

FDRE Vital Events Registration Agency Information
Management System

A Project Report

Submitted By:

| NAME | ID NUMBER |
|-----------------|-------------|
| MAHILET ZEWDU | TER/4740/07 |
| HANNA TILAHUN | TER/4734/07 |
| WULETAW YEHUALA | TER/4761/07 |
| TARIKU BEKELE | TER/4755/07 |

In partial fulfilment for the award of the degree of

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

Under the guidance of

Instructor Wubetu Shiferaw

.....

ADVISOR SIGNATURE



DEPARTMENT OF INFORMATION TECHNOLOGY
COLLEGE OF TECHNOLOGY
DEBRE MARKOS UNIVERSITY
DEBRE MARKOS

January 2010 E.C

Table of content

| | |
|--|----|
| List of Figure..... | 4 |
| List of Table | 5 |
| Acronyms and Abbreviations | 6 |
| Chapter one | 7 |
| Introduction | 7 |
| 1.1 Introduction..... | 7 |
| 1.2 Background of the organization..... | 7 |
| 1.3 Statement of the problem | 8 |
| 1.4 Objective the Project..... | 8 |
| 1.4.1 General Objective | 8 |
| 1.4.2. Specific Objective | 8 |
| 1.5 Scope of the project | 9 |
| 1.6 Overview of the proposed system..... | 9 |
| 1.7 Significance of the project | 10 |
| 1.8 System requirements | 10 |
| 1.8.1 Hardware Requirements..... | 11 |
| 1.8.2 Software Requirements | 11 |
| 1.8.3 Programming language | 11 |
| 1.9 Data Collection Methodology..... | 12 |
| 1.10 Feasibility Study | 12 |
| 1.10.1. Operational feasibility..... | 13 |
| 1.10.2. Economic feasibility | 13 |
| 1.10.3. Legal feasibility..... | 14 |
| 1.10.4 Technical feasibility | 14 |
| Chapter two..... | 16 |
| 2. System analysis..... | 16 |
| 2.1 Overview of the existing system..... | 16 |
| 2.2 System requirement specification | 18 |
| 2.2.1 Functional requirements..... | 18 |
| 2.2.2 Non functional requirements..... | 19 |
| 2.2.2. Technical requirement..... | 21 |

| | |
|--|----|
| 2.2.3. Business rules..... | 21 |
| 2.3. System requirement analysis..... | 22 |
| 2.3.1. Actor and use case identification | 22 |
| 2.3.3. UML Sequence Diagram | 39 |
| 2.3.4 Activity diagram | 45 |
| 2.3.5 Analysis Class Diagram | 50 |
| Chapter three..... | 52 |
| 3. System Design | 52 |
| 3.1. Design class diagram | 52 |
| 3.2. Database design | 54 |
| 3.2.1. Physical data model | 54 |
| 3.2.2. Normalized table | 54 |
| 3.3 system architecture..... | 60 |
| 3.3.1. Deployment diagram..... | 60 |
| References..... | 61 |

List of Figure

| | |
|--|-----------|
| Figure 1: Use Case Diagram | 24 |
| Figure 2: sequence diagram for login | 40 |
| Figure 3: sequence diagram for registering events | 41 |
| Figure 4: sequence diagrams for updating events. | 42 |
| Figure 5: sequence diagram for the calculating population size | 43 |
| Figure 6: sequence diagram for view events..... | 44 |
| Figure 7: Sequence diagram for create user account | 45 |
| Figure 8: Activity diagram for register events | 46 |
| Figure 9: activity diagrams for update event | 47 |
| Figure 10: Activity diagram for calculating population size | 48 |
| Figure 11: Activity diagram for view event..... | 49 |
| Figure 12: Activity diagram for create account | 50 |
| Figure 13: Analysis class diagram | 51 |
| Figure 14: Design class diagram..... | 53 |
| Figure 15: Home page of the system | 58 |
| Figure 16: registrars page of the system | 59 |
| Figure 17: event registration forms | 59 |

List of Table

| | |
|--|----|
| Table 1: Use case identification..... | 23 |
| Table 2: Use case description and actors | 25 |
| Table 3: Use case description of register events | 26 |
| Table 4: Use case description for update events | 27 |
| Table 5: Use case description for view events | 30 |
| Table 6: Use case description for give printed certificate | 31 |
| Table 7: Use case description for Generate report | 32 |
| Table 8: Use case description for Calculate population size..... | 34 |
| Table 9: Use case description for submit notification..... | 35 |
| Table 10: Use case description for view notification..... | 37 |
| Table 11: Use case description for login..... | 38 |
| Table 12: Birth table | 54 |
| Table 13: Notification table | 55 |
| Table 14: Event table | 55 |
| Table 15: Birth notification table..... | 55 |
| Table 16: Medical officer table..... | 56 |
| Table 17: Federal registrar table | 56 |
| Table 18: Kebele registrar table..... | 57 |
| Table 19: Zone officer table..... | 57 |
| Table 20: Regional officer table..... | 57 |
| Table 21: Statistician table..... | 57 |

Acronyms and Abbreviations

| | |
|------------|---------------------------------|
| HTML..... | HyperText Mark-up Language |
| CSS..... | Cascading Style Sheet |
| PHP..... | HyperTextPre-processor |
| MYSQL..... | My Structured Query Language |
| WAMP..... | Windows Apache MySQL PHP |
| JS..... | Java Script |
| UML..... | Unified Modeling Language |
| BR..... | Business Rule |
| UC..... | Use Case |
| ALT..... | Alternate Course of Action |
| VERA..... | vital event registration agency |

Chapter one

Introduction

1.1 Introduction

Vital event registration is started before the birth of Jesus Christ by orthodox religion followers by registering new born babies. After a long time in 1812 E.C vital event registration is included under the government as one of the governments function by France government. At the beginning it was started by registering (birth, death, marriage, divorce) but now because of the human need for basic, material and spiritual things is growing time to time the united nation posted the following vital events to be registered those are (birth, death, marriage, divorce, adoption, recognition of fatherhood and decide fatherhood through court, death of fetus, separating husband and wife by denying second time marriage and give recognition for children that are born during marriage time).

Vital event registration is now very important for countries including our country for various purposes like developing appropriate policies for certain place based on the registered data, used as evidence for courts and it also used for government planning and budgeting by providing the exact number of population. In general vital event registration means registering events that are so important or have great impact for certain country.

1.2 Background of the organization

Vital events registration in Ethiopia is started In July 28/2008 E.C after long time preparation. From the above vital events that are decide by UN, Ethiopia accepted to register (birth, death, marriage, divorce, adoption, recognition of fatherhood and decision of fatherhood through court). From those events Ethiopia works hardly in four of them (birth, death, marriage, divorce).

Child birth is registered 90 days after the baby is born and (death, marriage and divorce) are registered 30 days after they happen.

1.3 Statement of the problem

The FDRE vital events registration agency has the authority to form central database and to control it but as we ask the employees of East Gojjam vital events registration there is central database in head office they do not have any access to the database so they still work with papers. The registration papers are published and distributed by the FDRE vital events registration agency to the regions then to zones and zones distribute to Keble's.

As we stated earlier the problem of this system basically comes with paper. Papers by themselves bring many problems they can be:-

- Stolen by peoples
- Papers can be tear apart
- Errors that occur on papers cannot be corrected easily.
- Can be damaged easily by accidents.

As we have seen in the above the registration papers are transformed from one office to other using car by humans until they reach in federal vital events registration agency this is tedious and time consuming. The other is when we see the forms registrars may forget to fill spaces and leave them empty this leaves data incomplete.

It is also difficult to find information about users when it's needed especially for regions, zones, and Keble's because they don't have access to any automated system. Data is not well organized in those areas difficult to analyse statistically. To summarise this system have database but it is not distributed to branch offices because of this several problems occur.

1.4 Objective the Project

1.4.1 General Objective

The general objective of this project is to develop an automated vital event registration system. This project aims also to solve every problem that is faced during manual operation.

1.4.2. Specific Objective

To achieve the objective that are stated in the above we state the following specific objectives:-

- To register events like marriage, birth, divorce, death, and adoption more easily than the existing system.

- To enhance the system to better automated system.
- To avoid data loss and error during registration.
- To enable government to count the number of people more effectively by providing better information for the national census program.
- To reduce wastage of resource, time and effort.
- To give printed certificate by generating reports for users.
- To enable actors update data more easily.
- To view and retrieve users information more easily.

1.5 Scope of the project

The current system is manual and runs at many stages so our working boundary will be the overall structure of federal vital event registration offices that specially focuses on registration of birth, death, marriage, adoption, and divorce events. Our project will serve for all offices of vital event registration at all stages.

- ✓ The system will do the registration of marriage, birth, divorce, adoption, and death.
- ✓ The system administrator will create new accounts for actors of the system.
- ✓ Events will have their own form to register and can be viewed by actors that have privilege.
- ✓ The system will generate report and give printed certificate if it is necessary.
- ✓ Update users information
- ✓ This system will have well organized central database that is accessible by every stage employees.

We will try to include features of good system as much as possible.

1.6 Overview of the proposed system

The system that we will develop will eliminate all the paper based problems. The system will have account for every stage offices (Keble, zone, region a) so that they can access the central database to register, view, give certification, and produce report easily without major errors. Data will pass through offices by sending and receiving using online system. The system will be easy for further statistical analysis and it will be more secure. The errors that occur in paper forms will not occur in our system because the forms that we will generate will not let the users pass spaces empty so there will not be incomplete data. We will have many restrictions to get perfect data.

1.7 Significance of the project

The significance of the project would be:-

For the employees that work in every stage of registration:-

- It can facilitate the vital event registration system by changing it to more automated system.
- It transmits data quicker than the existing system.
- Help to avoid errors on registering forms of data.
- Effective and efficient data collection.
- Effectively manage data statistically.

For government:-

- ❖ Government can easily perform national census and categorize population into different group.
- ❖ Government can easily find out why some vital events are occurring more frequently in some places and also recommends the solution.
- ❖ Helps in designing appropriate policy by providing reasonable statistical data.
- ❖ Use as evidence in many areas like citizenship, courts, and to eliminate things that are done arbitrarily like early marriage etc.
- ❖ Provides a secure exchange of vital and statistical information.

For end users:-

Peoples can ask their rights using the registered data as evidence.

For example:-

- ✓ To ask for Keble id card
- ✓ To ask citizenship
- ✓ To ask government different types of infrastructures.
- ✓ To use it as evidence in courts. etc

1.8 System requirements

In order to develop a web base system, it is very important to choose the correct hardware, software and technology. Below there are some explanations of the hardware, software and technology chosen as development tools to the vital event registration system.

