



DEBRE BERHAN UNIVERSITY

COLLEGE OF COMPUTING

DEPARTMENT OF INFORMATION SYSTEMS

A PROJECT ON: WEB_BASED POPULATION AND HOUSING CENSUS MANAGEMENT SYSTEM FOR DEBRE BERHAN

**PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENT FOR THE AWARD OF THE DEGREE OF BACHELOR
OF SCIENCE IN INFORMATION SYSTEMS**

SUBMITTED BY

NAME OF STUDENTS

ID NO

- 1. YESHEWAGET GEBRESELLASSIE.....DBUR/1811/11**
- 2. FETENE ERKUTENA..... DBUR/1654/11**
- 3. EPHERME YIRGA..... DBUR/1729/11**
- 4. METI FEYISA..... DBUR/1871/11**
- 5. YONASE MARYE..... DBUR/1775/11**

PROJECT ADVISOR: KINDE ALEBACHEW

AUGUST, 2022

DEBRE BERHAN UNIVERSITY, DEBRE BERHAN, ETHIOPIA

DEBRE BERHAN UNIVERSITY

COLLEGE OF COMPUTING

DEPARTMENT OF INFORMATION SYSTEMS

A PROJECT ON: WEB_BASED POPULATION AND HOUSING CENSUS MANAGEMENT SYSTEM FOR DEBRE BERHAN

**PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENT FOR THE AWARD OF THE DEGREE OF BACHELOR
OF SCIENCE IN INFORMATION SYSTEMS**

SUBMITTED BY

NAME OF STUDENTS

ID NO

- 1. YESHEWAGET GEBRESELLASSIE.....DBUR/1811/11**
- 2. FETENE ERKUTENA..... DBUR/1654/11**
- 3. EPHERME YIRGA..... DBUR/1729/11**
- 4. METI FEYISA..... DBUR/1871/11**
- 5. YONASE MARYE..... DBUR/1775/11**

PROJECT ADVISOR: KINDE ALEBACHEW

AUGUST, 2022

DEBRE BERHAN UNIVERSITY, DEBRE BERHAN, ETHIOPIA

DECLARATION

The project entitled with **Web-Based Population and Housing Census Management System For Debre Berhan** is submitted to the **Debre Berhan University, College of Computing Department of information systems** for the award of BSC in **information systems** is based on our original work carried out under the guidance of **Kinde Alebachew**. The project has not been submitted elsewhere for award of any degree.

The material borrowed from other source and incorporated in the project has been duly acknowledged and/or referenced.

We will be responsible and liable for plagiarism, if any, detected later on.

Date: _____

Name of the students

Signature

1. Yeshewaget Gebresellassie

2. Fetene Erkutena

3. Epheme Yirga

4. Meti Feyisa

5. Yonase Marye

Name of the Advisor

Signature Date

APPROVAL FORM

This is to confirm that the project report entitled **Web-Based Population and Housing Census Management System For Debre Berhan** submitted to **Debre Berhan University, College of Computing Department of information systems** by: **Group 5** is approved for submission.

Advisor Name	Signature	Date
-----	-----	-----
Department Head Name	Signature	Date
-----	-----	-----
Examiner 1 Name	Signature	Date
-----	-----	-----
Examiner 2 Name	Signature	Date
-----	-----	-----
Examiner 3 Name	Signature	Date
-----	-----	-----
Examiner 4 Name	Signature	Date
-----	-----	-----

ACKNOWLEDGMENT

First, we would like to thank Almighty (omnipotent) GOD, for giving strength; he has given us throughout our life and the project. Secondly, we would like to express our deepest gratitude to our advisor instructor kinde Alebachew for his excellent guidance, caring, patience, and for providing us with an excellent atmosphere for doing this documentation. We would also like to express our gratitude to Debre Berhan'S population and housing census officer who gave enough information for our interviews with patience and full of voluntariness. Finally, thanks to people who have directly or indirectly supported our project.

Table of Contents

DECLARATION	ii
APPROVAL FORM	iii
ACKNOWLEDGMENT	iv
List of table	vii
List of figure	viii
Acronyms	x
ABSTRACT	xi
CHAPTER ONE	1
INTRODUCTION	1
1.1 Introduction.....	1
1.2 Background of the project.....	2
1.3 Statement of the problem	2
1.4 Objective the Project.....	3
1.4.1 General Objective	3
1.4.2 Specific Objective	3
1.5 Scope and Limitation of the Project.....	4
1.5.1 Scope of the Project	4
1.5.2 Limitation of the project	4
1.6 Significant of Study	4

1.7 Beneficiaries of the system	4
1.8 Feasibility study of the new system	5
1.8.1 Technical and Operational feasibility	5
1.8.2 Economic Feasibility.....	6
1.8.3 Political feasibility	6
1.9 Methodology and tool	6
1.9.1 Data gathering methodology	6
1.9.2 System Analysis and Design Methodology	6
1.9.3 Software requirements:	7
1.9.4 Hardware requirements:.....	7
1.9.5 Language.....	7
1.10 Requirement specification testing.....	8
1.11 Assumption and constraint.....	8
1.11.1 Assumption	8
1.11.2 Constraint.....	9
CHAPTER TWO	10
DESCRIPTION OF THE EXISTING SYSTEM.....	10
2.1 Introduction.....	10
2.2 Players in the existing system	10
2.2.1 Enumerator.....	10

2.2.2 Supervisor	10
2.2 Major functions.....	11
2.3 Business Rules	11
2.4 Report, forms, and other documents generated in the existing system	12
2.4.1 Report generated	12
2.4.2 Forms used in the existing system	12
2.5 Problem of the existing system	14
2.6 Practice to be preserved	15
2.7 Proposed solution.....	15
2.8 Requirements of the Proposed System.....	15
2.8.1 Functional requirements.....	15
2.8.2 Non- functional requirements	16
CHAPTER THREE	17
SYSTEM ANALYSIS	17
3.1 Introduction	17
3.2 Use Case Modelling.....	17
3.3 Use case description.....	18
3.4 Use case Scenario	19
3.5 Use Case description.....	19
3.6 Analysis level Class Diagram (Conceptual Modelling).....	26

3.7 Sequence diagram	26
3.8 Activity diagram	30
3.9 State chart diagram	36
3.11 User Interface Prototyping	39
3.12 Supplementary specifications	40
3.12.1 Business Rules	40
3.12.2 Constraints	41
CHAPTER FOUR.....	42
SYSTEM DESIGN	42
4.1 Introduction.....	42
4.2 Purpose of the system	42
4.3 Design goal	42
4.4 System Architecture.....	43
4.5 Class Modelling	44
4.6 Component Diagram.....	45
4.7 Subsystem decomposition.....	46
4.8 Deployment Diagram.....	48
4.9 Data Dictionary	49
4.10 Database Design.....	51
CHAPTER FIVE	52

IMPLEMENTATION	52
5.1 Introduction.....	52
5.2 Hardware software acquisitions	52
5.3 Language specification and selection strategy	53
5.4 Sample code	53
5.5 Test procedures	63
5.6 User manual preparation	63
5.6.1 Training.....	64
5.6.2 Installation Process	64
5.6.3 Start-up Strategy	64
CHAPTER SIX.....	65
CONCLUSIONS AND RECOMMENDATION.....	65
6.1 Conclusion	65
6.2 Recommendation	65
REFERENCE.....	67
Appendix.....	68

List of table

Table 1: login description	20
Table 2 view description	20
Table 3 for fill census form description	21
Table 4 for fill house form description	21
Table 5 search description	21
Table 6 manage account description	22
Table 7 manage user account description	22
Table 8 manage user account description	23
Table 9 manage enumerator account description	23
Table 10 View census description	24
Table 11 View census description	24
Table 12 View census description	25
Table 13 View census description	25
Table 14 Administrator table	49
Table 15 supervisor table	50
Table 16 enumerator table	50
Table 17 census data table	51

List of figure

Figure 1 generated report	12
Figure 2 Area identification form	13
Figure 3 Type of residence and housing identification form	14
Figure 4: use a case diagram	18
<i>Figure 5: conceptual modeling</i>	26
Figure 6: Sequence diagram for login	27
Figure 7: Sequence diagram for fill form	28
Figure 8: Sequence diagram for the update	29
Figure 9: Sequence diagram for delete	30
Figure 10: Activity diagram for login	31
Figure 11: Activity diagram for fill	32
Figure 12: Activity diagram for the update	33
Figure 13: Activity diagram for publishing	34
Figure 14; Activity diagram for adding actor	35
Figure 15: Activity diagram for search	36
Figure 16: State diagram for login	37
Figure 17: State diagram for populate	38
Figure 18: State diagram for generating the report	39

Figure 19: user interface prototype	40
Figure 20: Software Architecture.....	44
Figure 21: Class modeling diagram	45
Figure 22: component diagram	46
Figure 23: Subsystem decomposition diagram	47
Figure 24: Deployment diagram	48
Figure 25: database design.....	51
Figure 26 home page interface.....	53

Acronyms

Admin	Administrator
BR	Business Rule
CSS	Cascading Style Sheet
DBU	Debre Berhan University
GUI	Graphical User Interface
GB	Gigabyte
HTML	Hypertext modeling language
HTTP	hypertext transfer protocol
ID	identification number
OOSD	object-oriented system development
PHP	Hypertext pre-processor
RAM	Random Access Memory
SQL	Structured Query Language
SRS	System requirement specification
Sup	Supervisor
UC	use case
UI	user interface
UML	Unified Modelling Language

ABSTRACT

Population and housing censuses are the foundation of statistical systems. The system currently uses a manual system. Our project is done by automating the whole process allowing online. This project mainly concentrates on improving the efficiency of census data collection. There are many problems affecting the population and housing census. Some are Poor and unstable organization, Manual bulk carrying of data, Lack of automated statistical manipulations, inadequate manpower, Document Mismanagement, High Data redundancy and erroneous data storing.

The general objective of this project is to develop a web-based population and housing census system for Debre Berhan. The methodology that used in our project is data gathering methodology like Interviewing, Observation and Questionnaires. System analysis and design methodology we choose to use an iterative model. We use hardware (Desktop, 16GB flash) and software tool (Xampp, Edraw Max and Notepad++) and language (PHP, SQL, Java script, HTML). Requirement specification testing are Unit Testing, Integration Testing, System Testing, Functionality Testing.

The proposed solution that we used are developing an automated system like user-friendly, and interactive graphical user interface system that will: reduce the complexity of the existing system, manage time effectively, make work easy, make the data error-free, utilize available resources effectively, Enhance the efficiency and diversification of services activities. The system is going to be developed so as to help the user to enumerate the population online, validate census recording process, manipulate the census data and represent it statistically, calculate critical rates and indicators from the census data etc. the team would recommend that further work should done on the system in order to make the system perform better for interested organizations.

CHAPTER ONE

INTRODUCTION

1.1 Introduction

Population and housing censuses are the foundation of statistical systems, providing benchmarks of a country's population and housing stock, and baseline information for the production of other statistics.

Population Census is the total process of collecting, classifying evaluating, analysing, and publishing or distributing demographic, economic, and social data of the population during a certain period. Population considers an essential element for the production and distribution of material wealth, for planning and implementation of economic, social development or administrative activity or scientific research requires the existence of reliable and detailed data on the size, distribution, and composition of the population. Population Census forms a most important source, of these basic statistics for comparison.

The increasing complexity of modern life means a greater need to plan housing, schools, roads, transportation, and a vast range of social and economic requirements for the nation. This cannot be done without a detailed count of the population. Census is being officially managed by some organizations or governments, for example, the National Population Commission (NPC). Ethiopia has so far conducted 3 Population and Housing Censuses (PHC) in 1984, 1994, and 2007. The preparations to conduct the 2017 PHC (4th Census) are currently underway, and one of these major undertakings is related to Census data capturing and processing.

The installation of computer in these organizations or bodies that manages census information will assist not only in fast-recording information but also in solving certain problems, which cannot be easily resolved manually. The human resource of any enterprise is considered to be their most valuable assets if they are properly harnessed and are well motivated to perform their assigned tasks in such a manner as to enhance the enterprise's goals and objectives.

Therefore, adequate population records will provide all the necessary information that is associated with people, which includes the size of the population, age structure, educational attainment, labour force, and socio-economic characteristics, unlike the manual method which makes access to data and information very tedious. The integration of different databases so that

these databases can be merged and processed together and mainly other reasons, prompted the researcher to develop software for this organization, the National Population Commission (NPC).

1.2 Background of the project

Census taking in Ethiopia can be traced to have started as far back as 1989 year. Since then, there have been several attempts to count the Ethiopian population. However, these censuses are characterized by some difficulties and deliberate and ill-minded attempts to inflate population figures; just in favor of one geopolitical zone or the other. This does not and cannot represent the nation's image as regards the human population. As a result of this, the National Population Commission (NPC) was established using Decree No. 43 of 1989. The brain behind this was to have successful censuses each time and as well accurate demographic data.

The system currently uses a manual system such as paper for the management system which leads the system to be inefficient. Some of the problems are material records issues, searching and getting different data it takes a long time; loss of documents, security problems, and retrieval problems to have occurred.

1.3 Statement of the problem

There are many problems affecting the population and housing census from maintaining steady reliable figures and estimates. These are the more reasons, why the current system has faced a problem;

- Inadequate manpower
- Lack of equipment
- Poor and unstable organization
- Manual bulk carrying of data
- Requires more Resources: Consumes more resources such as manpower, cost, paper, pen, rubber, transportation, and storage place.
- Lack of automated statistical manipulations.
- High Data redundancy and erroneous data storing.

- The current system is not efficient, flexible, reliable, available, and difficult to get data.
- Document Mismanagement: Chances of losing the census data are high.

1.4 Objective the Project

1.4.1 General Objective

The general objective of this project is to develop a web-based population and housing census system for Debre Berhan.

1.4.2 Specific Objective

To achieve the above-mentioned general objective, the project includes the following specific objective

- To develop a web-based system that stores and retrieves all information on the human population.
- To create a user interface for enumerators to login into the system and organize the data adequately to solve the security problem.
- To develop a reliable system that could be used in collecting data/information on the human population.
- To develop a system that will support direct access to the specific and required information
- To reduce the time taken by management in coming up with concrete decisions
- To deal with data redundancy within the database
- To Help the efficient collection, storing, updating, processing, and analysis of data for ease in manipulation
- Report

1.5 Scope and Limitation of the Project

1.5.1 Scope of the Project

The major aim of the project is to design a system that will have all information about the human population and retrieving of data whenever needed in society. It focuses on the registration, retrieval, and management of information about individuals in society.

The proposed system focused on the Debre Birhan city statistics agency and covers only the population and housing units.

1.5.2 Limitation of the project

The proposed system may have the following limitation.

- ❖ The proposed system is not supported by other language rather than English.
- ❖ The proposed system is not accessed by blind people.
- ❖ The proposed system is not mobile based application.

1.6 Significant of Study

The proposed project will solve the problem associated with the acquisition, storage, and retrieval of information on the human population with ease.

- Timely retrieval of information is anticipated with efficiency and reliability.
- It will provide security to data that are unauthorized, users will not gain access to those files and fraud will be minimized in the society which will lead to improvement in administration processes.

1.7 Beneficiaries of the system

The beneficiary of this system includes:

- Administrator: - unlike the existing system the proposed system would enable the admin to manage the system much more easily since it's mostly done digitally.

