

# **Chapter 1**

## **Introduction**

### **1.1 Introduction**

“Accounting is as old as money itself”. Since in early ages commercial activities were based on barter system, record keeping was not a necessity. The Industrial Revolution of 19th century along with rapid rise in population, paved way for the development of commercial activities, mass production and credit terms. Thus recording of business transaction has become an important feature.

In recent years with the change of technologies and marketing along with stiff competition, accounting system has undergone remarkable changes. Accounting has become an important part of every company. Businesses are required to keep books on their credits and debits.

When a person starts a business, whether large or small, his main aim is to earn profit. He receives money from certain sources like sale of goods, interest on bank deposits etc. He has to spend money on certain items like purchase of goods, salary, rent, etc. These activities take place during the normal course of his business. He would naturally be anxious at the year end, to know the progress of his business. Hence, the details of the business transactions have to be recorded in a clear and systematic manner to get answers easily and accurately for the following questions at any time he likes. What has happened to his investment? What is the result of the business transactions? What are the earnings and expenses? How much amount is receivable from customers to whom goods have been sold on credit? How much amount is payable to suppliers on account of credit purchases? What are the nature and value of assets possessed by the business concern? What are the nature and value of liabilities of the business concern? These and several other questions are answered with the help of accounting.

If this accounting system is computerized then it can save a great deal of time when posting credits and debits. Rather than having to make entries in different books, as is common with more traditional methods. It creates reports much faster than manual systems. Calculations are done automatically in software programs, minimizing errors and increasing efficiency.

## **1.2 Previous Work**

Previous work covers a wide variety of topics such as internet accounting, Pastel Express 2009 Accounting tools, accounting information system(AIS),batch processing system.

Ohmori [1] proposed an internet accounting system where transactions are entered on the spot and business data created by transactions are distributed to appropriate staff of the company, becomes more important for companies operating worldwide. The accounting system described in this paper consists of a three-tier structure: Web, application and database servers. This system requires internet connection to access the system. Also it is less secure system because business transaction data may spread out through internet.

Mashapa [2] proposed an accounting tool called pastel express 2009 which was designed specifically for providing accounting solutions for SMMEs with basic entry accounting needs. But it supports up to 30 journal entries. Also the tool lacks of informative feedback and error tolerance. In addition the system is not user friendly.

Elena [3] proposed an information accounting system. This research study is aimed, based on empirical evidence, at measuring the relationship between the use of the Accounting Information Systems (AIS) by the Small and Medium Sized Enterprises (SMEs) in Spain. A limitation of this work it was developed for small medium enterprises. But it is not efficient for large and medium enterprises.

Mathias [4] proposed computerized accounting Information Systems and Perceived Security Threats in Developing Economies. But it security system is not clear to understand.

## **1.3 Motivation**

Manual accounting implies that employees perform the whole accounting cycle manually on a periodic basis. Manual systems put pressure on people to be correct in all details of their work at all times, the problem being that people aren't perfect, however much each of us wishes we were. With manual systems the level of service is dependent on individuals and this puts a requirement on management to run training continuously for staff to keep them motivated and to ensure they are following the correct procedures. It can be all too easy to accidentally switch details and end up with inconsistency in data entry or in hand written orders. This has the effect of not only causing problems with customer service but also

making information unable to be used for reporting or finding trends with data discovery. Reporting and checking that data is robust can be timely and expensive. This is often an area where significant money can be saved by automation.

It takes more effort and physical space to keep track of paper documents, to find information and to keep details secure. When mistakes are made or changes or corrections are needed, often a manual transaction must be completely redone rather than just updated. With manual or partially automated systems information often has to be written down and copied or entered more than once. Systemization can reduce the amount of duplication of data entry.

The need for computerized accounting arises from advantages of speed, accuracy and lower cost of handling the business transactions. Some necessity of computerized accounting system are:

### **Numerous Transactions**

The computerized accounting system is capable of handling large number of transactions with speed and accuracy.

### **Instant Reporting**

The computerized accounting system is capable of offering quick and quality reporting because of its speed and accuracy.

### **Reduction in paper work**

A manual accounting system requires large physical storage space to keep accounting records/books and vouchers/ documents. The requirement of stationery and books of accounts along with vouchers and documents is directly dependent on the volume of transactions beyond a certain point. There is a dire need to reduce the paper work and dispense with large volumes of books of accounts. This can be achieved by introducing computerized accounting system.

## **1.4 Objective of the Thesis**

For an accounting system to be considered as effective it must meet the basic objectives of information systems. The first objective is that they must pass the cost benefit principle or cost benefit relationship. Financial information is not free, other companies even spend millions every year just to gather and organize financial information to assemble into their

financial statements. Under this principle, the cost of providing financial information in the financial statements must not outweigh the benefit of that information to the users. If the firm is planning to improve their IT system, they must consider the cost-benefit principle. The second objective is to protect the entities assets, to ensure that data are reliable and minimize wastes and the possibility of theft or fraud. This is also known as the control principle. The third objective is to be in harmony with the entity's organizational and human factors. This can also be referred to as the compatibility principle. The last is to be able to accommodate growth in the volume of transactions and for the organizational changes, also called as the flexibility principle.

On this project, we will propose a computerized accounting system for small, medium and large enterprises. This system will provide real time information about accounting. It will analyze data and gives various accounting reports. The major goals are:

- Developing a computerized accounting system for a company.
- Generating ledger sheet, trial balance, income statement and balance sheet from journal entry to determine profit or losses of a single day of a company.
- Providing a user friendly interface.
- Provide a secure system.

## **1.5 Organization of the Report**

To represent our work in organized way, we divided this report in five chapters. Chapter 1 introduces the problem and states the scope of the problem in past and present time. In chapter 2, we discussed some related topic we need to note here in brief. The methodology we followed in the project is described with necessary sketch in chapter 3. In chapter 4, we have showed our implementation of the project and the experimental result on our developed application. Finally we have made a conclusion with overview of our work and future plan of the work in chapter 5.

# Chapter 2

## Literature Review

### 2.1 Introduction to Accounting

Accounting is a profession used to make financial and business decisions. Billions of dollars exchange hands every day, in millions of separate business transactions. These are recorded and reported on using a comprehensive set of guidelines, referred to as Generally Accepted Accounting Principles.

The practice of recording a business transaction in two equal parts, called debit and credit entries. Debit refers to the left column and credit refers to the right column, in an accounting journal.

Each transaction describes both:

1. the object of the transaction - such as rent, telephone, or payroll expense; sales, fee or interest revenue.
2. the source of payment - cash or credit.

Money eventually changes hands in almost all transactions, either at the time of the transaction, or perhaps at a future date in the case of items purchased on credit. (Adjusting and closing entries are an exception and not typical, and represent special entries made by accountants to prepare financial statements, and reset certain accounts at the end of a fiscal year.) Sometimes a transaction involves cash directly, at the time of the event, such as a cash sale at a grocery store. It is more common, and safer, to use a checking account for routine purchases. These are all considered part of the Cash account.

Many, and perhaps most, transactions in a business take place on a credit basis. Businesses usually purchase their supplies and merchandise on a 30-day account, known as a trade account, or Accounts Payable. Sales are typically made in a similar fashion, called Accounts Receivable.

## **2.2 The Accounting Process in a Nutshell**

1. Capture and Record a business transaction,
2. Classify the transaction into appropriate Accounts,
3. Post transactions to their individual Ledger Accounts,
4. Summarize and Report the balances of Ledger Accounts in financial statements.

There are 5 types of Accounts.

1. Assets
2. Liabilities
3. Owners' Equity (Stockholders' Equity for a corporation)
4. Revenues
5. Expenses

All the accounts in an accounting system are listed in a Chart of Accounts. They are listed in the order shown above. This helps us prepare financial statements, by conveniently organizing accounts in the same order they will be used in the financial statements.

## **2.3 The Accounting Cycle**

The Accounting Cycle is a sequence of procedures used to record, classify and summarize accounting information in financial reports, on a regular basis.

Steps in the Accounting Cycle

1. Record (journalize) transactions.
2. Post journal entries to Ledger accounts.
3. Prepare a Trial Balance.
4. Make adjusting entries.
5. Prepare an Adjusted Trial Balance.
6. Prepare financial statements.
7. Journalize and post closing entries.
8. Prepare After-Closing Trial Balance.

## **2.4 The General Journal and Journal Entries**

Every business transaction is recorded in the General Journal. The General Journal is called the book of original entry. A journal is a chronological record of transactions - they are in date order.

Each entry is called a journal entry, and represents a different business transaction. Each transaction is recorded once, and only once. All journal entries follow the rules of debit and credit.

Journal entries should be made contemporaneously with the event they are recording, or reasonably soon after the event. Keep in mind that a journal is a chronological record of events. A contemporaneous writing is one that takes place at the same time as the event. This is the best time to record an event, because the facts and details are still fresh in our minds. Necessary documents, conversations, calculations, etc., are readily available to create a correct record of the event. If we wait too long, the event will be much more difficult to reconstruct.

### **2.4.1 Definition and Explanation of Journal**

The word journal has been derived from the French word "Jour" Jour means day. So, journal means daily. Transactions are recorded daily in journal and hence it has named so. As soon as a transaction takes place its debit and credit aspects are analyzed and first of all recorded chronologically (in the order of their occurrence) in a book together with its short description. This book is known as journal. Thus we see that the most important function of journal is to show the relationship between the two accounts connected with a transaction. This facilitates writing of ledger. Since transactions are first of all recorded in journal, so it is called book of original entry or prime entry or primary entry or preliminary entry, or first entry.

## **Entry**

Recording a transaction in the appropriate place of the concerned book of account is called entry. Entry may be of the following two types:

### **Journal Entry**

Recording a transaction in a journal is called journal entry or journalizing.

### **Ledger Entry**

Recording a transaction from journal to the concerned account in the ledger is called ledger entry. It is also known as ledger posting.

## **2.4.2 Narration**

A short explanation of each transaction is written under each entry which is called narration. The subject matter of the transaction can be ascertained through narration. Besides this, if there be any mistake in determining debit or credit aspect of a transaction, it can be easily detected from narration. "A journal entry is not complete without narration".

## **2.4.3 Format of Journal**

Date	Particulars	L.F	Amount	Amount
	Account to be debited .....Dr. Account to be credited (Narration)		XXX	XXX

## **2.4.4 Rules for Journalizing**

How a transaction is recorded in journal, is discussed below:

Suppose the transaction is:

Purchased furniture from M. A on 10.01.05 for \$16,000.

Here furniture accounting is debited and M A account is credited.



Date	Particulars	L.F	Amount	Amount
10.01.05	Furniture A/C .....Dr. M. A A/C (Being cost of furniture purchased)		16,000	16,000

The various columns of journal are explained in details below:

### **Date**

This column is used to write the date of the business transaction. Different date formats are used in different countries. Different formats of date are: 15.03.1981, 03.15.1981, 15 March 1981 etc.

### **Particulars or Details Column**

In this column the names of the two connected accounts are written in two consecutive lines - in the first line the name of account debited and in the second line the name of account credited. While the name of account debited always placed close the the left hand margin line, the name of account credited is commenced a short distance away from the margin line. This arrangement will show clearly which account is debited and which credited. This also shows that credit amount is placed on the right side of debit amount. The word "Dr" is used at the end of the name of account debited. It is not necessary to place the word "Cr." after the name of the credited account, because if one account is Dr. it follows that the other account must be Cr. Below the names of the two accounts, i.e. in the third line narration is written usually within a bracket. According to tradition, narration is written starting with a word "being". But modern practice is not to use this word. In most of countries even in Great Britain using the word "To" at the beginning of the name of account credited has become out-dated. So, here it has not been used too. But it is optional for the students.

