School of Computer Science and Technology

Spatial Data Management for Livestock Distribution in Ethiopia

By: Sisay Webshet Yemnue
Mulugeta Solomon Retta

A Project Paper Submitted to the Graduate Programme Research Office of HiLCoE in Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Science

February, 2015
Spatial Data Management for Livestock Distribution in Ethiopia

BY: Sisay Webshet Yemnue
Mulugeta Solomon Retta

Name and Signature of members of the Examining Committee

<table>
<thead>
<tr>
<th>No.</th>
<th>Members of the Examining</th>
<th>Name</th>
<th>Date</th>
<th>Signature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Examiner-1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Examiner-2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Chairperson</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Acknowledgement

First and foremost, we would like to express our sincere gratitude to our advisor Dr. Getachew Berhan and Dr. Mesfin Kifle for their best valuable guidance and help to the progress and completion of our project work. Our heartfelt thanks and appreciation also extend to who generously spared their time for providing necessary information and explanation.

We would also like to thank our colleague Yassin Getahun from the LIVES project at ILRI for his support and encouragement during our study and throughout our work.

Finally, we thank all organizations and individuals who helped us during the course of producing this study document.
# Table of Contents

List of Tables ................................................................................................................................. v
List of Figures .................................................................................................................................. vi
Abbreviations ................................................................................................................................. vii
Abstract .......................................................................................................................................... viii

CHAPTER ONE ................................................................................................................................. 1

1. Introduction .................................................................................................................................. 1
   1.1. Overview ............................................................................................................................. 1
   1.2. Statement of the problem ................................................................................................. 3
   1.3. Objective of the project ................................................................................................. 4
   1.4. Data and Methodology ................................................................................................. 4
   1.5. Justification of the Project ............................................................................................ 5
   1.6. Scope and Limitation ...................................................................................................... 6
   1.7. Significance of the project ............................................................................................. 6
   1.8. Document organization ................................................................................................. 7

CHAPTER TWO .................................................................................................................................. 8

2. Literature review ......................................................................................................................... 8
   2.1. Mapping the Global Distribution of Livestock ............................................................. 8
   2.2. Livestock in Ethiopia ....................................................................................................... 9
   2.3. Overview of GIS ............................................................................................................ 11
   2.4. Application of GIS in the livestock sector ..................................................................... 14
   2.5. Web-Based GIS ............................................................................................................ 15
   2.6. Web Based APIs ........................................................................................................... 15
   2.7. Web-Based GIS Mapping Technologies ....................................................................... 16
   2.8. Spatial Data Management System (SDMS) ................................................................. 17
   2.9. Mapping Livestock in Ethiopia .................................................................................... 18

CHAPTER THREE ............................................................................................................................ 19

3. System Analysis ......................................................................................................................... 19
List of Tables

Table 1. Login use case description ................................................................. 25
Table 2. User registration use case description .............................................. 25
Table 3. ViewMap use case description .......................................................... 26
Table 4. Uploadshapefile use case description ............................................... 26
Table 5. Adduser use case description ............................................................ 27
Table 6. DeleteUser use case description ....................................................... 27
List of Figures

Figure 1. Geographical information [12] ................................................................. 13
Figure 2. Livestock Distribution  Figure 3. Cattle Distribution ................................22
Figure 4. Use case model for proposed system....................................................... 24
Figure 5. Sequence diagram for viewmap use case .............................................. 28
Figure 6. Sequence diagram for upload shape file use case .................................. 29
Figure 7. Activity diagram for the proposed system ............................................. 29
Figure 8. The architectural view of the proposed system ........................................ 39
Figure 9. Hardware-software mapping for the proposed system ......................... 39
Figure 10. Deployment diagram for the proposed system ...................................... 41
Figure 11. Login Screen ...................................................................................... 44
Figure 12. Users registration  screen ................................................................. 44
Figure 13. Default/home page of the system ...................................................... 45
Figure 14. The livestock page to search .............................................................. 45
Figure 15. Solution explorer of the system ......................................................... 47
Figure 16. The livestock page to view ............................................................... 52
## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACU</td>
<td>Adult Cattle Unit</td>
</tr>
<tr>
<td>AgSS</td>
<td>Agricultural Sample Survey</td>
</tr>
<tr>
<td>ASP .NET</td>
<td>Active Server Pages framework for Network</td>
</tr>
<tr>
<td>ATVET</td>
<td>Agricultural Technical and Vocational Education Training</td>
</tr>
<tr>
<td>CACC</td>
<td>Central Agricultural Census Commission</td>
</tr>
<tr>
<td>CSA</td>
<td>Central Statistical Authority</td>
</tr>
<tr>
<td>EASE</td>
<td>Ethiopian Agricultural Sample Enumeration</td>
</tr>
<tr>
<td>ESGPIP</td>
<td>Ethiopia Sheep and Goat Productivity Improvement Program</td>
</tr>
<tr>
<td>ESRI</td>
<td>Environmental Systems Research Institute</td>
</tr>
<tr>
<td>FAOSTAT</td>
<td>Food and Agriculture Organization Statistics Division</td>
</tr>
<tr>
<td>GDP</td>
<td>Gross Domestic Product</td>
</tr>
<tr>
<td>GIS</td>
<td>Geographic Information Systems</td>
</tr>
<tr>
<td>GLW</td>
<td>Gridded Livestock of the World</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>ILRI</td>
<td>International Livestock Research Institute</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Photographic Experts Group</td>
</tr>
<tr>
<td>LIVES</td>
<td>Livestock and Irrigation Value Chains for Ethiopian Smallholders</td>
</tr>
<tr>
<td>NDVI</td>
<td>Normalized Difference Vegetation Index</td>
</tr>
<tr>
<td>OSS</td>
<td>Open-Source-software</td>
</tr>
<tr>
<td>PNG</td>
<td>Portable Network Graphics</td>
</tr>
<tr>
<td>SDL</td>
<td>Spatial Distribution of Livestock</td>
</tr>
<tr>
<td>SDMS</td>
<td>Spatial Data Management System</td>
</tr>
<tr>
<td>SNNPR</td>
<td>Southern Nations, Nationalities, and Peoples Region</td>
</tr>
<tr>
<td>USAID</td>
<td>United States Agency for International Development</td>
</tr>
</tbody>
</table>
Abstract

Ethiopia has the largest livestock population in Africa. It is estimated at 105 million tropical livestock units, which includes 49.3 million heads of cattle, 47 million heads of sheep and goat, 8.3 million equines, 760 thousand camels and a poultry population of 38.13 million. The sector contributes to the livelihood of 60-70% of the Ethiopian population - this translates into approximately 44-52 million people whose subsidiary needs and economic activities rely on livestock production. Understanding the spatial distribution of livestock species is crucial in order to devise a feasible and geographically targeted livestock development policy.

The aim of this study is to identify the research gap on the spatial distribution of livestock in LIVES project and to fill the gap by reviewing different literatures and related works. Besides, we proposed and developed a web-based spatial data management application for the LIVES project that makes use of researchers and academicians to have a ground reference to make further research and improve the internal operations of the organization, raises community satisfaction, and simplifies spatial data management and exchange.

In order to conduct and justify this project we reviewed different journal articles and related works. In addition we have discussed with the GIS expert of LIVES project. In the mean time we have also collected livestock shape files from the LIVES project so as to use as an input to the proposed prototype web based application.

Through the course of this project we were able to observer and identify that the LIVES project handles and manages their spatial data using ArcGIS Desktop application that lacks to handle and render up-to-date and quality livestock information via the web.

We believed that having a web based application is necessary to permit timely livestock information dissemination, easy access, and render up-to-date livestock information to the
public. Furthermore, it improves the efficiency of spatial data management and provides quality of services through the web.

Key words: Livestock, ArcGIS, spatial, shape-file, prototype
CHAPTER ONE

1. Introduction

1.1. Overview

Ethiopia is a home for many livestock species and suitable for livestock production. Ethiopia is believed to have the largest livestock population in Africa CSA 2013; Solomon et al.2003; Tilahun and Schmidt 2012). An estimate indicates that the country is a home for about 54 million cattle, 25.5 million sheep and 24.06 million goats. From the total cattle population 98.95% are local breeds and the remaining are hybrid and exotic breeds. 99.8% of the sheep and nearly all goat population of the country are local breeds [7].

The livestock subsector has an enormous contribution to Ethiopia’s national economy and livelihoods of many Ethiopians, and still promising to rally round the economic development of the country. Livestock plays vital roles in generating income to farmers, creating job opportunities, ensuring food security, providing services, contributing to asset, social, cultural and environmental values, and sustain livelihoods. The subsector contributes about 16.5% of the national Gross Domestic Product (GDP) and 35.6% of the agricultural GDP (Metaferia et al. 2011). It also contributes 15% of export earnings and 30% of agricultural employment [7].

The livestock subsector currently support and sustain livelihoods for 80% of all rural population. The GDP of livestock related activities valued at birr 59 billion (Metaferia et al. 2011). Despite high livestock population and existing favorable environmental conditions, the current livestock output of the country is little. This is associated with a number of complex and inter-related factors such as inadequate On the other hand, the Ethiopia lowland pastoral areas which are affected by recurrent drought found to have spares livestock population (Tilahun and Schmidt 2012). However, no convincing study has been made so far to analyze the degree in which these factors hamper the production and distribution of livestock. According to Metaferia et al. (2011), cattle, sheep and goats are the three most important livestock species that have a considerable important to the GDP of the country [7].
Understanding the growth trend, spatial distribution and their relative access to market infrastructures of these livestock species is crucial in order to devise a feasible and geographically targeted livestock development policy. Due to the very important role the livestock subsector plays in the economy of the country, formulation of feasible and geographically targeted development plan regarding the subsector is indispensable. However, well-documented maps and descriptions of the geographic distribution of livestock populations are rare. This study attempts to fill this gap by producing maps based on Geographic Information Systems (GIS) of the geographical distribution of livestock populations [7].