- Supervisor: -benefit from the system by easily supervising the enumerator online by using the browser and avoiding physical workload.
- Journalists and researchers can access or get organized data easily from the system to update publish and use related articles for magazines and blogs.
- Government ministries and Local authorities: - They will get error-free census data to provide appropriate budgeting for infrastructure or other services and future policy making.
- Also we are a beneficiary from our project.
- Private and public companies: - They can access or get organized and accurate data for different purposes.

1.8 Feasibility study of the new system

The feasibility study is the preliminary study that determines whether a proposed system project is financial, technically, and operationally viable. The alternative analysis is usually included as part of the feasibility study and identifies viable alternatives for the system design and development.

1.8.1 Technical and Operational feasibility

Our project is technically feasibly because it is fast and easy to use but in a manual system it is difficult since it is manual

- Our proposed system is a time and resource saver in the census process it does not need more resources than the existing system
- Assessing the size of the manpower, census form, and other equipment required for the census is simple
- Easily Provide comparable workloads to the enumerators
- Ensure that the census is completed in the specified time
- Operationally our system is not complex.
- Our proposed system is digital so it reduces human power the supervisor assigns enumerator for each kebele.
- Our project has a user-friendly interface that can be implemented easily and can perform many tasks that can be used by users.

1.8.2 Economic Feasibility

In the existing system, many people are involved in the process but in the proposed system, the number of persons involved is reduced drastically.

After this project is finished it will reduce the cost of the paper, pens, and employees to organize the papers will be avoided since we automate the system from the manual system so that our project is economically feasible.

1.8.3 Political feasibility

Our project will not conflict with the rule and regulations of the Ethiopian constitution and Ethiopian statistical agency rather it gives an advantage to our country.

1.9 Methodology and tool

1.9.1 Data gathering methodology

The following data gathering methodology is to develop the proposed system;

- **Interviewing** A planned meeting during which the interviewer obtains information from the interviewee. We interview the Coordinator. This will enable us to know the requirements and if any training will be required before the system is implemented.
- **Observation** – we have observed the Ethiopian population and house census to gain basic information this helps us to see the problems the enumerator, and supervisors are facing. This can be done by attending one census process. In the existing system there is a different problem in terms of time and economics, so our system answer this problem,
- **Questionnaires:-**It contains fixed-response questions about various features of an organization. This is also another data gathering method that is used for collecting information from the stake holder. This can be in terms of written paper we have distributed for individuals for the Debre-Birhan population and housing census branch.

1.9.2 System Analysis and Design Methodology

We choose to use an iterative model of a software process model. This is because this model matches our situation. We plan to develop the basic features in the first iteration and

continuously add more features as we progress to the final iterations. Each iteration consists of a complete system development life cycle i.e. requirement gathering, design and development, testing, and implementation.

We choose the iterative model because of iterative model approach is suitable for changing requirements. In addition, it's easier risk management and promote teamwork. Furthermore, we can add a new feature to the system.

1.9.3 Software requirements:

- Windows 7 or higher
- Microsoft Office 2010: this software is used to write the documentation of the system.
- Notepad++: this editor is used to write the code for the system.
- Latest Browsers: is used to search for what we want.
- Xampp Apache, MySQL: we use Xampp as our server and database.
- Edraw Max: software is used to model the system model and used to draw the diagram.

1.9.4 Hardware requirements:

- Processor – dual-core
- Memory – 1GB RAM
- Desktop computer We use a desktop and laptop computer for all project activities
- Printer: to print our documentation.
- 16 GB flash drive is used to make a backup of important information in case of any damage.

1.9.5 Language

- PHP (We use PHP language for the system development; back end to create the user interface and our system (software) will be compatible on all hardware).
- SQL: a language used for database

- Java script: to validate the forms
- CSS: Using for the style of the website.
- HTML: is used to develop the structure of the system and the front end of the system.

1.10 Requirement specification testing

We set two approaches to test the proposed system. These approaches are functionality testing of the proposed system and implementation testing of the system. The examiners will test the functionality of the system and the developers will test the implementation of the proposed system. There are different testing levels for a system. However, we will use the following testing levels. These are:

- **Unit Testing:** While coding, the project team performs some tests on that unit of a program to know if it is error free. This testing level helps us to decide that individual units of the program are working as per requirement and are error free.
- **Integration Testing:** Even if the units of the system are working fine individually, there is a need to find out if the units were integrated together would also work without errors.
- **System Testing:** The software is compiled as product and then is tested as a whole. This can be accomplished using:
- **Functionality Testing:** Tests all functionality of the software against the requirement.

1.11 Assumption and constraint

1.11.1 Assumption

- The system should connect to the network and internet.
- The users should have a smartphone or computer with sufficient knowledge about usage.
- The users must know the English language
- The employee must have basic skills and knowledge of computer to accesses the proposed system.

1.11.2 Constraint

Constraints are situations or uncomfortable conditions that restrict us from fully achieving the goal of our project. The following are problems that hinder us from accomplishing our tasks.

Technical problems: Our knowledge and experience of project work are not that much sufficient and we kill more time studying and understanding some references and supportive courses about project work.

Unfortunate failure of system: To handle this problem the teams have some methods to resist not completely but partially by using backup mechanisms using flash disks and by storing the data on our Email account.

Power problem: we tried to use laptops to cover the gap that happened to our project during power failure.

Time management problem: we solve this problem by working cooperatively, dividing our time by schedule for each phase of the project and we try to use this schedule effectively.

CHAPTER TWO

DESCRIPTION OF THE EXISTING SYSTEM

2.1 Introduction

As is described in the first chapter the existing system is done manually. Registration, documentation, writing, search, and retrieval of the specific information of the population are done manually. This type of system makes the worker document erroneous and redundant information, lacks automated statistical manipulations, decreases flexibility, and also consumes the time of the worker for completing a specific task. Moreover, there is no logging function available to make the system secure.

When analysing an existing system, the note is taken on how the existing system works or the procedures on how jobs and activities are been carried out in the organization. During system analysis, an investigation of an existing system to understand its operation is carried out for a better understanding of the existing system, and the introduction of more efficient and economic means of achieving the desired goals is also made.

2.2 Players in the existing system

2.2.1 Enumerator

The Enumerator is the one who has the privilege to collect and fill the census by collecting the information of the people manually. Each enumerator is given the map of an enumeration area along with other census documents and he/she is responsible to record all persons and Households in that enumeration area without omission and duplication. Each enumerator has a national enumerator number given by the Debre Berhan city statics agency to identify every enumerator. The enumerator validates the collected census by its name and signature.

2.2.2 Supervisor

The supervisor is the one who has a privilege which is given by Debre Berhan city population and housing census to supervise and validate the collected census data by using its signature. A

supervisor is assigned to a supervision area and is responsible for ensuring the quality of the information collected in the area of his/her jurisdiction.

2.2 Major functions

Data Input: Data has been inputted directly by the enumerator.

Data Processing: Count the inputted information collected by the enumerator manually.

Data Output: The data collected so far is recorded in a document and the document is delivered to the Debre Berhan city population and housing census.

2.3 Business Rules

A business rule is an operating principle or policy the software must satisfy. It often pertains to access control issues, business calculations, or operating policies and principles of the system. Therefore, our new system has the following business rules:

BR1: the Enumeration area should be mapped and a unique code is given for each area.

BR2: A person to be counted as a member of a given family must live there at least for 6 months.

BR3: A newly married person is counted with his/her new family regardless of the time she/he started to live with her/his new family.

BR4: For one woreda 6 enumerators and one supervisor should be assigned.

BR5: 200 families are given to one enumerator for urban areas.

BR6: The enumerator is given up to 150 families for rural areas.

BR7: The enumerators should be well trained.

2.4 Report, forms, and other documents generated in the existing system

2.4.1 Report generated

In the existing system, the report was generated by a manual system from the collected information.

Table 7.5 Reproductive Age of Women by Five Year Age Group, Birth During the Last 12 Months Prior to the Census, Age-Specific Birth Rate, and Total Fertility Rate(TFRs), by Urban - Rural Residence: 2007

Age Group	Urban + Rural		Urban		Rural		Age-Specific Birth Rates		
	Women	Births	Women	Births	Women	Births	U+R	Urban	Rural
NORTH SHEWA ZONE									
15 - 49	425,689	47,492	65,299	3,226	360,390	44,266			
15 - 19	106,560	4,919	21,011	303	85,549	4,616	.046	.014	.054
20 - 24	78,597	12,786	12,874	1,009	65,723	11,777	.163	.078	.179
25 - 29	71,595	12,656	10,755	908	60,840	11,748	.177	.084	.193
30 - 34	49,192	7,799	6,631	499	42,561	7,300	.159	.075	.172
35 - 39	49,056	6,012	6,468	380	42,588	5,632	.123	.059	.132
40 - 44	36,215	2,200	3,814	79	32,401	2,121	.061	.021	.065
45 - 49	34,474	1,120	3,746	48	30,728	1,072	.032	.013	.035

Figure 1 generated report

2.4.2 Forms used in the existing system

These form sections are used to collect data from the population Area identification

Type of residence and housing identification

SECTION 2: TYPE OF RESIDENCE AND HOUSING IDENTIFICATION

1. Type of Residence

Conventional household	[1]
Hotel/Hostel	[2]
Orphanage	[3]
Boarding School/College/University	[4]
Correctional facility	[5]
Other Collective Quarters	[6]
Pastoralists (wandering in search of water and grass)	[7]
Homeless	[8]

2. Housing Serial Number

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]

3. Household Serial Number

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]

Figure 3 Type of residence and housing identification form

2.5 Problem of the existing system

Performance:

In the existing manual system, there is more paperwork and didn't respond very fast.

Input: In the existing system data was collected door to door by the enumerator.

Security: In the existing system there is no data security mechanism to protect the collected data.

Efficiency: Since the existing system is manual it didn't provide efficient data.

2.6 Practice to be preserved

- Classification of census area by zones to identify the specific area to the enumerator.
- Enumerator fills the census by collecting the information about the people.
- Giving code for the enumeration area.

2.7 Proposed solution

We are going to improve the drawbacks of the existing system by developing an automated, user-friendly, and interactive graphical user interface system that will:-

- reduce the complexity of the existing system,
- manage time effectively,
- make work easy,
- make the data error-free
- utilize available resources effectively,
- Enhance the efficiency and diversification of services activities

2.8 Requirements of the Proposed System

- Here we have two requirements.

2.8.1 Functional requirements

The main functional requirements of this system are:

- Add Employees
- Manage Employees
- View Reports
- Searching of data.
- Login
- View new report
- Validated Members
- Send feedback
- View feedback
- Generate report

- Add Family Member (with details like name, job, sex, email, phone, Education, Occupation, Address, Area, subCity, State) specify if any other,
- Upload ID Card / birth certificate / Document for each Member
- The system shall record information about persons and housing units.
- The system shall display the information of individuals.

2.8.2 Non- functional requirements

The non-functional requirement of the system deals with how well the system provides service to the user.

- **Performance:**

Easy to use: unlike manual, the automated system is easy to use.

Fast and reliable: The time needed to access the data is much less than the existing system.

- **Maintainability:** To ensure that the system continues to work properly by checking it regularly and making repairs and adjustments if required.
- **Scalability:** The system should provide flexibility and production of new versions suited for new environments and changing needs.
- **Usability:** The system should be easy to use by all.
- **Availability:** The system should be up and running whenever needed.
- **Security and access permission:** the system should provide controlled access to information while on transmission, only authorized users should access and modify data.
- **Supportability:** since our project is done by PHP, the code can run in any browser.
- **Legal:** our group uses paid versions of software to develop the system.
- **Interface:** our group designs the user interface by PHP and CSS to make the interface more attractive.

CHAPTER THREE

SYSTEM ANALYSIS

3.1 Introduction

System design can be regarded as the drawing, planning, sketching, or arranging of many separate elements into a viable unified whole. While the system analysis phase is concerned with the question of what the system is doing and what it should be doing to meet users' requirements, the system design phase centers on how the system is developed to meet the requirements.

The goal of the chapter is to prepare the designer for the project's needs and requirements before the actual system is implemented. The system is first evaluated to verify how it operates and then the proposed system is designed to its specifications with technological options. It is a description of the structure of the software to be implemented, the data which is part of the system, and the interface between the system component and the algorithms used. The objective of this document is to detail the operation of the information system. The design document will hence help in the final design of the system. The Graphical representation will be used to create a clear understanding and expected graphical user interface.

3.2 Use Case Modelling

Use cases are used to represent the functionality of the system. Use case focused on the behavior of the system from an external point of view. A Use case describes a function provided by the system that yields a visible result for an actor. An actor describes any entity that interacts with the system like an administrator, supervisor, enumerator or Guest.

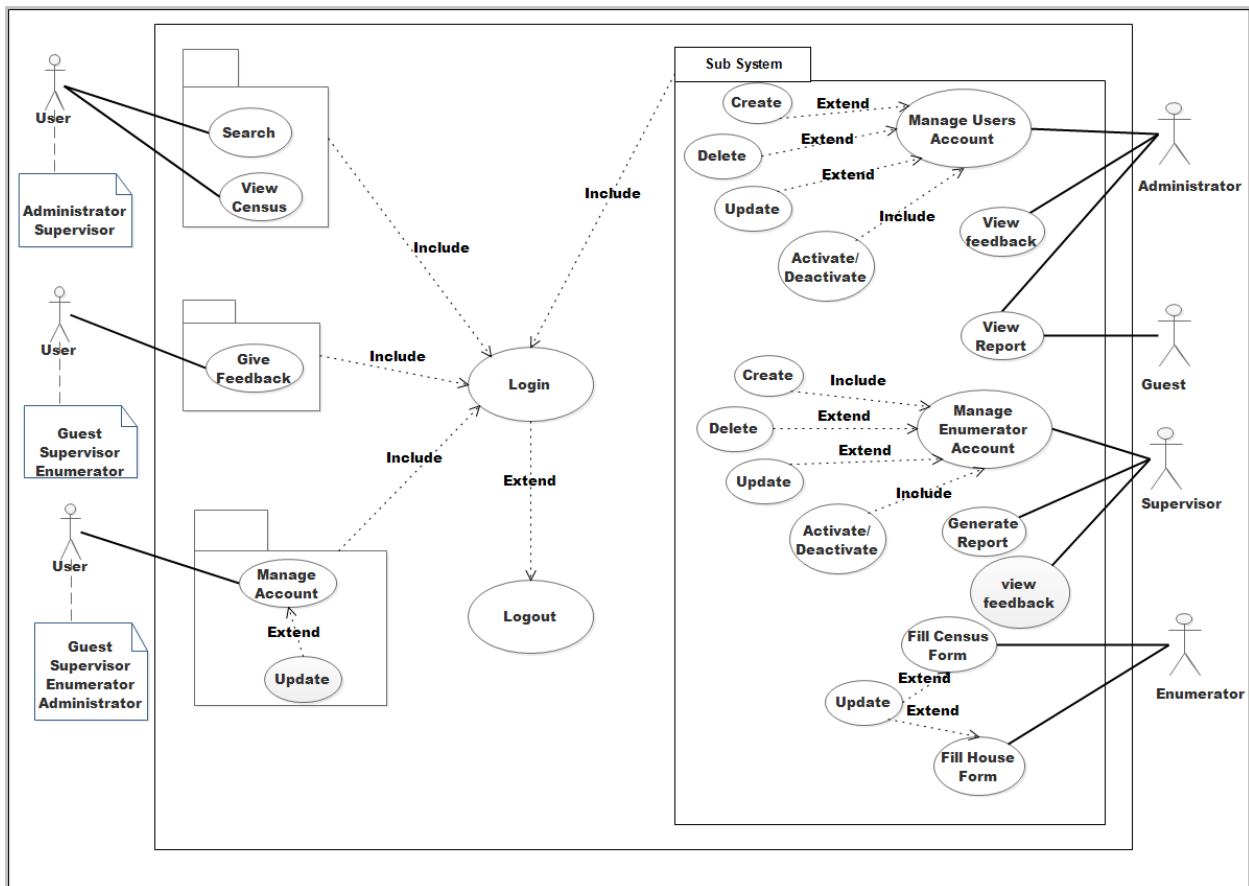


Figure 4: use a case diagram

3.3 Use case description

Use case diagrams are usually referred to as behavior diagrams used to describe a set of functions that some system or systems should perform in collaboration with one or more **external users** of the system.