### **Ledger Folio (L.F)**

The page numbers of the ledger where the two concerned accounts have been posted, are written in this column against the name of each account. This will help locating easily the

two concerned accounts from the ledger. On the other hand, when a transaction is posted to ledger, the concerned folio number of the ledger is written in this column. Thus if a folio number stands written in this column, it will mean that the transaction has already been posted to ledger.

### **Amount**

The debit amount is written in the first "amount" column against the name of account debited and the credit amount in the second "amount" column against the name of account credited.

## **2.5 Ledger**

### **2.5.1 Definition and Explanation of Ledger**

The book in which accounts are maintained is called ledger. Generally, one account is opened on each page of this book, but if transactions relating to a particular account are numerous, it may extend to more than one page. All transactions relating to that account are recorded chronologically. From journal each transaction is posted to at least two concerned accounts - debit side of one account and credit side of another account. Remember that, if there are two accounts involved in a journal entry, it will be posted to two accounts in the ledger and if the journal entry consists of three accounts (compound entry) it will be posted to three different accounts in the ledger. The process of transferring information from journal to ledger accounts is known as posting. The goal of all transactions is ledger. Ledger is known as the destination of entries in journal but it must be remembered that transactions cannot be recorded directly in the ledger - they must be routed through journal. So, the books in which all the transactions of a business concern are finally recorded in the concerned accounts in a summarized form is called ledger.

### 2.5.2 Standard Form of Ledger Account

To understand clearly as to how to write the accounts in ledger, the standard form of an account is given below with two separate transactions:

Date	Particulars	J.R	Amount	Date	Particulars	J.R	Amount
2005 Dec. 17	Cash A/C		1,200	2005 Dec. 17	Purchases A/C		2,000

It appears that each account in the ledger has two similar sides - left hand side is called debit side (briefly Dr.) and right hand side (briefly Cr.) side. Now a days these two words are not used, because it is obvious that the left hand side is debit side and right hand side is credit side.

### 2.5.3 Posting Procedure

Transferring information i.e. entries from journal to ledger accounts is called posting. The procedure of posting from journal to ledger is as follows:

1. Locate the ledger account from the first debit in the journal entry.
2. Record the date in the date column on the debit side of the account. The date is the date of transaction rather than the date of the posting.
3. Record the name of the opposite account (account credited in entry) in the particular (also know as reference column, description column etc) column.
4. Record the page number of the journal in the journal reference (J.R) column from where the entry is being posted.
5. Record the amount of the debit in the "amount column"
6. Locate the ledger account for the first credit in the journal and follow the same procedure.

### 2.5.4 Balancing An Account

The difference between the two sides of an account is its balance. The balance is written on the lesser side to make the two sides equal. The process of equalizing the two sides of an account is known as balancing.

The rules for balancing an account are stated as below:

1. Add up the amount columns of both the sides of an account and write the totals in a separate slip of paper.
2. Find out the difference of the two totals.
3. Write down the difference on the lesser side of the account.
4. Now total up both the sides and write the totals and draw double lines under them.
5. Again write the difference on the opposite side below the double line.

If the debit side of an account is heavier, its balance is known as debit balance. and if the credit side of an account is heavier its balance is known as credit balance. If the two sides are equal, that account will show zero balance. The rules for determining the balance is as follows:

Total debit	=	More than total credit	=	Debit balance
Total credit	=	More than total debit	=	Credit balance
Total debit	=	Total credit	=	Nil balance

It may be noted that at the time of balancing an account debit balance is placed on the credit side and credit balance on debit site. This balance is known as closing balance. What is closing balance in this year, is the opening balance of the next year.

## 2.6 What is a Trial Balance?

Trial Balance is a list of closing balances of ledger accounts on a certain date and is the first step towards the preparation of financial statements. It is usually prepared at the end of an accounting period to assist in the drafting of financial statements. Ledger balances are segregated into debit balances and credit balances. Asset and expense accounts appear on the

debit side of the trial balance whereas liabilities, capital and income accounts appear on the credit side. If all accounting entries are recorded correctly and all the ledger balances are accurately extracted, the total of all debit balances appearing in the trial balance must equal to the sum of all credit balances.

### **2.6.1 Purpose of a Trial Balance**

- Trial Balance acts as the first step in the preparation of financial statements. It is a working paper that accountants use as a basis while preparing financial statements.
- Trial balance ensures that for every debit entry recorded, a corresponding credit entry has been recorded in the books in accordance with the double entry concept of accounting. If the totals of the trial balance do not agree, the differences may be investigated and resolved before financial statements are prepared. Rectifying basic accounting errors can be a much lengthy task after the financial statements have been prepared because of the changes that would be required to correct the financial statements.
- Trial balance ensures that the account balances are accurately extracted from accounting ledgers.
- Trial balance assists in the identification and rectification of errors.

## **2.7 Income Statement**

### **2.7.1 What it is**

The income statement is one of the three primary financial statements used to assess a company's performance and financial position (the two others being the balance sheet and the cash flow statement). The income statement summarizes the revenues and expenses generated by the company over the entire reporting period.

### **2.7.2 How it works/Example**

The income statement is also known as a profit and loss (P&L) statement, statement of earnings, statement of operations or statement of income.

The basic equation on which an income statement is based is:

Revenues – Expenses = Net Income

All companies need to generate revenue to stay in business. Revenues are used to pay expenses, interest payments on debt and taxes owed to the government. After the costs of doing business are paid, the amount left over is called net income. Net income is theoretically available to shareholders, though instead of paying out dividends, the firm's management often chooses to retain earnings for future investment in the business.

## **2.8 Balance Sheet**

The balance sheet is a financial report that lists a company's assets (what it owns), liabilities (what it owes to others), and equity.

### **2.8.1 How it works/Example**

The first section of the balance sheet gives a detailed list of a company's assets, including long-term assets (such as real estate and machinery), current assets (anything that can easily be converted to cash in less than a year), and cash.

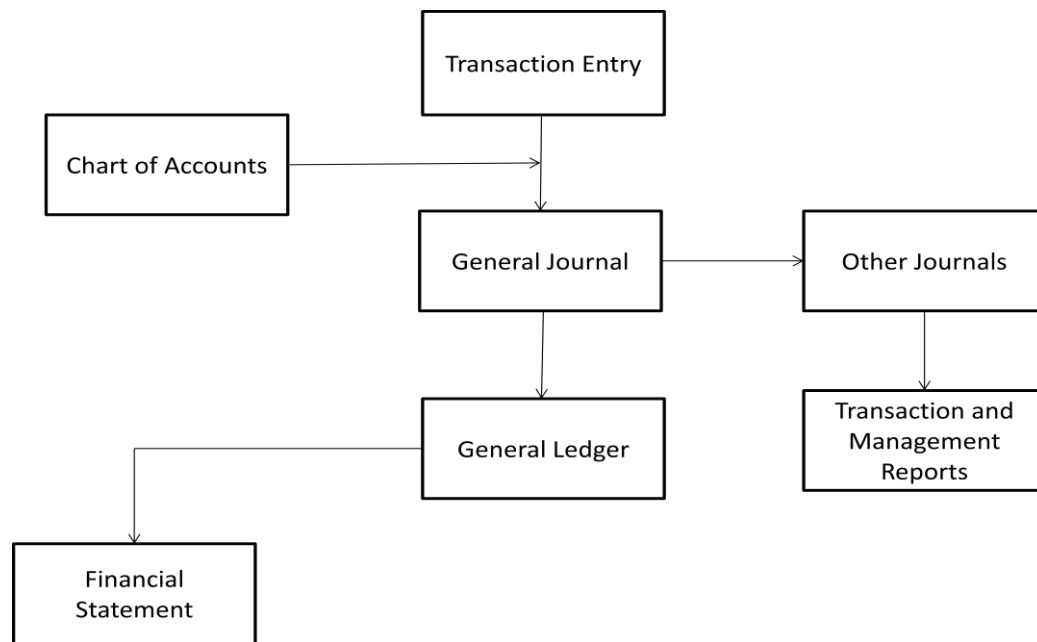
The second section goes over the company's liabilities, or what it owes others. This is always an important section for investors to read because even the most stable of companies will face problems if it has an unusually high amount of debt on its books (especially if it has to pay it back sooner rather than later).

The third section outlines stockholders' equity, and provides information on common and preferred stock, retained earnings, and capital surplus.

## **2.9 Manual Accounting System and its Shortcomings**

Briefly, a system is a set of interdependent elements that together accomplish specific objectives. Manual accounting system is an information system and Romney & Steinbart defined an information system as an organized means of collecting, entering, and processing data and storing, managing, controlling, and reporting information so that an organization can achieve its objectives and goals. Tanis and Dalci emphasised that, information system has the

following components; Goals and objective, Inputs, Output, Data storage, Instructions and Procedure, Users, Control and Measures. Accounting systems as an “information system is a man-made system that generally consists of an integrated set of computer-based and manual components establish to collect, store, and manage data and to provide output information to users” ( Gelinas et al).



**Figure 2.1:** A Manual Accounting System Model

Manual accounting implies that employees perform the whole accounting cycle manually on a periodic basis: they draft trial balances, journalize transactions, and prepare financial statements. Extensively, Waterfield and Ramsing (1998), highlighted that, accounting system can be a simple manual one based on the general journal (where transactions are recorded chronologically as debits and credits), general ledger (where the activity from the general journal is summarized by account number), and other journals required to manage the business, such as purchase, payment, sales, receipts, and payroll journals. (Because of the expense of maintaining multiple manual journals, institutions typically do not prepare all of these other journals.) They further stated that, a manual accounting system typically includes at least the following and thus presented in fig 2.1 below: Chart of accounts, General journal, General ledger, Subsidiary ledgers (accounts receivable, inventory, and fixed assets), Transaction reports and financial statements[5].

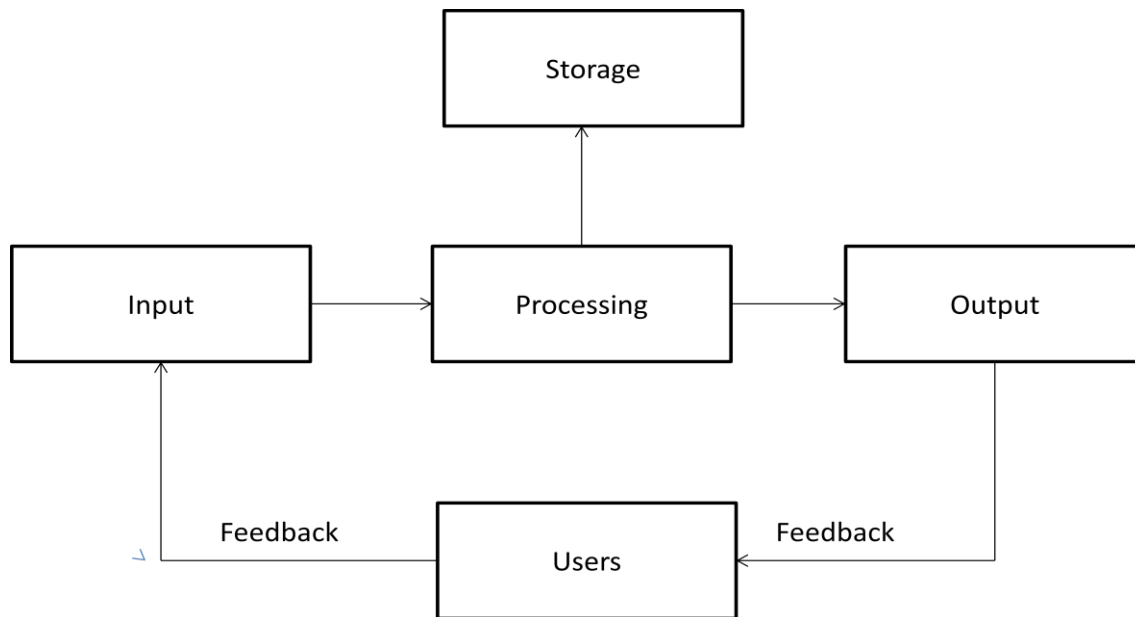
## **2.10 Computerized Accounting System**

In a study, Nash et al argued that with the improvements in technology, information systems have been computerized. Improvements in this technology have replaced manual bookkeeping systems with computerized ones, hence, accounting information systems that were previously performed manually are now performed by computers in most companies. While accounting systems have been around for centuries, the introduction of business technology and Computerized Accounting Systems radically changes the playing field. Lately, Vitez reviewed that paper ledgers, manual spreadsheets and hand-written financial statements have all been translated into computer systems that can quickly present individual transactions into financial reports. Computerized Accounting Systems follow the same logic of journal, ledgers, reports and statements in a manual system. Computerized systems simply consolidate posting functions and other basic tasks into a "behind the scenes" system. Companies can also generate reports and financial statements easier, allowing for better performance management reviews.

