Table of Contents

List of Tables

List of Figures

Introduction

1.1 Topic of the system

The system being developed is '*College Accountant'*. This is a college accounting software being developed exclusively for various colleges, institutes, universities etc.

1.2 Purpose of Implementation

In the current scenario, various colleges use archaic methods for managing the accounts of their college/ institute. Many colleges are still using manual records and others have invested in costly software for accounts-keeping. Some software like TALLY, QuickBooks, and Microsoft Money are expensive. Also, these software are huge applications with vast number of features, but colleges/universities hardly use more than 30-40 % of those features. In some colleges, they have implemented various management systems for various departments, for example, library management system, inventory/store management system, attendance-marking system etc. Implementing such management systems does reduce the load of managing a huge pile of data which would otherwise have been recorded on hard-bound registers and stored in lockers/cupboards. At the end of the day any organisation is eager to know how much it spent and how much it saved. So, the accounting part of these management systems need to be linked to a single system which would manage all money related transactions which would lead to better management of monetary resources. For example, in Library Management System, the fine collection, should be linked to Accounts system.

1.3 Target Reader

Target Reader/Audience for this system would be all those people who are directly or indirectly related to the account keeping of the college. The various users and their specific roles in relation to the system are as follows:

1.3.1 Directly Involved

S. No.	User	Roles
1.	**Accountant**	Manage accounts for college, issue receipts/bill, generate reports, send notifications etc.
2.	**Students**	Check/Payment of fees, fines etc.

Table 1: Directly Involved Target Audience

1.3.2 Indirectly Involved

S. No.	User	Roles
3.	**Librarian**	Fines on late submission /damages books, order new books/magazines etc.
4.	**Faculties/Staff**	Payroll Management
5.	**Examination Controller**	Issue admit card for exams after checking if all previous dues have been cleared
6.	**Hostel In-charge**	Check for payment of previous dues towards hostel fees/fines
7.	**Store In-charge**	Purchase of items, bill payments
8.	**Bus In-charge**	Maintenance and fuel charge for buses
9.	**Mess/Café In-charge**	Purchase of items , bill payments
10.	**Director**	Reports
11.	**College Management**	Reports

Table 2: Indirectly involved Target Audience

1.4 Topic Background

The system '*COLLEGE ACCOUNTANT*' is being developed for colleges/ universities/ institutes. It is an all-round software solution for managing the accounts of the college. This would be an online as well as a desktop based application. The online version of the system provides a gateway for the students to pay the tuition fees, examination fees, library fines, late fee fines, re-examination fees etc. Along with these students can view the previous transaction history. Also, students who have to obtain any refunds from the college can apply for the same. The desktop application is mainly to be used by the accountant. The system would very well integrate the payroll system for the faculties and other staffs of the college. The examination fee-fine, library, bus, hostel, mess, cafeteria, bill payment, inventory and store modules are also being integrated to this system. Notifications in the form of SMS and email are being provided. Graphical reports can be generated on the flick of a button. These reports help the management to take better decisions.

1.5 Problem Context

In the present scenario, there is no link between any of the management systems of the college. College uses Tally for accounting purposes. There is a constant growth in the amount of data. Also, there is also lot of wastage of resources. Poor security measures add up to the problems of the current system. Data can't be checked for accuracy and concurrency.

The problems being faced in the current system are:

✓ Use of expensive accounting software: TALLY is being presently used for managing the accounts in the college. Tally is an ERP and is feature-rich software. However, most of the features are not required in the current environment. And it costs a bomb to include additional features into the previously procured software.
✓ Experienced users required to operate system: Tally being complex software to handle needs to be operated by only experienced users. Prior training is required to exploit the full potential of this software. This makes the software less user friendly.
✓ Security: the current system provides very less security measures. Since, the system deals with money it is very important to make the system secure. There is increased security risk of web-based systems even if the hosts ensures that the system is fool-proof from hacking attacks.
✓ Waste of resources: the current system leads to wastage of many resources. Lot of wastage of paper in printing of receipts etc. Wastage of time and energy of students when students have to pay library fines. They first need to get a receipt from the librarian, then pay the fine at accounts and then go back to library to verify further details.

✓ Data accuracy and concurrency: there are no measures to check if the data entered is correct or not. The system doesn't check for ambiguous data. Since there is no linking between modules. Also, there are no measures to check for data concurrency.
✓ System upgrades: upgrading of the system is a costly affair. And it is important to keep the software updated with current times.
✓ Poor budget forecasting: Since reports being generated are poor or in some cases insufficient reports are being generated which otherwise could have helped in budget forecasting and hence proper measures could be taken to manage funds.

Table 3: Problem Context

1.6 Rationale

As discussed, there are various problems in the current system in method of account keeping. If the system 'College Accountant' is implemented, it can assure these Tangible and Intangible benefits be achieved:

1.6.1 Tangible Benefits

✓ Productivity of the process and personnel is improved. For example, the accountant can easily calculate the library fine for a student in the proposed system thus making the fine calculation process faster and easier and increasing the productivity of the accountant who otherwise would had have to calculate the fine manually by comparing the due date and the actual date of return of the book.
✓ Since the system is computerized and the data stored in databases, there is reduction in paper and postage records. A soft copy of fee/fine receipts can be mailed to the students.
✓ Addition of Payment gateway would increase the timely submission of fees/fines by the students.

✓	Automatic calculation of payroll of the staff reduces the workload from the accountant and improves accuracy and productivity.
✓	Better management of store/inventory would help in reducing stock mis-utilization.
✓	Third-party expenses like electricity, water, internet, maintenance etc. help in better management of the monetary resources.

Table 4: Tangible Benefits

1.6.2 In-Tangible Benefits

✓	Enormous amount of time and labour can be saved in data entry.
✓	Uniform report generation can facilitate in strategic planning of the monetary resources by concerned college authorities/management
✓	Increase in transparency and responsibility of the processes being done in the college/universities.
✓	Accurate and faster access to the data thus reducing the risk of mis-utilization of resources
✓	User friendly – a self-explaining system hence further user training need not to be imparted on the target users.
✓	Email and SMS notification/reminders have been provided which increases the customer response to the system.

Table 5: Intangible Benefits

1.7 Objectives of the System

The main objective of this project is to provide a proper Accounts keeping solution for Colleges, Universities and Institutes. The other objectives of this system are:

1.7.1 Project Objectives

- ✓ Provide the students with an option to pay their fines/fees online
- ✓ Manage Payroll of the faculties and staff of the college

- ✓ Manage Library fines and ordering of new books/magazines etc.
- ✓ Manage Examination fees, re-exam fees, etc.
- ✓ Manage the in-out expenditure of the Store, Mess and Café
- ✓ Provide with proper notifications and reminders
- ✓ Provide with graphical reports for review for the Director and college management team.
- ✓ Provide with a User-friendly interface

1.7. 2 Academic Objectives

- ✓ Having thorough knowledge of J2SE and J2EE.
- ✓ CAPTCHA code and other types of protection to make the system more secure
- ✓ Hibernate and MVC architecture to make database more effective
- ✓ Having thorough knowledge of Tally ERP
- ✓ Study other Accounting systems
- ✓ Research more about Payment gateway
- ✓ Study the concept of sending Emails and SMS
- ✓ Design User Interface using HCIU Principles

1.8 Functionalities

The developer has distributed whole project in various modules each under as core, enhanced and special functionality:

1.8.1 Core Functions

- ✓ *Student Fee Module:* In this module the fee to be submitted by a student corresponding to the year and semester they are studying is calculated. The system checks if other fees like registration, counselling, convocation, is paid or not. It also checks for any applicable scholarships, concessions etc. Late fee fine is calculated according to the fine slabs.

- ✓ *Payroll Management:* this module deals with the payment of salary of the staff and faculties of the college. The system calculates the amount of days worked to arrive at the salary to be given. If any bonuses or deductions are applicable they are adjusted to the above calculated salary. The salaries are deposited directly into the accounts of the staff and faculties.

- ✓ *Library Module:* this module calculates for the fine to be paid by the students for the books issued. Fine could be either late book return or misplaced/damaged books. Also, it helps in keeping a check on the books which are being ordered as new or as replacement or due to shortage.

- ✓ *Inventory/ Store / Departmental:* this module keeps a check on the items which are being ordered for the store. Items could be like sports equipment, furniture, markers, dusters, cleaning equipment, gardening equipment etc. Also keep a check on the items which are being sold off. To check for the rent payment to be received from the departmental store owner.

- ✓ *Examination Module:* this module checks for the attendance of the students. If a student has low attendance then he/she will have to pay the calculated amount to sit for extra class to recover for the classes which have been missed. This module also checks for the performance of the students. If a student fails in the internal or external then the required payment has to be paid by the student for sitting in the exams.

1.8.2 Enhanced Functions

- ✓ *Hostel Module:* this module checks for the payment received by students in respect of the hostel chosen by them, ac or non ac rooms, single-double-triple bedrooms etc. It also checks for the miscellaneous expenses of the hostel like laundry, newspaper, damaged goods replacement etc.

- ✓ *Bus Module:* this module keeps track of all the payments which have been received in respect of the bus services provided to the students and faculties. It also covers the fuel and maintenance charges of the buses.

- ✓ *Maintenance and Development:* this module calculates the expenses of the college in terms of building maintenance and improving the present infrastructure.

- ✓ *Mess /Cafeteria Module:* this module checks for the mess fee payment by student and faculties. Total charges to be calculated for those who avail the mess facilities occasionally. To check for the rent payment to be received from the mess and cafeteria owners.

- ✓ *Bill Payments:* this module checks for the payment the college has to do to keep various third-party services active. These services include the payment for the ISP provider, security, electricity, water, etc. Also, scanned copies of bill can be stored in the database.

- ✓ *Secure Login:* the system would provide secure login to its users with the help of CAPTCHA code implementation in order to prevent unauthorized login. Also, password and other essential data would be encrypted.

- ✓ *Three-tier Application*: the system too be developed would follow the three tier-architecture approach where the business logic, presentation layer and the data layer would be separate.

- ✓ *Notifications / History:* SMS / email notifications can be sent to the defaulters. Reminders can be sent beforehand so that students can pay the fees in time. Notifications can be sent to the staff and faculties on successful transfer of the salaries to their accounts. The whole account history of the students can be viewed in a glance.

1.8.3 Special Functions

- ✓ *Reports Generator:* Weekly/Monthly/Yearly reports can be generated showing the income and expenses of the college. These reports would be of importance to the management of the college which could help them further in the better managing of the college funds. These reports can be generated graphically for better understanding. Receipts could be printed for future references. Payslips would be generated along with the student defaulter lists.

✓ *Payment Gateway:* would be developed so that students can easily pay the fees online at any time. They can pay the fees by credit card, net banking etc. and can pay the college fees, fines, or other charges which needs to be paid to the college. This is more secure method and payments can be easily tracked.

Table 6: Functionalities

1.8.4 Evidence Limiting the Scope of Project

Although additional functionalities could be added to this project; however the project scope is confined to a specified time, budget and defined functionalities. Features which have not been provided in this project but could be a part of future enhancement are:

- ✓ Live payment gateway: in the current system due to legal restrictions, the developer had to implement a dummy database. However, in the future settling legal issues would help in making the gateway functional.
- ✓ Linking bank accounts/credit card: due to government rules and regulations the developer wasn't able link bank accounts/credit cards, which may be added in the future.
- ✓ Mobile Application: a mobile application for the same could be created showing only the necessary parts of this vast system.
- ✓ Attendance logs: RFID based attendance marking system could be implemented to ensure transparency of operations.
- ✓ Bar-code Reader: the store could use bar code reader to input the details of items automatically into the system, ensuring full transparency.

1.8.5 Assumptions

Since this proposed system is being designed to work in tandem with the other management systems of the college/university it is being assumed that the college/university implementing this software has already few of the management systems like library management system, attendance marking system for staff and students, student records, inventory management system, etc. implemented in their college. It is also assumed that data is accessed from table of the database of these management systems to be used in the proposed system.

1.9 Project Deliverables/ Milestones

	S No.	**PROJECT MILESTONE**	**DELIVERABLES**
Inception Phase	**1.**	Topic Selection	Draft Proposal Form
	2.	Topic Finalization	Project Proposal Form
	3.	Primary Research	Questionnaire, Interviews, Focus Group Study, Observation
	4.	Project Planning	Scope & Objectives Definition
	5.	Project Scheduling	Gantt Chart, Project development Plan
Elaboration Phase			
	6.	Secondary Research	Literature Review Documents
	7.	Project Primary Analysis Completion	Summarize Planning & Research, Use Case, Class Diagrams.
Construction Phase	**8.**	Project Secondary Analysis Completion	Activity and Sequence Diagrams
	9.	System Design Completion	Database Modelling Diagram (ERD, Database Design, and Normalization), System Interface Design.
	10.	System Implementation	System prototype is ready to be reviewed by users
Transition Phase	**11.**	System Testing	Test cases are prepared according to different type of testing
	12.	Project Documentation	Final Documentation and User Manual.
	13.	Project Close-Out	Critical Evaluation Report, Conclusion, Presentation and System Demonstration.

Table 7: Project Deliverable/Milestones

1.10 Evaluation on Success Criteria

To evaluate on how successful a project is, these criteria are considered:

- ✓ *Meeting user requirements* – A project which can fulfil all the requirements and expectations of all users is called a successful project. This project would be deemed successful if the system is able to fetch the data from various tables of the database and be able to do the necessary calculations. **Evaluator:** End users
- ✓ *Functionality* – Functionality is determined by the working of the system. The system functionality such as interaction of different module is working or not, proper calculations being done or not report has been generated or not. **Evaluator**: End users, Developer
- ✓ *Usability* – Elements such as user acceptance, accessibility, familiarity and recoverability will be evaluated here. These are the basic HCIU principles which would be looked upon on the interface of the system. **Evaluator**: End users
- ✓ *Level of Content* – it shows how well the information and content is available in the system and how much deep the contents are. **Evaluator:** End users
- ✓ *Research and Analysis* – A good amount of well investigated research and analysis work is important to the success of a project. How well the market research has been conducted. **Evaluator**: System Analyst
- ✓ *Documentation* – The ability to come up with a documentation which is very well formatted and good English and grammars are used. The document should be self-explanatory with clear diagrams and figures. **Evaluator**: End User

1.11 Conclusion

In this chapter 'Introduction', the project has been introduced, the target users identified, the benefits highlighted, the objectives of the system understood, functionalities of the system described and eventually laying out a plan on how this project would be completed.

2. Problem Description

2.1 Problem Area

The preliminary study and its analysis helps in clearly defining the functional and non-functional areas of the proposed system .With the help of available resources like web sites, research papers, journals etc. the developer does further investigations in mentioned areas , which results in following:

✓ *Improper integration with other management systems within an organisation*

The problem of integrating other management systems of the organization [college, universities in this case] is most frowned upon. One may have to work from scratch to implement the previously implemented management systems to an accounting system. Also, the data to be circulated among these systems should be in a similar format.

✓ *Expensive accounting software in market*

The other marketing software available in the market are either full of features or expensive or are free / cheap but have only basic functionalities. In some cases the purchased software contain loads of features and half of these are not even applicable for the organization.

✓ *Extensive User training required*

Majority of accounting software available in the market are complex and are difficult to operate by a layman. Extensive user training has to be imparted on the user before an attempt to operate the software is made. A user with good accounting skills and basic computer knowledge will not be able to use these software to the fullest.

✓ *Extensive User training required*

Majority of accounting software available in the market are complex and are difficult to operate by a layman. Extensive user training has to be imparted on the user before an attempt to operate the software is made. A user with good accounting skills and basic computer knowledge will not be able to use these software to the fullest.

✓ *Inefficient data manipulation*

There are very less software available which checks if the data is accurate, concurrent and non-redundant. Since there is no or very less integration between various modules which leads to these problems. Inconsistent data could lead to huge losses to the organization.

✓ *Waste of valuable resources*

In current environment, accounts keeping leads to wastage of resources such as time, paper, money etc. There is reduced efficiency and productivity of work. These need to be improved upon.

✓ *No Payment options*

Various software present in the market don't provide with the feature for online payment options for its users. It would have been an added advantage if such a feature is provided since tracking of transactions would be much easier.

Table 8: Problem Area

2.2 Nature of Challenge

This accounting software includes various modules of the college system, hence is considered to be a complex and time consuming which in itself is a herculean task. However the various challenges in developing the system would be:

2.2.1 Domain Challenge

- ✓ Collection of data: Since more than ten modules are being integrated into this system, hence a lot of useful data needs to be collected. The data needs to be analysed and checked for accuracy.
- ✓ Integrating the Modules: all these modules need to be made compatible and integrated properly with the system.
- ✓ Report Generation: extensive graphical reports are to be generated. For this the data collected need to be made compatible and used to generate the reports.
- ✓ Understand Account-keeping: being from a science background, the developer has very less knowledge of Account-keeping and hence the developer needs to study about this in detail in order to produce correct results.
- ✓ User-friendly: making the system as user friendly as possible. Designing the system by keeping in mind the various kinds of users [Novice, expert, casual]. Making the interface using HCIU Guidelines.

2.2.2 Technology

- ✓ Advanced knowledge of J2SE [java 2 Standard edition] and J2EE [java 2 Enterprise edition] is required to implement these features.
- ✓ Difficult concepts of database needs to be studied which are to be implemented along with gaining mastery on MySQL, which is the chosen database for this system.

2.2.3 Others [Infrastructural]

- ✓ Payment Gateway: a secure online payment gateway, like PayPal is being provided for the students to pay up their fees/fines. Developing and implementing this gateway would be a great challenge.
- ✓ SMS Gateway: the system requires to send SMS notifications and reminder which would require integration SMS services to the system which requires purchasing of bulk messages followed various legal rules and regulations.
- ✓ Hosting: for the payment gateway to work the database needs to be hosted on a web server so that transactional changes can be reflected back into the desktop application. For this purpose the developer needs to purchase a domain name and web-space.

Table 9: Nature of Challenge

2.3 Feasibility Study

Feasibility study is the analysis and evaluation of a project / system to check if it is technically and financially feasible and would be profitable after implementation. This study decides if one should go ahead with the project or not. Data gathered through the primary research is used in this study. This study is further classified into:

2.3.1 Economic Feasibility

Economic Feasibility or Cost Benefit Analysis is the technique of calculating the costs and financial benefits of a proposed system. Since, the proposed system, College Accountant is a complex and vast system, the best approach for cost estimation would be **Bottom-Up** Estimation.

- ✓ *Reason for choosing Bottom-up Approach*: In Bottom-Up approach, the estimation starts from the lowest level of the Work-Breakdown Structure and later they are aggregated upward into estimates of higher level deliverables and the project as a whole. Hence, cost estimate of each module can be done. Thus, the client depending on the budget can decide which all modules to be included in the proposed system. This approach produces the most accurate and refined result which is of most importance for such a vast project. (Billows, 2013)

- ✓ *Budget Estimation:* The proposed system is being developed by the developer, for partial fulfilment of his BE (Hons.) Software degree. Therefore, being a student, he would be using the available resources round him to develop the proposed system and there won't be any requirement of cost/budget estimation.

2.3.2 Technical Feasibility

Technical Feasibility is the technique of analysing the technical resources [hardware and software] which the organization has or may have to acquire in order to successfully implement the proposed system.

- ✓ Hardware Resources

The hardware resources which would be required for the proper functioning of this system would be:

Desktop Computer/Laptop
Processor : Core2 Duo/Dual Core/ Core i3/Core i5/ Core i7 equivalent or above
Memory : 2GB RAM or more
Hard-drive: 100GB or more [approx.. 100MB to install application]
Monitor Resolution : 1024 x 768 or Higher [Graphic card not required]
Server: for hosting the website part of the project.
Modem: [wired/wireless] for seamless Internet connection
All-in-One Printer: Copier-Scanner-Printer [Fax optional]
Ergonomic Keyboard and Mouse

Table 10: Hardware Resources Required

- ✓ Software Resource

The software resources required for the successful development of this project are as follows:

OS : Windows 98 / 2000 / ME / XP / Server 2003 / Vista / Server 2008 / Windows 7 / Windows 8 / Server 2012
IDE : Net Beans 7.4; Eclipse; with JDK 1.7
Database : MySQL 5.0 and NaviCat
Web Browsers : IE 6 or later, Chrome, Firefox, Opera
Documentation: Microsoft Office Word, Microsoft Visio, Microsoft Power-point, Microsoft Office Project, Adobe reader, Microsoft Paint.

Table 11: Software Resources Required

2.3.3 Schedule Feasibility

The overall period of the development of project involves strict intermediate deliverables and milestones and accordingly. Proper **Gantt chart** has been prepared to keep the track of all the activities with their completion dates hence the project will developed under rigid constraints of time so the project will definitely schedule feasible.

Note: *Please refer to Appendix A for Gantt chart and Project Development Plan [Work-Breakdown Structure]*

2.3.4 Operational Feasibility

Operational feasibility is mainly concerned with issues like whether the system will be used if it is developed and implemented. But as per the developer's research, it is found that majority of the college/ universities are really suffering from improper accounts keeping and the proposed system is going to designed will help in minimizing their problems. Also the system will design by following the proper HCIU principles to make the interface more user friendly so that the new users can easily understand the workflow of the system and easily perform their tasks. Hence it can be concluded that the proposed system is operational feasible.

2.3.5 Legal Feasibility

- ✓ Unauthorized access to essential records: If a person hacks into the system and misuses/alters the records, this may lead to legal issues arising due to unauthorized access to records. This is primarily one intentionally due to some negative intentions.
- ✓ Increase in un-employability: since this system is automated, thus it reduces the labour and sometimes employees would be laid-off due to no work. And legal issues may arise under the Human Rights law.
- ✓ Privacy: sometimes some records like salary, marks etc. can create privacy issues. And if that information is provided to a third-party, it possess great danger and may lead to losses for the company.

2.4 Conclusion

In this chapter, 'Problem Description' the problems have been explained in detail along with the challenges involved in developing such a system. Also, the feasibility study was done in this chapter.

3. Literature Review

3.1 Advanced Preliminary

The developer is a final year student at a college. The idea of this project ignited in the developers mind when he himself was annoyed by the functioning of the accounts department at his college.

At his college various management systems have been implemented but none of them were linked with each other or to a single main system. This created confusion. For example, if a student needed to pay a library fine, firstly he had to visit the librarian, get a note signed by the librarian mentioning the fine amount payable. Then, this note has to be shown to the accountant and then the fine has to be paid in cash to the accounts office. On successful payment of the fine, the accountant would sign the note. Now the student visits back to the library, shows the note and gets his name cut off from the list of defaulters. Then the student is applicable to sit for exams and issue further books. This whole process is depended on a small piece of paper and if that is lost, one needs to repeat the whole process. The overall process is time-consuming and leads to too much wastage of resources.

The developer visited neighbouring colleges and found this accounting problem to be much worse in those colleges.

Below are the references taken by the developer for this project:

- ✓ Questionnaire filled up by an Accountant working at a college
- ✓ Interview with a member of the College Management.

Conclusion

The basic idea of the project is incidentally found on real time scenario and developer performed research on various sources where it was found that there is a great need to implement such type of accounting software for college / universities.

3.2 Secondary Research

3.2.1 Domain Research

Domain research is required as project domains vary in various cases and once domain research is accomplished in particular module, then analysis and designing of particular module provides ease of its use.

- ✓ **Report Generation**– the types of report which can be generated for the system. There are various types of graphical reports which can be generated for the system are: Bar Graphs, Line Graphs and Pie Charts. These are used to convey the information in such a way that the viewer can easily understand it and clearly draw a difference between different pieces of data. These graphs and charts share some similarities and also differ from each other in some way which needs to be clearly examined. (Mathworksheetscenter, 2010)

Line Graph: in this representation two or more lines are used to compare different information showing how they are familiar and how they differ.

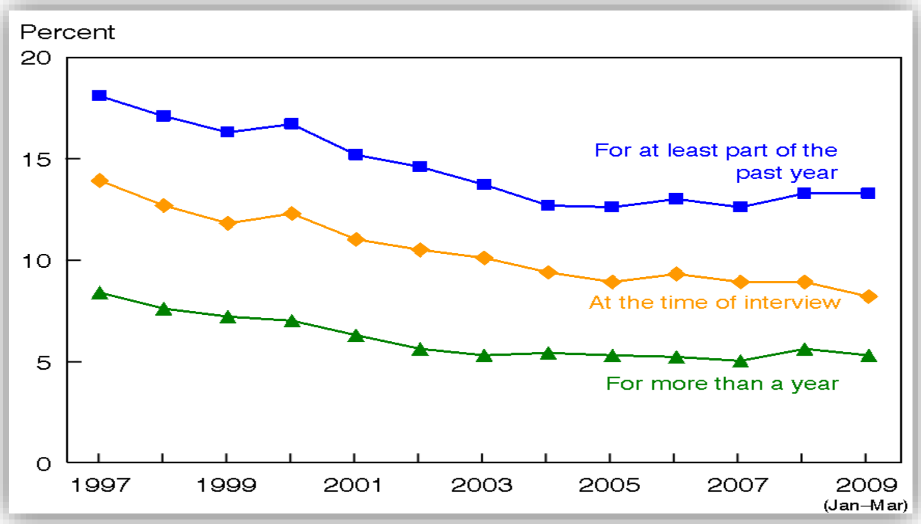


Figure 1: Line graph Example (Line Graph, n.d.)

Bar Graph: it is designed to show different values of two or more subjects using horizontal and vertical bars that represent a different value. There are numbers along the side of a bar graph and they are scales. This type of graph is somewhat easier to read than a line graph and it conveys informational equally as well.

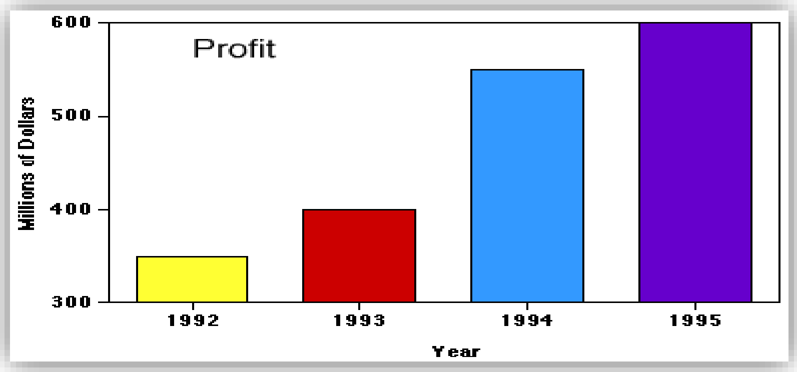


Figure 2: Bar graph Example (Bar graph, n.d.)

Pie Chart: is designed to show differences between two separate subjects although it eschews the common linear style found in the two other graphs. It is a very common type of graph that is in the shape of a circle with the circle representing a collective of 100%. Then, within the circle smaller percentage portions within the 100% will be presented in different colors.

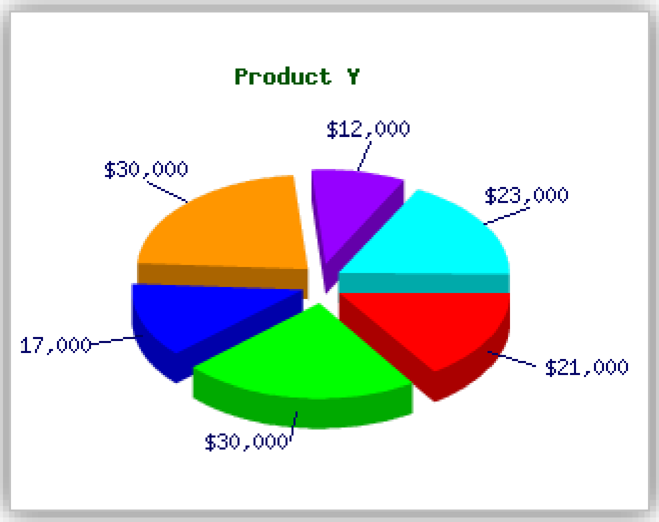


Figure 3: Pie Chart Example (Pie Chart, n.d.)

- ✓ **Email Notifications** – to send email notification in java, JavaMail API has to be used. It provides classes that model a mail system. The javax.mail package defines classes that are common to all mail systems. The javax.mail.internet package defines classes that are specific to mail systems based on internet standards such as MIME, SMTP, POP3, and IMAP. It includes the javax.mail package and sub packages. (JavaMail, 2008)

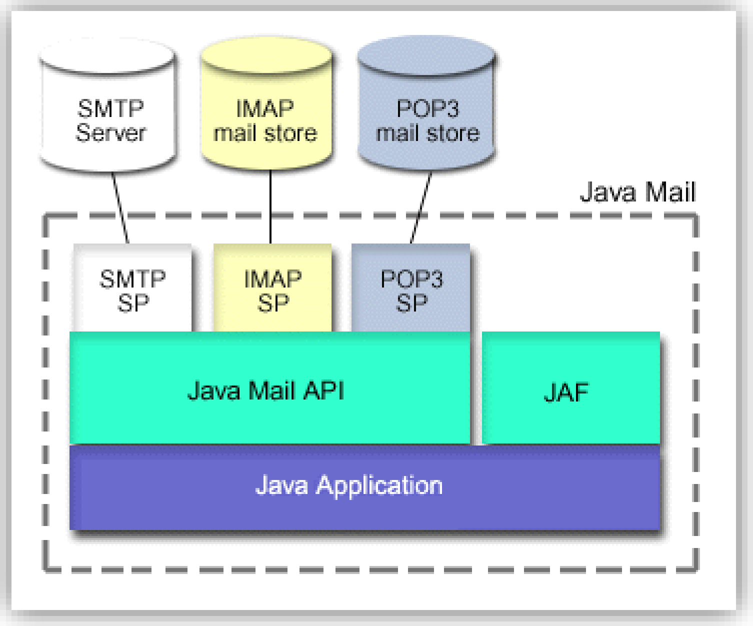


Figure 4: Java Mail API (JavaMail API, n.d.)

JavaMail API Packages	
Package	**Description**
javax.mail	The JavaMail API provides classes that model a mail system.
javax.mail.event	Listeners and events for the JavaMail API.
javax.mail.internet	Classes specific to Internet mail systems.
javax.mail.search	Message search terms for the JavaMail API.
javax.mail.util	JavaMail API utility classes.

Table 12: JavaMail API Packages

- ✓ **SMS Notifications** – SMS stands for **S**hort **M**essage **S**ervice. It is a technology that enables the sending and receiving of messages between mobile phones. As suggested by the name "Short Message Service", the data that can be held by an SMS message is very limited. One SMS message can contain at most 140 bytes (1120 bits) of data, so one SMS message can contain up to:

- 160 characters if 7-bit character encoding is used.
- 70 characters if 16-bit Unicode character encoding is used.

SMS is supported by 100% GSM mobile phones. Almost all subscription plans provided by wireless carriers include inexpensive SMS messaging service. Unlike SMS, mobile technologies such as WAP and mobile Java are not supported on many old mobile phone models.

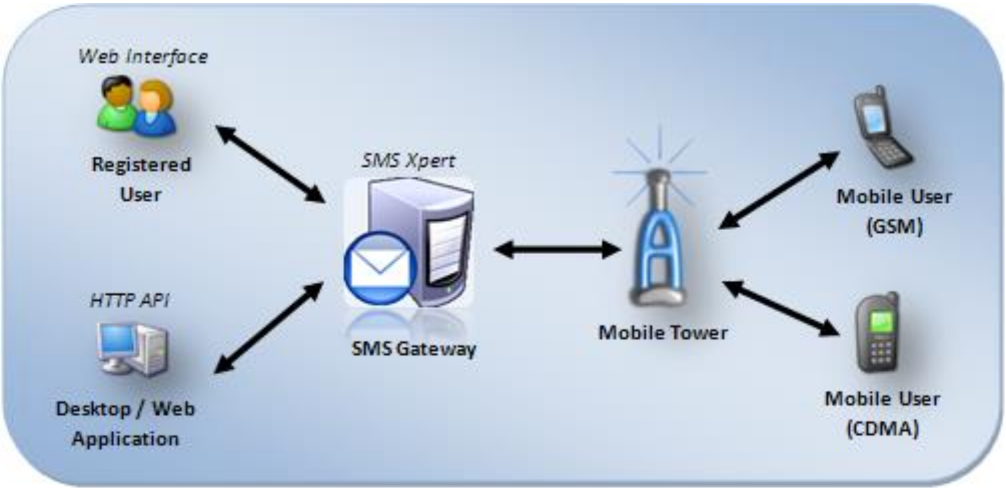


Figure 5: SMS Gateway (SMS Services, 2001)

✓ **Client-Server Software Architecture**: is modular in structure and was designed to improve flexibility, scalability, interoperability and usability. It comprises of comprise of 2-Tier, 3-Tier and N-Tier. And for the purpose of the proposed system [desktop application], 2-Tier application would be used. In this architecture, the desktop application i.e. Client, which requests for the resources, is equipped with the user interface [presentation layer] as well as the logic [business layer]. The database server provides the desktop application with the data required. For this purpose the database would be hosted on a web server. (Client Server, 2003)

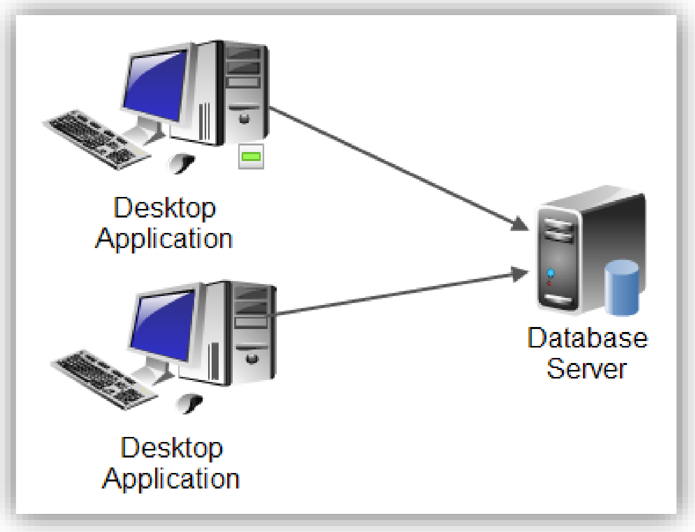


Figure 6: Client Server Architecture

For the purpose of the web application, 3-tier application would be used i.e.

- ✓ A client, i.e. the computer, which requests the resources, equipped with a user interface (usually a web browser) for presentation purposes
- ✓ The application server (also called middleware), whose task it is to provide the requested resources, but by calling on another server
- ✓ The data server, which provides the application server with the data it requires.

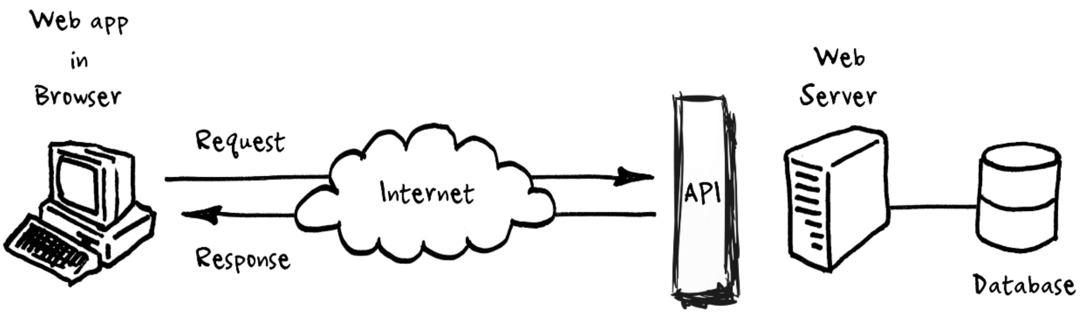


Figure 7: Client Server Architecture (FlopppyDisk, n.d.)

3.2.2 Available Resources

3.2.2.1 Literature Review for Technical Research

3.2.2.1.1 Books

- ✓ Bates, B. and Sierra, K. (2003). *Head First Java*. O'Reilly Media.
- ✓ Schildt, H. (2011). *Java The Complete Reference*. Tata McGraw-Hill Publishing Company.
- ✓ Schwartz, B. (2008). *High Performance MySQL*. 2nd ed. O'Reilly Media.
- ✓ (2012). *Tally .ERP 9 in Simple Steps*. 3rd ed. Dreamtech Press.
- ✓ Biafore, B. (2012). *QuickBooks 2013*. O'Reilly Media.
- ✓ Gulati, M. (2009). *Accounting & Tally.ERP 9*. 2nd ed. Delhi: Siliconmedia.

3.2.2.1.2 Websites

- ✓ *Java Programming.* Available: http://www.tutorialspoint.com/java/. Last accessed 10th March 2014.
- ✓ Eriksson, D. *Java Basics*. Available: http://www.avajava.com/. Last accessed 15th April 2014.
- ✓ *Learning Center*. Available: http://www.tallysolutions.com/website/html/services/learning-tallyerp9.php. Last accessed 10th April 2014.
- ✓ Averkamp, H. (2003). *Bookkeeping.* Available: http://www.accountingcoach.com/. Last accessed 05th April 2014.
- ✓ Kalabarigi, S. (2013). *Encrypt and Decrypt data.* Available: http://www.codeproject.com/Tips/635973/Encrypt-and-Decrypt-data. Last accessed 05th February 2014.