The wave of Internet and Web technology has also reached to the Geographic Information Systems (GIS) research and development sector. Geographic Information system (GIS) is an emerging technology encompassing many disciplines namely geography, cartography, Geometry, remote sensing, surveying, GPS technology, statistics and other disciplines concerned with handling and analyzing spatially referenced data [7].

GIS is mainly comprised of data handling tools for storage, retrieval, management and analysis of spatial data as well as solving complex geographical problems. GIS can also be used for the generation of new information by the user-defined combination of several existing information. Because of the various distinguishing features of GIS, it is considered as an indispensable tool for conducting spatial searches and overlays and association of the spatial data with the non-spatial attribute data to eventually generate useful information [4].

Internet-GIS is a Geographic Information System (GIS) distributed across a computer network that integrate, disseminate, and communicate geographic information visually on the World Wide Web [21]. Internet-GIS technology provides dissemination, sharing, displaying and processing of spatial information over the Internet. Internet based geographical data services mainly involve sharing and management of spatial and non-spatial data. With Internet access and a web browser, larger and larger amounts of information can be made available quickly and conveniently to users. Internet geographic information system, which can directly run on the Internet, is developed to transmit the geographic information data on the Internet and for real-time analysis, to distribute processing and calculating approach on spatial information data [7].
Internet-GIS provide a much more dynamic tool than a static map display. It enables users to access spatial distribution of livestock where there is an internet connection. The popularity of GIS-based applications for the Web constantly increases. Indeed, Internet simplifies remote access to geographic data via a common user interface [20].

1.2. Statement of the problem

The lives project under ILRI presently uses ArcGIS desktop application to manage and handle livestock shape files and to perform related tasks. In view of the fact that this application is inefficient to manage the valuable livestock data, shape it into insights and then provide accurate, timely, and up-to-date information online so that researchers and other interested groups couldn’t get livestock distribution information online. This makes us interested to find alternative ways to manage and publish shape files through the web.

Now a day’s web based GIS applications are more and more important. Some of the GIS software’s are available free of cost. By utilizing open-source resources, web based GIS software is capable of rapid implementation with minimal input costs. Also most information in the world is available on the internet. World Wide Web is a useful tool for gathering and manipulating the useful data and information. One of the significant powers of GIS is the ability to publish spatial data over the internet for retrieval by multiple users. Sharing information over the internet allows dynamic data to be updated and disseminated. This promotes accessibility and timely information.

In this work we propose a web based spatial data management application for the LIVES project as web-based applications offer a range of business advantages over traditional desktop applications. Users access the application from any computer connected to the Internet using a standard browser, instead of using an application that has been installed on their local computer. It also enables users to disseminate and display the spatial distribution of livestock through the World Wide Web.
1.3. Objective of the project

1.3.1. General objective

Web Based application is a new technology that is used to display and manage spatial data on the Internet. It combines the advantages of both Internet and GIS. It offers the public a new means to access spatial information without owning expensive GIS software.

The main objective of our project is to identify the research gap on the spatial distribution of livestock in LIVES project and to fill the gap by reviewing different literatures and related works. Besides, we proposed and developed a prototype web-based spatial data management application for the LIVES project that makes use of researchers and academicians to have ground reference to make further research and to improve the internal operations of the organization, raises community satisfaction, and simplifies spatial data management and exchange.

1.3.2. Specific Objectives

The specific objectives of this project are to:

- Explore the existing Web Based-GIS tools and technologies.
- Investigate the current spatial data management situation in LIVES and to identify the problems.
- Propose new web-driven application for managing their spatial livestock data.
- Design a model for Web Based solution for the LIVES.
- Develop a prototype to demonstrate the recommended tools and models.

1.4. Data and Methodology

1.4.1. Source of Data

In order to run our project we collected livestock distribution data (shape file) from the LIVES project office. Besides some relevant information were collected through discussion with the GIS experts of LIVES project and by observing different documents.
1.4.2. Literature review

To get and understand the spatial distribution of livestock we were reviewed Journal articles, related research work, internet, and discussion with some GIS analyst of LIVES project.

1.4.3. Tools Selection

To validate our approach and to facilitate the implementation of the system we used the following technologies.

- Microsoft visual studio 2010 express as working environment
- C# Programming Language
- ASP.NET 4.5 version
- Easy GIS .NET Web Edition controls as reference(like EGIS.ShapeFileLib.dll and EGIS.Web.Controls.dll)

Visual Studio is Microsoft's flexible development tool used for building web based business applications. Visual Studio built-in languages include VB.NET (via Visual Basic .NET), C# (via Visual C#) and so on. Easy GIS .NET is a suite of .NET 2.0 GIS and mapping tools and controls to let developers easily incorporate GIS functionality into their applications.

1.5. Justification of the Project

This project aims at developing a Web-GIS application through analysis and design of a Geographic Information System (GIS) that will go a long way in contributing towards a more scientific and efficient management of spatial distribution of livestock for the LIVES project. There has been a lot of documentation of works on it but these efforts have not been integrated into a system that provides for efficiency and time series aspects for which statistical analyses and spatial distribution of livestock information that will help in decision-making and intervention response can be made.

The project will show why and how Web-based GIS can overcome the shortcomings in current GIS implementation and improve the utility of the web driven GIS technology in LIVES project.
This application:

- Provides a Uniform Repository for spatial distribution of livestock data that is scalable.
- Provides more efficient data entry and editing through rule and constraint application.
- Provides for Multi-user editing capabilities through Web technology with versioning in a centralized environment.
- Enables easy extraction/mining of data via the intuitive Web Interface.

1.6. Scope and Limitation

1.6.1. Scope

The scope of this project is to develop a workable prototype web based spatial data management application for LIVES project in Ethiopia that provides livestock distribution information on time.

For demonstration purpose we selected the Ethiopia livestock distribution shape file. The prototype will be developed to show the distribution of livestock such as cattle, goat, and sheep at “wereda” level in a given region and zone.

1.6.2. Limitation

In order to perform this project we tried to review the implementation of a web based technologies for the spatial distribution of livestock, however some journal articles need subscription.

Due to time constraint and lack of skill on ArcGIS we couldn’t develop a full-fledged system that can support complex spatial data management activities such as advanced geo-processing and real-time data processing and monitoring.

1.7. Significance of the project

Web Based-GIS helps to maintain, manage, plan and analyze geographically referenced data on the web. In addition this project can serve as a ground reference and initiative to develop Web Based-GIS in other governmental and non-governmental offices. It also
enables the researchers at the LIVES project to have spatial data access and make effective decisions.

Some of the domains of applications that can benefit from the project result are the Ethiopia Sheep and Goat Productivity Improvement Program (ESGPIP) is a USAID-funded project operating with a goal to sustainably increase sheep and goat productivity in Ethiopia to consequently enhance economic and food securities.

This project believed to be helpful and important for those who have interest to do further research and the investigation may provide some important information about the use and application of system in the livestock sector. It is also important to initiate other interested researchers to undertake a better and detailed study in the field. From end-users perspective the proposed system used to enhance the accessibility of spatial distribution of livestock through web based GIS applications.

1.8. Document organization

The document consists of six chapters including this chapter. The learning theories along with related works to this project are reviewed in chapter two. Chapter three discusses about the requirement determination and requirement analysis. This chapter draws out the business model and specification of the problem domain. The system design is detailed in chapter four. The implementation is discussed in chapter five. The last chapter is all about conclusions and recommendations.
CHAPTER TWO

2. Literature review

The intention of this chapter is to provide the readers with a general overview of livestock distribution, application of web-based GIS in the livestock sector, and spatial data management technologies of livestock.

Therefore, it is subdivided into Mapping the Global Distribution of Livestock, Livestock in Ethiopia, Overview of GIS, Application of GIS in the livestock sector, Web-Based GIS, Web Based APIs, Web Based GIS Mapping Technologies, Spatial Data Management System (SDMS), and Mapping Livestock in Ethiopia.

2.1. Mapping the Global Distribution of Livestock

Livestock contributes directly to the livelihoods and food security of almost a billion people and affects the diet and health of many more. With estimated standing populations of 1.43 billion cattle, 1.87 billion sheep and goats, 0.98 billion pigs, and 19.60 billion chickens, reliable and accessible information on the distribution and abundance of livestock is needed for many reasons. These include analyses of the social and economic aspects of the livestock sector; the environmental impacts of livestock such as the production and management of waste, greenhouse gas emissions and livestock-related land-use change; and large-scale public health and epidemiological investigations. The Gridded Livestock of the World (GLW) database, produced in 2007, provided modeled livestock densities of the world, adjusted to match official (FAOSTAT) national estimates for the reference year 2005, at a spatial resolution of 3 minutes of arc (about 565 km at the equator). Recent methodological improvements have significantly enhanced these distributions: more up-to-date and detailed sub-national livestock statistics have been collected; a new, higher resolution set of predictor variables is used; and the analytical procedure has been revised and extended to include a more systematic assessment of model accuracy and the representation of uncertainties associated with the predictions. The research paper describes the current approach in detail and presents new global distribution maps at 1 km resolution for cattle, pigs and chickens, and a partial distribution map for ducks. These digital layers are made
publically available via the Livestock Geo-Wiki (http://www.livestock.geo-wiki.org), as will be the maps of other livestock types as they are produced [16].

2.2. Livestock in Ethiopia

Ethiopia has a land area of approximately 1.13 million km² and an estimated human population of 81 million growing at a rate of 2.7% per annum. About 85% of the population lives in rural areas and practices sedentary agriculture and livestock production. Agriculture accounts for 45% of the GDP and 85% of export earnings, and the sector employs about 85% of the population. Ethiopia’s highland comprises 40% of the country’s land area, holds 88% of the human population and 74% of the tropical livestock units. The main activity is a mixed crop-livestock farming system dominated by crop production and accounts for more than 90% of the country’s economic activity. Livestock are vital to crop cultivation as the traditional ox-plow method of farming is a defining feature in the highlands. In contrast, the lowland comprises 60% of the country’s land area and 12.2% of the total human population, and approximately 25% of the livestock population. The area is dominated by a pastoralist population whose economy is entirely dependent on livestock husbandry, which provides a basis of the food production and economic system [17].