Computerized Accounting System is therefore a computer based system which combines accounting principles and concepts as well as the concept of information system to record, process, analyze and produce financial information to its users for making economic decisions.

The definition of a Computerized Accounting System from above shows that a Computerized Accounting System has the following components with an illustrative diagram in figure 2.2





**Figure 2.2:** A Computerized Accounting System Model

**Input:** Data inputs are the facts that are collected and processed by the information system. Data input includes capturing data from a source document such as a sales order or purchase order.

**Processing:** In order to produce useful and meaningful information, the data captured must be processed and organized into a useful form.

**Output:** Output is the meaningful and useful information produced by the information system. It is usually presented in the form of a report.

**Feedback:** After the information has been presented in the form of a report, there is the need for a feedback. Feedback tends to serve as a source of input and also a control measure in the information system.

**Storage:** It serves as the repository of relatively permanent data maintained over an extended period of time.[5]

## 2.11 Importance Of Computerized Accounting Systems

Computerized Accounting Systems are important to businesses in various ways. The use of computers is time-saving for businesses and all financial information for the business is well organized. Some benefits of Computerized Accounting Systems are:

### **Time and Cost Savings**

Using a Computerized Accounting Systems saves companies time and money. The use of a computer makes inputting accounting information simple. Transactions are entered into the system and the system processes and posts transactions accordingly. Computerized Accounting Systems reduce staff time preparing accounts and reduce audit expenses as records are neat, up-to-date and accurate. Better use is made of resources and time; cash flow should improve through better debt collection and inventory control. More importantly, the system helps present financial reports on time to aid in the economic decision making process of external users.

### **Organization and Accuracy**

A Computerized Accounting System enables businesses to stay organized. When information is entered into the system, it makes finding the information easy. Employees can look up any financial information whenever it is needed. There is less room for errors as only one accounting entry is needed for each transaction rather than two (or three) for a manual system. The accounting records are automatically updated and so account balances (e.g. customer accounts) will always be up-to-date.

### **Storage and Speed**

Storing information is vital to a business. After information is entered into the system, the information is stored indefinitely. Companies perform backups on the system regularly to avoid losing any information. The introduction of Computerized Accounting Systems provides the ability to see the real-time state of the company's financial position.

### **Distribution**

Computerized Accounting Systems allow companies to distribute financial information easily. Financial statements are printed directly from the system and are distributed internally and externally to those needing the information. Reports can be produced which will help management monitor and control the business, for example the aged debtors analysis will show which customer accounts are overdue, trial balance, trading and profit and loss account and balance sheet. In effect, Computerized accounting systems enable financial statements to be prepared and presented to meet the relevance and faithful representation criteria of financial statements.

## Chapter 3

### Methodology

#### 3.1 Proposed Method

Our proposed system will combine the following steps:

**Data Entry:** Processing presumes data entry. The operator will input the data in the journal entry form defining debit/credit with the amount.

**Data Validation:** It ensures the accuracy and reliability of input data by comparing the debit, credit on both side in ledger and trial balance.

**Processing:** The processing of data occurs almost instantaneously. Data are directed to the desired location as defined in the code.

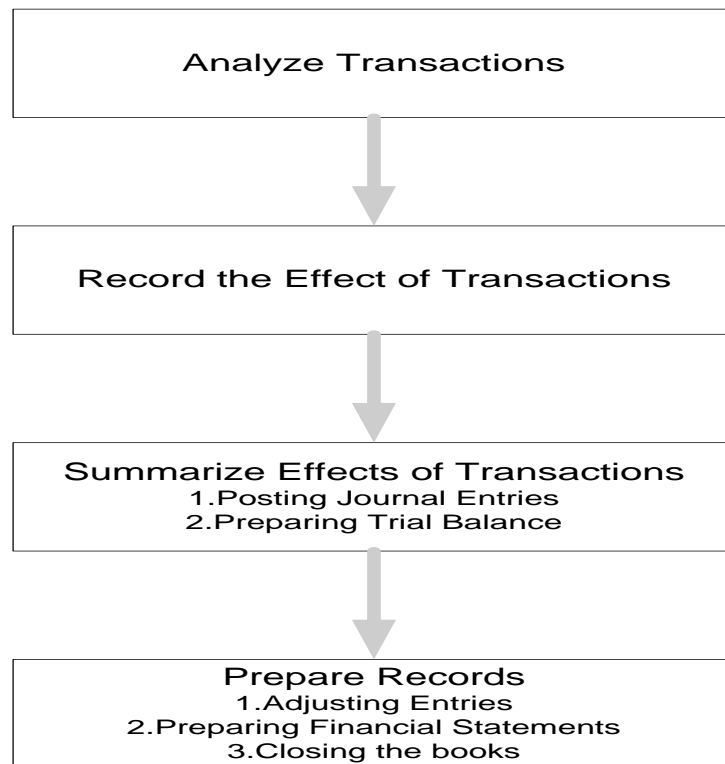
**Storage:** Processed actions result into financial transaction data which are stored in transaction database. This makes it absolutely clear that only valid transactions are stored in the database.

**Information:** The stored data is processed making use of the Query facility to produce desired information.

**Reporting:** Reports can be prepared on the basis of the required information content according to the decision usefulness of the report.

#### 3.2 The Accounting Process

The key objective of our proposal is to implement a computerized accounting system. Figure 3.1 shows the general flow of the accounting process. The four basic steps involved are analyze transactions, record the effects of transaction, summarize the effects of transactions and prepare records. This procedure is neutral; this means that the steps involved can be applied both in manual and technology based.



**Figure 3.1:** The Accounting Process

The first step is the analyzing of transactions, the transaction must be known to be financial in nature, recordable and non-recordable transactions are separated. In this step the transaction is being analyzed on how it affects the accounting equation. Source documents such as invoices, orders, checks are helpful in this stage. The second step is to record the effect of the transactions. Transactions are recorded using journal entries.

These journal entries are the accountant's way of recording the effect of both simple and complex business transactions. Journals provide a chronological record of all transactions of a business. They show the dates of the transactions, the amounts involved, and the particular accounts affected by the transactions. Sometimes a detailed description of the transaction is also included. It is also known as the books of original entries. The third step is to summarize the effects of transaction, under this step, the journal entries will be posted to the ledger and a trial balance will then be prepared. Once transactions have been analyzed and recorded in a journal, it is necessary to classify and group all similar items. This is accomplished by the bookkeeping procedure of posting all the journal entries to appropriate accounts. All accounts are

maintained in an accounting record called a ledger. A ledger is also referred to as the book of accounts. The next step is to determine the total balance of each account. After the account balances have been determined, a trial balance is usually prepared. A trial balance lists each account with its debit or credit balance. The fourth step is the preparation the reports, this includes adjusting entries, preparation of financial statements and closing of the books. There will be recording and posting of the some adjusting entries that is applicable for the period. Then the trial balance will again be recomputed. From the data in the trial balance, the financial statements are then prepared. This includes the statement of financial position, income statement, cash-flow statement and the notes. In the basis of accounting process we create block diagram .Block diagram has four parts. First part is data input second step is preprocessing part. In preprocessing part two tasks done here. Firstly check data validation and second entry in journal. Third step is processing part. Here data is processed from database and create reports of ledger, trial balance and final account. Final account consist of two parts, such as: income statement and balance sheet. Income statement consists of two parts income and expenditure, we compare this two parts and determine whether a company stays in profit or loss. Balance sheet consists of two parts liabilities and assets. From which we get a clear idea of liabilities and assets of a company. Forth part is the result display part. From which we get desired output. The block diagram is given below in fig 3.2



**Figure 3.2:**Block Diagram of Implementation Process

### 3.3 Implementation process

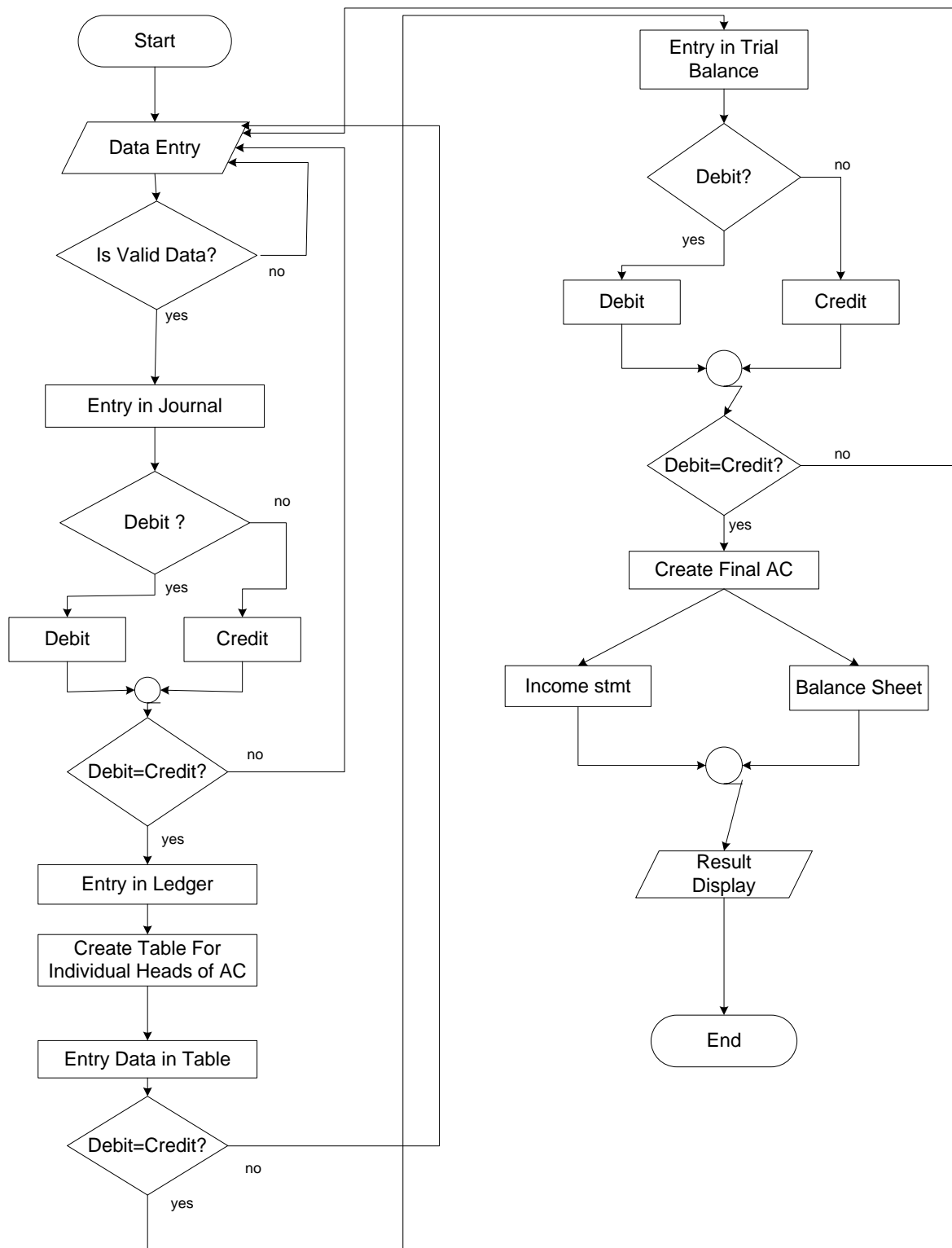
To implement our project we designed a methodology which covered step by step procedure is shown in fig:3.3. At first data entry in temporary database. Then check data validity, if data