1.8.1 Hardware Requirements

The Followings are hardware requirements for developing the system such as:

- Computer: used to develop the system or application.

Specifically:-

- ☞ CPU with 2.20GHZ capacity
- ☞ 8GB RAM
- ☞ Hard disk with 424GB

1.8.2 Software Requirements

There are many software requirements of the vital event registration system.

- Edrawmax: For draw diagrams.
- Microsoft office Visio 2007
- Notepad++: For text editor
- Window -7: for running and processing the tasks
- Mozilla Firefox: For displaying the system
- Wamp server : for implementing the project


1.8.3 Programming language

We will use different types of programming languages to implement or change our system from document to practice. The system that we are going to develop must be user interactive, easy to understand, and interesting to use in order to achieve those features we will use the following programming language.

- 🚦 Server side programming script, PHP.

Because:

- ✓ It is open source
- ✓ User friendly language.
- ✓ Ease of Use:
- ✓ PHP is easy to learn compared to many other scripting languages.
- ✓ PHP can be easily fixed directly into, HTML and CSS
- ✓ PHP is platform independent.
- ✓ Easy to understand

 Database system, MySQL

Because:

- ✓ Open –source
- ✓ Security
- ✓ Easy to develop
- ✓ Client side language, HTML, CSS, JavaScript

1.9 Data Collection Methodology

Data collection methodologies are methods used to collect different data from different data sources (documents, users and organizations etc).

The following are the data collection methods used for requirement

Primary data source

- **Interview:** We used interview as one of the major data collection methods. During the interview we have got different necessary information from the vital event registration offices.
- **Observation:** in order to get better information about the system we have got through the vital event registration process.
- **Document Analysis:** we have analyzed different documents and brochure from the East Gojjam vital event registration office.

Secondary source data

- **Internet:** Internet helps us to see the available samples and to download different types of tutorials which help us in developing the system.

1.10 Feasibility Study

This analysis helps the organization in determining whether they should proceed with a project or not. It determines the future of the project.

1.10.1. Operational feasibility

Operational feasibility is mainly concerned with issues like whether the system will be used if it is developed and implemented. This system will be operationally feasible and the system will be user friendly. The essential questions that help in testing the operational feasibility of a system are following.

- **Does management support the project?** The vital event registration office employees made a strong commitment to cooperate with the project team because they want to eliminate the manual system once and for all. They give us necessary information and resource about the existing system. All these shows that their willingness to the success of the new system.
- **For users:** The system is operationally feasible as it is very easy for the End users to operate it. It only needs to give some basic trainings about computer system.

1.10.2. Economic feasibility

Economic analysis involves the cost incurred on the system development team, estimated cost of hardware and software, the cost of performing implementation and so on.

The system will not require much more cost beyond the capacity of the vital event registration office capital when automating the system such as cost for networking equipment's, hardware and other infrastructures. Therefore, our system will be acceptable, economically towards solving problems.

The system that we are going to develop has many tangible and intangible benefits for the employees of FDRE vital events registration agency.

Tangible benefits

- ⇒ Reduces cost for printing and distributing registration papers.
- ⇒ Reduce cost for transporting registration papers from one stage to another.

Intangible benefits

- ❖ High user satisfaction.
- ❖ Reduces operation time

- ✓ Time to view events
- ✓ Time to update events
- ✓ Time to give printed certificate to the users.etc

1.10.3. Legal feasibility

The system will not have any conflict with the rule and regulation of country and also it is legally acceptable since it respects the business rule of federal vital registration office and regulation as well as other constitutional laws of the country.

1.10.4 Technical feasibility

This is study of resource availability that may affect the ability to achieve an acceptable system. This evaluation determines whether the technology needed for the proposed system is available or not.

- Can the work for the project be done with current equipment existing software technology & available staff?
- Can the system be implementing after development?
- If new technology is needed then what can be developed?

This is concerned with specifying equipment and software that will successfully satisfy the user requirement. An important issue for the development of a project is the selection of suitable front-end and back-end.

Front-end selection:

- ✓ It must have a graphical user interface that assists employees that do not have IT knowledge.
- ✓ Must be according to the organization requirement and the business rule.
- ✓ Must provide excellent reporting features with good printing support.
- ✓ Platform independent.
- ✓ Easy to debug and maintain.
- ✓ Front end must support some back end like Mysql

According to the above stated features we select Java script, HTML and CSS as the front-end for developing our project.

Back-end Selected:

- Multiple user support.
- Efficient data handling.
- Provide efficient security system.
- Efficient data retrieval and maintenance.
- Operating System compatible.
- Easy to install.
- Easy to debug and maintain.
- Easy to connect with the Front-end.

According to above stated features we selected Mysql, PHP as the backend. And to implement the automated system there must be network infrastructure with in different offices of the vital event registration and one dedicated server.

Chapter two

2. System analysis

2.1 Overview of the existing system

The existing system try to reach users from federal to Keble in every stage there is vital event registrar with employees that are responsible for every action. The registration starts from Keble. In Keble the registration is done by the Keble manager with help of nominators. These nominators have their own responsibility which is to give information for the Keble manger if there is new vital event (birth, death, marriage, adoption etc) with in Keble. Then data is copied within four papers called honour files and sent it to woreda and other stages. Within every office (regional, federal and one for central statistics agency) they take one copy from every type of vital event registration papers for themselves. The Ethiopian federal vital event registration agency has many responsibilities including print the registration papers (honour files) and distribute them to regions.

For peoples that live out of Ethiopia there are organizations that are responsible for registration of vital events like (defence ministry of Ethiopia, foreign ministry of Ethiopia and Ethiopian sea transportation and logistics service organization) this organization register vital events and send to federal vital event registration agency.

In Ethiopia there are four vital events that are being registered those are (birth, death, marriage, divorce). Child birth is registered 90 days after the baby is born and (death, marriage and divorce) are registered 30 days after they happen.

If someone wants to register vital events he or she needs to have Keble identity card and if it is birth event the parents of the child should appear physically to the registrars and get registered. If parents are not alive or cannot go to the registrars because of work or other reasons the person who is responsible for the child must appear and register but he or she should have delegation paper from courts.

When marriage is registered they must have four witnesses two on behalf of female and two on behalf then they bring three photos for the certificate. The users must pay to get vital event

registration certificate. For the birth and marriage certificate users pay 35 birr and 25 birr and 10 birr for divorce and death respectively.

The existing system has the following characteristics:-

- ❖ Has long waiting time to get certificate of different vital events to use it as evidence for different offices.
- ❖ Low speed and efficiency.
- ❖ It is difficult to view someone's profile.
- ❖ Is less Secure.
- ❖ Peoples may leave empty spaces on the registration papers.
- ❖ Duplication of data.
- ❖ Loss of personal information.

Users of the existing system

The current system has many users. Those are:-

- ❖ **Government:** - government is major user of this vital event registration system. This helps the government to make effective decisions, to design appropriate policies for appropriate place; it can give recommendations to solve some problems etc.
- ❖ **Courts:** - this system is useful for courts by giving good evidences for the accusations. For example in divorce cases, in decision of fatherhood, in the early marriage cases. It can also be used as evidence to decide whether someone has to go to jail or not because if a person is under eighteen years of age he/she cannot go to jail.
- ❖ **Central statistics agency:**-the vital event registration information management system can also be useful to central statistics agency by providing easy way to conduct statistical analysis on population. For example making easy to do national census, to know easily that how frequently occur those vital events and cause of them.
- ❖ **Federal and regional education offices and schools:**-vital event registration information is very important for education offices and schools because if children registered those offices can easily know that how many children are ready to go to school and can mobilize parents to send the children to school.

- ❖ **Health centers:** -are also users of this system. They give birth and death certificate to the peoples so that they can register the events. They can also view the registered information for various purposes.
- ❖ **Kebele offices:** -those offices have responsibility to register the vital events that occur within the Keble.

2.2 System requirement specification

System requirement specification contains the features that we intended to achieve with our project that we are going to develop.

We try to summarize the specifications in to two categories:-

- ❖ Functional requirement
- ❖ Non functional requirement

2.2.1 Functional requirements

Functional requirements are the intended features of the system. These features may be expressed as services, tasks or functions that the system is required to perform. Functional requirement is a function or feature that must be included in an information system to satisfy the system need and be acceptable by the members. These are statement of service the system should provide how the system should react to particular input and how the system should behave in particular situation. It specifies the software functionality that the developer must build in to the product to enables the user to accomplish tasks.

The following are functional requirement of our project

- ❖ Register the vital events
- ❖ Create user account for actors of the system
- ❖ Update the registered information if it is necessary
- ❖ View the registered information
- ❖ Generate statistical and non-statistical report based on the user's requirement or query.
- ❖ Print certificate and give it to the users

2.2.2 Non functional requirements

Non functional requirement is a requirement that specifies criteria that can be used to measure the quality of a system, rather than specific behaviours. This should be compatible with functional requirements that define specific behaviour or functions.

In general, functional requirements define what a system is supposed to do whereas non-functional requirements define how a system is supposed to be. Non-functional requirements are often called qualities of a system. Other terms for non-functional requirements are constraints, quality attributes, quality goals and quality of service requirements.

Qualities that are non-functional requirements are the following:

Performance

Performance is characterized by the amount of task that are accomplished by a system within given time and resources used.

Good performance may involve one or more of the following characteristics:

- ❖ Short response time for a given task
- ❖ High efficiency
- ❖ High availability means that the system must stay up 24 hours a day within 7 days in a week.
- ❖ High data transmission rate

So to improve the performance of the system that we are going to develop we will use the following methods:-

- Installing our system on multiple servers
- We will use high capacity processors
- Improving band width to make short response time

Security

In order to make the system safe from unauthorized users the system will use a log in account to differentiate authorized users from unauthorized users of the system. This enables the system to verify who has logged in.

- We will use two layer authentication (user name, password)
- We will restrict the privilege of users for example: - regional vital event registration office employees must access only their regions data.

- The federal vital event registration agency employees and statistician from statistician access the whole database.
- We will use also session to restrict users from accessing page without their privilege so we will give session time that it will expire after the time passes.
- We will protect the system from access of external intruders by using password encryption methods like MD5, HASH, SHAL etc

Usability

Usability often refers to how system is easy or clear to interact with users. The system should be easy to learn, easy to operate with the existing stuff.

It also includes the following points:-

- ⇒ More efficient to use—it takes less time to accomplish a particular task
- ⇒ Easier to learn—operation can be learned by observing the object
- ⇒ More satisfying to use

To improve our systems usability we will introduce the following methods

- ✓ Designing more intractable user interface by using more efficient codes like CSS latest versions and HTML.
- ✓ We will make the system that we are going to develop more usable by employees of the vital event registration office by giving continues training
- ✓ We will develop system that will work with at list two languages Amharic and English.

