

**Kathmandu University**  
**Department of Computer Science and Engineering**  
**Dhulikhel, Kavre**



A Project Report on 'e-Ledger'

[Code No.: 102]

(For partial fulfillment of 1st Year / 2nd Semester in Computer Engineering)

**Submitted by:**

Chandan Kumar Mahato (31)

Swornim Nakarmi (35)

Ashutosh B. Rajan (46)

Arya Shakya (50)

Sarin Sthapit (55)

**Submitted to:**

Mr. Satyendra Nath Lohani

Ms. Deni Shahi

Department of Computer Science and Engineering

**Submission Date:** September 28, 2020

## **Acknowledgement**

We would like to express a feeling of gratitude towards our project coordinator Ms. Deni Shahi and our project supervisor Mr. Satyendra Nath Lohani for providing us with this wonderful opportunity of creating a working application on our own. Our special thanks to their continuous efforts, active guidance, supervision and encouragement throughout the project duration.

We are extremely thankful to our COMP 116 lecturer Dr. Rajani Chulyadyo and the entire Department of Computer Science and Engineering who have helped us with this project.

## **Bonafide Certificate**

This project on

‘e-Ledger’

is the bonafide work of

Chandan Kumar Mahato (31), Swornim Nakarmi (35),

Ashutosh B. Rajan (46), Arya Shakya (50) and Sarin Sthapit (55)

who carried out the project under my supervision.

**Project Supervisor**

---

**Mr. Satyendra Nath Lohani**

Lecturer

Department of Computer Science and Engineering

Kathmandu University

## **Abstract**

A digitized version of a traditional ledger, 'e-Ledger' aims to ease the traders by providing a better method to keep track of transaction records. e-Ledger enables anyone involved in trade to efficiently use digital technology in an effective and organized manner, which would be difficult with traditional ledgers. We have used C++ along with Object Oriented Programming concepts, Qt for GUI, used SQLite to work with data and implemented a login system with username and password for the security and privacy of the user. We have developed a program which is easy to use and low on system resources. Therefore, it can be used by everyone. To put it in a nutshell, it has been developed for local traders; a step has been taken towards digitization of Nepal. The idea that inspired us for this project was to find an alternative to regular tedious work of managing records and accounts. The project can further be improved making it an online transaction system and also be developed into a cross-platform application.

**Keywords:** Object Oriented Programming with C++, transaction, record keeping, SQLite, Caesar Algorithm, Qt, CSV, graph

## Table of Contents

List of figures .....	i
List of tables.....	ii
Acronyms/Abbreviations .....	iii
Chapter 1: Introduction .....	1
1.1 Introduction .....	1
1.2 Background .....	1
1.3 Objectives.....	2
1.4 Motivation and significance .....	2
1.5 Basic Characteristics of the project.....	3
Chapter 2: Related Works.....	4
Chapter 3: Design and Implementation .....	6
3.1 System Requirement Specification .....	6
3.1.1 Software Specification.....	6
3.1.2 Hardware Requirement.....	7
3.2 Caesar Algorithm .....	7
3.3 Diagrams .....	9
3.3.1 ER Diagram .....	9
3.3.2 Activity Diagram .....	10
3.3.3 Use Case Diagram .....	11
3.3.4 Flow Charts.....	13
Chapter 4: Discussion on the Achievements .....	18
4.1 Features: .....	19
Chapter 5: Conclusion and Recommendation.....	27
5.1 Limitations .....	27
5.2 Future Enhancements .....	27
References.....	29
Appendix.....	30

## List of figures

Figure 3. 1 Encryption with Caesar Algorithm.....	8
Figure 3.3. 1 Entity Relationship Diagram .....	9
Figure 3.3. 2 Activity Diagram .....	10
Figure 3.3. 3 Use Case Diagram .....	12
Figure 3.3.4. 1 Login Flowchart .....	13
Figure 3.3.4. 2 Create Account Flowchart.....	14
Figure 3.3.4. 3 Forgot password flow chart.....	15
Figure 3.3.4. 4 Main window flow chart .....	16
Figure 3.3.4. 5 View/Export flow chart .....	17
Figure 4.1. 1 Registration window.....	20
Figure 4.1. 2 Login window.....	20
Figure 4.1. 3 Window to reset password.....	21
Figure 4.1. 4 Main window of the program where a user can perform different tasks.....	22
Figure 4.1. 5 Records of buyer in descending order of date. ....	23
Figure 4.1. 6 Records of buyers whose names start from P to T .....	24
Figure 4.1. 7 Graph of Transaction amounts vs day .....	25
Figure 4.1. 8 Graph of no. of transaction vs day.....	25
Figure 4.1. 9 A filtered and exported file opened in Microsoft Excel. ....	26
Figure i Gantt chart .....	30

## **List of tables**

Table 1: Table for Project Timeline.....	30
--	----

## **Acronyms/Abbreviations**

FMCG: Fast-Moving Consumer Goods

GUI: Graphical User Interface

ERMS: Enterprise Record Management System

CSV: Comma Separated Values

RDBMS: Relational Database Management System

GPL: General Public License

LGPL: Lesser General Public License

CPU: Central Processing Unit

RAM: Random Access Memory



# **Chapter 1: Introduction**

## **1.1 Introduction**

Ledger is an account book or computer application for recording economic transactions measured in terms of monetary units of account. It includes debits, credits, transactions, net-worth and all the minor details about the transactions like transaction object, person involved in particular transaction, and basic personal details of second person involved in particular transaction as an outcome. e-Ledger is an electronic transaction management system i.e., computer application that focuses on easy and secure alternatives to traditional record keeping. It is a digital version of traditional record books with enhanced features so that the tedious record keeping work gets more convenient and manageable.

## **1.2 Background**

The application, e-Ledger is specially designed to support users for record keeping which is electronically accessible, reliable, easily usable, portable, accurate and secure. It provides quick information about the records and makes the calculation easy. The project also records the information like name, contact number, address, etc. of the second person who is involved in the transaction. Digital record keeping provides a secure connection between people who are involved in particular transactions. It is sure that digital record keeping systems are very useful to save time and resources. Many organizations and people now prefer digital record keeping and transaction management systems over traditional one. Transaction management and record keeping software similar to e-Ledger are leading in managing and keeping records easily and securely. One of the modern technologies that have revolutionized such systems and software, in general, is cloud computing. It has a profound impact on document management systems.

Such web-based document management system also allows for scalability, making it a solution for business both for small and large enterprises. Since it offers ease of access and reduces costs, cloud computing drives the best document management systems. Another similar example is Enterprise Record Management System (ERMS). These programs have

advanced capabilities and can store, organize and process large amounts of data. With an ERMS, it is easier to manage physical and digital records, track the status and location of each file and transfer data from legacy systems to the latest software. Systems with vulnerabilities and bugs are always bound to security issues. So, adequate testing and debugging is necessary to create a secure and reliable program to meet all the requirements of a ledger. It is an economical and the easiest way to keep records.

### **1.3 Objectives**

The basic purpose of our project ‘e-Ledger’ is to provide easy, economical, and secure digital record management systems.

#### **List of objectives of our project**

1. Easy record keeping and information retrieval.
2. Simplify accounting.
3. Cost-saving from materials and space used by traditional record keeping.

### **1.4 Motivation and significance**

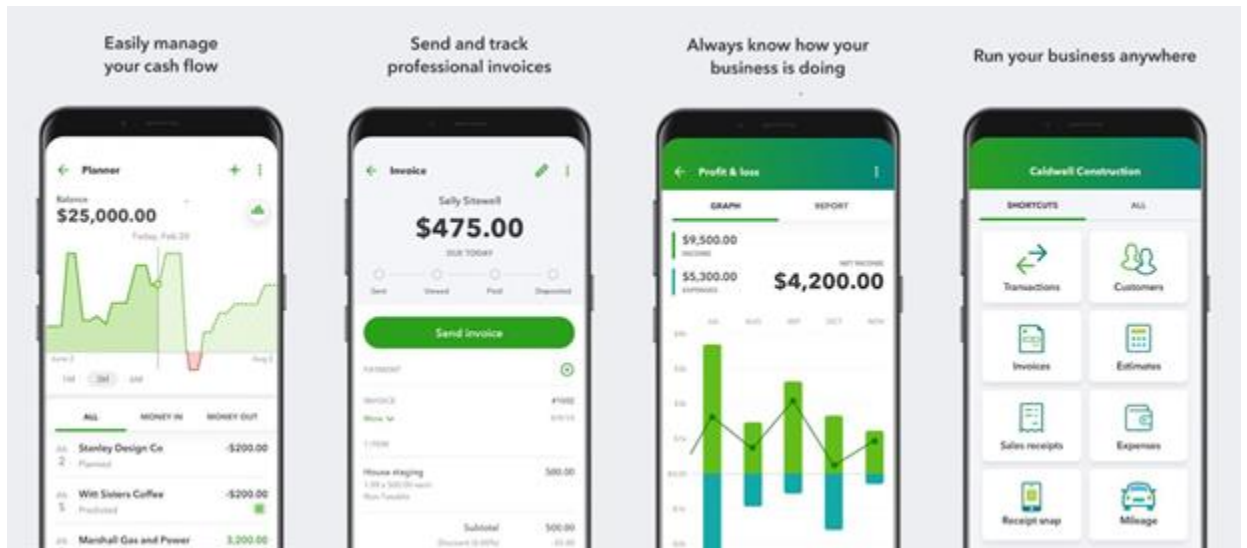
We were inspired to work on this project due to monotonous, tedious and unmanageable statements of incomes, expenditures and other records of various trades, for example as in a FMCG shop. ‘e-Ledger’ addresses such drawbacks of traditional record keeping. The user can view transactions based on date, name, address, phone number, amount, etc. stored in their computer. Traditional ledger is convenient for small transactions. However, transactions with hundreds of customers make it quite difficult to handle and becomes tedious. Use of e-Ledger, enables us to store name, phone number, address, date etc. and most importantly transactions made with them. One can save, update, delete and export the transaction records without any hassles as we face in case of traditional ledger. Data can be viewed easily even if there are thousands of unique data using sorting and filter features. No need to find someone manually one by one in a particular paper in the ledger of hundreds of pages.