Ethiopia has the largest livestock population in Africa. It is estimated at 105 million tropical livestock units, which includes 49.3 million heads of cattle, 47 million heads of sheep and goat, 8.3 million equines, 760 thousand camels and a poultry population of 38.13 million. 16 Cattle play the most important role in the farming economy followed by sheep and goats. Poultry farming is widely practiced in Ethiopia and small farmers use them for consumption purposes and a source of cash income [17].

2.2.1. Geographic Distribution

The livestock population densities were computed as number of animals per km² at district level. The Study shows that cattle, sheep and goat population are unevenly distributed across space. The Ethiopian highlands are found to be highly populated. SNNPR, Amhara, Harari and Oromia are the highly populated region with respect to cattle population. High populations of sheep per km² were found in Amhara, SNNPR and Dire Dawa. Dire Dawa, Harari and Tigray regions have the highest goat population.
per km2. On the other hand, the pastoral areas of Somali and Borena are found to have very low cattle density and relatively fair number of sheep and goat [15].

2.2.2. Market access

The proximity of cattle, sheep and goat population to a given market infrastructure is calculated using Zonal statistics. As figures on the study report shows the highest numbers of livestock (46% of cattle, 43% of sheep and 40% goats) population were found to be located in zone one (within 10 km radius from the main towns and all-weather roads). These areas mainly cover the central highlands of Tigray, Amhara, Oromia and SNNPR regional states. This zone covers only 11% of the country’s landmass. On the other hand, 75% of the country’s land mass is located in zone three (greater than 30 km radius from market infrastructures). These areas are found to be not accessible to market infrastructure and support only 15, 20, and 21% of the country’s cattle, sheep and goat population respectively [15].

In livestock and livestock products marketing road and market infrastructure is important to farmers, traders and consumers seeking to sell and/or buy livestock or livestock products. Underdevelopment and lack of market-oriented production, lack of adequate information on livestock resources and market information, inadequate marketing infrastructures, presence of trans-boundary animal diseases, and illegal trades are the major factors affecting livestock marketing in Ethiopia [15].

Study was made to assess livestock population growth trend, distribution and their access to market. Regression analysis was used to assess the cattle and shoat population growth trend and Geographic Information Systems (GIS) techniques were used to determine the spatial distribution of cattle and shoats, and their relative access to market. The data sets used are agricultural census (2001/02) and annual CSA agricultural sample survey (1995/96 to 2012/13). In the past eighteen years, the livestock population namely cattle, sheep and goat grew from 54.5 million to over 103.5 million with average annual increment of 3.4 million. The current average national cattle, sheep and goat population per km2 are estimated to be 71, 33 and 29 respectively (excluding Addis Ababa, Afar and Somali regions). From the total livestock population the country owns about 46% cattle, 43% sheep and 40% goats are reared within 10 km radius from major livestock market centres and all-weather roads. On the other hand, three fourth of the country’s land mass which comprises 15% of the cattle, 20% of the sheep and 21% of goat
population is not accessible to market (greater than 30 km from major livestock market centres). It is found that the central highland regions account for the largest share of livestock population and also more accessible to market. Defining the spatial and temporal variations of livestock population is crucial in order to develop a sound and geographically targeted livestock development policy [15].

2.2.3. Economic Benefit

The livestock subsector has an enormous contribution to Ethiopia’s national economy and livelihoods of many Ethiopians. The subsector contributes about 16.5% of the national Gross Domestic Product (GDP) and 35.6% of the agricultural GDP. It also contributes 15% of export earnings and 30% of agricultural employment. The livestock subsector currently support and sustain livelihoods for 80% of all rural population. The GDP of livestock related activities valued at 59 billion birr. Ethiopian livestock population trends, distribution and marketing vary considerably across space and time due to a variety of reasons [15].

2.3. Overview of GIS

A Geographical Information System (GIS) is a computer system for processing, storing, checking, integrating, manipulating, analyzing and displaying data related to positions on the surface of the earth. It is then presented cartographically, graphically or as a report. GIS users add to the specialized software, procedures, operating staff and also the spatial data that feed the system. Most of the information about our world contains a location reference that places this information at some point on the earth. Without this location reference, there is no geographical information [12].

To locate geographical information on a map, you use either maps that use a coordinate system to allow locations to be read, or you use shapes (polygons) of the geographic information, where shapes of the features and themes are drawn onto the map. Applications of GIS could reveal links between different sources of information, when it is presented on a map and can find out relationships between features that are not readily apparent in spreadsheets or statistical packages. It creates often new information from existing data, resources very useful for decision making (Veterinary Services), which can lead to better management of disease control programmes and emergency situations [12].
2.3.1. How does a GIS work?

GIS is usually a computer-based system and it relies on the development of dynamically-linked databases having common geographical components. Maps can be represented as several different layers where each layer holds data about a particular kind of feature (characteristic, variable). Each characteristic is linked to a position on the graphical image on a map and a record in an attribute table. The strength of a GIS is related to its ability to relate different information into a spatial context and to reach a conclusion about this spatial relationship that cannot be seen if the information is looked at independently [12].

Therefore it is used to determine where or what an individual feature is and to help in finding patterns by looking at the distribution of overlapped features on the map instead of just a set of individual features not linked together [12].

2.3.2. Applications of GIS

GIS can relate otherwise disparate information, on the basis of common geography, revealing hidden patterns, relationships, and trends that are not readily apparent in spreadsheets or statistical packages, often creating new information from existing data resources. This information is very important as a management tool and produces valuable information needed for better decision making. GIS applications are very wide and are used in all human activities. It is used for marketing studies, telecommunications, location of restaurants, museums and hospitals; in tracking trucks traffic; in establishing maps of animal population density by species or maps of vegetation coverage change; in locating forests, rivers, mountains; in determining soil compositions, etc. [12].

2.3.3. How can a GIS use the information in a map?

Information relating to what is where, and sometimes when, on the earth, represents simple information that can be located [12].

The most known source of geographic information is maps, in which information about the world around us is plotted within a structured framework (a coordinate system) that allows us to find its location [12].

Maps use a coordinate system and shapes (polygons) to allow locations to be read (see Figure 1).
If the data to be used are not already in digital form, i.e. in a format that the computer can recognize, various techniques can capture the information [12].

Maps can be digitized by hand-tracing with a computer mouse on the screen or on a digitizing tablet to collect the coordinates of features [12].

There are also raster data which consist of a matrix of pixels organized into a grid. Each grid contains information on location co-ordinates and an attribute value. Areas containing the same attribute value are recognized as such. However, raster structures cannot identify the boundaries of such areas as polygons. Satellite images and aerial photographs are some examples of raster data [12].

Geo-coordinates from Global Positioning System (GPS) receivers can be uploaded into a GIS. The GPS uses the Global Navigation Satellite System (GNSS) that utilizes a constellation of at least 24 medium earth orbit satellites that transmit precise microwave signals. The latter enables the GPS receiver to determine its location, speed/direction and time [12].
2.4. Application of GIS in the livestock sector

The application of the GIS in the veterinary field has developed over the last decade. Specialized software has become more affordable and more user-friendly; in addition to the development of open source tools for mapping. GIS is an excellent tool for spatial data presentation, inclusion of additional layers (e.g. environmental factors) for better analysis. It is an excellent tool for decision making not yet implemented or used by many Veterinary Services around the world [12].

GIS helps in understanding and explaining disease dynamics and spreading patterns. It helps increasing the speed of response in case of an emergency linked with the introduction of a disease. Overlapping maps of location of outbreaks, with the map of location of farms, abattoirs or roads, for example, can help better drawing the perimeter of security, surveillance zones as well as available facilities to implement the decided control measures. The addition of other factors, such as the Normalized Difference Vegetation Index (NDVI), satellite images and vectors distribution in case of vector-borne diseases, can correlate disease trends and be used as an early warning tool, for example, or for the prediction of the evolution of a newly introduced disease [12].

Recent years have seen an enormous increase in the number of web-based applications that use techniques derived from geographic information systems (GIS). Thus, for example, Haklay et al. (2008) describe how, by mid-2005, Multi-map had attracted 7.3 million visitors in UK and Map Quest was used by 47 million in US; and how, by mid-2007, there were an estimated 50,000 websites that took feeds from the Google Maps (http://maps.google.com) site. Web-based GIS applications can be as simple as presenting a simple map of the world to front-ending complex spatial analyses of spatial distributions and processes. GIS-enabled web applications can utilize any of a number of technologies and database platforms, the choice of which can impact on the performance of the web application [19].

MS-Access is not the software of choice for a GIS application built around millions of records because the software has not been developed to handle very large amounts of geographically referenced data efficiently. An important consideration in software selection is speed of response as users may expect the result of GIS analysis to be produced in real time. These requirements make the choice of mapping, database, and development technologies and standards very important [19].
2.5. Web-Based GIS

Web-Based GIS is becoming more and more important at time passes. The World Wide Web (WWW) is a useful tool for gathering and manipulation. Most information that is available in the world is now available over the internet. There is no need to buy any expensive software packages for manipulating the data needed for GIS [23].

Web-Based GIS, software applications are available with the advent of Java based programming. Some required no special software but some of programs required to buy and others are required to plug-ins to use capabilities of the web browsers. Because of these advancements, people are getting information easily [23].

Another useful fact of using web-based GIS is that the people giving the information are completely in charge of the amount of information made available to the public. If there were privacy issues surrounding certain bit of information, don’t make it accessible to others [23].