Reliability

The system is expected to perform perfectly without major errors and failure. The system should satisfy the user and should be fault tolerant.

To achieve this

- ✓ We will install our system on high quality hard wares
- ✓ Back up regularly
- ✓ Control external factors like power loss as much as possible.

Modifiability

The system that we will develop will be easy to modify by its users if it is necessary.

- ✓ If we want to add new functionalities

- ✓ If we want to add new features

The system will continue performing as previous time even we change some features.

2.2.2. Technical requirement

The technical requirement of the system:

- ☞ The interface of the system will be user friendly easy to use.
- ☞ The interface will display error message if it detects invalid input
- ☞ The system will deny unauthorized accesses to the system domain
- ☞ The system will provide help for the user.
- ☞ Training the users to access the system.

2.2.3. Business rules

The business rule defines or constrains one aspect of your business that is intended to assert business structure or influences the behavior of your business. Business rules often focus on access control issues or sometimes they focus on polices of the organization.

Business rule are principles, requirements and polices that must be fulfilled and obligated in order the system will function properly and effectively.

The business rules that must be considered for this project are described below.

BR1: Any vital event cannot be registered more than once.

BR2: The person who wants to register the vital event they should appear physically in the Keble registrar offices.

BR3: the vital event information's should be registered on the forms without any error.

BR4: the registrars update events with the order of the courts

BR5: the registrar must assure the correctness of the registered information by showing to the users.

BR6: after reading the registered information the user must sign on it.

BR7: the vital events should be registered with working language of the region that event happens and with Amharic language.

BR8: the registrar must also sign on registration papers after assuring its correctness.

BR9: the certificate is given to the user after they pay the service payment.

BR10: The registrar should keep the confidentiality of the vital event information.

BR11: every employee in every stage access the system according to their privilege

2.3. System requirement analysis

2.3.1. Actor and use case identification

Actor: An actor represents a type of users of the system or external system that plays a role in one or more interactions with our system.

Use case: A use case describes a sequence of actions that provide a measurable value to an actor. A use case is drawn as horizontal ellipse on a UML use case diagram.

2.3.1.1 Actor identification

As we ask from East Gojjam zone vital event registration office employee and as we have seen from the organizational structures the following are actors of the FDRE vital event registration information management system.

System Administrator: a person who is going to administer system we are going to develop.

Keble Registrar: Is a person who has responsibility to register vital events in Keble level.

Statistician: Person who works central statistics agency that groups population according to the calculated information.

Federal registrar: is captain, soldier or employee of Ethiopian embassies in foreign country which is responsible to register vital events.

Regional vital event registration office employees: views the registered information and send feed back if any problem on forms.

Zonal vital event registration office employees: views the registered information and send feed back if any problem on forms to the kebele registrars.

Woreda vital event registration office employees: - views the registered information and send feed back if any problem on forms to the kebele registrars.

2.3.1.2 Use case identification

Table 1: Use case identification

| Use case name | Use case ID | includes |
|---------------------------|-------------|----------|
| Register vital events | UC01 | Login |
| Update registered events | UC02 | Login |
| View registered events | UC03 | Login |
| Give certificate | UC04 | Login |
| Generate report | UC05 | Login |
| Calculate population size | UC06 | Login |
| Count number of birth | UC07 | Login |
| Count number of death | UC08 | Login |
| Count number of marriage | UC09 | Login |
| Count number of divorce | UC10 | Login |
| Send feed back | UC11 | Login |
| Create user account | UC12 | Login |
| Block account | UC13 | Login |
| Update account | UC14 | Login |
| Search account | UC15 | Login |
| View feedback | UC16 | Login |
| View report | UC17 | Login |
| Login | UC18 | |

2.3.1.3 Use case diagram

A system use case model is composed of a use case diagram and the accompanying documentation describing the use cases, actors, and associations. The main difference between an *essential use case* and a *system use case* is that in the system use case you include high-level implementation decisions.

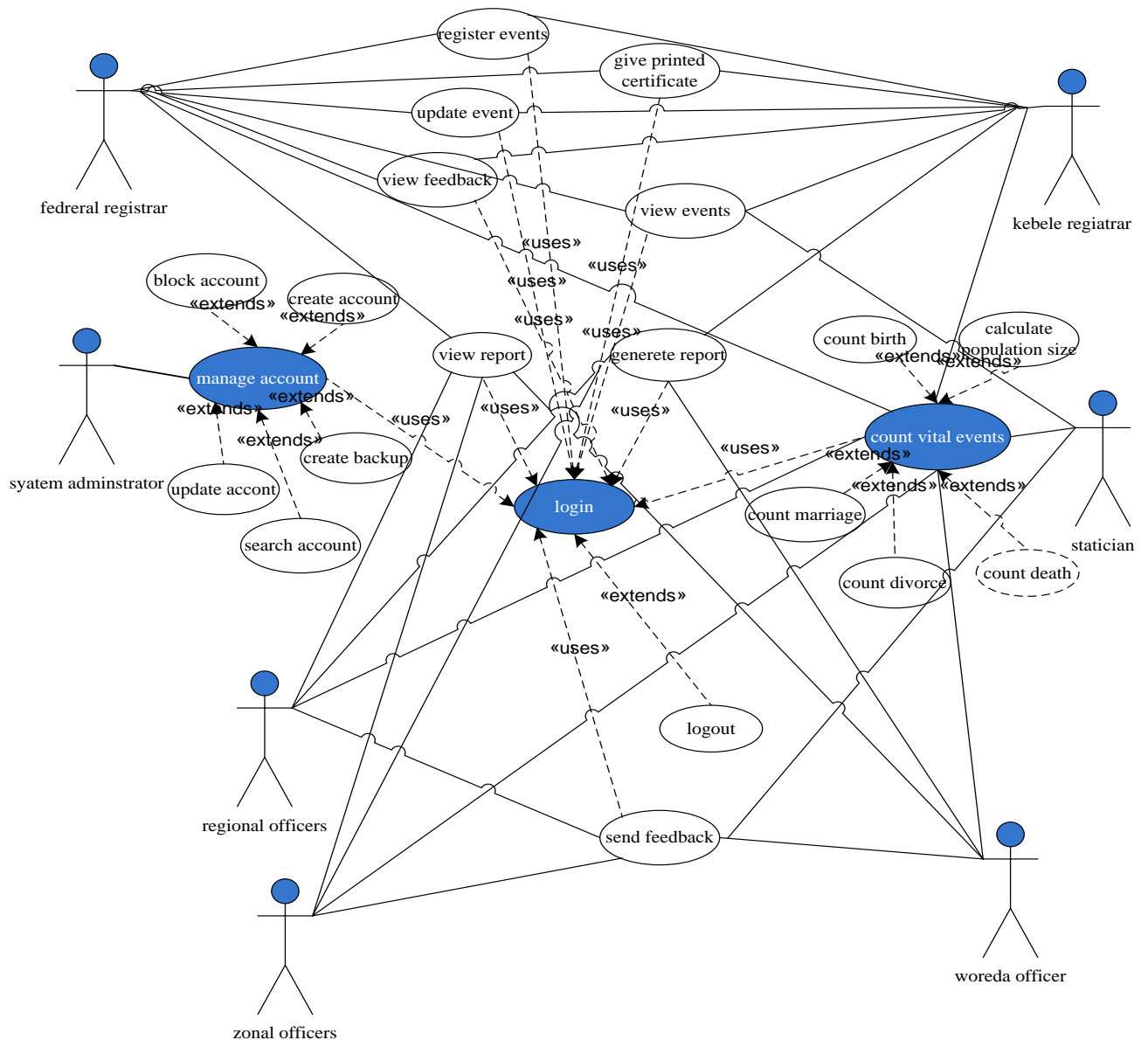


Figure 1: Use Case Diagram

2.3.2 Use case description

Table 2: Use case description and actors

| Actors | Use case |
|--------------------------------------|---|
| System administrator | <ul style="list-style-type: none">✓ login✓ Create account✓ Block account✓ Update account✓ Search account✓ Create backup✓ logout |
| Keble registrars, federal registrars | <ul style="list-style-type: none">✓ login✓ register events<ul style="list-style-type: none">1. birth2. death3. marriage4. divorce✓ view events✓ update events✓ generate report✓ give printed certificate✓ view feedback✓ count vital events<ul style="list-style-type: none">✓ view report✓ logout |
| Statistician | <ul style="list-style-type: none">✓ login✓ calculate population size✓ count birth✓ count death✓ count marriage✓ count divorce✓ view events |

| | |
|---|---|
| | <ul style="list-style-type: none"> ✓ view report ✓ generate report ✓ send feedback ✓ count vital events ✓ logout |
| Zone, woreda and regional employees of VERA | <ul style="list-style-type: none"> ✓ login ✓ view events ✓ generate report ✓ count vital events ✓ view report ✓ send feedback ✓ logout |

Table 3: Use case description of register events

| | |
|------------------------|---|
| Name | Register event |
| Use case ID: | UC01 |
| Participating Actors: | Keble Registrar, federal registrars |
| Description: | Used to register vital events at Keble and federal level. |
| Precondition: | <ul style="list-style-type: none"> ✓ The user must have an account to register events. ✓ Have notification paper from health centers |
| Post condition: | <ul style="list-style-type: none"> ✓ System adds event information to database. ✓ Give certificate |
| Basic course of action | <ol style="list-style-type: none"> 1. The use case starts when registrars indicate they want to register events. 2. Then registrar enters user name and password. |

| | |
|--------------------------------|---|
| | <ol style="list-style-type: none"> 3. Click login button. [Alt: A] 4. The system provides a list of operations. [BR11: access its own page only] 5. The Registrar selects register events link it may be birth, death, marriage, divorce. 6. The Registrar fills and submits the form. [BR1,BR3,BR5,BR6 and BR7: register without error, do not duplicate events] [Alt: B] 7. The system receives data and prepares it to be printed on certificate if it is necessary. 8. The use case ends |
| Alternative course of action : | <p>A:user entered invalid user name and password</p> <p>A3: The system displays error message saying “Invalid user name or password.”</p> <p>A4: Use case continues from step 2</p> <p>B: Registrar filled invalid form data</p> <p>B6: The system shows error message invalid form data message</p> <p>B7: System asks user to re-enter the invalid data again</p> <p>B8. Use case continues from step 6</p> |