## **1.5 Basic Characteristics of the project**

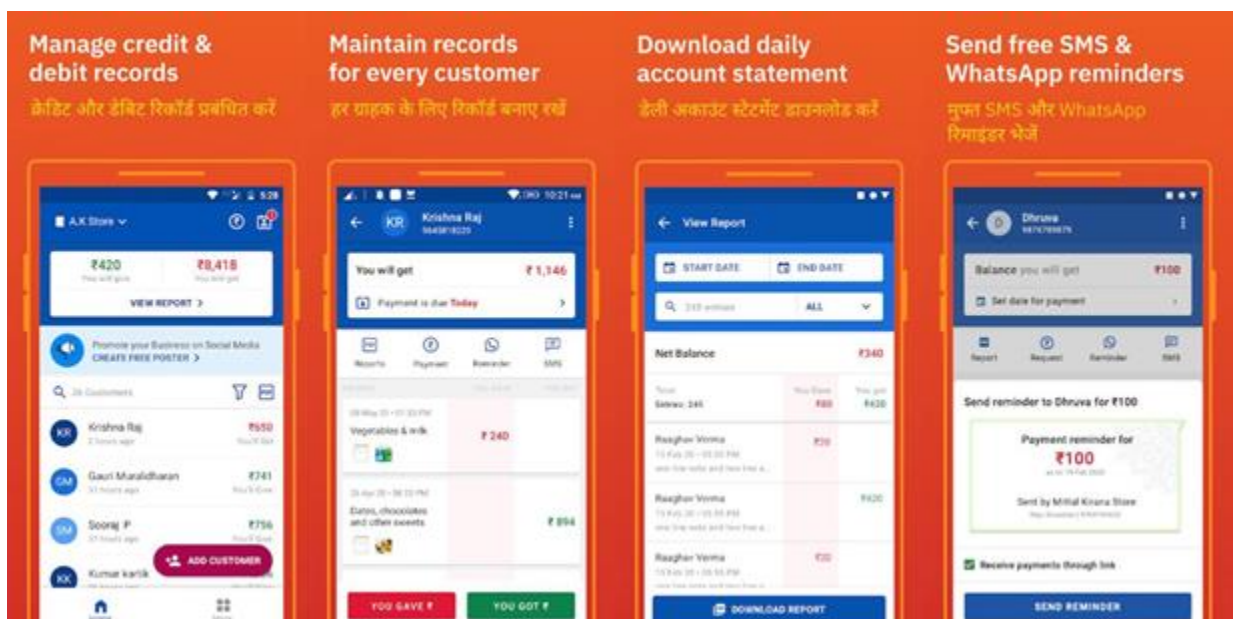
1. Multiple users can use the application.
2. Easy and quick audit, modification and retrieval of data.
3. Transaction records can be saved for buyers and sellers separately.
4. Records can be viewed by sorting and filtering with provided options.
5. One can have a quick look of his/her transaction with graphs.
6. Data can be easily exported as a CSV file with or without sorting or filtering.

## Chapter 2: Related Works

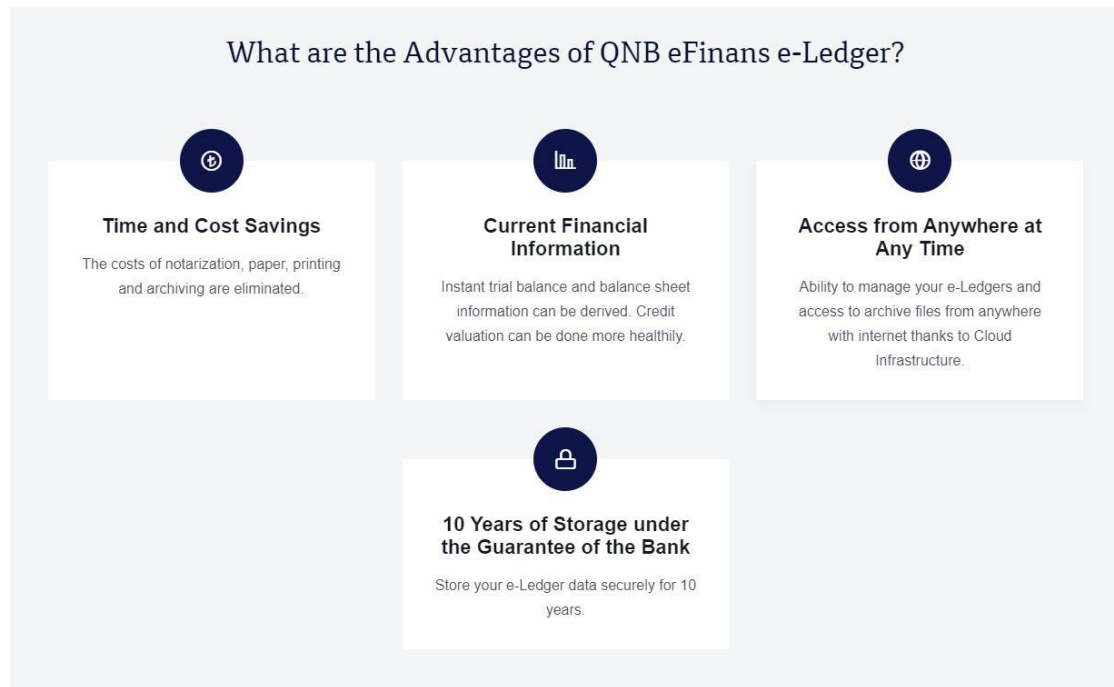
QuickBooks (intuit, 2018) can track income & expenses, prepare custom invoices and receipts and even connect to the bank.



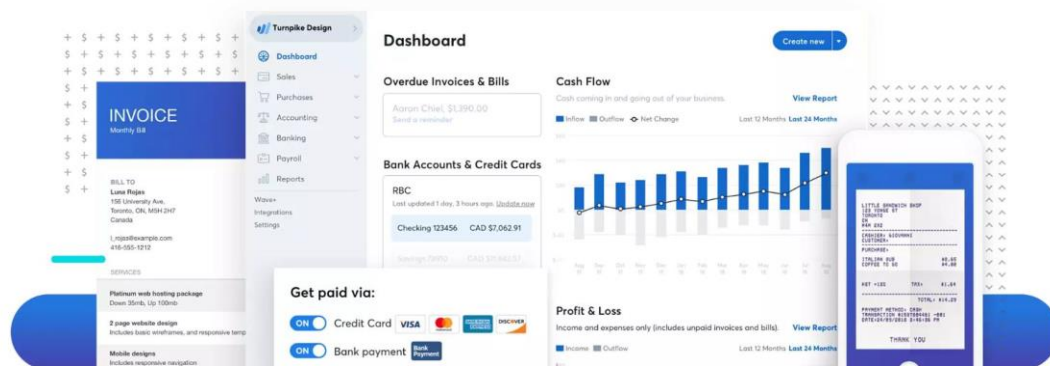
KhataBook (KhataBook, 2020) claims that it is a free and secure digital ledger cash book that provides reminders in the form of SMS on every transaction, allowing to view all details of credit-debit for any number of customers across multiple businesses on our phone.



e-Ledger (QNB eFinans, n.d.) has features like managing and accessing files like trial balance and balance sheet information from anywhere with the internet.



Waveapps (Wave Financial Inc., n.d.) is a free accounting software for small businesses and aims to provide organized and stress-free accounting.



## **Chapter 3: Design and Implementation**

The project has been initiated to suffice the needs of shopkeepers and traders as well as to function capably as a ledger. Thus, the application can be used as a multi-purposed application.

First of all, the login page is loaded. An existing account holder can enter their username and password to log into the main window whereas a new user needs to create a new account. To create an account, valid username, password and answers to two security questions needs to be provided. As soon as the user logs in or creates an account, the main window of the application opens with which the details of any transactions can be saved, updated, deleted, viewed and exported. The user can also view graphs to check the entire transaction details of a month. The transaction data can be exported as a CSV file with or without sorting or filtering the records. System diagrams have been included in the section 3.3.

