# ACKNOWLEDGEMENT

It gives us immense pleasure to express our deepest sense of gratitude and sincere thanks to our highly respected and esteemed guide Mr. **Jeetendra Manandhar** for his/ her valuable guidance, encouragement and help for completing this work. His/ her useful suggestions for this whole work and co-operative behavior are sincerely acknowledged.

We would like to express our sincere thanks to **Er.** **Rajan Karmacharya** (Lecturer) for giving us this opportunity to undertake this project. We would also like to thank Mr. **Vishnu Kumar Rana** (Head of Department) for whole hearted support.

We are also grateful to our teachers (teachers name with full designation) for their constant support and guidance. At the end we would like to express our sincere thanks to all our friends and others who helped us directly or indirectly during this project work.

Diwash Sapkota (15175/074)

Mahesh Singh Madai (15184/074)

# 

# ABSTRACT

As an agricultural country, more than two-thirds of the total population are dependent on agriculture and livestock farming for their livelihood. Agriculture has been the mainstay of Nepalese economy, contributing more than 30% of the total national GDP. There is a common feeling among the people to grow crops and rear cattle for fulfilling the family needs. Besides creation of self-employment opportunities, livestock farming helps in increasing soil fertility and generating animal proteins for healthy living. Thus it is important to get the best outcome from livestock farming. In this regard, use of modern technology and proper record keeping is a must.

Cattle Management System is a project to computerize the database of an agricultural farm so that it facilitates the farm-owner as well as other individuals and organizations to keep and retrieve records of each cattle in easy, fast, systematic and reliable way. The database is administered by the farm-owner to study/update the current status of his/her farm as well as to make special preparations for future. In most agricultural farms in Nepal, the records are kept manually. So this project is intended to help farmers collect and analyze data on all aspects of their farm operations.

***Keywords:*** *Farming, Cattles, Database, Farm-owner, Visitor, Commercial, Monitor*

[ACKNOWLEDGEMENT i](#_Toc513649206)

[ABSTRACT ii](#_Toc513649207)

[List of Figures: iv](#_Toc513649208)

[Chapter 1: Introduction 1](#_Toc513649209)

[1.1 Background: 1](#_Toc513649210)

[1.2 Problem Statement: 1](#_Toc513649211)

[1.3 Objective and Scope: 2](#_Toc513649212)

[1.3.1 Objective: 2](#_Toc513649213)

[1.3.2 Scope: 2](#_Toc513649214)

[1.4 Project Features: 2](#_Toc513649215)

[1.5.1 Requirement Analysis: 2](#_Toc513649216)

[1.5.2 Feasibility Analysis: 3](#_Toc513649217)

[1.6 System Requirement- Hardware and Software Platform: Hardware Requirement: 4](#_Toc513649218)

[Software Requirement: 4](#_Toc513649219)

[Chapter 2: Literature Review 5](#_Toc513649220)

[2.1 Background and Contextual Research 5](#_Toc513649221)

[2.2 History 5](#_Toc513649222)

[2.3 National Context 6](#_Toc513649223)

[2.4 Related Works 7](#_Toc513649224)

[2.5 Review of Tools 7](#_Toc513649225)

[Chapter 3: System Development 8](#_Toc513649226)

[3.1 Project Tools 8](#_Toc513649227)

[3.2 System Analysis 8](#_Toc513649228)

[3.3 System Design 8](#_Toc513649229)

[3.3.1 Waterfall Model: 8](#_Toc513649230)

[3.2 Algorithm 9](#_Toc513649231)

[3.3 Flowchart 10](#_Toc513649232)

[3.4 Project Schedule: 11](#_Toc513649233)

[3.4 Testing: 11](#_Toc513649234)

[Chapter 4: Result Analysis 13](#_Toc513649235)

[4.1 Results 13](#_Toc513649236)

[4.2 Critical Analysis 18](#_Toc513649237)

[4.3 Limitation and Future Enhancement 19](#_Toc513649238)

[4.4 Conclusion: 19](#_Toc513649239)

[References 20](#_Toc513649240)

# List of Figures:

[Figure 1 Waterfall Model 8](#_Toc513648285)

[Figure 2 Flowchart 10](#_Toc513648286)

[Figure 3 Gantt chart 11](#_Toc513648287)

[Figure 4 Login Screen 13](#_Toc513648288)

[Figure 5 Welcome Screen 14](#_Toc513648289)

[Figure 6 Entry Screen 15](#_Toc513648290)

[Figure 7 Display Screen 16](#_Toc513648291)

[Figure 8 Edit Screen 16](#_Toc513648292)

[Figure 9 Search Screen 17](#_Toc513648293)

[Figure 10 Delete Screen 18](#_Toc513648294)

# Chapter 1: Introduction

## Background:

Being an agricultural country with more than sixty percent of the total population involved in farming and animal husbandry, the economy of Nepal is largely dependent on agriculture. Livestock farming is an important aspect of Nepalese agricultural practice. This sector constitutes more than 30% of the national GDP. With the improving economic status of people, the demand of agricultural products is increasing daily. In such condition, livestock farms have to be competitive enough to supply the demands. This can only be done by using modern technologies and a proper record keeping. [1]

Cattle Management System is a database project to computerize the animal records of a livestock farm so that it may help the farm owner to identify the cattle as well as retrieve the individual cattle details easily. This project is completed using the Microsoft Access 2016. [2]

Using this tool, the farm owner can store, retrieve and analyze records about several aspects of his/her farm in reliable and systematic way. The database is administered by the farm-owner to study/update the current status of his/her farm as well as to make special preparations for future.