3.2.2.1.3 Journals

- ✓ Sellars, P. (2006). Java. *Reporting Made Easy with JasperReports and Hibernate*. 1, Available at: http://java.sys-con.com/node/171467.
- ✓ Hindle, A. (2008). k. *Reverse Engineering CAPTCHAs*. p59-68. Available at: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4656374

3.2.2.1.4 White Papers

- ✓ Sosnoski, D.A. (2000). Managing collections. *Java performance programming*. (3), Available at: http://www.javaworld.com/article/2076908/build-ci-sdlc/java-performance-programming--part-3--managing-collections.html.
- ✓ Farber, B. (2014). Hibernate validation in a standalone implementation. *Java validations*, Available at: http://www.javaworld.com/article/2137346/data-storage/java-tip-hibernate-validation-in-a-standalone-implementation.html.

3.2.2.2 Literature review for Software Development

- ✓ Schwalbe, K (2006). *Information Technology Project Management*. 4th ed. Minneapolis: Cours Technology.
- ✓ Valacich S Joseph, *Essentials of System Analysis and Design*, Second Edition (Prentice Hall of India), India.
- ✓ Boehm, B.W.. (2002). A spiral model of software development and enhancement. Available at: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=59&queryText%3Dspiral+model.

3.3 Market Research

<table>
<tr><th colspan="2">3.3.1 Tally.ERP 9</th></tr>
<tr><td><b>Description</b></td><td>

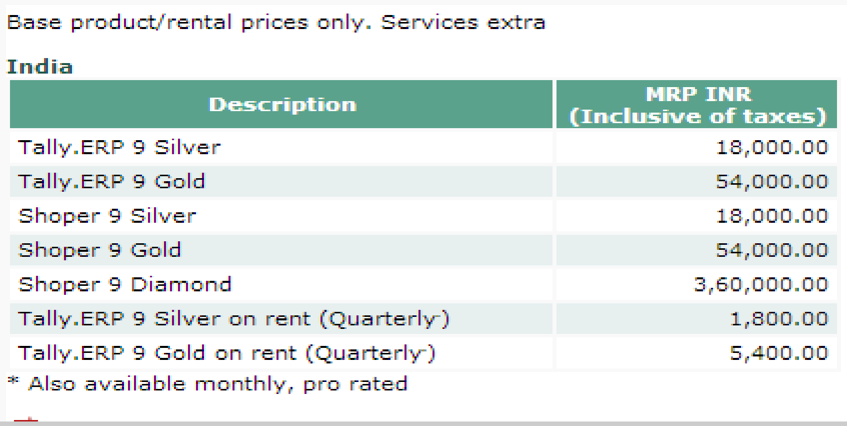
MANAGE MULTIPLE BRANCHES WITHOUT LEAVING YOUR OFFICE...
Tally.ERP 9 - Multiple Offices And Even Multiple Languages...!
Opening offices across cities and even in other countries? All you need is a reliable branch management system - Tally.ERP 9 - that helps you to see all branch transactions in one go.
With Tally.ERP 9 you manage branch offices in multiple ways viz, remote access, control centre, SMS, data synchronisation and group consolidation. Your branch offices can operate in their local language and you can consolidate in your preferred language.
Benefit from greater insights, business intelligence for improved business decisions and synergies.
Award Winning Features &amp; Functions Of Tally.ERP 9
Sales | Purchase | Inventory | Manufacturing | Costing | Payroll | Accounts | Finance | Job Costing | Point of Sales
Branch Management | Integrated Support Centre | Control Centre | Trusted Remote Access | SMS | Statutory Compliance
Buy Online @ www.tallysolutions.com   Call Your IT Reseller   ☎ 1800-425-8859</td></tr>
<tr><td><b>Pros</b></td><td>Simple and rapid installation
Internal backup/restore
Data security
Import/ Export of data
Multilingual capability</td></tr>
<tr><td><b>Cons</b></td><td>Costly
No Notifications/Reminders</td></tr>
<tr><td><b>Source</b></td><td>http://www.tallysolutions.com/website/html/tallyerp9/tallyerp9-main.php</td></tr>
<tr><td><b>Market Value</b></td><td>
Base product/rental prices only. Services extra
<b>India</b>
<table>
<tr><th>Description</th><th>MRP INR
(Inclusive of taxes)</th></tr>
<tr><td>Tally.ERP 9 Silver</td><td>18,000.00</td></tr>
<tr><td>Tally.ERP 9 Gold</td><td>54,000.00</td></tr>
<tr><td>Shoper 9 Silver</td><td>18,000.00</td></tr>
<tr><td>Shoper 9 Gold</td><td>54,000.00</td></tr>
<tr><td>Shoper 9 Diamond</td><td>3,60,000.00</td></tr>
<tr><td>Tally.ERP 9 Silver on rent (Quarterly)</td><td>1,800.00</td></tr>
<tr><td>Tally.ERP 9 Gold on rent (Quarterly)</td><td>5,400.00</td></tr>
</table>
* Also available monthly, pro rated</td></tr>
</table>

Table 13: Market research - Tally ERP 9.0

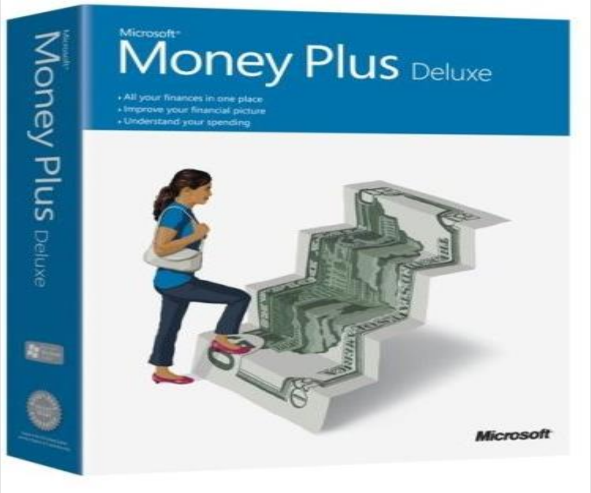
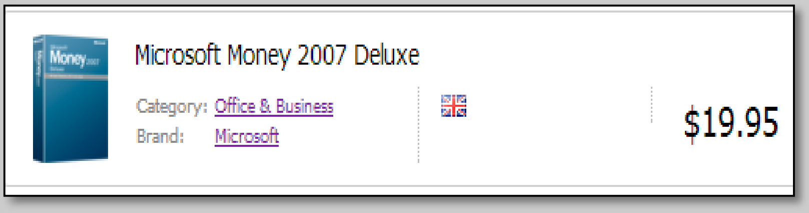
3.3.2 Microsoft Money Plus	
Description	 Microsoft Money Plus Deluxe • All your finances in one place • Improve your financial picture • Understand your spending Microsoft
Pros	Data security Import/ Export of data
Cons	Slow software Frequent Updating No Notifications/Reminders
Source	http://www.microsoft.com/en-in/download/details.aspx?id=7564
Market Value	 Microsoft Money 2007 Deluxe Category: Office & Business Brand: Microsoft $19.95

Table 14: Market research - Microsoft Money Plus

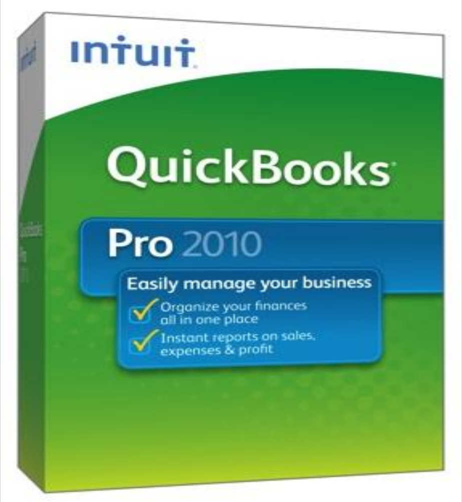
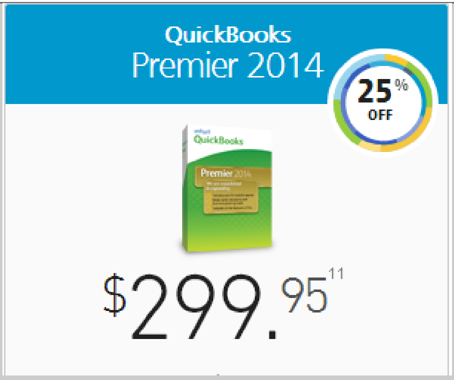
3.3.3 QuickBooks Pro	
Description	 intuit QuickBooks Pro 2010 Easily manage your business Organize your finances all in one place Instant reports on sales, expenses & profit
Pros	Real-time Cash flow Import/ Export of data High-speed Invoicing
Cons	Lacklustre Audit Trail Upgrade Fees No Fixed-Asset Section
Source	http://quickbooks.intuit.com/
Market Value	

Table 15: Market research - QuickBooks

3.3.4 Product Analysis

3.3.4.1 Qualitative Analysis

S. No.	Features	Tally ERP	Microsoft Money	QuickBooks	College Accountant
1.	Payroll Module	✓	✓	✓	✓
2.	Store Module	✓	✓	✓	✓
3.	Student Fee/Fine	-	-	-	✓
4.	Library Module	-	-	-	✓
5.	Bus Module	-	-	-	✓
6.	Hostel Module	-	-	-	✓
7.	Online Payment	-	-	✓	✓
8.	SMS Notifications	✓	-	-	✓
9.	Graphical Report Generation	✓	✓	✓	✓
10	Customizations	✓	-	✓	✓

3.3.4.2 Quantitative Analysis

S. No.	Rating [1-10]	Tally ERP	Microsoft Money	QuickBooks	College Accountant
1.	Payroll Module	8	6	8	9
2.	Store Module	7	7	8	9
3.	Student Fee/Fine	-	-	-	8
4.	Library Module	-	-	-	8.5
5.	Bus Module	-	-	-	8
6.	Hostel Module	-	-	-	8
7.	Online Payment	-	-	8	7
8.	SMS Notifications	7	-	-	9.5
9.	Graphical Report Generation	6	7	7	9
10	Customizations	6	-	5	8.5
Total value out of 100		34	20	36	***84.5***

Scale	**1-3**	**4-5**	**6-7**	**8-9**	**10**
Rating	Poor	Fair	Good	Excellent	Outstanding

3.3.5 Conclusion

From the market research, and further doing the qualitative and quantitative analysis of the similar products in the market, it has been found out that very less software are available which have comparable features with the proposed system. There were many features missing from similar systems, hence there is a dire need to develop a system like College Accountant.

3.4 Critical Evaluation of Literature Review

The literature review provides the grounds to carry out further research on different aspects dealt by the proposed system.

Advance preliminary research done in the literature review is the developer's insight of reaching to a real time problem and trying to innovate a system from scratch.

For this project, developer has done two types of research; one is to understand accounts keeping and market situations.

The other one is to do a research on the programming technologies. Also, research on how to host and implement the payment gateway in this system.

Vast majority of people will adopt the system, if it is affordable and available.

According to the research, the proposed system will be accepted by colleges/institutes if it provides convincing features at an affordable price.

On completion of Literature review, it is finalized that the developer can now proceed with the Research related to the system.

4. Research Methods

4.1 Primary Research

The Primary Research aims to gather the contextual and preamble background of the current problem (Scenario), so that the developer can start work upon the development of the system. The developer has targeted four fact finding techniques: Questionnaire, Interview, Focus Group and Naturalistic Observation.

4.1.1 Questionnaire

A Questionnaire consists a series of questions asked to users to obtain statistically useful information about a given topic. User requirements is made clear through questionnaires. The data received from questionnaire is analysed and then referred while developing the system.

Objective:

- ✓ To assist the developer's research thereby clearing requirements more clearly based upon the problem background
- ✓ To get the viewpoints of intended system user.
- ✓ To verify the nature and seriousness of the identified problems with a high response rate
- ✓ For verification of operational feasibility

The developer has designed eight (8) leading questions and three (3) open format questions, to gather data from the accountants of different colleges/universities. These questionnaires were distributed in the paper form.

A sample template of the Questionnaire with justification for each question has been provided in the Appendix part of this documentation.

Also, a filled Questionnaire with analysis has also been provided in the Appendix C.

4.1.2 Interview

Interview is a direct conversation between two or more people. The interviewer asks questions to obtain information from the interviewee. It contains both open and close ended questions.

Objective: The interviewee can clarify the questions then and there, thus leading to more satisfactory responses.

The developer has designed ten (10) questions, to gather data from the Librarians of different colleges/universities. These interviews were held face-face.

A sample template of the Interviews with justification for each question has been provided in the Appendix part of this documentation.

Also, a filled Interview with analysis has also been provided in the Appendix C.

4.1.3 Focus Group Study

A focus group is a discussion among a group of people who are related by the same topic. The group should have just sufficient members to generate rich discussions. These are structured around a set of carefully predetermined questions – usually no more than 10 – but the discussion is free-flowing.

Objective: Focus groups can reveal a wealth of detailed information and deep insight.

The developer has designed two (2) engagement questions, five (5) exploration questions and one (1) open format question for the students studying in colleges/universities.

A sample template of the Focus Group Questions with justification for each question has been provided in the Appendix part of this documentation.

Also, a Focus Group Study discussions with analysis has also been provided in the Appendix C.

4.1.4 Naturalistic Observation

In naturalistic observation, the subjects are observed by the researcher in their natural environment. The researcher doesn't intervene in the working of the subjects and silently observes them. The findings of the observation are noted down.

Objective: This is the most un-biased method of data gathering since subjects are observed in their natural environment and there is no compulsion or threat from anyone.

The developer has designed six (6) question for the Store/Inventory In-charge working in colleges/universities.

A sample template of the Naturalistic Observation question has been provided in the Appendix part of this documentation.

Also, the observations through this data gathering technique has been noted down with analysis have been provided in the Appendix C.

4.2 Academic Research

4.2.1 Technical Research

4.2.1.2 Design Principles

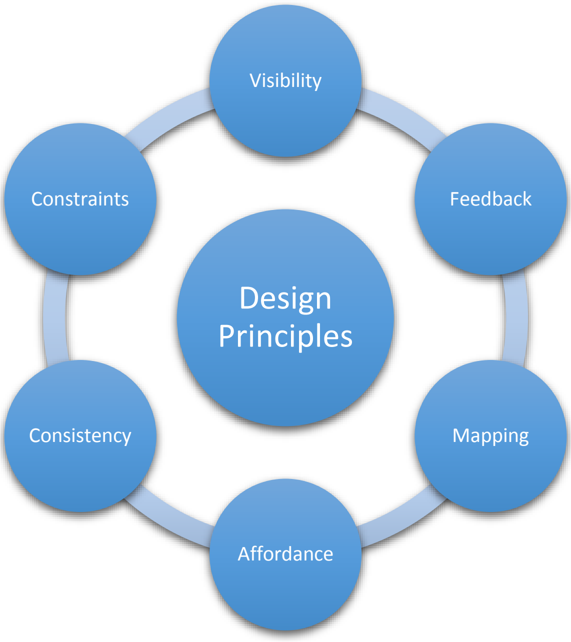


Figure 8: Design Principles

- ✓ **Visibility** – The more visible functions are, the more likely users will be able to know what to do next. In contrast, when functions are "out of sight," it makes them more difficult to find and know how to use. *Justification: The* developer will have make sure that the system uses proper schemes to make all functionalities.
- ✓ **Feedback** – Feedback is about sending back information about what action has been done and what has been accomplished, allowing the person to continue with the activity. *Justification:* The developer will make sure that the system provides with proper and immediate feedback on user input.
- ✓ **Constraints** – The design concept of constraining refers to determining ways of restricting the kind of user interaction that can take place at a given moment. *Justification:* The developer will provide constraints such as no text in fields where numbers should be there (e.g. Phone number).
- ✓ **Mapping** – This refers to the relationship between controls and their effects in the world. *Justification:* The developer needs to provide metaphors.
- ✓ **Consistency** – This refers to designing interfaces to have similar operations and use similar elements for achieving similar tasks. In particular, a consistent interface is one that follows rules, such as using the same operation to select all objects. *Justification:* The developer will provide consistent Colour, Same background, same font size and colour for headings throughout all pages.
- ✓ **Affordance** – is a term used to refer to an attribute of an object that allows people to know how to use it. *Justification:* The developer will provide roll over link wherever it is necessary **(J Preece, 2002)**

4.2.2 System Development Methodology

The proposed system – "College Accountant" integrates various modules of the college, hence is a vast and critical project. Hence, the best suited methodology for the development of this system would be the **Spiral Model.**

Spiral Model arranges the various and tasks and activities in loops. The loops represent development phases and for a project there can be any number of loops. It is an iterative model which inculcates into it the features of both the *Waterfall Model* and *Prototype Model,* hence eliminating the demerits of these models and including good features into it. The development of the software takes place in a systematic order [waterfall] over the loops and prototypes are developed and shown to end-users for evaluation [prototype] at end of a phase. This ensures complete user satisfaction and also reduces risk of project failure. This model shows special emphasis on Risk Analysis.

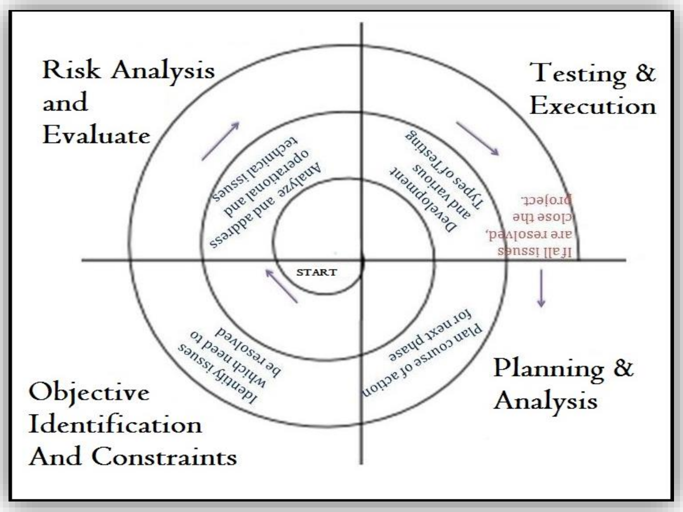

Risk Analysis and Evaluate
Testing & Execution
Analyze and address operational and technical issues
Development and various Types of Testing
If all issues are resolved, close the project.
START
Identify issues which need to be resolved
Plan course of action for next phase
Objective Identification And Constraints
Planning & Analysis

Figure 9: Spiral Model (Spiral Model, n.d.)

Phases of Spiral Model

- ✓ **Planning**: this phase deals in studying the project objectives, alternatives in design and constraints imposed because of cost, technology, schedule, etc.
- ✓ **Risk Analysis**: in this phase other approaches are studies which can be implemented in order to fulfil the identified constraints. Operational and technical issues are addressed here. Risk mitigation is in focus in this phase. And evaluation of all these factors determines future action.

- ✓ **Engineering**: in this phase the planned product is developed. Testing is also done. In order to do development, waterfall or incremental approach can be implemented.
- ✓ **Evaluation**: in this phase the progress is reviewed and judged considering all parameters. Issues which need to be resolved are identified in this phase and necessary steps are taken. **(Munassar, 2010)**

Reason for choosing Spiral Model

Developing an Accounting software is a huge task and involves lot of time and money.

- ✓ A spiral model is usually applied on large and mission-critical projects, hence best suited for this proposed system which in itself is large due to integration of many modules with time and budget constraints.
- ✓ The spiral model lays special emphasis on Risk Management. Hence there is a very high chance of reducing the risk of project failure. This is necessary since this project is vast and costly, and any failure will lead to huge losses for the organisation as well as for the developer.
- ✓ Complex and vast systems like the proposed system demands for strong documentation and approval control which are well covered by this modelling technique. The prototype of each module is reviewed by the users. Documentation helps in building the user manual which eventually reduces the load after the project completion.
- ✓ The project estimations like cost, schedule and other resources become more realistic with the progress of the project through the loops of this model.
- ✓ Since a prototype is developed after the end of each phase, it ensures maximum user satisfaction and helps in the proper management of the project. It helps accepting a wide variety of changes in the development process.
- ✓ Additional functionalities can be added at a later stage which in turn helps in the development of a highly customizable project which is the need of the hour. The design is flexible.

Reasons for not choosing other models

- ✓ RAD (Rapid Application Development) : is used to develop a system when there is a time constraint but in the proposed system there is no such problem and also to use this model very skilled and experienced developer are required.

- ✓ Incremental Model: The main reason for not choosing this model is that this model is the series of waterfall model but it is not possible to use this model because of the time constraint. This model is mainly used for very large projects where the requirements are not defined clearly

- ✓ Prototype Model: will lead to poorly designed systems and false expectations (where the customer mistakenly believes that the system is "finished" when in fact it is not). In this model, the documentation is not given any importance contrary to which the proposed system requires a lot of documentation and also designed systems and false expectations are not acceptable.

4.2.3 Programming Language Research

J2SE is the premier platform for rapidly developing and deploying secure, portable applications that run on server and desktop systems spanning most operating systems. Java is an object-oriented consists of a virtual machine, set of libraries/ packages which are needed to allow the use of file systems, networks, graphical interfaces etc. Java is platform independent; means write a program once and run anywhere. (Webdotdev, 2008).

The various advantages of Java language are:

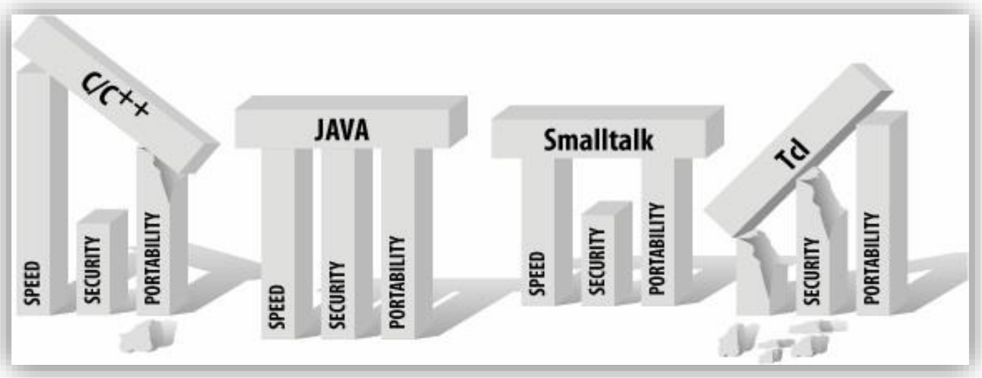


Figure 10: Language Comparison (Phipps, 1999)

- ✓ **Speed**: The execution of program in java is very fast which would be an added boon to this vast and complex accounting system.
- ✓ **Security**: Java provides various security measures which would help vastly to make this system more secure. Java language, compiler, interpreter, and runtime environment were each developed with security in mind.
- ✓ **Portability**: as a programming language Java provides the simplest and most familiar form of portability -- source code portability. A given Java program would produce identical results regardless of the underlying CPU, operating system, or Java compiler. Hence the system developed in Java can be run on a computer of any configuration such as such as Windows 95 and NT, Macintosh, Linux and UNIX.
- ✓ **Robustness**: Java is a robust language wherein the errors are detected in early stages which helps in seamless development of the system.

- ✓ **Distributed:** Java makes distributed computing easy with seamless network capabilities which are already built into it. This is important for this project since the desktop and web part of this project share their database on the network. **(Webdotdev, 2008)**

Java is a major programming language in the market now days too.

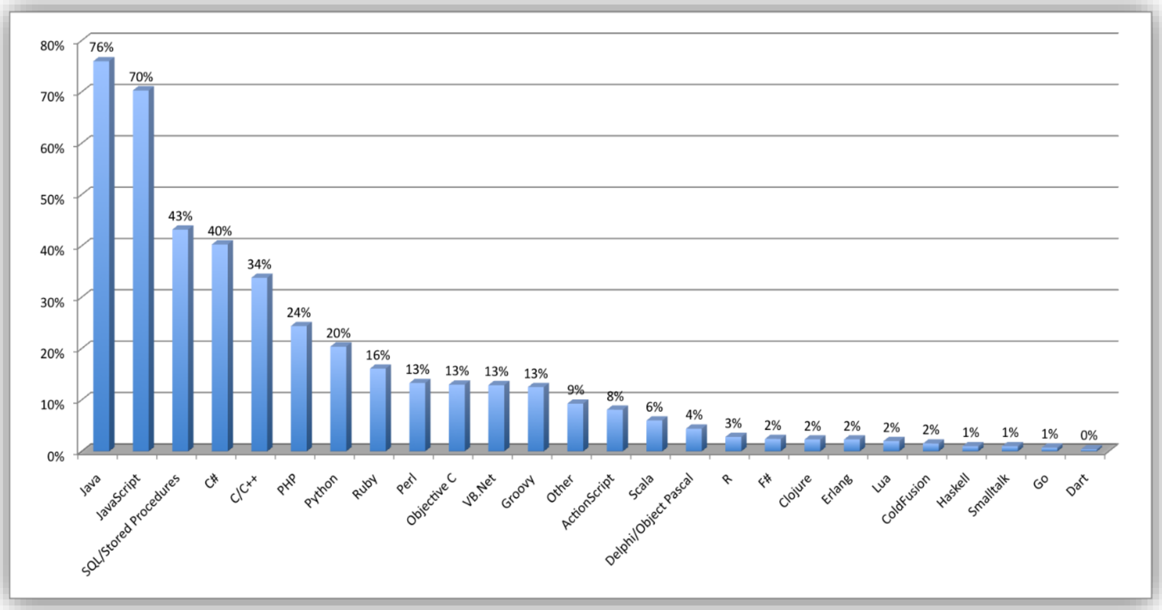


Figure 11: Java Market Share (Bharti, 2012)

4.2.4 Database Management System Research

The main reason for this database research is to get to be acquainted with the best suitable database for the proposed system so that the tribulations concerning to the information that is being stored to perform the better level of contentment and to grant the security in satisfactory level. Developer has conducted research on Microsoft SQL Server, Oracle and MySQL so as to choose the best suitable database management system for the proposed application.

Figure 12: Database Comparison

MySQL

MySQL is the most popular ***Open Source*** SQL database management system, which is developed, distributed and supported by MySQL AB. It is a second generation open source company that unites open source values and methodology with a successful business model. It ensures that transactions comply with the ACID Model, allows the building of indexes, supports standard data types, and allows for database replication, among other features.

Reason for choosing MySQL:

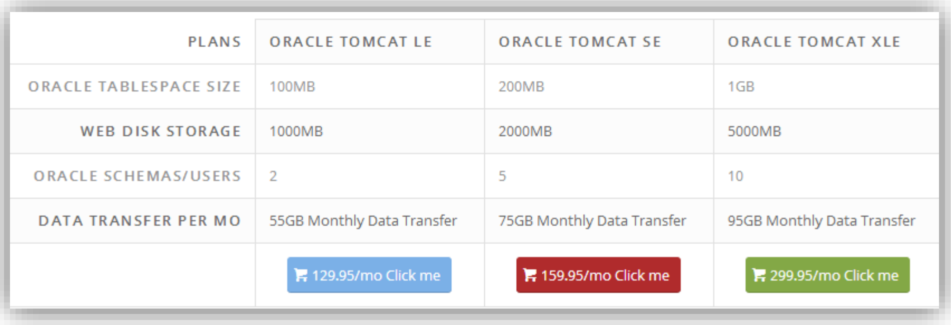
- ✓ **Minimum System Requirements** – MySQL is the database that can run in minimum requirement of configuration.
    - o Windows 2000 or later version
    - o Minimum 512 MB RAM
    - o If table size increases run the NTFS file system of MySQL
- ✓ **Open Source** – MySQL is easily available in the market. Moreover licensing and renewal of MySQL can be generally done very easily and quickly since it's an open source software.
- ✓ **Cost saving** – It is freely available as there is no cost of purchasing. Due to the budget of the organization MySQL seems to be best. If the goal can be achieved in minimum cost then why to use other expensive database!
- ✓ **Backup and Restore facility** – MySQL automatically provides the facility of backup from one MySQL system to another MySQL system. Data for any organization is much important than other things. In case of data loss, to run an organization will become impossible. As it provides the facility to take back up to another system can be advantageous for the organization.
- ✓ **SQL Query IDE** – Query Editor can be used to test the results of any query, even more it also provides the facility to generate the query by using SQL Query editor.
- ✓ **Graphical Interface** – It also provides the graphical user interface for creating tables and running queries. NaviCat is being used to access MySQL 5.0 database.

- ✓ **Compatibility with programming language:** As it is owned by Oracle Corporation, it has best compatibility with JAVA.
- ✓ Lower maintenance cost and easy to manage.
- ✓ Offers corruption detection techniques.

The primary reason for choosing MySQL over other databases was the cheap cost of JSP hosting with this database. The system demands for the database to be hosted, hence the developer has no option other than to host it. If database had been Oracle or MS SQL, dedicated server needs to be purchased which works out to be very expensive.

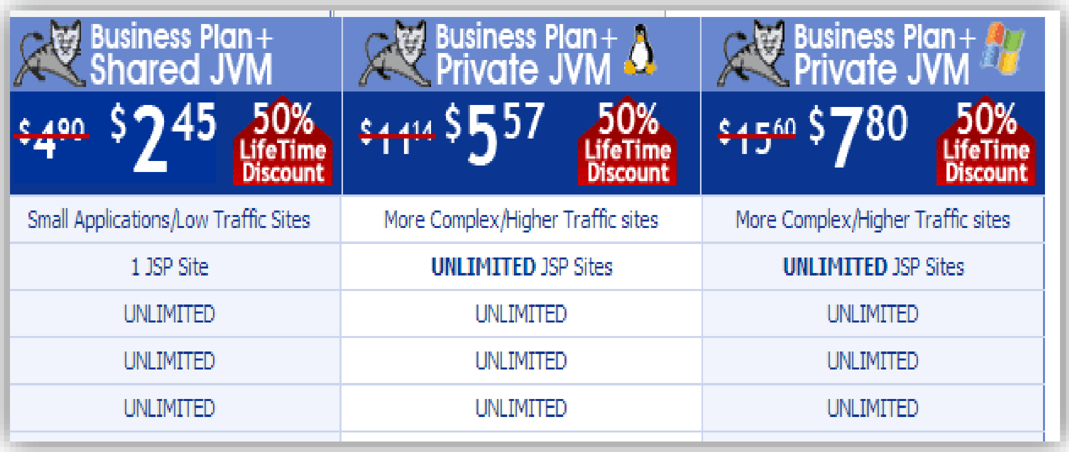
Price Comparisons [prices in $]:

Oracle Hosting Price:



PLANS	ORACLE TOMCAT LE	ORACLE TOMCAT SE	ORACLE TOMCAT XLE
ORACLE TABLESPACE SIZE	100MB	200MB	1GB
WEB DISK STORAGE	1000MB	2000MB	5000MB
ORACLE SCHEMAS/USERS	2	5	10
DATA TRANSFER PER MO	55GB Monthly Data Transfer	75GB Monthly Data Transfer	95GB Monthly Data Transfer
	129.95/mo Click me	159.95/mo Click me	299.95/mo Click me

MySQL Hosting Price:



Business Plan+ Shared JVM	Business Plan+ Private JVM	Business Plan+ Private JVM
~~$4 90~~ $2 45 50% LifeTime Discount	~~$11 14~~ $5 57 50% LifeTime Discount	~~$15 60~~ $7 80 50% LifeTime Discount
Small Applications/Low Traffic Sites	More Complex/Higher Traffic sites	More Complex/Higher Traffic sites
1 JSP Site	**UNLIMITED** JSP Sites	**UNLIMITED** JSP Sites
UNLIMITED	UNLIMITED	UNLIMITED
UNLIMITED	UNLIMITED	UNLIMITED
UNLIMITED	UNLIMITED	UNLIMITED

Why other database weren't chosen?

- ✓ *Microsoft Access*: is an information management tool that helps you store information for reference, reporting, and analysis. However, it is not object-oriented and is much slower in comparison to other database management systems. Query executions takes a longer time if multiple users are accessing the database at the same time. In the proposed system, there could be a case where many students have logged into the system and are performing some transactions, in such case the database needs to respond quickly which may be difficult if we use Access.
- ✓ *Microsoft SQL Server*: is a relational model database server created by Microsoft. It strictly follows client/server architecture wherein the Database resides on a central computer (server) and is used by clients. However, for using MSSQL, the developer needs to purchase its license which incurs additional cost to the

software. Also, Java is being used for the front-end, and using MSSQL may add to some complexity in the development of the system.

- ✓ *Oracle*: is a fourth generation relational database management system which is able to reliably manage a large amount of data in a multi-user environment so that many users can concurrently access the same data. However, for using Oracle, the developer needs to purchase its license which incurs additional cost to the software. And also, the development time is more while using this database.

Conclusion

MySQL is a sound database having all the functionalities which are required for developing this system. It was preferred over other databases because both MSSQL and Oracle are not freely available and needs to be purchased incurring additional cost to the software. Also, in comparison to other databases, the development time of MySQL is better. MySQL is more preferable because the proposed system front-end is in Java and as both are open source and freely available so it’s easy to integrate and both can be made easily available on multiple platform. Moreover, hosting of databases like Oracle and MS-SQL requires dedicated servers which turns out to be expensive incurring additional costs to the company.

4.3 Conclusion

In thus chapter, ‘Research Methods’, Primary and Academic research were done and the developer has explained these research works in detail and summarized the study.

5. Analysis

5.1 Questionnaire

Instructions:

This questionnaire has been prepared for the Accountants working at a college/ university. It comprises of four sections:

Section A: Personal Information.

Section B: Domain Specific Information.

Section C: Feel free to share your views/ suggestions.

Section A: Personal Details	
Designation	
Experience	
Email	

Section B: [Tick one]

B.1 For how long have you been working in this college/university?

- □ 0-5 years
- □ 5-10 years
- □ More than 10 years

Justification: This question will help decide how well the accountant knows about the various operations and procedures which are being followed in the college/university they are working in.

B.2 How is accounts book-keeping done in this college?

- □ Manually : records kept in hard-copy registers
- □ Computerized : sophisticated accounting software

Justification: This question helps the researcher understand the current working scenario for accounts book-keeping in the college.

B.3 If you answered computerized in previous question, then which accounting software is being used currently?

- □ TALLY
- □ Microsoft Office Accounting
- □ Microsoft Money
- □ Quickbooks
- □ Others, please specify ________________________

Justification: This question would help the researcher to gain knowledge about the software mentioned above, and try to improve it with additional features for this organization.

B.4 What is the price range of the software mentioned above?

- □ ₹ 1000 - ₹ 2000
- □ ₹ 2000 - ₹ 5000
- □ ₹ 5000 - ₹ 8000
- □ ₹ 8000 - ₹ 10000
- □ More than ₹ 10000
- □ Free of Cost

Justification: This question will help decide the cost for the proposed software.

B.5 Does this software require any renewal?

- □ Yes
- □ No

Justification: This question helps decide the total amount being spent on the current software.

B.6 If yes, how often?

- □ Monthly
- □ Quarterly
- □ Half-yearly
- □ Yearly

Amount: ₹ _______________

Justification: This question helps decide the total amount being spent on the current software

B.4 For how long have you been using the computer system?

- □ 1-2 years
- □ 2-5 years
- □ 5-10 years

Justification: This will help decide what kind of training needs to impart to make the users [accountant] comfortable with the proposed system.

B.5 How do you rate your Computer knowledge? [Rating: 1-low 5-high]

- □ 1
- □ 2
- □ 3
- □ 4
- □ 5

Justification: This question will help the developer to decide how much user-friendly the system has to be along with the kind of training to be imparted on users to make them sync with the proposed system.

Section C: Fill in detail

C.1 Please specify the advantages of the current system being used in the college for accounting.

____

_____---

Justification: This questions help decide the basic features which need to be included in the proposed system.

C.2 Please specify the dis-advantages of the current system being used in the college for accounting.

____
_____---

Justification: This questions help decide the features which need to be given in order to overcome these problems.

C.3 Please specify the features which you would like to have in the system being developed for you?

____
_____---

Justification: This questions help decide the enhanced features which could be added in the proposed system to make it more desirable to the users.

Thanks for your participation!!!

Please check if you have answered all questions. It is not mandatory to answer all questions. After completion please re-send @ dilip_dmk@yahoo.com

5.2 Interview

Instructions:

These interview questions has been prepared for the Librarian employed at a college/university. It comprises of four sections:

Section A: Personal Information.

Section B: Domain Specific Information.

Section C: Feel free to share your views/ suggestions.

Section A: Personal Details

Designation	
Experience	
Email	

Section B: [Answer in detail]

B.1 For how long have you been working in this college?

____

Justification: This question would help the researcher to judge the extent to which the responses given by the person is accurate.

B.2 For how long have you been working on computer?

____

Justification: This question would help the researcher decide how user friendly the proposed system should be along with the training which need to be imparted to the users.

B.3 How is the ordering of books/magazines done?

____

____

Justification: This question would help the researcher to understand the whole process of books/magazines etc. ordering being done in this college/university.

B.4 How is the fine for late submission of book calculated?

____

____

Justification: This question would help the researcher understand the fine collection process for books which are submitted late

B.5 How is the fine for damaged. /lost books calculated?

____

____

Justification: This question would help the researcher understand the fine collection process for books which were lost/damaged.

B.6 How often is the list of defaulters generated?

____

Justification: This question would help the researcher to know about the frequency of defaulter list generation.

B.7 Explain the process for paying the fines?