### **3.1 System Requirement Specification**

#### **3.1.1 Software Specification**

##### **3.1.1.1 Front End Tools**

##### **Qt**

According to (*About Qt - Qt Wiki*, n.d.), Qt (pronounced ‘cute’) is a free and open-source widget toolkit for creating graphical user interfaces as well as cross platform applications that run on various software and hardware platforms. Qt is currently being developed by The Qt Company. Qt is available under both commercial licenses and open-source GPL 2.0, GPL 3.0, and LGPL 3.0 licenses.

e-Ledger uses Qt to have a clean and minimalist look which provides a very user-friendly interaction. Libraries like QSql and QCustomPlot have been used for database connection and plotting graphs respectively.

### **3.1.1.2 Back End Tools:**

#### **SQLite**

According to (*About SQLite*, n.d.), SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private. SQLite is the most widely deployed database in the world with more applications than we can count, including several high-profile projects. SQLite is a relational database management system contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. It is small, fast and reliable.

e-Ledger uses SQLite as a database which is supported by Qt and various options for manipulating databases are supported out of the box. SQLite is a popular choice as embedded database software for local/client storage in application software which is exactly the current use case of e-Ledger.

### **3.1.2 Hardware Requirement**

The hardware requirements for e-Ledger is bare minimum and most of the modern working computer will suffice it. The project was tested on 1080p screen and if used on low or high screen resolution, appropriate scaling should be applied. The setup file requires 22 MB and occupies 80 MB of storage after installation. The CPU and RAM usages are variable.

## **3.2 Caesar Algorithm**

Caesar Algorithm is an encryption method in which every single character is substituted by another character some fixed position away from the prior one down the set of characters. The position is determined by an integer value, which is referred to as a key or shift. Once the key is known, encrypted text can be easily decrypted. Mathematically, the encryption method can be explained as:

$$E_i = (D_i + k) \bmod L$$

$$D_i = (E_i - k) \bmod L$$

Where,  $E_i$  =  $i$ -th character of the encrypted text

$D_i$  =  $i$ -th character of the plain or decrypted text

$k$  = key

$L$  = length of a set of characters

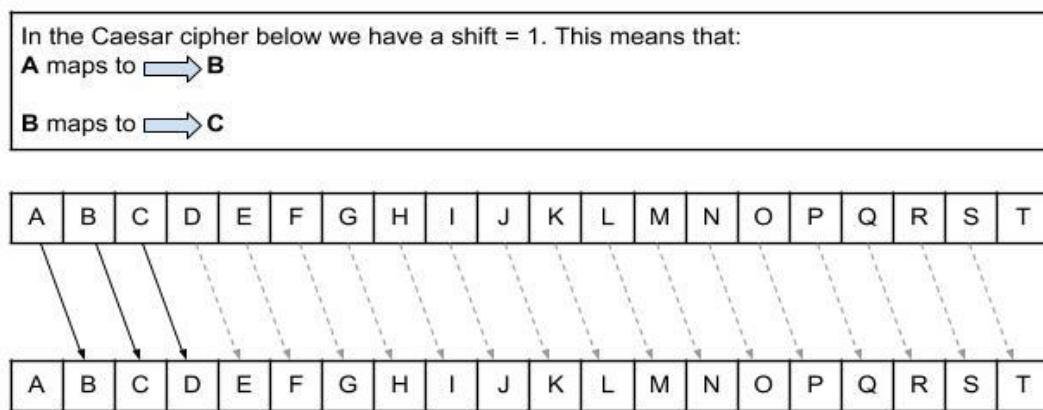


Figure 3. 1 Encryption with Caesar Algorithm



### 3.3 Diagrams

#### 3.3.1 ER Diagram



Figure 3.3. 1 Entity Relationship Diagram

The Entity Relationship (ER) diagram describes the entities, its attributes and the relationships between entities.

Figure 3.3.1 shows how different entities interact with each other in our application. The user entity has one and only one credential and vice versa. The credential entity has attributes like username, password and two security answers. The user entity can receive from one or many buyer(s). The buyer entity has Transaction ID as the primary key and other attributes like

Date, Name, Address, Phone No., Total amount, Received amount, Receivable and Remarks.  
 The user entity can pay one or many seller(s). The seller entity has Transaction ID as the primary key and other attributes like Date, Name, Address, Phone No., Total Amount, Paid amount, Payable and Remarks.

### 3.3.2 Activity Diagram

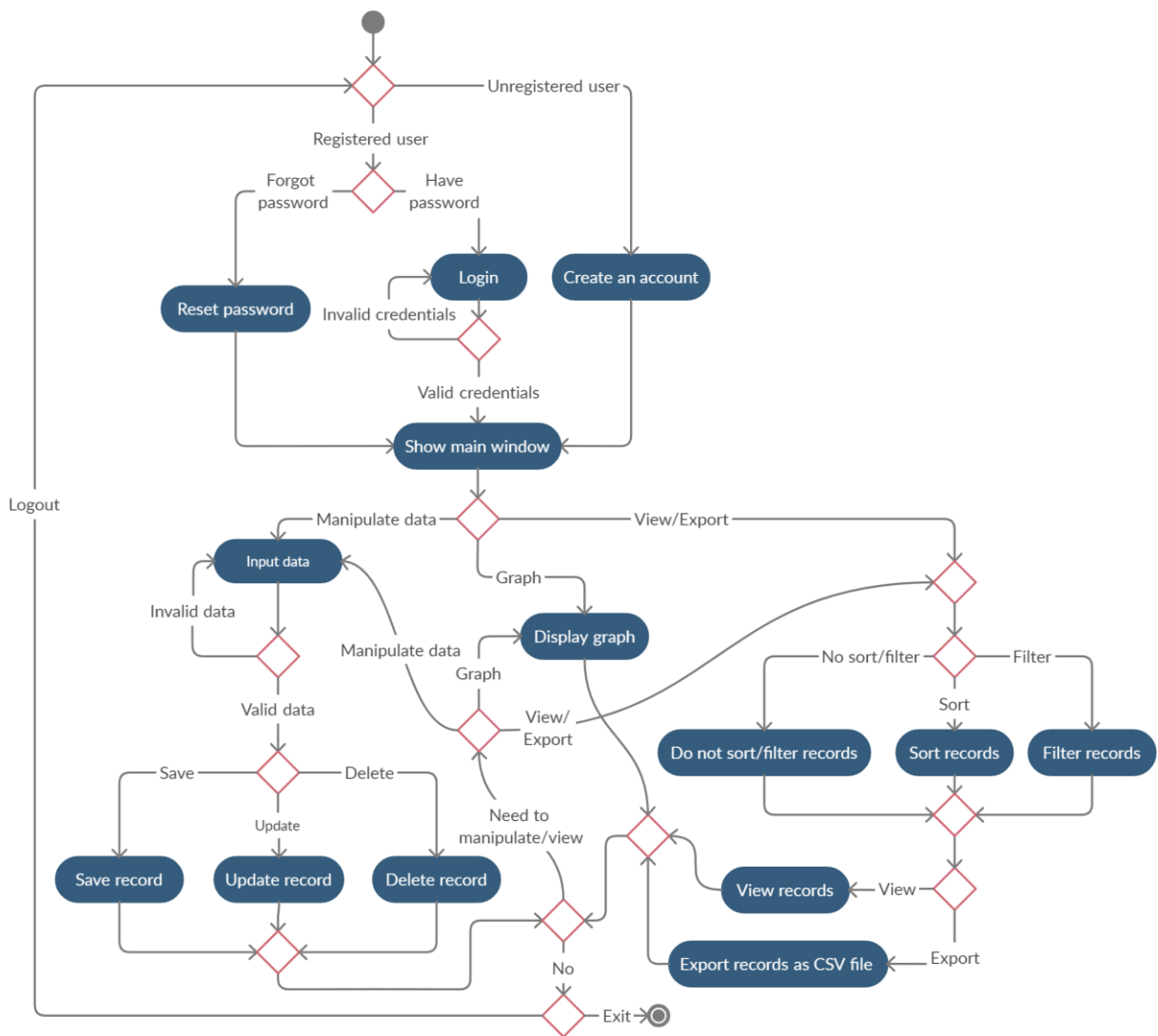


Figure 3.3. 2 Activity Diagram

Activity Diagram is the graphical representation of the workflow from one activity to another in the program.

At the start of the program, the workflow is divided for a registered user and an unregistered user. If the registered user has the password, then he/she can login to the account. If the entered credentials are valid, then the main window is displayed otherwise it returns back to the login page. The user can also reset the password if they cannot remember. In case of the unregistered user, they can create an account with their valid credentials and then the main window is displayed.

The user can either manipulate data, view/export the records or display graphs. They can view/export the records by applying or not applying sorting, filtering. Additionally, graphs can be viewed based on Transaction amounts per day or No. of transaction per day in a month. To manipulate data, the user can save, update or delete the record by providing valid data (mainly Transaction ID). The user can also logout from the application if he/she wants.

### **3.3.3 Use Case Diagram**

A use case diagram can be defined as a representation of users' interaction with the program that represents the relationship between the user and the different use cases in which the users are involved. Its main purpose is to provide a complete functional and technical view to the user.