2.6. Web Based APIs

Application Programming Interfaces (APIs) are software with a clearly defined set of functions that expose the functionality of an underlying system so that external developers can make use of that software’s functionality. In the context of web-based GIS these API can be seen as web-services that specialize in GIS functionality and allow developers to “plug-in” to them and use their specialized mapping features. These web-based API’s provide basic GIS functionality including mapping and querying tools. More complex features are now beginning to emerge in these systems [24].

Web based API have further extended the accessibility of GIS services. Internet-GIS systems can now be created without developers having to have an extensive knowledge of how to construct GIS maps. Instead the developer can focus on the customization of the application. There are a wide variety of commercial web-based API’s that provide features including map creation, editing, display, spatial analysis, and web site design tools useful in developing and implementing an online GIS [24].
2.7. Web-Based GIS Mapping Technologies

An important part of every web-based GIS application is its mapping or visualization technology, which makes it possible to show data in the form of maps. Visualization of data as maps has become increasingly popular, with hundreds of websites presenting geographic data. The popularity of web based mapping applications arises in large part through the wide dissemination of software that makes it easy for users and developers to publish map data. Improvements in usability through improved user interfaces also account for the increased popularity of visualization techniques (Aoidh et al. 2008). In similar vein, Cashera et al. (2008) have demonstrated that growing interest in visualization and analysis of social networks has led to the development of several methods of structural analysis in order to analyze individual and group behavior. This visualization is not limited to the display of raw data in maps but is increasingly widely applied in the representation of large spatial databases [19].

One such example is the World names website (http://www.publicprofiler.org/worldnames), which accesses a database of 1 billion geo-referenced names around 26 different countries of the world. This website generates flash maps to show the concentration of surnames around different countries of the world. And this explains the extent of technology which is used nowadays in web-based GIS applications [19].

There are three ways in which visualization functionality is implemented; static map renderers, slippy maps, and flash mapping [19].

2.7.1. Static Map Renders

Static map renderers have been heavily used in the creation of maps. They render maps in the form of images. They work by rendering the map to an image (such as a PNG image), and then send the image back to the user who can view it using a web or desktop application. Static map renderers do not allow any user interaction with the maps because they are just static images, so they do not allow users to pan or zoom in, out or around the map. On the one hand, the resulting map is very simple and easy to interpret, because there is nothing required on the client side, but on the other hand the maps
produced are very inflexible. Because, a user cannot interact with the maps they are not suitable for rich Internet applications, or for the new Web 2.0 applications [19].

2.7.2. Slippy (Tile-Based) Maps

One of the important and widely used geo-visualization techniques is tile-based or slippy maps. This technique works by dividing the map into a discrete number of zoom levels, so that each zoom level has an identical number of tiles. Instead of accessing the whole map at once, a tile-based client builds the map by accessing individual tiles and then assembling them to form a map. Google Maps, Microsoft Virtual Earth, Yahoo Maps, Mapnik, Open Layers, and ArcGIS as well as derivative products such as MapTube are based on tile based or slippy maps [19].

2.7.3. Flash Mapping

Flash maps are based on vector images. Busselle (2008) describes how a raster image is a rectangular array of elements called pixels, and a vector image is a collection of connected lines and curves that can be used to delineate objects. Raster images are resolution dependent, so their shape distorts if the image is made bigger than its actual resolution. Flash maps are based on vector images. Busselle (2008) describes how a raster image is a rectangular array of elements called pixels, and a vector image is a collection of connected lines and curves that can be used to delineate objects. Raster images are resolution dependent, so their shape distorts if the image is made bigger than its actual resolution [19].

2.8. Spatial Data Management System (SDMS)

Spatial data is essential in a wide range of application domains today. Spatial data management is a technique for organizing and retrieving information by positioning it in a graphical data space (GDS). This graphical data space is viewed through a color raster-scan display which enables users to traverse the GDS surface or zoom into the image to obtain greater detail. In contrast to conventional database management systems, in which users access data by asking questions in a formal query language, a spatial data management system (SDMS) presents the information graphically in a form that seems
to encourage browsing and to require less prior knowledge of the contents and organization of the database [13].

2.9. Mapping Livestock in Ethiopia

The latest digitized livestock and human population data from the Central Statistics Authority (CSA) and the International Livestock Research Institute (ILRI) were used to produce maps on geographic distribution of livestock and human population densities. The CSA conducted a census in 2001/02 covering the highlands and sedentary areas in the highland–lowland interface, and a separate limited sample survey in some lowland pastoral areas. The density of livestock per 1000 human population indicates the extent of association between human and livestock population growth, and the nature of evolution of the production systems across the country. The positive relationship between human and livestock densities reflects similar developments elsewhere in the world [22].

The CSA implemented an aerial survey in 2003 to estimate livestock production in seven zones of Somali region (Somali region has a total of nine zones) that had previously not been surveyed due to inaccessibility. The 2002/03 aerial data reported similar cattle population data as the figures reported for all nine zones of Somali region from the Agricultural Census (Ethiopian Agricultural Sample Enumeration, EASE) data from 2001/02 (670 and 643 thousand cattle respectively). Aerial data on sheep and goat production for the same seven zones of Somali region (11.94 million), however, is about 8 times greater than reported figures from the Agricultural Census for Somali Region in 2001/02 (1.45 million) [22].
CHAPTER THREE

3. System Analysis

3.1 Overview

This is a requirement for spatial distribution of livestock is to assess the requirement and as a result come up with a right solution. This requirement provides the bases for the consideration of functional and non-functional requirements of the spatial data management for livestock distribution system by understanding the business processes in order to allow the translation of core processes for automation. The software functional requirements have been captured. Most importantly, the functional requirements of the system are presented in the form of use cases identifying the functional boundaries of the system and thus it also determines the software requirement scope. The use-cases are developed based on goals needed to be achieved by actors who represent group of users interacting with the system.

To deal with developing systems accurate assessment of requirements are captured and presented in a manner that satisfies all stakeholders. This spatial requirement document is prepared for the development of data management for livestock distribution web based application. It is strongly believed that, the extensive use of this requirement is expected to play an important role throughout the development and deployment lifecycle of the project.

The software requirement specification defines what the software must do to add value for its stakeholders. These functional requirements define the capabilities of the software product; what the software must do, to add value for its stakeholders. The nonfunctional requirements define the characteristics, properties, or qualities that the software product must possess and what limitations there will be due to absence of that.

Thus, the business requirements define the business problems to be solved or the business opportunities to be addressed by the software product. In general, the business requirements define why the software product is being developed. Business requirements are typically stated in terms of the objectives of stakeholder or organization requesting the development of the software.
Whereas, from the user’s perspective, the user requirements look at the functionality of the anticipated software product to that need to be delivered. User requirements define what the software has to do in order for the users to accomplish their objectives. Multiple user level requirements may be needed in order to fulfill a single business requirement.

Finally, the product’s functional requirements that define the software functionality must be built into the product to enable users to accomplish their tasks, thereby satisfying the business requirements. Multiple functional level requirements may be needed to fulfill user requirements.

3.2. Objective

The main objective of is to assess the requirement of spatial data management for livestock distribution system with view to come up with an automated solution to serve its internal and external clients and customers and develop Software Requirement Specification (SRS) to meet the same.

3.3. Scope

The scope of the project entails mainly to undertake detailed requirement study and prepare a complete document covering Software Requirement for the System. The scope specifically covers the following two major areas:

- Develop Functional Requirements in the form of use cases
- Develop and specify Non-Functional Requirements

3.4. The existing system

The first step in software development process is collecting and identifying the needs of the organization to which the software is developed. In this project the need assessment was done through interview and review of related documents.

LIVES contributes to enhanced income and gender equitable wealth creation for smallholders and other value chain actors in Ethiopia through increased and sustained market off-take of high value livestock and irrigated crop commodities.
Over the last 20 years, researchers at the International Livestock Research Institute (ILRI) have collected and generated an extensive range of spatial data layers. A number of these layers are directly related to livestock, such as distribution, health and production. Other layers, however, cover more general topics such as human population density, climate and infrastructure. Some of the datasets cover only specific project, while others are county-wide, regional, continental or even global.

This collection of data can be of great value for students, researchers, farmers, policy makers or anybody else interested in spatial analysis. There is therefore a need to share it with both for their collaborators and the general public. However currently the ILRI Addis office doesn’t have a GIS department that can manage and handle the online database and it seems to be stops functioning and this static data based that show the General livestock distribution of this country and it seems to be obsolete and at the same time the system doesn’t have dynamic nature and doesn’t show the latest figures or detailed spatial distribution of the livestock. Figure 2 and 3 shows the screen shot of the old system.

Currently the LIVES project which is operated under ILRI has spatial data at “wereda” level showing the livestock distributing of the country. However they couldn’t make this data available to the public even to the researchers locally through their intranet. There are a number of factors like the high coast and licensing issue related to ArcGIS Server, lack of adaption and commitment and lack of skill to use the latest open source GIS application. Currently there are a lot free and open source desktop and Web Based GIS applications available on the internet.
3.5. Proposed system

The new system is designed to solve the problems of the existing system. The system is expected to be web-based GIS application which uses mapping or visualization technology, which makes it possible to show data in the form of maps and it should satisfy the following functional requirements and non-functional requirements.

3.5.1. Functional Requirements

The proposed system should provide the following functionalities to the users of the system.

- The system shall enable authorized users to view and display spatial distribution of livestock.
- The system shall browse and find livestock distribution information geographically.
- The system shall store a rich collection of spatial data in a centralized location.
• The system should provide common and up-to-date information to the general public using websites.
• The web application shall be able to produce and display information about livestock distribution at "wereda", zone and regional level.
• The system should enable system administrator to upload shape files to the system.

Notice that each requirement is directly related to what we expect the web application to do. They represent some of the core functions.