is valid then data entry in journal. Journal consists of debit and credit, if the data is debit then it save in the debit part else data go to credit part. If debit balance is equal to credit part then it create ledger for individual account. If debit and credit balance of all accounts are equal, then create trial balance, again check debit and credit balance, if equal then it create final account. From which we get desired output and determine whether a company stays in profit or losses. So the input interface of our accounting system:

- Main A/c
- Sub A/c
- A/c Opening Balance
- Journal Entry
- Invoice
- Account Transaction Date Range Setting
- Accounts Final Component
- Accounts Type

And the output interface of our accounting system:

- Account Details Transaction
- Accounts List
- Current Trail Balance Details
- Current Trail Balance Summery
- Final A/c Details on date range
- Final A/c Summery on date range
- Trail Balance Details on date range
- Trail Balance Summery on date range
- Periodic Final A/c Summery
- Periodic Final A/c Details
- Periodic Trail Balance Summery
- Periodic Trail Balance Details
- Periodic A/c Transaction Summery
- Periodic A/c Transaction Details



**Figure 3.3:**Flow Diagram of Implementation Process



# Chapter 4

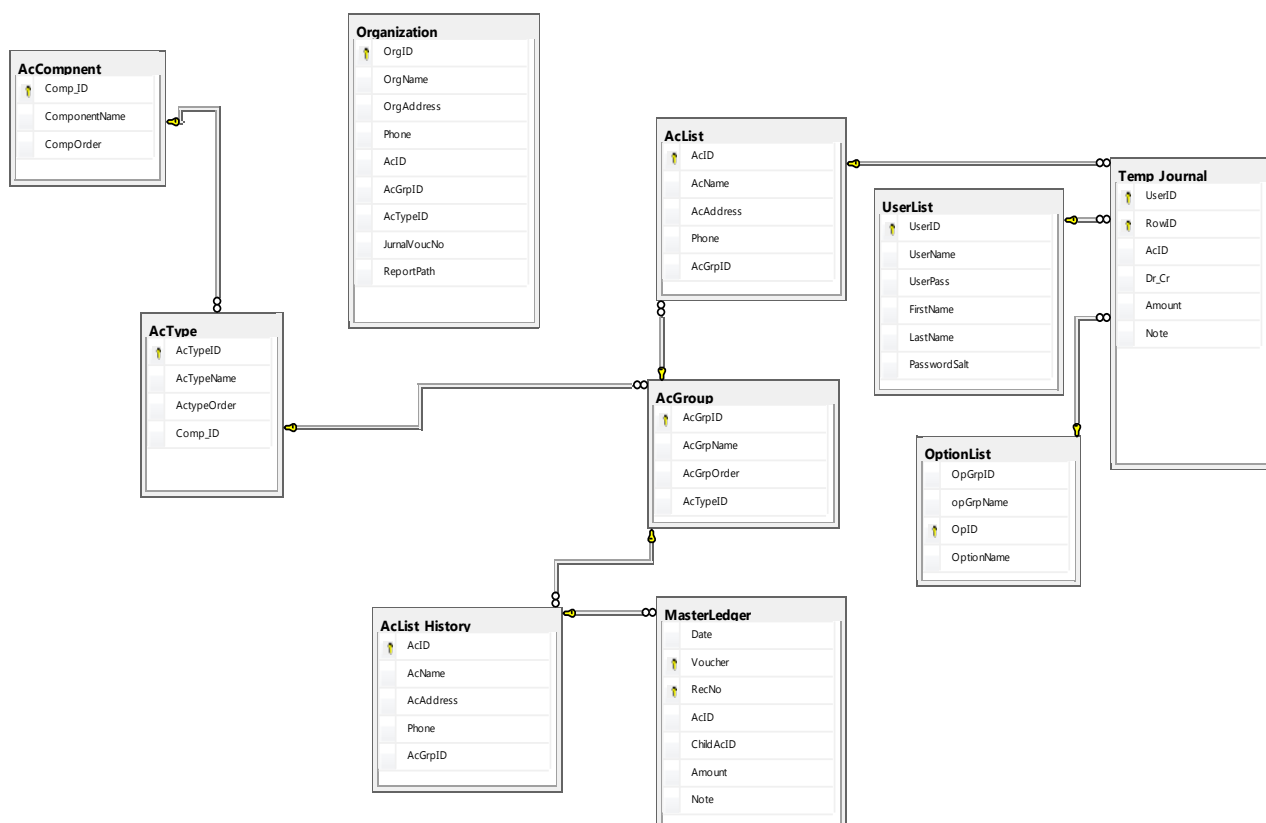
## Implementation

We divided the whole system into four parts;

- a) Database setup
- b) Data entry
- c) Security
- d) Output generation

### 4.1 Database setup

Accounting software describes a type of application software that records and processes accounting transactions within functional modules. During this stage, final products take place in physical form from the virtual form of design. Here is the database diagram:

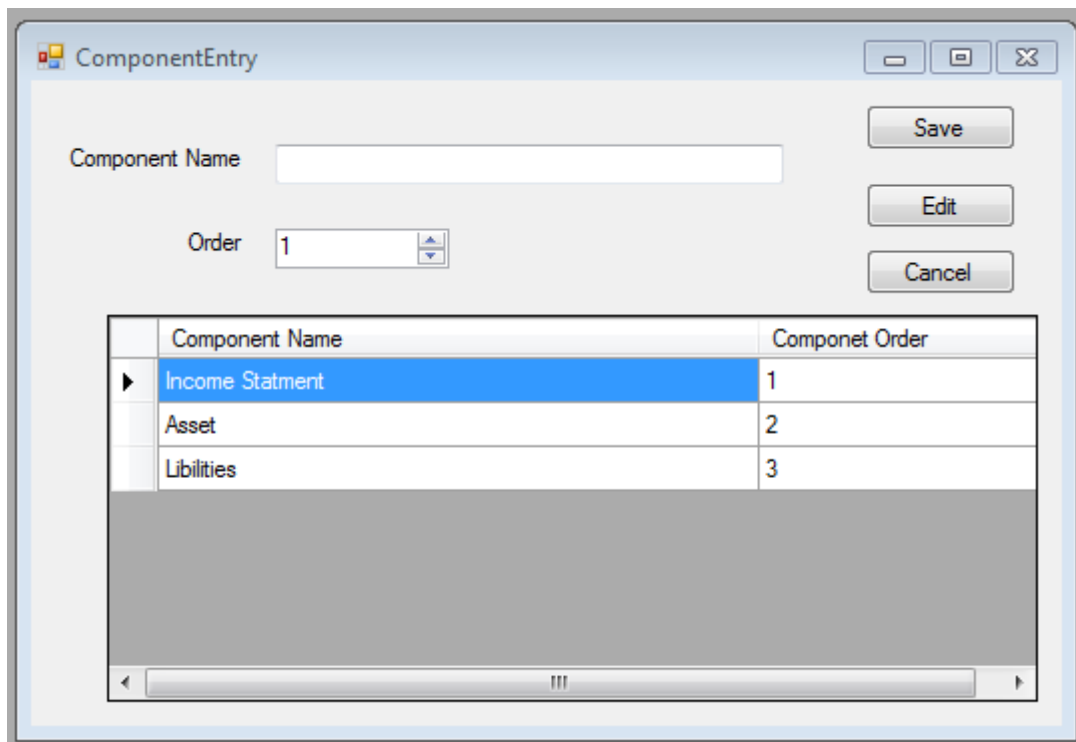


**Figure 4.1:** Database Diagram

To develop our database we maintained a tree strategy. Account component is the root of the tree, account type is the children of the account component, then account group is under the account type. Under the account group accounts are held.

As account type is dependent on account component, so the primary key of account component has become the foreign key of account type. Similarly account group is dependent on account type, so the primary key of account type has become the foreign key of account group. At last account list is dependent on account group, so the primary key of account group has become the foreign key of account list. AcList history table is back up for Aclisttable. Temp journal table is used for journal entry and master ledger table is used for ledger. To predefine which account stay under which account group or which group stay under which account type or which type is under the which component. We have worked on three steps.

#### 4.1.1 1st step(Account component)



The screenshot shows a window titled "ComponentEntry" with a light blue border. At the top right are three small icons: a minus sign, a maximize button, and a close button. Below these are three buttons: "Save", "Edit", and "Cancel". The main area contains two input fields: "Component Name" with a text box and "Order" with a spinner box set to "1". Below these is a table with two columns: "Component Name" and "Componet Order". The table has three rows: "Income Statment" (highlighted in blue), "Asset", and "Libilities". The table is contained within a larger frame with a scrollbar at the bottom.

Component Name	Componet Order
Income Statment	1
Asset	2
Libilities	3

**Figure 4.1.1: Account Component**

Account component consists of three components such as income statement, asset and liabilities. Here the order increase automatically. We can easily save, edit and cancel in this component entry screen

#### 4.1.2 2nd step(Account type)

The screenshot shows a window titled "AcTypeEntry" with a "Component List" dropdown set to "Income Statment". Below it is an "A/c Type Name" text field and an "Order" spinner set to "1". To the right are "Save", "Edit", and "Cancel" buttons. At the bottom is a table with the following data:

	Component Name	A/c Type Name	Order
▶	Income Statment	Income	1
	Income Statment	Expense	2
	Asset	Current asset	3
	Asset	Fixed Assets	4
	Libilities	current liabilities	5

**Figure 4.1.2:** Account Type

As account type dependent on account component, so every account component holds various account type such as income statement account component holds two account type-income and expense. Similarly asset holds current asset and fixed asset. As usual liabilities holds current liabilities. Here the order increase automatically. We can easily save, edit and cancel operation in this AcTypeEntry screen.

#### 4.1.3 3rd step(Account group)

As account group dependent on account type, so every account type holds various account group such as income holds various income, stock and expense holds salary and wages, office expense etc. Here the order automatically increases. We can easily perform save, edit and cancel operation in this AcGroupEntry screen.