____

____

Justification: This question would help the researcher understand the fine payment process for books which are submitted late

B.8 How often is the system updated to show the updates in regard to fine collection?

____

____

Justification: This question would help the researcher understand the time taken for the current system to reflect the updates

Section C [Optional]

C.1 Any complaints about the current system function?

____

____

Justification: This question would help the researcher understand the problems which are present in the current system. This would provide the basic functions for the proposed system.

C.2 Any features which you would like to see in the proposed system?

____

____

Justification: This question would help the researcher decide the various other additional feature which could be added to proposed system.

5.3 Focus Group

Instructions:

This focus group study has been prepared for the Students studying at a college/university. It comprises of four sections:

Section A: Engagement Question

Section B: Exploration Question

Section C: Feel free to share your views/ suggestions.

Section A

A.1 Do you pay your fees yearly or half-yearly? Any particular reasons?

Justification: This question helps the researcher understand the regularity with which the students pay their fees and particular reasons for them.

A.2 What is your mode of payment of college fees / fines?

Justification: This question helps the researcher understand the current mode of payment of fees to college/university.

Section B

B.1 How many of us pay our fees/fines on time? If not, reason.

Justification: This question helps the researcher understand the percentage of students who are not able to pay their fees/fines on time with reasons.

B.2 What are the pros and cons of the current fee/fine payment process?

Justification: This question helps the researcher understand the advantages and disadvantages of the current system processes.

B.3 Do you feel you are provided with proper and timely notifications regarding fees/ fine payments?

Justification: This question helps the researcher understand if the suggested feature would be preferred by the user's viz. students in this case.

B.4 Share your views on the 'Accounts department' of our college.

Justification: This question helps the researcher understand the views of the students about the accounts department of their college/university

B.5 Will providing the feature of 'Net-banking' be of any benefit to the students?

Justification: This question helps the researcher understand if the suggested feature would be preferred by the user's viz. students in this case.

Section C

C.1 Any suggestions or features which would make the proposed system more desirable?

Justification: This question helps the researcher understand the various additional features which can be added to the proposed system.

5.4 Naturalistic Observation

Naturalistic Observation would be conducted for the Store/Inventory In-charge working at a college/university. They would be silently observed in their natural working environment.

The data which is intended to be collected through this technique would be:

- The method of inserting new items into records
- The method of updating the records [on deletion/consumption]
- Time taken for insertion/updating
- Accuracy of data in records
- Method of calculating weekly/monthly/yearly expenses
- Method of ordering new items

The researcher may also observe something different, apart from the above questions, which could help in making the system more user friendly would be also noted down.

5.5 Analysis

Refer Appendix C for detailed analysis of Questionnaire, Interviews, Focus Group and Naturalistic Observation.

5.6 Conclusion

After collecting the data from the various data gathering methods used such as: Questionnaire, Interviews, Focus Group and Observation.

It was found out that the features like online payment and SMS/email notifications were appreciated by the users (students).

The librarians wanted a system which was fast and did fine calculation automatically and updates the system after the fine was paid. Also, request for ordering of books/magazines could be done online rather than physically meeting up the accountants for approval.

The accountants wanted a system which reduced their work load and reduces data discrepancies. They wanted a system which could create graphical reports instantaneously and reduced manual laborious work.

The research findings have been summarized, analysed and a final conclusion has been drawn on every aspect of the research work.

6. System Design

6.1 Design Methodology

To successfully map the requirements into an actual system, a proper design methodology needs to be adopted. It helps in:

- ✓ Reducing the gap and inconsistency in the requirements
- ✓ Capturing the details about the system for requirements/analysis
- ✓ Determining the sequence in which the features need to be developed.
- ✓ Reducing the effort required to respond to requirement changes.

(UML 2.0 in a Nutshell, 2005)

Moreover there are many Design methodologies to choose from, but here we are going to discuss about the two design models which are most suitable for the proposed system:

Object-oriented	Process-oriented
Behaviour is distributed across various object types	Behaviour is distributed across chains of logically related tasks
Behaviour captured in terms of object types and associations, class diagrams, activity diagrams, sequence diagrams etc.	Behaviour captured in terms of tasks flow dependencies, resources, etc.
Uses UML notations which contains use case diagram, use case specification, activity diagram, class diagram state diagrams	Uses designs such as Data flow diagram

Table 16: Design approaches

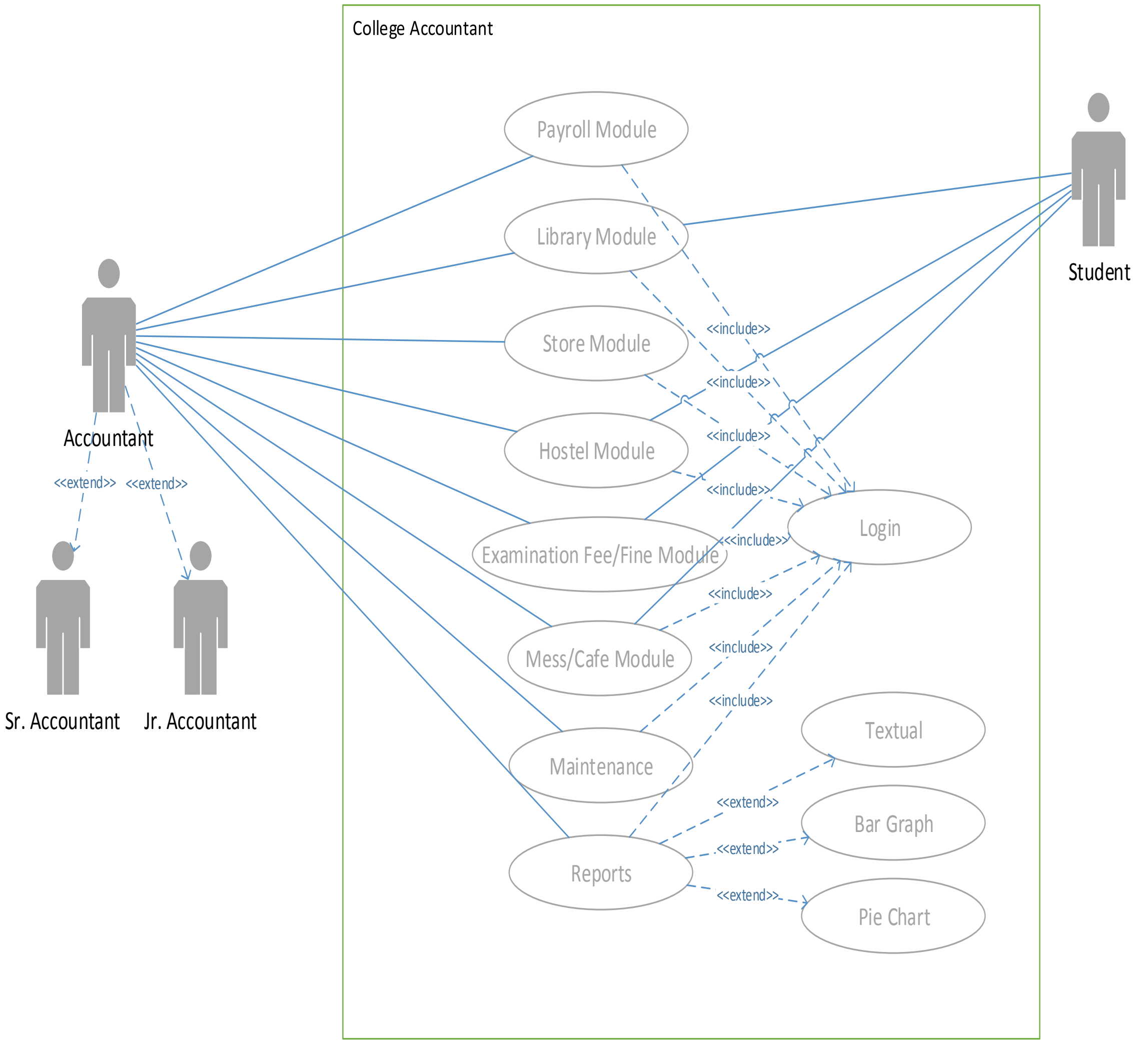
\

Object-oriented approach has been selected for modelling the proposed system due to the following reasons:

- ✓ It uses activity diagram which helps the developer analyse and determine activities which will be performed at both sides: end user as well as of the system. Process-oriented approach focuses only on the abstract model of the system.
- ✓ It uses class diagram which can be directly implemented in an object-oriented programming language like Java, C++. The proposed system is being developed in Java, hence it would be easier as well as less time consuming to model the diagrams into actual implementation. Process-oriented approach doesn't mention about any such diagrams.
- ✓ It clearly defines the actual object and its behaviour which aids the developer to identify the objects easily and early, and accordingly design the system.

6.2 Use Case Diagram

6.2.1 Use Case – Context



Actors and Description

Actor Name	Description
Accountants	Accountants are the main users of the system who enjoys maximum privilege
Students	Students also contribute too many modules of the system.
Admin	Admin is responsible for the upkeep of the system.

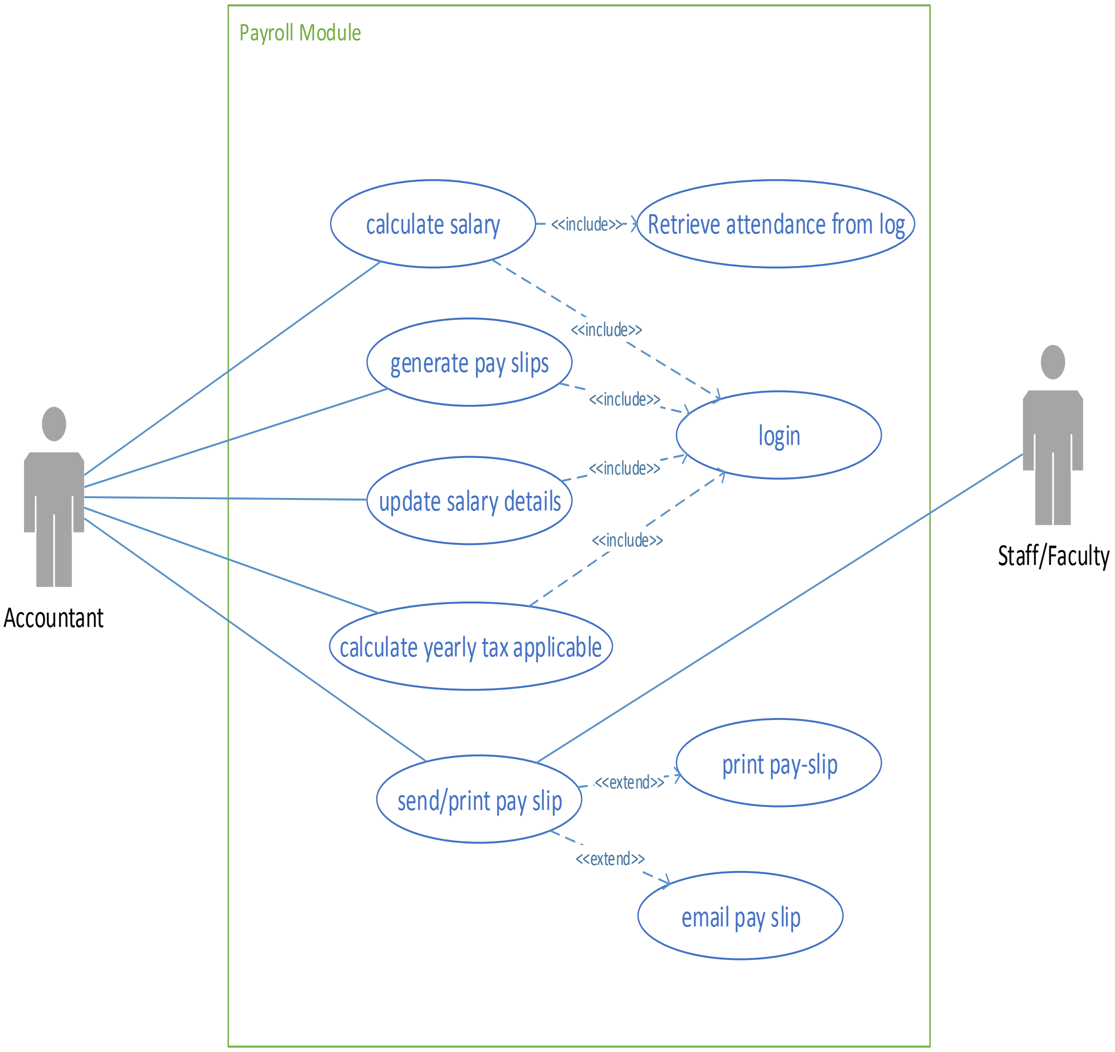
Table 17: Use Case - Actors & Description

Use-case Specification

Name	**Login**
Id	UC01
Description	This use case describes the process of signing into the system with the provided username, password and captcha code.
Actors	Accountant, Students
Goal	To get successful login to the system.
Pre-Condition	✓ The user should be registered with the system and account must be active.
Assumption	✓ None at this time.
Post-Condition	✓ Once the user is logged, they are shown a number of modules which can be accessed by them.
Happy Pathway	User successfully logins to system Operates the system without any issues Logs out of the system after finishing work.
Alternate Pathway	-
Exception Pathway	✓ User is not able to login to the system. ✓ User provides correct user name and password, but failed to log-in due to system network security

Table 18: Use Case: Login

6.2.2 Use Case – Payroll

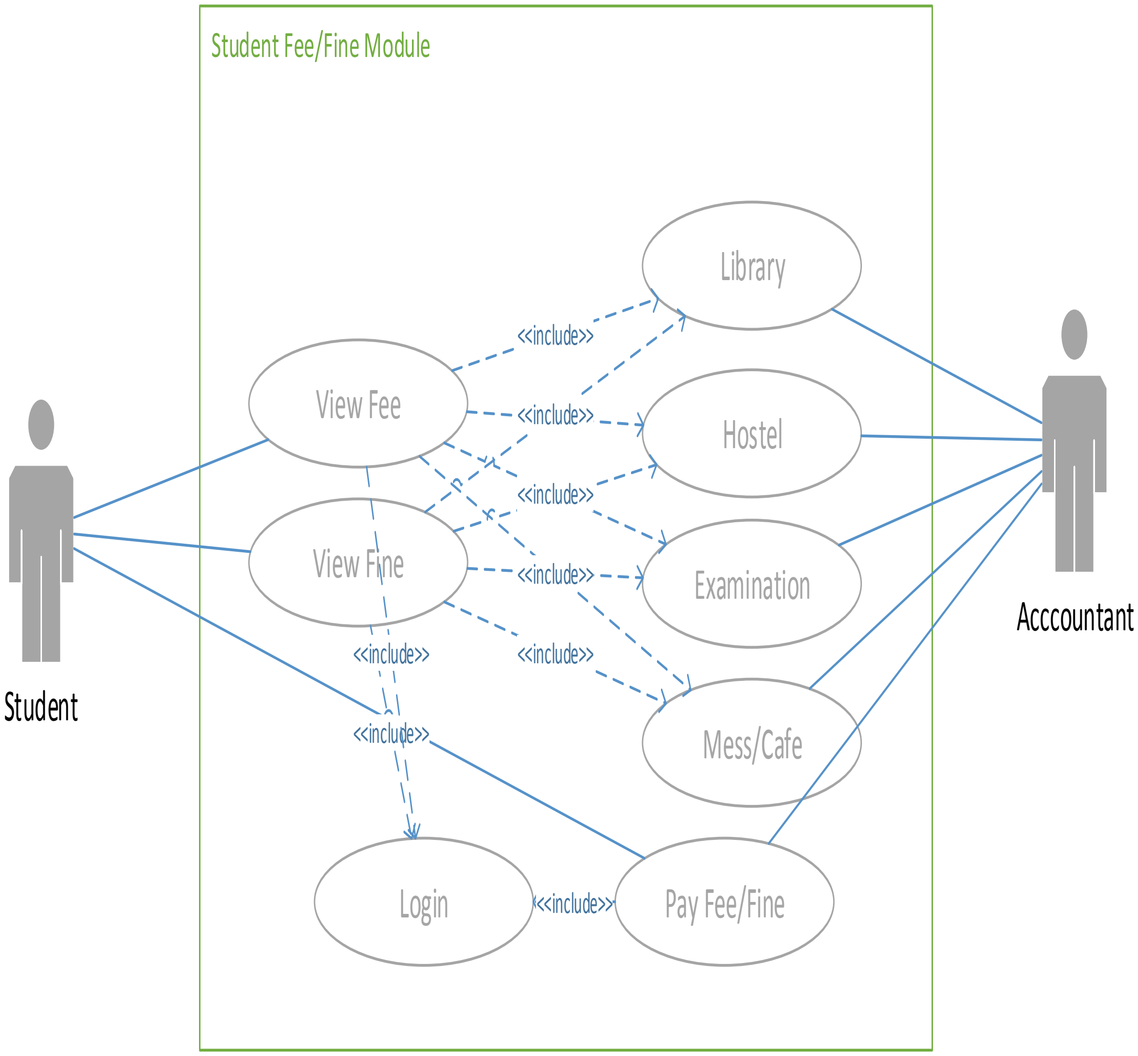


Use-case Specification

Name	**Payroll**
Id	UC02
Description	This use case describes the process of payroll calculation
Actors	Accountant, Staff
Goal	To successfully calculate payroll with the help of attendance logs of the staff/faculties and pre-set salary parameters
Pre-Condition	✓ The user should be able to successfully login into the system and should be able to fetch the attendance logs of the staff.
Assumption	✓ Attendance log is already present in the present
Post-Condition	✓ Once the user is logged, they are shown a number of options which can be accessed by them. They can update the basic salary for a staff. Or they can calculate the salary for a staff. Email/print the pay slip for the employees.
Happy Pathway	User successfully calculates the payroll for the staff and is able to generate the pay slip and send it through mail or print it.
Alternate Pathway	The user can either email the pay-slip or directly print the pay-slip.
Exception Pathway	✓ User is not able to login to the system. ✓ User provides correct user name and password, but failed to log-in due to system network security ✓ Salary calculation was

Table 19: Use Case - Payroll

6.2.3 Use Case – Student Module

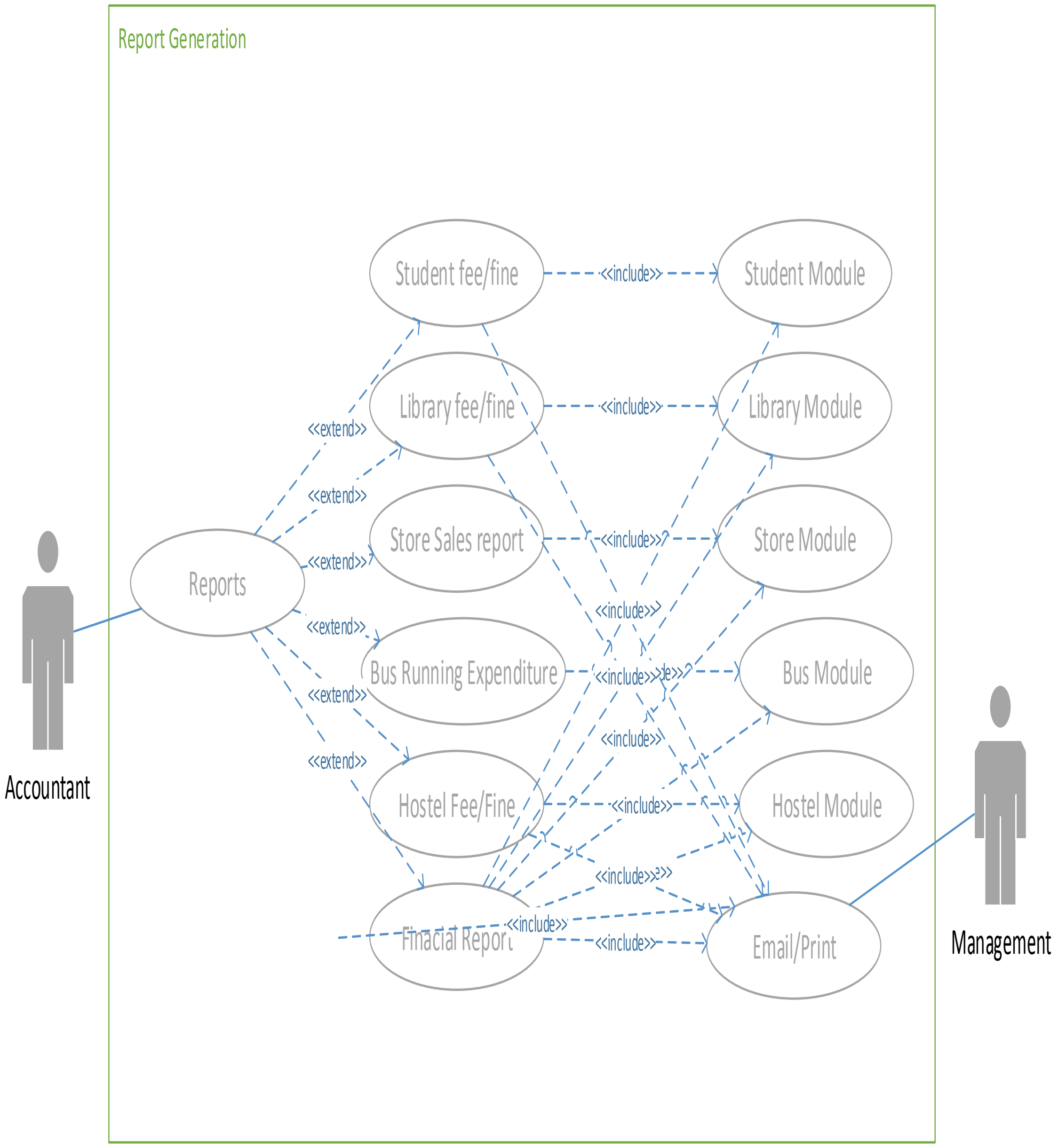


Use-case Specification

Name	Student
Id	UC03
Description	This use case describes the process of student fee/fine module
Actors	Accountant, Student
Goal	To successfully view the fee and fine and student should also be able to pay the fee and fines applicable.
Pre-Condition	✓ The user should be able to successfully login into the system and should be able to fetch the attendance logs of the staff.
Assumption	✓ Students are already registered to the system.
Post-Condition	✓ Once the user is logged, they are shown a number of options which can be accessed by them. Students can view the various fees and fines applicable to them. And have an option to pay the fee or fine which they wish too.
Happy Pathway	The user was able to view the fees and fines applicable. And the user was also able to pay the fee and fines and the same was updated to the database.
Alternate Pathway	The user can either view the fees and fines. Or can pay for the fees and fines.
Exception Pathway	✓ User is not able to login to the system. ✓ User provides correct user name and password, but failed to log-in due to authentication error and was blocked out of the system.

Table 20: Use Case: Student Module

6.2.4 Use Case – Reports

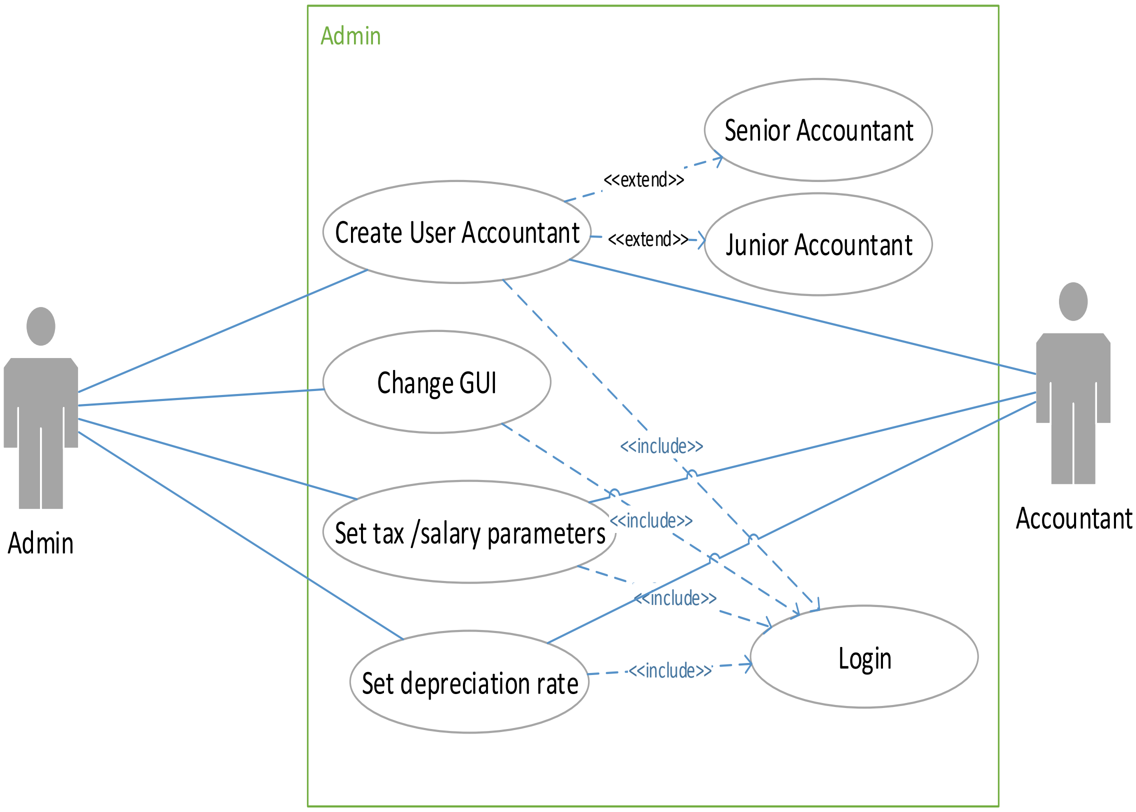


Use-case Specification

Name	Reports
Id	UC04
Description	This use case describes the process of report generation
Actors	Accountant, Management
Goal	To successfully generate the reports with the data already available in the system
Pre-Condition	✓ The user should be able to successfully login into the system and should be able to generate the reports of graphical/textual/bar graph types.
Assumption	✓ Accountants are already registered to the system. ✓ Sufficient data is already present in the system to generate the reports
Post-Condition	✓ Once the user is logged, they are shown a number of options which can be accessed by them. They have an option to select the reports of the modules which they want like library, store, defaulters etc.
Happy Pathway	The user was able to generate various kinds of reports
Alternate Pathway	The user can either generate graphical or the textual report User has an option to generate reports for various modules
Exception Pathway	✓ User is not able to login to the system. ✓ User provides correct user name and password, but failed to log-in due to authentication error and was blocked out of the system. ✓ Report was not generated since sufficient data was not available

Table 21: Use Case - Reports

6.2.5 Use Case – Admin



Use-case Specification

Name	Admin
Id	UC05
Description	This use case describes the operations which can be performed by Admin
Actors	Accountant, Admin
Goal	To successfully handle operations of the admin
Pre-Condition	✓ The user should be able to successfully login into the system and should be able to create user accounts, set various parameters etc.
Assumption	✓ Admin is already registered to the system.
Post-Condition	✓ Once the user is logged, they are shown a number of options which can be accessed by them. They have an option to create user account, set tax slabs/depreciation rates, and change GUI etc.
Happy Pathway	The user was able to login into the system and do GUI changes, create user account, set depreciation rates/tax slabs
Alternate Pathway	The user has the option to perform the various operations such as GUI changes, create user account, set depreciation rates/tax slabs
Exception Pathway	✓ User is not able to login to the system. ✓ User provides correct user name and password, but failed to log-in due to authentication error and was blocked out of the system.

Table 22: Use Case - Admin

6.3 Class Diagram

The developer has designed the class diagram to describe the structure of this system by showing the system classes. Class's interrelationships are shown including inheritance, aggregation and association. The operations and attributes are also shown to make it clearer and easier to understand.