3.5.1.1. Stakeholders

The following lists are the stakeholders of the system:

  • System Administrator
  • Clients/End-users

**System Administrator** is a person responsible for controlling and administering the system.

**Clients/End Users** who are directly interact with the system and need fast and quality service from the system.

3.5.1.2. Actors

The following list shows the roles the stakeholders play in interacting with the system:

  • System Administrator
  • Clients/End-users

The use case diagram that depicts requirements identified are shown in figure 4.
3.5.1.3. System Model

Use case

A use case describes the functionality of the system from the user's point of view. After analyzing the functional requirements of the new system, the following use cases and actors are identified. Figure 4 depicts the use case model.

![Use case model for proposed system](image)

**Figure 4. Use case model for proposed system**

The descriptions briefly explain how the functionalities precede using natural language in a step-wise manner. It is for all use cases which is stated in figure 4.
Use case: login

**User Case description:** User is able to enter username/password combination and login to the system successfully.

<table>
<thead>
<tr>
<th>Use-case name</th>
<th>login</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participating Actors:</td>
<td>Anonymous user, Registered user and Administrator.</td>
</tr>
<tr>
<td>Precondition(s):</td>
<td>User must exist in the database.</td>
</tr>
<tr>
<td>Post condition(s):</td>
<td>User can login successfully</td>
</tr>
<tr>
<td>Flow of events</td>
<td>1. User launches the login screen.</td>
</tr>
<tr>
<td></td>
<td>2. User enters a combination of username and password.</td>
</tr>
<tr>
<td></td>
<td>3. System validates the combination and logs in the user successfully.</td>
</tr>
<tr>
<td>Exit condition</td>
<td>User closes /logs out from the system.</td>
</tr>
</tbody>
</table>

**Table 1. Login use case description**

Use case: User Registration

**User Case description:** A new user needs to first register into the system before performing any transaction.

<table>
<thead>
<tr>
<th>Use-case name</th>
<th>Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participating Actors:</td>
<td>Anonymous user</td>
</tr>
<tr>
<td>Precondition(s):</td>
<td>Unregistered user to the system.</td>
</tr>
<tr>
<td>Post condition(s):</td>
<td>Users can Registered successfully</td>
</tr>
<tr>
<td>Flow of events</td>
<td>1. The user clicks the register button on the page.</td>
</tr>
<tr>
<td></td>
<td>2. The system displays the register page.</td>
</tr>
<tr>
<td></td>
<td>3. The user enters all the required information.</td>
</tr>
<tr>
<td></td>
<td>4. The user clicks the Register button</td>
</tr>
<tr>
<td></td>
<td>5. The system check that all the required information is entered, if yes the system update the user account and displays Ok</td>
</tr>
<tr>
<td>Exit condition</td>
<td>The user views the page and closes the browser</td>
</tr>
</tbody>
</table>

**Table 2. User registration use case description**
Use case: ViewMap

User Case Description: A new user needs to first register in to the system to view and display the map

<table>
<thead>
<tr>
<th>Use-case name</th>
<th>ViewMap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participating Actors:</td>
<td>Registered user, Administrator</td>
</tr>
<tr>
<td>Precondition(s):</td>
<td>The user should already login to the system</td>
</tr>
<tr>
<td>Post condition(s):</td>
<td>The user has viewed the resource</td>
</tr>
</tbody>
</table>
| Flow of events | 1. The user clicks the Livestock page  
2. The user selects the dropdown list (cattle, goat, sheep)  
3. The user clicks the Generate map button  
4. The user hover his mouth over the desired location to see the output |
| Exit condition | The user closes /logs out from the system |

Table 3. ViewMap use case description

Use case: Upload shapefile

User Case Description: Administrator initiates the “file upload” button to move new or latest shape files to the web server

<table>
<thead>
<tr>
<th>Use-case name</th>
<th>Uploadshapefile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participating Actors:</td>
<td>Administrator</td>
</tr>
<tr>
<td>Precondition(s):</td>
<td>The Administrator should already login to the system</td>
</tr>
<tr>
<td>Post condition(s):</td>
<td>The Administrator has moved or uploaded the files</td>
</tr>
</tbody>
</table>
| Flow of events | 1. The Administrator login to the page  
2. The administrator clicks browse button  
3. The administrator selects files and click the upload button  
4. The system shows the “File successfully uploaded” message |
| Exit condition | The Administrator logout from the system |

Table 4. Uploadshapefile use case description
Use case: AddUser

User Case Description: Administrator adds user to the system by providing identification information

<table>
<thead>
<tr>
<th>Use-case name</th>
<th>AddUser</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participating Actors:</td>
<td>Administrator</td>
</tr>
<tr>
<td>Precondition(s):</td>
<td>The user should be unregistered user</td>
</tr>
<tr>
<td>Post condition(s):</td>
<td>A new user account is created and ready to login</td>
</tr>
</tbody>
</table>
| Flow of events | 1. The Administrator login to the page  
2. The administrator fills the required information  
3. The administrator selects the Create New Account option  
4. The system shows the “New Account successfully created” message |
| Exit condition | The Administrator logout from the system |

Table 5. Adduser use case description

Use case: DeleteUser

User Case Description: This use case describes how the Administrator removes the User from the system.

<table>
<thead>
<tr>
<th>Use-case name</th>
<th>DeleteUser</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participating Actors:</td>
<td>Administrator</td>
</tr>
<tr>
<td>Precondition(s):</td>
<td>The Administrator is logged in and wants to remove the existing user from the system.</td>
</tr>
<tr>
<td>Post condition(s):</td>
<td>The user is removed from the system</td>
</tr>
</tbody>
</table>
| Flow of events | 1. The system requires confirmation that the user is to be deleted.  
2. The Administrator confirms that the user is to be removed.  
3. The system presents the administrator with a success message, confirming that the user was removed. |
| Exit condition | The Administrator logout from the system |

Table 6. DeleteUser use case description
3.5.1.4. Dynamic Model

**Sequence Diagram**

The sequence diagram represents the sequence of actions that will perform in the system. Below are some of the sequence diagrams for a few use cases (viewmap, uploadshapefile)

**Sequence diagram for viewmap use case.**

The sequence of actions for the view map use case is that the client will type the URL of the LIVES site on the client Browser. The LIVES displays the home page.

The client will select the dropdown list and clicks **Generate map** button.

The user can zoom in or Zoom out the map. The client should hover the mouth over any region or zone to see the no of livestock on the given “Wereda”. The diagrammatic representation is shown in figure 5.

![Sequence Diagram for Viewmap Use Case](image)

**Figure 5. Sequence diagram for viewmap use case**
Sequence diagram for uploadshapefil use case

The sequence of actions for the uploadshapefile use-case is that the administrator will browse and upload the shape file to the web server. The diagrammatic representation is shown in figure 6.

![Sequence diagram](image)

**Figure 6. Sequence diagram for upload shape file use case**

### 3.5.1.5. Activity Diagram

The activity diagram below on figure 7 depicts the activities performed on the system.

![Activity diagram](image)

**Figure 7. Activity diagram for the proposed system**
3.5.2 Nonfunctional requirements

This section describes other features, characteristics, and constraints that should be satisfied in addition to the above system functional requirements of the system. The non-functional requirements are listed in this section to identify the major operations of the system. There are various types of non-functional requirements like reliability, availability, security, performance, usability, scalability, maintainability, extensibility, feasibility, economical, operational, technical which are listed in this section and they are often called qualities.

The following lists of categories of quality attributes that the system is required to have in order to execute its functions, reliably, cost-effectively and evolve overtime to respond to the challenges so as to harness opportunities, time, inevitably and creates:

- Reliability
- Availability
- Security
- Performance
- Usability
- Scalability
- Maintainability

3.5.2.1. Reliability

- The system should be available for service when requested by end-users and failure rate low to deliver the service as expected by end-users
- The system has to be consistently performs according to its specification.
- The system should be the ability to deliver service that can justifiably be trusted by users.

3.5.2.2. Availability

- The system shall be accessible anytime, anywhere, where there is an Internet connection, putting the user in charge of where and when they access the application.
3.5.2.3. Security

- The application shall identify all of its client applications before allowing them to use its capabilities.
- Being part of the services the system is required to be made accessible to all users based on their role.
- The access permissions for system data may only be changed by the system’s data administrator.
- The security should be implemented using user name and password which should be entered by the user when he or she logs into the system.
- All information on the application should be secured so that only system administrator has access to upload and edit shape files.
- All system data must be backed up every 24 hours and the backup copies stored in a secure location which is not in the same building as the system.
- The system should no longer operate if security attacks have become obvious.

3.5.2.4. Performance

- The system shall be able to responds to huge number of user requests simultaneously.
- This system should be the site providing information to researchers and scientists.
- Users can get updated and consistent information through this application.

3.5.2.5. Usability

- The system administrator and end-user should be able to use the system with little training.
- The web application shall be easy to use by all employees of LIVES project as well as external users.

3.5.2.6. Scalability

- The system to be designed shall undergo a significant evolution during the first few years of operation.
3.5.2.7. Maintainability

- Installation of a new version shall leave all data contents and all personal settings unchanged.
- The product shall provide facilities for tracing any data file to places where it is used.
CHAPTER FOUR

4. Design of the System

4.1. Choosing an Appropriate Mapping Technology

The choice of an appropriate mapping technology is based on the requirement of the time it takes for the user to submit a query and to receive the results. Static map renderers are an old style mapping technology, which produces map in the form of an image before display of the image by the web application. Static map renderers are nonetheless useful in situations where the number of geographically referenced areas is moderate. This can be described as any number of geographically referenced areas less than 1000. Because each user request generates a new image, the application becomes slow if there are a great number of geographically referenced areas to be displayed in the image produced. However, static map renderers are fast for small number of geographically referenced areas, although they are not a flexible solution for representing maps because of the non-interactive nature of the final maps produced [19].