## 1.2 Problem Statement:

Although Livestock farming has been a crucial component of farming, providing manure, animal protein and generating employment opportunities, there has been so little development in this field. Most of the farms are still using traditional methods for operating their farms and maintaining records. People mostly rely on their memory for keeping cattle records, which is very unreliable in the long run. Besides, the records people keep mostly belong to sales and production only. However, there are several factors that affect the productivity of a cattle.

In more advanced farms, the records are kept manually. Though this is good to some extent, this method is not reliable for long period. Primarily, the record is more prone to errors due to tedious and complex process in updating it. The farm owner has to spend lots of time accessing and updating each and every data regularly. The data also may get lost or damaged during the course of time. And it is more complex process to recover the lost data. Sometimes it is even impossible. Besides, the probable customers have to spend much time regularly visiting the farms so as to get the information about their desired cattle. [3]

## Objective and Scope:

### 1.3.1 Objective:

The objectives of Cattle Management System are listed below:

* To help farm-owner to keep track of its cattle.
* To help farm-owners generate records easily for visitors.
* To a build a system that can keep track of milk records effectively.

### 1.3.2 Scope:

This database project is designed to overcome the drawbacks of traditional manual record keeping system. Cattle Management system allows users with basic computer knowledge to view, edit and analyze the records of individual cattle. Besides, the user can also update and view the daily sales record of milk, expenditures occurring daily, and other several important aspects of farm. The user can also create a backup of the original data, so that s/he can recover the original data in case it gets lost or damaged.

## 1.4 Project Features:

This project will allow the following features to the farm owners

* An organized way of keeping all the details of the cattle efficiently.
* The addition, deletion, update and saving the new data is much more efficient.
* The daily record of milk: milk produced, rate, fat amount and the total sales can be recorded daily as per the need.
* The data of the particular cattle can be searched through the input by the user.  
  1.5 Requirement Analysis and Feasibility:

### 1.5.1 Requirement Analysis:

Before starting the development of the new system, different requirements are taken into consideration. As our project is solely based on cattle farming, the requirement analysis is done concentrating on various aspects of cattle farming that are prevalent in the country. The requirement analysis is done with following outline:

#### 1.5.1.1 Study of Existing System:

Although Livestock farming has been a crucial component of farming, providing manure, animal protein and generating employment opportunities, there has been so little development in this field. Most of the farms are still using traditional methods for operating their farms and maintaining records. People mostly rely on their memory for keeping cattle records, which is very unreliable in the long run. Besides, the records people keep mostly belong to sales and production only. However, there are several factors that affect the productivity of a cattle. [4]

#### 1.5.1.2 Source of Data Collection:

* **Field Visit:**

It is certain that the information about existing systems is best understood if they are observed in real world situations. This moreover enhances the practical approach while studying about any dedicated matter. Chitwan and Nawalparasi are the districts renowned to have number of agricultural farms. So, we preferred visiting a cow farm in Nawalparasi to collect data sample for our project.

* **Interview:**

Interacting with concerned people in the field to comprehend better awareness about the systems is always preferred after the field visit. Various questions related to the study are raised, and the experts try their best to make things clear. As part of our interview, we questioned the farm owner about the database management system used by the farm and the difficulties in managing the data entry system.

* **Internet:**

The internet always is equipped with myriad of information and ideas. To conclude the study and revise the ending, internet was used in order to understand better about the database and its various aspects. We took reference to online platforms for making the database more smooth and systematic. [5] [6] [7]

### 1.5.2 Feasibility Analysis:

Feasibility Analysis is an assessment of the practicality of a proposed project. It provides the degree of viability of a proposed project. A feasibility analysis helps us determine the value of the proposed project, determine whether or not there is a market for the proposed project, determine if the proposed project is financially viable, and eventually, decide whether or not it is worth investing time and money into the proposed project.

In short, a feasibility analysis evaluates the projects potential for success.

Following Feasibility Analysis was performed prior to working on the project:

#### 1.5.2.1 Operational Feasibility:

The database application solves problem faced by the farm owners that was underlined as the problem statement. The farm owners can easily use the feature with the little knowledge of the computer. Furthermore, features such as login, save, add, go to records helps the database system to be operationally feasible.

#### 1.5.2.2 Economic Feasibility:

The database application is available and easily accessible when once installed on the computer which makes it economically feasible as the farm owners do not have to buy reinstall many times. Farm owners only require the computer with the below mentioned specifications.

#### 1.5.2.3 Technical Feasibility:

The database application is technically feasible; complies with current technology, including both the hardware and the software. The database application is supported by almost all latest computers with minimum hardware and software requirement.