Please Turn Over

6.4 Activity Diagram

Login

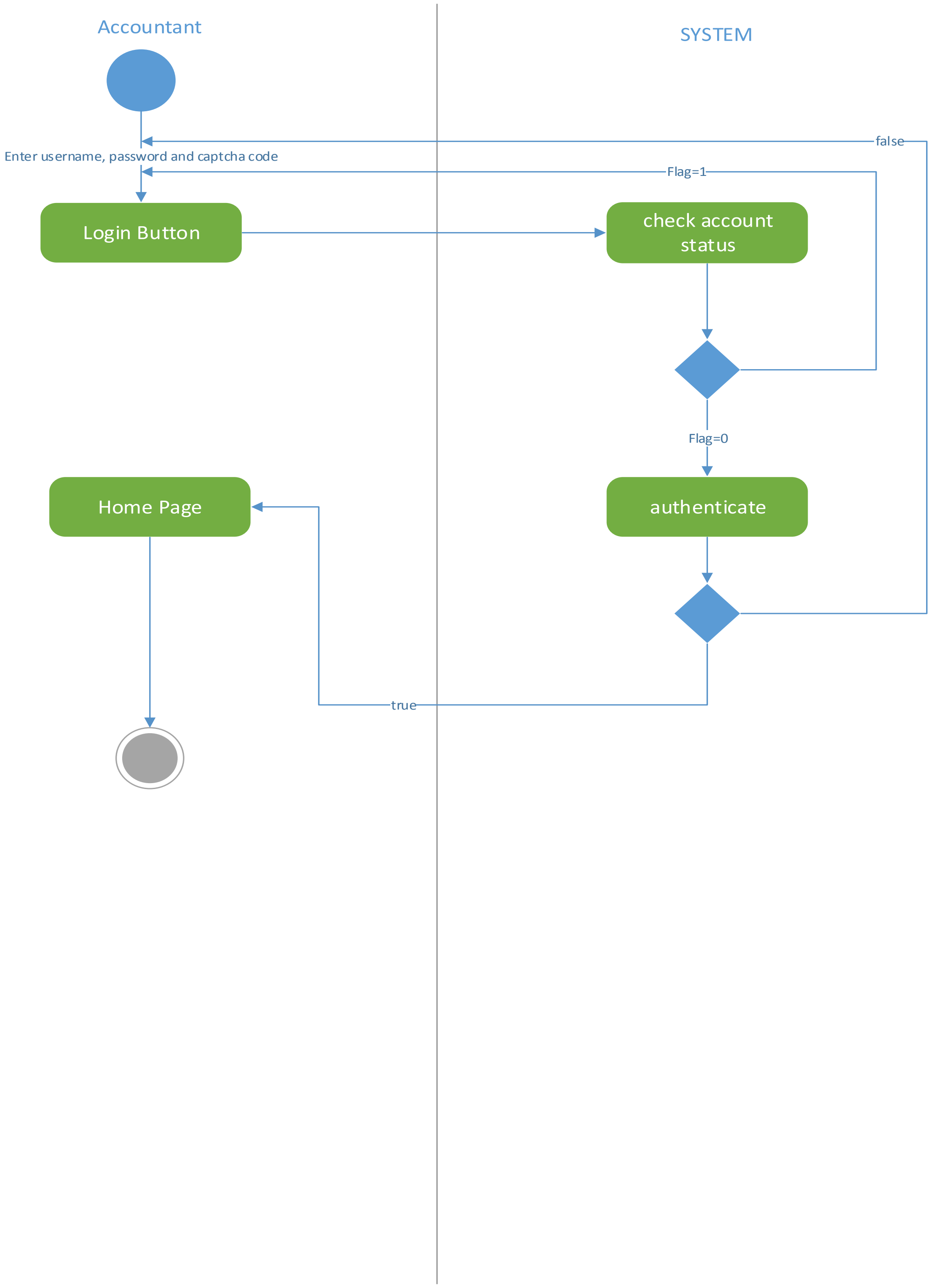


Figure 13: Activity Diagram - Login

Payroll

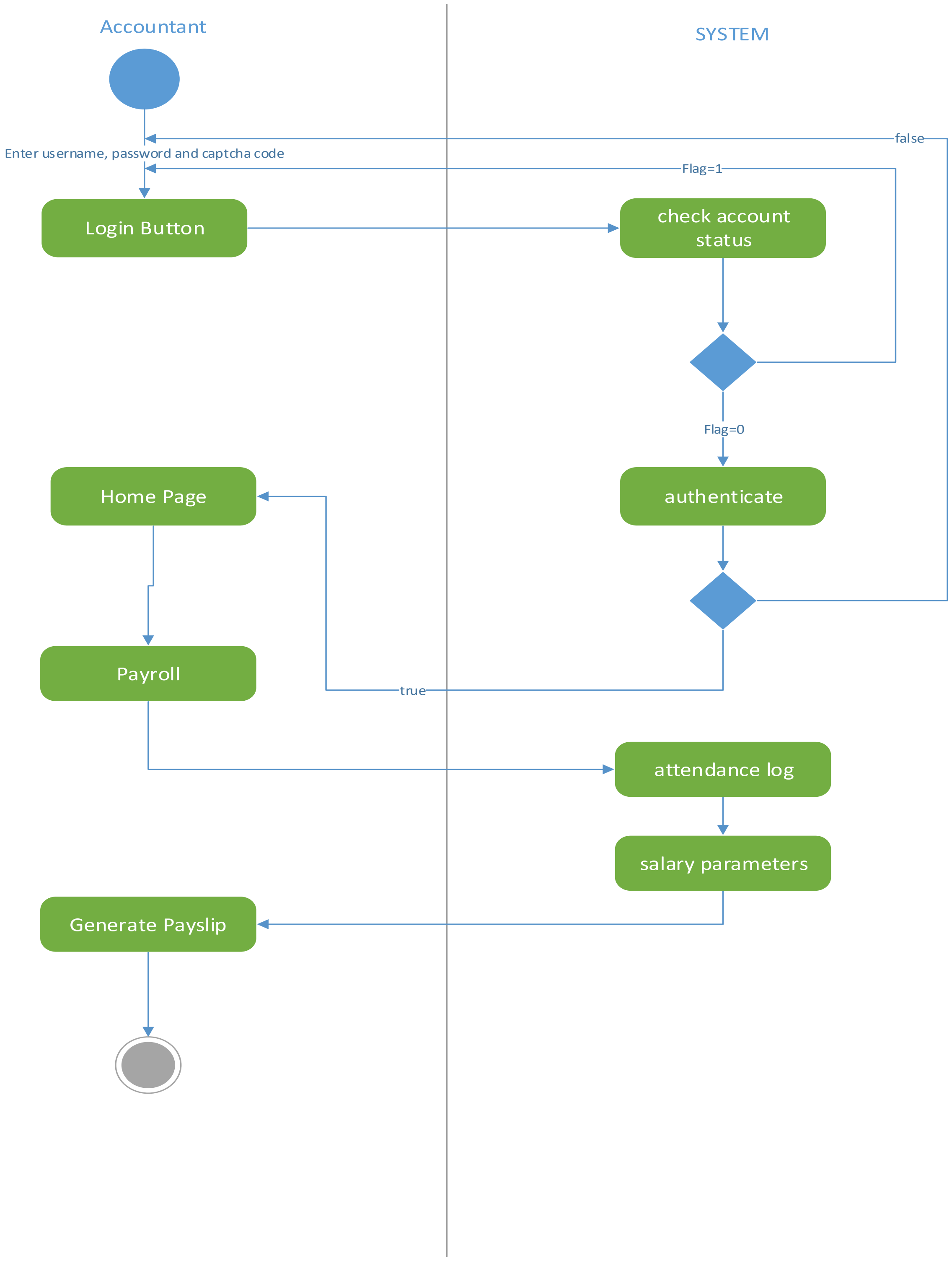


Figure 14: Activity diagram: Payroll

Student Payment Gateway

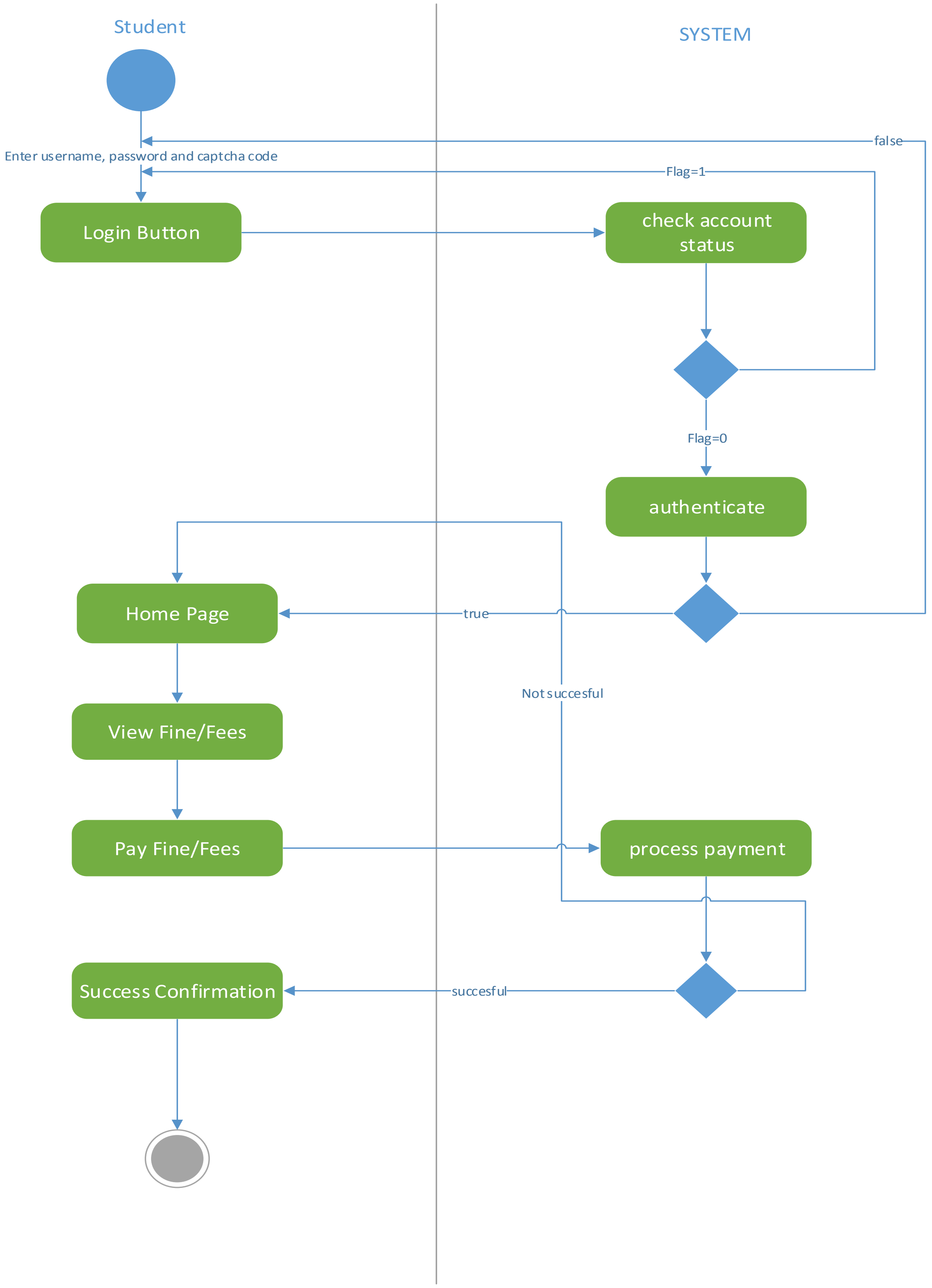


Figure 15: Activity diagram: Payment

Reports

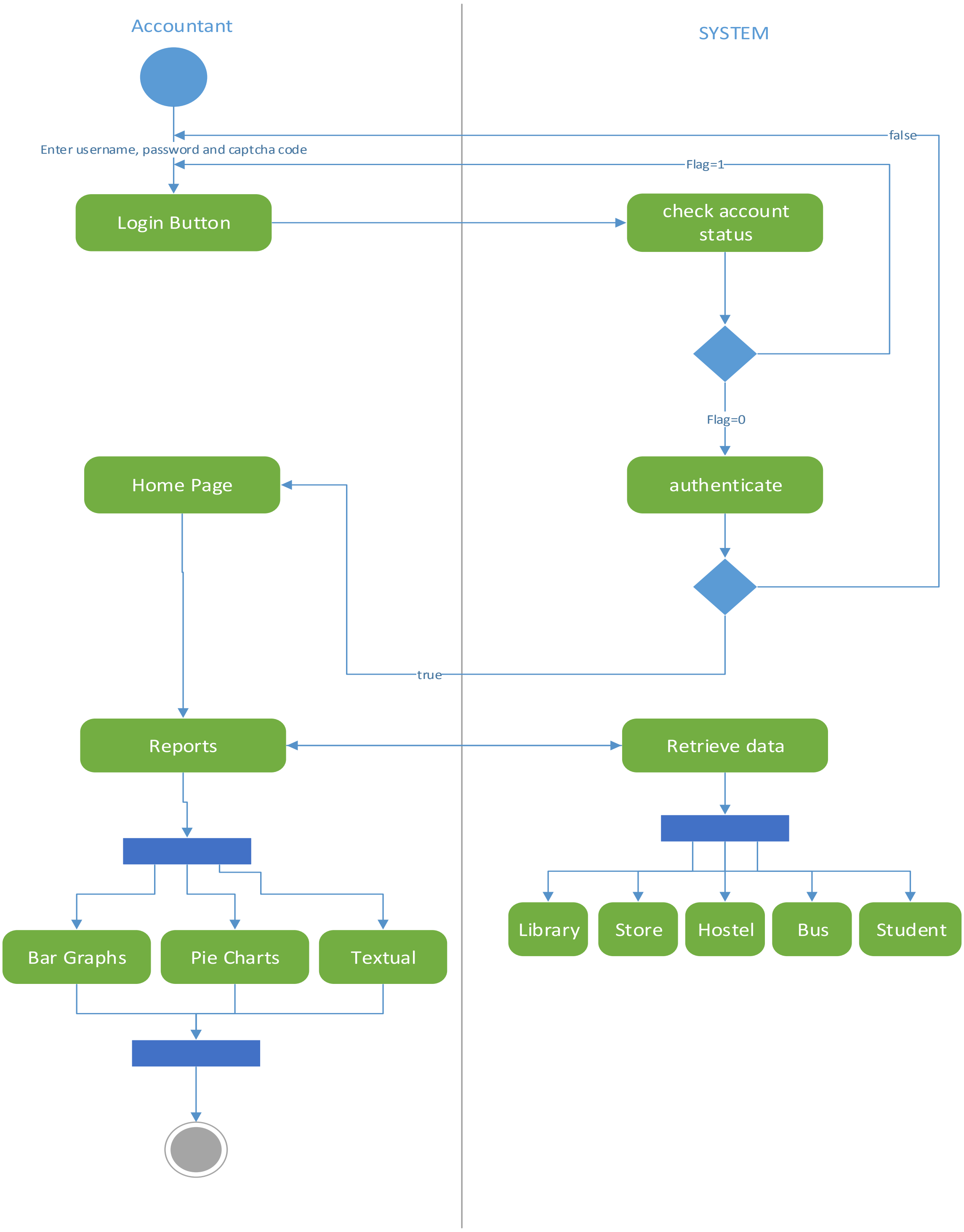


Figure 16: Activity diagram - Reports

6.5 Sequence Diagram

Context

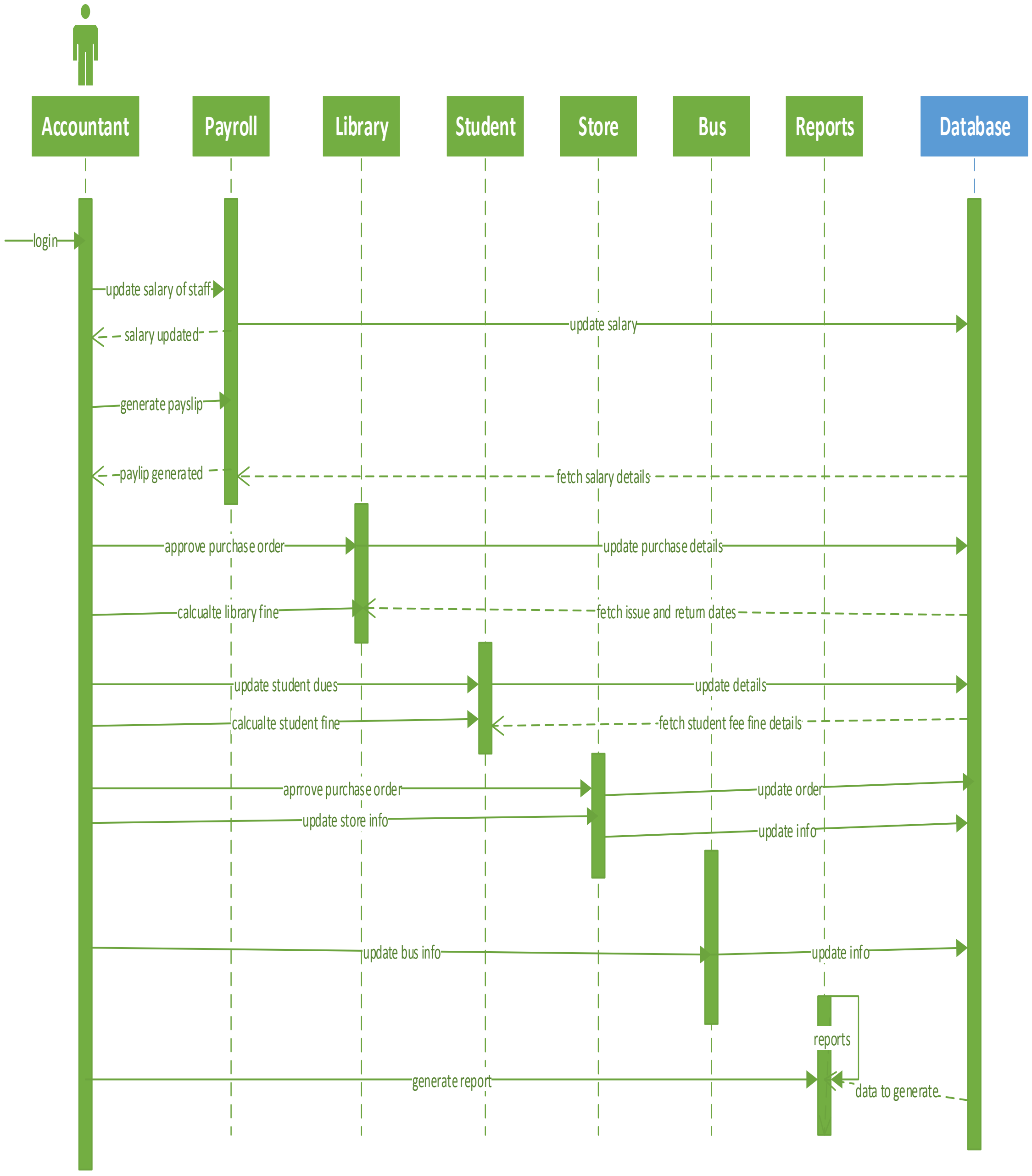


Figure 17: Sequence Diagram - Context

Login

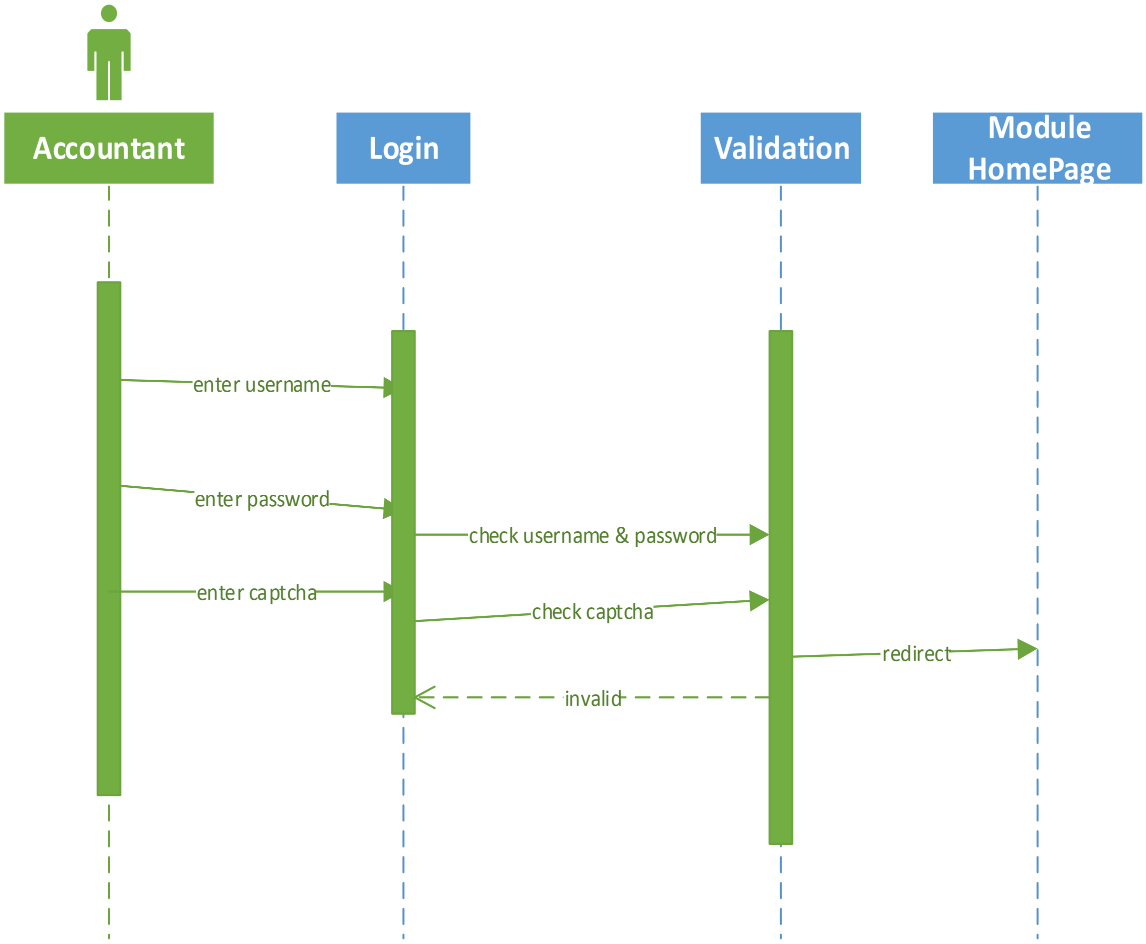


Figure 18: Sequence Diagram - Login

Payment

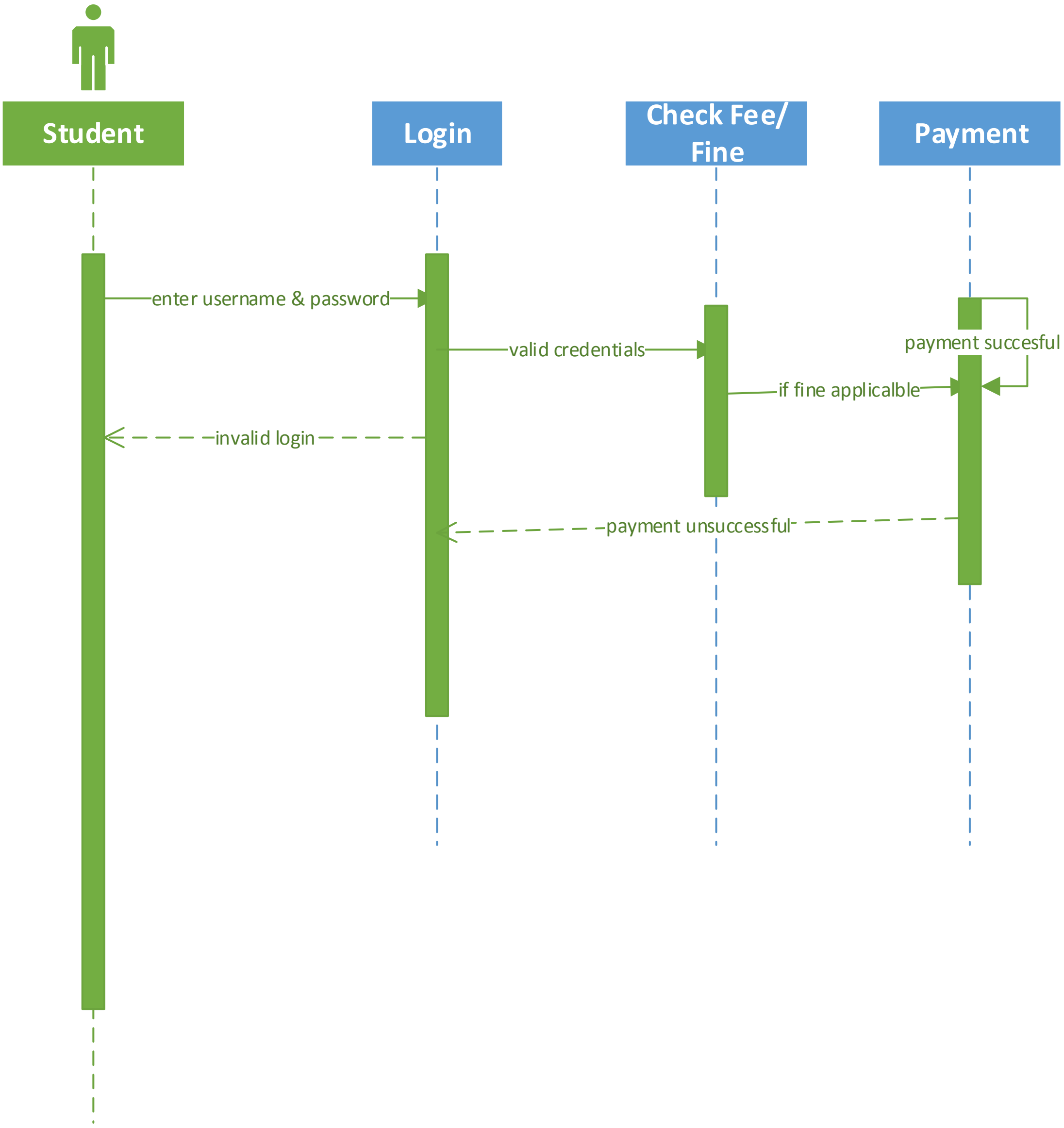


Figure 19: Sequence Diagram - Payment

6.6 Database Design

6.6.1 Entity-Relationship Diagram

The ER Diagram has been made using Crow Foots Notation:

Please Turn Over

6.6.2 Mapping ERD to Relational Schema

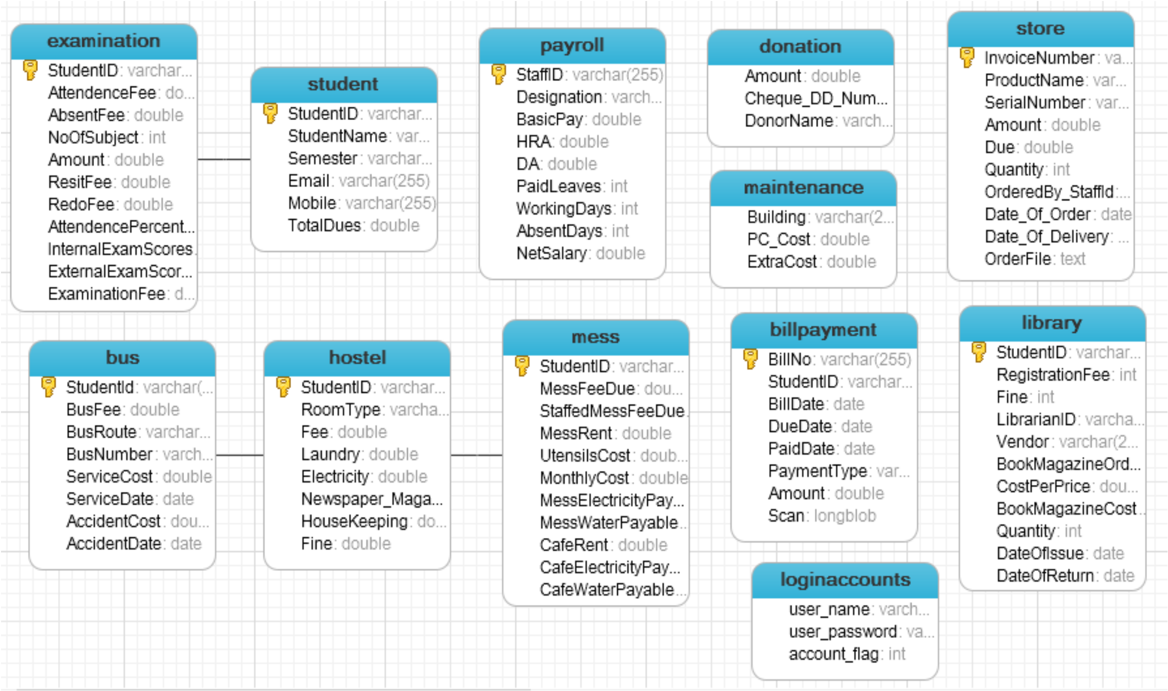


Figure 20: ERD Mapping

6.6.3 Normalization

Normalization is the process of efficiently organizing data in a database. There are two goals of the normalization process: eliminating redundant data and ensuring data dependencies make sense. Both of these are worthy goals as they reduce the amount of space a database consumes and ensure that data is logically stored. However the developer has chosen not to normalize the database due to the following reasons:

- ✓ Normalization leads to creation of n number of tables. Presently, the developer has designed twelve tables and normalizing these tables would lead to creation of a large number of tables. The normalized tables are related to each other with the use of joins and executing a query with joins is an extremely slow process. Hence, de-normalizing would be the best option keeping in mind that the data integrity is maintained.
- ✓ Normalized table designs are complex and difficult to understand. The creation of n number of tables leads to more confusion and error probing also becomes difficult. **(Chpple, 2014)**

6.6.4 Physical Design of Tables

Table Name : loginaccounts				
Attributes	Description	Data Type	Size	Null
User_name	Used to store the username	Varchar	25	No
User_password	Used to store the user password	varchar	25	No
Account_flag	Used to represent the account active/disabled	Int		No
Primary Key		Foreign Key		
-		-		
Relationship description: -				

Table 23: Physical Table Design- loginaccounts

Table Name : payroll				
Attributes	Description	Data Type	Size	Null
StaffID	Used to store the unique Staff id	Varchar	25	No
Designation	Used to store the staff designation	varchar	25	Yes
BasicPay	Used to store the basic pay of staff	Double	-	Yes
HRA	Used to store the HRA of staff	Double	-	Yes
\`DA\`	Used to store the DA of staff	Double	-	Yes
\`PaidLeaves\`	Used to store the paid leaves of staff	Int	-	Yes
\`WorkingDays\`	Used to store the total working days of staff	Int	-	Yes
\`AbsentDays\`	Used to store the absent days of staff	Int	-	Yes
\`NetSalary	Used to store the net salary of staff	Double	-	Yes
Primary Key		Foreign Key		
StaffID		-		
Relationship description: -				

Table 24: Physical Table Design- Payroll

Table Name : library				
Attributes	Description	Data Type	Size	Null
\`StudentID\`	Used to store the unique Student id	Varchar	25	No
\`RegistrationFee\`	Used to store the registration fee	double	-	Yes
\`Fine	Used to store the fine	Double	-	Yes
\`LibrarianID\`	Used to store the librarian id	varchar	25	Yes
\`Vendor\`	Used to store the vendor name	varchar	25	Yes
\`BookMagazineOrdered\`	Used to store ordered ooks	Int	-	Yes
\`CostPerPrice\`	Used to store the cost of book per piece	double	-	Yes
\`BookMagazineCost\`	Used to store the book cost	double	-	Yes
\`Quantity\`	Used to store the quantity of books	int	-	Yes
\`DateOfIssue\`	Used to store the date of issue of books	Date	-	Yes
\`DateOfReturn\`	Used to store the date of return	date	-	yes
Primary Key		Foreign Key		
StudentID		LibrarianID		
Relationship description: This table has relation with student table				

Table 25: Physical Table Design- Library

Table Name : student				
Attributes	Description	Data Type	Size	Null
StudentID\`	Used to store the unique Student id	Varchar	25	No
\`StudentName\`	Used to store the student name	varchar	25	Yes
\`Semester\`	Used to store the semester	varchar	25	Yes
\`Email\`	Used to store the email id	varchar	25	No
\`Mobile\`	Used to store the mobile	varchar	25	No
Totaldues	Used to store the total dues	double	-	Yes
Primary Key		Foreign Key		
StudentID				
Relationship description: -				

Table 26: Physical Table Design- Student

Table Name : store				
Attributes	Description	Data Type	Size	Null
\`InvoiceNumber\`	Used to store the invoice number	Varchar	25	Yes
\`ProductName\`	Used to store the product name	varchar	25	No
\`SerialNumber\`	Used to store the serial number	varchar	25	No
\`Amount\`	Used to store the amount	double	-	No
\`Due\`	Used to store the dues	double	-	No
\`Quantity\`	Used to store the quantity	int	-	No
\`OrderedBy_StaffId\`	Used to store the staff id	Varchar	25	No
\`Date_Of_Order\`	Used to store the order date	Date	-	No
\`Date_Of_Delivery\`	Used to store the delivery date	Date	-	No
\`OrderFile\`	Used to store the purchase form	text	-	no
Primary Key		Foreign Key		
InvoiceNumber				
Relationship description: -				

Table 27: Physical Table Design- Store

Table Name : mess				
Attributes	Description	Data Type	Size	Null
\`StudentID\`	Used to store the student id	Varchar	25	Yes
\`MessFeeDue\`	Used to store the mess fee due	Double	-	No
\`StaffedMessFeeDue\`	Used to store the staff mess fee	Double	-	No
\`MessRent\`	Used to store the mess rent	Double	-	No
\`UtensilsCost\`	Used to store the utensils cost	Double	-	No
\`MonthlyCost\`	Used to store the monthly cost	Double	-	No
\`MessElectricityPayable\`	Used to store the electricity bill	Double	-	No
\`MessWaterPayable\`	Used to store the water bill	Double	-	No
\`CafeRent\`	Used to store the café rent	Double	-	No
\`CafeElectricityPayable\`	Used to store the café electricity bill	Double	-	no
\`CafeWaterPayable\`	Used to store the café water bill	Double		
Primary Key		Foreign Key		
StudentID		StudentID [student]		
Relationship description: - This table has relationship with Student table				

Table 28: Physical Table Design- MessCafe

Table Name : hostel				
Attributes	Description	Data Type	Size	Null
StudentID`	Used to store the unique Student id	Varchar	25	No
`RoomType`	Used to store the room type	varchar	25	Yes
`Fee`	Used to store the fee	Double	-	Yes
`Laundry`	Used to store the laundry	Double	-	Yes
`Electricity`	Used to store the elctrcity	Double	-	Yes
`Newspaper_Magazine`	Used to store the total newspapaer/magazine	Varchar	25	Yes
`HouseKeeping`	Used to store the housekeeping charges	Double	-	Yes
`Fine`	Used to store the fine	Double	-	Yes
Primary Key		Foreign Key		
StudentID				
Relationship description: - this table has relationship with the Student table				

Table 29: Physical Table Design- hostel

Table Name : examination				
Attributes	Description	Data Type	Size	Null
StudentID`	Used to store the unique Student id	Varchar	25	No
`AttendenceFee`	Used to store the attendance fee	double	-	Yes
`AbsentFee`	Used to store the absent fee	Double	-	Yes
`NoOfSubject`	Used to store the no. of subjects	Int	-	Yes
`Amount`	Used to store the amount	Double	-	Yes
`ResitFee`	Used to store the resit fee	Double	-	Yes
`RedoFee`	Used to store the redo fee	Double	-	Yes
`AttendencePercentage`	Used to store the attendance %	Double	-	Yes
`InternalExamScores`	Used to store the internal exam score	Double	-	Yes
`ExternalExamScores`	Used to store the external exam score	Double	-	Yes
`ExaminationFee`	Used to store the exam fee	double	-	yes
Primary Key		Foreign Key		
StudentID		StudentID		
Relationship description: - this table has relationship with the Student table				

Table 30: Physical Table Design- examination

Table Name : bus				
Attributes	Description	Data Type	Size	Null
\`StudentId\`	Used to store the unique Student id	Varchar	25	No
\`BusFee\`	Used to store the bus fee	double	-	Yes
\`BusRoute\`	Used to store the bus route details	varchar	25	Yes
\`BusNumber\`	Used to store the bus number	varchar	25	Yes
\`ServiceCost\`	Used to store the service cost	Double	-	Yes
\`ServiceDate\`	Used to store the service date	Date	-	Yes
\`AccidentCost\`	Used to store the accident cost	Double	-	Yes
\`AccidentDate\`	Used to store the accident date	Date	-	Yes
Primary Key		Foreign Key		
StudentID		StudentID		
Relationship description: - this table has relationship with the Student table				

Table 31: Physical Table Design- bus

Table Name : billpayment				
Attributes	Description	Data Type	Size	Null
BillNo\`	Used to store the unique bill number	Varchar	25	No
\`BillDate\`	Used to store the bill date	Date	-	Yes
\`DueDate\`	Used to store the due date	Date	-	Yes
\`PaidDate\`	Used to store the paid date	Date	-	Yes
\`PaymentType\`	Used to store the payment type	Varchar	25	Yes
\`Amount\`	Used to store the amount	Double	-	Yes
\`Scan\`	Used to store the scan of bill	Long Blob	-	Yes
Primary Key		Foreign Key		
BillNo		-		
Relationship description: - -				

Table 32: Physical Table Design- billpayment

6.7 Architecture Design

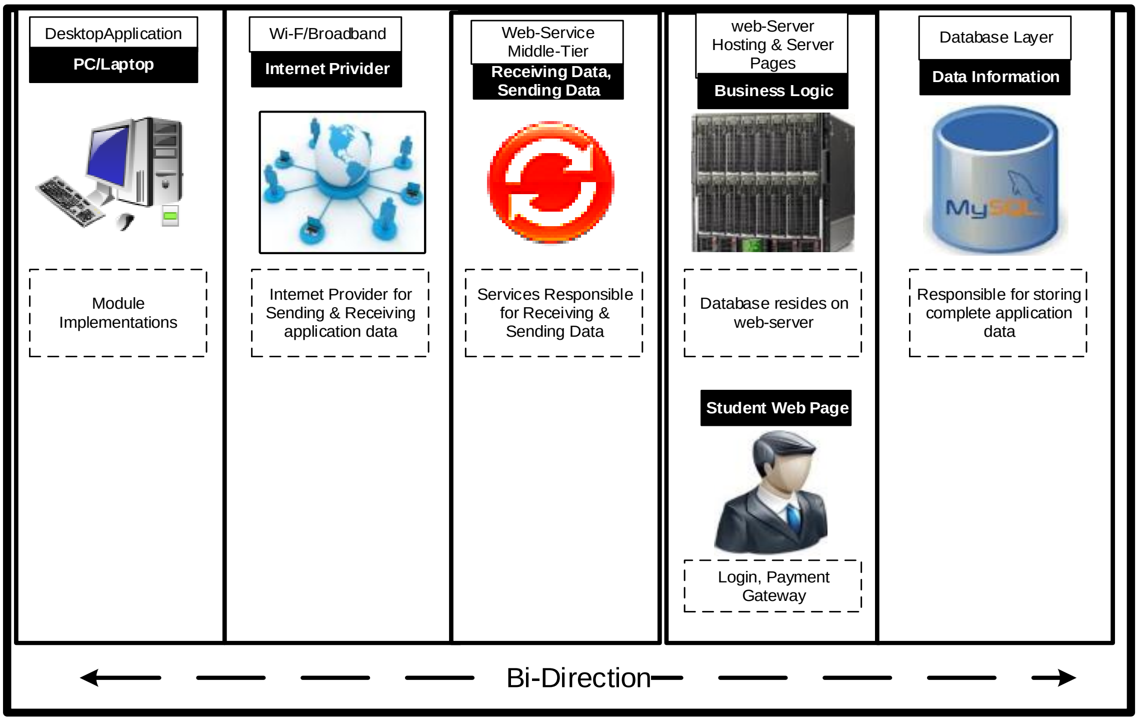

DesktopApplication
PC/Laptop
Module Implementations
Wi-F/Broadband
Internet Privider
Internet Provider for Sending & Receiving application data
Web-Service Middle-Tier
Receiving Data, Sending Data
Services Responsible for Receiving & Sending Data
web-Server Hosting & Server Pages
Business Logic
Database resides on web-server
Student Web Page
Login, Payment Gateway
Database Layer
Data Information
MySQL
Responsible for storing complete application data
Bi-Direction

Figure 21: Architecture Design

7. Implementation

After the completion of the design phase, now it's the task of the developer to convert those designs into the system functionalities. The proposed system is divided into several modules. The modules are prepared on the basis of their priority.

7.1 Implementation Plan

7.1.1 Tools used during Implementation

No.	Development Tools	Purpose
1	NetBeans 7.4	Primary IDE used for the purpose of coding.
2.	Eclipse	Secondary IDE used for functionality testing.
3.	Apache Tomcat	Primary HTTP server used in development phase.
5.	JDK 1.7	Primary SDKs used.
6.	MySQL 5.0	Used as the backend database.
7.	Google Chrome	Primary browser to check for all web activities during development.
8.	Microsoft Picture Manager 2007/Paint	For Image editing
9.	Microsoft Project 2013	For designing Gantt Chart
10.	Microsoft Visio 2013	For drawing UML and ER diagrams.

Table 33: Case Tools – Implementation

7.1.2 Implementation Approach

The implementation steps which needs to be followed in order to complete the system within time and fulfilling all requirements:

- ✓ First and foremost, the database tables needs to be designed on the basis of ERD and database design done in the System Design part of the documentation.
- ✓ After the database is created, the login and registration forms need to be designed.
- ✓ Afterwards, work on complex modules like Payroll, Student Fee/fine, Store/Inventory is started.
- ✓ The remaining modules are worked upon.
- ✓ Then, all the modules are integrated with each other.
- ✓ If the integration is successful, the entire system is checked for any faults
- ✓ Then, the web part of the project is designed.
- ✓ This web part and the desktop part are integrated with each other with a common database.