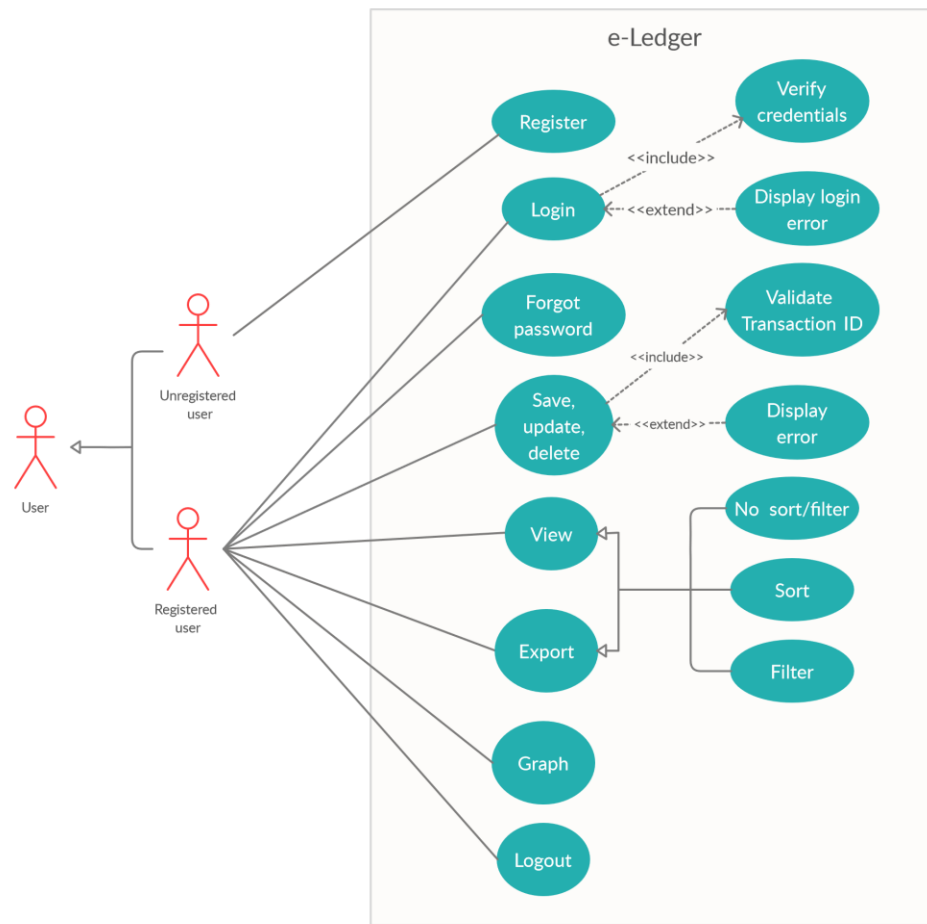


Figure 3.3. 3 Use Case Diagram

There are two distinguished types of users in the application i.e. registered user and unregistered user. The new user has to create an account.

The registered user has to enter their valid username and password to login. If the user for some reason, forgets his/her password then he/she can reset the password by entering the correct username and security answers and valid new password. The registered user can add, update and delete the currently existing records. The user can choose to sort or filter or not to sort or filter before viewing or exporting the records. Moreover, the user can also view graphs and logout from the application.

### 3.3.4 Flow Charts

Flowchart is a diagrammatic representation of the flow of a program which explains the processes involved in the execution of a program. Following flowcharts explain various conditions and their actions that take place in the application.

#### 3.3.4.1 Login Flow Chart

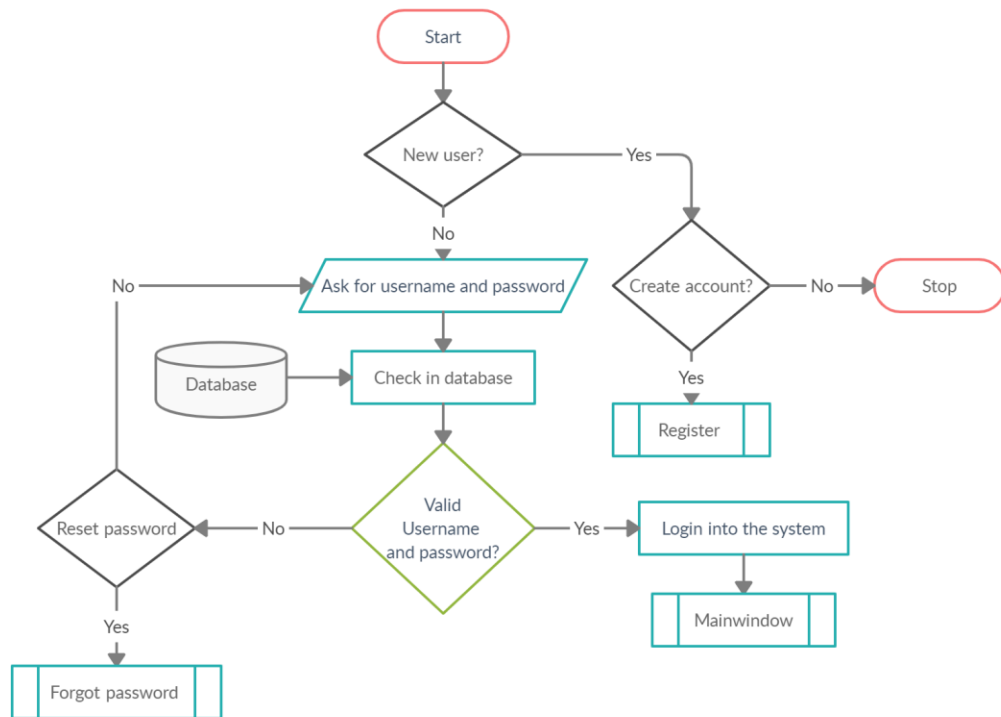


Figure 3.3.4. 1 Login Flowchart

The 'login' process starts with the decision whether the user is new or registered. If the user is new and wants to create an account, he/she can switch to the 'Registration' window. For the registered user, he/she needs to enter a valid username and password to login. The application checks if the given credentials exist in the database. If the entered credentials are valid, it logs the user into the system and shows the main window. If the user wants to reset the password, they can switch to the 'Forgot password' window.

### 3.3.4.2 Register Flow Chart

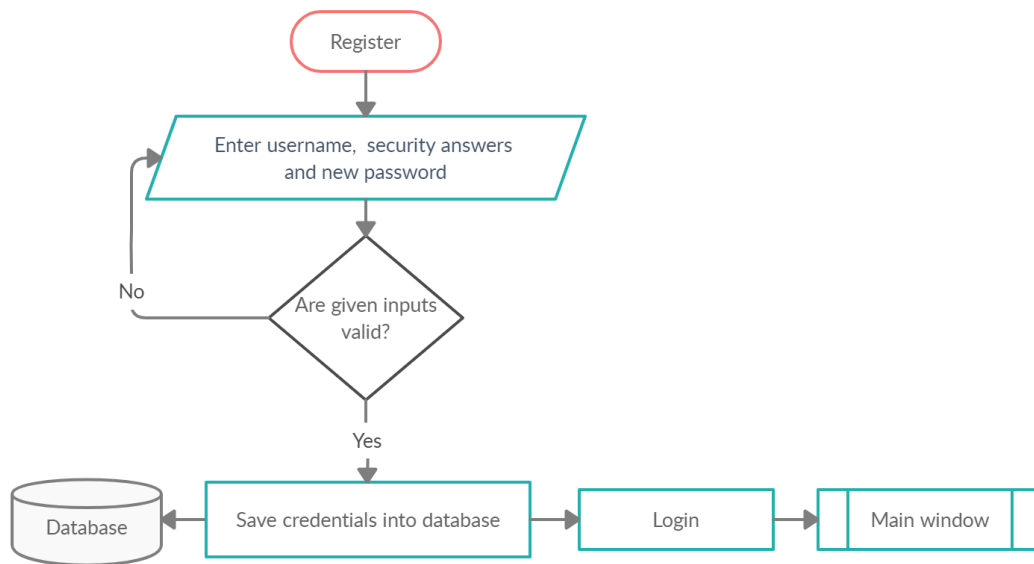


Figure 3.3.4. 2 Create Account Flowchart

When a user chooses to create a new account, the ‘Registration’ window is shown. The user needs to set a unique username and password in a valid format as mentioned in the manual attached to the application. They also need to answer the security questions which are used to recover their account if they forget their password later. If the given inputs are valid then he/she will be logged into the application and the ‘main window’ is shown. Otherwise, the user has to repeat the same process.