## 1.6 System Requirement- Hardware and Software Platform: **Hardware Requirement:**

* 512 RAM or higher
* Dual Core processor or higher
* USB port
* Input Device: Mouse, Keyboard
* Output Device: Monitor

### Software Requirement:

* OS: Windows XP/7/8/10

# Chapter 2: Literature Review

## 2.1 Background and Contextual Research

In the past, there have not been any development in the method of record keeping for cattle farms. The farmers have been dependent in their memory regarding the cattle details. They observe the physical appearance, aggressiveness, sound made by cattle, grazing habits, etc. over a period of time and assign a name to cattle based on these characteristics. Then they memorize the milk productivity of each cattle. Since farm-owners themselves handle and milk the cattles in most farms, they can either remember or guess the milk produced by each of their cattle. [8]

Considering the needs for requirement of database management in agriculture, many individuals and organizations have taken an initiative to create a database management software for agriculture. We found out several such projects like "ICT in Agriculture", [9] "AgriInfo", "Agriculture Information and Communication Center", "Agricultural Information Management System", "Krishi Koseli" and many more. During the study of these projects, we found out that almost every of these projects are developed for the benefit of farmers and keeping track of crops and their production. While research have been done about the performance of livestock farms and to increase their productivity, there are several technologies developed to increase the production of livestock farms. However, regarding livestock database management, there has been very little development. We found the concept of livestock database management in the book entitled "Non-ruminant production and management", we found the manual method of database management for livestock farms. Though there are no database management software focused for cattle farms in Nepal, we found that organizations like FAO and Agriculture Development Bank have been training the farmers for manual record keeping. In this method, the data about several aspects of farm are kept in a notebook. [10]

## 2.2 History

C is a general-purpose programming language which features economy of expression, modern control flow and data structures, and a rich set of operators. C is not a "very high level" language, nor a "big" one, and is not specialized to any particular area of application. But its absence of restrictions and its generality make it more convenient and effective for many tasks than supposedly more powerful languages.

The history of C programming language is quite interesting. C was originally designed for and implemented on the UNIX operating system on the DEC PDP-ll, by Dennis Ritchie. C is the result of a development process that started with an older language called BCPL. BCPL was developed by Martin Richards, and it influenced a language called B, which was invented by Ken Thompson. B led to the development of C in the 1970s.

For many years, the de facto standard for C was the version supplied with the UNIX operating system. In the summer of 1983 a committee was established to create an ANSI (American National Standards Institute) standard that would define the C language. The standardization process took six years (much longer than anyone reasonably expected). [11]

The ANSI C standard was finally adopted in December 1989, with the first copies becoming available in early 1990. The standard was also adopted by ISO (International Standards Organization), and the resulting standard was typically referred to as ANSI/ISO Standard C. In 1995, Amendment 1 to the C standard was adopted, which, among other things, added several new library functions. The 1989 standard for C, along with Amendment 1, became a base document for Standard C++, defining the C subset of C++. The version of C defined by the 1989 standard is commonly referred to as C89.

During the 1990s, a new standard for C was being developed. It was the 1999 standard for C, usually referred to as C99. In general, C99 retained nearly all of the features of C89. The C99 standardization committee focused on two main areas: the addition of several numeric libraries and the development of some special-use, but highly innovative, new features, such as variable-length arrays and the restrict pointer qualifier. These innovations have once again put C at the forefront of computer language development. [12]

## 2.3 National Context

At present, very few developments have taken place regarding the awareness for the importance of record keeping. However, there are still lack of proper record keeping softwares. We also found out that some of the cattle farms have been using manual method of keeping records involving daily milk production, milk sales and manure sales, daily expenditure, date of insemination and reproduction of cattle, valuation of cattle by insurance company and so on. The farmers keep these records in a notebook, which they can access whenever necessary. They make plans and policies of their farms based on the records in their daily notebook. Regarding the recognition of cattle, farmers observe the physical appearance of each cattle, their aggressiveness, milk productivity, grazing habits and the sound produced by them for certain period of time. After this, farmers assign a memorable name to each cattle based on above mentioned characteristics. [13] [9]

## 2.4 Related Works

At international level, there have been several developments regarding the database management of a cattle farm. We found several websites, database softwares and journals in this regard. First and foremost, 'Tambero' [14]is one of the most popular sites for the effective management and handling of individual cattle records. It records the details like Herd, Breed, Milking record, health records, heat records and many more. Besides this, the user can extract the details of individual cattle in spreadsheet and other formats.

CattleRecs is also software that used by overseas cattle farmers. CattleRecs offer a technology-based solution that allows farmers to keep up in the rapidly-changing cattle business. This program was created to meet the record keeping needs of cattle farmer. CattleRecs is based on Microsoft Access technology for maximum compatibility with Microsoft Windows. In addition, records and reports can easily be exported to Excel, Word/text and PDF formats. [14]

## 2.5 Review of Tools

While most of the database projects regarding cattle farm management are done in Microsoft Access, we discuss about the reviews of this tool. Actually, Microsoft Access is the industry leading desktop database creation tool for quick creation of relational SQL databases. It is an excellent tool for those not familiar with SQL - though add-ins, frameworks, templates & new Database Builder tools can help further simplify that, so truly don't need to learn SQL or VBA. MS Access is by far the easiest to use for rapid database creation and most empowering for non-technical users to client.