Today, the tile based or slippy mapping technique is the most widely used mapping technique. Slippy maps are better than static maps because tiles are served to users based on requests rather than serving the whole map at once. Slippy maps are dynamic and faster to load because the tiles that a user wants to see only load up and it is not necessary to load all the tiles at once. This saves computational processing power and memory resources, both on the client and the server side. In addition, it provides a powerful architecture for the creation of rich Internet GIS web applications. Rich Internet GIS web applications are defined as defined as web-based applications having characteristics of desktop applications and are delivered by way of a standard web browser. [19]

4.1.1. Web Application Development Standards

Web applications development standards are the third important part of any web-based application. They provide the standards concerned with the actual development of the web application and entail different techniques and technologies that, once used efficiently, can enhance the performance of a web application. They can be divided into design standards and development standards [19].
4.1.1.1. Design Standards

Design standards are concerned with the pre development structure of a web application, and are usually decided prior to commencing development of the web application.

Web Wireframing

Web wireframing is a unique concept in web site design and development. Miller (2008) has described web wireframes as ‘simple line drawing’ to show the placement of elements in the webpage. Web wireframing is used to provide a visual guide to the website and demonstrates the links between different pages in the website. Web wireframing helps in creating a final design of the Wireframing has been used for a number of years in web site design and development and can help in enhancing the usability of a GIS web application. Because developers can see the web application design in prototype form and give feedback, it is possible to develop an application which marries user expectations with requirements. Several tools are available to design web wireframes, of which Adobe Illustrator (http://www.adobe.com/products/illustrator), Microsoft Visio (http://office.microsoft.com/visio), and ProtoShare (http://www.protoshare.com) are some of the popular [19].

Design Patterns

Chambers et al. (2000) have described design patterns as a recent software engineering innovation that explains common problems and solutions in object-oriented development. Design patterns have become very popular because they can ease the development of software engineering projects. They provide general solutions to commonly occurring problems by using a template to clarify the relationship between different entities of a software or web application. Abstract factory, Factory method, Builder, Lazy initialization, Singleton, and Prototype are examples of software engineering design patterns. They can be used in web-based GIS applications to standardize the application according to the problem domain and hence tuning the web applications to work better [19].
4.1.1.2. Development Standards

Development standards describe the choice of an appropriate development technology for the web application. For example, .NET is a development technology used to develop software or web applications and Oracle & SQL Server are database servers which can be used as the back end of web applications.

4.1.2. Choosing the Appropriate Development Technology

A range of development technologies is available for different operating platforms in order to develop GIS web-based applications. Important development technologies include ASP (Active Server Pages), JSP (Java Server Pages), ASP.NET (Active Server Page dot Net), and PHP (Hypertext Preprocessor), the choice between which depending upon the developer’s operating environment. ASP (Active Server Pages) is a Microsoft’s technology used to develop web applications for deployment on Windows. Users of an ASP Web application can access it from any platform. ASP.NET (Active Server Pages Dot Net) is the enhanced extension of Microsoft’s ASP technology which offers object oriented support. Web applications developed in ASP.NET can also only be deployed on Windows platforms, but can be accessed from any platform. JSP (Java Server Pages) is a Sun Microsystems technology used to develop web applications. They can be deployed or accessed on any operating platform because java technologies are platform independent. The same is true of PHP (Hypertext Preprocessor), which is an open source technology used to develop web applications. So in this context we preferred to use ASP.net for developing our application [19].

4.1.3. Use of Object Oriented Development Standards

The object oriented programming paradigm uses objects and their interactions to design and develop software or web applications. The use of object orientation in web applications has increased in the recent years since applications are well structured, scalable to new changes, and offer perform better operating performance. Examples of
object-orientated development frameworks include Microsoft Visual Studio (2005 and 2008) and JSP (Java Server Pages).

From a web-based GIS application point of view, object oriented programming presents a better choice for developing a web-based GIS application and copes with the new challenges involving performance and scalability of applications in a real world environment [19].

4.2 Design Goals

Design goals describe the qualities of the system that developers should optimize. Such goals are normally derived from the non-functional requirements of the system.

Design goals are grouped into five categories. These are:

- Performance
- Dependability
- Maintenance
- End User Criteria

4.2.1 Performance Criteria

The part of the system to be used for the system should have a fast response time (real time) with maximum throughput. Furthermore, the system should not be taking up too much space in memory and hence the system should try to be more interactive. In the case of the timetabling subsystem, the system should be more reliable in order to satisfy the constraints than fast response time.

4.2.2 Dependability

The system needs the system to be highly dependable as it is expected to be used by non-IT professionals. The system should be robust and fault tolerant. Furthermore, as the system is handling sensitive data of the LIVES project, high emphasis should be given with regards to security, as there are subsystems to be accessed through web.
4.2.3 Maintenance

The system should be easily extensible to add new functionalities at a later stage. It should also be easily modifiable to make changes to the features and functionalities.

4.2.4 End User Criteria

Usability: Usability is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use. From the end users’ perspective the system should be designed in such a way that it is easy to learn and use, efficient and having few errors if any.

Trade-off is inevitable in trying to achieve a particular design goal. One best case is the issue of security versus response time. Checking User-Id and Password before a member can enter to the system. The other case is the issue of response time versus quality. There is some amount of time taken by the system to generate the timetable. So the user has to wait a little after telling the system to generate the timetable and getting the result to get a quality timetable.

4.3. System Architecture

Different architectures can be used to develop web-based GIS applications. 2-tier architectures retain the user interface and functional part of the web application on the first tier and deploy the database and data storage functions on the second tier. However, a more efficient architecture is a 3-tier architecture in which the user interface and functional parts are separated on different tiers. In such circumstances, if any part of the application needs to be changed, other parts do not get affected by that change, and the web application remains scalable to changes in different development environments.

When a user requests a page, using web browser and Internet, the request is received by web server (Host Gator). The web server hosts and handles the request for the website. The server prepares to send a requested page to the user using the Internet. As this process undergoes three layer of interaction, this web application can be considered as Three-Tiered Web Application, which is made of:
• The First Tier: Content management system pages created in ASP.net which is rendered by the browser.
• The Middle Tier: Application engine, to generate pages dynamically using web technology (e.g. ASP.net)
• The Third Tier: Database, allows to store data for the web application. The system can manage application content data by interacting with the database.

4. 3.1. Proposed System Architecture

To implement a web application client-server architecture is required. The most popular client-server architectures are the two-tier and the three-tier architecture. The choice of architecture affects the development time and the future flexibility and maintenance of the application. While selecting the architecture most suitable for an application, many factors including the complexity of the application, the number of users and their geographical dispersion are considered.

This system is designed based on a traditional three-tier architecture used by many web applications. Three-tier architecture includes a presentation layer, business rules/logic layer, and the data layer.

The client layer consists of a personal computer running a Web browser. This layer provides the user interface and operates by generating requests to the application server via HTTP and displays the resulting HTML file in a Web browser. The middle layer is itself a layered system consisting of a Web server layered on an application server. The Web server receives requests from the client which are processed by the application servers, then passed to the application server. The application server makes requests to the data layer via TCP/IP and ODBC. The data layer is a data repository consisting of a relational SQL database that handles user’s profiles and credentials, and one or more directories of flat files in ESRI shape file format. The ESRI shape file format is built and maintained through an off-line data migration process that involves updating the data with new shape files.
Figure 8. The architectural view of the proposed system

4.5. Hardware/Software mapping

Web Based GIS enables to distribute geographic information in a variety of forms, including maps, images, datasets, spatial analysis operations, and reports. Since Internet-GIS runs on top of the Internet, it uses World Wide Web technology. To develop an Internet-GIS a browser on the client side, secured web server that interacts with the browser and the data or shape file will be integrated as shown in figure 9.

Figure 9. Hardware-software mapping for the proposed system
4.5.1. Browser

Web browser is a client program that uses the Hypertext Transfer Protocol (HTTP) to make requests of Web servers throughout the Internet on behalf of the browser user. The browser, shown in figure 10 enables the client to interact and display spatial distribution of livestock. This software requests texts and images from the web server and executes on the client’s machine. The browser acts as an interface to the web server and represents multimedia information. These interfaces are used to send map request to the web server and receive response to view the map.

4.5.2. Web server

Web server delivers Web pages to browsers and other files to applications via the HTTP protocol. The web server processes requests for maps and related information. When a request is received, the web server performs functions such as displaying and generating livestock distribution from the shape files.

4.5.3. Data/shape file

The data represents a collection of shape files reside in single folder that can be accessible by the web server. Each shape file has the following mandatory files.

.shp — shape format; the feature geometry itself

.shx — shape index format; a positional index of the feature geometry to allow seeking forwards and backwards quickly

.dbf — attribute format; columnar attributes for each shape, in dBase IV format

4.5.4. Deployment Diagram

The deployment diagram for the Internet-GIS as shown in figure 10 represents the configuration of the run time processing nodes and other components of the system. The different clients like researchers on remote areas, other offices, NGO’s etc. can access the system using the Internet.
Figure 10. Deployment diagram for the proposed system
CHAPTER FIVE

In this chapter, the tools used in developing the prototype and the developed system are described.

5. The Prototype LIVES

5.1. Overview

Internet technologies are offering advanced solutions for GIS problems. With Internet-GIS spatial and non-spatial datasets can be provided to a broad range of users. Web Based GIS has a lot of applications with different functionalities. Some of them could be Simple functionalities like interactive mapping (zoom and pan) with spatial queries of the actual data and a visual overlay of the information and also functionalities like measuring distances, analysis and intersections of the data etc. Nevertheless costs are playing an important role. So, a cost-effective Web Based GIS solution for communities and counties based on Open-Source-software (OSS) is desirable. Using OSS is advantageous in that it is possible to maintain the source code, freely available and also will make independent from the sole-source company. Easy GIS .NET Web is a collection of ASP .NET 2.0 Mapping Web Controls for developers to add GIS map functionality to web sites, utilizing the same rendering components as the desktop version of Easy GIS .NET.