Each use case should provide some observable and valuable result to the actors or other stakeholders of the system.

The proposed system use case includes the following:

- Login
- Create/Modify the account
- View census report

- Fill census form
- View census data
- Search
- Add enumerator
- Give feedback
- View feedback

3.4 Use case Scenario

A scenario is a sequence of steps that represents a single-use case execution. It is made up of several simple, discrete steps that are designated as performed by either the "System" or a "User". This scenario considers the goal of describing our project fulfillment in a short history. In this project, some participate in the system and each has its role. In the population census management system, Actors have many roles. Thus Administrator manages (create, update, delete) the supervisors, and the Supervisors manage (create, update, delete) the enumerators finally Enumerators fill out the census data and submit the form to their respective supervisors. Besides managing supervisors, the administrator has the role of summarizing, verifying the data, and finally publishing it to the main site. And also other parties who are interested in the data and want to use it for different matters like the Federal government National Population commission and non-governmental companies like journalists and foreign institutes.

3.5 Use Case description

Use case scenario for login

Use case name	Login
Use case ID	UC-1
Participating Actors	Administrator, Enumerator, Supervisor& Guest
Description	The authentication for authorized users in the system and deliver them the right to visit their specified page
Precondition	user must have email and password
Basic course of action	action Step: <ol style="list-style-type: none"> 1. User initiate login system 2. System display login form

	<ol style="list-style-type: none"> 3. The user inputs the user email and password 4. The system checks the validity of the entry and then verifies whether the user is authenticated and authorized. 5. The system displays the requested page for further action.
The Alternate flow of Action	1. If the user's entry (email and Password) is not correct the system displays an error message and returns to step 2
Post condition	The actor can now use the system.
Exit condition	The actor logs out from the system

Table 1: login description

Use case scenario to view census report

Use case name	view census report
Use case ID	UC-2
Participating Actor	Administrator and Guest
Precondition	Users must have to go the home page
Basic course of action	<p>Action</p> <p>Step:</p> <ol style="list-style-type: none"> 1. The admin/guest goes to the census home page 2. The admin/guest clicks the census report option from the menu 3. The admin/ guest selects different census report 4. The admin/ guest can see the census report
Post condition	The user has seen the view census report
Exit condition	The user view gives a census report

Table 2 view description

Use case scenario to fill census form

Use case name	fill census form
Use case ID	UC-3
Participating Actors	Enumerator
Precondition	The enumerator first logs in and click the fill form button
Basic course of action	<p>Action</p> <p>Step:</p> <ol style="list-style-type: none"> 1. User login to the system as enumerator 2. Enumerator select fill census form option from enumerator page 3. The enumerator fills that form

	4. enumerator clicks one Register form button
Postcondition	The actor fills out a census form
Exit condition	The enumerator can now populate the census

Table 3 for fill census form description

Use case scenario to fill house form

Use case name	fill house form
Use case ID	UC-4
Participating Actors	Enumerator
Precondition	The enumerator first logs in and click the fill house button
Basic course of action	Action Step: 1. User login to the system as enumerator 2. Enumerator select fill house form option from enumerator page 3. The enumerator fills that form 4. enumerator clicks one Register form button
Postcondition	The actor fills out a house form
Exit condition	The enumerator can now populate the house

Table 4 for fill house form description

Use case scenario for Search

Use case name	Search
Use case ID	UC-5
Participating Actors	Administrator, Supervisor
Precondition	The administrator or supervisor must log in
Basic course of action	Action Step: 1. Administrator/supervisor login to the system 2. Administrator/supervisor select one of the options from the menu 3. The administrator/supervisor inserts the keyword 4. The system automatically search
Post condition	The Administrator & Supervisor can now search the census and get the searched result
Exit condition	Admin & Supervisor log out from the system

Table 5 search description

Use case scenario for Manage account

Use case name	Manage account
Use case ID	UC-6
Participating Actors	Supervisor, Administrator, Enumerator and Guest
Precondition	The actor must login to the system.
Basic course of action	<p>Action</p> <p>Step:</p> <ol style="list-style-type: none"> 1. Supervisor, Administrator login to the system 2. Supervisor, Administrator Enumerator and Guest click manage button 3. Supervisor Administrator, Enumerator and Guest update account
Post condition	The Supervisor, Administrator, Enumerator and Guest is now updating account
Exit condition	The Supervisor, Administrator, Enumerator and Guest logout from the system

Table 6 manage account description

Use case scenario for Manage user account

Use case name	Manage user account
Use case ID	UC-7
Participating Actors	Administrator
Precondition	The Administrator must login to the system.
Basic course of action	<p>Action</p> <p>Step:</p> <ol style="list-style-type: none"> 1. Administrator login to the system 2. The Administrator click on manage supervisor/enumerator button 3. Administrator click create, update, delete and activate /deactivate button
Post condition	The Administrator create, update, delete and activate /deactivate supervisor/ enumerator account
Exit condition	Administrator logout from the system

Table 7 manage user account description

Use case description for Generate report

Use case name	Generate report
Use case ID	UC-8
Participating Actors	Supervisor
Description	How the supervisor generates the report
Precondition	supervisor must login to system
Basic course of action	Action Step : <ol style="list-style-type: none"> 1. Supervisor initiate login 2. The system display login form 3. Supervisor enters email and password. 4. The system display menus. 5. Supervisor click generate report button and report criteria. 6. The system generates report based on the criteria.
Post condition	The supervisor generate report
Exit condition	The supervisor logout from the system

Table 8 manage user account description

Use case scenario for Manage enumerator account

Use case name	Manage enumerator account
Use case ID	UC-9
Participating Actors	supervisor
Precondition	The supervisor must login to the system.
Basic course of action	Action Step: <ol style="list-style-type: none"> 1. Supervisor login to the system 2. The Supervisor click on manage enumerator button 3. Supervisor click create, update, delete and activate /deactivate button
Post condition	The Supervisor create, update, delete and activate /deactivate enumerator account
Exit condition	Supervisor logout from the system

Table 9 manage enumerator account description

Use case scenario view census

Use case name	view census
Use case ID	UC-10
Participating Actors	Administrator, Supervisor
Precondition	The administrator or supervisor must log in
Basic course of action	Action Step: 1. Administrator/supervisor login to the system 2. Administrator/supervisor click on view census 3. The system automatically display the census
Post condition	The Administrator & Supervisor can now see the census
Exit condition	Admin & Supervisor log out from the system.

Table 10 View census description

Use case scenario for view feedback

Use case name	View feedback
Use case ID	UC-11
Participating Actors	Administrator
Precondition	The administrator must log in
Basic course of action	Action Step: 1. Administrator login to the system 2. Administrator click on view feedback 3. The system automatically display the user feedback
Post condition	The Administrator can now see the user feedback
Exit condition	Administrator log out from the system.

Table 11 View census description

Use case scenario for Give feedback

Use case name	Give feedback
Use case ID	UC-12
Participating Actors	Supervisor, Enumerator and guest
Precondition	The Supervisor, Enumerator and guest must log in

Basic course of action	Action Step: <ol style="list-style-type: none"> 1. Supervisor, Enumerator and guest login to the system 2. Supervisor, Enumerator and guest click on Give feedback 3. The system display the dialog box 4. Supervisor, Enumerator and guest write on the dialog box and click submit button 5. The system send feed back to the administrator
Post condition	The Supervisor, Enumerator and guest can now send the feedback
Exit condition	Supervisor, Enumerator and guest log out from the system.

Table 12 View census description

Use case scenario for logout

Use case name	logout
Use case ID	UC-13
Participating Actors	All users
Precondition	The users must logout
Basic course of action	Action Step: <ol style="list-style-type: none"> 1. users click on this profile button of the system 2. Users click logout button 3. The system display the home page of the system
Post condition	The users logout from system

Table 13 View census description

3.6 Analysis level Class Diagram (Conceptual Modelling)

Class models show the classes of the system, their interrelationships (including inheritance, aggregation, and association), and the operation and attributes of the class. It shows the static structure of the model, in particular, the things that exist (such as classes and types), their internal structure, and their relationships to other things. This project used the class diagram to design the structures that will be included in the system and the things that will exist in the system.

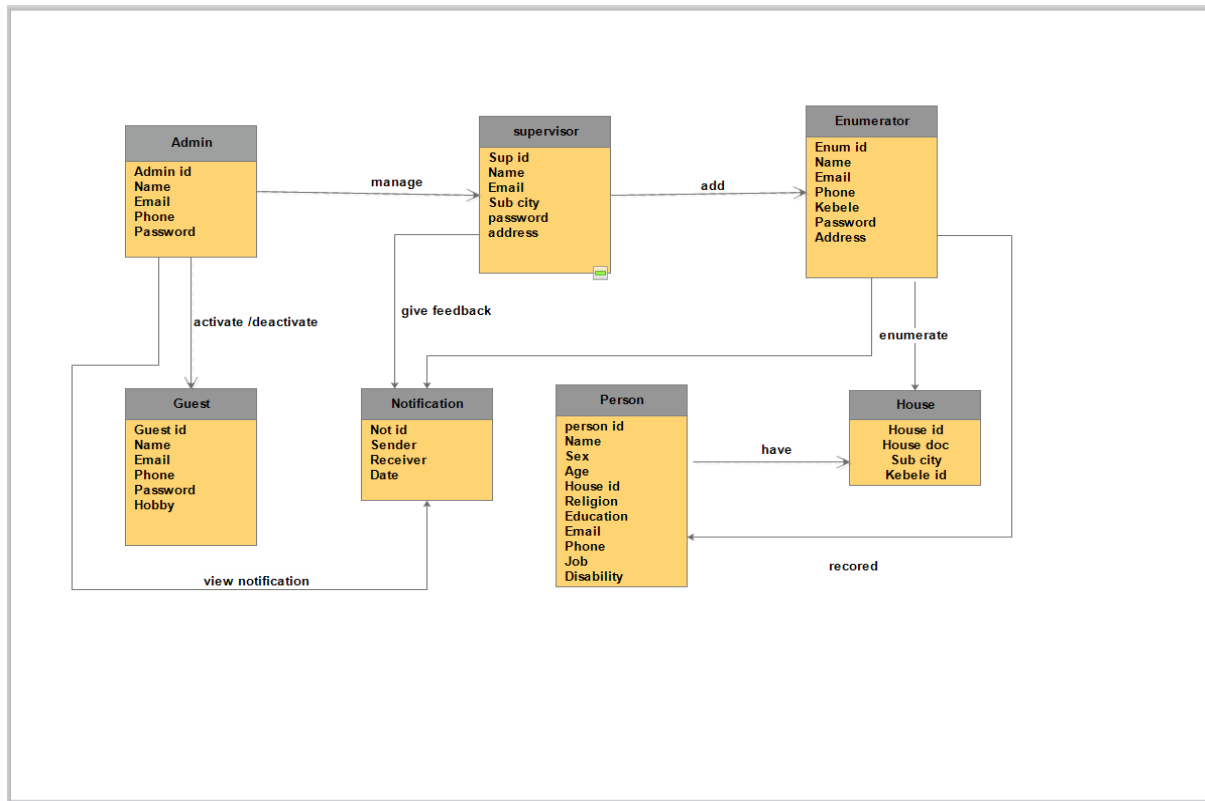


Figure 5: conceptual modeling

3.7 Sequence diagram

Sequence diagrams are used to show how objects interact in a given situation. An important characteristic of a sequence diagram is that time passes from top to bottom: the interaction starts near the top of the diagram and ends at the bottom. A popular use for them is to document the dynamics in an object-oriented system.