	A/c Type Name	A/c Group Name	Order
▶	Income	Various income	1
	Income	Stock	2
	Expense	Salary and wages	3
	Expense	Office expences	4
	Current asset	Cash	5
	Current asset	Bank	6
	Current asset	Sundry detor	7

**Figure 4.1.3: Account Group**

## 4.2 Data entry

There are two parts in data entry steps such as : A/c name entry and journal entry. Here in this two fields all account are inserted. Every single account dependent on account group. So in A/c entry screen, there is a combo box of Ac group list. We can select any account group then put the account name in account field. Every single account may has or has not account address and phone number. We can easily perform save, edit and cancel operation in this Account name Entry screen.

### 4.2.1 A/c name entry

The screenshot shows a window titled "AccountEntry" with a standard Windows interface (minimize, maximize, close buttons). Inside the window, there are four input fields on the left, each with a label: "Ac Group List:", "Ac Name:", "Ac Address:", and "Phone:". The "Ac Group List:" field is a dropdown menu currently showing "Various income". To the right of these fields are three buttons: "Save", "Edit", and "Cancel". Below the input fields is a table with three columns: "A/c Name", "A/c Address", and "Phone". The table contains seven rows of account names: "Fees revenue", "Office supplies", "Rent expense", "Utility expense", "Cash in hand", "Accounts receivable", and "Prepaid rent". The "Fees revenue" row is highlighted in blue. The table has a scrollbar on the right side.

A/c Name	A/c Address	Phone
Fees revenue		
Office supplies		
Rent expense		
Utility expense		
Cash in hand		
Accounts receivable		
Prepaid rent		

**Figure 4.2.1:** Account Name Entry Screen

### 4.2.2 Journal entry

In journal entry screen, at first we have to select a date from date field. Then auto voucher is automatically generated by computer. If all account check box is selected then all account will be present in A/c list combo box, otherwise we have to select account group and then fill up account list. Then we have to select debit or credit and put the amount. If we press ok button then data will be shown in data grid view which also shows total debit amount and total credit amount and total balance. Data are saved in temporary journal table in database. At last if we press save transaction button then the software checks the one debit against multiple credit transaction or one credit against multiple debit transaction and also checks if the total balance is zero or not. If total balance is zero then data is saved in the master ledger table of database.

The screenshot shows a window titled "JournalEntry" with the following elements:

- Date:** 04/09/2015 (with a calendar icon)
- Auto Voucher:** 11
- A/c Group:** A dropdown menu.
- All Accounts:** A checked checkbox.
- A/c List:** A dropdown menu.
- Dr/Cr:** A dropdown menu set to "Debit".
- Amount:** A text input field.
- Buttons:** "Save Transaction", "Ok", and "Delete".
- Table:** A table with columns "AcID", "A/c Name", "Dr/Cr", and "Amount". The table body is currently empty.
- Totals:**
  - Tot. Dr.: 0.00
  - Tot. Cr.: 0.00
  - Balance: 0.00

**Figure 4.2.2:** Account Journal Entry Screen

## 4.3 Security

There are four parts in security such as: create user entry, edit user, change personal password and login.

### 4.3.1 User Entry

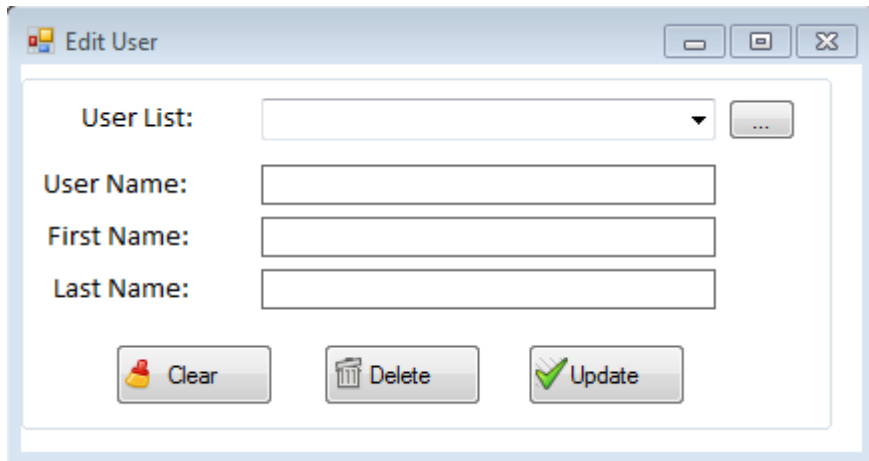
The screenshot shows a window titled "New User" with the following elements:

- User Name:** A text input field.
- First Name:** A text input field.
- Last Name:** A text input field.
- New Password :** A text input field.
- Confirm Password :** A text input field.
- Buttons:** "Cancel" (with a red prohibition icon) and "Save" (with a green checkmark icon).

**Figure 4.3.1:** User Entry Screen

- Designed a system which gives access only to those persons who are selected by the admin.
- Here we give user name, first name and last name.
- When new password and confirm password are matches then it will be saved otherwise not.

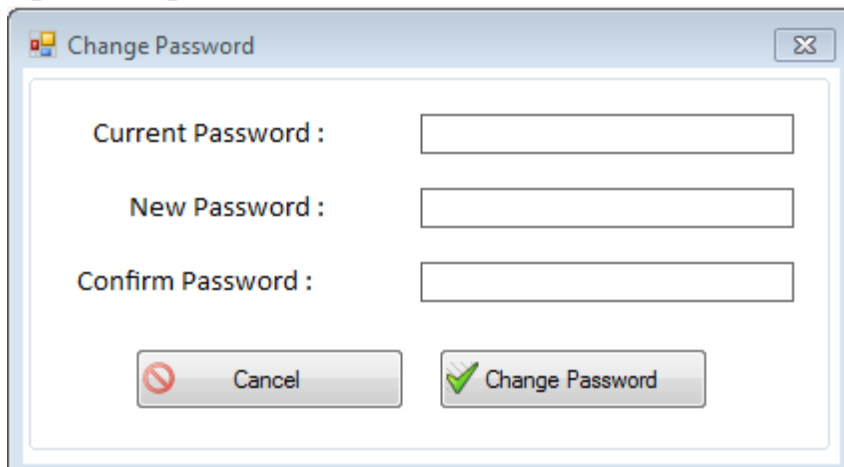
#### 4.3.2 Edit User



**Figure 4.3.2:** Edit User Entry Screen

- User List combo box will show all the users information which are saved in the database.
- Then give new user name, first name and last name .
- At Last we can update, delete or clear the user information.

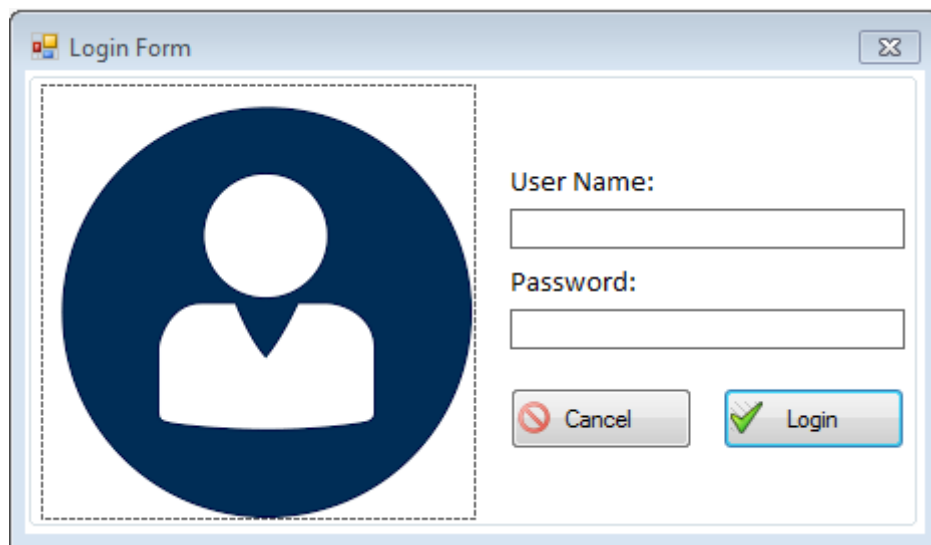
#### 4.3.3 Change personal password



**Figure 4.3.3:** Change Personal Password Entry Screen

- To change personal password this form is necessary. At first we have to select user from the combobox whose password we want to change.
- Then we have to write the current password, new password and confirm password and then press change password button to change a user's password.

#### 4.3.4 Login Form



**Figure 4.3.4:** Login Form Screen

Finally we have created login form for security purpose Thus only valid user can access the system.

We have defined in the program that the password must be seven character long & there must be an uppercase and a lowercase letter. Use of number and nonalpha character and number is optional. When creating an user, these conditions are checked. If these conditions are fulfilled, then the username and password are accepted and stored in the database.

Moreover we have applied SHA-1 algorithm to increase the security .According to this algorithm, the program generates a hash for the given plain text password value and returns abase64-encoded result. Before the hash is computed, a random salt is generated and appended to the plain text. This salt is stored at the end of the hash value, so it can be used later for hash plaintext value to be hashed. The hash algorithms used in the program

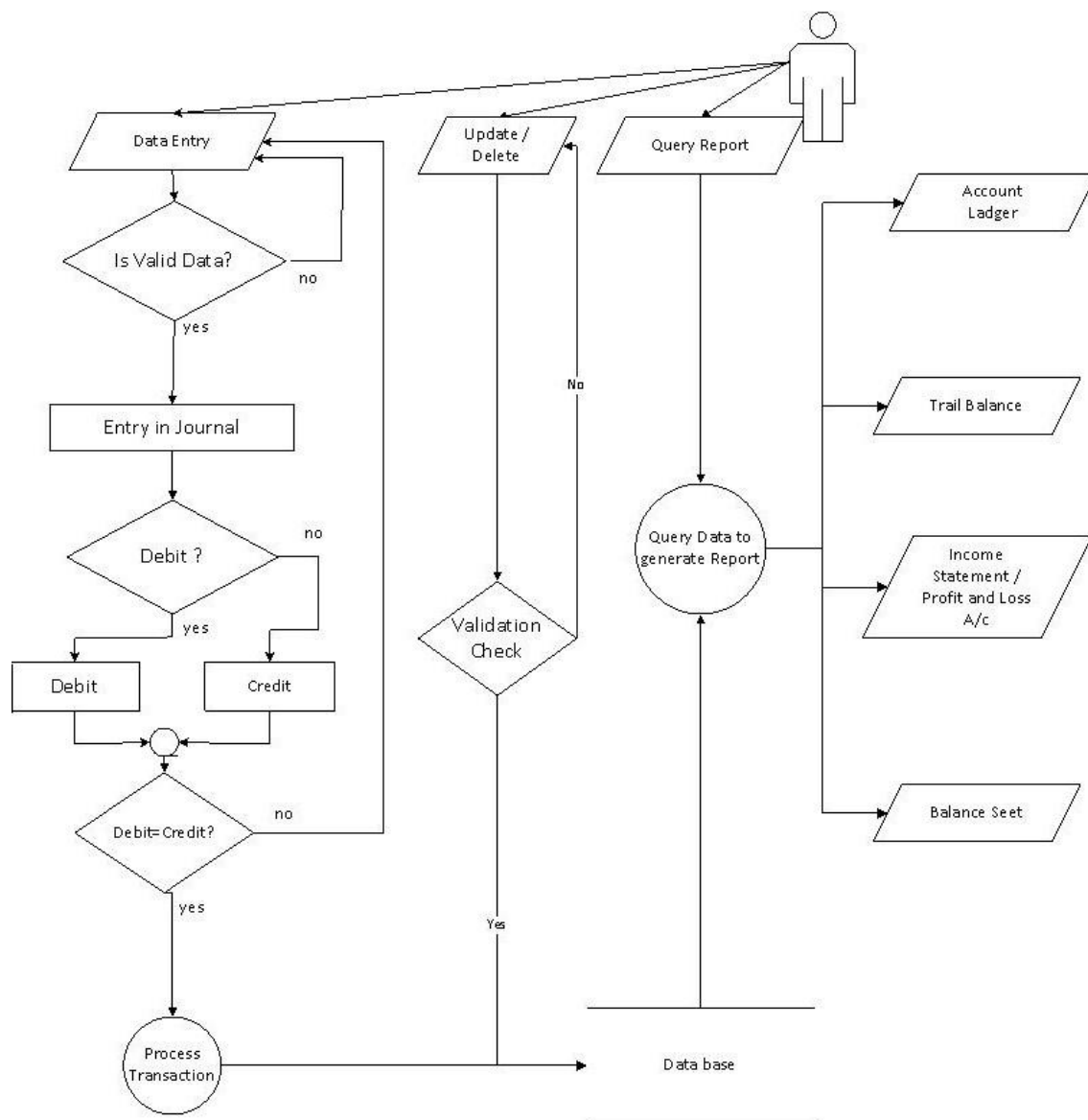


are:"SHA1","SHA256", "SHA384", "SHA512" and "MD5" (if any other value is not specified, SHA1 hashing algorithm will be used). This value is case-insensitive. The salt bytes parameter can be null, in which case a random salt value will be generated.