### 3.3.4.3 Forgot password flow chart

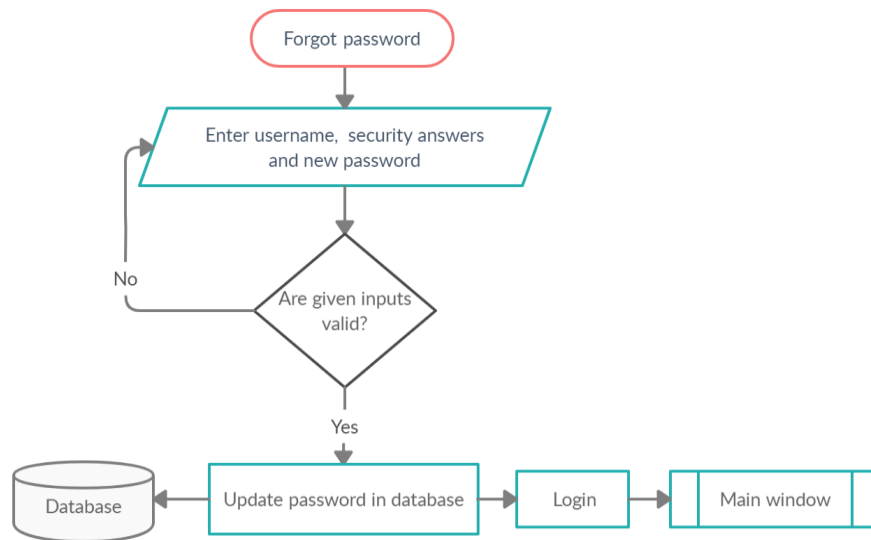


Figure 3.3.4. 3 Forgot password flow chart

If a registered user has forgotten their password, they can input their username, answer security questions and a new password. If the entered username and security answers are valid, it updates the password in the database and logs the user into the application and displays the 'main window'. Otherwise, the user needs to repeat the same process.

### 3.3.4.4 Main window Flow Chart

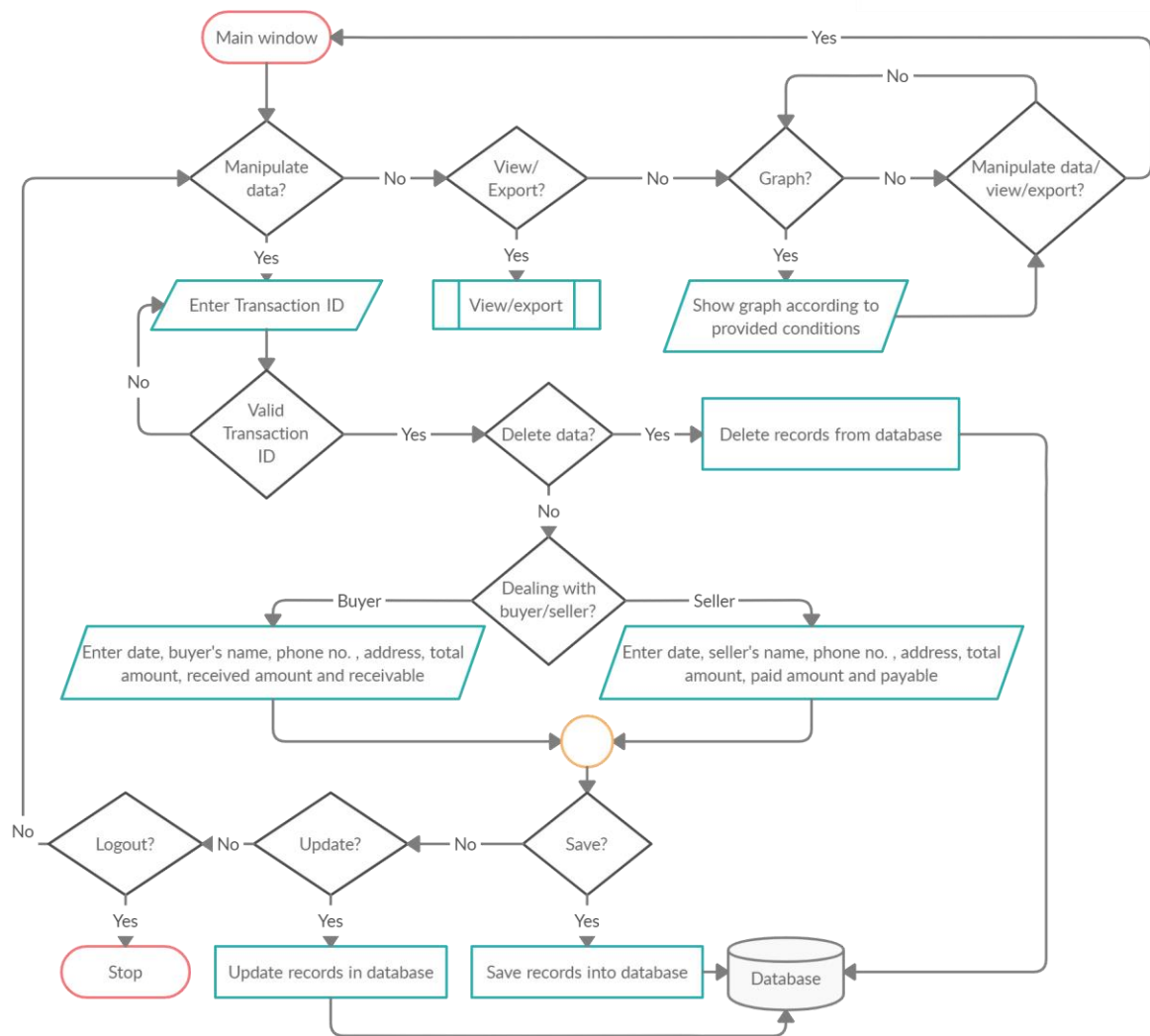


Figure 3.3.4. 4 Main window flow chart

As the user of e-Ledger logs into their account, he/she can easily manipulate, view or export the transaction records and can also view graphs. e-Ledger provides two modes of transaction recording system i.e. with buyer and seller. Users can manipulate data i.e. users can save new records and update or delete existing records. The transaction records can also be viewed and exported. Graphs can be viewed to have a quick look how the transaction went in different months. The user can log out from the application if he/she has got his job done.



### 3.3.4.5 View/Export Flow Chart

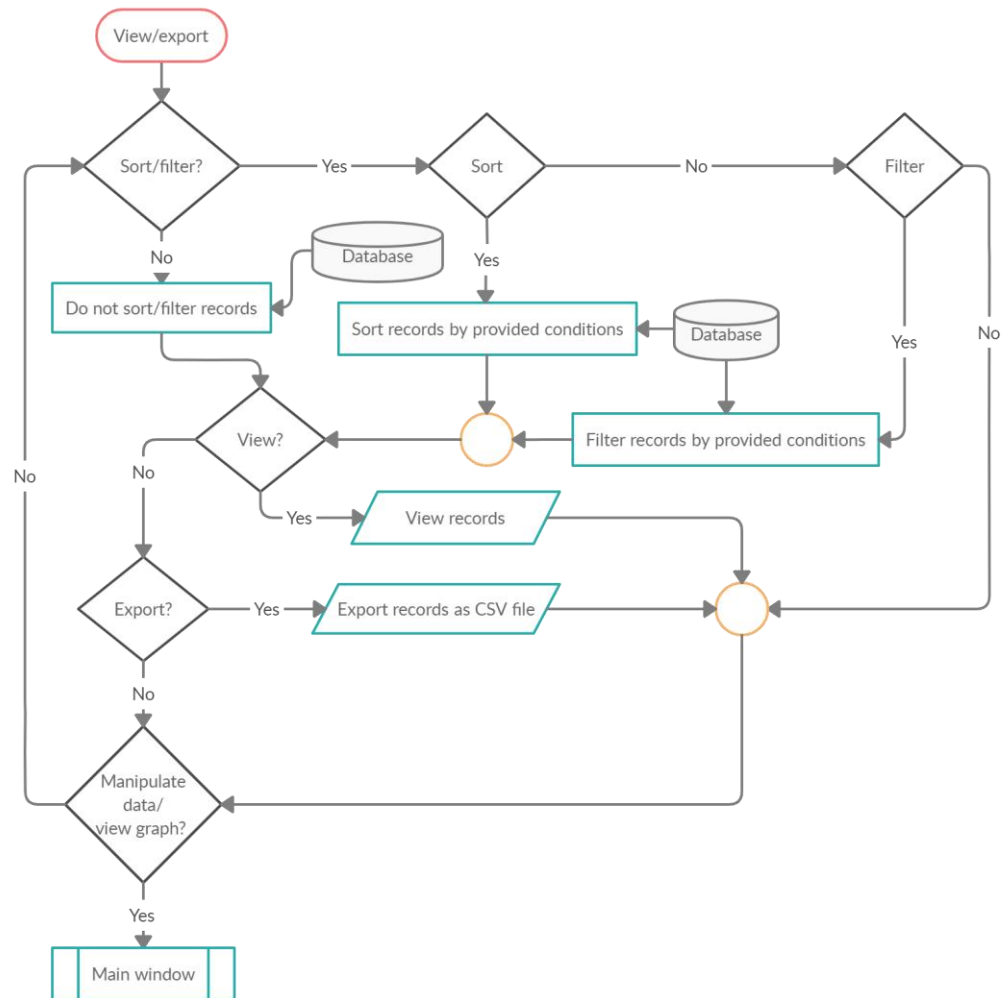


Figure 3.3.4. 5 View/Export flow chart

Users can view or export the records by sorting or filtering based on provided conditions. The records are displayed in a table and exported the user's desktop as a CSV file. They can also choose not to sort or filter the records.

## **Chapter 4: Discussion on the Achievements**

After the completion of this project, we observed that the final project has met our objectives. We are certain that e-Ledger is able to solve the problem that we have with traditional record keeping. It can be used by small-scale businessmen as well as extensive traders. It can help to reduce the practice of keeping records in paper, and ensures the safety of accounts and store them in a more manageable manner so that it would be easy to retrieve whenever required. The project helped us to learn many technologies and concepts, which are n like Object Oriented Programming blended, SQLite and Qt.