5.2. The environment and the tools used for the development

The prototype is developed for LIVES project. As it is shown on the questionnaire analysis, the LIVES project doesn’t have a web based system used to manage their livestock spatial data. Currently the LIVES perform its activities only with ArcGIS Desktop application which doesn’t have the capability to publish spatial data online unlike Web based GIS applications.

5.3. Programming Tools

To validate our approach and to facilitate the implementation of the system we used the following technologies. Visual Studio is Microsoft's flexible development tool used for building web based business applications. Visual Studio built-in languages
include VB.NET (via Visual Basic .NET), C# (via Visual C#) and so on. Easy GIS .NET is a suite of .NET 2.0 GIS and mapping tools and controls to let developers easily incorporate GIS functionality into their applications.

**Tools used are:**

- Microsoft visual studio 2010 express as working environment
- C# Programming Language
- ASP.NET 4.5 version
- Easy GIS .NET Web Edition controls as reference (like EGIS.ShapeFileLib.dll and EGIS.Web.Controls.dll)

**5.4. Outcome/expected benefits**

Upon completion of the system outcomes and benefits will be the following:

- Deliver up-to-date livestock information to everyone.
- Render quality online service to interested groups 24 hours a day.
- Creating online livestock information/mapping
- Provide adequate and organized livestock information to the public
- Obtain the current view of the map with the overlay of spatial data.
- It is easily updatable to use and require less skill

**5.5. The Spatial Distribution of Livestock – SDL Prototype**

Here, the implemented system is described. How the user interacts with the system and some of the results of interaction with the system along with the screen shots are described. In order to access the system the user has to be registered by clicking **Registration** link and required to fill his/her user name, email and password. Figure 12 show the Registration page.

After the user has registered he/she can login to the system using the login page shown in Figure 11. If the users have login successfully to the system they are directed to the home page of the system. Here, system administrator can upload new or updated ESRI shape files and the users can also click the livestock link in order to browse the livestock page.
which allows them to select and display the spatial distribution of livestock in a given area. Figure 13 shows the default/home page of the system.

Figure 11. Login Screen

Figure 12. Users registration screen
Figure 13. Default/home page of the system

Figure 14. The livestock page to search
5.6. How to Configure

The system has a practical and intuitive interface, it works simply with internet browser and it allows to view on the web any type of spatial data and to get all the information associated with one-click. It is able to display quickly complex maps on custom bases and it requires the least amount of resources possible.

The system is compatible with the latest versions of the most popular browsers, any operating system and it doesn't need any external plug-ins.

Step 1
We download the required .dll from http://www.codeproject.com/KB/asp.net/824027/dll.zip

Step 2
We start a new project as ASP.NET Web application using Visual Studio 2010 express and we give the name as Livestock. This gives us one Default.aspx and .cs file.

Step 3
Then, we add these references:

- EGIS.ShapeFileLib.dll
- EGIS.Web.Controls.dll
- geomutil_lib.dll
- geomutil_libx64.dll

Step 4
We add the below mentioned code in Web.config file under "system.web" section:

```xml
<httpHandlers>
  <add path="egismap.axd" verb="*"
       type="EGIS.Web.Controls.SFMapImageProvider, EGIS.Web.Controls,
            Version=4.3.0.0, Culture=neutral, PublicKeyToken=05b98c869b5ffe6a"
       validate="true"/>
</httpHandlers>
```

Step 5
We get the shape file from the LIVES project.

Step 6
We add the folder in our project.
Step 7
We modify .egp file

Step 8
Now our Solution Explorer will look like this:

Figure 15. Solution explorer of the system

5.7. Using the Code

Our "Livestock.egp" file:

```xml
<sproject version="1.0">
<MapBackColor>Blue</MapBackColor>
```
<name>Livestock.shp</name>  
<path>Livestock_files/Livestock</path>

- <name> field contains the shapefile name, so change it accordingly.
- <path> field contains the location of the shapefile, so change it accordingly.
- Change rest of the fields accordingly.

### 5.8. What is shape file?

The shape file format is now a common format for storing GIS data. Shape files stored non-topological vector data along with related attribute data. Developed by ESRI, shape files can be directly read by a number of GIS software programs such as ArcGIS and QGIS. A shape file is actually a collection of at least three basic files: .shp, .shx and .dbf. All three files must be present in the same directory for them to be viewable. There may
be additional files such as a .prj with the shape file’s projection information. Commonly, shape files are compressed in a .zip file for transfer such as emailing as an attachment or via a web site download.

5.9. Render with Data

using System;
using System.Data;
using System.Configuration;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using EGIS.Web.Controls;
using System.Drawing;
namespace Livestock
{
    public partial class Livestock : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            this.MapPanControl1.SetMap(this.SFMap1);
        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            if (DropDownList1.SelectedIndex == 1)
            {
                Setupcattle();
            }
            else if (DropDownList1.SelectedIndex == 2)
            {
                Setupgoat();
            }
            else if (DropDownList1.SelectedIndex == 3)
            {
                Setupsheep();
            }
        }
    }
}
private static double[] GetQuintiles(double[] samples)
{
    Array.Sort(samples);
    double[] quintiles = new double[4];
    quintiles[0] = samples[(int)(samples.Length * 0.2)];
    quintiles[1] = samples[(int)(samples.Length * 0.4)];
    quintiles[2] = samples[(int)(samples.Length * 0.6)];
    quintiles[3] = samples[(int)(samples.Length * 0.8)];
    return quintiles;
}

private void Setupcattle()
{
    TooltipHeaderFieldNamePair[] tooltipPairs = new TooltipHeaderFieldNamePair[] {
        new TooltipHeaderFieldNamePair("Region: ", "REGION"),
        new TooltipHeaderFieldNamePair("Zone: ", "ZONE"),
        new TooltipHeaderFieldNamePair("Wereda: ", "WEREDA"),
        new TooltipHeaderFieldNamePair("Number Cattle: ", "CATTLE99"),
        //new TooltipHeaderFieldNamePair("Number Donkey: ", "DONKEY"),
    }
    SetupCustomRenderSettings("CATTLE99", 0, tooltipPairs);
}

private void Setupgoat()
{
    TooltipHeaderFieldNamePair[] tooltipPairs = new TooltipHeaderFieldNamePair[] {
        new TooltipHeaderFieldNamePair("Wereda: ", "WEREDA"),
        // new TooltipHeaderFieldNamePair("Number Horse: ", "HORSE"),
        new TooltipHeaderFieldNamePair("Number Goat: ", "GOAT99"),
    }
    SetupCustomRenderSettings("GOAT99", 0, tooltipPairs);
}

private void Setupsheep()
{
    TooltipHeaderFieldNamePair[] tooltipPairs = new TooltipHeaderFieldNamePair[] {
        new TooltipHeaderFieldNamePair("Wereda: ", "WEREDA"),
    }
}
private void SetupCustomRenderSettings(string fieldName, int layerIndex, TooltipHeaderFieldNamePair[] tooltipFields)
{
    // get the required layer
    EGIS.ShapeFileLib.RenderSettings renderSettings = SFMap1.GetLayer(layerIndex).RenderSettings;
    int numRecords = SFMap1.GetLayer(layerIndex).RecordCount;
    EGIS.ShapeFileLib.DbfReader dbfReader = renderSettings.DbfReader;
    int fieldIndex = dbfReader.IndexOfFieldName(fieldName);

    double[] samples = new double[numRecords];
    // find the range of population values and obtain the quintile quantiles
    for (int n = 0; n < numRecords; n++)
    {
        double d = double.Parse(dbfReader.GetField(n, fieldIndex), System.Globalization.CultureInfo.InvariantCulture);
        samples[n] = d;
    }
    double[] ranges = GetQuintiles(samples);

    // create the quintile colors - there will be 1 more color than the number of elements in quantiles
    Color[] cols = new Color[] {
        Color.FromArgb(80, 0, 20),
        Color.FromArgb(120, 0, 20),
        Color.FromArgb(180, 0, 20),
        Color.FromArgb(220, 0, 20),
        Color.FromArgb(250, 0, 20)};

    // setup the list of tooltip fields
    System.Collections.Generic.List<TooltipHeaderFieldNamePair> tooltipPairList = null;
    if (tooltipFields != null)
    {
        tooltipPairList = new System.Collections.Generic.List<TooltipHeaderFieldNamePair>();
        tooltipPairList.AddRange(tooltipFields);
    }

    // create a new QuantileCustomRenderSettings and add it to the SFMap
    QuantileCustomRenderSettings rcrs = new QuantileCustomRenderSettings(renderSettings, cols, ranges, fieldName, tooltipPairList);
    SFMap1.SetCustomRenderSettings(layerIndex, rcrs);
After running the above code, you will see the page provided on the figure 16 with tooltip that shows Region, Zone, “wereda” name with the total number of livestock.

Users can select the livestock type they want and able to generate map. When a user generates a map the data is extracted from the shape files and displayed directly into the information box as shown on the figure below.

![Livestock Distribution](http://localhost:51285/livestock.aspx)

**Livestock Distribution**

**Select Custom Render Settings:**
- Cattle

![Map with tooltip](image)

Region: Affar  
Zone: Zone 5(AF)  
Wereda: Telalak  
Number Cattle: 32094

**Figure 16. The livestock page to view**
5.10. Minimum technical specifications to implement the system

5.10.1. Server

For operating the system needs to rely on a server machine, which must have the following minimum requirements:

- CPU 1,6 GHz
- 1GB of RAM memory
- 10 GB of free space on the disk
- Operating system Linux, Windows Server 2008

The minimum characteristics may vary in relation to the amount of data to be published and the number of users of the system.

