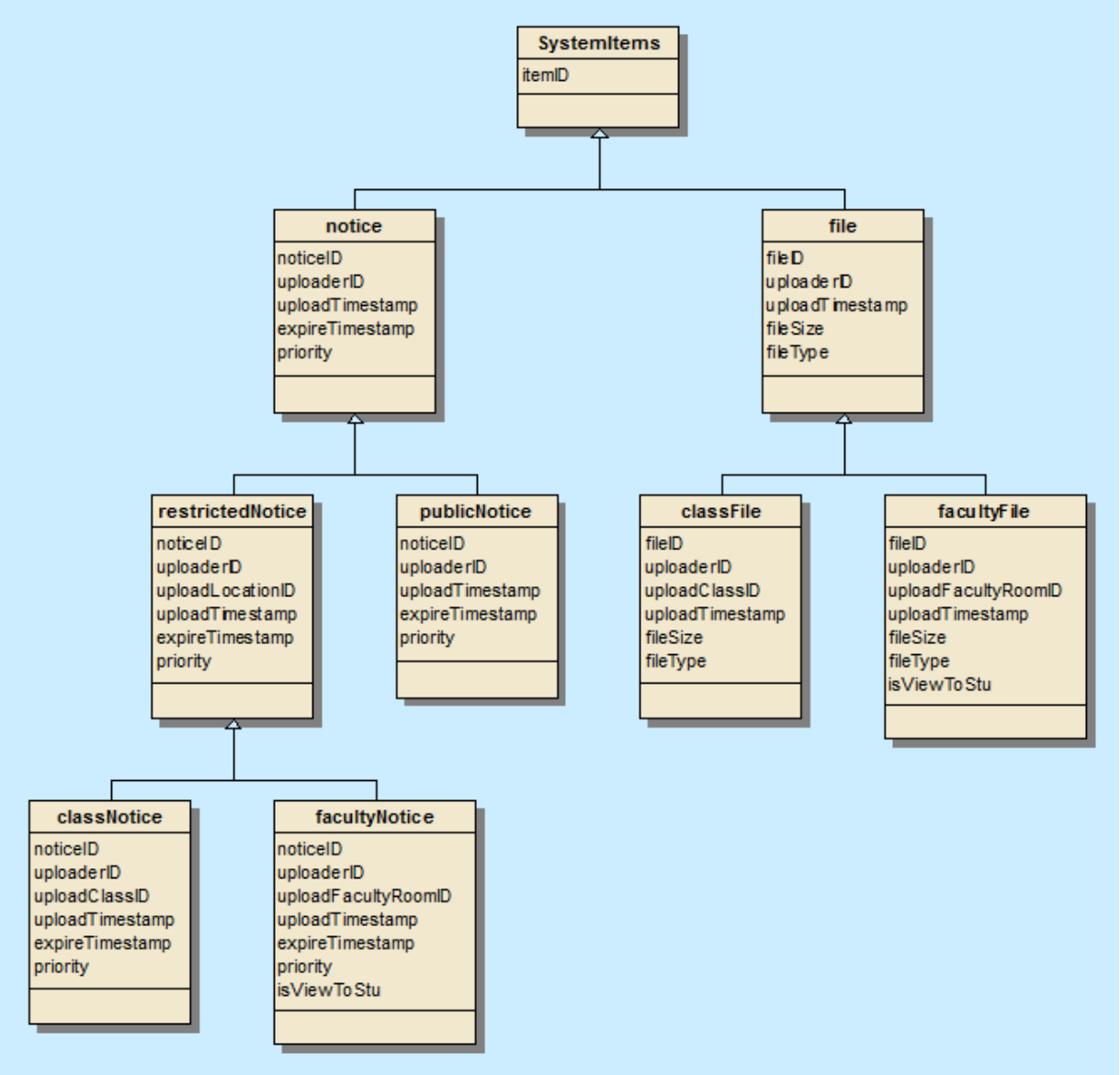


Figure 16-5 Object Diagram I

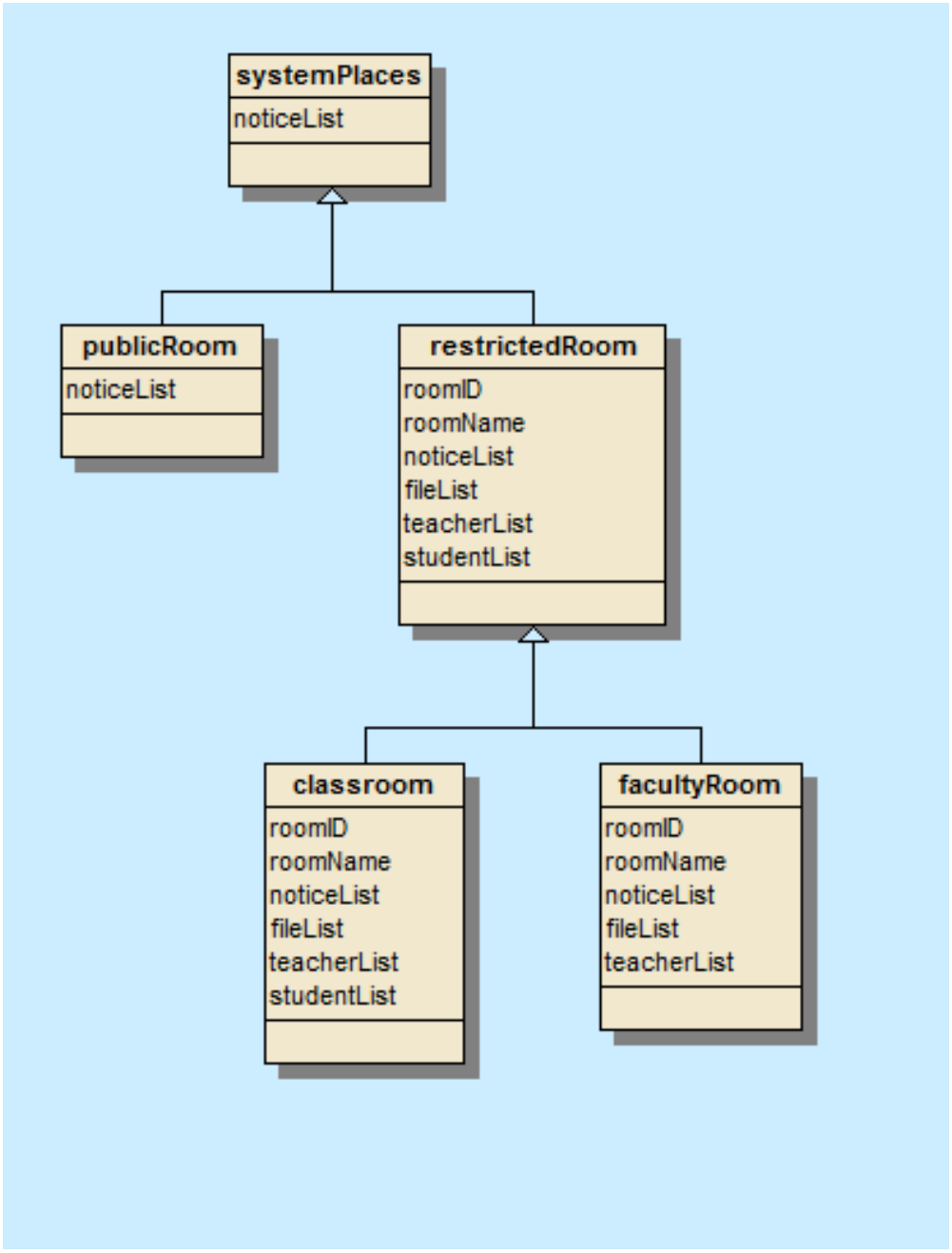


Figure 16-6 Object Diagram II

16.6 Class Diagram

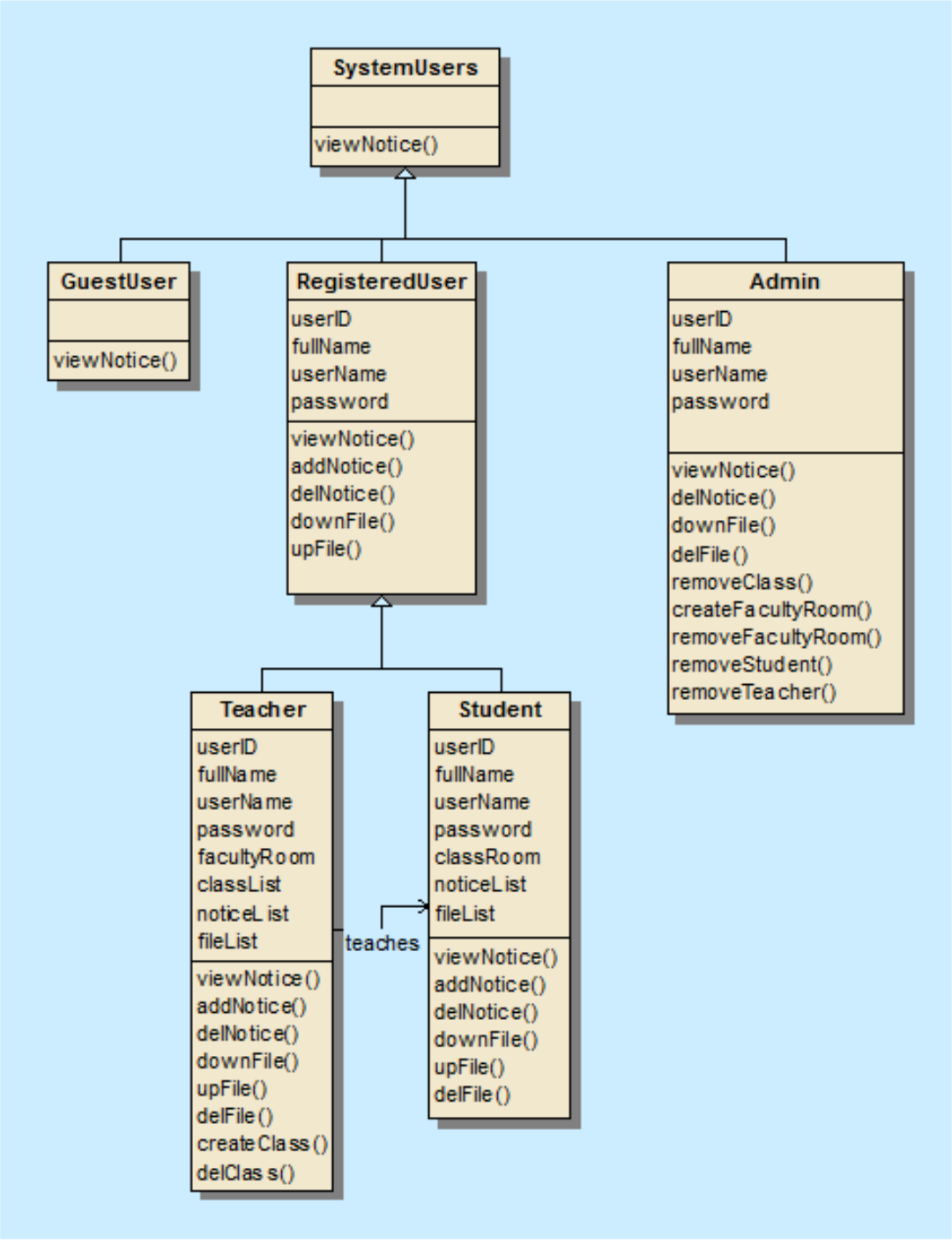


Figure 16-7 Class Diagram

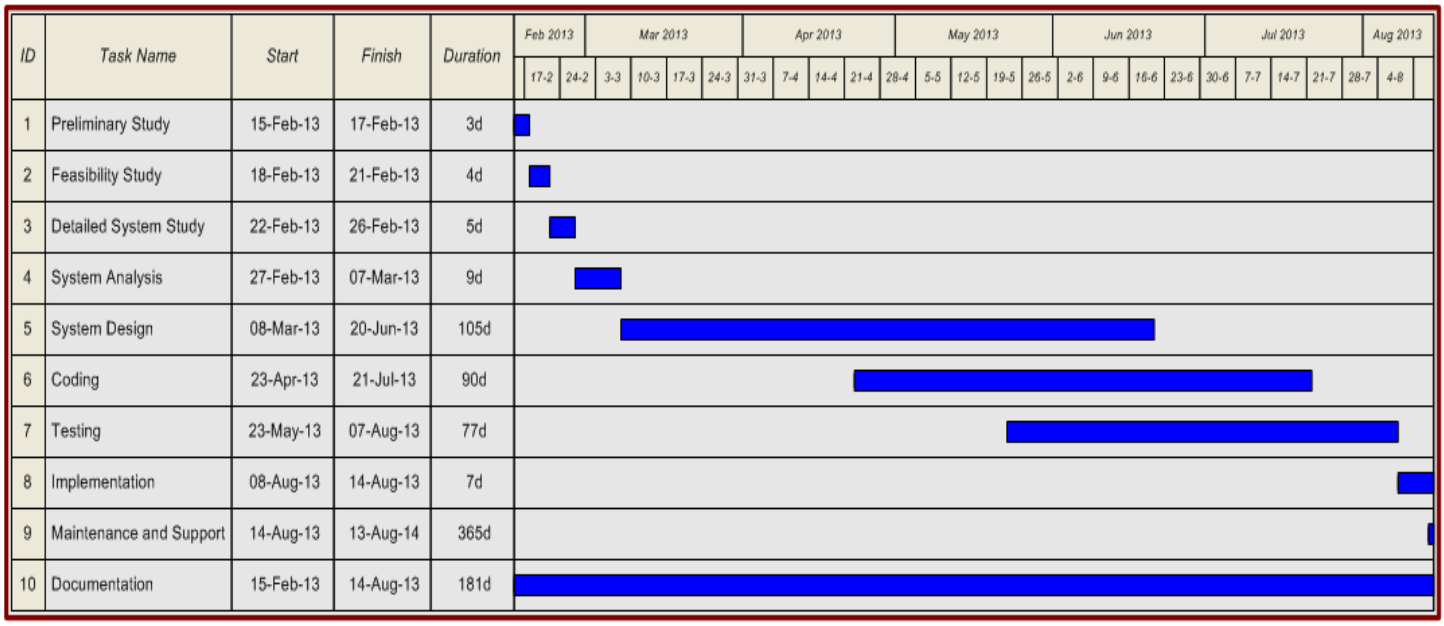
17 Cost Estimation

The following factors will draw financial expenses for the proposed system to be a fully-functional software:

- **Domain Name** and **Web Space** cost