MS Access is great for Excel users looking to scale-up from maintaining and merging dozens of copies of workbooks each with dozens of worksheet tabs. Beyond that, even SQL pros use it for easily prototyping, as can easily upscale to SQL Server, Azure SQL or other hosted databases using various tools available.

One of the key benefits of MS Access is that you can build, edit and maintain web/cloud/mobile-enabled multi-user databases entirely yourself, without requiring cost or hassle or involving IT or contracting consultants every time you want to make even the most minor change, as is the case for most subject matter experts (non-developers) for most other database solutions.

# Chapter 3: System Development

## 3.1 Project Tools

* Dev C++: used for coding and compiling purpose
* Photoshop for Image editing and Diagrams
* MS-Word: used for documentation
* MS-Excel for making Gantt Chart

## 3.2 System Analysis

The proposed system that we developed for Jay Messiah Cow farm will be efficient in terms of time consuming and less cost as the handwritten database is replaced by the computerized database.

In General, our proposed system will be made from Code Block: An IDE for C and C++ programs. The proposed system is simple, interactive and has user-friendly interface such that even those with little knowledge of computer can easily operate it.

## 3.3 System Design

### 3.3.1 Waterfall Model:

Waterfall model is an example of a Sequential model. In this model, the software development activity is divided into different phases and each phase consists of series of tasks and has different objectives. [15]

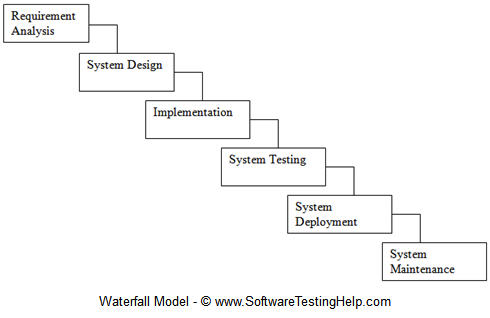


Figure 1 Waterfall Model

## 3.2 Algorithm

Algorithm for the main menu of the program

Step 1: Start

Step 2: Login; Enter Username & Password

Step 3: If Username & Password correct, display Menu Screen

Else try again

Step 3: Enter the choice from 1-5

Step 4: Is input 1?

True- Data entry the record

False – go to step 5

Step 5: Is input 2?

True- Display record.

False- Go to step 6.

Step 6: Is input 3?

True- Edit the record.

False – go to step 7.

Step 7: is input 4?

True – Search in the record.

False - Go to step 8.

Step 8: Is input 5?

True- Delete the record.

False – Go to step 9.

Step 9: Is the input 6?

Display “Invalid Choice”