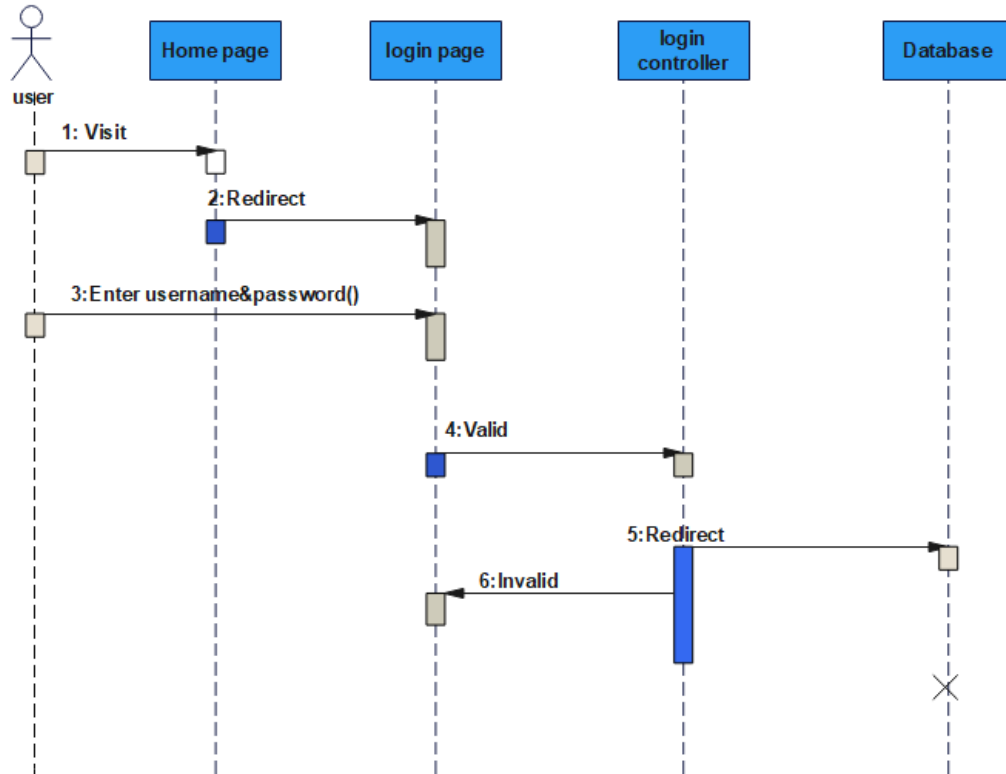


Figure 6: Sequence diagram for login

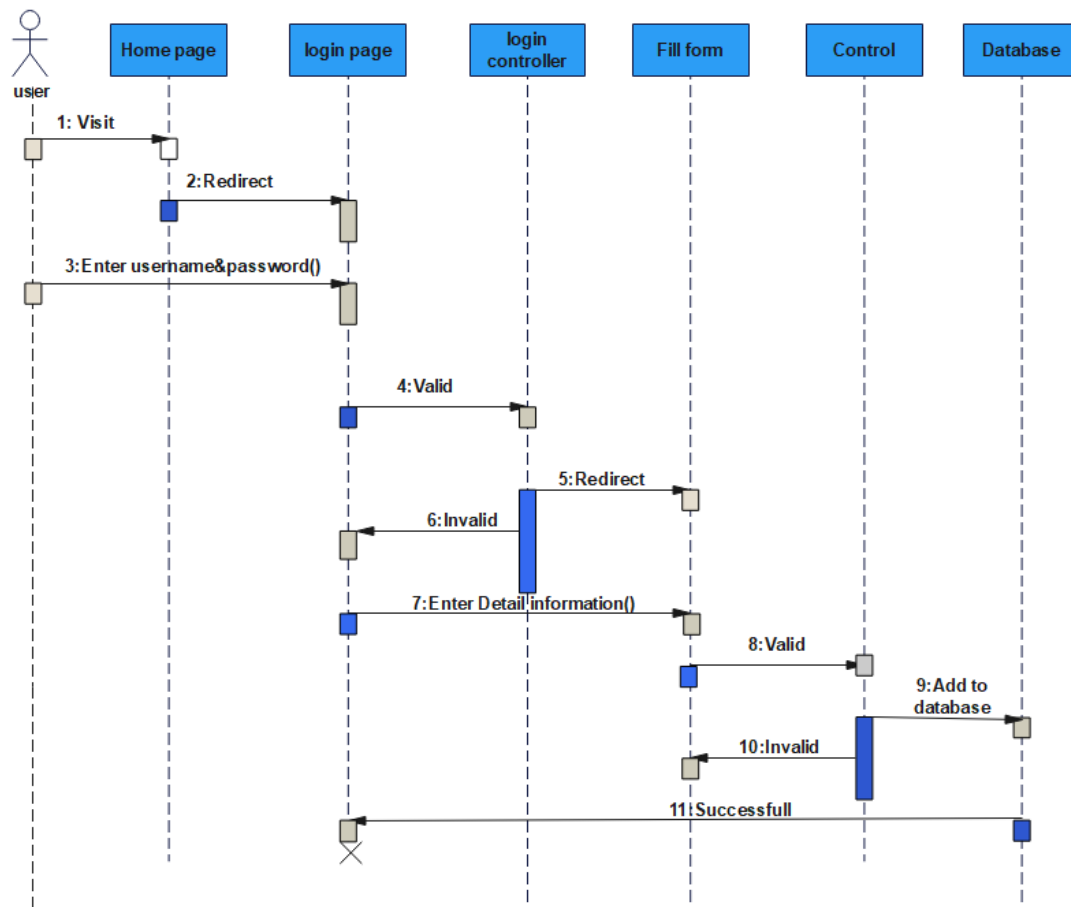


Figure 7: Sequence diagram for fill form

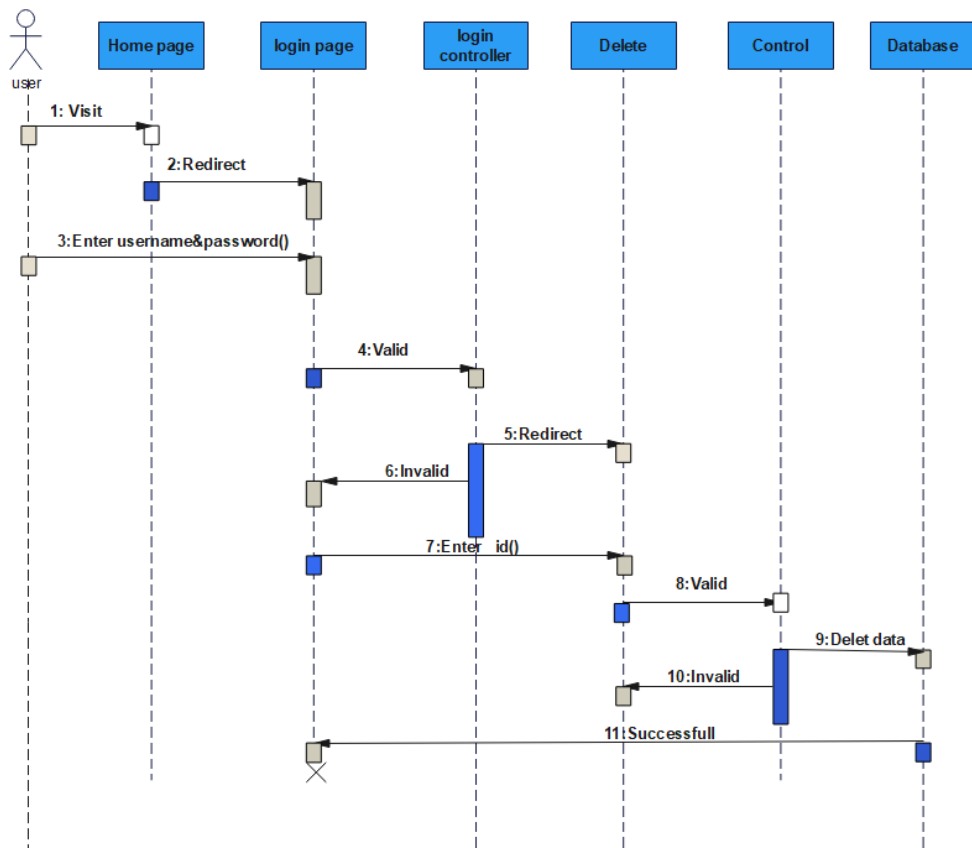


Figure 9: Sequence diagram for delete

3.8 Activity diagram

Activity diagrams are typically used for business process modeling, for modeling the logic captured by a single use case or usage scenario, or for modeling the detailed logic of a business rule. Although UML activity diagrams could potentially model the internal logic of a complex operation it would be far better to simply rewrite the operation so that it is simple enough that you don't require an activity diagram. In many ways, UML activity diagrams are the object-oriented equivalent of flow charts and data flow diagrams (DFDs) from structured development