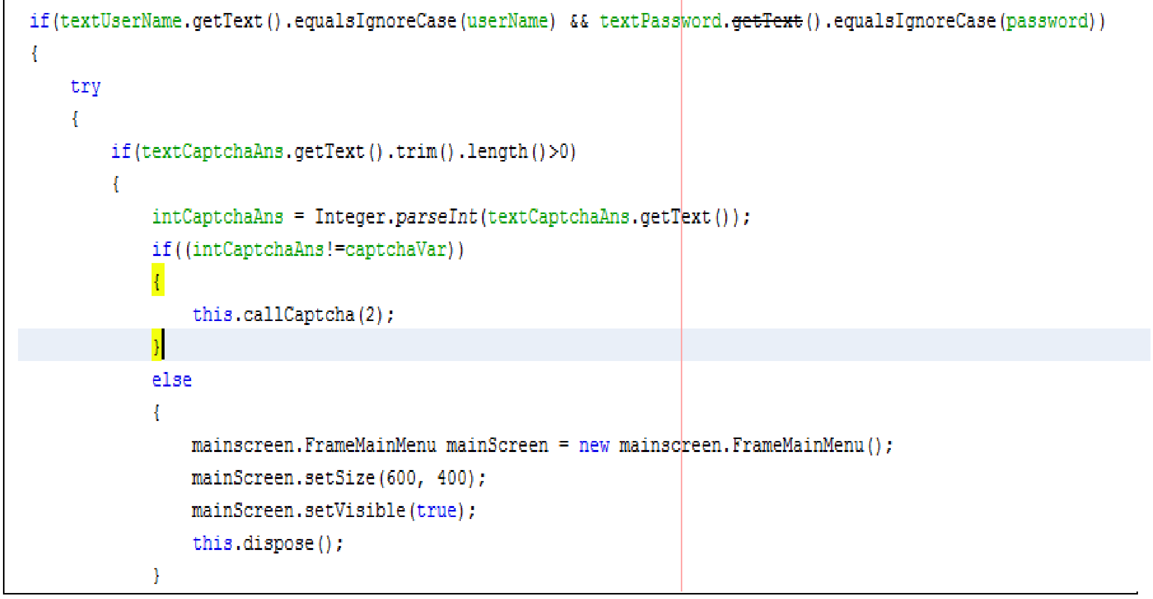
7.2 Implementing Different Complex Modules

Login Module	
Objective	In this module, the user needs to input the username, password and the captcha code in order to login into the system
Screen Shot	Screen Shot has been provided in User Manual – System Walkthrough

Pseudo code for User Login

```
STEP1 START
STEP 2.1 Enter username, password, captcha code
STEP 2.2 Click Login
STEP 2.3 IF flag=0  entered
STEP 2.4 Then check captcha code is correct
STEP 2.5 END IF
STEP 2.6 ELSE display “account blocked”
STEP 2.7 END IF
STEP 2.8 ELSE back to login page and display “Username/password/captcha  is incorrect ”
STEP 3 END IF
STEP 4 END
```

Command Applied



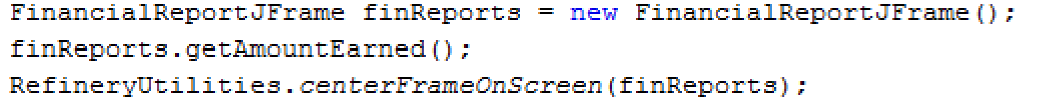
```
if(textUserName.getText().equalsIgnoreCase(userName) && textPassword.getText().equalsIgnoreCase(password))
{
    try
    {
        if(textCaptchaAns.getText().trim().length()>0)
        {
            intCaptchaAns = Integer.parseInt(textCaptchaAns.getText());
            if((intCaptchaAns!=captchaVar))
            {
                this.callCaptcha(2);
            }
            else
            {
                mainscreen.FrameMainMenu mainScreen = new mainscreen.FrameMainMenu();
                mainScreen.setSize(600, 400);
                mainScreen.setVisible(true);
                this.dispose();
            }
```

Report Generation	
Objective	In this module, the user can generate various types of reports
Screen Shot	Screen Shot has been provided in User Manual – System Walkthrough

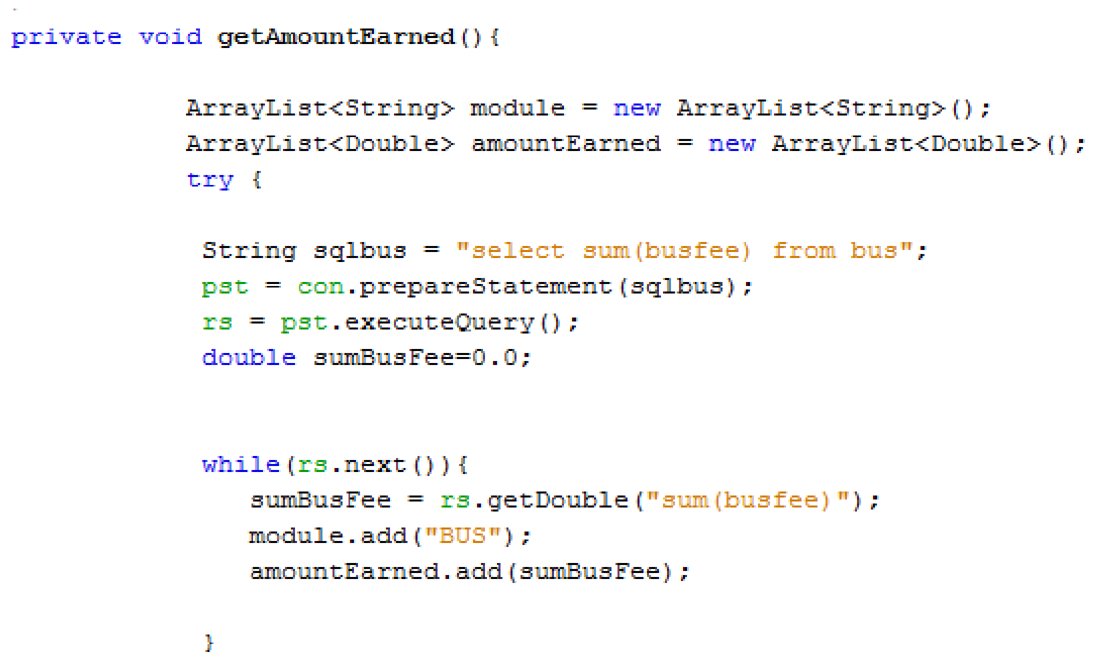
Pseudo code for User Login

```
STEP1 START
STEP 2.1 Select report type
STEP 2.2 If report type=’financial’
STEP 2.3 then get totalamount
STEP 2.4 map totalamount to jfreechart
Step 3 EndIF
STEP 4 END
```

Command Applied



```
FinancialReportJFrame finReports = new FinancialReportJFrame();
finReports.getAmountEarned();
RefineryUtilities.centerFrameOnScreen(finReports);
```



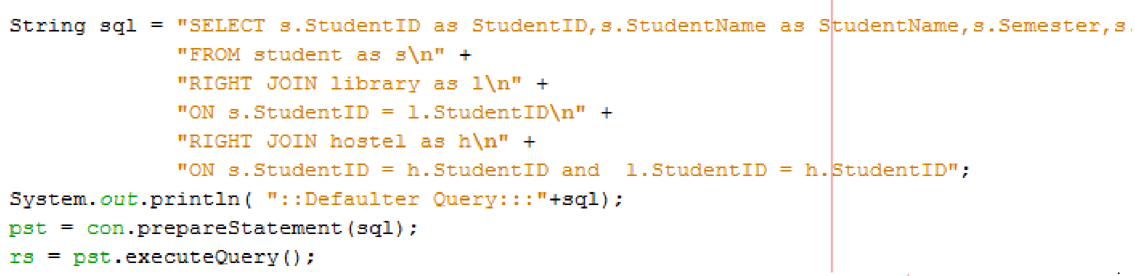
```
private void getAmountEarned(){

            ArrayList<String> module = new ArrayList<String>();
            ArrayList<Double> amountEarned = new ArrayList<Double>();
            try {

             String sqlbus = "select sum(busfee) from bus";
             pst = con.prepareStatement(sqlbus);
             rs = pst.executeQuery();
             double sumBusFee=0.0;


             while(rs.next()){
                 sumBusFee = rs.getDouble("sum(busfee)");
                 module.add("BUS");
                 amountEarned.add(sumBusFee);

             }
```

Defaulter List + Notification	
Objective	In this module, the user can generate a defaulters list and send notifications to the defaulters
Screen Shot	Screen Shot has been provided in User Manual – System Walkthrough

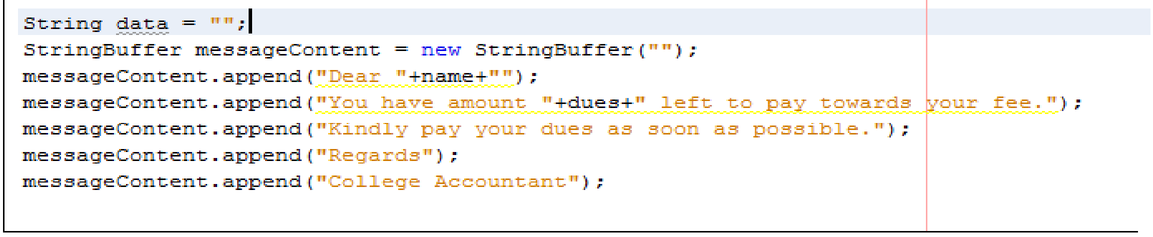
Pseudo code for User Login

```
STEP1 START
STEP 2.1 Generate Defaulter list
STEP 2.2 populate list of defualters ’
STEP 2.3  send sms to mobile numbers
STEP 2.4 send email to email ids
STEP 3 END
```

Command Applied



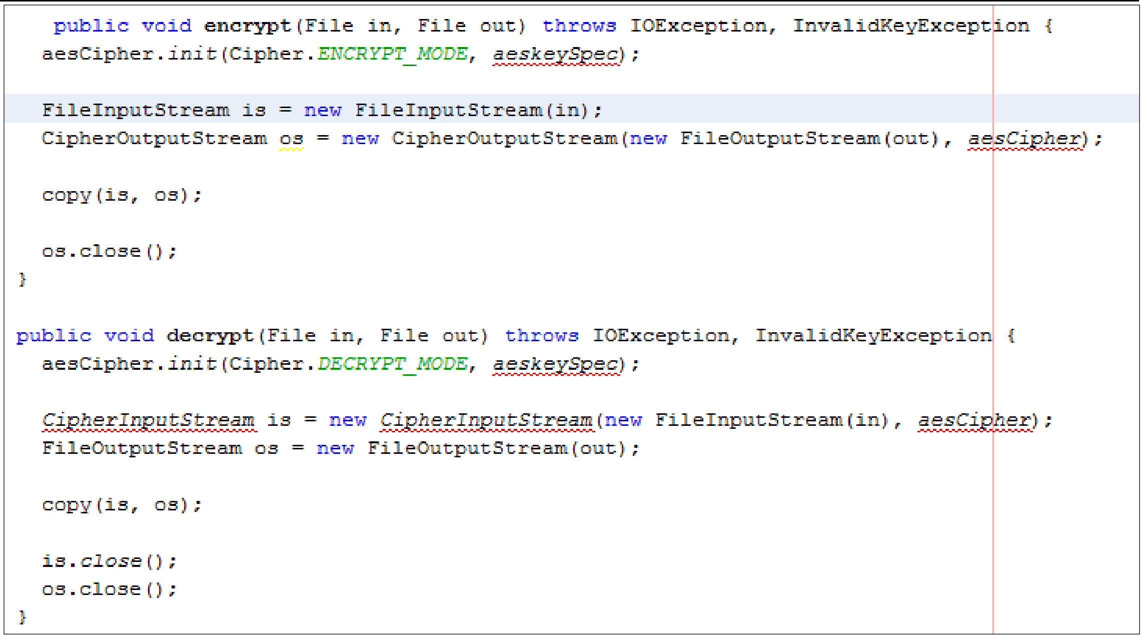
```
String sql = "SELECT s.StudentID as StudentID,s.StudentName as StudentName,s.Semester,s.
             "FROM student as s\n" +
             "RIGHT JOIN library as l\n" +
             "ON s.StudentID = l.StudentID\n" +
             "RIGHT JOIN hostel as h\n" +
             "ON s.StudentID = h.StudentID and  l.StudentID = h.StudentID";
System.out.println( "::Defaulter Query:::"+sql);
pst = con.prepareStatement(sql);
rs = pst.executeQuery();
```



```
String data = "";
StringBuffer messageContent = new StringBuffer("");
messageContent.append("Dear  "+name+"");
messageContent.append("You have amount "+dues+" left to pay towards your fee.");
messageContent.append("Kindly pay your dues as soon as possible.");
messageContent.append("Regards");
messageContent.append("College Accountant");
```

7.3 Parts of code that didn’t work

The developer had tried to secure the data of the database using the concept of encryption and decryption. However, due to a ‘secret key storage’ problem the data was not able to be encrypted.



```
  public void encrypt(File in, File out) throws IOException, InvalidKeyException {
  aesCipher.init(Cipher.ENCRYPT_MODE, aeskeySpec);

  FileInputStream is = new FileInputStream(in);
  CipherOutputStream os = new CipherOutputStream(new FileOutputStream(out), aesCipher);

  copy(is, os);

  os.close();
}

public void decrypt(File in, File out) throws IOException, InvalidKeyException {
  aesCipher.init(Cipher.DECRYPT_MODE, aeskeySpec);

  CipherInputStream is = new CipherInputStream(new FileInputStream(in), aesCipher);
  FileOutputStream os = new FileOutputStream(out);

  copy(is, os);

  is.close();
  os.close();
}
```

7.4 Technical Manual

7.4.1 Installing and setting-up NetBeans 7.4

Step1: Download NetBeans IDE 7.4 setup from: *https://netbeans.org/downloads/7.4/start.html?platform=windows&lang=en&option=javaee*

Step2: After, downloading, the NetBeans setup, install it by following the onscreen instructions.

Step3: On successful installation, open NetBeans, go to File->Open Project

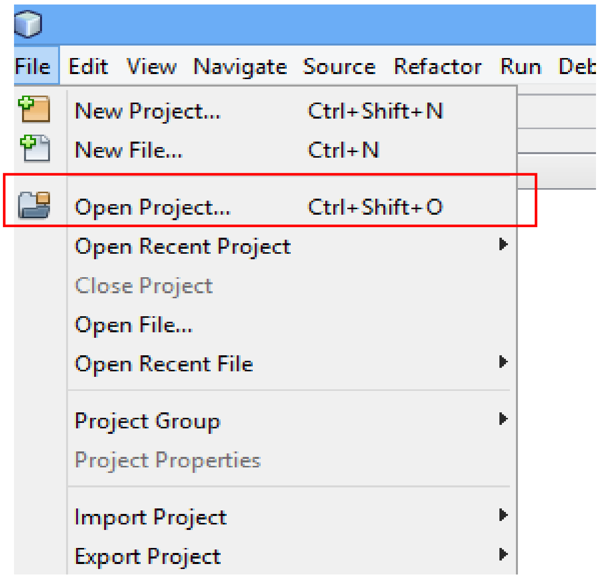


Figure 22: Technical manual

Step4: After clicking Open Project, browse for the folder named ‘CollegeAccountant’ in the CD.

Step5: On successfully opening the project, then right-click on the project name in Project Explorer and go to Resolve Project Problems

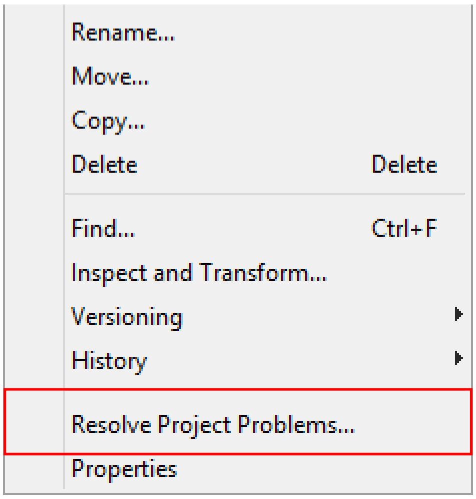


Figure 23: Technical manual

Step6: After clicking, resolve project problems, a list of problems would be shown. Click on a problem, then click Resolve, then search for the folder named ‘jar’ inside the ‘CollegeAccountant’ folder. Find the problem name and the corresponding file name inside the jar folder.

Step7: On resolving all the problems related to missing jar files, the project can be run by right-clicking the project name in Project Explorer and then click run.

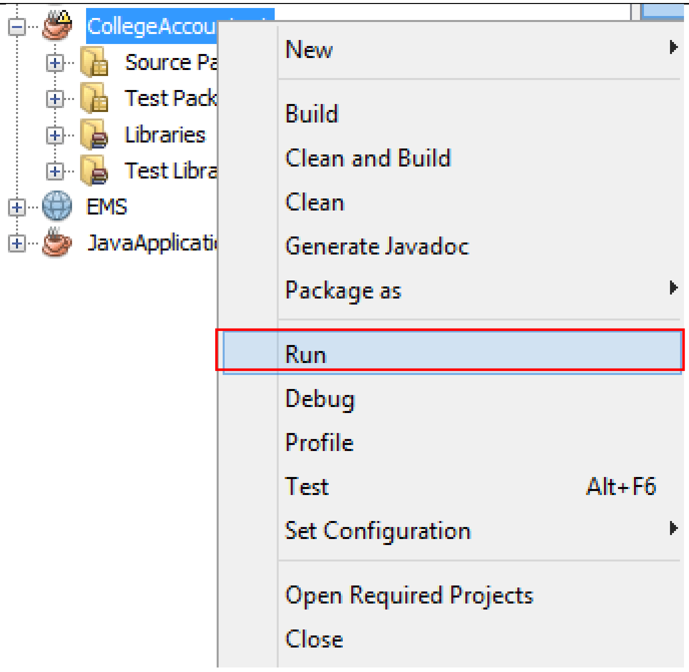


Figure 24: Technical manual

7.4.2 Installing and Setting-up MySQL 5.0

Step1: Download setup for MySQL5.0 from: *http://dev.mysql.com/downloads/*

Step2: Install MySQL setup and follow the onscreen instructions with *Username*: root and *Password*: 1234.

Step3: Open MySQL5.0 through NaviCat, create a new database with name: ‘collegeaccountant’

Step4: Click on the database name, then go to Query-> New Query-> Load

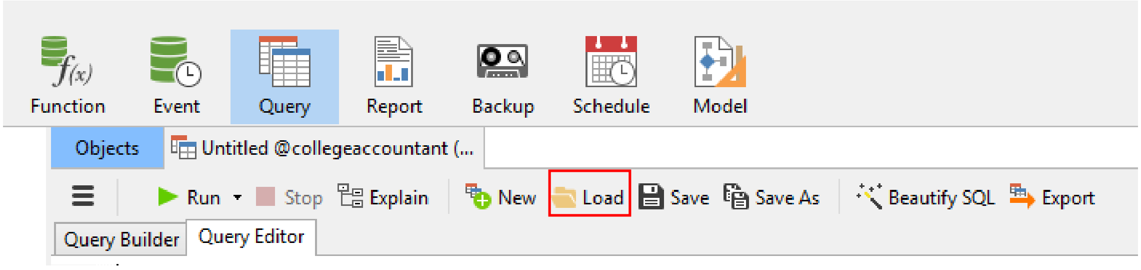


Figure 25 Technical manual

Step5: After clicking load, browse for the file name ‘collegeaccountant.sql’ inside the CollegeAccountant folder and then click on RUN.

Step6: Check if the tables have been created inside the database named ‘collegeaccountant’.

7.5 User Manual

7.5.1 Sitemap

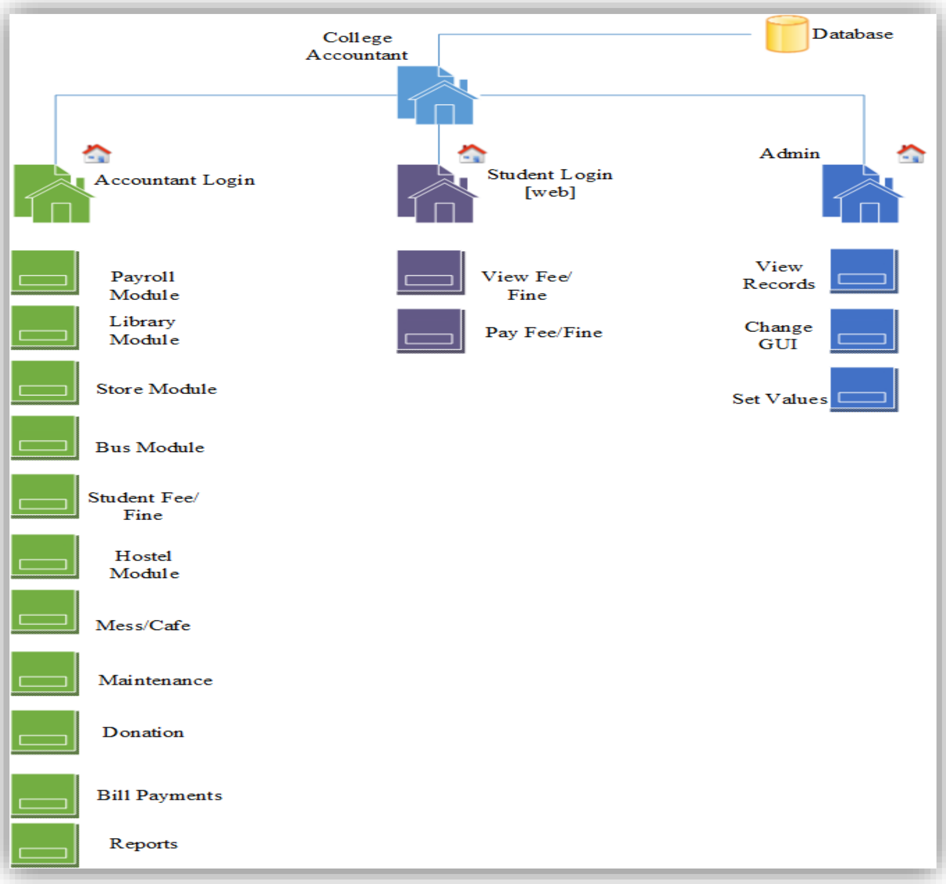
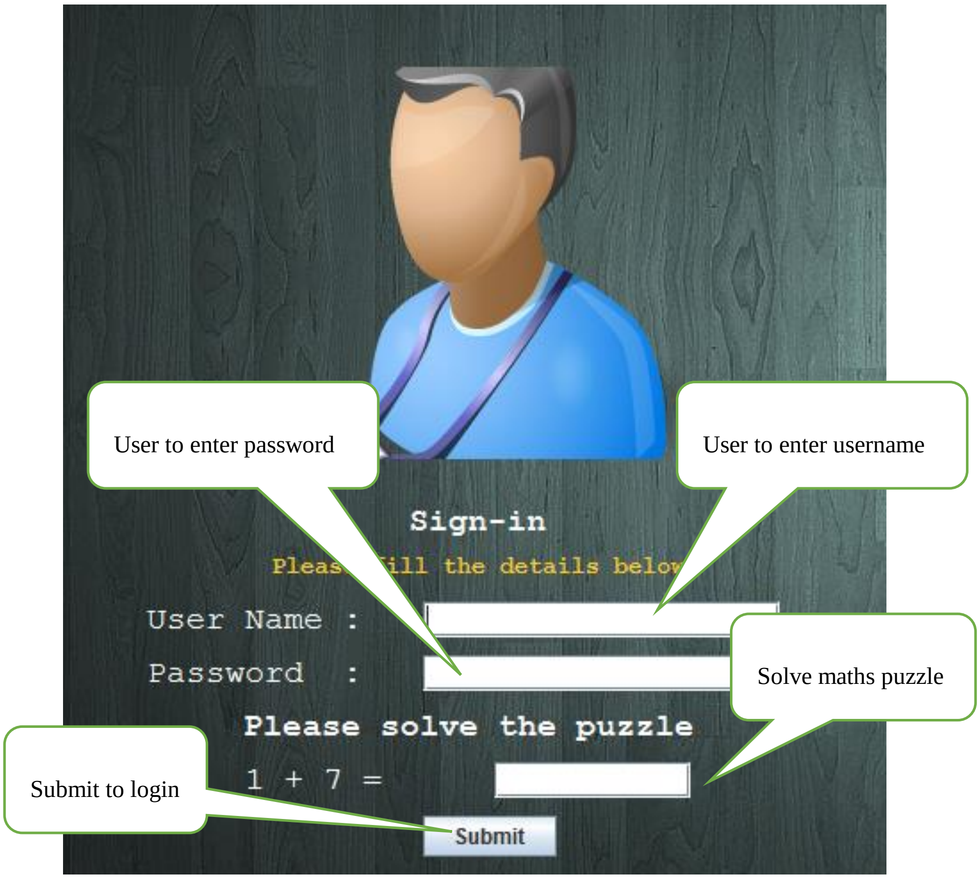


Figure 26: Sitemap

7.5.2 System Walk-Through

Login Screen



User to enter password

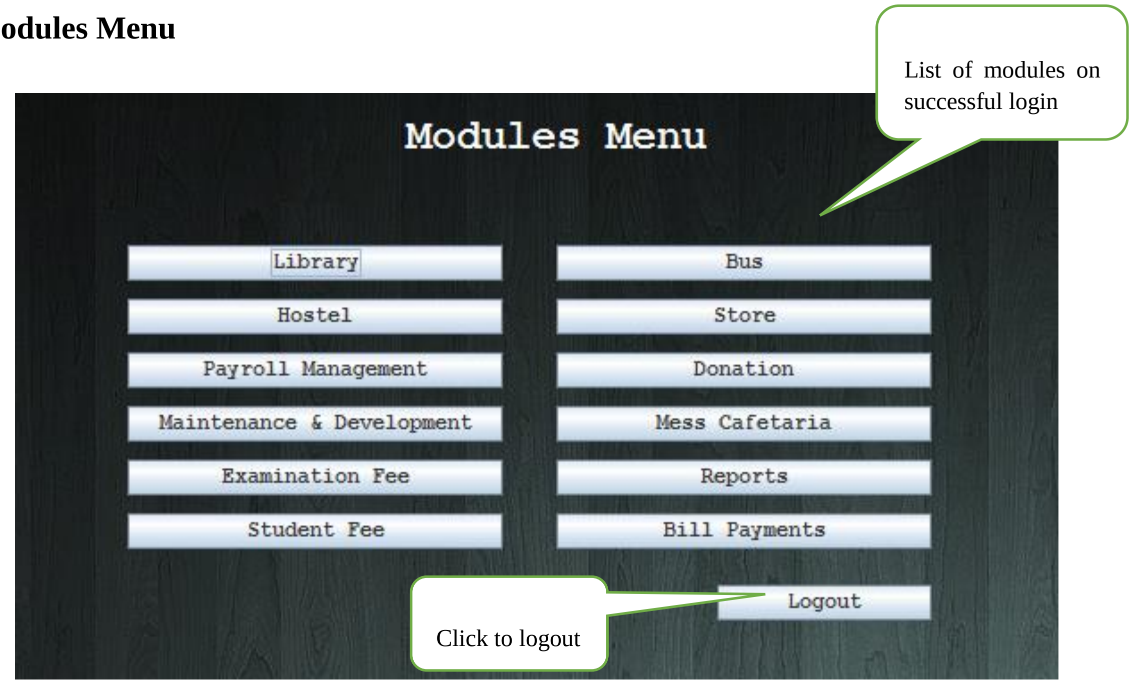
User to enter username

Solve maths puzzle

Submit to login

Figure 27: Screenshot - Login

Modules Menu

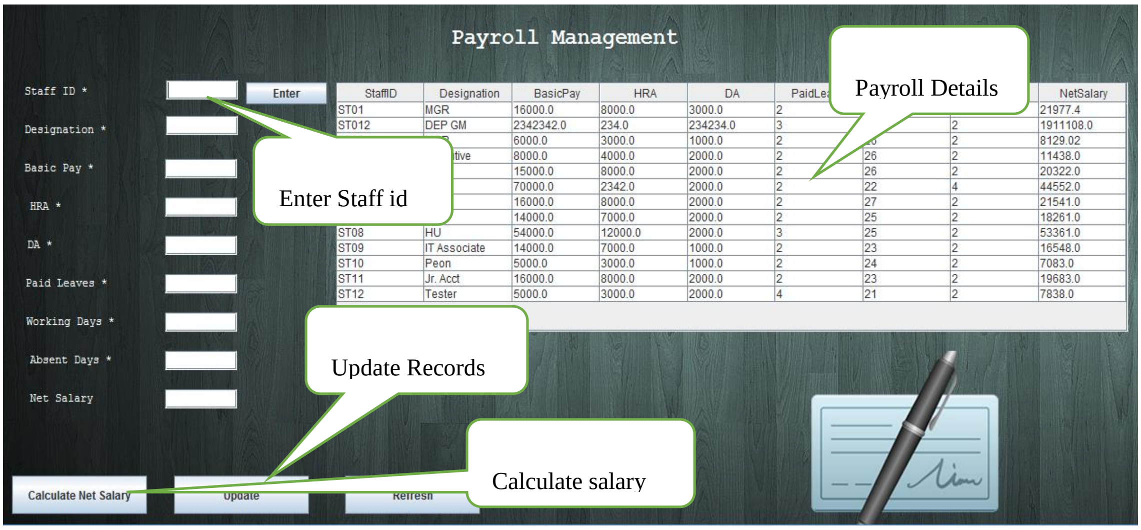


List of modules on successful login

Click to logout

Figure 28: Screenshot- Modules Menu

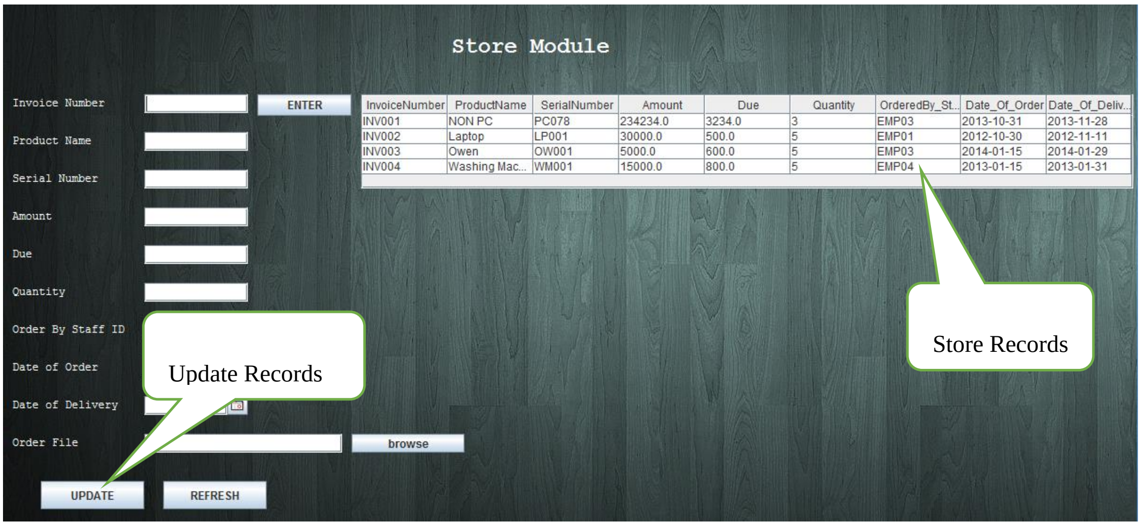
Payroll Module



StaffID	Designation	BasicPay	HRA	DA	PaidLe			NetSalary
ST01	MGR	16000.0	8000.0	3000.0	2			21977.4
ST012	DEP GM	2342342.0	234.0	234234.0	3		2	1911108.0
		6000.0	3000.0	1000.0	2	[illegible]	2	8129.02
	tive	8000.0	4000.0	2000.0	2	26	2	11438.0
		15000.0	8000.0	2000.0	2	26	2	20322.0
		70000.0	2342.0	2000.0	2	22	4	44552.0
		16000.0	8000.0	2000.0	2	27	2	21541.0
		14000.0	7000.0	2000.0	2	25	2	18261.0
ST08	HU	54000.0	12000.0	2000.0	3	25	2	53361.0
ST09	IT Associate	14000.0	7000.0	1000.0	2	23	2	16548.0
ST10	Peon	5000.0	3000.0	1000.0	2	24	2	7083.0
ST11	Jr. Acct	16000.0	8000.0	2000.0	2	23	2	19683.0
ST12	Tester	5000.0	3000.0	2000.0	4	21	2	7838.0

Figure 29: Screenshot - Payroll Module

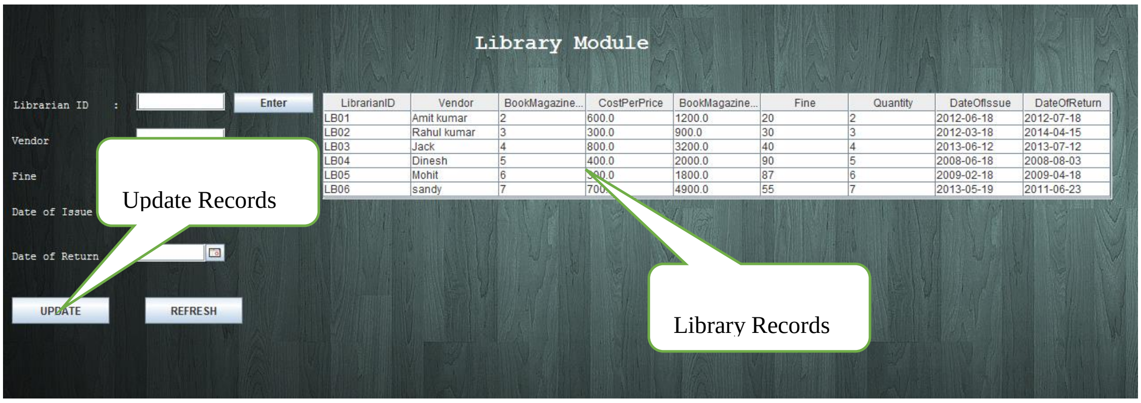
Store Module



InvoiceNumber	ProductName	SerialNumber	Amount	Due	Quantity	OrderedBy_St...	Date_Of_Order	Date_Of_Deliv...
INV001	NON PC	PC078	234234.0	3234.0	3	EMP03	2013-10-31	2013-11-28
INV002	Laptop	LP001	30000.0	500.0	5	EMP01	2012-10-30	2012-11-11
INV003	Owen	OW001	5000.0	600.0	5	EMP03	2014-01-15	2014-01-29
INV004	Washing Mac...	WM001	15000.0	800.0	5	EMP04	2013-01-15	2013-01-31

Figure 30: Screenshot - Store Module

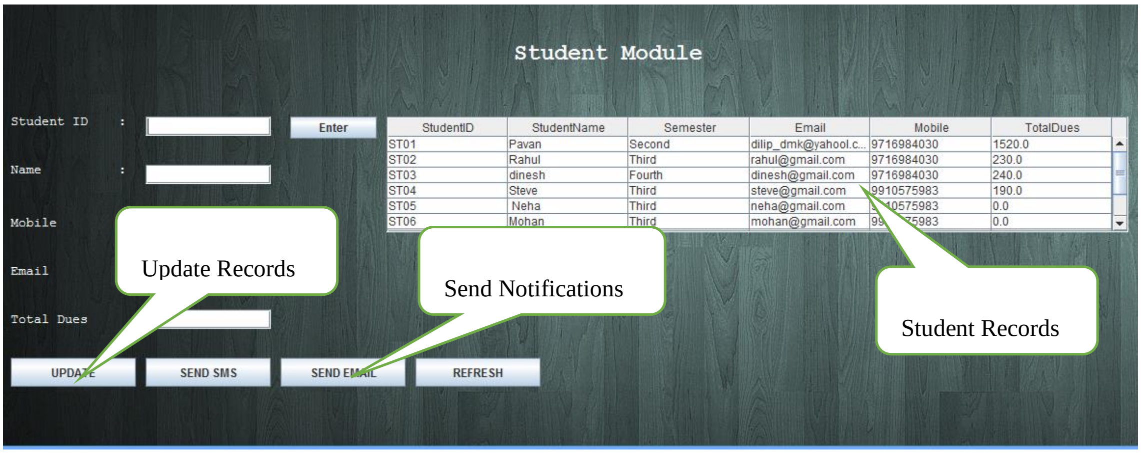
Library Module



LibrarianID	Vendor	BookMagazine...	CostPerPrice	BookMagazine...	Fine	Quantity	DateOfIssue	DateOfReturn
LB01	Amit kumar	2	600.0	1200.0	20	2	2012-06-18	2012-07-18
LB02	Rahul kumar	3	300.0	900.0	30	3	2012-03-18	2014-04-15
LB03	Jack	4	800.0	3200.0	40	4	2013-06-12	2013-07-12
LB04	Dinesh	5	400.0	2000.0	90	5	2008-06-18	2008-08-03
LB05	Mohit	6	[illegible]00.0	1800.0	87	6	2009-02-18	2009-04-18
LB06	sandy	7	700[illegible]	4900.0	55	7	2013-05-19	2011-06-23

Figure 31: Screenshot - Library Module

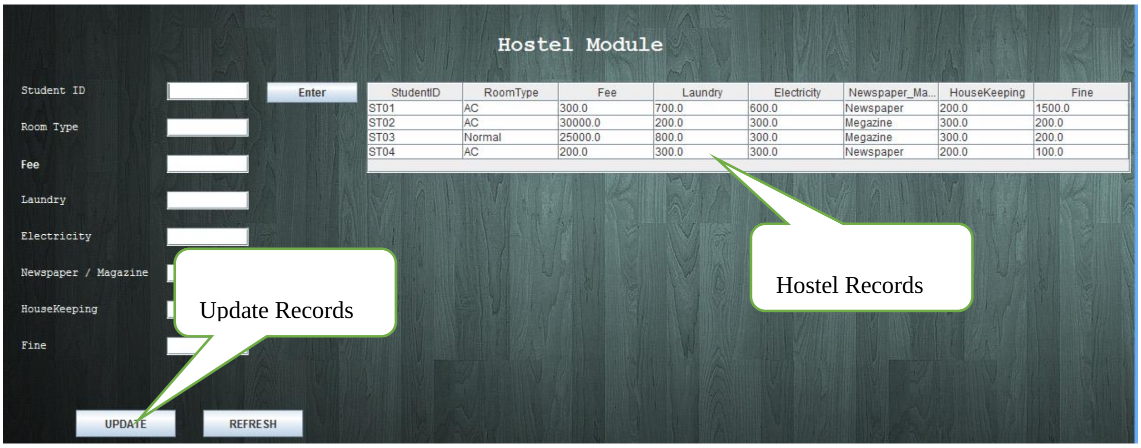
Student Fee/Fine Module



StudentID	StudentName	Semester	Email	Mobile	TotalDues
ST01	Pavan	Second	dilip_dmk@yahool.c...	9716984030	1520.0
ST02	Rahul	Third	rahul@gmail.com	9716984030	230.0
ST03	dinesh	Fourth	dinesh@gmail.com	9716984030	240.0
ST04	Steve	Third	steve@gmail.com	9910575983	190.0
ST05	Neha	Third	neha@gmail.com	[illegible]10575983	0.0
ST06	Mohan	Third	mohan@gmail.com	99[illegible]75983	0.0

Figure 32: Screenshot - Student Module

Hostel Module



StudentID	RoomType	Fee	Laundry	Electricity	Newspaper_Ma...	HouseKeeping	Fine
ST01	AC	300.0	700.0	600.0	Newspaper	200.0	1500.0
ST02	AC	30000.0	200.0	300.0	Megazine	300.0	200.0
ST03	Normal	25000.0	800.0	300.0	Megazine	300.0	200.0
ST04	AC	200.0	300.0	300.0	Newspaper	200.0	100.0

Figure 33: Screenshot - Hostel Module

Reports

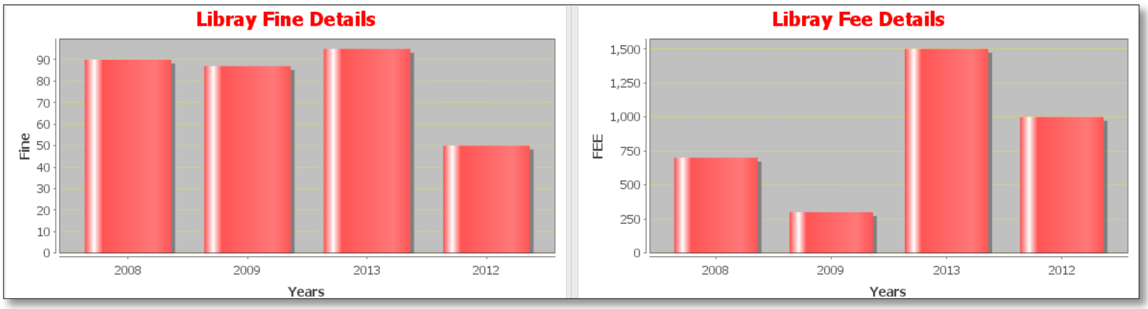
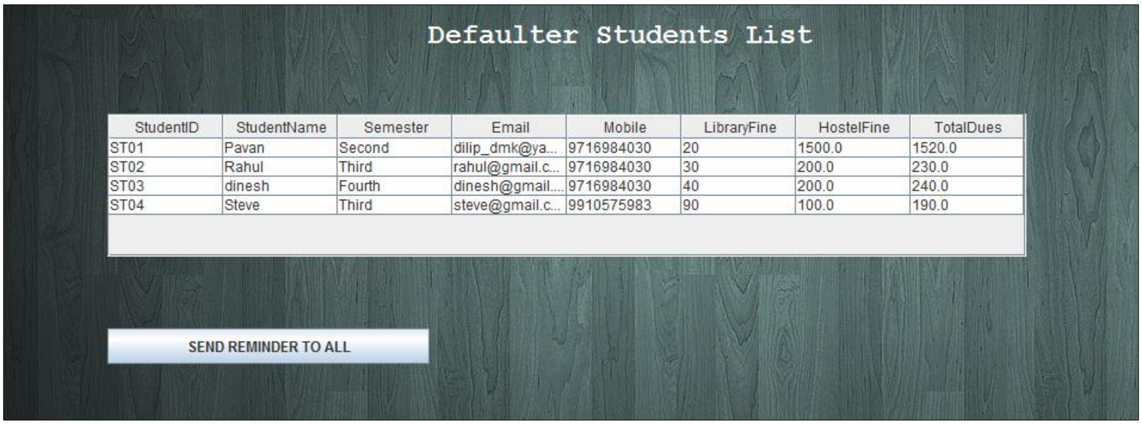


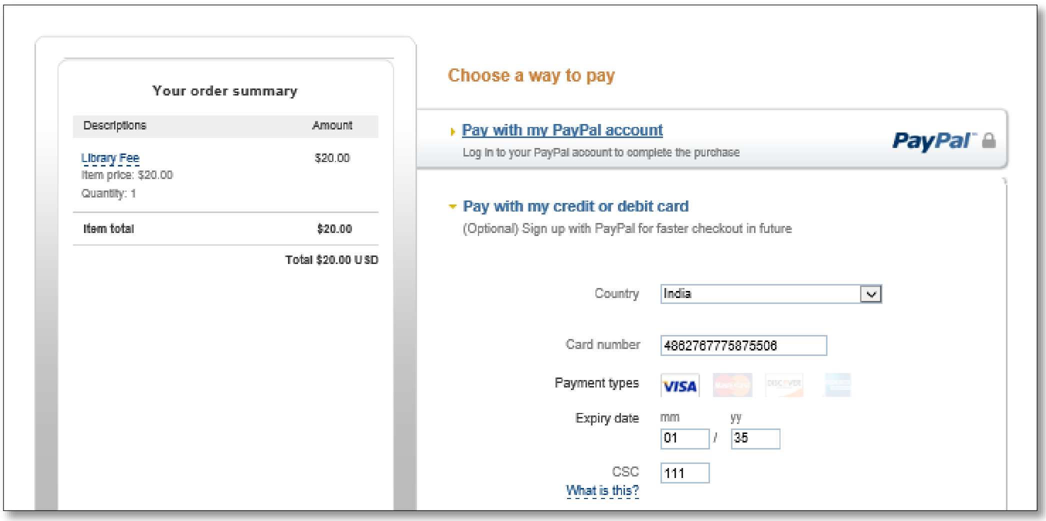
Figure 34: Screenshot - Reports



StudentID	StudentName	Semester	Email	Mobile	LibraryFine	HostelFine	TotalDues
ST01	Pavan	Second	dilip_dmk@ya...	9716984030	20	1500.0	1520.0
ST02	Rahul	Third	rahul@gmail.c...	9716984030	30	200.0	230.0
ST03	dinesh	Fourth	dinesh@gmail....	9716984030	40	200.0	240.0
ST04	Steve	Third	steve@gmail.c...	9910575983	90	100.0	190.0

Figure 35: Screenshot – Reports

Payment Gateway



Your order summary

Descriptions	Amount
Library Fee Item price: $20.00 Quantity: 1	$20.00
Item total	$20.00
	Total $20.00 USD

Choose a way to pay

Pay with my PayPal account
Log in to your PayPal account to complete the purchase

Pay with my credit or debit card
(Optional) Sign up with PayPal for faster checkout in future

Country: India

Card number: 4862767775875506

Payment types: VISA

Expiry date: mm 01 / yy 35

CSC: 111

What is this?