## 4.4 Experimental Results and Analysis

### 4.4.1 Output generation

Our task is to generate automatically Account Ledger, Trial Balance, Income Statement or profit and loss account and Balance Sheet from the Journal. From the journal, data are saved in the database and we independently generate those reports.



**Figure 4.1:** Output generation Process

This above figure 4.1 carried out the output generation process of our work. Admin can entry data, update or delete data, and query data. Validation is checked for entry and update data. Admin can also query data to generate report from database.

RAHAT-PC.Final_Ac...dbo.MasterLedger					
RAHAT-PC.Final_Account - dbo.AcList X RAHAT-PC.					
	AcID	AcName	AcAddress	Phone	AcGrpID
▶	0001	Fees revenue			001
	0002	Office supplies			004
	0003	Rent expense			004
	0004	Utility expense			004
	0005	Cash in hand			005
	0006	Accounts recei...			007
	0007	Prepaid rent			007
	0008	Office equipme...			008
	0009	Accumulated d...			008
	0010	Accounts paya...			009
	0011	Notes payable			009
	0012	Unearned fees			009
	0013	Salaries expense			003
	0014	Capital_chowd...			010
	0015	withdrawals-ch...			010
	0016	Repair supplies			004
	0017	Sonali Bank			006
*	NULL	NULL	NULL	NULL	NULL

**Figure 4.2:** Database Account List

Figure 4.2.carried out the total database account list. Account Id is generated automatically by program. AcName holds the account name. AcAddress and Phone is optional column in database. AcGrpId holds the account group id. If any new account is entered by the admin it holds in database account list. In this way we can insert peta byte of accounts in database. When we entered the above account in the journal entry as a debit or credit account with amount then it saved temp journal table in database. When this data satisfied data validation then it saved in master ledger. The following figure shows the master ledger table in the database.

RAHAT-PC.Final_Ac...dbo.MasterLedger		RAHAT-PC.Final_Ac...dbo.MasterLedger					
	Date	Voucher	RecNo	AcID	ChildAcID	Amount	Note
	2015-08-31	10	1	0013	0005	20000.0000	used in salary s...
	2015-08-31	10	2	0005	0013	-20000.0000	used in salary s...
	2015-08-12	4	1	0014	0008	-3100.0000	purchase office...
	2015-08-12	4	2	0008	0014	3100.0000	purchase office...
	2015-08-12	4	3	0014	0015	-5500.0000	personal useA...
	2015-08-12	4	4	0015	0014	5500.0000	personal useA...
	2015-08-12	4	5	0014	0013	-8000.0000	Paid salariesAm...
	2015-08-12	4	6	0013	0014	8000.0000	Paid salariesAm...
	2015-08-12	4	7	0014	0007	-2000.0000	From Cash Acc...
	2015-08-12	4	8	0007	0014	2000.0000	From Cash Acc...
	2015-08-28	5	1	0005	0006	25000.0000	Reecived form ...
	2015-08-28	5	2	0006	0005	-25000.0000	Reecived form ...
	2015-08-28	5	3	0005	0007	15000.0000	Reecived form ...
	2015-08-28	5	4	0007	0005	-15000.0000	Reecived form ...
▶	2015-08-28	6	1	0005	0008	-5000.0000	
	2015-08-28	6	2	0008	0005	5000.0000	
	2015-08-28	6	3	0005	0002	-6000.0000	
	2015-08-28	6	4	0002	0005	6000.0000	
	2015-08-28	6	5	0005	0003	-14000.0000	
	2015-08-28	6	6	0003	0005	14000.0000	
	2015-08-28	6	7	0005	0012	-8000.0000	
	2015-08-28	6	8	0012	0005	8000.0000	
	2015-08-28	7	1	0005	0001	100000.0000	
	2015-08-28	7	2	0001	0005	-100000.0000	

**Figure 4.3:** Database Masterledger table

In figure 4.3.master ledger table , Date express when date transaction occur. Voucher is automatically generated by program. AcId express the parent accounts and ChildAcId express the child account. Then Amount express the debit or credit amount balance. Note describes the particulars description. At last we create a view from the combination of four database set up table. In figure 4.4. Account view Transaction List holds the total summary of the

transaction. After that we can create data set and generate reports independently.

Comp_ID	ComponentN...	AcTypeID	AcTypeName	AcGrpID	AcGrpName	AcID	AcName	ChildAcID
03	Liabilities	005	current liabilities	010	owners equity	0014	capital_chowdh...	0013
01	Income Statment	002	Expense	003	Salary and wages	0013	salaries expense	0014
03	Liabilities	005	current liabilities	010	owners equity	0014	capital_chowdh...	0007
02	Asset	003	Current asset	007	Sundry detor	0007	prepaid rent	0014
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0006
02	Asset	003	Current asset	007	Sundry detor	0006	Accounts recei...	0005
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0007
02	Asset	003	Current asset	007	Sundry detor	0007	prepaid rent	0005
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0008
02	Asset	004	Fixed Assets	008	Furniture and e...	0008	office equipment	0005
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0002
01	Income Statment	002	Expense	004	Office expences	0002	Office supplies	0005
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0003
01	Income Statment	002	Expense	004	Office expences	0003	Rent expense	0005
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0012
03	Liabilities	005	current liabilities	009	Sundry creditor	0012	Unearned fees	0005
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0001
01	Income Statment	001	Income	001	Various income	0001	Fees revenue	0005
02	Asset	003	Current asset	006	Bank	0017	Sonali Bank	0014
03	Liabilities	005	current liabilities	010	owners equity	0014	capital_chowdh...	0017
03	Liabilities	005	current liabilities	009	Sundry creditor	0010	Accounts paya...	0005
02	Asset	003	Current asset	005	Cash	0005	Cash in hand	0010
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

**Figure 4.4:** Account view TransactionList

For output generation we have created report controller form. We can select ledger, trialbalance, income statement and balance sheet from report list combo box of which report we want to see.

**Figure 4.5:** Report Controller

Then we can select account list from account list combo box. We can also select start date and end date and press show report to get the report.

#### 4.4.2 Report generation

<b>Chittagong University of Engineering &amp; Technology</b> Chittagong-4349 031-714911					
<b><u>Account Ledger Report</u></b>					
Account ID: 0005					
Account Name: Cash in hand ( 0005 )					
Period: 27/08/2015 to 31/08/2015					
Date	Particulars	Vouc.No.	Debit	Credit	Balance
31/08/2015	salaries expense ( used in salary salary purpose )	10		20000.0000	-20000.0000
28/08/2015	Accounts receivable ( Reecived form sundry detor )	5	25000.0000		5000.0000
28/08/2015	prepaid rent ( Reecived form sundry detor )	5	15000.0000		20000.0000
28/08/2015	office equipment	6		5000.0000	15000.0000
28/08/2015	Office supplies	6		6000.0000	9000.0000
28/08/2015	Rent expense	6		14000.0000	-5000.0000
28/08/2015	Unearned fees	6		8000.0000	-13000.0000
28/08/2015	Fees revenue	7	100000.0000		87000.0000
27/08/2015	Accounts payable	9		30000.0000	57000.0000

**Figure 4.6:** Account Ledger Report

When we click on the button of show report of ledger against a record, the software generates a ledger form which contains account ID, account name and the time period. In the column date, particulars, voucher no, Debit, Credit and balance are shown. The software can generates ledger of all the accounts.

## Chittagong University of Engineering & Technology

Chittagong-4349

031-714911

### Trail Balance Report

Account Titles	Debit	Credit
0001 - Fees revenue		100,000.00
0002 - Office supplies	6,000.00	
0003 - Rent expense	14,000.00	
0005 - Cash in hand	57,000.00	
0006 - Accounts receiv able		25,000.00
0007 - prepaid rent		13,000.00
0008 - office equipment	8,100.00	
0010 - Accounts payable	30,000.00	
0012 - Unearned fees	8,000.00	
0013 - salaries expense	28,000.00	
0014 - capital_chowdhury		98,600.00
0015 - withdrawals-chowdhury	5,500.00	
0017 - Sonali Bank	80,000.00	
	236,600.00	236,600.00

**Figure 4.7:** Account Trial Balance Report

When we click to show the trial balance, the software generates a trial balance form which contains Account Titles where the inputted accounts are shown. In the column of Debit, Credit the inputted balance are shown and at the end the total values are shown which we find equal. That means the calculation has no error.

<b>Chittagong University of Engineering &amp; Technology</b> Chittagong-4349 031-714911			
<b><u>Income Statement Report</u></b>			
	Account Titles	Amount	
<b>Income</b>			
	0001 - Fees revenue	100,000.00	
	Total Income(s)		100,000.00
<b>Expense</b>			
	0002 - Office supplies	6,000.00	
	0003 - Rent expense	14,000.00	
	0013 - salaries expense	20,000.00	
	0013 - salaries expense	8,000.00	
	Total Expense(s)		48,000.00
<b>Net Income</b>			<b>52,000.00</b>

**Figure 4.8:** Income Statement Report

When we click to show the income statement, the software generates a income statement report form which contains all those Account Titles that are under Income and Expense as income statement is a report that shows Income vs Expense. The total income and expenses are shown and at the end net income is displayed by subtracting expense from income.

## Chittagong University of Engineering & Technology

Chittagong-4349

031-714911

### Balance Sheet Report

	Account Titles	Amount
	<b>Asset</b>	
Current asset		
	005 - Cash	57,000.00
	006 - Bank	80,000.00
	007 - Sundry detor	38,000.00
	<b>Total Current asset(s)</b>	<b>175,000.00</b>
Fixed Assets		
	008 - Furniture and equipment	8,100.00
	<b>Total Fixed Assets(s)</b>	<b>8,100.00</b>
<b>Total Asset(s)</b>		<b>183,100.00</b>
	<b>Liabilities</b>	
current liabilities		
	009 - Sundry creditor	38,000.00
	010 - owners equity	93,100.00
	<b>Total current liabilities(s)</b>	<b>131,100.00</b>
<b>Total Liabilities(s)</b>		<b>131,100.00</b>
Total Owner's Equity		<b>52,000.00</b>
Total Liability & Owner's Equity		<b>183,100.00</b>

**Figure 4.9:** Balance Sheet Report

When we click to show the balance sheet, the software generates a report form which contains all those Account Titles that are under Asset and Liabilities. As we can see it is showing the accounts under current asset and fixed asset which is under Assets and showing the accounts under current liabilities which is under Liabilities. The total asset and total liabilities are shown and after that total owner's equity are shown which is = Total asset-Total liability. After that total liability & owners equity are shown which is = total liability +owners equity.