The challenges we faced during the development of this project were as follows:-

1. To develop a GUI based application.
2. To integrate a database in the application.
3. To determine the actual users of this application as it could be used for both buyers and sellers in different perspectives.
4. To work for deployment of the application.
5. To test, debug and optimize the codes.

The difference/improvements seen in the actual project that were different or unplanned in the proposed project were as follows:-

1. The application was supposed to be a console application. However, we have developed a GUI application.
2. The proposed project was supposed to implement the merge sort to sort the records. However, we later used the 'sort' clause of SQL to sort the data as it would be easier and effective.

The project mainly had some positive deviations that made the concept of the project even broader than form the initial plans about the project. Some of the aspects that had deviated from the objectives are as follows:

1. The project was supposed to be a sales recording application for sellers but it was further improved to be suitable for both buyer and seller.
2. The main objective of the program was to build a digital ledger. However, the program has been developed as a digital ledger, virtual balance sheet and transaction management system.

Our main motivation for building this program was to take a step towards digitization of Nepal and to reduce the use of paper in business and trading sectors. We also wanted to allow the user to store all the records of monetary transactions of his/her customers and to help the user view and retrieve the detailed information of the stored records anytime as well as to maintain a well-organized record of sales.

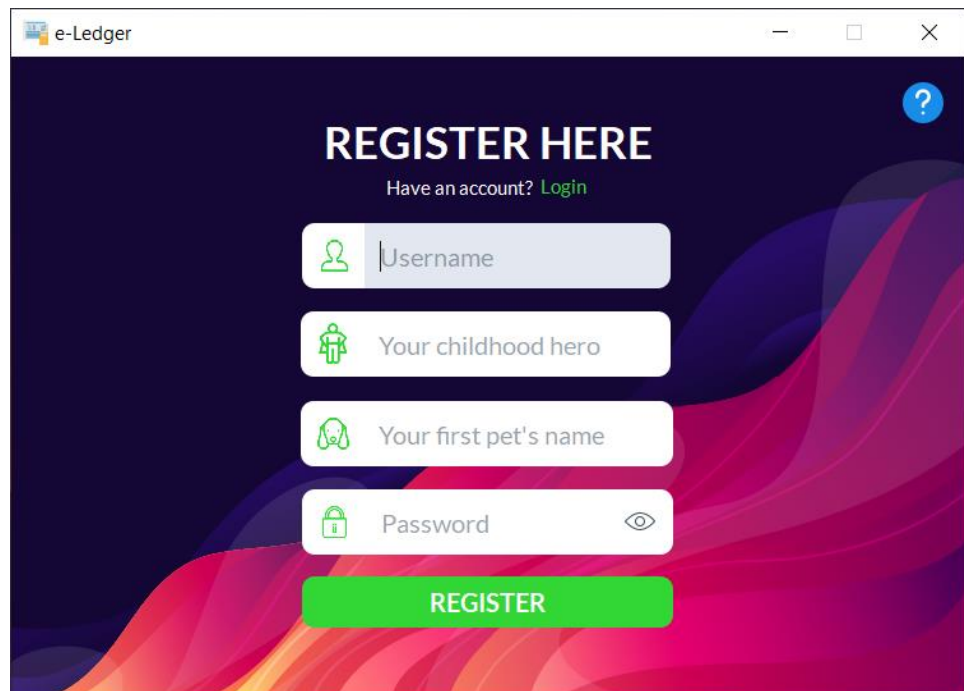
The project was on a digital transaction management system, which helps to keep track of all the monetary work, which is similar to our traditional ledger. We have developed the application by using concepts and technologies like Object Oriented Programming, Caesar Algorithm, SQLite and Qt.

#### **4.1 Features:**

There are many remarkable and extremely significant features developed in our project, 'e-Ledger', which are enlisted below:-

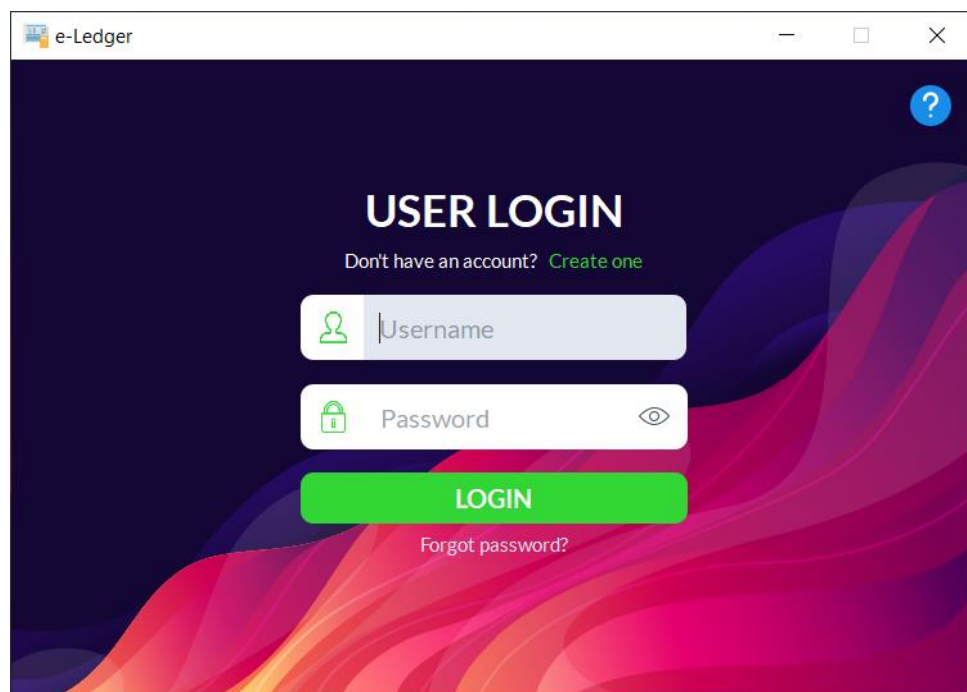
##### **1. Registration and login:**

The program can be used by multiple users by creating their separate accounts. Once they create the account, they can login into the program to manipulate and view their transaction records.



The image shows a web browser window titled "e-Ledger". The main heading is "REGISTER HERE" in white capital letters. Below it, a link "Have an account? Login" is displayed in green. The registration form consists of four input fields, each with a green icon on the left: a person icon for "Username", a child icon for "Your childhood hero", a dog icon for "Your first pet's name", and a lock icon for "Password". A green eye icon is visible to the right of the password field. At the bottom of the form is a large green button labeled "REGISTER" in white capital letters. The background is a dark purple gradient with abstract, flowing, colorful shapes in shades of pink, orange, and yellow.

Figure 4.1. 1 Registration window



The image shows a web browser window titled "e-Ledger". The main heading is "USER LOGIN" in white capital letters. Below it, a link "Don't have an account? Create one" is displayed in green. The login form consists of two input fields, each with a green icon on the left: a person icon for "Username" and a lock icon for "Password". A green eye icon is visible to the right of the password field. At the bottom of the form is a large green button labeled "LOGIN" in white capital letters. Below the button is a link "Forgot password?" in white. The background is a dark purple gradient with abstract, flowing, colorful shapes in shades of pink, orange, and yellow.

Figure 4.1. 2 Login window

## 2. Account recovery

The account of any individual can be recovered even after the user forgets his/her password. The application allows its users to reset the password based on the security questions they answered when creating their account.

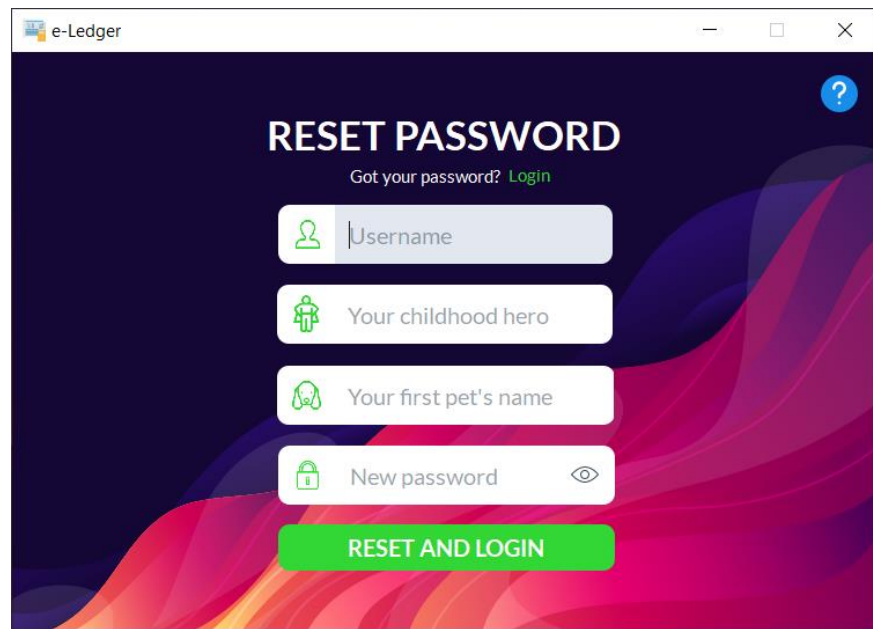


Figure 4.1. 3 Window to reset password

## 3. Safe and Secure:

It is a multi-user program. The data entered and saved in the program have minimum chances of being misplaced. Also, passwords and security answers are encrypted so that other users can not log in or reset passwords.