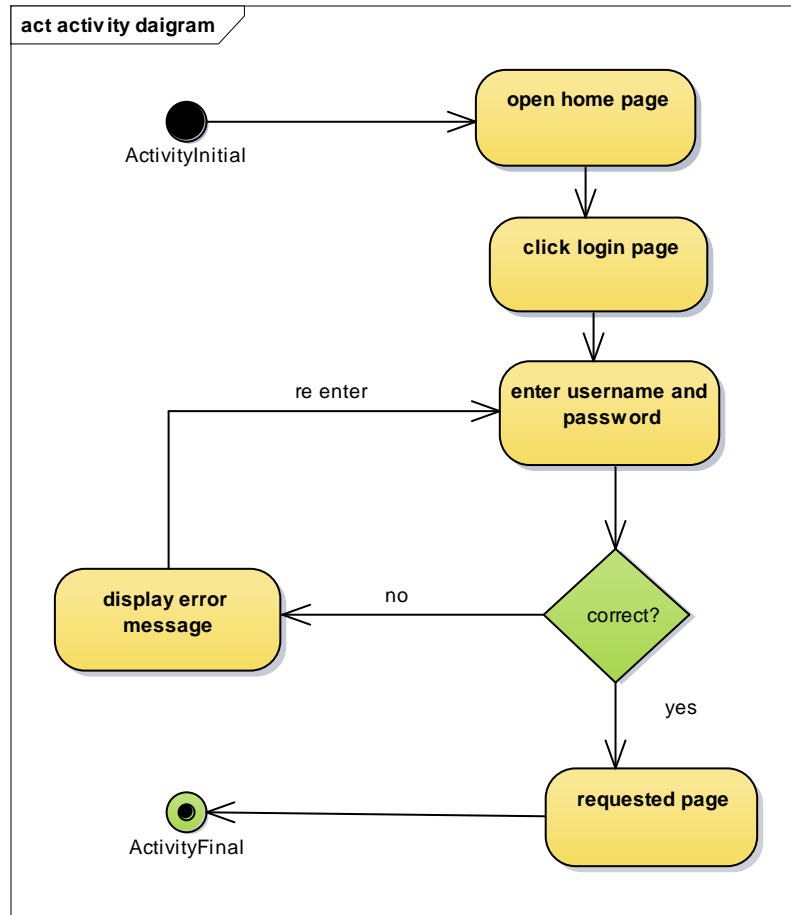


Figure 10: Activity diagram for login

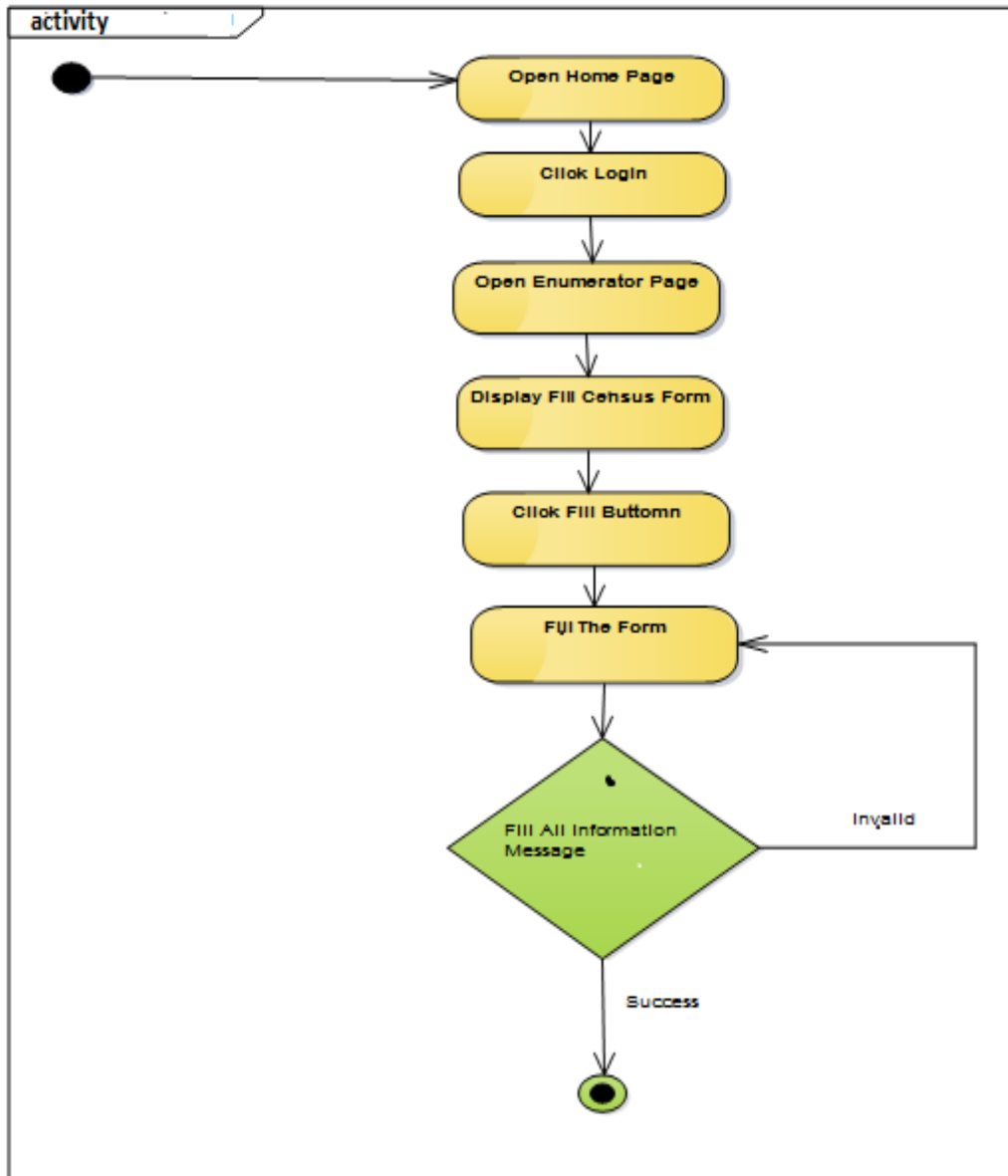


Figure 11: Activity diagram for fill

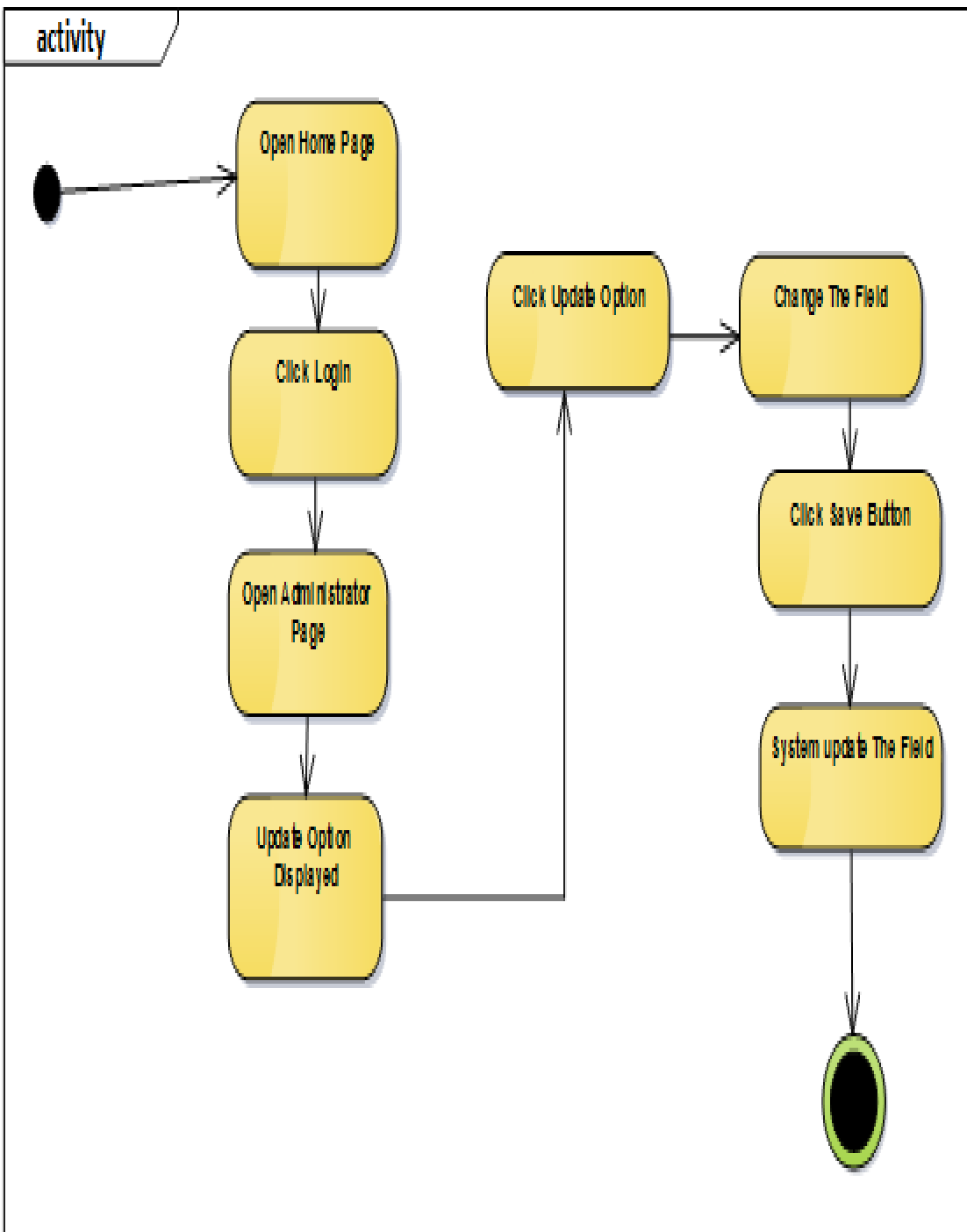


Figure 12: Activity diagram for the update

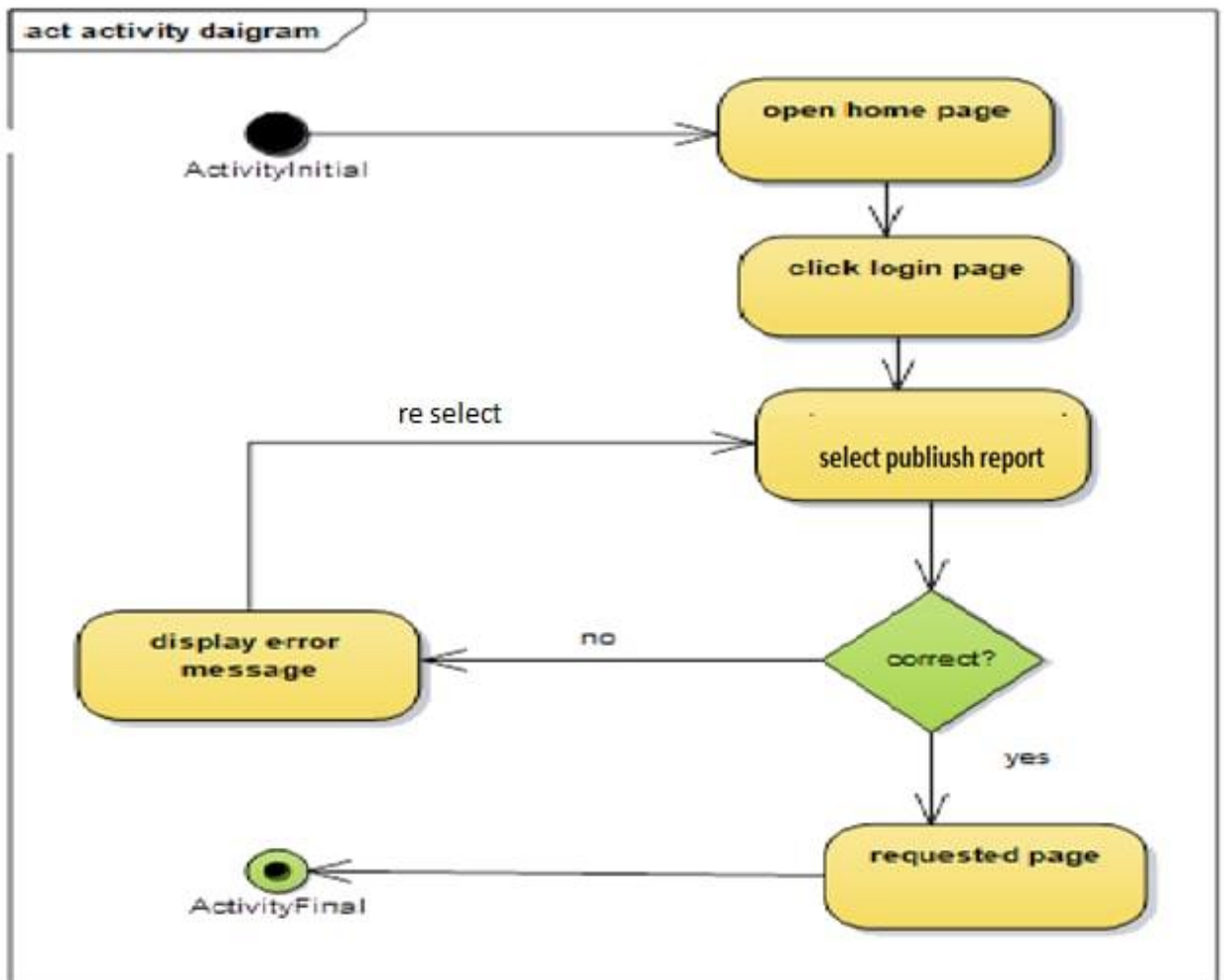


Figure 13: Activity diagram for publishing

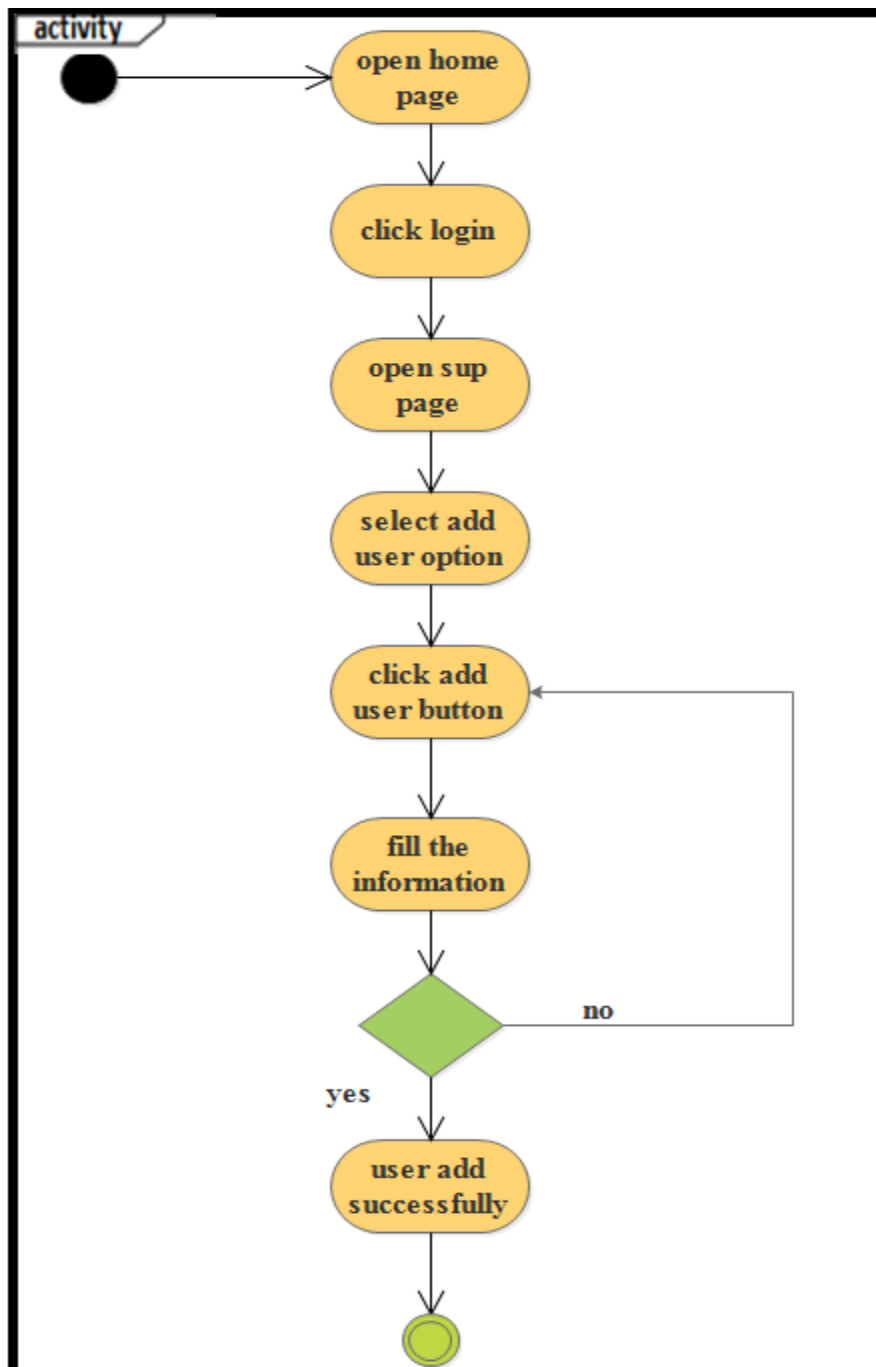


Figure 14; Activity diagram for adding actor

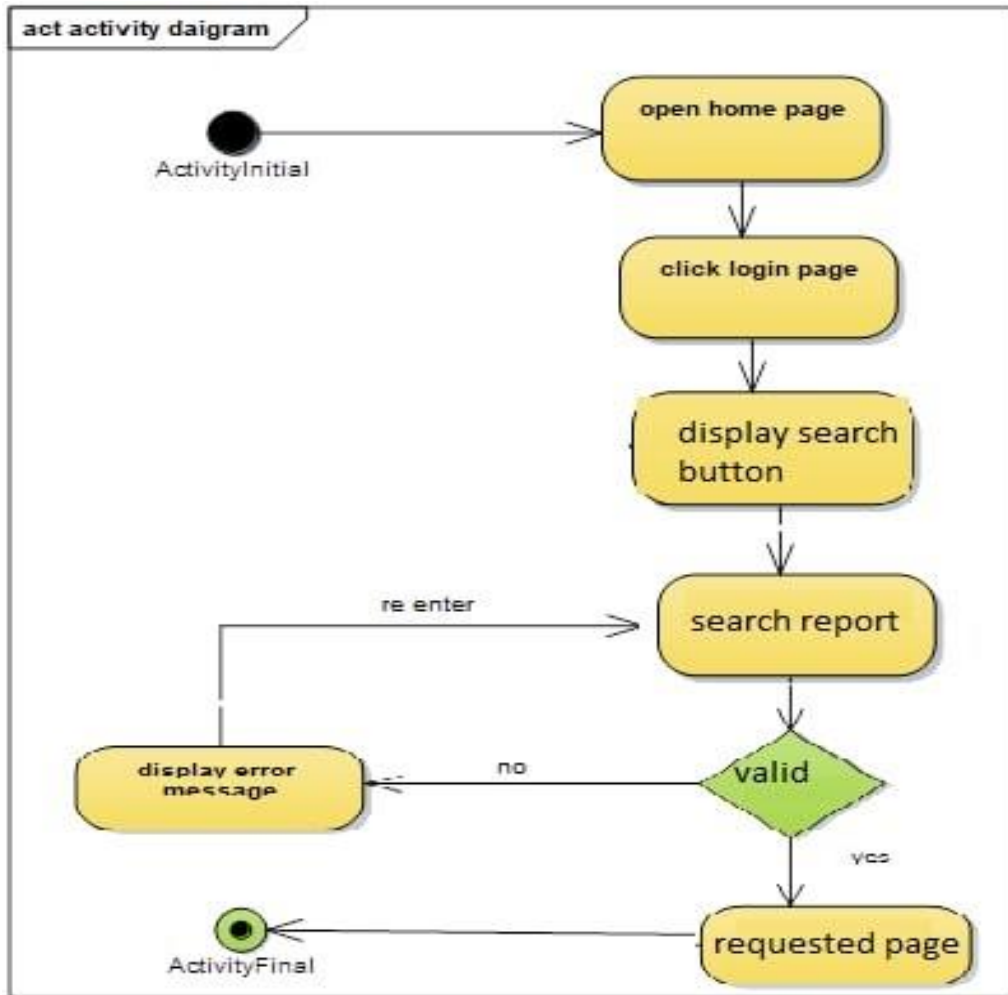


Figure 15: Activity diagram for search

3.9 State chart diagram

State-chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. Therefore, the most important purpose of the State-chart diagram is to model the lifetime of an object from creation to termination. The main purpose of using a state -chart includes modelling dynamic aspects of a system, to a model lifetime of an object, and to describe different states of an object during its lifetime