### 4.4.3 Software Testing

Software testing is the process of evaluating a software item to detect differences between given input and expected output. Also to assess the feature of a software item. Testing



assesses the quality of the product. Software testing is a process that should be done during the development process. In other words software testing is a verification and validation process.

### **Verification**

Verification is the process to make sure the product satisfies the conditions imposed at the start of the development phase. In other words, to make sure the product behaves the way we want it to.

### **Validation**

Validation is the process to make sure the product satisfies the specified requirements at the end of the development phase. In other words, to make sure the product is built as per customer requirements.

### **Testability**

In testability we find these characteristics:

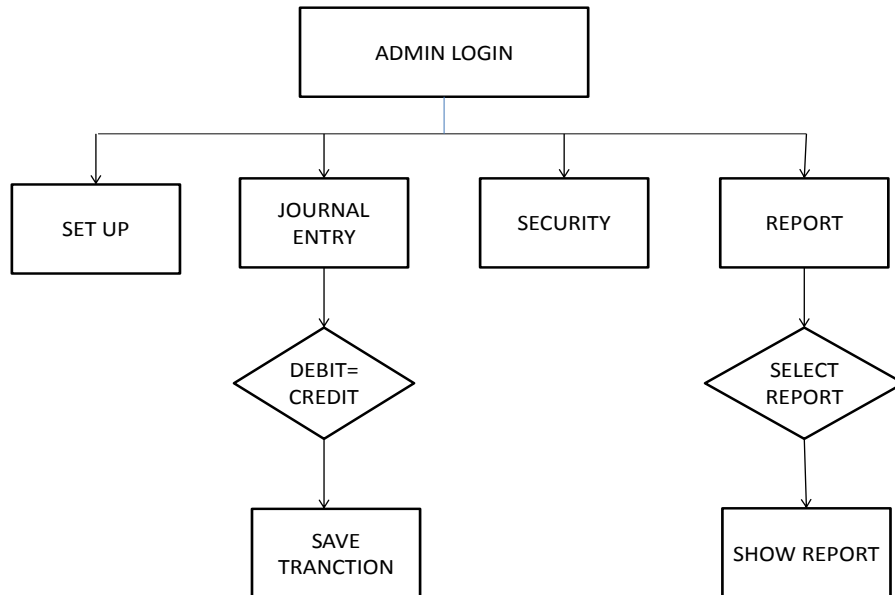
- Operability—it operates cleanly
- Observability—the results of each test case are readily observed
- Controllability—the degree to which testing can be automated and optimized
- Decomposability—testing can be targeted
- Simplicity—reduce complex architecture and logic to simplify tests
- Stability—few changes are requested during testing
- Understandability—of the design

### **Good Test**

In good test we find these characteristics:

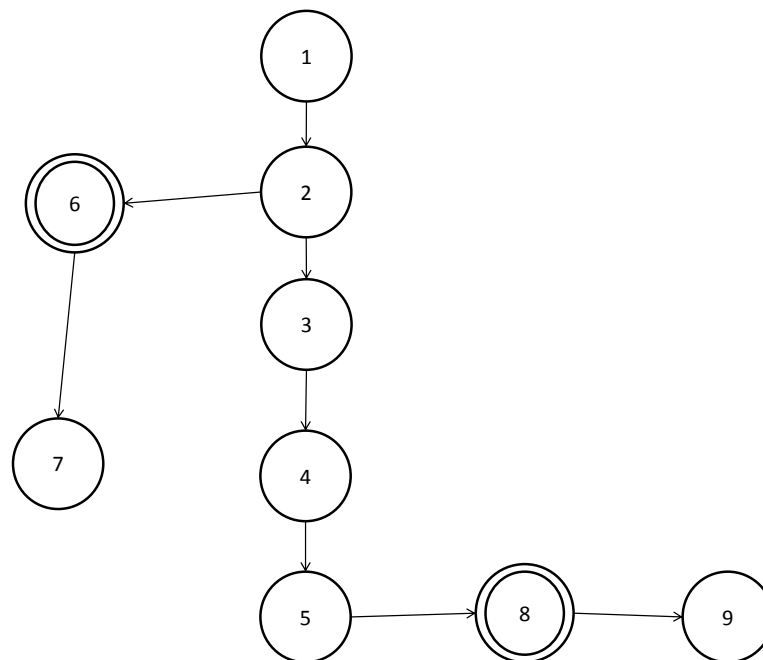
- A good test has a high probability of finding an error
- A good test is not redundant.
- A good test should be “best of breed”
- A good test should be neither too simple nor too complex

## Testing Flow Chart



**Figure 4.10:** Testing flow chart

## Flow Graph



**Figure 4.11:** Testing flow graph

## Cyclometric Complexity

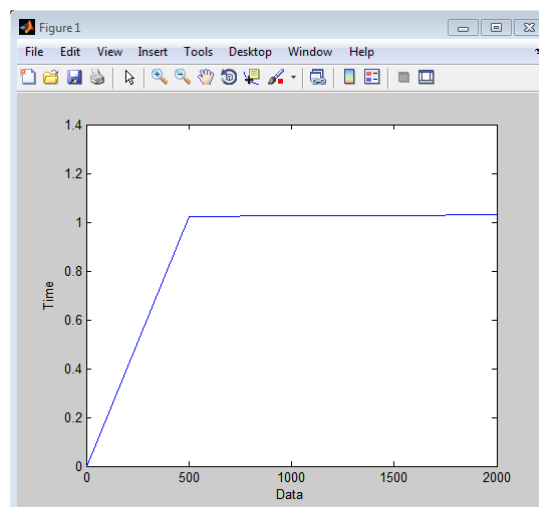
$$\begin{aligned} V(G) &= \text{Number of simple decision} + 1 \\ &= 2 + 1 \\ &= 3 \end{aligned}$$

Or

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 8 - 7 + 2 \\ &= 3 \end{aligned}$$

We have tested all possible test cases for our application. We have completed the development of this application and performed the testing to ensure that our software performs properly and efficiently.

## Output Generation time:



**Figure 4.12:** Time vs data graph

We have tested the output generation time for different amount of data like 0,500,1000, 1500 and 2000 for which output time was 1,1.01,1.02,1.033,1.044,1.05 second. Here we can find that the output generation time haven't changed very much with the increase of data.

# **Chapter 5**

## **Conclusion**

### **6.1 Overview**

Our computerized accounting system will be a beneficial tool for business and large organizations. The total system is dynamic and more accurate than the other existing softwares. Our system has more user friendly interface and it is very easy to use and operate. We have tried to make a system that is faster than manual accounting system. It will maintain up-to-date accounting records and reduce the cost and increase the efficiency. It can generate the report of ledger, trial balance, income statement and balance sheet within a second from the journal entry. Our proposed system can save a great deal of time when posting debits and credits. We have also tried to make the system secure. Our system is easy to use. An user can manage and organize the data and generate the desired output very easily using our system.

### **6.2 Direction for future research:**

The study recommends the following for future empirical studies:

1. Future research should investigate the determination or recommendation of debit and credit automatically of a transaction and check errors using artificial intelligence.
2. Generate more data processing output and show relative graphs, static analysis and different kinds of charts.
3. Enable this software to work on different devices like smart phones and tabs and different operating system.

## Bibliography

- [1] Kenji Ohmori, “*An Internet Accounting System*”, in Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA’03).
- [2] Job Mashapa and Darelle van Greunen, “*User Experience Evaluation Metrics for Usable Accounting Tools*”, in Electronic Journal On Information Systems In Developing Countries 10(4), 1–13 June 2012.
- [3] Elena Urquía Grande, Raquel Pérez Estébanez and Clara Muñoz Colomina “*The impact of Accounting Information Systems (AIS) on performance measures: empirical evidence in Spanish SMEs*”, The International Journal of Digital Accounting Research Vol. 11, 2011 pp. 25 - 43 ISSN: 1577-8517.
- [4] Mathias Ogundeji and, Tijani, Oladipupo Muhrtala, “*Computerized Accounting Information Systems and Perceived Security Threats in Developing Economies: The Nigerian Case*”, Universal Journal of Accounting and Finance 1(1): 9-18, 2013.
- [5] Asaba Rebecca Kisakye “*Computerised Accounting and Processing of Financial Statements in Manufacturing Firms.*” A research report submitted to the college of Education and external studies in partial Fulfillment of the requirement for the award of Bachelors of commerce degree of Makerere University

# Appendix

## AccountEntry.cs

```
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AccountingSoftware
{
    public partial class AccountEntry : Form
    {
        public AccountEntry()
        {
            InitializeComponent();

        }

        Final_AccountEntities db = new Final_AccountEntities();

        private void AccountEntry_Load(object sender, EventArgs e)
        {
            dgvAcEntryList.AutoGenerateColumns = false;

            ScreenRefresh();
        }
    }
}
```

```

cboAcGroupList.DataSource = db.AcGroups.OrderBy(x => x.AcGrpOrder).ToList();

cboAcGroupList.ValueMember = "AcGrpID";

cboAcGroupList.DisplayMember = "AcGrpName";

}

private void ScreenRefresh()
{
    this.txtAcEntryName.Text = "";

    //numOrder.Value = numOrder.Value + 1;

    dgvAcEntryList.DataSource = db.vew_AcEntryList.OrderBy(x => new { x.AcID, x.AcGrpID
}).ToList();

    btnSave.Text = "Save";

    btnEdit.Text = "Edit";

    dgvAcEntryList.Enabled = true;
}

private void btnCancel_Click(object sender, EventArgs e)
{
    ScreenRefresh();
}

private void btnSave_Click(object sender, EventArgs e)
{
    if (txtAcEntryName.Text == "")
    {
        MessageBox.Show("Invalid Name");
    }
}

```

```

        return;
    }

    if (btnSave.Text == "Save")
    {
        if (MessageBox.Show("Are you sure! Want to save?", "Confirmation",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)
        {
            return;
        }

        Int32 myAcEntryId = Convert.ToInt32(db.Organizations.Select(x => x.AcID).First() + 1);

        try
        {
            AcList myEntry = new AcList();

            myEntry.AcID = myAcEntryId.ToString().PadLeft(4, '0');
            myEntry.AcName = txtAcEntryName.Text;
            myEntry.AcAddress = txtAcEntryAddress.Text;
            myEntry.Phone = txtAcEntryPhone.Text;

            // myEntry.AcGrpOrder = Convert.ToInt16(numOrder.Value);
            myEntry.AcGrpID = cboAcGroupList.SelectedValue.ToString();
            db.AcLists.Add(myEntry);

            Organization myOrg = db.Organizations.First(x => x.OrgID == "01");
            myOrg.AcID = myAcEntryId;

            db.SaveChanges();

```



```

    }

    catch (Exception ex)
    {
        MessageBox.Show("Error:" + ex.InnerException.Message);
    }

    ScreenRefresh();

}

else
{
    if (btnSave.Text == "Update")
    {
        if (MessageBox.Show("Are you sure! Want to update?", "Confirmation",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)
        {
            return;
        }

        String myAcId =
        dgvAcEntryList.Rows[this.dgvAcEntryList.CurrentRow.Index].Cells["AcID"].Value.ToString();

        AcList myAcList = db.AcLists.First(x => x.AcID == myAcId);

        myAcList.AcName = txtAcEntryName.Text;

        myAcList.AcAddress = txtAcEntryAddress.Text;

        myAcList.Phone = txtAcEntryPhone.Text;

        myAcList.AcGrpID = cboAcGroupList.SelectedValue.ToString();
    }
}