## 4. Easy-to-Use Interface:

The program has easy to use Graphical User Interface (GUI). Users can also view the manual just by clicking the help icon (icon with a question mark). Furthermore, shortcuts and tooltips have been provided to assist users.



## 6. Sorting and filtering data:

The program allows the user to sort and filter the transaction records based on Transaction ID, Name of buyer/seller, Address, Phone No., Total amount, Received/Paid amount and Receivable/Payable amount. Additionally, the user can also sort the records according to the date of transaction and filter it according to the months.

The screenshot displays the 'e-Ledger' application interface. On the left, a sidebar contains 'General Info' (Account # 6, Date 2020-01-08, Buyer/Seller toggle, Name Durgendra Kushwaha, Address Dangraha Tole, Phone 9842701343) and 'Transaction Amounts' (two input fields with 100 and 0, a Calculate button, and an Other section with Drinks Sold). The main area, titled 'Transaction details', shows a table of 17 transactions. The table columns are Transaction ID, Date, Name, Address, Phone no., Total amount, Received, Receivable, and Remarks. The transactions are sorted by date in descending order. At the bottom, there are controls for sorting (No sort/filter, Sort, Filter), filtering (Buyer, Date), and viewing (Recent first, View, Export). Buttons for Save, Update, and Delete are also present.

Transaction ID	Date	Name	Address	Phone no.	Total amount	Received	Receivable	Remarks	
1	25	2020-01-30	Ashutosh	Banepa	9842723434	200	50	150	Geometry Box
2	24	2020-01-29	Svarnim	Banepa	9842723434	100	100	0	Compass Sold
3	23	2020-01-28	Gahendra Chand	Kathmandu	9811334242	80	70	10	Choclate
4	21	2020-01-27	Ayush Dhungana	Bhaktapur	9811343544	100	100	0	Stationery Item
5	22	2020-01-27	Ayush Dhungana	Bhaktapur	9811343544	100	100	0	Stationery Item
6	20	2020-01-26	Dilip Chaudhary	Saptari	9811234234	150	100	50	Chicken Meat
7	19	2020-01-24	Dipak Kumar	Lahan	9811771823	300	300	0	Chair Sold
8	18	2020-01-23	Dipendra	RaghunathPur	9811771892	100	70	30	Edible Oil 10pack
9	17	2020-01-21	Rangita Singh	HariPur	9807713243	300	300	0	Cosmetic Item
10	16	2020-01-20	Sumitra Kumari	Sarlahi	9807717456	400	350	50	Vegitables Sold
11	15	2020-01-19	Dev Narayan Mahato	Hospital Chowk	9842843035	100	70	30	Wallet Sold
12	14	2020-01-18	Yashodha Kumari Mahato	Ward-3	9834212443	130	130	0	Groceries Items
13	13	2020-01-17	Raja Babu	Dhangadi	9876542342	500	200	300	RajniGhandha
14	12	2020-01-15	Gokul Raj	Lalitpur	9876542342	200	200	0	Toys Sold
15	11	2020-01-14	Paban Singh	BhaghwanPur	9876543234	500	500	0	Sweets
16	10	2020-01-12	Paban Singh	BhaghwanPur	9876543234	300	200	100	Cosmetic Item
17	9	2020-01-11	Binod Singh	Ram Pur	9876543232	30	20	10	Biscuit Sold

Figure 4.1. 5 Records of buyer in descending order of date.

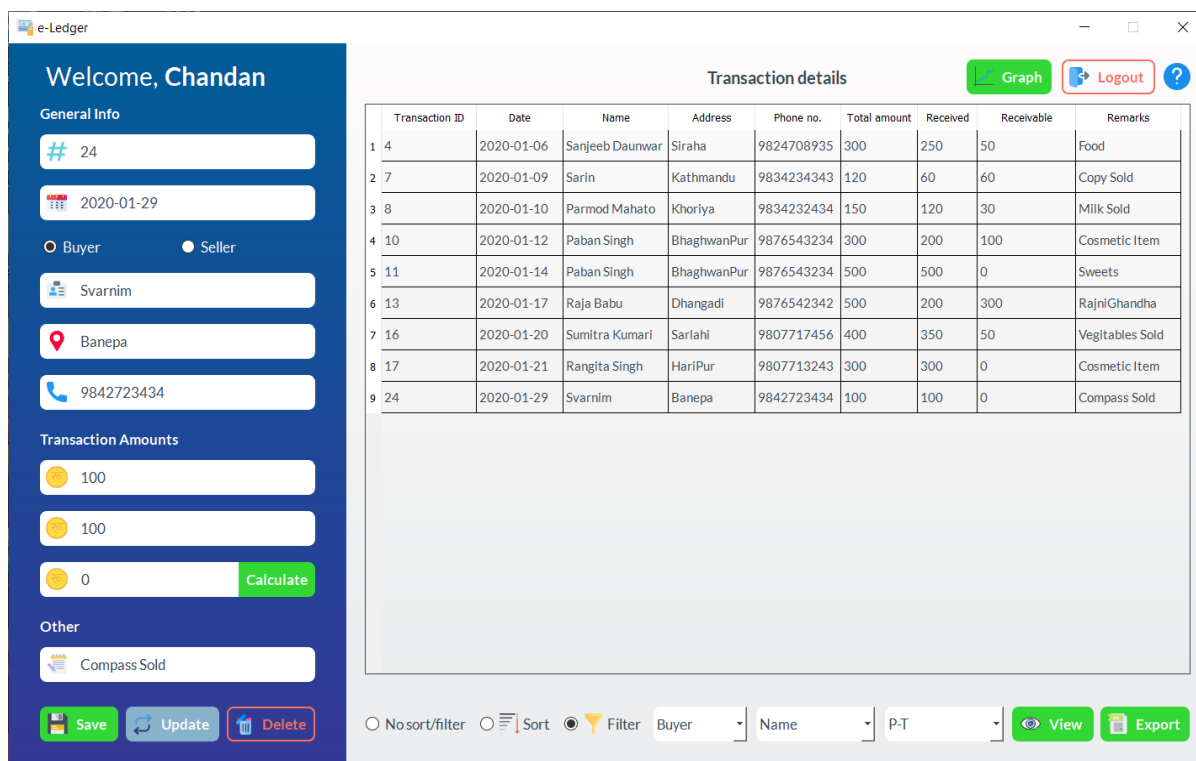


Figure 4.1. 6 Records of buyers whose names start from P to T

## 7. Graphs:

A user can have a quick look of his/her transaction per day in a month. They can view graphs based on Transaction amounts or No. of transactions per day in a month, which are shown in Figure 4.1.7 and 4.1.8 respectively.