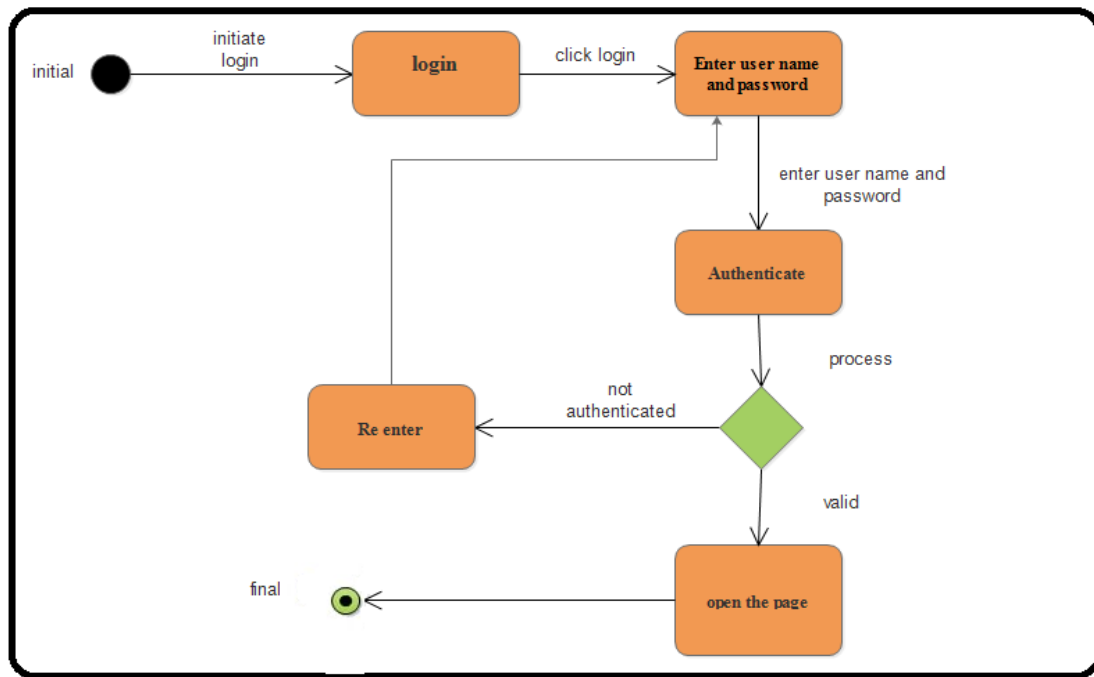


Figure 16: State diagram for login

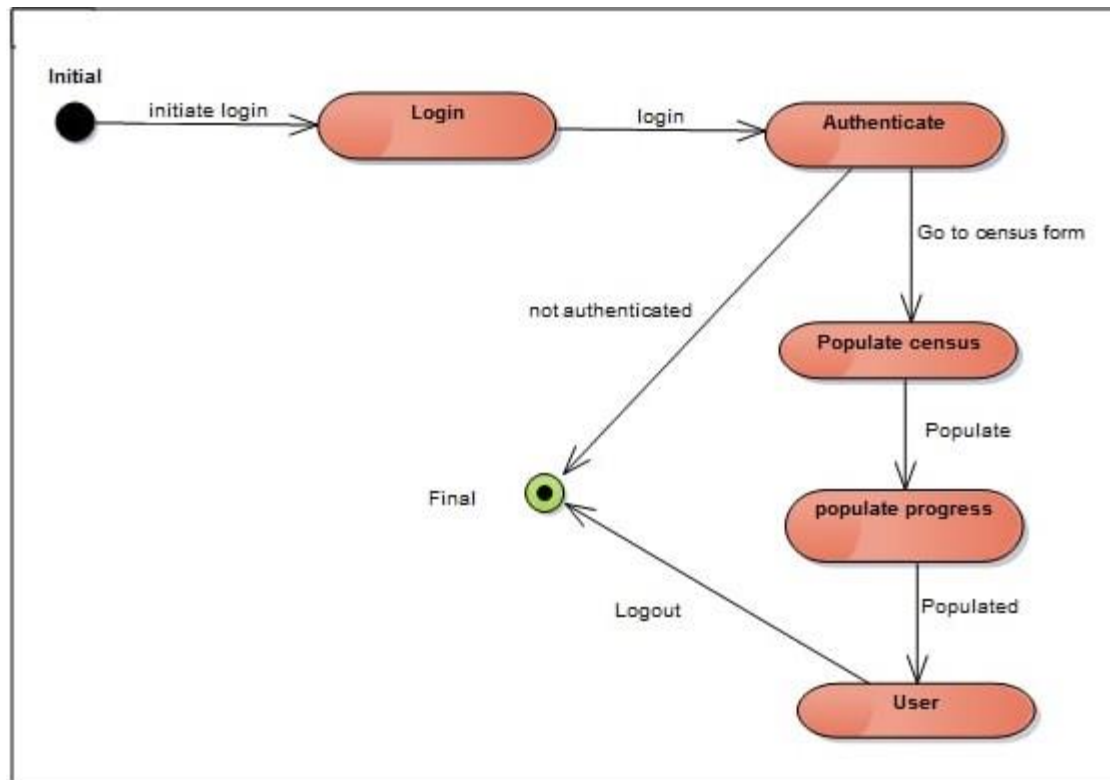


Figure 17: State diagram for populate

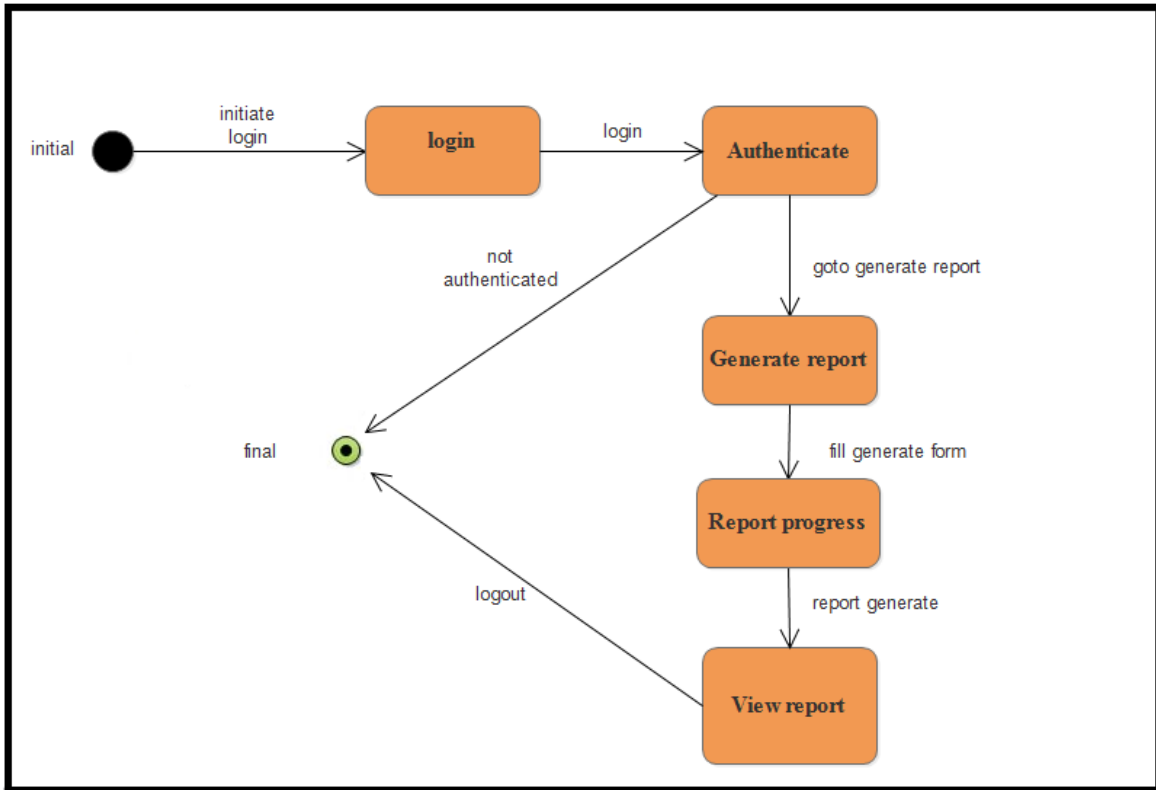


Figure 18: State diagram for generating the report

3.11 User Interface Prototyping

For the user interface below the menus are displayed as the user role. The following shows the navigation path for the menu items.

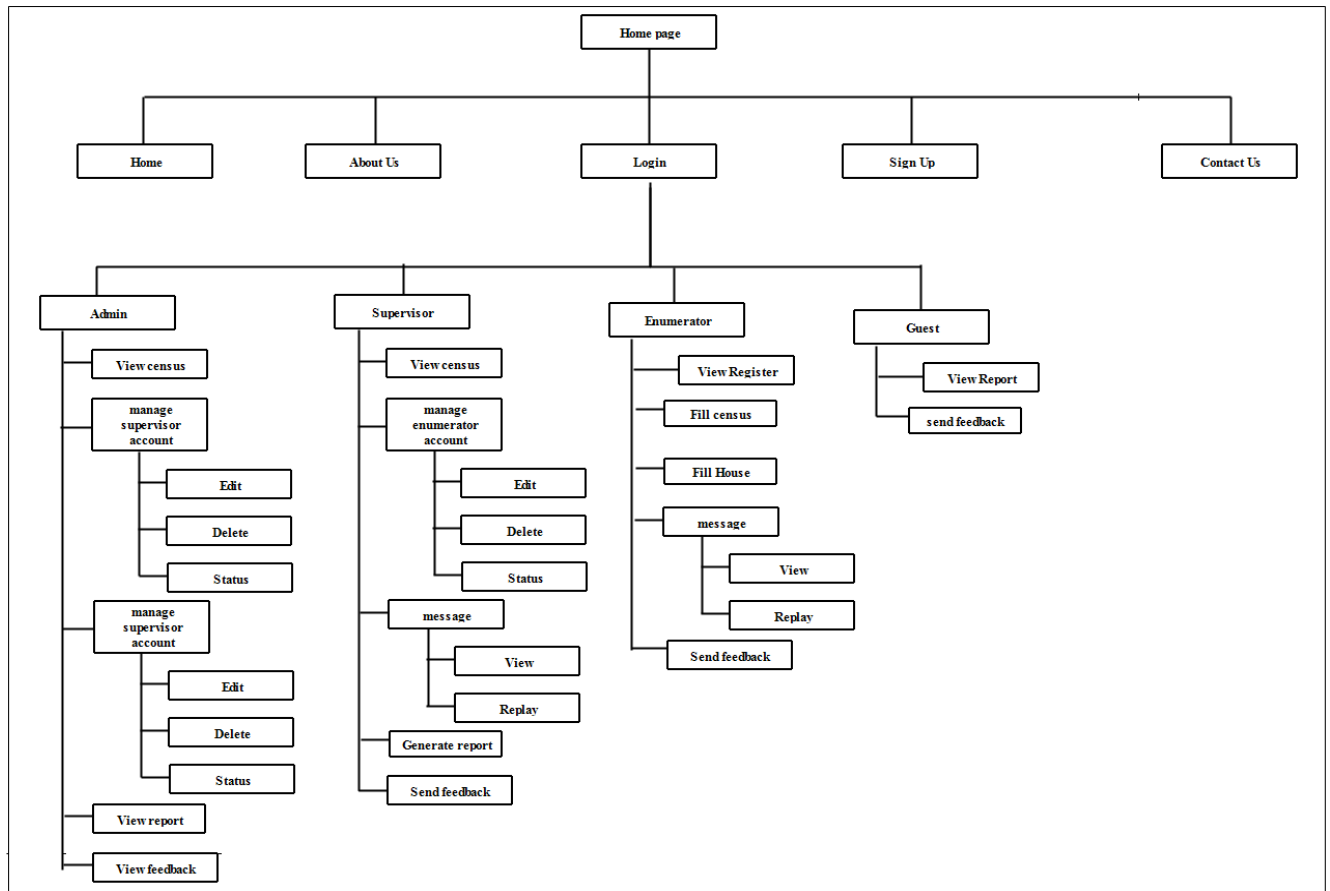


Figure 19: user interface prototype

3.12 Supplementary specifications

3.12.1 Business Rules

BR1: Administrator creates Supervisor account

BR2: Supervisor creates Enumerator's account

BR3: Unregistered Supervisor or Enumerator has no access to those privileges

BR4: Admin, Supervisors & Enumerators have to own a unique username & password to enter the system

BR5: The Admin, Supervisors & Enumerators have to use the system and input data carefully & responsibly

3.12.2 Constraints

Internet: - This system needs an internet connection to be accessed.

Time: As we take other courses in addition to the project, the workload may occur which results in time constraints.

Technical problems: Our knowledge and experience in project work are not that much sufficient and we kill more time studying and understanding some references and supportive courses about project work.

CHAPTER FOUR

SYSTEM DESIGN

4.1 Introduction

The purpose of designing is to show the direction of how the system is built and to obtain clear and enough information needed to drive the actual implementation of the system. It is based on the understanding of the model the software is built on. System design is a phase of creativity rules where customer requirements, business needs, and technical considerations all come together in the formulation of the system. Design is the first step in the development phase. The objectives of the design are to model the system with high quality. Implementing a high-quality system depends on the nature of the design created by the designer. Generally, this chapter describes how the project is designed, what tasks to be done under the project, and different modules and their way of functioning.

4.2 Purpose of the system

Software Design Specification Document contains design-level statements regarding the system to be developed. This document is a detailed description of the system design for the online human population census system. i.e, the modules in the system and how they integrate and relate to each other. It is going to act as a guide for the system developer for it describes how system requirements contained in the requirement specification document are carried out or implemented in the system. The document would also show how users would visualize the system and detailed descriptions of the inputs, processes, and anticipated outputs of specific sub-parts of the system. System Design Specification (SDS) is a description of how the anticipated functional and non-functional system requirements specified in the Software Requirements Specifications (SRS) are achieved in the system. This document generally includes the design of the use case, sequence, and activity diagrams.

4.3 Design goal

The design goal can be inferred from non-functional requirements which will be discussed as follows

- **Reliability:** Reliability is "the probability that a system will perform a required function, under stated conditions, for a stated time". Our system is reliable to provide reliable service for the user.
- **Usability:** The system that we develop should be easy to learn and operate. The system interface should be user-friendly.
- **Manageability:** it is easy for system administrators to manage the application, usually through sufficient and useful instrumentation exposed for use in monitoring systems and for debugging and performance tuning.
- **Maintainability** is the ability of the system to change with a degree of ease. These changes could impact components, services, features, and interfaces when adding or changing the functionality, fixing errors, and meeting new business requirements. our system can be maintained easily if any change has happened.
- **Performance:** Performance is an indication of the responsiveness of a system to execute any action within a given time interval. It can be measured in terms of latency or throughput. Latency is the time taken to respond to any event. Throughput is the number of events that take place within a given amount of time.
- **Security:** our system prevents malicious or accidental actions outside of the designed usage, and prevents disclosure or loss of information. A secure system aims to protect assets and prevent unauthorized modification of information.
- **Robustness:** If there is an error in one module then it does not affect the remaining part of the software.
- **Correctness:** The system result should be pure and accurate.

4.4 System Architecture

The proposed system is developed to replace the current Manual dominant dominance system in a fast and efficient way. In this case, it uses three tier architecture. Three tiers architecture is an architecture in which the functional process logic, data access, computer data storage, and user interface are developed and maintained as independent modules on separate platforms.

Those 3 tiers are the presentation layer (UI), Service/application tier or business logic (BL), and data tier (DB).

The presentation layer (UI): -this layer is the top-level part and displays information related to services available on the server. This is the user interface displayed when the user opens the application.

The Service/application tier or business logic (BL): - is the middle tier that is pulled from the user interface. It performs controlling the application functionalities by performing detailed processing. The main responsibility of this tier is connecting the user interface and the database.

The data tier (DB): - is a house for database servers where information is stored.

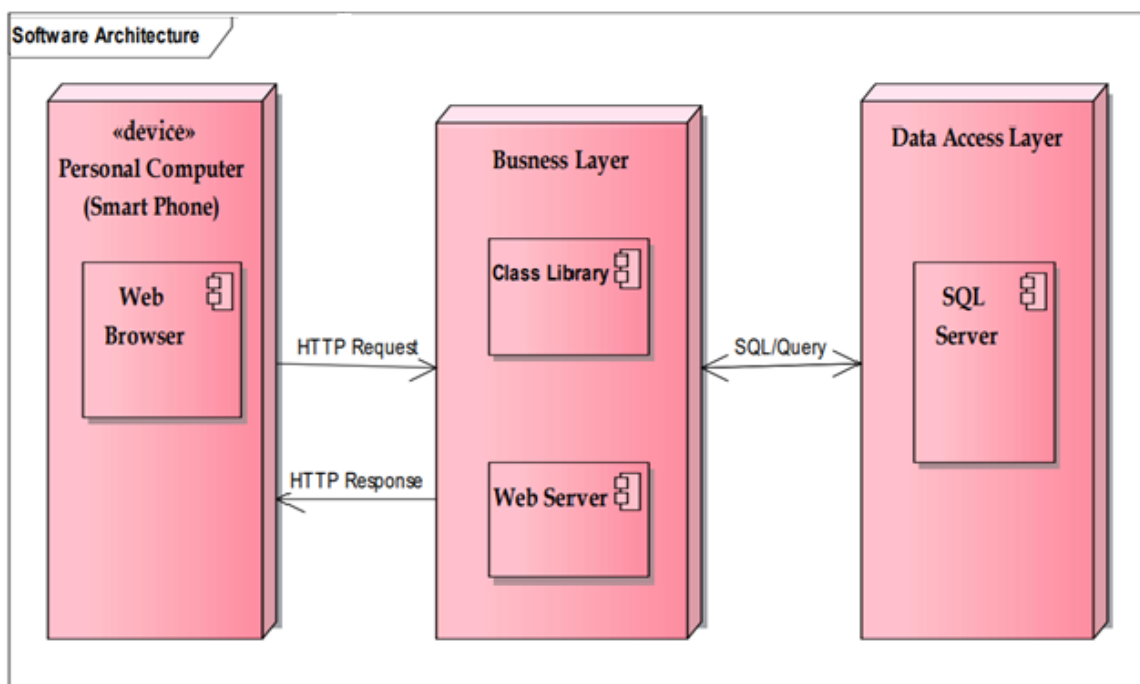


Figure 20: Software Architecture

4.5 Class Modelling

To store information persistently, we map objects into tables and the attributes into fields to the specific table based on the objects found in the system. Therefore, we identified four

major tables that will be implemented in the system. For this reason, the mapping of objects to tables is displayed as follows:

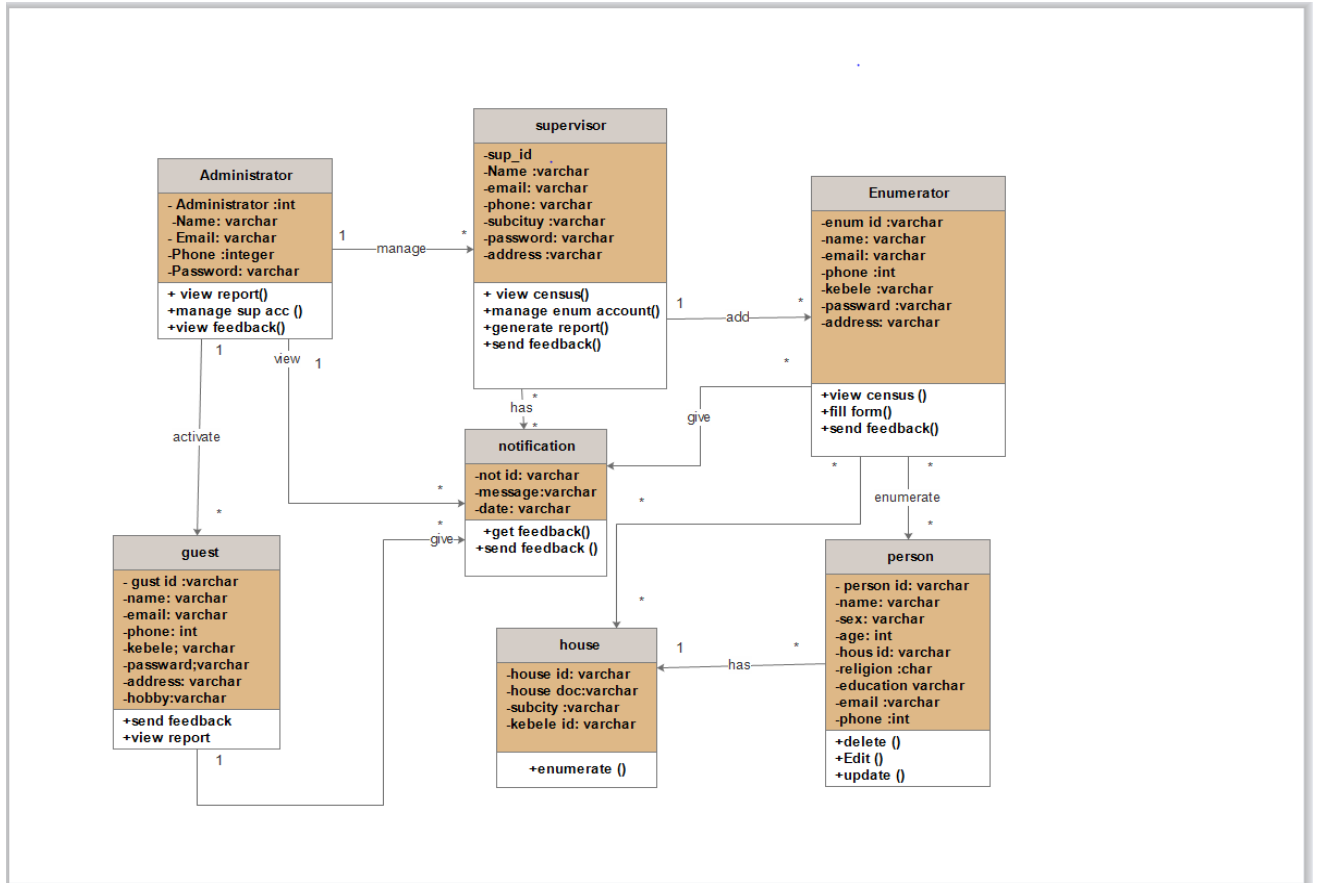


Figure 21: Class modeling diagram

4.6 Component Diagram

The component diagram depicts the components of the system and their interaction. The component diagram illustrates actors and their corresponding activities.

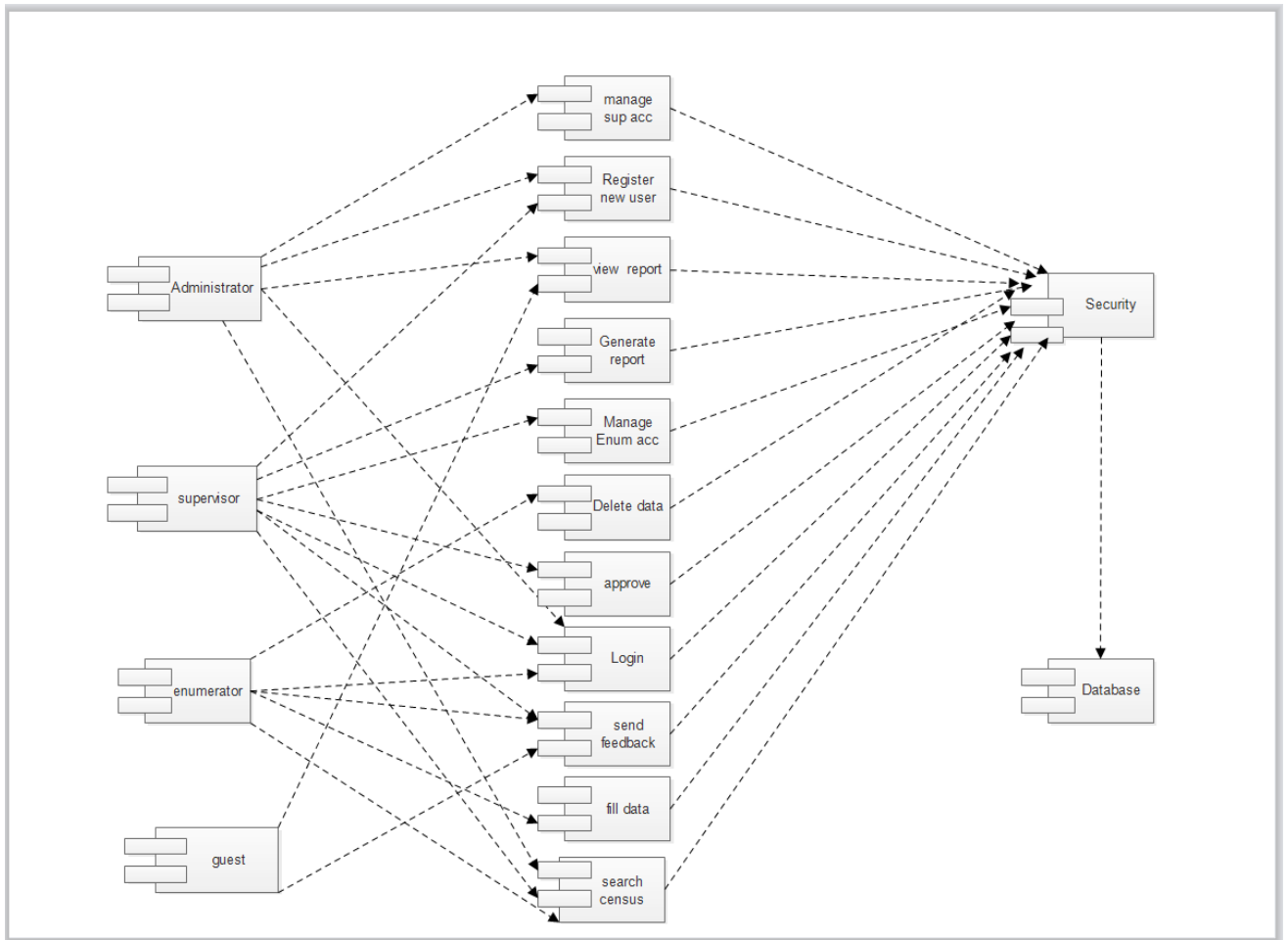


Figure 22: component diagram

4.7 Subsystem decomposition

System decomposition is undertaken to reduce the complexity of the system and gain insight into the identity of the constituent components. The system is decomposed into sub-systems which are a collection of classes, associations, operations, events, and constraints that are closely interrelated with each other.

The major subsystems of the Online Population census system are described as follows: -

Administrator: an employee, who works in the census service office, which has the following responsibilities.

- Add/ Manage Supervisor account.
- Search report.

- View feedback.
- View Report.

Supervisor: an employee in the census system, who is responsible to generate the report and add enumerators.

- Add/ Manage Enumerator account.
- Generate report.
- Search Census Data
- Send feedback

Enumerator: A professional person who performs counting functions such as enumeration of persons and housing units and filling other census forms.

Guest: A customer, who want the information about record person in the census. they view the report and send feedback.

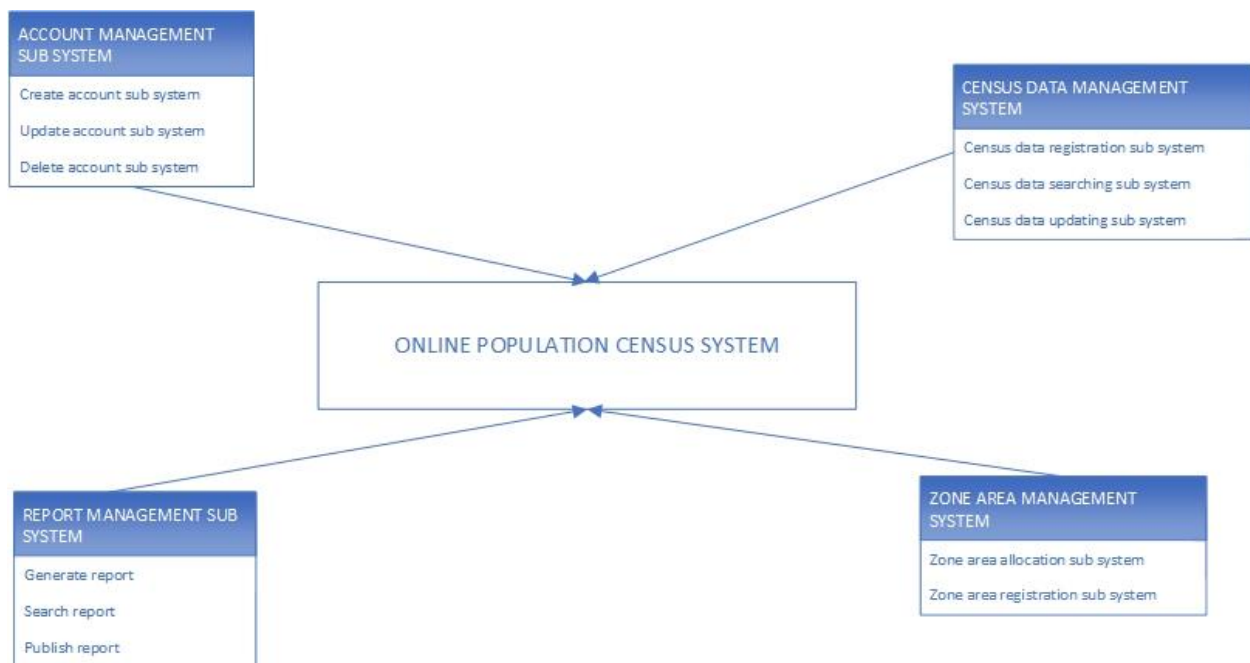


Figure 23: Subsystem decomposition diagram

4.8 Deployment Diagram

Deployment modelling is used to show the hardware of the system, the software that is installed in the hardware, and also the middleware that is used to connect the disparate machines to one, and the other.

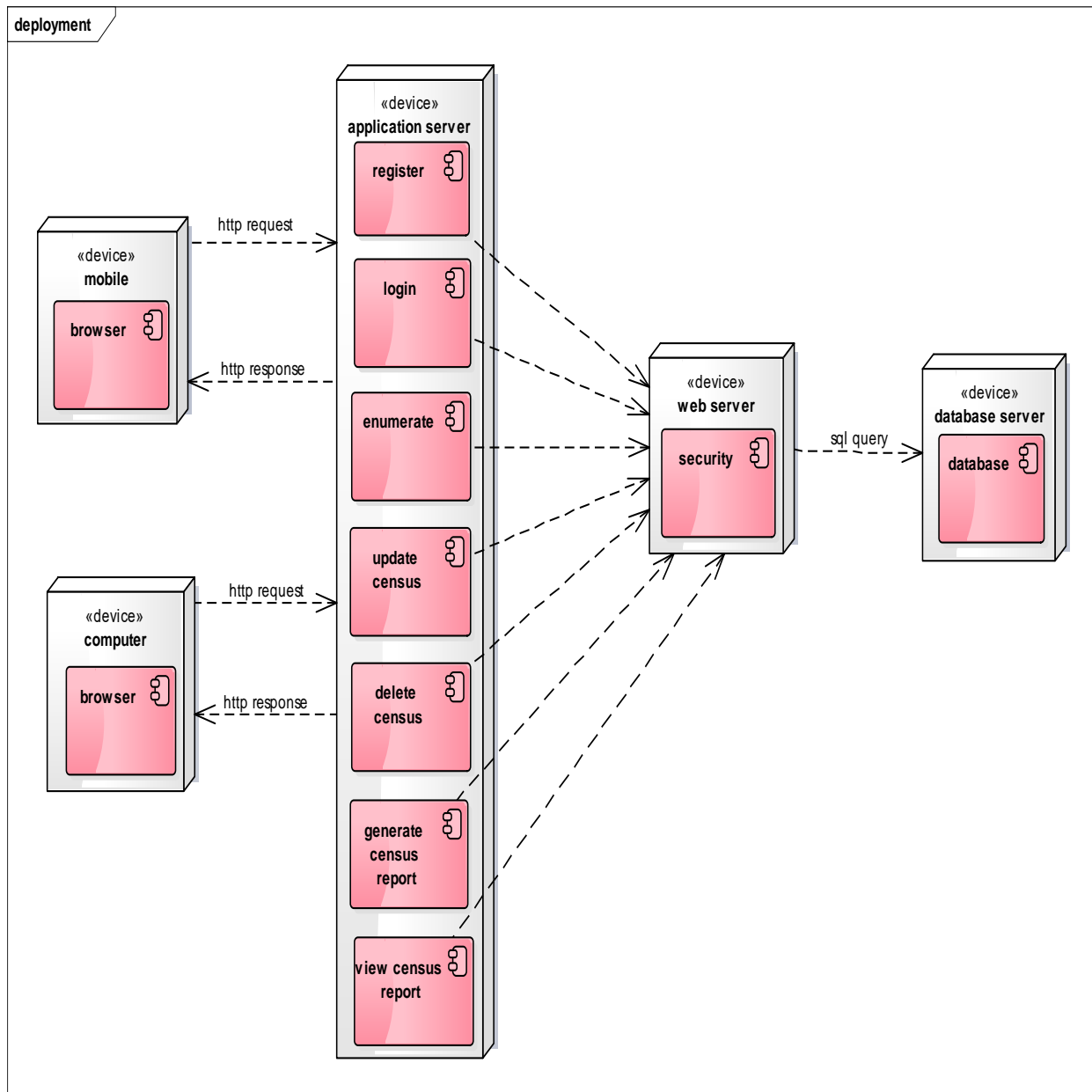


Figure 24: Deployment diagram

4.9 Data Dictionary

This gives a brief description of the field names used in the tables and what they define as per the databases.