Step 10: End

## 3.3 Flowchart [16]

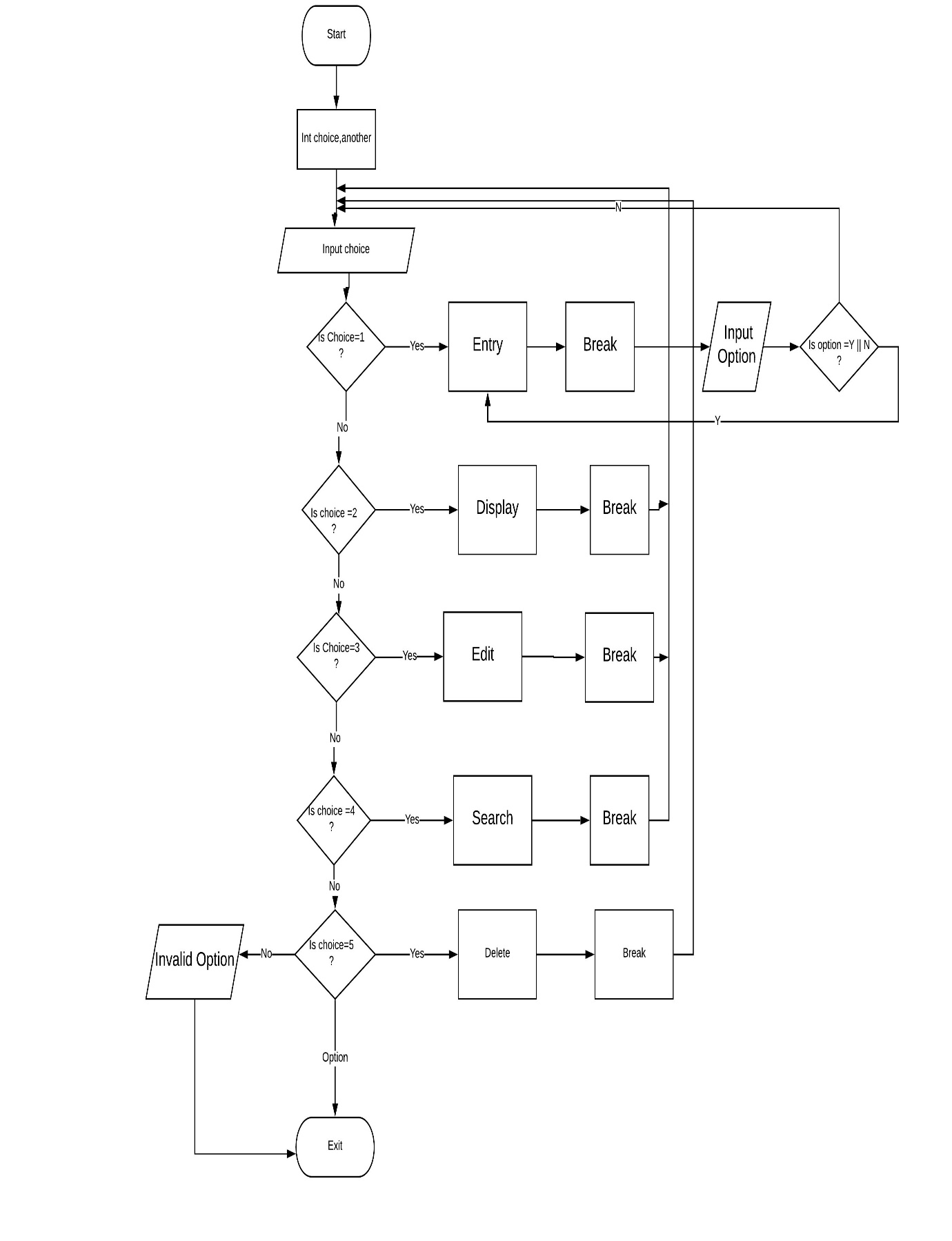


Figure 2 Flowchart

## 3.4 Project Schedule:

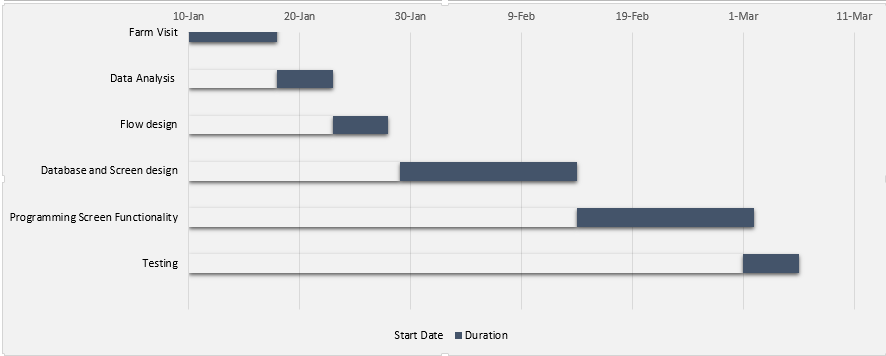
**Gantt chart:**

Figure 3: Gantt chart

## 3.4 Testing:

**3.4.1 Unit Testing**

Unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application.

**3.4.2 Integration Testing**

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group . It occurs after unit testing and before system testing.

**3.4.3 System Testing**

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic. The system testing cases that have been done are as follows:

Table 1 System Testing

|  |  |  |
| --- | --- | --- |
| 1 | To check whether program runs or not | successful |
| 2 | To check if password display menu takes password or not | successful |
| 3 | To check if the program menu displays all menu options or not | successful |
| 4 | To check if the options are right or not | successful |
| 5 | To check id the adding of cattle records are stored or not | successful |
| 6 | To check if the display menu displays the record or not | successful |
| 7 | To check if the update option in the main menu works properly or not | successful |
| 8 | To check if the search option in the main menu works or not | successful |
| 9 | To check if the delete option in the main menu works or not | successful |
| 10 | To check if the delete option in the main menu works or not | successful |

# Chapter 4: Result Analysis

4.1 Results

**Login**

char userid[]="c",password[]="1",p[15],u[15];

int n=1,a,b;

printf("\tEnter Username & Password(3 tries only)\n");

getch();

while(n<=3)

{

printf("\n\n\tUSER ID: ");

scanf("%s",u);

printf("\n\tPASSWORD: ");

scanf("%s",p);

a=strcmp(u,userid);

b=strcmp(p,password);

if(a==0&&b==0)

{

printf("\n\tYou have logged in successfully!!!");

h=1;

break;

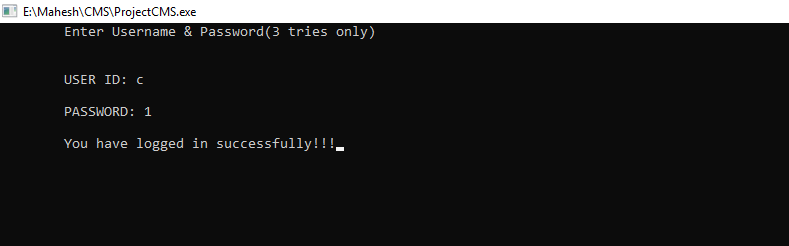


Figure 4 Login Screen

**Welcome Screen**

printf("Welcome\n\t\t\t\t\t\t\t To \n \t\t\t\t\t\tCattle Management System"); printf("\n=============================================================================================================");

printf("\n1.Entry of Record \n2.Display \n3.Edit \n4.Search \n5.Delete the record\n");

printf("\nEnter your choice:");

scanf("%d",&ch);

printf("\n\n");