5.10.2. Client

The system doesn't need third-part plug-ins and it is compatible with any operating system that is able to operate one of the following browsers:

- Mozilla Firefox
- Google Chrome
- Apple Safari
- Microsoft Internet Explorer 8+

For Internet Explorer is recommended version 9 or later.
CHAPTER 6

6. Conclusion and Recommendation

6.1. Conclusion

Livestock sector has a significant contribution to the national economy of the country. Livestock plays vital roles in generating income to farmers, creating job opportunities, ensuring food security, providing services, contributing to asset, social, cultural and environmental values, and sustain livelihoods. To properly manage and control this resource, adequate and organized information is required. Developing application and representing new technology in livestock management is useful for the reporting of livestock monitoring, inventory, epidemiology, distribution, information, as well as the study and modeling of specific problems.

GIS is mainly comprised of data handling tools for storage, retrieval, management and analysis of spatial data as well as solving complex geographical problems. GIS is an excellent tool for spatial data presentation, inclusion of additional layers for better analysis. The purpose of this project is to identify the research gap on the spatial distribution of livestock in LIVES project and to fill the gap by reviewing different literatures and related works and to build a prototype web based application that allows performing GIS tasks including mapping, data management, and visualization.

This project has developed an accessible Web GIS prototype, through which the spatial information of livestock data can be accessible and viewed to the world wide users. The system can help the researchers and other interested users from far destinations to explore the livestock distribution in a given area and to make decision in an appropriate manner by performing on-line GIS queries. This application can also assist the Ethiopia Sheep and Goat Productivity Improvement Program (ESGPIP), Central Statistical Agency of Ethiopia (CSA), etc. to get up to date livestock distribution information.
6.2. Recommendation

To enhance the efficiency of the system, in the following we have listed some recommendations and future works. During our study we find out that very little attention is paid on how to enhance the accessibility in GIS application and there are issues which could be handled with no effort.

- To improve the functionality of the system and upgrade the proposed system we recommend organizations to have a web developer with knowledge of C# and ArcGIS applications.
- We recommend open source web-based GIS application that allows users to share, process and edit geospatial data and can read a variety of data formats.
- There should be a standard icon for each button like Print, Save etc., which will improve the usability.
- This project is to lay the foundation to build a full-fledged web based application that allows to perform spatial distribution of livestock tasks, from simple to advanced, including mapping, geographic analysis, data editing and compilation, data management, visualization, and geo processing. Further research should be conducted to enhance the functionality of the system.
- The project is important to initiate other interested researchers to undertake a better and detailed study in the field.
REFERENCES


ANNEXES

A. Prototype

The Solution Explorer will look like this:

Livestock Home Page

Select Custom Render Settings

Please Select... Generate Map
Login Screen

Registration page of the system
Default/home page of the system

The livestock page to search
The livestock page to search

File Upload Page
B. Sample Source Code

Our "Livestock.egp" file:

```xml
<sfproject version="1.0">
  <MapBackColor>Blue</MapBackColor>
  <layers>
    <shapefile>
      <name>Livestock.shp</name>
      <path>Livestock_files/Livestock</path>
      <renderer>
        <MinZoomLevel>-1</MinZoomLevel>
        <MaxZoomLevel>-1</MaxZoomLevel>
        <MinRenderLabelZoom>-1</MinRenderLabelZoom>
        <FieldName>REGION</FieldName>
        <FieldName>ZONE</FieldName>
        <FieldName>WEREDA</FieldName>
        <FieldName>CATTLE99</FieldName>
        <FieldName>GOAT99</FieldName>
        <FieldName>SHEEP99</FieldName>
        <Font Size="12" Name="Microsoft Sans Serif" Style="Regular">
          <FontColor>Black</FontColor>
          <FillColor>#F5EEE8</FillColor>
          <OutlineColor>#969696</OutlineColor>
          <PenWidthScale>0.0001350453</PenWidthScale>
          <FillInterior>True</FillInterior>
          <Selectable>False</Selectable>
          <LineType>Outline</LineType>
          <ShadowText>True</ShadowText>
          <PointSize>5</PointSize>
          <UseToolTip>True</UseToolTip>
          <ToolTipFieldName>CATTLE99</ToolTipFieldName>
          <ToolTipFieldName>GOAT99</ToolTipFieldName>
          <ToolTipFieldName>SHEEP99</ToolTipFieldName>
        </Font>
      </renderer>
    </shapefile>
  </layers>
</sfproject>
```
using System;
using System.Data;
using System.Configuration;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using EGIS.Web.Controls;
using System.Drawing;

namespace Livestock
{
    public partial class Livestock : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            this.MapPanControl1.SetMap(this.SFMap1);
        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            if (DropDownList1.SelectedIndex == 1)
            {
                Setupcattle();
            }
            elseif (DropDownList1.SelectedIndex == 2)
            {
                Setupgoat();
            }
            elseif (DropDownList1.SelectedIndex == 3)
            {
                Setupsheep();
            }
        }
    }

    // simple method to return quintile quantiles from an array of double samples
    private static double[] GetQuintiles(double[] samples)
    {
        // Code to calculate quintiles
    }
}
Array.Sort(samples);

double[] quintiles = new double[4];
    quintiles[0] = samples[(int)(samples.Length * 0.2)];
    quintiles[1] = samples[(int)(samples.Length * 0.4)];
    quintiles[2] = samples[(int)(samples.Length * 0.6)];
    quintiles[3] = samples[(int)(samples.Length * 0.8)];
return quintiles;
}

private void Setupcattle()
{
    TooltipHeaderFieldNamePair[] tooltipPairs;
    tooltipPairs = new TooltipHeaderFieldNamePair[] {
        new TooltipHeaderFieldNamePair("Region: ", "REGION"),
        new TooltipHeaderFieldNamePair("Zone: ", "ZONE"),
        new TooltipHeaderFieldNamePair("Wereda: ", "WEREDA"),
        new TooltipHeaderFieldNamePair("Number Cattle: ", "CATTLE99"),
        //new TooltipHeaderFieldNamePair("Number Donkey: ", "DONKEY"),
    }
    SetupCustomRenderSettings("CATTLE99", 0, tooltipPairs);
}

private void Setupgoat()
{
    TooltipHeaderFieldNamePair[] tooltipPairs;
    tooltipPairs = new TooltipHeaderFieldNamePair[] {
        new TooltipHeaderFieldNamePair("Wereda: ", "WEREDA"),
        // new TooltipHeaderFieldNamePair("Number Horse: ", "HORSE"),
        new TooltipHeaderFieldNamePair("Number Goat: ", "GOAT99"),
    }
    SetupCustomRenderSettings("GOAT99", 0, tooltipPairs);
}

private void Setupsheep()
{
    TooltipHeaderFieldNamePair[] tooltipPairs;
    tooltipPairs = new TooltipHeaderFieldNamePair[] {
        new TooltipHeaderFieldNamePair("Wereda: ", "WEREDA"),
        // new TooltipHeaderFieldNamePair("Number Horse: ", "HORSE"),
        new TooltipHeaderFieldNamePair("Number Sheep: ", "SHEEP99"),
    }
    SetupCustomRenderSettings("SHEEP99", 0, tooltipPairs);
}
private void SetupCustomRenderSettings(string fieldName, int layerIndex, TooltipHeaderFieldNamePair[] tooltipFields) {

    //get the required layer
    EGIS.ShapeFileLib.RenderSettings renderSettings = SFMap1.GetLayer(layerIndex).RenderSettings;
    int numRecords = SFMap1.GetLayer(layerIndex).RecordCount;
    EGIS.ShapeFileLib.DbfReader dbfReader = renderSettings.DbfReader;
    int fieldIndex = dbfReader.IndexOfFieldName(fieldName);

    double[] samples = new double[numRecords];
    //find the range of population values and obtain the quintile quantiles
    for (int n = 0; n < numRecords; n++)
    {
        double d = double.Parse(dbfReader.GetField(n, fieldIndex),
                     System.Globalization.CultureInfo.InvariantCulture);
        samples[n] = d;
    }
    double[] ranges = GetQuintiles(samples);

    //create the quintile colors - there will be 1 more color than the number of elements in quantiles
    Color[] cols = new Color[] {
        Color.FromArgb(80, 0, 20),
        Color.FromArgb(120, 0, 20),
        Color.FromArgb(180, 0, 20),
        Color.FromArgb(220, 0, 20),
        Color.FromArgb(250, 0, 20)};

    //setup the list of tooltip fields
    System.Collections.Generic.List<TooltipHeaderFieldNamePair> tooltipPairList = null;
    if (tooltipFields != null)
    {
        tooltipPairList = new System.Collections.Generic.List<TooltipHeaderFieldNamePair>();
        tooltipPairList.AddRange(tooltipFields);
    }

    //create a new QuantileCustomRenderSettings and add it to the SFMap
    QuantileCustomRenderSettings rcrs = new QuantileCustomRenderSettings(renderSettings, cols, ranges, fieldName, tooltipPairList);
    SFMap1.SetCustomRenderSettings(layerIndex, rcrs);
}
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.IO;
namespace Livestock
{
    public partial class _Default : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        bool correctExtension = false;
        if (FileUpload1.HasFile)
        {
            string fileName = FileUpload1.PostedFile.FileName;
            string fileExtension = Path.GetExtension(fileName).ToLower();
            string[] extensionsAllowed = { ".dbf", ".docx", ".txt" };
            for (int i = 0; i < extensionsAllowed.Length; i++)
            {
            }
            if (fileExtension == extensionsAllowed[i])
            {
            }
correctExtension = true;
}

if (correctExtension)
{
try
{
string fileSavePath = Server.MapPath("~/Livestock_files/");
FileUpload1.PostedFile.SaveAs(fileSavePath + fileName);
Label3.Text = "File successfully uploaded";
}
catch (Exception ex)
{
Label3.Text = "Unable to upload file";
}
}
else
{
Label3.Text = "File extension " + fileExtension + " is not allowed";
}
}
DECLARATION

This project is our original work and has not been submitted as a partial requirement for a degree in any university

(Name and Sig)  (Name and sig)  (Name and sig)

(Name and Sig)  (Name and sig)  (Name and sig)