Administrator Table

Field name	Data type	Size	Description
Administrator_name	char	20	Holds the name of the Administrator
Administrator ID	Varchar	20	The field for the identifier of the Administrator
first_name	char	20	Holds the First name of the Administrator
Last_name	char	20	Holds the last name of the Administrator
Address	varchar	10	Holds the contact of the Administrator
email	varchar	20	Highlights the email of the Administrator
Phone_no.	Int	15	Field holds the phone number of the Administrator
Password	Varchar	20	Holds the password of the Administrator

Table 14 Administrator table

Supervisor Table

Field name	Data type	Size	Description
supervisor_name	char	20	Holds the name of the Supervisor
Supervisor_id	varchar	20	The field for the identifier of the Supervisor
first_name	char	20	Holds the First name of the Supervisor
last_name	char	20	Holds the last name of the Supervisor
Address	varchar	10	Holds the contact of the supervisor
email	varchar	20	Highlights the email of the supervisor

Phone_no.	Int	15	Field holds the phone number of the supervisor
Password	Varchar	20	Indicates the password

Table 15 supervisor table

Enumerator Table

Field name	Data type	Size	Description
Enumerator_name	char	20	Holds the name of the Enumerator
Enumerator ID	varchar	20	The field for the identifier of the Enumerator
first_name	char	20	Holds the First name of the enumerator
Last_name	char	20	Holds the last name of the enumerator
Address	varchar	10	Holds the contact of the Enumerator
email	varchar	20	Highlights the email of the Enumerator
Phone_no.	Int	15	Fields hold the phone number of the Enumerator
Password	Varchar	20	Holds the password of the Enumerator

Table 16 enumerator table

Census Data

Field Name	Data type	Size	Description
Location	Varchar	20	Indicates the location of household
Address	Varchar	20	Indicates the Address for household
Age	Int	20	Indicate the age of residents
Place_of_birth	Varchar	20	Indicates the birthplace of residents
Religion	char	20	Indicates the religion of residents

Marital_status	char	10	Indicates the marital status of the residence
Orphan_hood	Varchar	20	Indicates the orphan in the residence
Previous_residence	Varchar	20	Indicates the previous residence of residents
Nationality	char	20	Indicates the nationality of residents

Table 17 census data table

4.10 Database Design

Database modeling is used to communicate the design of the database, usually the database to both the users and the developers. It is also used to describe the persistence data aspect of the system. The following diagram indicates the database design in a diagram of the system.

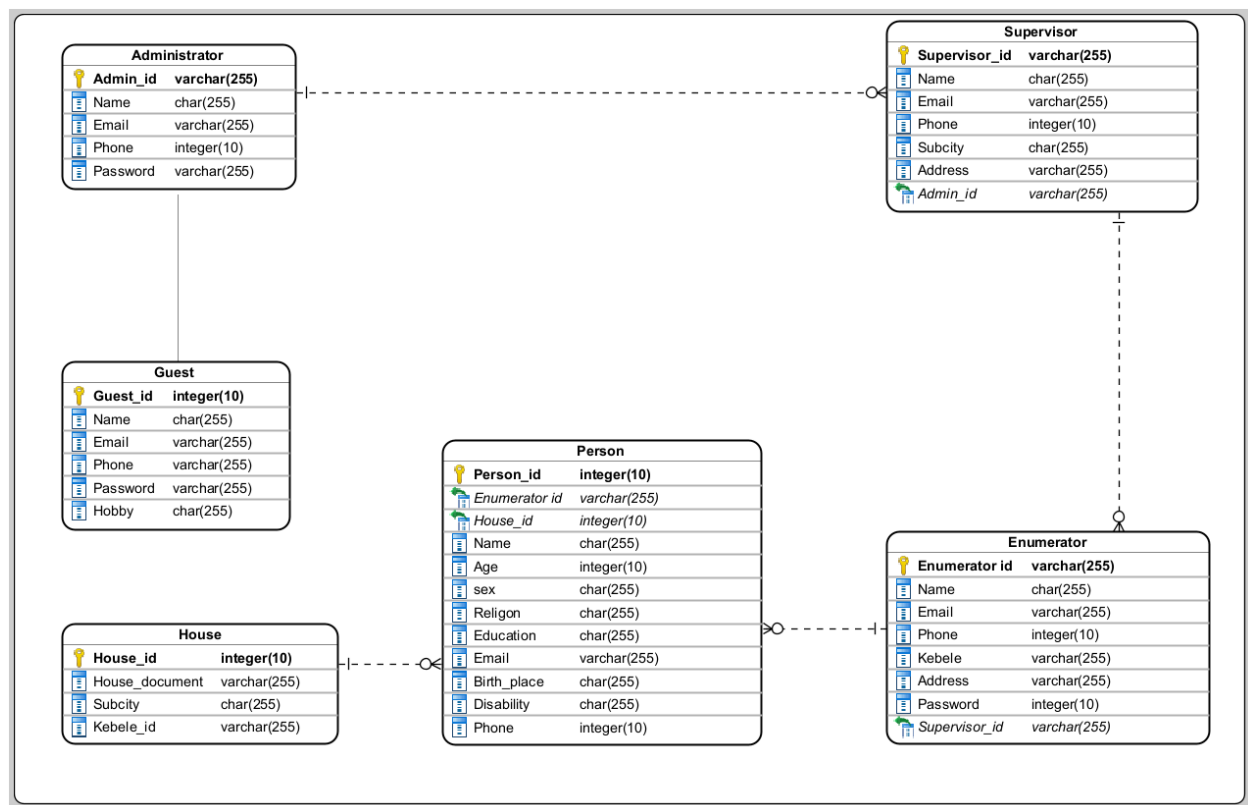


Figure 25: database design

CHAPTER FIVE

IMPLEMENTATION

5.1 Introduction

The implementation phase is the most crucial phase in which it transform the design and analysis of the system into a tangible system by writing the code to the system to be developed and make it operational and applicable by testing and debugging the functionalities that are done. This makes the implementation stage more essential step to develop the required system. So, it is the most vital and necessary stage in achieving a successful system and in giving the users confidence that the new system will work and be effective by testing the system that is already implemented.

5.2 Hardware software acquisitions

For the project implementation; we used the following Software and hardware tools.

Hardware Requirement

- Desktop Computer
- Flash disk 8GB
- Compactable CD-ROM 700MB
- Laptop

Software Requirements

- Notepad++
- Microsoft word 2013
- Microsoft PowerPoint 2013
- Xampp Server
- Browsers (Mozilla Firefox, Google chrome)

5.3 Language specification and selection strategy

This project used PHP, CSS, HTML and JS for select the best interface programming technology integrated with xampp server database with a programming language. We choose PHP as server side scripting language because PHP is:

- Clear and easy to understand.
- It runs on many different operating systems.
- Operates much faster than other scripting languages.
- Open source. In general PHP is a widely used open source general purpose programming Language that is especially suited for web development and can be embedded into HTML.

5.4 Sample code



Figure 26 home page interface

```

<!DOCTYPE HTML>

<html>

    <head>

        <title>DB-CMS</title>

        <link href="assets/css/style2.css" rel="stylesheet" type="text/css" media="all" />

        <link href='http://fonts.googleapis.com/css?family=Ropa+Sans' rel='stylesheet'
type='text/css'>

        <link rel="stylesheet" href="assets/css/responsiveslides.css">

        <script
src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"></script>

        <script src="assets/js/responsiveslides.min.js"></script>

        <script src="assets/js/anychart-base.min.js"></script>

        <script src="assets/js/anychart-core.min.js"></script>


        <script>

            $(function () {

                // Slideshow 1

                $("#slider1").responsiveSlides({

```

```

        maxwidth: 1600,

        speed: 600

    });

});

</script>

</head>

<body>

    <!--start-wrap-->

        <!--start-header-->

            <div style="background:#2E55C7" class="header">

                <div class="wrap">

                    <!--start-logo-->

                        <div class="logo">

                            <a href="index.html" ></img></a>

                        </div>

                    <!--end-logo-->

                    <!--start-top-nav-->

                        <div class="top-nav">

```

```

        <ul>

            <li class="active"><a
href="index.html">Home</a></li>

            <li><a href="contact_us.php">contact</a></li>

            <li><a href="about_us.html">About</a></li>

            <li><a href="/cms.php">Login</a></li>

            <li><a href="/signup.php">Signup</a></li>

        </ul>

    </div>

    <div class="clear"> </div>

    <!--end-top-nav-->

</div>

<!--end-header-->

</div>

<div class="clear"> </div>

    <!--start-image-slider----><div style="display:inline-block; height:
400px;width:400px;" id="graph"></div>

    <div style="display:inline;" class="image-slider">

        <!-- Slideshow 1 -->

```

```
<ul style="display:inline;" class="rslides" id="slider1">
```

```
    <li></li>
```

```
    <li></li>
```

```
    <li></li>
```

```
    <li></li>
```

```
    <li></li>
```

```
    <li></li>
```

```
    <li></li>
```

```
</ul>
```

```
<!-- Slideshow 2 -->
```

```
</div>
```


<!--End-image-slider---->

<div class="clear"> </div>

<div class="content-grids">

<div class="wrap">

<div class="section group">

<div class="listview_1_of_3 images_1_of_3">

<div class="listimg listimg_1_of_2">

</div>

<div class="text list_1_of_2">

<h3>Statistical Reports</h3>

<p>view the census report for Each subcity</p>

<div class="button">Click Here</div>

</div>

</div>

<div class="listview_1_of_3 images_1_of_3">

<div class="listimg listimg_1_of_2">

</div>

<div class="text list_1_of_2">

<h3>Guest Login</h3>

<p>Log in as a guest to get census

reports</br></br></br></p>

<div class="button">Click Here</div>

</div>

</div>

<div class="listview_1_of_3 images_1_of_3">

<div class="listimg listimg_1_of_2">

</div>

<div class="text list_1_of_2">

<h3>Staff Login</h3>

</br></br></br></br>

```
        <div class="button"><span><a
href="/cms.php">Click Here</a></span></div>
```

```
    </div>
```

```
  </div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="wrap">
```

```
<div class="content-box">
```

```
<div class="section group">
```

```
    <div class="col_1_of_3 span_1_of_3 frist">
```

```
    </div>
```

```
    <div class="col_1_of_3 span_1_of_3 second">
```

```
    </div>
```

```
    <div class="col_1_of_3 span_1_of_3 frist">
```

```
    </div>
```

```
  </div>
```

```
</div>
```

</div>

<div class="clear"> </div>

<div class="footer">

<div class="wrap">

<div class="footer-left">

Home

contact

</div>

<div class="clear"> </div>

</div>

</div>

<!--end-wrap-->

</body>

</html>

<script>

```

var data = [

["1900", 16018],

["1910", 17816],

["1920", 17816],

["1940", 23303],

["1950", 25130],

["1960", 30951],

["1970", 39936],

["1990", 64573],

["2000", 80371],

["2020", 107424]

];

// create the chart

var chart = anychart.bar();

var series = chart.bar(data);

chart.height('400px');

chart.title("Total population of Debre Berhan Since 1900")

chart.width('100%');

chart.container("graph");

```

```
chart.draw();
```

```
</script>
```

5.5 Test procedures

The System will be tested using the standard and recommended system testing procedures before directly deploying. We will perform different testing methodologies for its Functionality and meeting users need

Unit testing

Verification (A set of operations that the software correctly implemented a particular function) on the smallest element of the program i.e. the modules are tested alone in order to discover any error in the code. Features to be tested:

- ❖ When the field is not filled it displays the message the following message
- ❖ When the field is not correct, the system consume it

Integration testing

In integration testing we integrate and test all components of the system modules that a program comprises for testing purpose.

System testing

System test insures that the entire integrated software system meets requirements. It tests a configuration to insure known and predictable results. System testing is based on process description and flows, emphasizing pre-driven process links and integration points.

We apply System testing to insures the functional and non-functional requirements correctness, performance, accuracy, and security have been met correctly

In general in this testing, the team performs over all functional testing by checking whether it meets the required target or not.

5.6 User manual preparation

User manual preparation is important for the user will be explained and implemented while preparation of short training document when the system is deployed. And also notify the user

whether the teams prepare to give training. But in our system no user manual because it is web based system.

5.6.1 Training

The organization user group might have a different training strategy although there would be a lot in common.

End-user training

End-user training will be provided using separate modules. This system allows the creation of web-based content and provides a step-by-step walk through of the business process.

- ❖ The developer will install the system on the server.
- ❖ The system content will be published on a set schedules using the modules on the server
- ❖ Ensure users have logon access
- ❖ End users will access training materials through web browsers
- ❖ Ensure server has capability to handle usage

5.6.2 Installation Process

Required items to request an agent installation: To request a web based installation, we need to provide the following information:

- ❖ Name of organization requesting the installation
- ❖ Ip address of the server being protected
- ❖ Host name of the servers
- ❖ URL being protected
- ❖ Operating system of the web application server being protected. Give the specific version of windows, specify whether 32/64 bit
- ❖ Type of web application server (Apache)

5.6.3 Start-up Strategy

The start-up strategy will be:

- ❖ Buying a domain name
- ❖ Hosting the system
- ❖ Finally available internet connection

CHAPTER SIX

CONCLUSIONS AND RECOMMENDATION

6.1 Conclusion

We have implemented the secure online census system based on browser-structure. It is our believe that with the adoption of this system prototype and implementation on a large scale, a guaranteed safe, secured and reliable to both the quality and use of data on the population census to encourage cooperation and collaboration among federal agencies by:

- ❖ Adopting the population model it could help the demographers to forecast and advise the government to plans for new generations.
- ❖ Widening access to information on the population census through periodic publications and other means.
- ❖ Promoting communication among data producers, researchers, and public policy-makers.
- ❖ Coordinating the development and use of statistical databases among federal agencies.
- ❖ Identifying information gaps and data inconsistencies.
- ❖ Investigating questions of data quality
- ❖ Encouraging cross-national research and data collection on the census.

6.2 Recommendation

According to scope of our project the group develops web application system for census system. Because of the time constraint we cannot do beyond to our scopes, but in the future the team believes that this system can be fully operational by having enough time and fully information. Census system is difficult project to finish in little time like other project. Generating report takes more time even 1 year cannot be enough to complete it. To reduce these complexities we have included backup system in the form of excel. This may it help full for further calculation of reports.

Finally the team would recommend that further work should be done on the system in order to make the system perform better for interested organizations and the team would like to recommend that the system is open for whom who wishes to add new functionalities and finally the team would like to recommend to Central Statistics Agency of Ethiopia to use this system by enhancing it to the way that it can give full web based census system.

REFERENCE

- G. David Garson, North Carolina state university, USA, Handbook of Research on Public Information Technology volume 1
- [3] Booch, G. (1994). Object-Oriented Analysis and Design. Redwood City: ADDISON-WESLEY.
- [4] Grady Booch, James Rumbaugh, and Ivar Jacobson. (October 20, 1998). The Unified Modeling Language User Guide. Addison Wesley.
- <http://www.census.ie/-in-history/the-census-through-history.150.1.aspx>
- http://www.google.com/industry/diagram/Sequence_Diagram-UML_Diagrams-Unified_Modeling_Language_Tool.html
- http://www.google.com/industry/diagram/UML_Class_Diagram.html

Appendix

Appendix 1

Questions asked during requirement analysis using an interview:

Q1.What is the objectives of the organization?

Q3.How many actors do you have in your system?

Q4. How does your current system work?

- Is it manual?
- Is it computerized?
- Is it semi-computerized?

Q6.What are the procedures or steps?

Q7.Where do you keep the populated person and house information?

Appendix 2

Symbol

Description



Actor



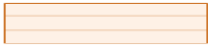
System boundary



Decision



Use case



class



Object lifeline



Deployment diagram



The message line extends from the lifeline of one object to the lifeline of another object.



Include, relates to the include use case to indicate inserted behavior



Create Return Message



The starting point of the activity/state diagram