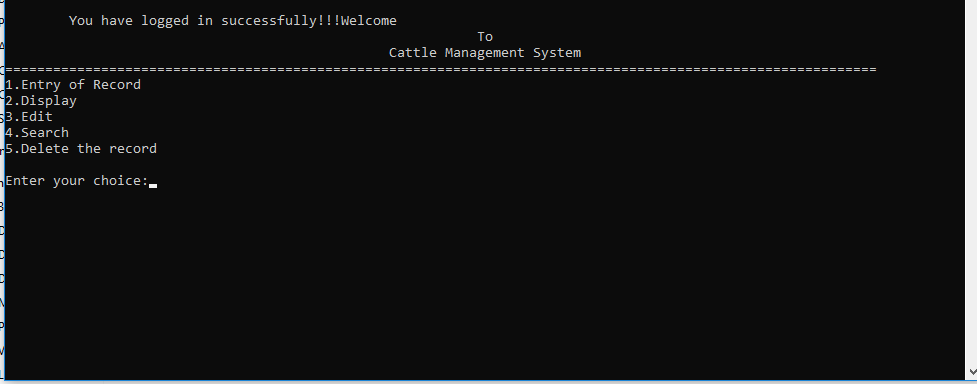
****

Figure 5 Welcome Screen

**Entry Screen**

printf("\nEnter the ID: ");

scanf("%s",cow.ID);

fflush(stdin);

printf("\nEnter the Name: ");

scanf("%[^\n]",cow.name);

printf("\nEnter the Valuation: ");

scanf("%d",&cow.valuation);

fflush(stdin);

printf("\nEnter Breed Type: ");

scanf("%[^\n]",cow.breed);

fflush(stdin);

printf("\nEnter the Date of Birth: "); scanf("%d/%d/%d",&cow.dob.day,&cow.dob.month,&cow.dob.year);

fflush(stdin);

printf("\nEnter the Age: ");

scanf("%d",&cow.age);

fflush(stdin);

printf("\nVaccines Given[Y/N]: ");

scanf("%s",cow.vaccines);

fflush(stdin);

printf("\nEnter the Color: ");

scanf("%[^\n]",cow.color);

fflush(stdin);

printf("\nEnter the Herd Type: ");

scanf("%[^\n]",cow.herdType);

printf("\nMilk Record: ");

printf("\n\t\tMilk Produced: ");

scanf("%d",&cow.daily.milkProduced);

printf("\n\t\tMilk Sold:");

scanf("%d",&cow.daily.milkSold);

printf("\n\t\tRate: ");

scanf("%d",&cow.daily.rate);

printf("\nSales Amount :");

scanf("%d",&cow.daily.sales);

printf("\nAdd another record(y/n): ");

another = getche();

fwrite(&cow,sizeof(struct record),1,cms);

fflush(stdin);

****

Figure 6 Entry Screen

**Display Screen**

printf("ID\tName\t\tValuation\tBreed\tDate of Birth\tAge\tVaccines\tColor\tHerd Type\n");

printf("--------------------------------------------------------------------------------------------------------\n"); printf("%s\t%s\t\t%d\t\t%s\t%d/%d/%d\t%d\t\t%c\t%s\t%s",cow.ID,cow.name,cow.valuation,cow.breed,cow.dob.day,cow.dob.month,cow.dob.year,cow.age,cow.vaccines,cow.color,cow.herdType);

printf("\n");

printf("\n\n\nMilk Data:\n");

printf("Milk Produced\t\tMilk Sold\t\tRate\t\tTotal Sales\n"); printf("%d\t\t\t%d\t\t\t%d\t\t\t%d\n\n",cow.daily.milkProduced,cow.daily.milkSold,cow.daily.rate,cow.daily.sales);

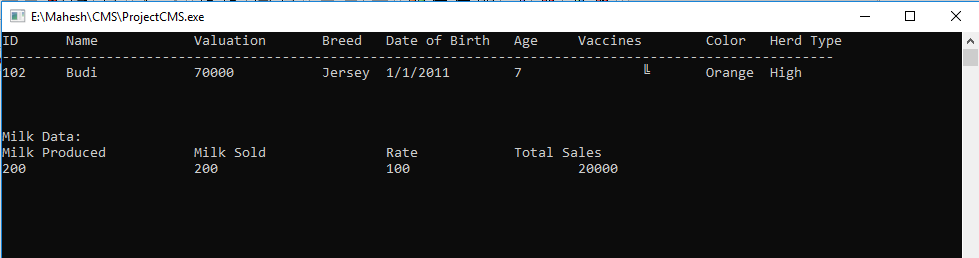
****

Figure 7 Display Screen

**Edit Screen**

while(another=='y')

{

printf("\nEnter the Cow ID to modify: ");

scanf("%s",cowID);

rewind(cms);

while(fread(&cow,sizeof(struct record),1,cms)==1)