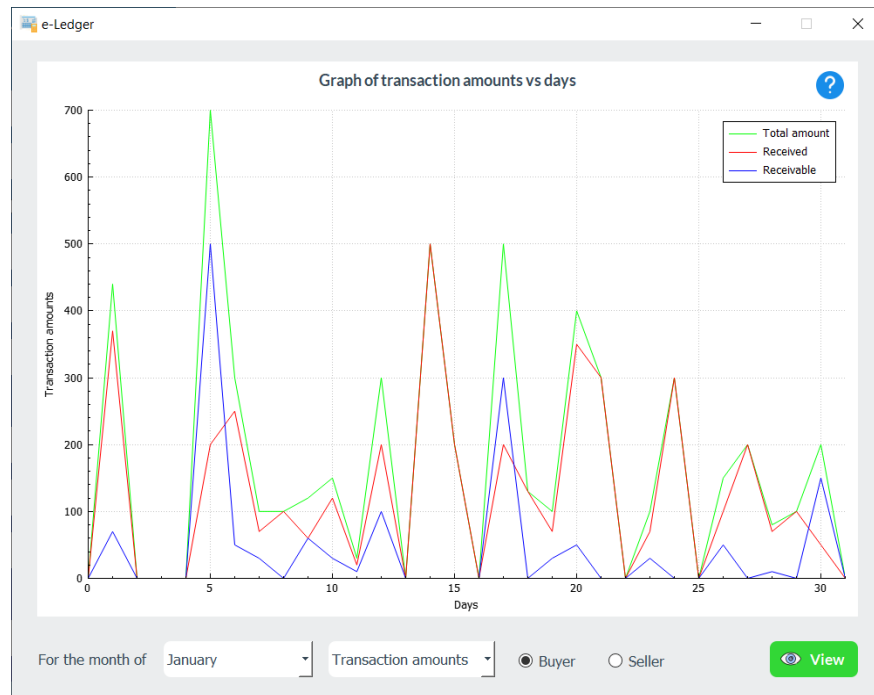


Figure 4.1. 7 Graph of Transaction amounts vs day

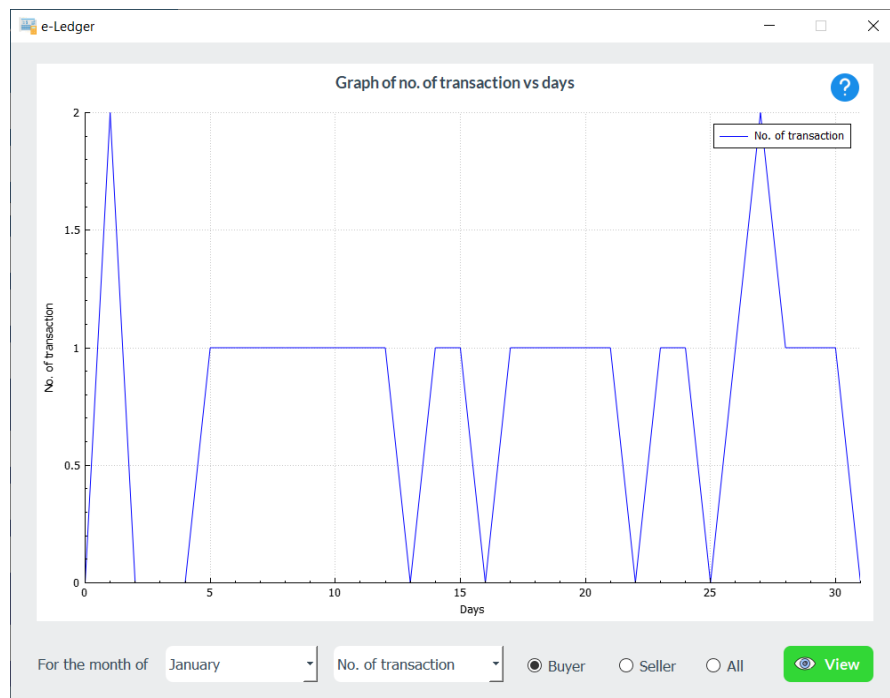


Figure 4.1. 8 Graph of no. of transaction vs day.

## 8. Exporting records

The user can easily export the transaction records with or without sorting or filtering as a CSV file, which can be opened in a spreadsheet software like Microsoft Excel for future reference.

The image shows a Microsoft Excel spreadsheet on the left and a web application interface on the right. The Excel spreadsheet is titled 'Chandan\_buyers\_filtered - Excel' and contains a table with columns: Transaction ID, Date, Name, Address, Phone Number, Total amount, Received, Receivable, and Remarks. The web application interface shows a 'Transaction details' table with the same columns. The table contains 10 rows of data. The web application also has a 'Graph' button, a 'Logout' button, and a 'View' button. The Excel spreadsheet is filtered to show only the 'Chandan\_buyers\_filtered' data.

Transaction ID	Date	Name	Address	Phone no.	Total amount	Received	Receivable	Remarks
4	2020-01-06	Sanjeeb Daunwar	Siraha	9824708935	300	250	50	Food
7	2020-01-09	Sarin	Kathmandu	9834234343	120	60	60	Copy Sold
8	2020-01-10	Parmod Mahato	Khoriya	9834232434	150	120	30	Milk Sold
10	2020-01-12	Paban Singh	BhaghwaniPur	9876543234	300	200	100	Cosmetic Item
11	2020-01-14	Paban Singh	BhaghwaniPur	9876543234	500	500	0	Sweets
13	2020-01-17	Raja Babu	Dhangadi	9876543242	500	200	300	RajniGhandha
16	2020-01-20	Sumitra Kumari	Sariahi	9807717456	400	350	50	Vegitables Sold
17	2020-01-21	Rangita Singh	HariPur	9807713243	300	300	0	Cosmetic Item
24	2020-01-29	Svarnim	Banepa	9842723434	100	100	0	Compass Sold

Figure 4.1. 9 A filtered and exported file opened in Microsoft Excel.

## **Chapter 5: Conclusion and Recommendation**

The final outcome of the semester project work was an application that was convenient for record keeping and information retrieval and simplified accounting. This digital record keeping system could ultimately help in saving materials, time and space used by traditional record keeping. This project was completed in eleven weeks. Throughout the project development phase, we got opportunities to learn many new technologies and concepts like Qt and databases. Thus, it could be said that the objectives of the proposed project had been met by the final outcome.

We would like to thank our supervisor Mr. Satyendra Nath Lohani and our Project Coordinator Ms. Deni Shahi who guided us throughout this project. We appreciate their continuous support and guidance.

### **5.1 Limitations**

The project had to be completed in a limited time constraint and without advanced concepts due to which few needs of the project could not be fulfilled making it our limitation for this application. They are as follows:

1. Our approach to encrypt passwords is not strong enough and can be easily cracked.
2. There could be an even more professional method of resetting passwords for an account rather than relying on just the two security questions.
3. e-Ledger has only been deployed for Windows OS x64bit and needs to be packaged for Linux and Apple Macintosh as well.

### **5.2 Future Enhancements**

The application can be further enhanced in the future by implementing the following aspects:

1. Improvement of the security and privacy of the accounts using better encryption methods.

2. Allowing the user to save his/her more details like contact no. and email address for easy retrieval of the account.
3. Generating codes for the users to recover their accounts on their email or as SMS.
4. Making the application to work online and offline modes.
5. Developing cross platform application.

## References

- intuit. (2018). *Account Plan Comparison / QuickBooks Global*. From QuickBooks:  
<https://quickbooks.intuit.com/global/online-compare/>
- KhataBook. (2020, February 11). *KhataBook*. From KhataBook: <https://khatabook.com/>
- QNB eFinans. (n.d.). *e-Ledger / Electronic Accounting Ledger / QNB eFinans*. From qnb  
finans: <https://www.qnbefinans.com/en/e-ledger>
- Qt. (n.d.). *About Qt - Qt Wiki*. From qt: [https://wiki.qt.io/About\\_Qt](https://wiki.qt.io/About_Qt)
- SQLite. (n.d.). *About SQLite*. From sqlite: <https://sqlite.org/about.html>
- Wave Financial Inc. (n.d.). *Free small business accounting software - Wave Financial*. From  
waveapps: <https://www.waveapps.com/accounting>

## Appendix

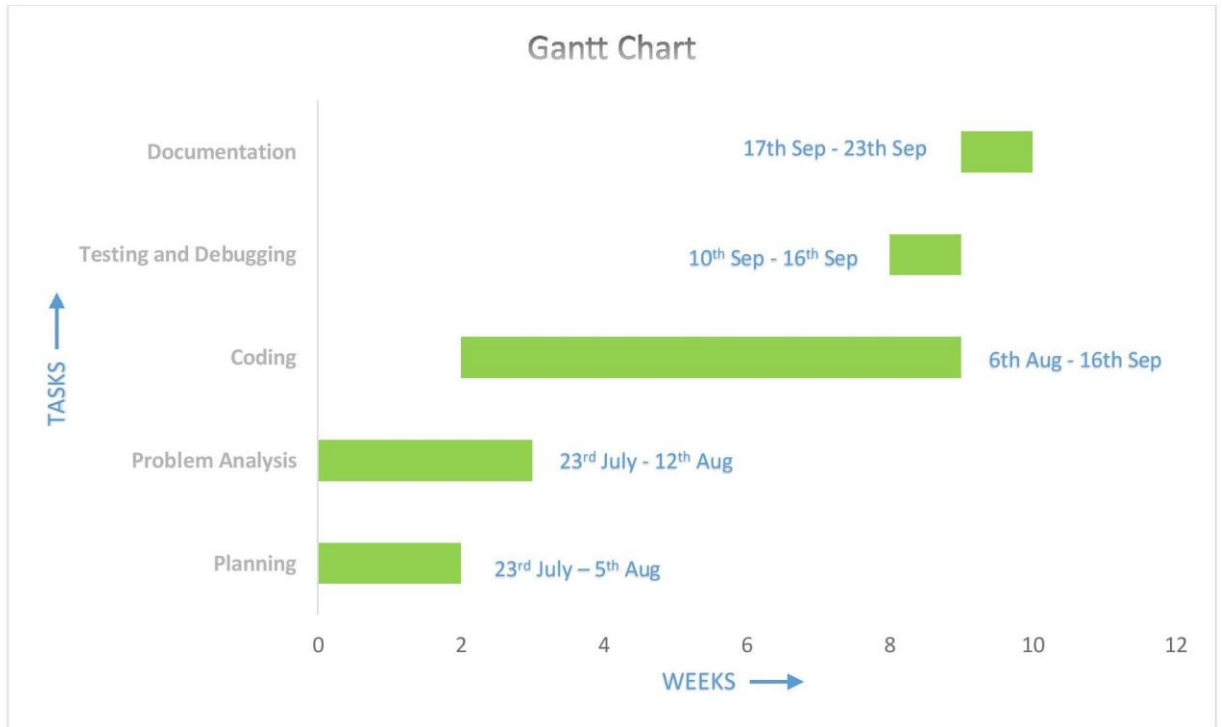


Figure i Gantt chart

S. No.	Tasks	Duration
1.	Planning and Preparation	2 Weeks
2.	Problem Identification and Requirement Analysis	3 Weeks
3.	Coding	7 Weeks
4.	Testing and Debugging	1 Week
5.	Documentation	1 Week

Table 1: Table for Project Timeline

Source code of the project and its binary release can be found in the following link:

[https://github.com/Project-e-Ledger/e-Ledger\\_GUI\\_alpha](https://github.com/Project-e-Ledger/e-Ledger_GUI_alpha)