Table 4: Use case description for update events

| | |
|-------------------------|---|
| Name | Update events |
| Use case ID: | UC02 |
| Participating Actors: | Keble Registrar, federal registrars |
| Description: | The registrars updates of the registered vital events. |
| Preconditions: | <ul style="list-style-type: none"> ✓ The user must bring order paper from courts that enables the registrar to update the registered event. ✓ The registrar must have an account with update privilege. |
| Basic course of action: | <ol style="list-style-type: none"> 1. The use case starts. 2. Then registrar enters user name and password. 3. Click login button. [Alt: A] 4. The system provides a list of operations. [BR11: access its own page only] 5. The Registrar selects update events link it may be birth, death, marriage, divorce. 6. The registrar enters the events identification Number and submits. [Alt: B] 7. The Registrar updates and submits the form. [BR4, BR5, BR6: update events |

| | |
|-------------------------------|---|
| | <p>according to the order of courts and assure the correctness of data.][Alt: C]</p> <p>8. The system receives data and prepares it to be printed on certificate if it is necessary.</p> <p>9. The use case ends.</p> |
| Alternative course of action: | <p>A:user entered invalid user name and password</p> <p>A3: The system displays error message saying “Invalid user name or password.”</p> <p>A4: Use case continues from step 2</p> <p>B: Registrar filled invalid form data</p> <p>B6: The system shows error message “invalid ID number”.</p> <p>B7: System asks user to re-enter the invalid data again</p> <p>B8. Use case continues from step 6</p> <p>C: registrar fills the data that is needed to be updated.</p> <p>C7: The system displays error message saying “Invalid form data.”</p> <p>C8: System asks user to re-enter the invalid data again.</p> <p>C9: Use case continues from step 7</p> |
| Post conditions: | The system successfully updates the vital events (birth, death, marriage, divorce) |

Table 5: Use case description for view events

| | |
|-------------------------------|--|
| Use case name: | View events |
| Use case ID: | UC03 |
| Participating Actors: | Keble registrars, federal registrars, statistician, regional officers, zone officers |
| Description: | The registrars view events may be for error check or just viewing. |
| Preconditions: | The event must exist in the database and actors must have privilege to view events. |
| Basic course of action: | <ol style="list-style-type: none"> 1. The use case starts 2. The actor enters user name and password. 3. Click login button.[Alt: A] 4. The system provides a list of operations. [BR11: access its own page only] 5. The actor selects viewing event links. 6. The actor fills necessary information and clicks on view button.[Alt: B] 7. The actor viewing events. 8. The use case ends. |
| Alternative course of action: | <p>A:user entered invalid user name and password</p> <p>A3: The system displays error message saying “Invalid user name or password.”</p> <p>A4: Use case continues from step 2</p> |

| | |
|------------------|---|
| | <p>B: Registrar filled invalid event ID number.</p> <p>B6: The system shows error message “invalid ID number”.</p> <p>B7: System asks user to re-enter the invalid data again</p> <p>B8. Use case continues from step 6</p> |
| Post conditions: | The actors successfully views vital event. |

Table 6: Use case description for give printed certificate

| | |
|-------------------------|---|
| Use case name: | Give printed certificate |
| Use case ID: | UC04 |
| Participating Actors: | Keble Registrar, federal registrars |
| Description: | The registrars give printed certificate by the request of the customers. |
| Preconditions: | The event must exist in database and the certificate is printed if the end user request. |
| Basic course of action: | <ol style="list-style-type: none"> 1. The use case starts. 2. The actor enters user name and password. 3. Click login button.[Alt: A] 4. The system provides a list of operations. [BR11: access its own page only] |

| | |
|-------------------------------|--|
| | <p>5. The actor selects generate certificate link.</p> <p>6. Retrieve necessary information's from the database. [Alt: B]</p> <p>7. Click on print button</p> <p>8. The use case ends</p> |
| Alternative course of action: | <p>A:user entered invalid user name and password</p> <p>A2: The system displays error message saying "Invalid user name or password."</p> <p>A3: Use case continues from step 2</p> <p>B: Registrar Retrieve invalid data.</p> <p>B6: The system shows error message "information does not exist".</p> <p>B7: System asks user to re-enter the valid data again</p> <p>B8. Use case continues from step 6</p> |
| Post conditions: | Give printed certificate filling correct information to end users. |

Table 7: Use case description for Generate report

| | |
|-----------------------|--|
| Use case name: | Generate report |
| Use case ID: | UC05 |
| Participating Actors: | Statistician, Keble registrar, federal registrar |
| Description: | ✓ Generating report that gives general |

| | |
|-------------------------------|---|
| | <p>information about the records in database.</p> <p>✓ The reports are used for further researches and analysis.</p> |
| Preconditions: | The actor must have an account and privilege to generate report. |
| Basic course of action: | <ol style="list-style-type: none"> 1. The registrar enters username and password 2. The registrar click login button. [Alt: A] 3. The system provides a list of operations. [BR11: access its own page only] 4. The registrar selects generate Report link. 5. The registrar selects the report type 6. then generate report[Alt: B] 7. the use case ends |
| Alternative course of action: | <p>A:user entered invalid user name and password</p> <p>A2: The system displays error message saying “Invalid user name or password.”</p> <p>A3: Use case continues from step 2</p> <p>B: Registrar wants Retrieve the data that is not in the database.</p> <p>B6: The system shows error message “information does not exist”.</p> <p>B7: System asks user to retrieve again.</p> |

| | |
|------------------|---|
| | B8. Use case continues from step 5 |
| Post conditions: | Generate reports successfully. |

Table 8: Use case description for Calculate population size

| | |
|-------------------------|---|
| Use case name: | Calculate population size |
| Use case ID: | UC06 |
| Participating Actors: | Statistician |
| Description: | Allows the statistics agency to do national census easily and calculate the size of people over Keble, woreda, zone, region and over all country's population. |
| Preconditions: | <ul style="list-style-type: none"> ✓ The actor must have an account with calculate population size privilege. ✓ The events must be registered in central database. |
| Basic course of action: | <ol style="list-style-type: none"> 1. The actor enters user name and password. 2. Click login button. [Alt: A] 3. The system provides a list of operations. [BR11: access its own page only] |

| | |
|--------------------------------|--|
| | <p>4. The Statistician selects calculate the population size link.</p> <p>5. The Statistician clicks calculate button.[Alt: B]</p> <p>6. Generates reports if it is necessary. [Alt: C]</p> <p>7. The use case ends.</p> |
| Alternative course of action : | <p>A:user entered invalid user name and password</p> <p>A2: The system displays error message saying “Invalid user name or password.”</p> <p>A3: Use case continues from step 2</p> <p>B: The Statistician wants to calculate data that is not in the database.</p> <p>B6: The system shows error message “information does not exist”.</p> <p>B7: System asks user to retry again.</p> <p>B8. Use case continues from step 5</p> |
| Post conditions: | Calculate the size of population without any error |

Table 9: Use case description for submit notification

| | |
|----------------|---------------------|
| Use case name: | submit notification |
| Use case ID | UC22 |

| | |
|--------------------------------|---|
| Participating Actors: | Medical officers |
| Description: | Approves the vital events that happens in health centers and submit notification to registrars. |
| Preconditions: | <ul style="list-style-type: none"> ✓ The vital events must happen at health centers. ✓ The medical officer must have an account |
| Basic course of action: | <ol style="list-style-type: none"> 1. The actor enters user name and password. 2. Click login button [Alt: A] 3. Select the submit notification link 4. Fill necessary information's. 5. Click submit buttons [Alt: B] 6. The use case ends |
| Alternative course of action : | <p>A:user entered invalid user name and password</p> <p>A2: The system displays error message saying "Invalid user name or password."</p> <p>A3: Use case continues from step 1</p> <p>B: medical officer filled invalid form data</p> <p>B5: The system shows error message "invalid form data".</p> <p>B6: System asks user to re-enter the valid data again</p> <p>B7. Use case continues from step 4</p> |
| Post conditions: | To approve the happening of vital events |

| | |
|--|--|
| | |
|--|--|

Table 10: Use case description for view notification

| | |
|-------------------------------|--|
| Use case name: | view notification |
| Use case ID: | UC03 |
| Participating Actors: | Keble registrars, federal registrars |
| Description: | The registrars view notification to know if there is any new event that is not registered. |
| Preconditions: | The event must happen at health centers and the actor must have privilege to view notifications. |
| Basic course of action: | <ol style="list-style-type: none"> 1. The use case starts 2. The actor enters user name and password. 3. Click login button. [Alt: A] 4. The system provides a list of operations. [BR11: access its own page only] 5. The actor selects viewing notification link. 6. clicks on view button [Alt: B] 7. The actor viewing notification. 8. The use case ends |
| Alternative course of action: | A: user entered invalid user name and password |

| | |
|------------------|--|
| | <p>A3: The system displays error message saying “Invalid user name or password.”</p> <p>A4: Use case continues from step 2</p> <p>B: the actors clicks on view button</p> <p>B6: The system shows error message “there is no notification”.</p> <p>B7: System asks user to retry again</p> <p>B8. Use case continues from step 5</p> |
| Post conditions: | The actors successfully views notifications that comes from health organizations. |

Table 11: Use case description for login

| | |
|------------------------|---|
| Use case name : | Login |
| Use case ID | UC24 |
| Participating Actors: | Keble registrars, federal registrars, statistician, regional officers, zone officers, medical officer, administrator. |
| Description : | Users are authenticated and enter to their own user interface page. |
| Precondition | Users must get an account from the system administrator |
| Post condition | The user is authenticated and taken to his/her own user interface |
| Basic course of action | 1. The user opens the main home page |

| | |
|-----------------------------|---|
| | <p>by writing the URL of the website.</p> <ol style="list-style-type: none"> 2. The system display the Main Home page 3. Actor clicks a login link 4. The user inputs user name and password and submit [Alt: A] 5. The system validates the account and displays the user require information. 6. The system directs to his own page [BR11: access its own page only] |
| Alternate course of action: | <p>A: If the login name or password is invalid</p> <p>A4: The system displays invalid username or password message</p> <p>A5: The user re-enters the username and password</p> <p>A6. Turn back to step 4</p> |

2.3.3. UML Sequence Diagram

A UML Sequence diagram showing the sequence of interactions among objects and used to represent or model the flow of messages, events and actions between the objects or components of a system. Sequence Diagrams are also used primarily to design, document and validate the architecture and interfaces of the system by describing the sequence of actions that need to be performed to complete a task.