{

Input statements;

}

}

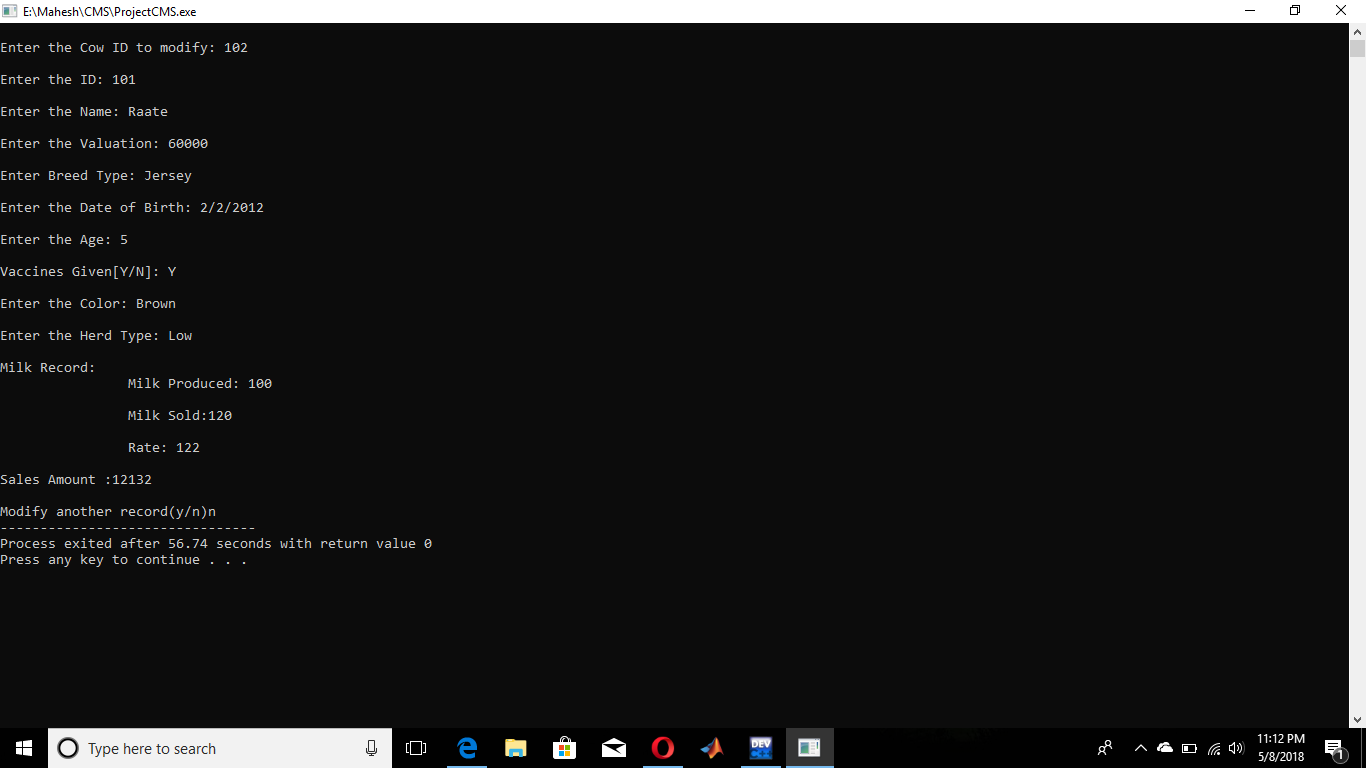
****

Figure 8 Edit Screen

**Search Screen**

while(another=='y')

{

printf("\nEnter the Cow ID to search: ");

scanf("%s",cowID);

rewind(cms);

while(fread(&cow,sizeof(struct record),1,cms)==1)

{

display statements;

}

}

****

Figure 9 Search Screen

**Delete Screen**

while(another == 'y')

{

printf("\nEnter the Cow ID delete: ");

scanf("%s",cowID);

cmsNew = fopen("Temp.dat","wb");

rewind(cms);

while(fread(&cow,sizeof(struct record),1,cms) == 1)

{

if(strcmp(cow.ID,cowID) != 0)

{

fwrite(&cow,sizeof(struct record),1,cmsNew);

}

}

fclose(cms);

fclose(cmsNew);

remove("CMS.DAT"); /// remove the orginal file

rename("Temp.dat","CMS.DAT");

cms = fopen("CMS.DAT", "rb+");

printf("Delete another record(y/n):");

fflush(stdin);

another = getche();

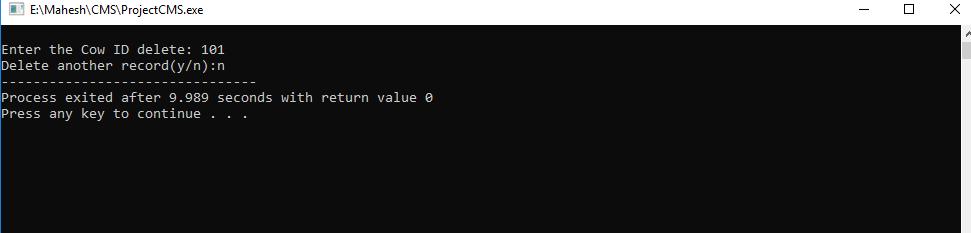
****

Figure 10 Delete Screen

## 4.2 Critical Analysis

During the designing of our project, we took references from few of the similar projects and trends of maintaining cattle records. We did a comparative study of those projects and checked their feasibility and system requirements of those projects. The analysis of those projects led to the development to "Cattle Management System" with added features. The critical analysis of two of them is described below.