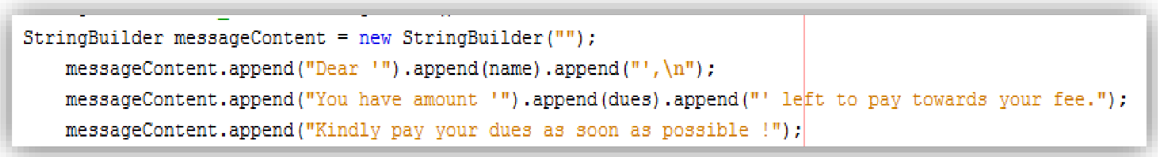
Figure 36: Screenshot - Payment Gateway

7.6 Technical Quality

This system "College Accountant" has been developed in Java, an object-oriented language. The developer needed to follow some guidelines in order to maintain the object-oriented characteristics of the system by:

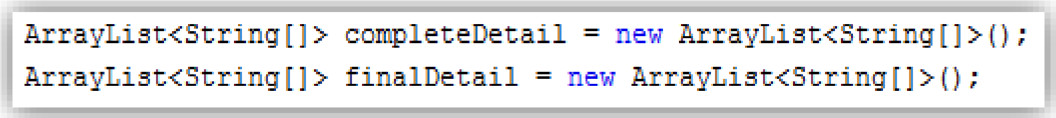
- ✓ *Use of Object Principles:* this system uses OOP's design principle wherever required:
  1. Encapsulation (Class and Methods) – Proper class and method are used to encapsulate program. For each task a class is created which contains data members and methods.
  2. Inheritance & Interface – Appropriate Inheritance and interface is used to implement the design.
  3. Polymorphism – System developed uses adequate amount of polymorphism concept such as method overloading, overriding etc**.**

- ✓ *Proper Memory Utilization*: to improve the efficiency of the program, the developer has made use of String builder, array list in place of String and Arrays/vectors as these utilizes less memory and are more dynamic and efficient.



```
StringBuilder messageContent = new StringBuilder("");
    messageContent.append("Dear '").append(name).append("',\n");
    messageContent.append("You have amount '").append(dues).append("' left to pay towards your fee.");
    messageContent.append("Kindly pay your dues as soon as possible !");
```

Figure 37: Technical Quality- Memory Utilization

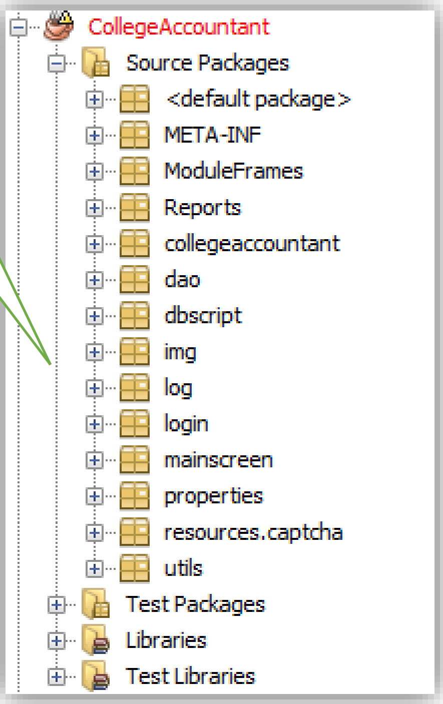


```
ArrayList<String[]> completeDetail = new ArrayList<String[]>();
ArrayList<String[]> finalDetail = new ArrayList<String[]>();
```

Figure 38: Technical Quality- Memory Utilization

- ✓ *Performance*: since this system requires very less memory utilization it can easily work without any performance issues. Also, the system is not too much dependent on the speed of the internet connection.
- ✓ *Loosely coupled design*: The factor of dependency of one class to other class is known as coupling. A system having high coupling should be avoided since:
    - Its highly unstable and requires huge efforts to modify
    - It decreases reusability of classes as dependant classes also need to be included
    - Individual classes are difficult to understand in isolation

Hence, systems with low/loose coupling is preferred. In ‘College Accountant’, the developer has bifurcated different classes into different packages.



Different classes being bifurcated into different packages

Figure 39: Technical Quality- Loosely Coupled

- ✓ *User Interface*: all the design principles were followed in the designing of the system.
  1. *Visibility*- is achieved by placing the controls in a highly visible location, thus developer kept the check on visibility of content in every page.
  2. *Feedback*- is the provision of information to a user about the result of an action. The developer has used proper message-boxes/in-line messages providing feedback to an user input,
  3. *Constraints*- are properties of an object that limit the ways in which it can be used. The developer has used constraints wherever possible, for example, used mobile phone constraints i.e. Mobile phone entered should have 10 digits only.
  4. *Mapping* – For mapping in the system, the developer has provided various mapping marks such as metaphors on various buttons to map them with the real world entity. .
  5. *Consistency*- Design interfaces to have similar operations and use similar elements for similar tasks, for this purpose the developer has provided consistent form layouts throughout the application.

6. *Affordances*- To accomplish affordances, real world accounting feel is provided in whole application, so that affordances of various object with which any new accountant deals in daily life can use system effectively.

- ✓ *Proper Indentation:* is very important to make the code more readable and presentable so that there will not be any problem to read the code properly. It makes the code more attractive and readable. The developer has ensured that the implementation uses proper indentation.

- ✓ *Proper Naming Convention:* it is very important to follow the proper naming conventions to define the classes, objects and methods. It helps avoid dangerous chaotic situation which can lead to logical error. The developer has ensured that the implementation uses proper naming convention.

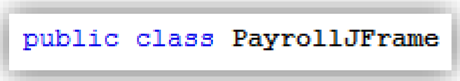
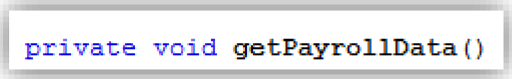
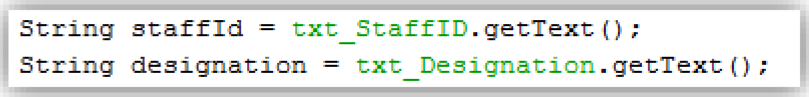
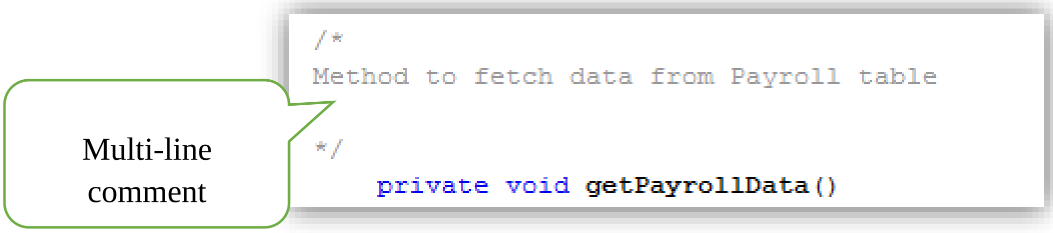
Naming Convention for Class
 `public class PayrollJFrame`
PayrollJFrame is the name of the class which contains operations related to payroll module.
Naming Convention for Method
 `private void getPayrollData()`
getPayrollData() is the name of the method which fetches the payroll details from the database
Naming Convention for Variables
 `String staffId = txt_StaffID.getText();` `String designation = txt_Designation.getText();`
Staffidid is the name for the unique id given to staff and designation is the designation of the staff .

Figure 40: Naming Convention

- ✓ *Proper Comments*: Usage of comment is a sign of good programming. It is utmost important for the maintenance of the system. Commenting the codes in the program enables the third party to understand the program logic and behaviour.

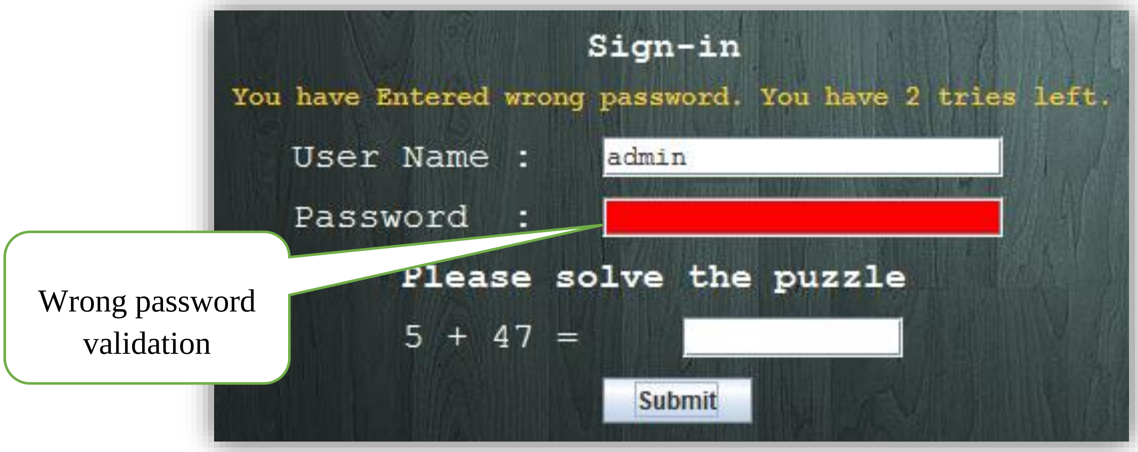


```
/*
Method to fetch data from Payroll table

*/
    private void getPayrollData()
```

Multi-line comment

Figure 41: Comment Example

- ✓ *Proper Validations*: is given to ensure that the user provided necessary and properly formatted information needed to successfully complete an operation. The developer has ensured that the implementation uses proper validations.



Sign-in

You have Entered wrong password. You have 2 tries left.

User Name : admin

Password :

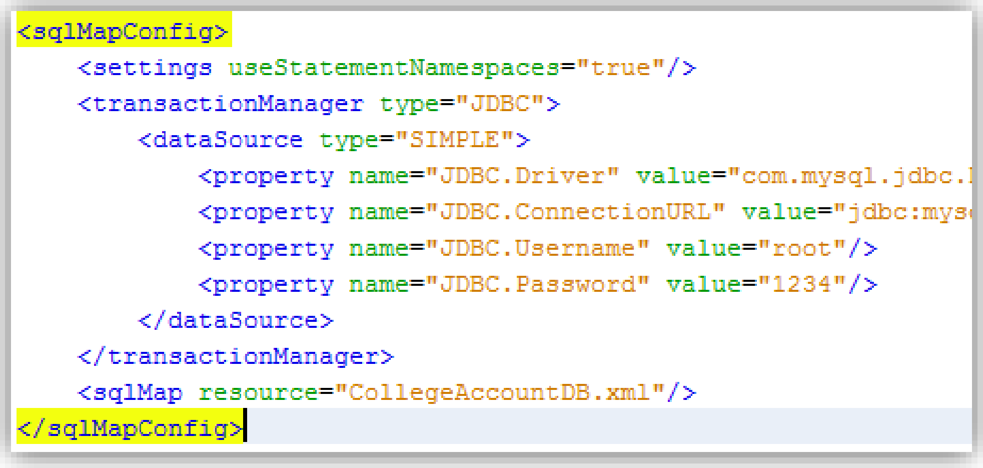
Please solve the puzzle

5 + 47 =

Wrong password validation

Figure 42: Validation Example

- ✓ *Use of Transaction management and ORM (Hibernate)* – this system is designed with the help of Hibernate which is an Object relation mapping tool which used to map each table with a class file. Moreover system also implement transaction management, code snippet of which is as follows:



```
<sqlMapConfig>
    <settings useStatementNamespaces="true"/>
    <transactionManager type="JDBC">
        <dataSource type="SIMPLE">
            <property name="JDBC.Driver" value="com.mysql.jdbc.
            <property name="JDBC.ConnectionURL" value="jdbc:mys
            <property name="JDBC.Username" value="root"/>
            <property name="JDBC.Password" value="1234"/>
        </dataSource>
    </transactionManager>
    <sqlMap resource="CollegeAccountDB.xml"/>
</sqlMapConfig>
```

Figure 43: Code Snippet - Transaction Management

- ✓ *Multithreaded Programming:* the Developer has used multithreading for sending email and SMS so that system has not to wait for an email or SMS to be sent and it can perform other task simultaneously. If it were not used then system may hang in case of slow internet connection as the system has to wait for email process to finish but now with multithreaded programming a separate thread (small process) is created to handle this task.

7.7 Conclusion

The system design was successfully implemented. Now, the next step is to conduct testing of the implemented system.

8. Testing

Testing is an integral part of software development and is deemed as an activity of evaluating the capability of a software in terms of its usability, maintainability, reliability, performance and meeting user requirements. It consumes more than half of the development time and is deployed in every phase of software development cycle.

(Hetzel, 1988).

8.1 Test Plan

The various types of testing which can be performed on the system are: Unit Testing, Integration Testing, System Testing and User-Acceptance testing.

8.1.1 Types of Testing

ID	Type	Application	Start	End
T01	**Unit Testing**	Unit testing is further classified into White and Black box testing. Each functionality is tested in an isolated environment.	25 March, 2014	05 April, 2014
T02	**Integration Testing**	The interaction between modules is checked in this type of testing. Modules are combined and tested on.	06 April, 2014	08 April, 2014
T03	**System Testing**	The complete system is tested for its compatibility and usability. Also, the user manual and system GUI and documentation is tested.	08 April, 2014	14 April, 2014
T04	**User Acceptance Testing**	The system was given for testing to the target users and their feedback was collected.	15 April, 2014	19 April, 2014
Total Duration : 4 weeks				

Table 34: Testing types & Duration

8.1.2 Success Criteria

The system should be able to run without any bugs/errors and should clearly match and justify the requirements and design specifications. The system will be deemed a pass only if:

- ✓ The salaries, fees and fines are being calculated properly.
- ✓ Reports are being generated properly.
- ✓ The user interface is consistent and following the HCI guidelines.
- ✓ The level of performance of the system is within permissible limits.

8.13 Users involved in Testing

Testing Type	Tester Name
Unit Testing	Developer
Integration Testing	Developer
System Testing	Mr Prince Kumar Mr Bijayant K. Jha Mr Tanish G. Mehta Mr Rakesh R. Pawar
User Acceptance Testing	Mr Gokul Gopinath [Accountant] Mr Prashant Kumar

Table 35: Users involved in Testing

8.1.4 Hardware/software Requirements for Testing

The minimum hardware and software requirements for the system would be:

Hardware	Desktop PC/Laptop : 2 to 3 with Internet Connectivity
Software	Windows XP/7/8, Linux, JDK 1.7, NetBeans IDE 7.4, Web Browser

Table 36: Hardware/Software for Testing

8.1.5 Test Case Template

<table>
<tr><th colspan="5"><b>Project Title :</b> Name of the project</th></tr>
<tr><td colspan="2">Test Case Name</td><td colspan="3">Name of test case</td></tr>
<tr><td colspan="2">Test Case ID</td><td colspan="3">ID of the test case conducted</td></tr>
<tr><td colspan="2">Conducted By</td><td colspan="3">Test case conducted by</td></tr>
<tr><td colspan="2">Description</td><td colspan="3">Test case description</td></tr>
<tr><th colspan="5">MODULE EXECUTION</th></tr>
<tr><th>S.No.</th><th>Steps</th><th>Result Expected</th><th>Actual output</th><th>Result</th></tr>
<tr><td>1.</td><td>Steps executed by tested</td><td>Expected result for the steps undertaken</td><td>Output from the system to the steps conducted</td><td>If the expected result matches actual result</td></tr>
<tr><td colspan="5"><b>Corrective Measure</b>: The measures taken by the developer to match the expected result with actual result</td></tr>
<tr><td colspan="5"><b>Conclusion:</b> After the corrective measures, if the module performed to expectations or not.</td></tr>
</table>

Table 37: Test Case Template

8.2 Test Cases

8.2.1 Unit Testing

8.2.1.1 Black Box Testing

Project Title : College Accountant	
Test Case Name	Login
Test Case ID	**T01- 01**
Conducted By	Developer
Description	Performing the test on the Login module for Accountant

MODULE EXECUTION

S.No.	Steps	Result Expected	Actual output	Result
1.1	Start the system	Login page for the system to be shown	Identical to expected result	Pass
1.2	Insert correct Login credentials	If credentials has been input correctly along with the captcha code, then accountant main page to be shown	Identical to expected result	Pass
1.3	Insert wrong Login credentials [attempt:1]	The system should prompt that wrong credentials has been input	Identical to expected result	Pass
1.4	Insert wrong Login credentials [attempt:2]	The system should prompt that wrong credentials has been input	Identical to expected result	Pass
1.5	Insert wrong Login credentials [attempt:3]	The system should prompt that wrong credentials has been input	Identical to expected result	Pass
1.6	Insert wrong Login credentials [attempt:4]	The user should be blocked. And further attempts to login would be withheld	System prompted that the login details were wrong and enter the details again	Fail
Corrective Measure [1.6]: The developer reviewed the code and found out that the condition of flag to check the wrong attempts was logically incorrect. It was flag>3 whereas it should be flag<=3.				
1.6	Insert wrong Login credentials [attempt:4]	The user should be blocked. And further attempts to login would be withheld	User was blocked and further logins withheld from that user	Pass
Conclusion: After the corrective measures, the module performed to expectations.				

Table 38: Unit Testing - Login

<table>
<tr><th colspan="5">Project Title : College Accountant</th></tr>
<tr><td colspan="2">Test Case Name</td><td colspan="3">Payroll Management</td></tr>
<tr><td colspan="2">Test Case ID</td><td colspan="3"><b>T01- 02</b></td></tr>
<tr><td colspan="2">Conducted By</td><td colspan="3">Developer</td></tr>
<tr><td colspan="2">Description</td><td colspan="3">Performing the test on the Payroll Management for Accountant</td></tr>
<tr><th colspan="5">MODULE EXECUTION</th></tr>
<tr><th>S.No.</th><th>Steps</th><th>Result Expected</th><th>Actual output</th><th>Result</th></tr>
<tr><td>2.1</td><td>Select the Payroll Module</td><td>Payroll Management form should be shown by the system</td><td>Identical to expected result</td><td>Pass</td></tr>
<tr><td>2.2</td><td>Display the Staff ID’s along with the Payroll details</td><td>A table of payroll details should be shown</td><td>Identical to expected result</td><td>Pass</td></tr>
<tr><td>2.3</td><td>Check the calculated salary for Staff</td><td>Correct payroll calculation should be shown</td><td>Identical to expected result</td><td>Pass</td></tr>
<tr><td colspan="5"><b>Conclusion:</b> The module performed to expectations, without any errors.</td></tr>
</table>

Table 39: Unit Testing – Payroll

Project Title : College Accountant				
Test Case Name	Donation			
Test Case ID	**T01- 03**			
Conducted By	Developer			
Description	Performing the test on the Donation module for Accountant			
MODULE EXECUTION				
S.No.	**Steps**	**Result Expected**	**Actual output**	**Result**
3.1	Select the Donation	Donation Module form should be shown by the system	Identical to expected result	Pass
3.2	Display the previous donation details	A table of Donation details should be shown	Identical to expected result	Pass
3.3	Insert the following information: Amount:asd Cheque No: sddfs Donor Name:ABC	The system should not allow insertion of String values in place of integer/double values	Identical to expected result	Pass
3.4	Insert the following information: Amount:20000 Cheque No: 587964 Donor Name:DEF	The data was successfully inserted	Identical to expected result	Pass
Conclusion: The module performed to expectations, without any errors.				

Table 40: U Unit Testing: Donation

Project Title : College Accountant	
Test Case Name	Hostel
Test Case ID	**T01- 04**
Conducted By	Developer
Description	Performing the test on the Hostel module for Accountant

MODULE EXECUTION

S.No.	Steps	Result Expected	Actual output	Result
4.1	Select the Hostel Module	Hostel Module form should be shown by the system	Identical to expected result	Pass
4.2	Display the previous Hostel details	A table of Hostel details should be shown	Identical to expected result	Pass
4.3	Select a row of details	The system should display the details of the row into corresponding textboxes	Identical to expected result	Pass
4.4	Update the information of Fee : abc	The system should not expect String value in place of double value	Identical to expected result	Pass
4.5	Update the information of Fee : 2000	The system should update the fee value	Identical to expected result	Pass
4.6	Click refresh button	Clear the textboxes	Identical to expected result	Pass

Conclusion: The module performed to expectations, without any errors.

Table 41: Unit Testing – Hostel

Project Title : College Accountant				
Test Case Name	Bus			
Test Case ID	**T01- 05**			
Conducted By	Developer			
Description	Performing the test on the Bus module for Accountant			
MODULE EXECUTION				
S.No.	**Steps**	**Result Expected**	**Actual output**	**Result**
5.1	Select the Bus Module	Bus Module form should be shown by the system	Identical to expected result	Pass
5.2	Display the previous Bus record details	A table of Bus details should be shown	Identical to expected result	Pass
5.3	Select a row of details	The system should display the details of the row into corresponding textboxes	Identical to expected result	Pass
5.4	Update the information of Bus route : Model Town	The system should update the bus route value	Identical to expected result	Pass
5.5	Click refresh button	Clear the textboxes	Identical to expected result	Pass
Conclusion: The module performed to expectations, without any errors.				

Table 42: Unit Testing – Bus

Project Title : College Accountant				
Test Case Name	Store			
Test Case ID	**T01- 06**			
Conducted By	Developer			
Description	Performing the test on the Store module for Accountant			
MODULE EXECUTION				
S.No.	**Steps**	**Result Expected**	**Actual output**	**Result**
6.1	Select the Store Module	Store Module form should be shown by the system	Identical to expected result	Pass
6.2	Display the previous Store record details	A table of Store details should be shown	Identical to expected result	Pass
6.3	Select a row of details	The system should display the details of the row into corresponding textboxes	Identical to expected result	Pass
6.4	Update the information of Quantity : abc	The system should not accept String values in place of double values	Identical to expected result	Pass
6.5	Update the information of Quantity : 5	The system should update the quantity	Identical to expected result	Pass
6.6	Upload Purchase form	The system should upload the purchase form	No file uploaded	Fail
Corrective Measure [6.6]: The developer reviewed the code and found out that the filename was not being renamed with new filename i.e. *txt_OrderFile.setText(filename);*				
6.6	Upload Purchase form	The system should upload the purchase form	Identical to expected result	Pass
Conclusion: After corrective measures taken, the module performed to expectations, without any errors.				

Table 43: Unit Testing- Store Module

8.2.1.2 White Box Testing

Contrary to black-box testing, software is viewed as a white-box, or glass-box in white-box testing, as the structure and flow of the software under test are visible to the tester. Testing plans are made according to the details of the software implementation, such as programming language, logic, and styles. Test cases are derived from the program structure. White-box testing is also called glass-box testing, logic-driven testing. (Myers, 1979)

The developer used control flow chart shown below to conduct white box testing along with code:

```
/*A*/ if((LoginUtils.getLoginInfo(textUserName.getText().trim())).size()>0)
{
loginInfo = LoginUtils.getLoginInfo(textUserName.getText().trim());
/*B*/  userName = loginInfo.get(0);
System.err.println( new Date().toString()+":LoginScreen: "+"userName: "+userName);
/*C*/  password = loginInfo.get(1);
System.err.println( new Date().toString()+":LoginScreen: "+"password: "+password);
accountFlag = Integer.parseInt(loginInfo.get(2));
System.err.println( new Date().toString()+":LoginScreen: "+"accountFlag: "+accountFlag);
}
/*Q*/ else
{
labelWarning.setText("Account cannot be found.");
}
/*E*/  if(accountFlag == 0)
{
/*F*/  if(textUserName.getText().trim().length()>0 && textPassword.getPassword().length>0
&& textCaptchaAns.getText().trim().length()>0)
{
/*G*/  if(textUserName.getText().equalsIgnoreCase(userName) &&
textPassword.getText().equalsIgnoreCase(password))
{
try
{
if(textCaptchaAns.getText().trim().length()>0)
{
/*D*/  intCaptchaAns = Integer.parseInt(textCaptchaAns.getText());
/*H*/  if((intCaptchaAns!=captchaVar))
{
this.callCaptcha(2);
}
else
{
/*I*/  mainscreen.FrameMainMenu mainScreen = new mainscreen.FrameMainMenu();
mainScreen.setSize(600, 400);
mainScreen.setVisible(true);
this.dispose();
}
}
}
catch(NumberFormatException e)
{
this.callCaptcha(4);
}

}
else
{
/*J*/ if(count!=1)
{
count--;
labelWarning.setText("You have Entered wrong password. You have "+count+" tries left.");
this.callCaptcha(3);
}
/*K*/ else
{
labelWarning.setText("Your account has been De-Activated.");

}
}
}
else
{
/*L*/ if((textUserName.getText().trim().length()==0) &&
(textPassword.getPassword().length==0) && (textCaptchaAns.getText().trim().length()==0))
{
textUserName.setBackground(Color.red);
textPassword.setBackground(Color.red);
textCaptchaAns.setBackground(Color.red);
labelWarning.setText("Please fill the details below.");
}
/*M*/ else if((textUserName.getText().trim().length()==0))
{
textUserName.setBackground(Color.white);
textPassword.setBackground(Color.white);
textCaptchaAns.setBackground(Color.white);
labelWarning.setText("Please enter User Name.");
}
/*N*/ else if((textPassword.getPassword().length==0))
{
textUserName.setBackground(Color.white);
textPassword.setBackground(Color.red);
textCaptchaAns.setBackground(Color.white);
labelWarning.setText("Please enter Password.");
}
/*O*/ else if((textCaptchaAns.getText().trim().length()==0))
{
textUserName.setBackground(Color.white);
textPassword.setBackground(Color.white);
```

```
textCaptchaAns.setBackground(Color.red);
labelWarning.setText("Please solve the puzzle.");
}
}
}
/*P*/ else
{
labelWarning.setText("Your account is already Disabled.");
}
}
```

In code each node is labelled for simplicity in describing the paths.

A set of seven (7) basic paths can be followed which are:

- ✓ ABCDEFGHI
- ✓ ABCDELGHI
- ✓ ABCDEMGHI
- ✓ ABCDNGHI
- ✓ ABCDOGHI
- ✓ ABCDEFGHJHI
- ✓ ABCDEFGKLHI

The cyclomatic complexity of this diagram is (Edges-nodes+2): 27-18+2 = **11**

This basic path is now implemented as test cases:

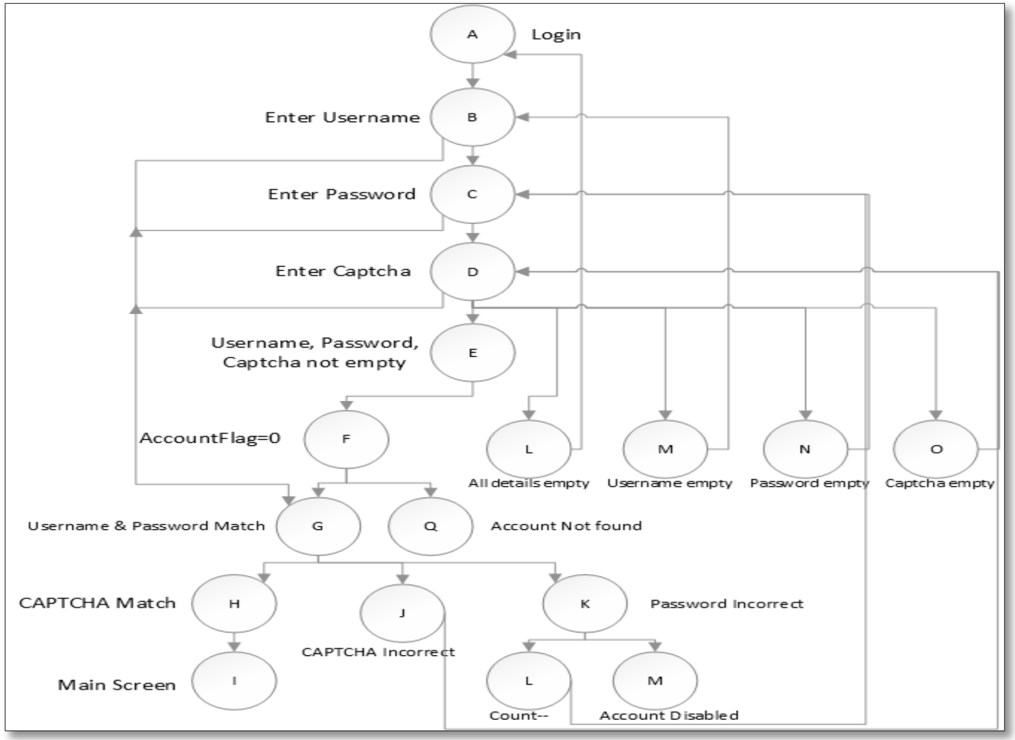


Figure 44: White Box Testing

Table 44: White Box Testing

ID	Username (B)	Password (C)	CAPTCHA (D)	Condition1 (L)	Condition2 (M)	Condition3 (N)	Condition4 (O)	Decision	Result
1.1	<empty>	<empty>	<empty>	True	False	False	False	All textbox red	Pass
1.2	admin	<empty>	46	False	False	True	False	Password textbox red, rest white	Pass
1.3	Admin	1234	<empty>	False	False	False	True	Captcha textbox red, rest white	Pass
1.4	<empty>	1234	214	False	True	False	False	Username textbox red, rest white	Pass
1.5	Admin	1234	79	False	False	False	False	All textbox white	Pass

8.2.2 Integration Testing

In this type of testing, various modules are clubbed together and checked for their interoperability. There are various ways in which integration testing can be done and the developer has chosen the Top-Down Integration testing approach.

Project Title : College Accountant				
Test Case Name	Integration Testing			
Test Case ID	**T02- 01**			
Conducted By	Developer			
Description	Performing the test on Library + Student + Email + SMS			
MODULE EXECUTION				
S.No.	**Steps**	**Result Expected**	**Actual output**	**Result**
1.1	Accountant logins into the system	The system shows the home page if the credentials entered are correct.	Identical to expected result	Pass
1.2	Library Module is selected	The system should show the library module form	Identical to expected result	Pass
1.4	Update the record of Student with student id “ST01’ , and include library fine = 100	The system should update the information	Identical to expected result	Pass
1.5	Student Fee/Fine module is selected	The system should show the Student fee/fine module	Identical to expected result	Pass
1.6	The update done in Library module should be shown in Student module	The system should reflect the same update here	Identical to expected result	Pass
1.7	Send reminders [email/SMS]	The system should send email and SMS notification	Identical to expected result	Pass
Conclusion: The module performed to expectations, without any errors.				

Table 45: Integration Testing

<table>
<tr><th colspan="5">Project Title : College Accountant</th></tr>
<tr><td colspan="2">Test Case Name</td><td colspan="3">Integration Testing</td></tr>
<tr><td colspan="2">Test Case ID</td><td colspan="3"><b>T02- 02</b></td></tr>
<tr><td colspan="2">Conducted By</td><td colspan="3">Developer</td></tr>
<tr><td colspan="2">Description</td><td colspan="3">Performing the test on Student fee + Payment gateway</td></tr>
<tr><th colspan="5">MODULE EXECUTION</th></tr>
<tr><th>S.No.</th><th>Steps</th><th>Result Expected</th><th>Actual output</th><th>Result</th></tr>
<tr><td>2.1</td><td>Accountant logins into the system</td><td>The system shows the home page if the credentials entered are correct.</td><td>Identical to expected result</td><td>Pass</td></tr>
<tr><td>2.2</td><td>Report module -> Defaulter list is generated</td><td>The system should generate a defaulter list</td><td>Identical to expected result</td><td>Pass</td></tr>
<tr><td>2.3</td><td>Student logins into the web-part of the system</td><td>The system shows the home page if the credentials entered are correct</td><td>Identical to expected result</td><td>Pass</td></tr>
<tr><td>2.4</td><td>Student pays the fees/fines using Payment Gateway</td><td>The system should be able to receive the payment</td><td>Payment was done in $, rest as expected</td><td>Partial Pass</td></tr>
<tr><td>2.5</td><td>Student's name should be removed from defaulter's list</td><td>The system removed the name of student from defaulters list</td><td>Identical to expected result</td><td>Pass</td></tr>
<tr><td colspan="5"><b>Conclusion:</b> The module performed to expectations, without any errors.</td></tr>
</table>

Table 46: Integration Testing

8.2.3 System Testing

In this type of testing, a series of tests is conducted on the completed and functional system to check its compliance with the before mentioned requirements.

Project Title : College Accountant	
Test Case Name	System Testing
Test Case ID	**T03- 01**
Conducted By	Developer
Description	To check if all Modules are integrated properly and working seamlessly

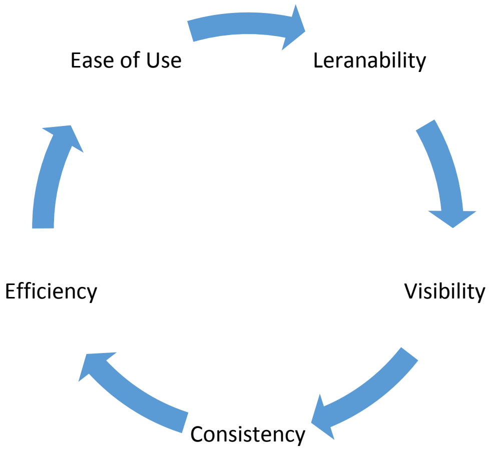
MODULE EXECUTION					
S.No.	**Steps**	**Result Expected**	**Actual output**	**Result**	**Corrective Measures**
1.	Start the system	Login page for the system to be shown	Identical to expected result	Pass	Not Required
2.	Accountant Login-> Username, Password and CAPTCHA code	If the credentials entered are right, then system should re-direct to home page	Identical to expected result	Pass	Not Required
3.	Accountant Payroll Module View details-> Calculate payroll-> Update details-> generate pay slip	The system should show the payroll details, calculate payroll, update details and generate pay slip	Identical to expected result	Pass	Not Required
4.	Accountant Store module View details-> Update details->Upload Purchase form	The system should show store details, update details and upload purchase form	Identical to expected result	Pass	Not Required
5.	Accountant Library Module View details-> Update details -> Calculate fine	The system should show library details, update details and calculate fine	Identical to expected result	Pass	Not Required

6.	Accountant Hostel Module View details-> Update details	The system should show hostel details, update details	Identical to expected result	Pass	Not Required
7.	Accountant Bus Module View details-> Update details	The system should show bus details, update details	Identical to expected result	Pass	Not Required
8.	Accountant Examination Module View details-> Update details -> Calculate fine	The system should show examination details, update details and calculate fine	Identical to expected result	Pass	Not Required
9.	Accountant Student Fee Module View details-> Update details -> View fine ->Send reminder [Email/SMS]	The system should show Student fee/fine details, update details and send reminders [email/SMS]	Identical to expected result	Pass	Not Required
10.	Accountant Report Module View reports	The system should show graphical and textual reports	Identical to expected result	Partial Pass	Not Required
11.	Student Login to Website	If the credentials entered are right, then system should re-direct to home page	Identical to expected result	Pass	Not Required
12.	Student View Fee/Fine Pay Fee/fine	The system should show the applicable fee/fine on the student and redirect them to Payment gateway	Identical to expected result	Pass	Not Required
Conclusion: After the corrective measures, the module performed to expectations.					