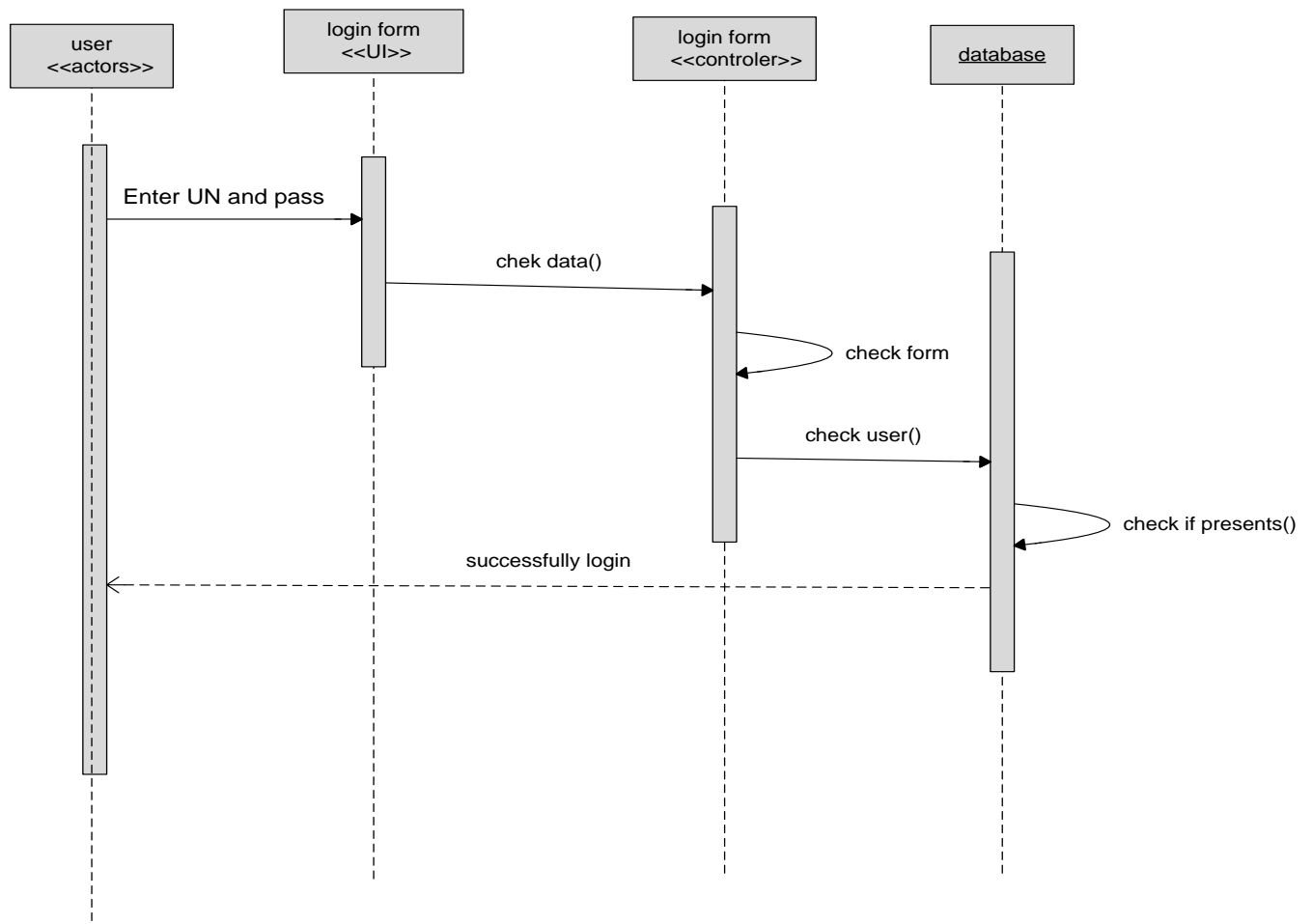


Figure 2: sequence diagram for login

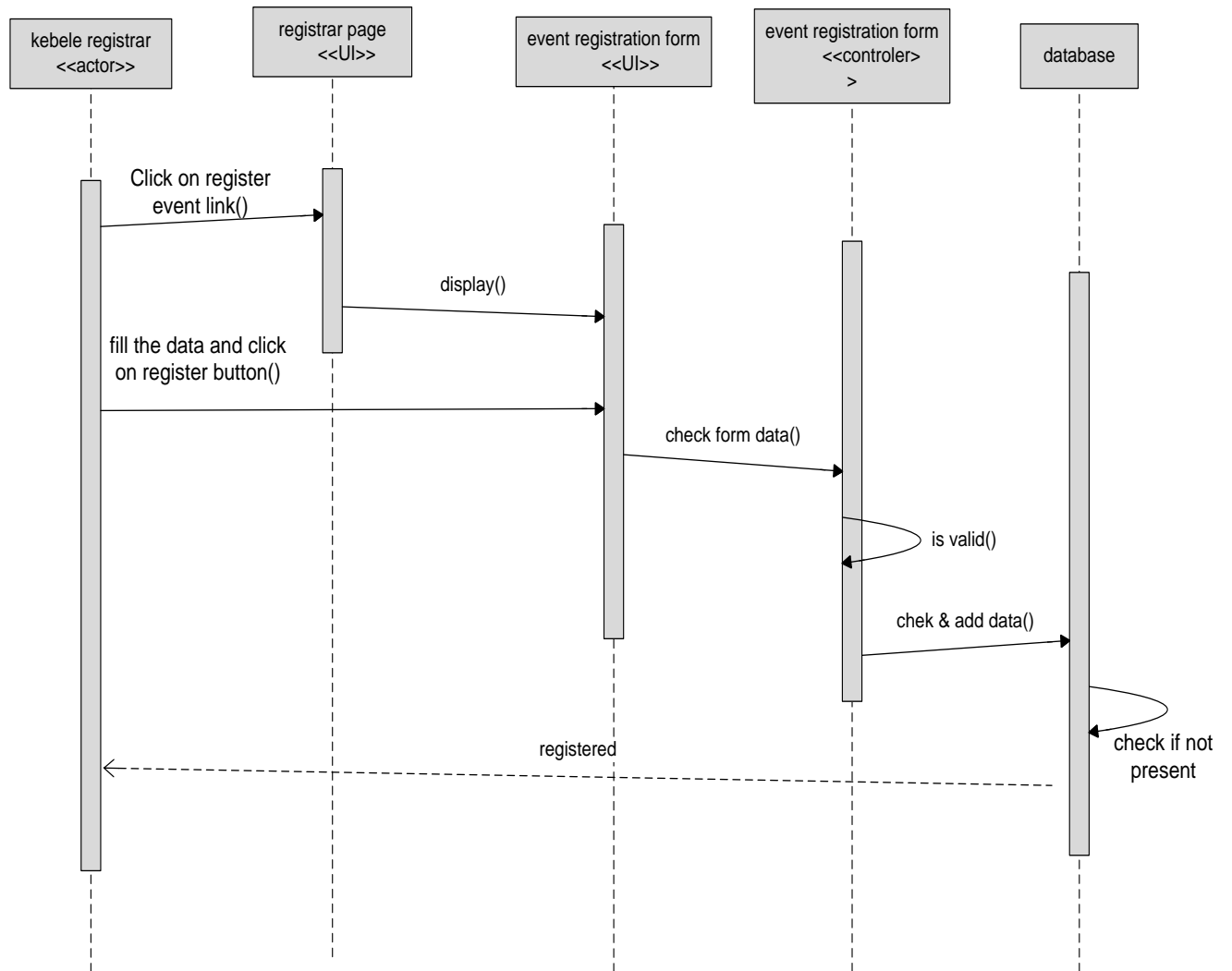


Figure 3: sequence diagram for registering events

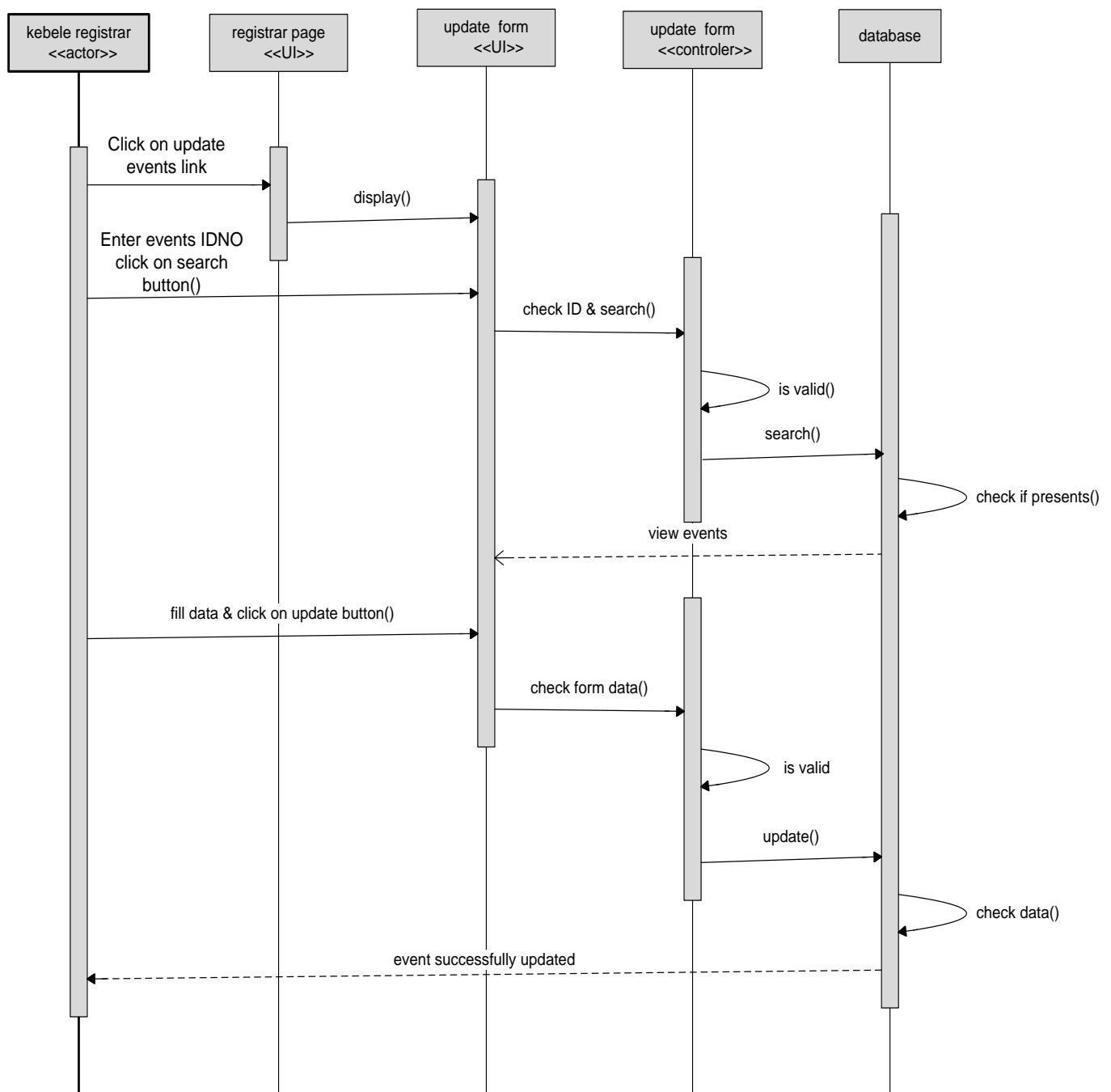


Figure 4: sequence diagrams for updating events.