In this system, the farm-owners are supposed to observe the physical characteristics of cattles. As a result, farmers spend lots of time observing the appearance and habits of each cattle. They memorize the milk productivity of each cattle. The cattle behavior and grazing is used as the data of knowing the cattle health and preceding condition. However, this system is suitable only for very small farms. It is actually very hard to memorize all the cattle details in a large farm. Besides, when the farm is sold to a new person, the new person has to spend huge amount of time observing and memorizing all the cattles, which is a really difficult job. Human memory is not reliable for a long term basis. Thus, maintenance of systematic records of all the aspects of farm is of prime importance for better productivity and results.

CattleRecs is a software mostly used in foreign countries. This is also a project developed on Microsoft Access platform. This project was actually designed to meet the record keeping needs of cattle farmer. Besides regular updating and analysis of data, the records and reports can easily be exported to Excel, Word/text and PDF formats. However, this project does not have the functionality through which the farm owner can update the regular milk records. While sales is an important aspect of a farm, a proper sales record is crucial in making preparations for future. Besides, there is no form for keeping records of dead and sold cattles. The only way the farmer can update the database when a cattle is dead or sold is by deleting the record. Thus, the owner cannot keep records of prime reasons for the death of cattles, which may increase even further in future.

Thus, studying the drawbacks of current trend and few similar projects, we were able to develop Cattle Managment System. This system primarily fulfils the need for record keeping in a livestock farm. Also, the farm owner can have access to large aspects of his/her farm. Besides individual cattle records, the farm owner can keep record of daily milk record as well as dead and sold cows. This helps the farm owner analyze daily production, health status of individual cattle and the primary reason for death of cattles and decrease in milk production.

## 4.3 Limitation and Future Enhancement

In this section, we discuss about the limitations of Cattle Management System. Though this project could be a great assistance for farm owners to maintain and analyze cattle records of their farms, there are still few drawbacks of this project. They are listed below:

* Only the individual with computer knowledge can handle the database using this project. However, there are many livestock farms where the owners are still computer-illiterate.
* The visitors cannot access the database on their own. Only the authorized users can extract the cattle details and handle it to the visitors.

In future, we hope to provide basic computer training to the customers if they wish to apply this project in their farms. Also, we are working on the possible method to create a visitor login form, through which any visitor can have access to limited cattle data as the administrator sets.

## 4.4 Conclusion:

In conclusion, Cattle Management System is a database project made by using C programming in order to assist livestock farm owners maintain and analyze all the records of farm. The records can be maintained by a user with basic computer knowledge. This method of record keeping is very reliable, systematic and upgradable, through which the growth of individual cattle and the productivity of the farm can be tracked and controlled as per the necessity

# **References**

|  |  |
| --- | --- |
| [1] | "Cow farming," [Online]. Available: http://www.fao.org/nepal/fao-in-nepal/nepal-at-a-glance/en/. |
| [2] | Tutorialspoint, ms\_accese\_tutorial, http://www.tutorialspoint.com. |
| [3] | SimonFreiz, "MS Access 2016 Tutorial," [Online]. Available: http://www.youtube.com. |
| [4] | "doanepal," [Online]. Available: http://www.doanepal.gov.np. |
| [5] | "MS Access Tutorial," [Online]. Available: http://www.youtube.com. |
| [6] | "Cow Farm," [Online]. Available: http://www.agricultureinnepal.com. |
| [7] | Unaab, "Unaab Journal," *Cattle Rearing,* vol. 2. |
| [8] | "Nepalese Cattle Rearing," [Online]. Available: https://thehimalayantimes.com/opinion/livestock-farming-enhance-economy/. |
| [9] | "ICT In Agriculture," 2015. [Online]. Available: http://elibrary.sxc.edu.np/pages/search.php. |
| [10] | "Cattle Farming," [Online]. Available: http://www.dairyglobal.net. |
| [11] | "Introduction to C," [Online]. Available: https://www.le.ac.uk/users/rjm1/cotter/page\_06.htm. |
| [12] | "History of C," [Online]. Available: http://www.codingunit.com. |
| [13] | "Catlle Rearing in Nepal," [Online]. Available: http://www.krishiinepal.com. |
| [14] | "Tamberoo," [Online]. Available: http://www.tambero.com. |
| [15] | "Waterfall Model," Software Testing Help, [Online]. Available: https://www.softwaretestinghelp.com/what-is-sdlc-waterfall-model/. |
| [16] | "Flowchart Draw," [Online]. Available: http://www.lucidchart.com. |