```

```

        db.SaveChanges();

        ScreenRefresh();

    }

}

}

private void btnEdit_Click(object sender, EventArgs e)
{
    if (dgvAcEntryList.RowCount < 1)
    {
        MessageBox.Show("Nothing to change");
        return;
    }

    if (btnEdit.Text == "Edit")
    {
        txtAcEntryName.Text =
dgvAcEntryList.Rows[this.dgvAcEntryList.CurrentRow.Index].Cells["AcName"].Value.ToString();

        txtAcEntryAddress.Text =
dgvAcEntryList.Rows[this.dgvAcEntryList.CurrentRow.Index].Cells["AcAddress"].Value.ToString();

        txtAcEntryPhone.Text =
dgvAcEntryList.Rows[this.dgvAcEntryList.CurrentRow.Index].Cells["Phone"].Value.ToString();
    }
}

```

```

        // numOrder.Value =
        Convert.ToDecimal(dgvAcGroupList.Rows[this.dgvAcGroupList.CurrentRow.Index].Cells["AcGrpOrder
        "].Value.ToString());

        cboAcGroupList.SelectedValue =
        dgvAcEntryList.Rows[this.dgvAcEntryList.CurrentRow.Index].Cells["AcGrpID"].Value.ToString();

        dgvAcEntryList.Enabled = false;

        btnSave.Text = "Update";

        btnEdit.Text = "Delete";

    }

    else

    {

        if (btnEdit.Text == "Delete")

        {

            if (MessageBox.Show("Are you sure! Want to deleted?", "Confirmation",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)

            {

                return;

            }

            String myListId =
            dgvAcEntryList.Rows[this.dgvAcEntryList.CurrentRow.Index].Cells["AcID"].Value.ToString();

            AcList myAcList = db.AcLists.First(x => x.AcID == myListId);

            db.AcLists.Remove(myAcList);

            db.SaveChanges();

            ScreenRefresh();

        }

    }

}

}

```

```
}
```

### **AcTypeEntry.cs**

```
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AccountingSoftware

{

    public partial class AcTypeEntry : Form

    {

        public AcTypeEntry()

        {

            InitializeComponent();

        }

        Final_AccountEntities db = new Final_AccountEntities();

        private void AcTypeEntry_Load(object sender, EventArgs e)

        {

            dgvAcTypeList.AutoGenerateColumns = false;
```

```

ScreenRefresh();

cboComponentList.DataSource = db.AcCompnents.OrderBy(x=>x.CompOrder).ToList();
cboComponentList.ValueMember = "Comp_ID";
cboComponentList.DisplayMember = "ComponentName";
}

private void ScreenRefresh()
{
    this.txtAcTypeName.Text = "";
    numOrder.Value = numOrder.Value + 1;
    dgvAcTypeList.DataSource = db.vew_AcTypeList.OrderBy(x=>new {x.Comp_ID,
x.AcTypeOrder}).ToList();

    btnSave.Text = "Save";
    btnEdit.Text = "Edit";
    dgvAcTypeList.Enabled = true;
}

private void btnCancel_Click(object sender, EventArgs e)
{
    ScreenRefresh();
}

private void btnSave_Click(object sender, EventArgs e)
{
    if (txtAcTypeName.Text=="")

```

```

{
    MessageBox.Show("Invalid Name");

    return;
}

if (btnSave.Text == "Save")
{
    if (MessageBox.Show("Are you sure! Want to save?", "Confirmation",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)
    {
        return;
    }

    Int16 myTypeId = Convert.ToInt16(db.Organizations.Select(x => x.AcTypeID).First() + 1);

    try
    {
        AcType myType = new AcType();

        myType.AcTypeID = myTypeId.ToString().PadLeft(3, '0');
        myType.AcTypeName = txtAcTypeName.Text;
        myType.AcTypeOrder = Convert.ToInt16(numOrder.Value);
        myType.Comp_ID = cboComponentList.SelectedValue.ToString();
        db.AcTypes.Add(myType);

        Organization myOrg = db.Organizations.First(x => x.OrgID == "01");
        myOrg.AcTypeID = myTypeId;
    }
}

```

```

        db.SaveChanges();
    }

    catch (Exception ex)
    {
        MessageBox.Show("Error:" + ex.InnerException.Message);
    }

    ScreenRefresh();

}

else
{
    if (btnSave.Text == "Update")
    {
        if (MessageBox.Show("Are you sure! Want to update?", "Confirmation",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)
        {
            return;
        }

        String myTypeId =
        dgvAcTypeList.Rows[this.dgvAcTypeList.CurrentRow.Index].Cells["AcTypeID"].Value.ToString();

        AcType myAcType = db.AcTypes.First(x => x.AcTypeID == myTypeId);

        myAcType.Comp_ID = cboComponentList.SelectedValue.ToString();
    }
}

```

```

        myAcType.AcTypeName =txtAcTypeName.Text;

        myAcType.ActypeOrder = Convert.ToInt16(numOrder.Value);

        db.SaveChanges();

        ScreenRefresh();

    }

}

}

private void btnEdit_Click(object sender, EventArgs e)
{
    if (dgvAcTypeList.RowCount < 1)
    {
        MessageBox.Show("Nothing to change");

        return;

    }

    if (btnEdit.Text == "Edit")
    {

        txtAcTypeName.Text =
dgvAcTypeList.Rows[this.dgvAcTypeList.CurrentRow.Index].Cells["AcTypeName"].Value.ToString();

        numOrder.Value =
Convert.ToDecimal(dgvAcTypeList.Rows[this.dgvAcTypeList.CurrentRow.Index].Cells["ActypeOrder"]
.Value.ToString());
    }
}

```



```

        cboComponentList.SelectedValue =
dgvAcTypeList.Rows[this.dgvAcTypeList.CurrentRow.Index].Cells["Comp_ID"].Value.ToString();

        dgvAcTypeList.Enabled = false;

        btnSave.Text = "Update";

        btnEdit.Text = "Delete";
    }

    else

    {

        if (btnEdit.Text == "Delete")

        {

            if (MessageBox.Show("Are you sure! Want to deleted?", "Confirmation",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)

            {

                return;

            }

            String myAcTypeID =
dgvAcTypeList.Rows[this.dgvAcTypeList.CurrentRow.Index].Cells["AcTypeID"].Value.ToString();

            AcType myAcType = db.AcTypes.First(x => x.AcTypeID == myAcTypeID);

            db.AcTypes.Remove(myAcType);

            db.SaveChanges();

            ScreenRefresh();

        }

    }

}

}

}

```

### **AcGroupEntry.cs**

```
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AccountingSoftware

{

    public partial class AcGroupEntry : Form

    {

        public AcGroupEntry()

        {

            InitializeComponent();

        }

        Final_AccountEntities db = new Final_AccountEntities();

        private void AcGroupEntry_Load(object sender, EventArgs e)

        {

            dgvAcGroupList.AutoGenerateColumns = false;

            ScreenRefresh();

        }

    }

}
```

```

        cboAcTypeList.DataSource = db.AcTypes.OrderBy(x => x.ActypeOrder).ToList();

        cboAcTypeList.ValueMember = "AcTypeID";

        cboAcTypeList.DisplayMember = "AcTypeName";
    }

    private void ScreenRefresh()
    {
        this.txtAcGroupName.Text = "";

        numOrder.Value = numOrder.Value + 1;

        dgvAcGroupList.DataSource = db.vew_AcGroupList.OrderBy(x => new { x.AcTypeID,
x.AcGrpOrder }).ToList();

        btnSave.Text = "Save";

        btnEdit.Text = "Edit";

        dgvAcGroupList.Enabled = true;
    }

    private void btnCancel_Click(object sender, EventArgs e)
    {
        ScreenRefresh();
    }

    private void btnSave_Click(object sender, EventArgs e)
    {
        if (txtAcGroupName.Text=="")

        {
            MessageBox.Show("Invalid Name");
        }
    }

```

```

        return;
    }

    if (btnSave.Text == "Save")
    {
        if (MessageBox.Show("Are you sure! Want to save?", "Confirmation",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)
        {
            return;
        }

        Int16 myGroupId = Convert.ToInt16(db.Organizations.Select(x => x.AcGrpID).First() + 1);

        try
        {
            AcGroup myGroup = new AcGroup();

            myGroup.AcGrpID = myGroupId.ToString().PadLeft(3, '0');

            myGroup.AcGrpName = txtAcGroupName.Text;

            myGroup.AcGrpOrder = Convert.ToInt16(numOrder.Value);

            myGroup.AcTypeID = cboAcTypeList.SelectedValue.ToString();

            db.AcGroups.Add(myGroup);

            Organization myOrg = db.Organizations.First(x => x.OrgID == "01");

            myOrg.AcGrpID = myGroupId;

            db.SaveChanges();
        }

        catch (Exception ex)

```

```

{
    MessageBox.Show("Error:" + ex.InnerException.Message);
}

ScreenRefresh();

}
else
{
    if (btnSave.Text == "Update")
    {
        if (MessageBox.Show("Are you sure! Want to update?", "Confirmation",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)
        {
            return;
        }

        String myGroupId =
        dgvAcGroupList.Rows[this.dgvAcGroupList.CurrentRow.Index].Cells["AcGrpID"].Value.ToString();

        AcGroup myAcGroup = db.AcGroups.First(x => x.AcGrpID == myGroupId);

        myAcGroup.AcTypeID = cboAcTypeList.SelectedValue.ToString();

        myAcGroup.AcGrpName = txtAcGroupName.Text;

        myAcGroup.AcGrpOrder = Convert.ToInt16(numOrder.Value);

        db.SaveChanges();

        ScreenRefresh();
    }
}

```

```

    }

    }

}

private void btnEdit_Click(object sender, EventArgs e)

{
    if (dgvAcGroupList.RowCount < 1)
    {
        MessageBox.Show("Nothing to change");
        return;
    }

    if (btnEdit.Text == "Edit")
    {

        txtAcGroupName.Text =
dgvAcGroupList.Rows[this.dgvAcGroupList.CurrentRow.Index].Cells["AcGrpName"].Value.ToString();

        numOrder.Value =
Convert.ToDecimal(dgvAcGroupList.Rows[this.dgvAcGroupList.CurrentRow.Index].Cells["AcGrpOrder
"].Value.ToString());

        cboAcTypeList.SelectedValue =
dgvAcGroupList.Rows[this.dgvAcGroupList.CurrentRow.Index].Cells["AcTypeID"].Value.ToString();

        dgvAcGroupList.Enabled = false;

        btnSave.Text = "Update";

        btnEdit.Text = "Delete";
    }
}

```

```

    }

    else

    {

        if (btnEdit.Text == "Delete")

        {

            if (MessageBox.Show("Are you sure! Want to deleted?", "Confirmation",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)

            {

                return;

            }

            String myGroupId =
dgvAcGroupList.Rows[this.dgvAcGroupList.CurrentRow.Index].Cells["AcGrpID"].Value.ToString();

            AcGroup myAcGroup = db.AcGroups.First(x => x.AcGrpID == myGroupId);

            db.AcGroups.Remove(myAcGroup);

            db.SaveChanges();

            ScreenRefresh();

        }

    }

}

}

}

```