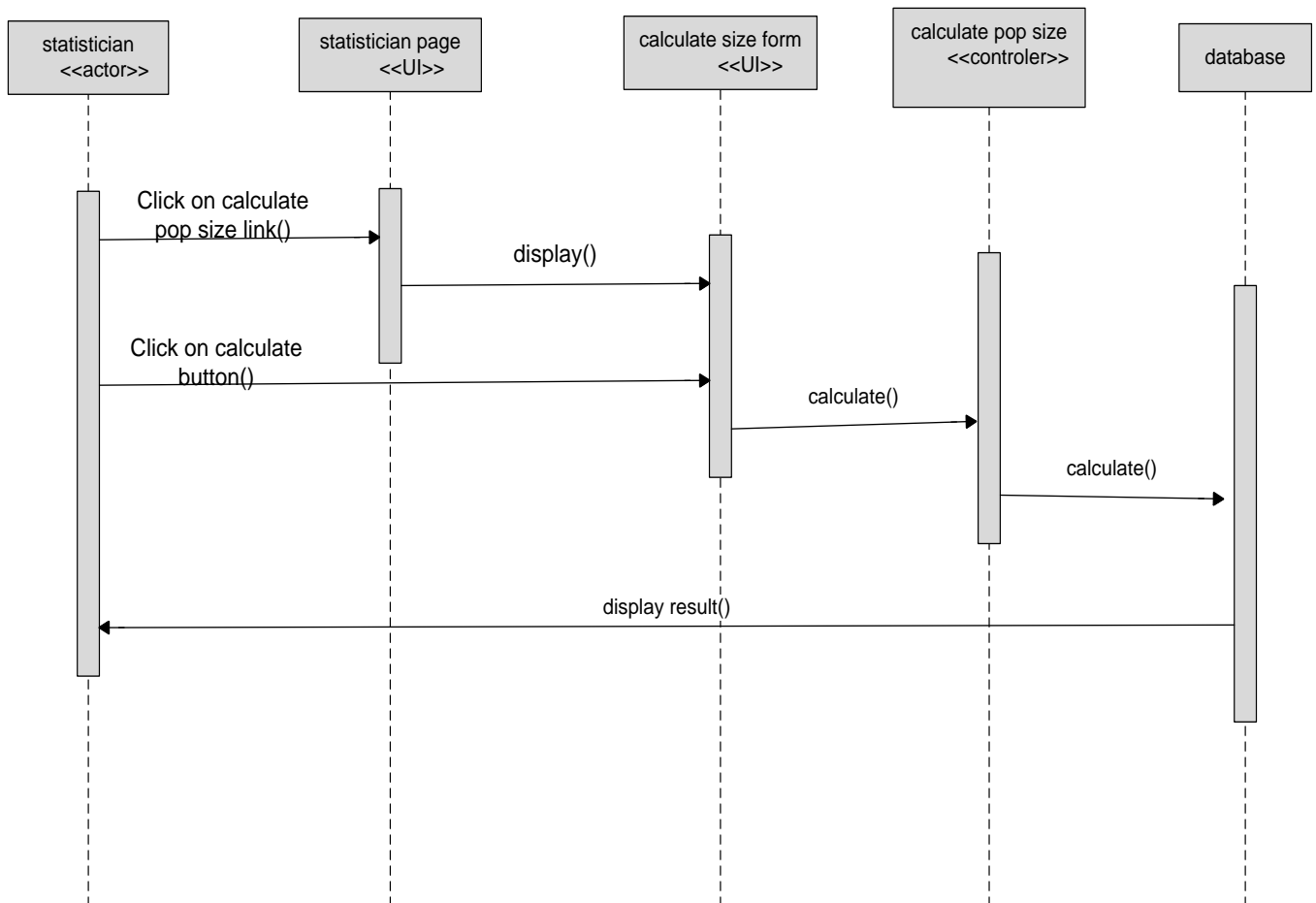


Figure 5: sequence diagram for the calculating population size

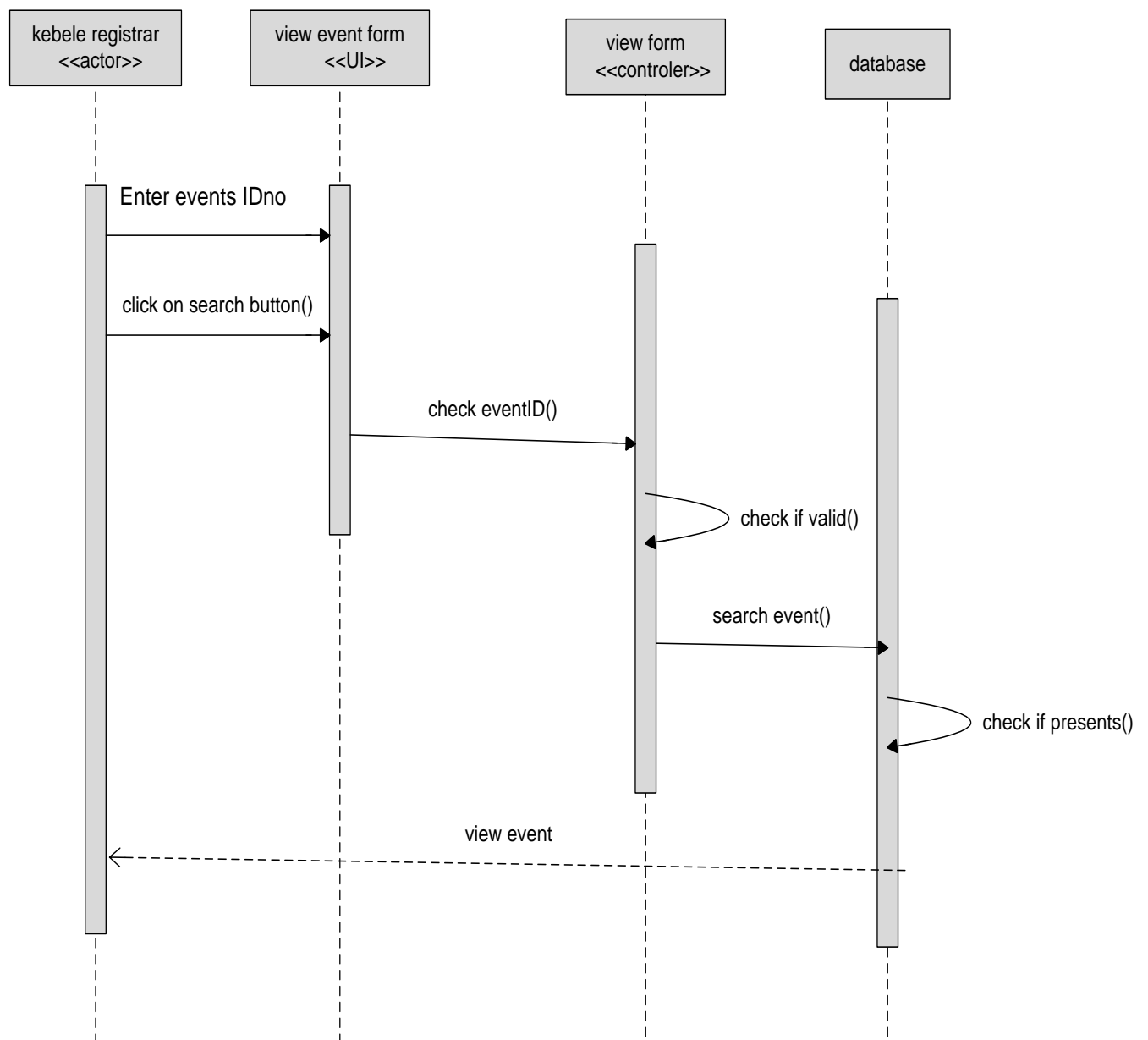


Figure 6: sequence diagram for view events

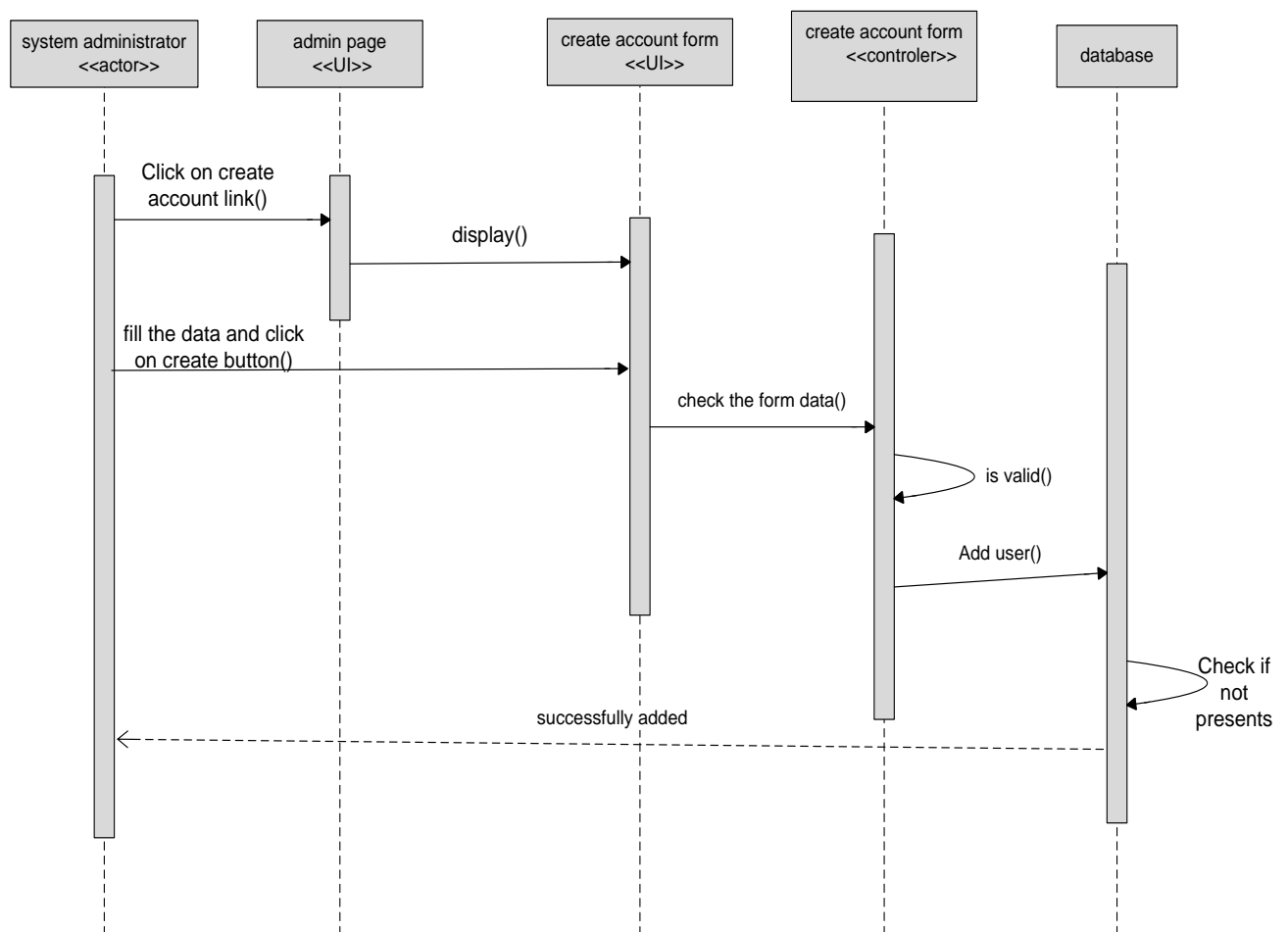


Figure 7: Sequence diagram for create user account

2.3.4 Activity diagram

Activity diagram is basically a flow chart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagrams deal with all types of flow control by using different elements like fork, join etc