Table 47: System Testing

8.2.3.1 Usability Testing

In this type of testing, the system is tested by some specific users and the system is tested on some specific goals to check effectiveness, efficiency and satisfaction in a specific context. Usability is tested on the following parameters:



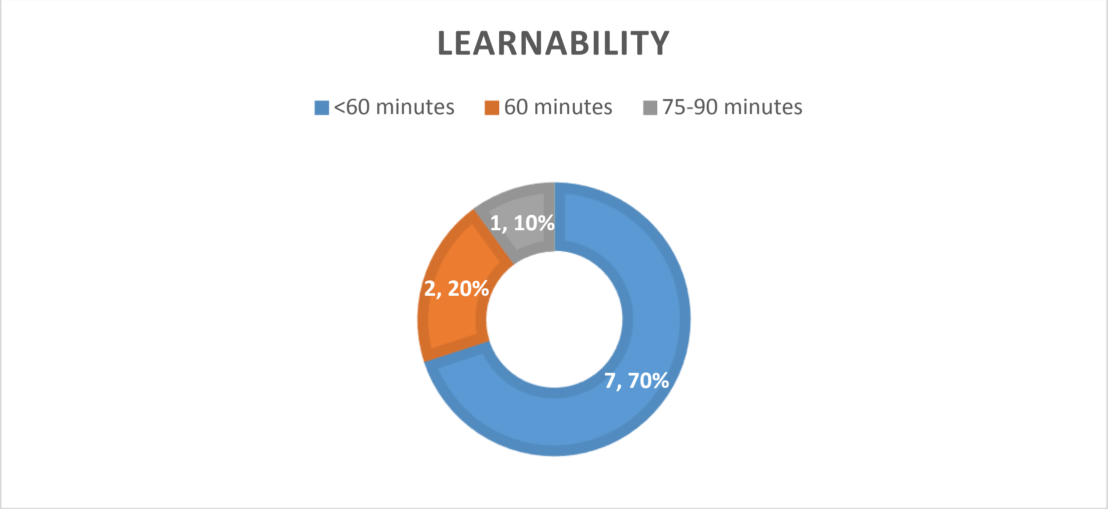
<table>
<tr><th colspan="5">Questionnaire To perform Usability Testing</th></tr>
<tr><td colspan="5"><b>Q1:</b> How much time do you think you would require in learning the functionality of this system?</td></tr>
<tr><td>□ &lt; 60 minutes</td><td>□ 60 minutes</td><td>□ 60-75 minutes</td><td>□ 75-90 minutes</td><td>□ whole day</td></tr>
<tr><td colspan="5"><b>Q2:</b> How is the visibility of the various functionalities provided in this system?</td></tr>
<tr><td colspan="5">□ Yes □ No □ Can’t Say</td></tr>
<tr><td colspan="5"><b>Q3</b>: How do you find the consistency of the pages of the system?</td></tr>
<tr><td>□ Poor</td><td>□ Average</td><td>□ Good</td><td colspan="2">□ Excellent</td></tr>
<tr><td colspan="5"><b>Q4:</b> How easy was the system to use?</td></tr>
<tr><td>□ Difficult</td><td>□ Average</td><td>□ Easy</td><td colspan="2">□ Very easy</td></tr>
<tr><td colspan="5"><b>Q5:</b> Did you find the system efficient?</td></tr>
<tr><td>□ Not efficient</td><td>□ Average</td><td>□ Efficient</td><td colspan="2">□ Very efficient</td></tr>
</table>

Table 48: Usability Testing

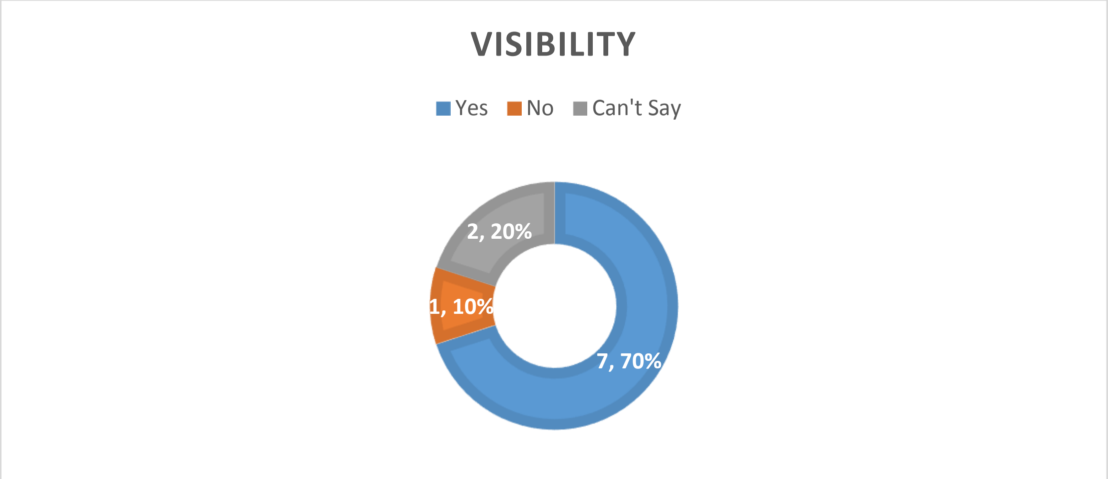
This questionnaire was distributed among ten people and these were the findings:

[*Refer Appendix D for filled forms*]

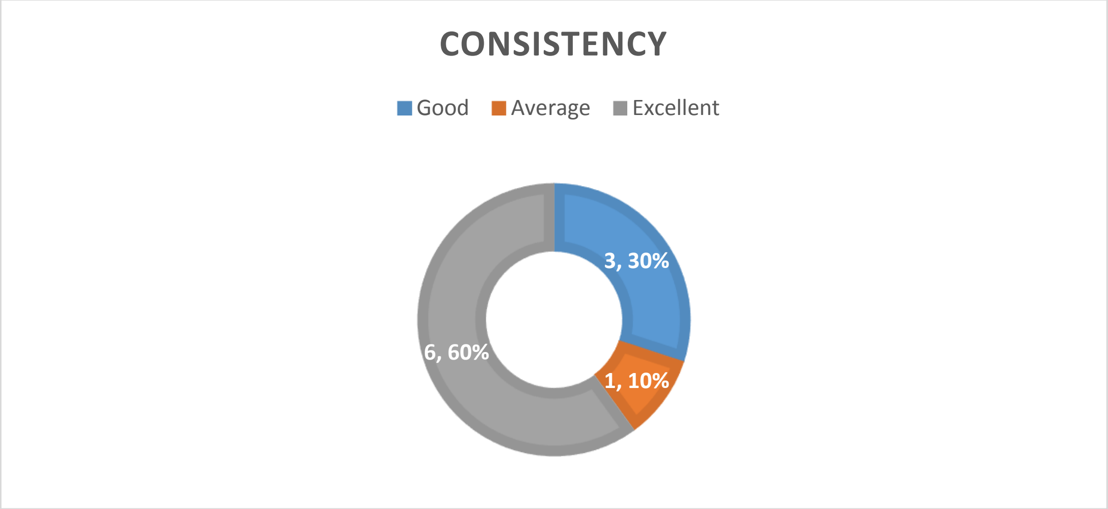
Usability Testing Analysis
Q1: How much time do you think you would require in learning the functionality of this system?



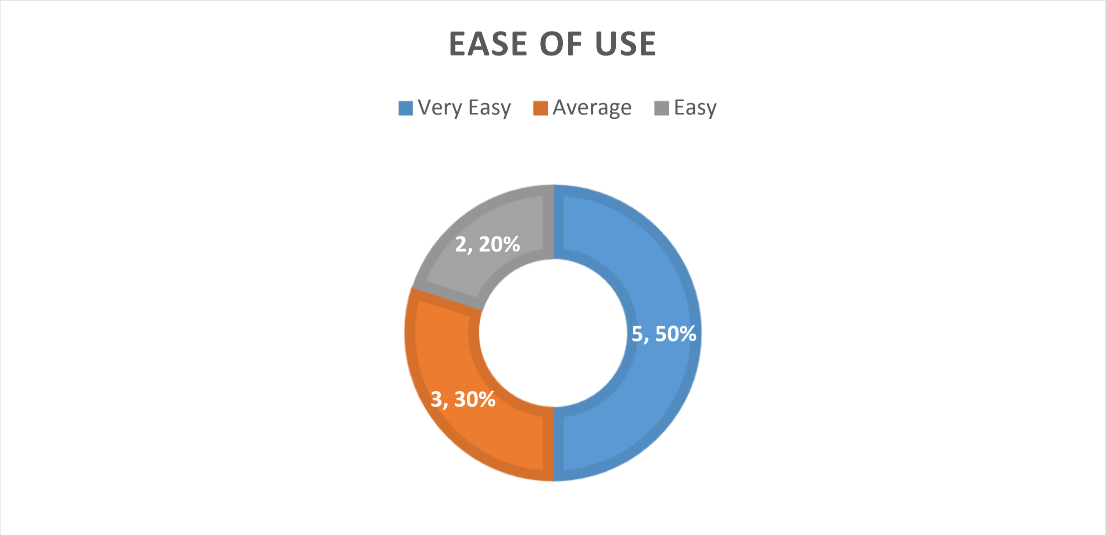
Q2: How is the visibility of the various functionalities provided in this system?



Q3: How do you find the consistency of the pages of the system?



Q4: How easy was the system to use?



Q5: Did you find the system efficient?

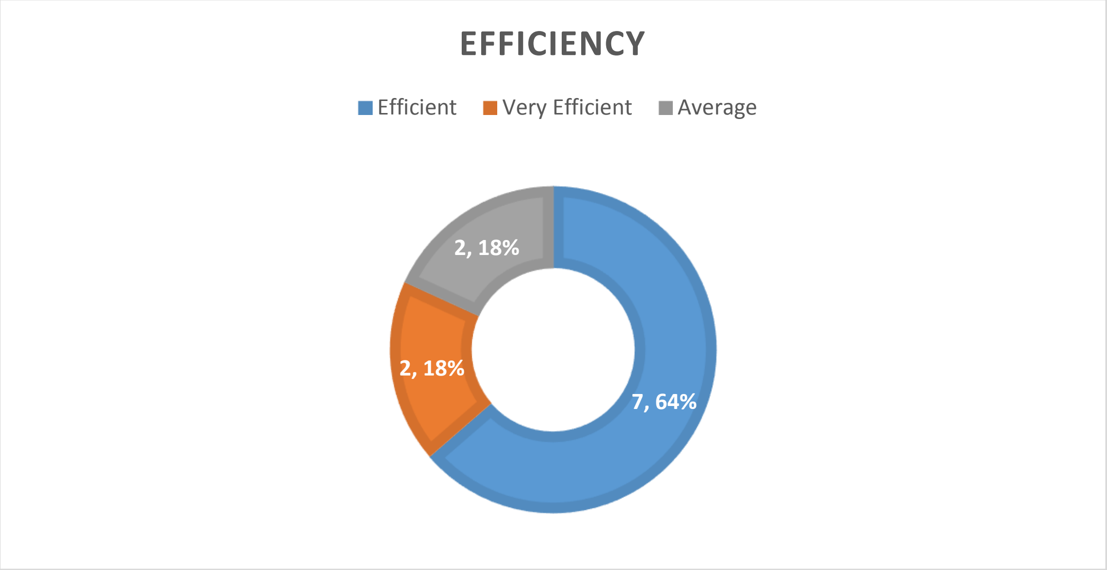


Table 49: Usability Testing Analysis

Conclusion: From the above analysis it is evident that more than 87 per cent of the volunteers found the system usable according to the before mentioned parameters. Hence, the system can be deemed usable. Analysis of usability is positive and the developer has found that everything is in place and in accordance with user review which was taken in prior during analysis part.

8.2.3.2 Compatibility Testing

In this type of testing, the system is installed and run on different platforms and hardware configurations to check its compatibility with it. The desktop application should be able to work on popular OS like Windows XP/7/8 and the web application should work with popular web browsers like Mozilla Firefox, Microsoft Internet Explorer, Google Chrome, Apple Safari, Opera in different screen resolutions like 15.4 (1024*768), 15'(800*600) and 17' (1200*768).

	Test Method	Expected Result	Actual Result	Remarks
For Desktop Application	Hardware Test 1: Core 2 Duo, Hard Disk(160 GB), 512MB RAM, Internet Connectivity	System should run smoothly without any issues	System was running smoothly without any issues	Pass
	Platform Test 1: Window XP JDK1.5	System should run smoothly without any issues	System failed to start	Fail : JDK1.7 need to be installed
	Platform Test 2: Window XP JDK1.7	System should run smoothly without any issues	System was running smoothly without any issues	Pass
	Platform Test 3: Window 7 JDK1.7	System should run smoothly without any issues	System was running smoothly without any issues	Pass
For Web Application	Hardware Test 1: Core 2 Duo, Hard Disk(160 GB), 512MB RAM, Internet Connectivity	System should run smoothly without any issues	System was running smoothly without any issues	Pass
	Platform Test 1 : Google Chrome	System should run smoothly without any issues	System was running smoothly without any issues	Pass
	Platform Test 1 : Mozilla Firefox	System should run smoothly without any issues	System was running smoothly without any issues	Pass
Conclusion: The system passed in almost all the hardware and platform parameters and hence passes in Compatibility testing.				

Table 50: Compatibility Testing

8.2.3.3 Documentation Testing

In this type of testing, the user manual, technical manual and the system documentation is tested on three parameters:

- ✓ Completeness: which can be verified with the Final Year Documentation specification provided by the Project manager, APIIT SD India.
- ✓ Correctness: which is verified by the spell check and use of correct grammar
- ✓ Understandable: which is verified by checking if the written sentence is trying to convey the same message which the developer is thinking.

Parameters	Mr. Prince	Mr. Bijayant	Mr. Tanish	Mr. Rakesh
Completeness	95%	92%	93%	87%
Correct	89%	91%	89%	87%
Understandable	97%	95%	94%	92%
Conclusion: With average hovering around 90%, it can be said that the document was tested successfully and was deemed to be of high standards.				

Table 51: Document Testing

8.2.3.4 GUI Testing

In this type of testing the systems user interface is checked on certain parameters.

Project Title : College Accountant		
Test Case Name	GUI Testing	
Test Case ID	**T03/GUI**	
Conducted By	Developer	
Description	Test the system's user interface	
Criteria	**Questions**	**Answers**
Buttons	Do the buttons have meaningful labels?	**Yes**
	Do they describe the appropriate action?	**Yes**
	Do the names given to the labels are consistent all the screens?	**Yes**
	Is the grouping of buttons is appropriate.	**N/A**
	Are the buttons size consistent (width and height).	**Yes**
Textbox	Do the maximum length of the text box matches their corresponding data storage sizes?	**Yes**
	Do the check constraints available	**Yes**
Menu Bars	Do the menu labels are meaningful?	**Yes**
	Do they describe their associated	**Yes**
Screen Design	Is the layout logical so that the user does not have to search for typical functions?	**Yes**
	Are graphics and text arranged on the screens in such a way that they are easy to view and are not clustered?	**Yes**
	Is the used text providing meaningful information	**Yes**
Colour	Are colours used consistently when designating functionality?	**No**
	Do the used colours are sufficient contrast to reduce eye strain?	**Yes**
Form Labels	Do the label sizes and lengths are appropriate.	**Yes**

	Are the labels given proper text so that they describe properly what they meant?	**Yes**
	Are the form labels" lengths sufficient to accommodate common screen solution?	**Yes**
Fonts	Are fonts consistent to all modules	**Yes**
	Are the fonts used are available are available in all types of operating system?	**Yes**
Conclusion: The system's GUI was found to be user-friendly and the users won't have any difficulty in accessing this system		

Table 52: GUI Testing

8.2.4 User Acceptance Testing

Project Title : College Accountant	
Test Case Name	User Acceptance Testing
Test Case ID	**T04-01**
Conducted By	Mr Sanjeev Jawa [Accountant]
Description	Test in real business environment.

Test	Status	Excellence (%)
Able to Login	Yes	**100**
Able to check Payroll Module	Yes	**96**
Able to check Store Module	Yes	**89**
Able to check Student fee/Fine module	Yes	**92**
Able to check Hostel Module	Yes	**89**
Able to check Bus Module	Yes	**93**
Able to check email and SMS notifications	Yes	**91**
Able to generate reports	Yes	**95**
Conclusion: The system was accepted by the user.		

Table 53: User Acceptance Testing 1

Project Title : College Accountant		
Test Case Name	User Acceptance Testing	
Test Case ID	**T04-02**	
Conducted By	Mr Prashant Kumar [Student]	
Description	Test in real business environment.	
Test	**Status**	**Excellence (%)**
Able to Login	Yes	**100**
Able to check Fees applicable	Yes	**95**
Able to check Fines applicable	Yes	**95**
Able to do online payment [mock]	Yes	**94**
Conclusion: The system was accepted by the user.		

Table 54: User Acceptance Testing 2

8.3 Conclusion

Testing proves to be the most important task contributing to the success of a project. For this project, “College Accountant”, testing was carried out side-by-side of the Implementation phase. Various types of testing was conducted to check for and resolve errors, to check whether the system developed is in accordance to the requirements laid down by the user or not.

The developer had conducted Unit testing which proved beneficial in the identification and rectification of errors. It involved two kinds of testing: Black-box and White-box testing. Many black-box testing were conducted, however only the important ones have been mentioned in this documentation. White-box testing was done on the details of software implementation such as programming language, logic and styles.

Integration Testing helped the developer to check the inter-operability between the different modules of the developed system. Some problems which were creating mis-communication between modules was rectified accordingly.

Lastly, System testing was conducted to check the accuracy, completeness and performance of the functions designed. This was done with the help of Usability, Compatibility and Documentation testing. On successfully completing this testing, it ensured the developer that the developed software is usable and fit for launch in the market.

9. Critical Evaluation

9.1 Introduction

Critical evaluation is the process of assessing the relative merit of a piece of work, which may have been presented as a journal article, in a text-book or an academic project. After completion of the various phases of project development it is utmost important that the developer does the critical evaluation of the complete project module-wise.

So, on the completion of this project, it is evaluated for the benefits it gives to the targeted end users.

9.2 Factors of Benefit

In Factor of Benefits we assess how the developed system is useful for the target users.

- ✓ Better management of the resources. Resources are easily accessible.
- ✓ There is reduction in the amount of labour which needs to be put in accounts book-keeping.
- ✓ Generation of graphical repots has helped in easily assessing the financial condition of the college/institute.
- ✓ It has helped in better time-saving. Students need not waste their time queuing up in lines to pay fee/fines and they can do it with the help of the gateway.
- ✓ Increase in productivity of the accountant. Fee/fine/salary calculations, reports generation happens at the click of a button
- ✓ Email and SMS notification/reminders have been provided which increases the customer response to the system.

9.3 Comparison with similar systems

Systems	Functionalities									
	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Tally ERP 9.0	✓	✓	-	✓	-	-	-	-	✓	✓
Microsoft Money	✓	✓	-	✓	-	✓	✓	✓	✓	-
Intuit QuickBooks	✓	✓	-	✓	✓	✓	✓	✓	✓	✓

Table 55: Comparison with similar systems

ID	Functionality	ID	Functionality
F1	Secure login	F6	Hostel/Mess expenses
F2	Payroll calculation	F7	Bills tracking
F3	Payslip generation	F8	SMS Notification
F4	Store calculations	F9	Email notification
F5	Bus expense	F10	Graphical Report generation

Table 56: Comparison with similar systems

9.4 Degree of Success

This project would be deemed successful only if it adheres to the requirements and the developed system is fulfilling all the functional requirement which have been stated in the software requirement specifications prepared by the developed during the planning stage. This system College Accountant was able to rectify all the problems which has been mentioned in the problem description. Lastly, the developer can conclude that this project has been successfully completed with the following criteria:

Criteria	Description	Success
Project Objectives	Check if the project objectives have been achieved	8
Academic Objectives	Check if the project objectives have been achieved	9
Requirement fulfilment	Check if the user requirements have been fulfilled	8
User interaction	Take time-time feedback from users	7
Project Management	Check if the project followed the software development guidelines and was completed within time and budget.	7
Testing	Check if proper testing was conducted to eradicate errors/bugs	9
Graphical Interface with validations	Check the visibility of interface along with the validations	8
Overall Quality	Check the quality of the system and documentation	9
Total : 65/80 = 81.25%		

Scale	**1-3**	**4-5**	**6-7**	**8-9**	**10**
Rating	Poor	Fair	Good	Excellent	Outstanding

Table 57: Degree of Success

Conclusion: The overall success rate of 81.25 per cent thus proves that the system has been successfully developed to an extent along with proper documentation.

10. Conclusion

10.1 Limitations of the project

Though the developer has tried to implement all the functionalities which were mentioned in the software requirement specifications, however there could have been scope for adding more functionalities to the developed system which couldn’t be included due to time, legal, monetary constraints etc. Those functionalities would be:

- ✓ The desktop application needs to be connected with internet in order to be updated with the database updates. Since, the payment from the students’ needs to be updated in the database which is further updated in the desktop application.
- ✓ Proper integration with other modules could be improved.
- ✓ Live payment could have been implemented. But due to new government norms, non-registered web-sites couldn’t implement a live payment gateway.
- ✓ The payment gateway being used for this project is PayPal Sandbox and the transactions which are carried out in that account is in dollars only.
- ✓ The User interface could have been made more attractive and user-friendly.

10.2 Future Enhancements

Further to the functionalities provided in the developed system, some additional functionalities could have also been added which weren’t added due to time, budget and legal constraints. The functionalities which could be easily added up to this project would be:

- ✓ Development of a mobile application on different platforms like Android, IOS, Windows, Blackberry etc. in order to cover a larger audience.
- ✓ Linking of the system with the bank account for actual transaction updates. Also, direct delivery of salaries into the accounts of the employees.
- ✓ Live payment gateway: which would enable the students to actually pay up the applicable fees/fines.
- ✓ Full-Automation: currently, the system is partially automated. However in the near future, this system could be made fully automated. This can be done by using bar-code readers in store/inventory to track the in/out goods. Using, FID based attendance marking system for the students and staff.

10.3 Computational Challenge

The main computational challenge during the development of this system would be:

- ✓ Programming Language: this project is implemented in technologies like J2SE and J2EE along with using Hibernate in order to improve the efficiency and performance of the system.
- ✓ CAPTCHA: Completely Automated Public Turing test which helps identify the Computer and Humans apart. It was used to make the logins more secure. However, to implement Captcha was a very difficult task.
- ✓ Session: was used in J2SE application in order to maintain the privileges to the users i.e. which user can view/control what was done sing session.
- ✓ Report Generation: there are various kinds of reports available in java like jfreechart, jasper reports etc. Effective and productive reports had to be generated for the proper analysis of the budget in the college/institute.
- ✓ Linking desktop application and online payment gateway: for this purpose the database was hosted and being used by both the desktop and online application. Web services was used to link the database with the desktop application.
- ✓ Account Book-keeping: the developer had to invest a lot of time in learning the concepts of accounts book-keeping in order to perform right calculations in the system.

10.4 What If A Chance Will Be Given To Redo The Project?

- ✓ Graphical Interface: if given a chance to redo, the developer would try to make the interface more attractive and user-friendly by using Jquery, java scripts, Modena themes etc.
- ✓ Implementing SPRING Framework: though the developed project does follow the MVC architecture, but by using this framework the efficiency of the program could have been increased.

10.5 Learning Experience

10.5.1 Tools Techniques & Technologies

- ✓ The developer used tools like questionnaire, interviews, focus group studies, etc. to collect data for research purposes.
- ✓ The developer made himself well versed with the concepts of J2SE and J2EE. He also got a chance to work on Net Beans 7.4, Eclipse, and MySQL which were required to develop, implement and test this system.
- ✓ Developer also tried to gain knowledge in Captcha, Web services, Report generation in java, encryption-decryption, payment gateway, email and SMS notifications, Session, MVC architecture etc.

10.5.2 Project Management

- ✓ The developer made a point that the development of this system strictly follows the Software development life cycle. The development methodology which best suits this project was chosen [Spiral Model]. From planning to research and implementation to testing, the developer had made sure that the requirements of the user are met to the fullest.
- ✓ The developer learned how to deliver time-sensitive projects on time. The project timeline was of roughly 34 weeks and the developer wrapped up the whole project within time while meeting almost all requirements.
- ✓ During the development of the project, the developer also learned how to identify and manage risks in very effective and efficient manner.

Concluding, the developer after the development of this project gained the ultimate confidence in developing a full-fledged system within the given resources with proper detailed documentation to accompany with the system.

Summing up, it could be said that this project could see the light at the end of the tunnel only because of the proper and precise guidance provided to the developer by *Ms. Priyanka Sachdeva* and *Ms .Geeta Nagpal*.

References

Bar graph. (n.d.). Retrieved from http://mste.illinois.edu/courses/ci330ms/youtsey/BAR2.GIF Last Accessed : 25 Feb 2014

Bharti, N. (2012, April). *Results from InfoQ 2012 User Survey*. Retrieved from InfoQ: http://www.infoq.com/articles/infoq-user-survey-results-2012 Las Last Accessed : 25 November 2013

Billows, D. (2013, April 4). *Bottom-up Estimating*. Retrieved from Project Manager Training and Certification: http://4pm.com/bottom-up-estimating/ Last Accessed : 18 October 2013

Client Server. (2003). Retrieved from Software Architecture: http://tutorials.jenkov.com/images/software-architecture/client-server-2.png Last Accessed : 28 October 2013

FlopppyDisk. (n.d.). Retrieved from WordPress: http://thefloppydisk.files.wordpress.com/2013/05/web20.png Last Accessed : 18 December 2013

Group, G. (2008). *Market Share*. Retrieved from MySQL: http://www.mysql.com/why-mysql/marketshare/ Last Accessed : 18 January 2014

J Preece, Y. R. (2002). Design Principles. In *Interaction Design: Beyond Human-Computer Interaction* (p. 21). New York: Wiley. Last Accessed : 28 October 2013

JavaMail. (2008). Retrieved from Oracle: https://javamail.java.net/nonav/docs/api/ Last Accessed : 10 March 2014

JavaMail API. (n.d.). Retrieved from http://publib.boulder.ibm.com/infocenter/iseries/v5r4/topic/rzamy/50/program/rzaiz523.gif Last Accessed : 1 March 2014

Line Graph. (n.d.). Retrieved from http://www.cdc.gov/nchs/data/nhis/earlyrelease/insur200909_Fig_06.png Last Accessed : 10 March 2014

Mathworksheetscenter. (2010). Retrieved from Charts: http://www.mathworksheetscenter.com/mathtips/barlinepiegraph.html Last Accessed : 10 February 2014

Munassar, A. G. (2010). Spiral Model. *A Comparison Between Five Models Of Software Engineering*, 98-99.

Myers, G. J. (1979). *The art of software testing.* New York: Wiley.

Phipps, G. (1999). *Java Compared with Other Languages*. Retrieved from http://thinklikecs.webs.com/articles/JavaVSOther.html Last Accessed : 15 March 2014

Pie Chart. (n.d.). Retrieved from http://www.jpowered.com/php-scripts/pie-chart/images/pie-chart-small.png Last Accessed : 5 February 2014

SMS Services. (2001). Retrieved from http://thesms.in/india-bulk-sms-services-delhi/image/Gateway.png Last Accessed : 10 March 2014

Spiral Model. (n.d.). Retrieved from http://www.sdlc.ws/wp-content/uploads/2011/12/Spiral-model-diagram.jpg Last Accessed : 10 March 2014

UML 2.0 in a Nutshell. (2005). In D. Pilone. CA,USA: O'Reilly.

Webdotdev. (2008). *Advantages of JAVA.* Retrieved from http://www.webdotdev.com/nvd/content/view/1042/204/ Last Accessed : 1 March 2014

Bibliography

Books

- ✓ Bates, B. and Sierra, K. (2003). *Head First Java*. O'Reilly Media.
- ✓ Schildt, H. (2011). *Java The Complete Reference*. Tata McGraw-Hill Publishing Company.
- ✓ Schwartz, B. (2008). *High Performance MySQL*. 2nd ed. O'Reilly Media.
- ✓ (2012). *Tally .ERP 9 in Simple Steps*. 3rd ed. Dreamtech Press.
- ✓ Biafore, B. (2012). *QuickBooks 2013*. O'Reilly Media.
- ✓ Gulati, M. (2009). *Accounting & Tally.ERP 9*. 2nd ed. Delhi: Siliconmedia.

White Papers/Journals

- ✓ Sosnoski, D.A. (2000). Managing collections. *Java performance programming*. (3), Available at: http://www.javaworld.com/article/2076908/build-ci-sdlc/java-performance-programming--part-3--managing-collections.html.
- ✓ Farber, B. (2014). Hibernate validation in a standalone implementation. *Java validations*, Available at: http://www.javaworld.com/article/2137346/data-storage/java-tip-hibernate-validation-in-a-standalone-implementation.html.
- ✓ Sellars, P. (2006). Java. *Reporting Made Easy with JasperReports and Hibernate*. 1, Available at: http://java.sys-con.com/node/171467.
- ✓ Hindle, A. (2008). k. *Reverse Engineering CAPTCHAs*. p59-68. Available at: http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4656374

Appendix

Appendix A

A1: Project Development Plan [Work Breakdown Structure]

Start date:	**13th September 2013**
End Date:	**03rd May 2014**
Duration:	**34 weeks**

✓ **Project Definition (Inception Phase) Duration: 2 weeks**

- Draft Proposal Submission

✓ **Project Planning (Inception and Elaboration Phase) Duration: 4 Weeks**

- Project Proposal Form Submission
- Gantt Chart and Project Development Plan [WBS]
- Other Project Management Activities.

✓ **Requirements Engineering (Elaboration and Construction Phase) Duration: 3 week**

- Identify Project Specifications
- Project Specification Form submission
- Data Gathering through Academic and Primary Research
- Data Gathering Through Primary Research
- Requirement Analysis

✓ **System Design (Construction phase) Duration: 7 weeks**

- Database Design
- Architectural Design
- Interface Design
- Navigational Design
- Design For Test Plan

✓ **Developing the System (Construction Phase) Duration: 12 weeks**

- End of Semester Viva.
- Creating Database and populating with sample Data.

- Developing User Interface and Populating with Contents.

- Implementation of various Project modules
- Integration Of modules into one Single System.

✓ **Testing and Evaluation (Construction Phase and Transition) Duration: 4 weeks**

- Revising Test Plan and Making Necessary Changes.
- Unit Testing Each Module as Described in the Test Plan.
- System Testing- Once the Complete System is ready.
- Critical Evaluation.
- Releasing the System for Beta Testing.
- Making Changes according to the Beta Test Findings.
- Optimizing the Entire System for Performance.

✓ **Project Ending (In Transition Phase) Duration: 2 weeks**

- Submission of the Final System.
- Presentation and Demonstration of Developed system.

A2: Gantt chart

<u>Please Turn Over</u>

Appendix B

B1: Project Proposal Form

Title: College Accountant – College Accounting Software

- ***Please describe the type of the system being developed.***

This system '*College Accountant*' is being exclusively developed for colleges/ universities/institutes. It is an all-round software solution for managing the accounts of the college. This would be an online as well as a desktop based application. The online version of the system provides a gateway for the students to pay the tuition fees, examination fees, library fines, late fee fines, re-examination fees etc. Along with these students can view the previous transaction history. Also, students who have to obtain any refunds from the college can apply for the same. The desktop application is mainly to be used by the accountant. The system would very well integrate the payroll system for the faculties and other staffs of the college. The examination fee-fine, library, bus, hostel, mess, cafeteria, bill payment, inventory and store modules are also being integrated to this system. Notifications in the form of SMS and email are being provided. Graphical reports can be generated on the flick of a button. These reports help the management to take better decisions.

- ***Who is system being developed for?***

This system is being developed for colleges /universities / institutes. The system would be directly used by the accountant and students. Indirectly the system is being used by the librarian, examination controller, faculties, store and departmental in-charge, bus in-charge, director, college management etc.

- ***Why is such a system required?***

In the current scenario the college uses software like Tally, MS Excel for accounting purposes. Though there is wide use of computers in the current system, still it's not efficient since various departments are not linked with each other. For example, the library is not connected to the accounts module, so if a student needs to pay a fine he/she needs to waste a lot of time jostling here and there to get his/her dues settled. This system is being developed by keeping college in mind and thus has all the necessary modules which would be required. Other software can be expensive and may contain unnecessary modules since those software are created in general and not specific to any organization. Thus the proposed system helps in saving space, time and cost. There is less wastage of paper which would otherwise be used for record keeping. Also, the reports generated by the system could easily help the management understand the current financial situation of the college and eventually help them take steps either for further improvement or recover losses.

- ***What will be the main challenge for you in building the system?***

Since this system deals with money transactions; the system needs to be highly secure. The system provides the students a payment gateway to pay up their fees and fines. Designing and making this gateway secure and user friendly would be the most difficult challenge. Among other challenges would be on how to integrate the modules with this system. Already few management systems have already been implemented in the college like the Library management system etc. hence it has to be checked how data can be fetched from that system and made usable for this system. In the payroll system, the salary is decided on the attendance of the faculties and staff. Since these records are of utmost importance hence these data needs to be stored in a robust database with excellent backup and recovery. The system would be developed in Java since Java is a platform-independent language, hence this system would be compatible with most of the operating system. The payment gateway would be designed in JSP using Servlets in order to have maximum security.

- ***What new ideas and theory will you need to learn to build the system?***

For the development of this system new technologies need to be studied in detail. The developer has no prior knowledge of JSP, Struts, and Hibernate etc. and would require studying these in detail for the successful completion of this project. Thorough research of Tally has to be done in order to make system comparable but better suited for various

Colleges. Concepts of encryption and decryption need to be learned in order to provide good security mechanisms for this project.

B2: Project Specification Form

Section B: Brief description on project background

Problem Context and Description of problem area

In the present scenario, there is no link between any of the management systems of the college. College uses Tally for accounting purposes. There is a constant growth in the amount of data. Also, there is also lot of wastage of resources. Poor security measures add up to the problems of the current system. Data can't be checked for accuracy and concurrency.

The problems being faced in the current system are:

- ✓ Use of expensive accounting software: TALLY is being presently used for managing the accounts in the college. Tally is an ERP and is feature-rich software. However, most of the features are not required in the current environment. And it costs a bomb to include additional features into the previously procured software.
- ✓ Experienced required users to operate system: Tally being complex software to handle needs to be operated by only experienced users. Prior training is required to exploit the full potential of this software. This makes the software less user friendly.
- ✓ Security: the current system provides very less security measures. Since, the system deals with money it is very important to make the system secure.
- ✓ Waste of resources: the current system leads to wastage of many resources. Lot of wastage of paper in printing of receipts etc. Wastage of time and energy of students when students have to pay library fines. They first need to get a receipt from the librarian, then pay the fine at accounts and then go back to library to verify further details.
- ✓ Data accuracy and concurrency: there are no measures to check if the data entered is correct or not. The system doesn't check for ambiguous data. Since there is no

linking between modules. Also, there are no measures to check for data concurrency.

- ✓ System upgrades: upgrading of the system is a costly affair. And it is important to keep the software updated with current times.

Rationale

There are various problems associated with the current system which have been identified earlier. To resolve these problems the proposed system is *College Accountant*.

Tangible Benefits: which the proposed system provides are as follows-

- ✓ Data is stored and managed in a more proper manner.
- ✓ Less wastage of paper.
- ✓ Easier to search records
- ✓ Transparent transactions to college accounts

In-Tangible Benefits: which the proposed system provides are as follows-

- ✓ Lot of time can be saved
- ✓ Reports generated can help in better management of the college funds
- ✓ Less human effort and errors
- ✓ Guarantees user satisfaction
- ✓ User friendly – very less training to be imparted for users

Target Audience

The primary uses of this system would be:

- ✓ Accountant and
- ✓ Students

Indirectly the system would be used by:

- ✓ Librarian,

- ✓ Examination Controller,
- ✓ Hostel In-charge,
- ✓ Store In-charge,
- ✓ Bus In-charge,
- ✓ Faculties and Staff,
- ✓ Mess and Café in-charge,
- ✓ Director,
- ✓ College Management.