- Cost for **application registration** on respective markets
- Platform and language specific **trainings and courses**

18 Gantt Chart

The following is a Gantt chart representing the scheduling of the project for the initial version of the software (V.1.0.0).



ID	Task Name	Start	Finish	Duration
1	Preliminary Study	15-Feb-13	17-Feb-13	3d
2	Feasibility Study	18-Feb-13	21-Feb-13	4d
3	Detailed System Study	22-Feb-13	26-Feb-13	5d
4	System Analysis	27-Feb-13	07-Mar-13	9d
5	System Design	08-Mar-13	20-Jun-13	105d
6	Coding	23-Apr-13	21-Jul-13	90d
7	Testing	23-May-13	07-Aug-13	77d
8	Implementation	08-Aug-13	14-Aug-13	7d
9	Maintenance and Support	14-Aug-13	13-Aug-14	365d
10	Documentation	15-Feb-13	14-Aug-13	181d

Figure 18-1 Gantt Chart

19 References

- Feasibility Analysis Guide, Department of Computer Science, Cornell University
  http://www.cs.cornell.edu

- Drewry, Tony, *Logical Data Structures (LDSs) - Getting started*, University of West of England, 2005
  http://www.cems.uwe.ac.uk/tdrewry/lds.htm

- Prof. H. Levent AKIN, Department of Computer Engineering, Boğaziçi University, Istanbul, Turkey
  http://www.cmpe.boun.edu.tr/~akin/

- *IEEE Recommended Practice for Software Requirement Specifications*, IEEE Standard 830–ISM4331

- *IEEE Recommended Practice for Software Design Descriptions*, IEEE Standard 1016–1998