Nature of challenge

This accounting software includes various modules of the college system, hence is considered an ERP. And it is a well-known fact that developing an ERP is in itself a herculean task. However the various challenges in developing the system would be:

- ✓ Collection of data: Since more than ten modules are being integrated into this system, hence a lot of useful data needs to be collected. The data needs to be analysed and checked for accuracy.
- ✓ Integrating the Modules: all these modules need to be made compatible and integrated properly with the system.
- ✓ Payment Gateway: a payment gateway is being provided for the students to pay up their fees/fines. Developing and implementing this gateway would be a great challenge.
- ✓ Report Generation: extensive graphical reports are to be generated. For this the data collected need to be made compatible and used to generate the reports.
- ✓ Understand Account-keeping: being from a science background, we have very less knowledge of Account-keeping and hence the developer needs to study about this in detail.
- ✓ Programming Language: gaining master of the Programming language [Java in this case] in order to develop all the functionalities.
- ✓ User-friendly: making the system as user friendly as possible. Designing the system by keeping in mind the various kinds of users [Novice, expert, casual].

Section C: Brief description of project objectives

Scope of Proposal

This system '*COLLEGE ACCOUNTANT*' is being exclusively developed for colleges/ universities/ institutes. It is an all-round software solution for managing the accounts of the college. This would be an online as well as a desktop based application. The online version of the system provides a gateway for the students to pay the tuition fees, examination fees, library fines, late fee fines, re-examination fees etc. Along with these students can view the previous transaction history. Also, students who have to obtain any refunds from the college can apply for the same. The desktop application is mainly to be used by the accountant. The system would very well integrate the payroll system for the faculties and other staffs of the college. The examination fee-fine, library, bus, hostel, mess, cafeteria, bill payment, inventory and store modules are also being integrated to this system. Notifications in the form of SMS and email are being provided. Graphical reports can be generated on the flick of a button. These reports help the management to take better decisions.

Deliverables

The developer has distributed whole project in various modules each under as core, enhanced and special functionality:

Core Functions

- ✓ *Student Fee Module:* In this module the fee to be submitted by a student corresponding to the year and semester they are studying is calculated. The system checks if other fees like registration, counselling, convocation, is paid or not. It also checks for any applicable scholarships, concessions etc. Late fee fine is calculated according to the fine slabs.

- ✓ *Payroll Management:* this module deals with the payment of salary of the staff and faculties of the college. The system calculates the amount of days worked to arrive at the salary to be given. If any bonuses or deductions are applicable they are adjusted to the above calculated salary. The salaries are deposited directly into the accounts of the staff and faculties.

- ✓ *Library Module:* this module calculates for the fine to be paid by the students for the books issued. Fine could be either late book return or misplaced/damaged books. Also, it helps in keeping a check on the books which are being ordered as new or as replacement or due to shortage.

- ✓ *Inventory/ Store / Departmental:* this module keeps a check on the items which are being ordered for the store. Items could be like sports equipment, furniture, markers, dusters, cleaning equipment, gardening equipment etc. Also keep a check on the items which are being sold off. To check for the rent payment to be received from the departmental store owner.

- ✓ *Examination Module:* this module checks for the attendance of the students. If a student has low attendance then he/she will have to pay the calculated amount to sit for extra class to recover for the classes which have been missed. This module also checks for the performance of the students. If a student fails in the internal or external then the required payment has to be paid by the student for sitting in the exams.

Enhanced Functions

- ✓ *Hostel Module:* this module checks for the payment received by students in respect of the hostel chosen by them, ac or non ac rooms, single-double-triple bedrooms etc. It also checks for the miscellaneous expenses of the hostel like laundry, newspaper, damaged goods replacement etc.

- ✓ *Bus Module:* this module keeps track of all the payments which have been received in respect of the bus services provided to the students and faculties. It also covers the fuel and maintenance charges of the buses.

- ✓ *Maintenance and Development:* this module calculates the expenses of the college in terms of building maintenance and improving the present infrastructure.

- ✓ *Mess /Cafeteria Module:* this module checks for the mess fee payment by student and faculties. Total charges to be calculated for those who avail the mess facilities occasionally. To check for the rent payment to be received from the mess and cafeteria owners.

- ✓ *Bill Payments:* this module checks for the payment the college has to do to keep various third-party services active. These services include the payment for the ISP provider, security, electricity, water, etc.

- ✓ *Notifications / History:* SMS / email notifications can be sent to the defaulters. Reminders can be sent beforehand so that students can pay the fees in time. Notifications can be sent to the staff and faculties on successful transfer of the salaries to their accounts. The whole account history of the students can be viewed in a glance.

Special Functions

- ✓ *Graphical Reports/Receipt Generator*

  Weekly/Monthly/Yearly reports can be generated showing the income and expenses of the college. These reports would be of importance to the management of the college which could help them further in the better managing of the college funds. These reports can be generated graphically for better understanding. Receipts could be printed for future references.

- ✓ *Payment Gateway:* would be developed so that students can easily pay the fees online at any time. They can pay the fees by credit card, net banking etc. and can pay the college fees, fines, or other charges which needs to be paid to the college. This is more secure method and payments can be easily tracked.

Evidence of Limiting the Project Scope

Maximum effort would be utilized in developing all the key features of the system, still some functionality may remain unfulfilled, such as:

- ✓ Authentication: since different modules are integrated with each other various authentications would be required before accessing the database. And to work on actual data would be risky; hence the system would use dummy databases to show the working.
- ✓ Payment gateway: the gateway would be developed in PHP or ASP.net and not in Java [as the other system]. Since it takes a lot of money to configure a server for JSP.

Other Learning Objectives

The concepts and ideas in lieu of the trained skills which would be required in the development of the project:

- ✓ Encryption and decryption for password and other types of protection
- ✓ Hibernate to make secure database
- ✓ Having thorough knowledge of Tally ERP
- ✓ Study other accounting systems
- ✓ Research more about Payment gateway

Section D: Brief description of resources needed by the proposal

Hardware Resources

The hardware resources which would be required for the proper functioning of this system would be:

- ✓ Desktop Computer/Laptop
- ✓ Processor : Core2 Duo, Dual Core, Core i3,Core i5, Core i7 equivalent or above

- ✓ Memory : 2GB RAM or more
- ✓ Hard-drive: 100GB or more [approx.. 100MB to install application]
- ✓ Monitor Resolution : 1024 x 768 or Higher [Graphic card not required]
- ✓ Server: for hosting the website part of the project.
- ✓ Modem: [wired/wireless] for seamless Internet connection
- ✓ All-in-One Printer: Copier-Scanner-Printer [Fax optional]
- ✓ Ergonomic Keyboard and Mouse

Software Resource

The software resources required for the successful development of this project are as follows:

- ✓ OS : Windows 98 / 2000 / ME / XP / Server 2003 / Vista / Server 2008 / Windows 7 / Windows 8 / Server 2012
- ✓ IDE : Net Beans , Eclipse
- ✓ Database : MySQL
- ✓ Documentation: Microsoft Office Word, Microsoft Visio, Microsoft Power-point, Microsoft Office Project, Adobe reader etc.

Access to Information/Expertise

For the development of any project, the foremost task is to collect the data which would be used by the system. And this data can be collected from the users of the system, experts of that particular field, studying other similar systems, books and other internet resources. To collect data from users/experts the best data gathering method would be to conduct interviews and fill-up questionnaires from:

- ✓ Accountants: they are the primary users of the system and have the exact knowledge of the problems in the current system and their expectations from the proposed system. Also, they are clear about the data and the functionalities which the system would need to fulfil the user needs. The data which can be collected can be: payroll of employees, student fees, bill payments etc.
- ✓ Librarian: can help in providing the book details and how the fine is calculated for overdue/damaged/lost books. This data can help the students pay up the fines in time and make the transactions transparent.

- ✓ Examination Controller: can provide the data about how the re-examination fee needs to be calculated.
- ✓ Store In-charge: can provide data about the things which are being procured for the college; the actual cost, monthly costs, maintenance costs etc.
- ✓ Bus In-charge: can provide the data bout the bus details, daily kilometres run details, diesel/petrol consumption, maintenance and servicing charges etc.
- ✓ Mess/Café In-charge: can provide the data about the rent being paid by them to work on college grounds, electricity consumption etc.

Other reliable resources for data gathering:

- ✓ Reference Books
- ✓ Internet : only research materials [forums and blogs to be avoided]
- ✓ E-books

User Involvement

In –course of the development of this project various users would be involved to obtain valuable suggestions and feedbacks from them. Also, some users can be testers of the system and help with the evaluation of the system. The users can be:

- ✓ Accountants : primary users [can also help in testing]
- ✓ Students : Primary users [can also help in testing]
- ✓ Librarian
- ✓ Faculties and other Staff
- ✓ Store In-charge
- ✓ Examination In-charge
- ✓ Hostel Wardens
- ✓ Bus In-charge

- ✓ Mess and Café In-charge
- ✓ Supervisors/Advisors

Section E: Academic research being carried out and other information, techniques being learnt

Academic/Theoretical Research Areas

The academic research area for developing the proposed system would be:

- ✓ *Tally:* since this project is the development of accounting software, a thorough research has to be done of the other similar software in the market. And among these Tally is the most popular accounting software.
- ✓ *Account-keeping*: the developer comes from a Science [computers] background; he/she would need to study deeply about how accounts are managed.
- ✓ *Payment Gateway*: is one of the special features being implemented in the project. However, thorough research has to be done in this context to implement this in the project in the most suitable way.

Technical/Programming Research Areas

- ✓ Java: The desktop application would be developed using Core Java and the payment gateway is being proposed to be designed in JSP.
- ✓ Net Beans: is the IDE [integrated development environment] in which the application would be designed
- ✓ MySQL: is the database for this proposed system
- ✓ Hibernate: is concept of securing the database
- ✓ Encryption & Decryption: these would be used in order to make the data more secure and confidential.

Information and Resources

Books:

- ✓ Bert Bates, Kathy Sierra (2003). *Head First Java*. 2nd ed. O'Reilly Media.

- ✓ Katherine Sierra, Bert Bates (2008). *SCJP Sun Certified Programmer for Java 6 Exam 310-065*. Tata McGraw - Hill Education.
- ✓ Santosh Kumar K., Kogent Solutions (2008). *JDBC, Servlets, And JSP Black Book*. Dreamtech Press.
- ✓ Chris Zeis, Chris Ruel, Michael Wessler (2009). *Oracle 11G For Dummies*. Wiley India Pvt Ltd.
- ✓ Santosh Kumar K. (2013). *Spring and Hibernate*. 2nd ed. Tata McGraw - Hill Education.

Internet Resources:

- ✓ Srinivas Kalabarigi. (2013). *Encrypt and Decrypt data.* Available: http://www.codeproject.com/Tips/635973/Encrypt-and-Decrypt-data. Last accessed 20 September 2013.
- ✓ (2013). *Learn Tally.ERP 9.* Available: http://www.tallysolutions.com/website/html/services/learning-tallyerp9.php. Last accessed 10 September 2013.
- ✓ (2012). *Hibernate.* Available: http://www.tutorialspoint.com/hibernate/. Last accessed 12 September 2013.

Human Resources

- ✓ Accountants : primary users [can also help in testing]
- ✓ Students : Primary users [can also help in testing]
- ✓ Librarian
- ✓ Faculties and other Staff
- ✓ Store In-charge
- ✓ Examination In-charge
- ✓ Hostel Wardens
- ✓ Bus In-charge
- ✓ Mess and Café In-charge
- ✓ Supervisors/Advisors

Real World Methods and Practices

Methods and practices used in the real world can be useful and effective in researches, in order to carry out the research required for this project are:

- ✓ *Interviews:* since the maximum data needs to be collected from the Accountant, there would be a face-face interview with him/her. The interview conducted would be informal with a range of open-ended questions.
- ✓ *Questionnaires*: would be distributed to those users of the system who are not directly using this system, like librarian, examination controller etc. The questionnaire would be of paper-pencil type with a range of bounded questions.
- ✓ *Competitive Analysis*: similar systems would be searched and thoroughly studied analyzed for their merits and demerits in-order to provide the proposed system with maximum usable functionalities.

Section F: Brief description of the development plan for the proposed project

Software Methodology

Since the proposed system integrates various modules of the college, it is considered to be a vast and complex system. Best suited methodology for the development of this system could be the **Spiral Model.**

Spiral Model arranges the various and tasks and activities in loops. The loops represent development phases and for a project there can be any number of loops. It is an iterative model which inculcates into it the features of both the *Waterfall Model* and *Prototype Model,* hence eliminating the demerits of these models and including good features into it. The development of the software takes place in a systematic order [waterfall] over the loops and prototypes are developed and shown to end-users for evaluation [prototype] at end of a phase. This ensures complete user satisfaction and also reduces risk of project failure. This model shows special emphasis on Risk Analysis.

Reason for choosing Spiral Model

Developing such a vast and complex system is a huge task and involves lot of time and money.

- ✓ A spiral model is usually applied on large and mission-critical projects, hence best suited for this proposed system which in itself is large with time and budget constraints.
- ✓ The spiral model lays special emphasis on Risk Management. Hence there is a very high chance of reducing the risk of project failure. This is necessary since this project is vast and costly, and any failure will lead to huge losses.
- ✓ Vast and complex system demands for strong documentation and approval control which are well covered by this modelling technique.
- ✓ The project estimations like cost, schedule and other resources become more realistic with the progress of the project through the loops of this model.
- ✓ Since a prototype is developed after the end of each phase, it ensures maximum user satisfaction and helps in the proper management of the project.
- ✓ Additional functionalities can be added at a later stage which in turn helps in the development of a highly customizable project which is the need of the hour.

Phases of Spiral Model

- ✓ Planning: this phase deals in studying the project objectives, alternatives in design and constraints imposed because of cost, technology, schedule, etc.
- ✓ Risk Analysis: in this phase other approaches are studies which can be implemented in order to fulfil the identified constraints. Operational and technical issues are addressed here. Risk mitigation is in focus in this phase. And evaluation of all these factors determines future action.
- ✓ Engineering: in this phase the planned product is developed. Testing is also done. In order to do development, waterfall or incremental approach can be implemented.

- ✓ Evaluation: in this phase the progress is reviewed and judged considering all parameters. Issues which need to be resolved are identified in this phase and necessary steps are taken.

Hardest Tasks

The most hardest and challenging portion of the development plan would be:

- ✓ Research and Planning: the proposed system is vast since many modules need to be integrated and hence a lot of data needs to collect in this relation. Even proper planning needs to be done about how to go about developing this system.
- ✓ Implementation: after development of this software it would be another herculean task to implement this project. All the modules would have to be made to work in sync with each other. Also, basic training needs to be imparted to the end users.
- ✓ Presentation: this is the most crucial and hardest part of any project. A proper and convincing presentation needs to be prepared for the customer. The developer needs to convince the customer through his/her presentation that the system which has been developed is the system which the customer wished for.

Development Plan

The order in which the task would be accomplished:

Start Date: 13 September 2013

Duration: 33 weeks

End Date: 17 April 2014

1. Planning : 8 weeks
2. Risk Analysis : 8 weeks
3. Engineering : 15 weeks
4. Evaluation : 2 weeks

Section G: Brief description of the evaluation and test plan for the proposed project

Test Plan Strategies

The test plan strategies used in this project will be:

- ✓ *Unit and Integration Testing* – This system will be developed in different modules or units, based on these units, testing are done individually, once the modules are tested, they are been integrated for the integration testing. These testing will be performed by the developer himself.

- ✓ *Black-box testing* - allows the tester to interact with the system to assess the test results, through this testing, user get to know what the system is supposed to do, but not how it is done.
- ✓ *White-box testing* – this type of testing allows the user to access the system along with its inner codes. This will help the user to know the system deeply and can better understand the inner functionality or working of the system. This test will be performed by a supervisor. The supervisor will be giving evaluation based on the runtime effectiveness and efficiency of the system codes, in order to produce a better performance.
- ✓ *Alpha Testing* – this testing will be carried out to if all the functionalities in the project are working according to user's need or not. This testing is done to know whether the system is acceptable to end-user or not. The tester will be carrying this testing by presenting real data to know actual input and output of the system. Later on he can give comments on the user-friendliness of the system.

To accomplish the mentioned testing strategies, these types of testing are to be applied:

- ✓ *Usability Testing* –it will test the usability of the system. It basically considers the user interface design, familiarity, user acceptance and how the system recovers from an error. Tester: End user, supervisor. Data sets: 3
- ✓ *Functionality Testing* – Functions featured in the system will be tested thoroughly to ensure that it is working properly. Tester: Developer, supervisor, end user. Data sets: 3
- ✓ *Compatibility Testing* – The system will be tested on different operating systems and software. Tester: Developer, end user. Data sets: 3
- ✓ *Runtime Testing* – System loading times, execution problems and other runtime issues are to be tested. Tester: Developer, end user. Data sets: 3
- ✓ *Configuration Testing* – The system will be tested using different hardware configurations. Tester: Developer, end user. Data sets: 3

Evaluation on Success Criteria

To evaluate on how successful a project is, these criteria are considered:

- ✓ *Meeting user requirements* – A project which can fulfil all the requirements and expectations of all users is called a successful project. Evaluator: End users
- ✓ *Functionality* – Functionality is determined by the working of the system. The system functionality such as in traction of different module is working or not, report generated or not. Evaluator: End users, Developer
- ✓ *Usability* – Elements such as user acceptance, accessibility, familiarity and recoverability will be evaluated here. Evaluator: End users
- ✓ *Level of Content* – it shows how well the information and content is available in the system and how much deep the contents are. Evaluator: End users
- ✓ *Research and Analysis* – A good amount of well investigated research and analysis work is important to the success of a project. Evaluator: Supervisor
- ✓ *Documentation* – The ability to come up with a documentation which is very well formatted and good English and grammars are used. Evaluator: End User

Appendix C

Questionnaire Consent Form

Questionnaire ID: CllgACcc-Q-001

Questionnaire

College Accountant – a college accounting software

Hello,

The project is entitled: “College Accountant – a college accounting software”. As the name suggests, this is an accounting software which is being tailor made for a college/university/institute. This software may replace the current software in place which is being used for accounting. It contains various modules which are inter-linked which each other for the better management of the account records. The various modules this system will provide are: student fee, payroll, library, store, examination, hostel, bus modules. Since you are a part of a college/university and have direct access to management systems your views are of great importance to this research.

The purpose of this research is to gather as much information from the current environment which can help the researcher in developing a system which is user-friendly, meets user requirements and is future–proof.

Requesting you to spare about 15 minutes from your time to fill up this questionnaire through email itself. Please fill correct information since your answers are of great importance to the success of this research.

Privacy Information

The information collected through this questionnaire will be kept completely anonymous and confidential and will be used for research purposes only. The information will not be shared with competitors or other sources without prior permission.

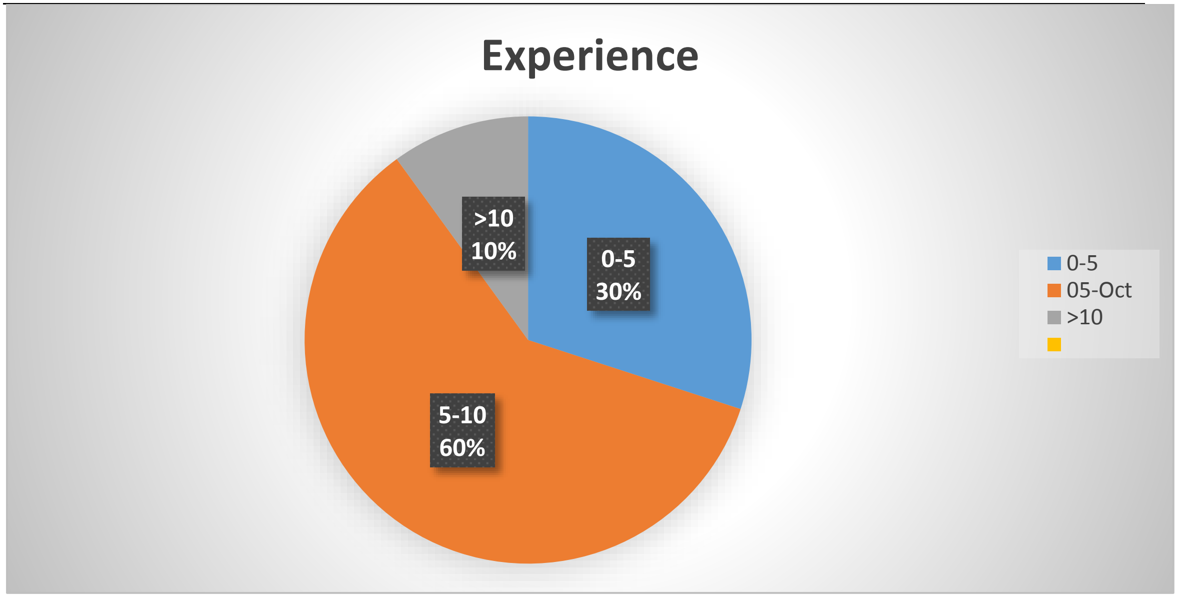
The questionnaire is being distributed by the student researcher currently, pursuing BE (Hons.) in Computing with Specialization in Software from Asia Pacific Institute of Information Technology, SD India (affiliated to Staffordshire University). This Final Year project needs to be submitted, towards partial fulfilment of BE (Hons.) degree.

Questionnaire Analysis

Section B: [Tick one]

B.1 For how long have you been working in this college/university?

- □ 0-5 years
- □ 5-10 years
- □ More than 10 years

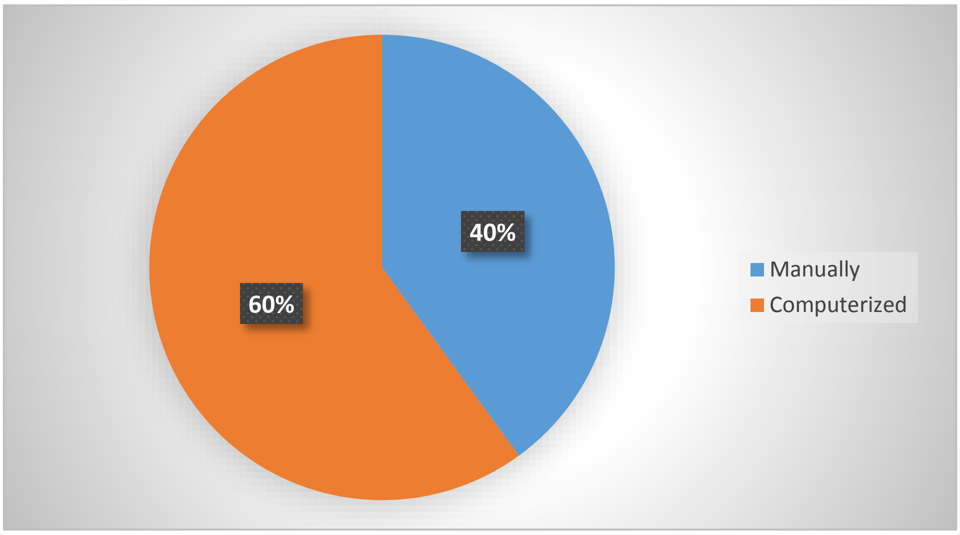


Analysis: From the data gathered it is very clear that the 60% of the users have experience between 5-10 years. Thus, this data gathering has been conducted for those people who are experienced and are well aware of the current scenario.

B.2 How is accounts book-keeping done in this college?

□ Manually : records kept in hard-copy registers

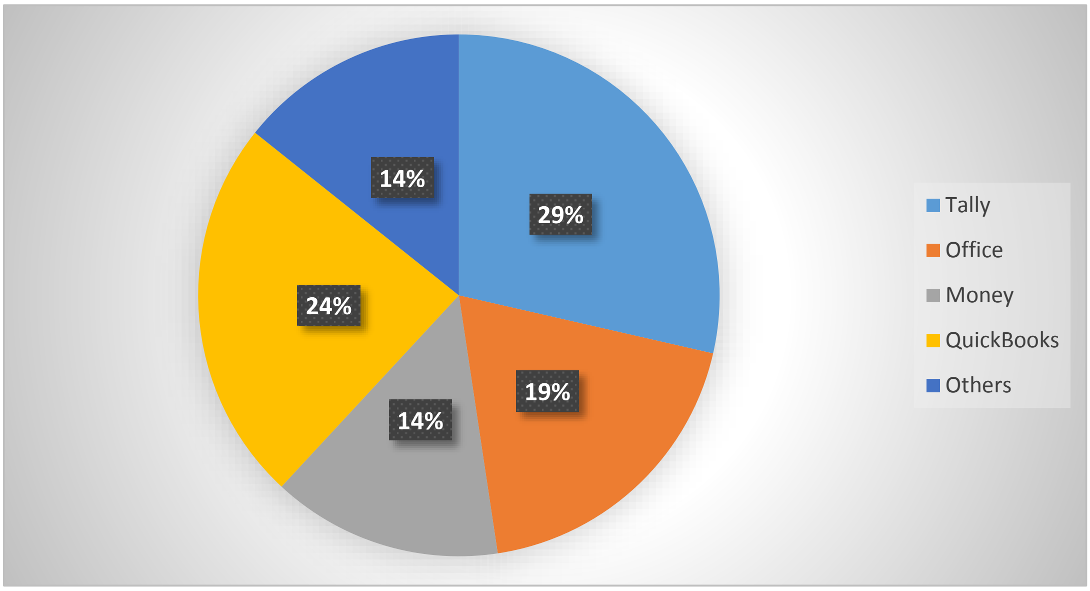
□ Computerized : sophisticated accounting software



Analysis: From the data gathered it is clear that 60% of the college reviewed did have some computerized way of managing the account keeping.

B.3 If you answered computerized in previous question, then which accounting software is being used currently?

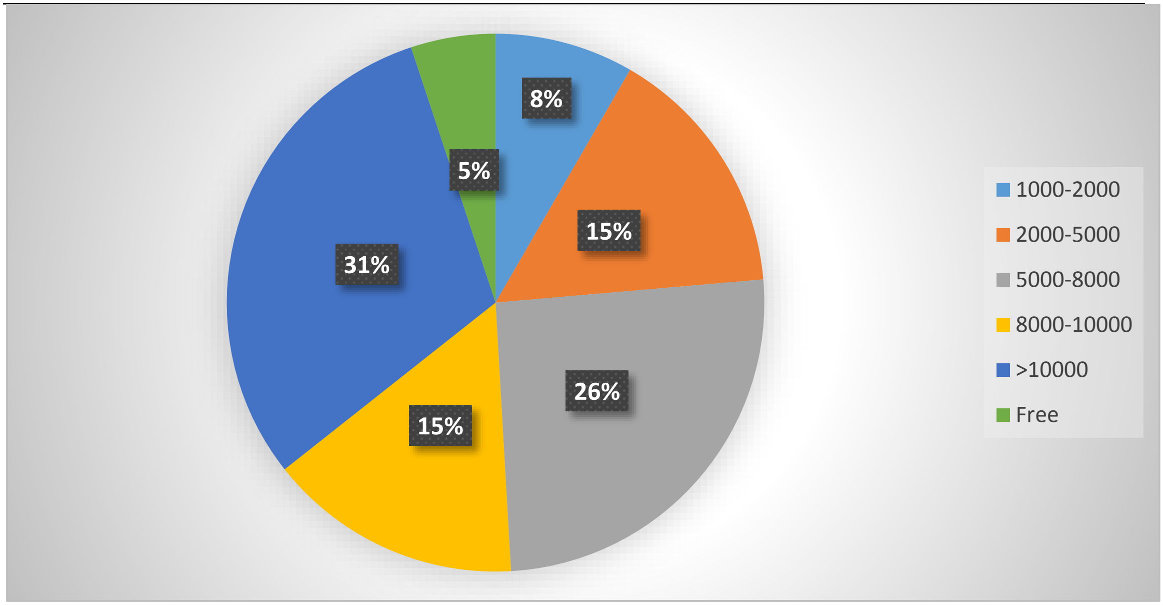
- □ TALLY
- □ Microsoft Office Accounting
- □ Microsoft Money
- □ QuickBooks
- □ Others, please specify ________________________



Analysis: From the data gathered, Tally is the most widely used software after which QuickBooks come. Both of these are expensive software with loads of features. The organization is forced to use these software due to un-availability of required software.

B.4 What is the price range of the software mentioned above?

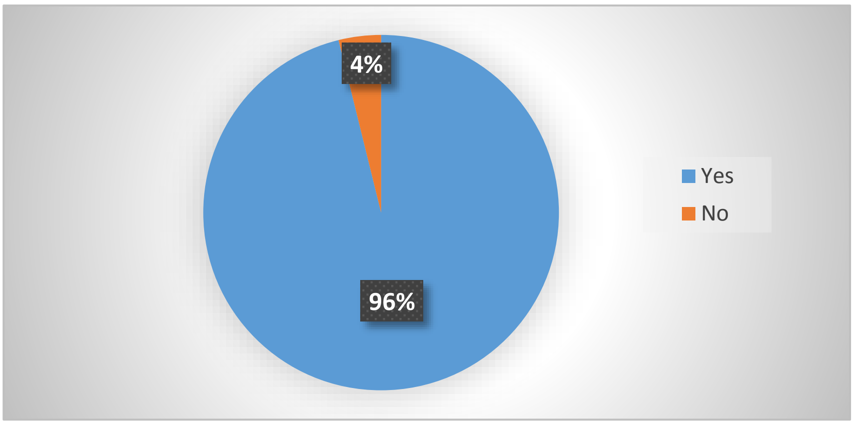
- □ ₹ 1000 - ₹ 2000
- □ ₹ 2000 - ₹ 5000
- □ ₹ 5000 - ₹ 8000
- □ ₹ 8000 - ₹ 10000
- □ More than ₹ 10000
- □ Free of Cost



Analysis: From the data gathered, the colleges/ universities are spending a huge amount of money in purchasing these accounting software.

B.5 Does this software require any renewal?

- □ Yes
- □ No



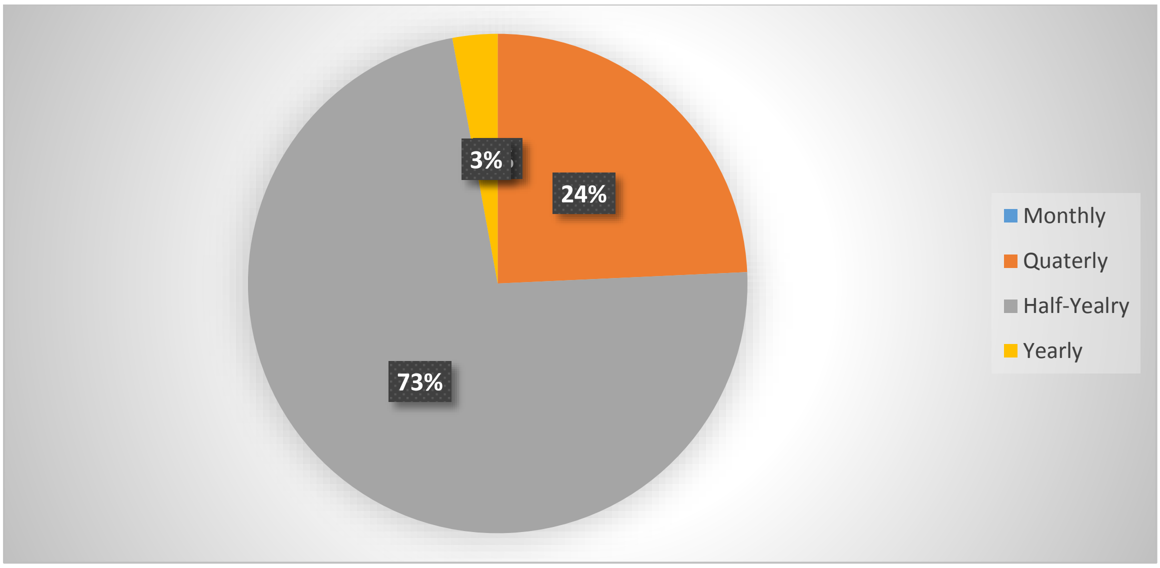
Analysis: From the data gathered it is quite clear that most of the software being expensive required renewal on top of it.

B.6 If yes, how often?

- □ Monthly
- □ Quarterly

- □ Half-yearly
- □ Yearly

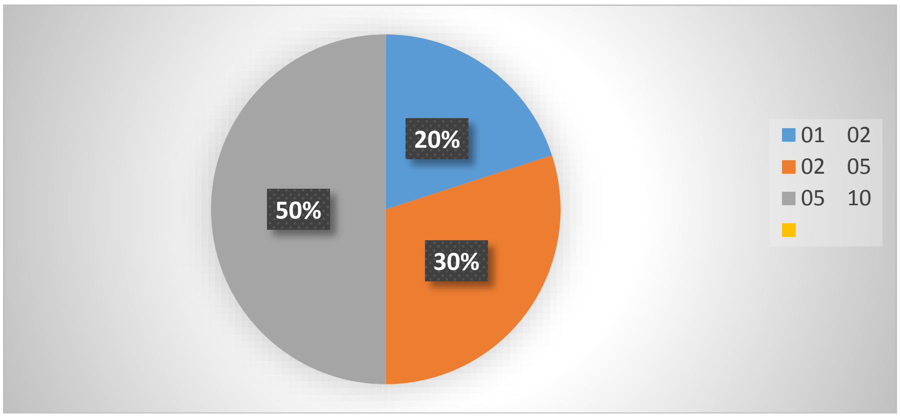
Amount: ₹ _______________



Analysis: From the data gathered, a yearly renewal of the software is required. And mostly the cost of such renewals is at least half the cost of actual product.

B.4 For how long have you been using the computer system?

- □ 1-2 years
- □ 2-5 years
- □ 5-10 years

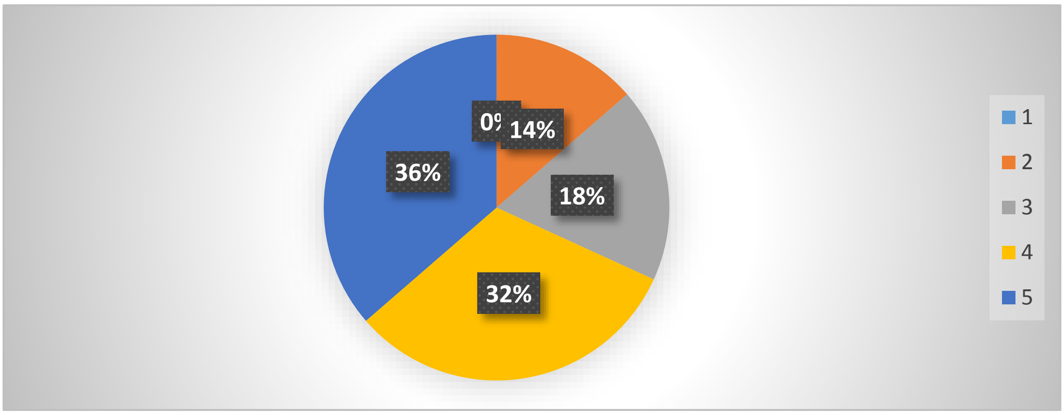


Analysis: From the data gathered, 50% of the accountants were well versed with a computer system.

B.5 How do you rate your Computer knowledge? [Rating: 1-low 5-high]

- □ 1
- □ 2

- □ 3
- □ 4
- □ 5



Analysis: From the data gathered, it is quite evident that most of the accountants have excellent computer knowledge. Hence, they will be able to understand the system well and too much effort in training won't be required.

Section C: Fill in detail

C.1 Please specify the advantages of the current system being used in the college for accounting.

Analysis: Most of the accountants said that in the current system they liked how easy the calculation processes are. Also, it is really easy to generate monthly reports.

C.2 Please specify the dis-advantages of the current system being used in the college for accounting.

Analysis: Most of the accounts said that they were fed up with slow systems which took lot of time to load and consumed oodles of memory space.

C.3 Please specify the features which you would like to have in the system being developed for you?

Analysis: Most of the accountant wished for features like SMS notifications, online payments, proper integration of modules etc.

Questionnaire Filled Form

Interview Analysis

From the data gathered through this study, the librarians were not satisfied by the current method book ordering and fine calculation. The current systems were slow. The fine calculation and payment was a long tedious time consuming process. They wished for a system where fine calculation is automatic and records are updated automatically after fine collection.

Focus Group Study Analysis

From the data analysed the researcher has come to conclusion that the students of these colleges preferred paying up the fees half-yearly [semester-wise]. Most of them paid through DD and Cheque. However, there were times they were not able to pay fees on time since the fee payment schedule coincided with holidays and most of the students were at their respective homes. They were very eager to have the features of notifications and reminders. They also supported the fact that online fee payment facility would be a great feature.

Naturalistic Observation Analysis

From this data gathering technique the following conclusions were made:

- ✓ The insertion and updating of records takes a long time
- ✓ To order for new products a monthly/weekly list is made which is deposited to accounts department for review. Then the part is ordered by the store manager
- ✓ The updating was not a fool proof process
- ✓ Duplicity of data was evident
- ✓ Ordering process is time consuming and inefficient.

Appendix D

Usability Testing Filled Forms

Please Turn Over

Appendix E

E1: Fast Track Ethical Form

E2: Log Sheets

Please Turn Over