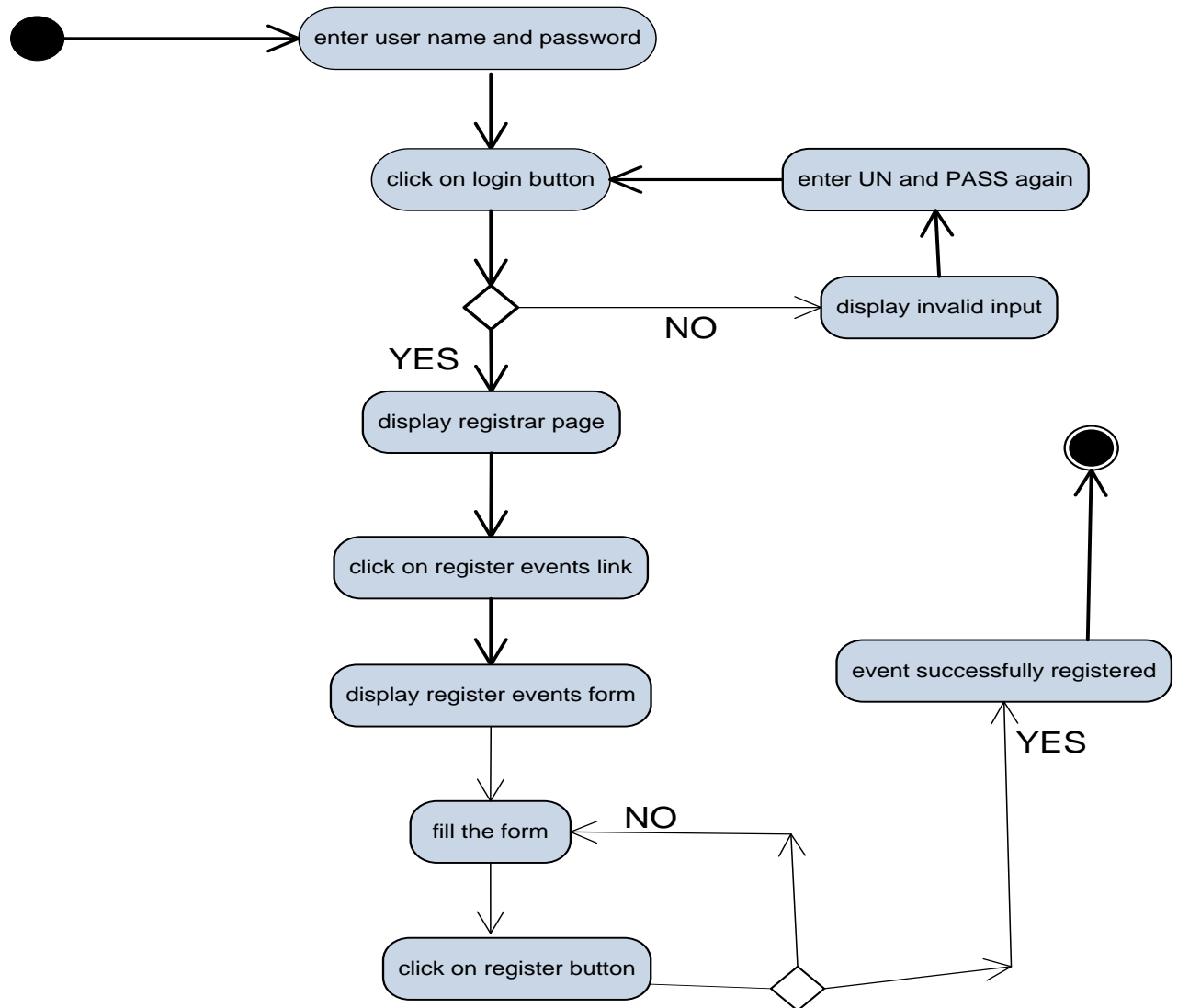


Figure 8: Activity diagram for register events

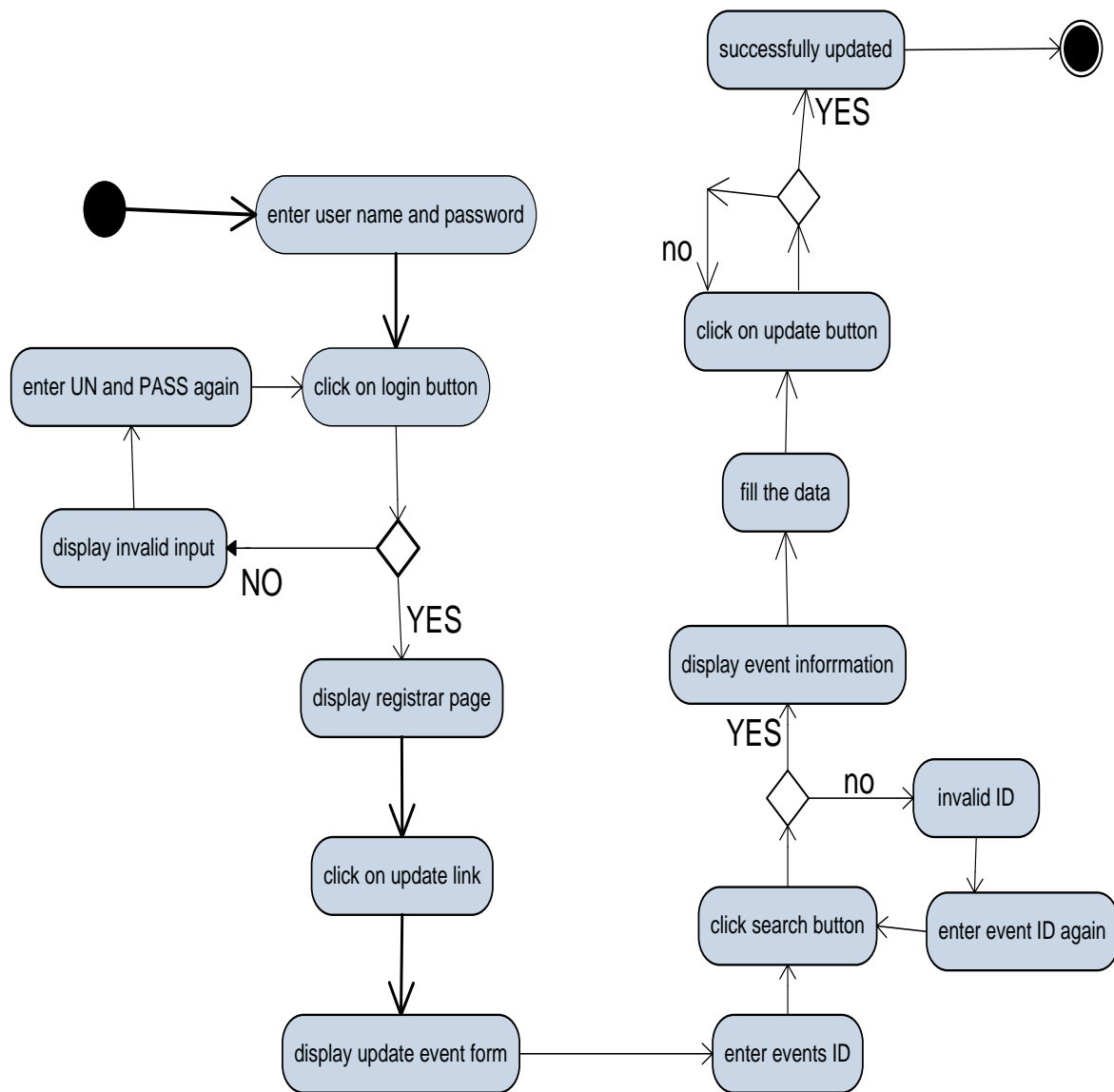


Figure 9: activity diagrams for update event

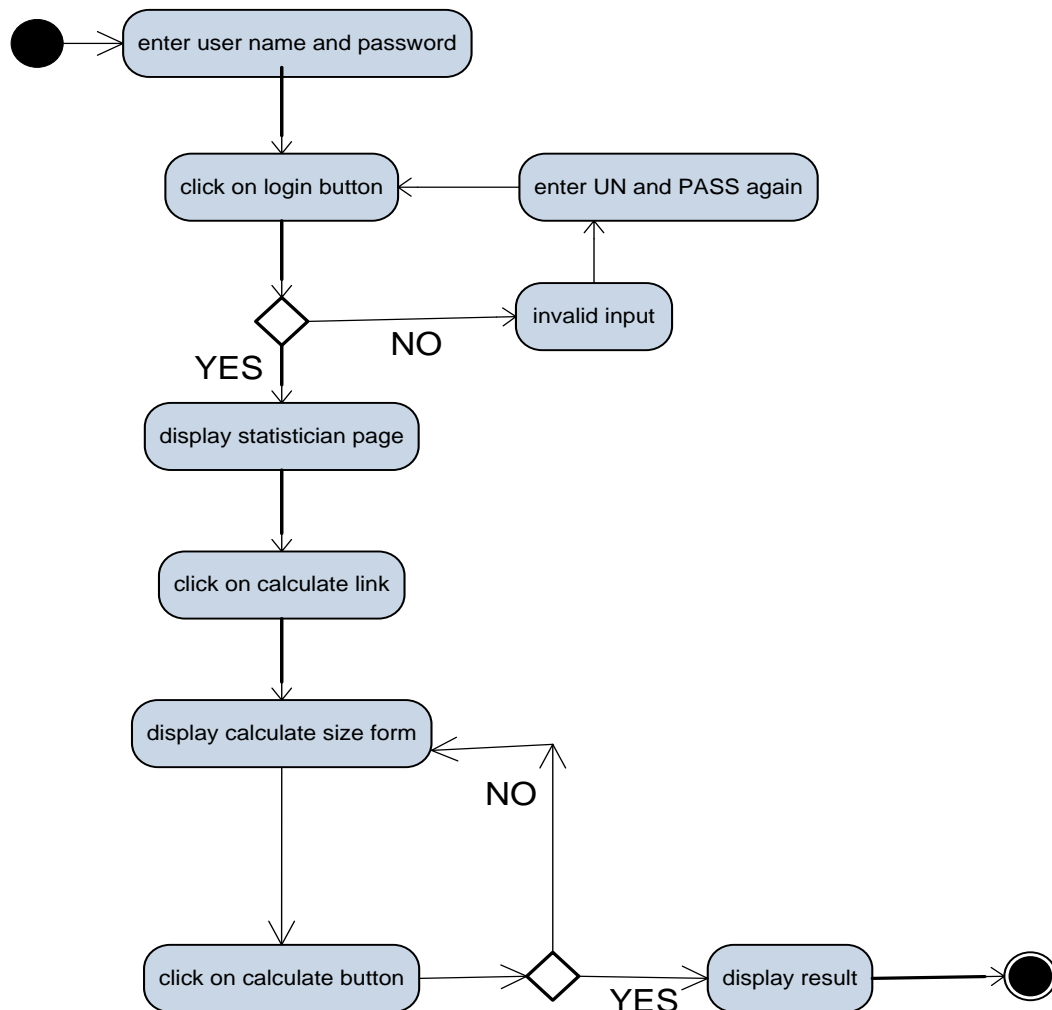


Figure 10: Activity diagram for calculating population size

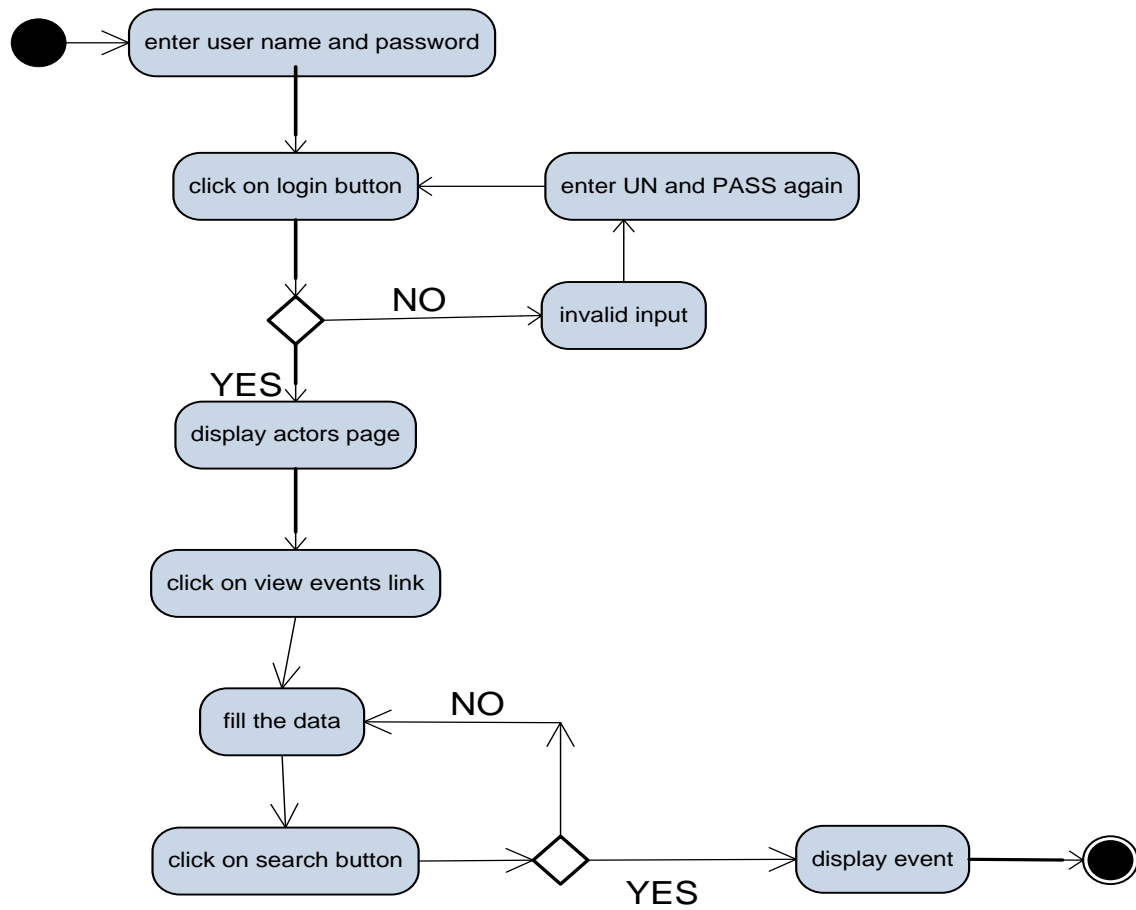


Figure 11: Activity diagram for view event

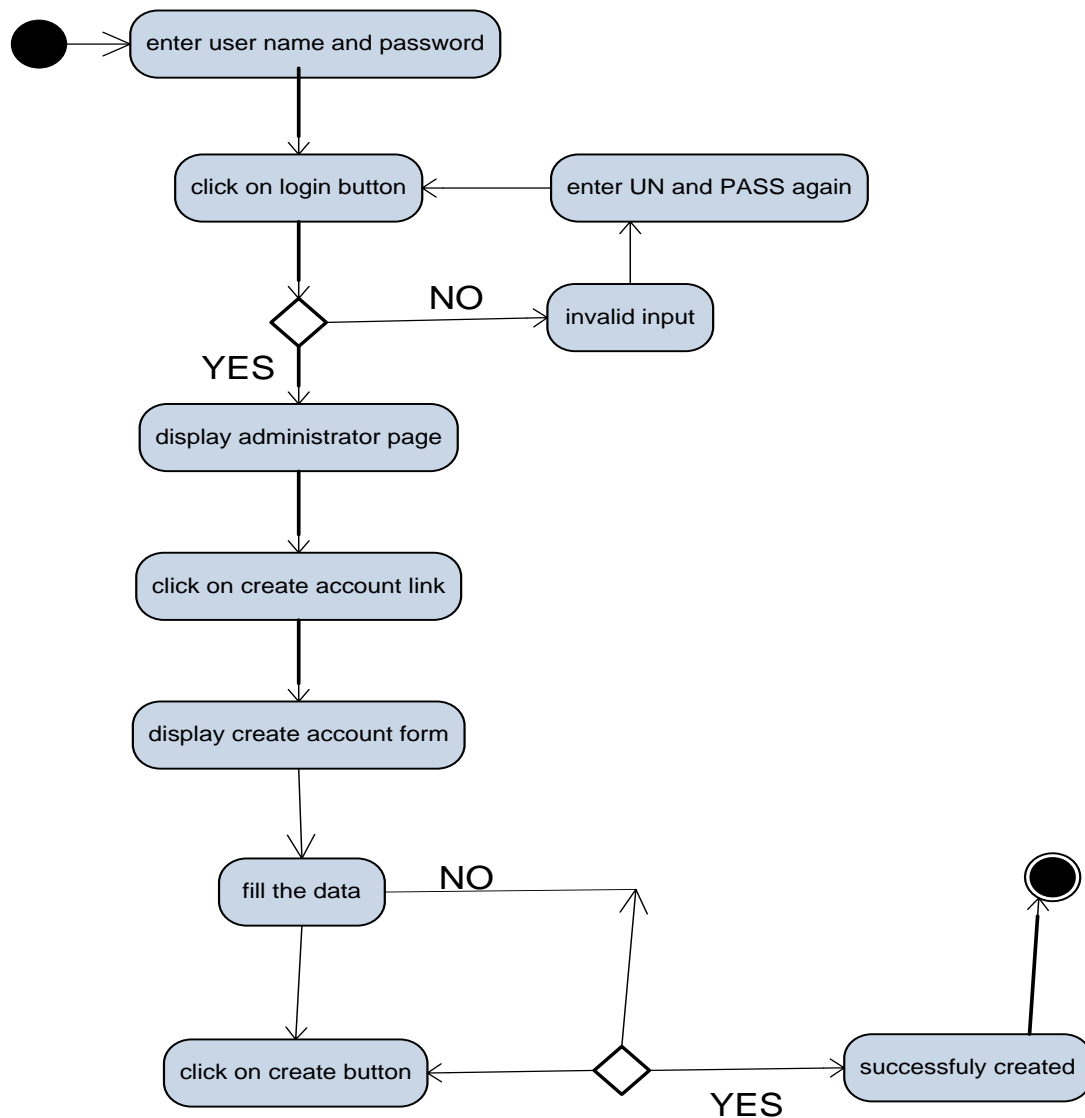


Figure 12: Activity diagram for create account

2.3.5 Analysis Class Diagram

UML class diagrams are the mainstay of object-oriented modelling. Class models show the classes of the system, their interrelationships (including inheritance, aggregation, and association), and the operations and attributes of the classes. Class diagrams are used for a wide variety of purposes, including both conceptual/domain modelling and detailed structural design modelling.

Chapter three

3. System Design

System design is the transformation of the analysis model into a system design model. System design is the first part to get into the solution domain in a software development. System analysis transform the analysis model into the design model that takes into account the non-functional requirements and constraints described in the problem statement and requirement analysis. The purpose of designing is to show the direction how the system is built and to obtain clear and enough information needed to drive the actual implementation of the system. The objectives of design are to model the system with high quality.

Implementing of high quality system depend on the nature of design created by the designer. If one wants to change to the system after it has been put in to operation depends on the quality of the system design.

3.1. Design class diagram

The class diagram represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application. The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The classes diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages. The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints.

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

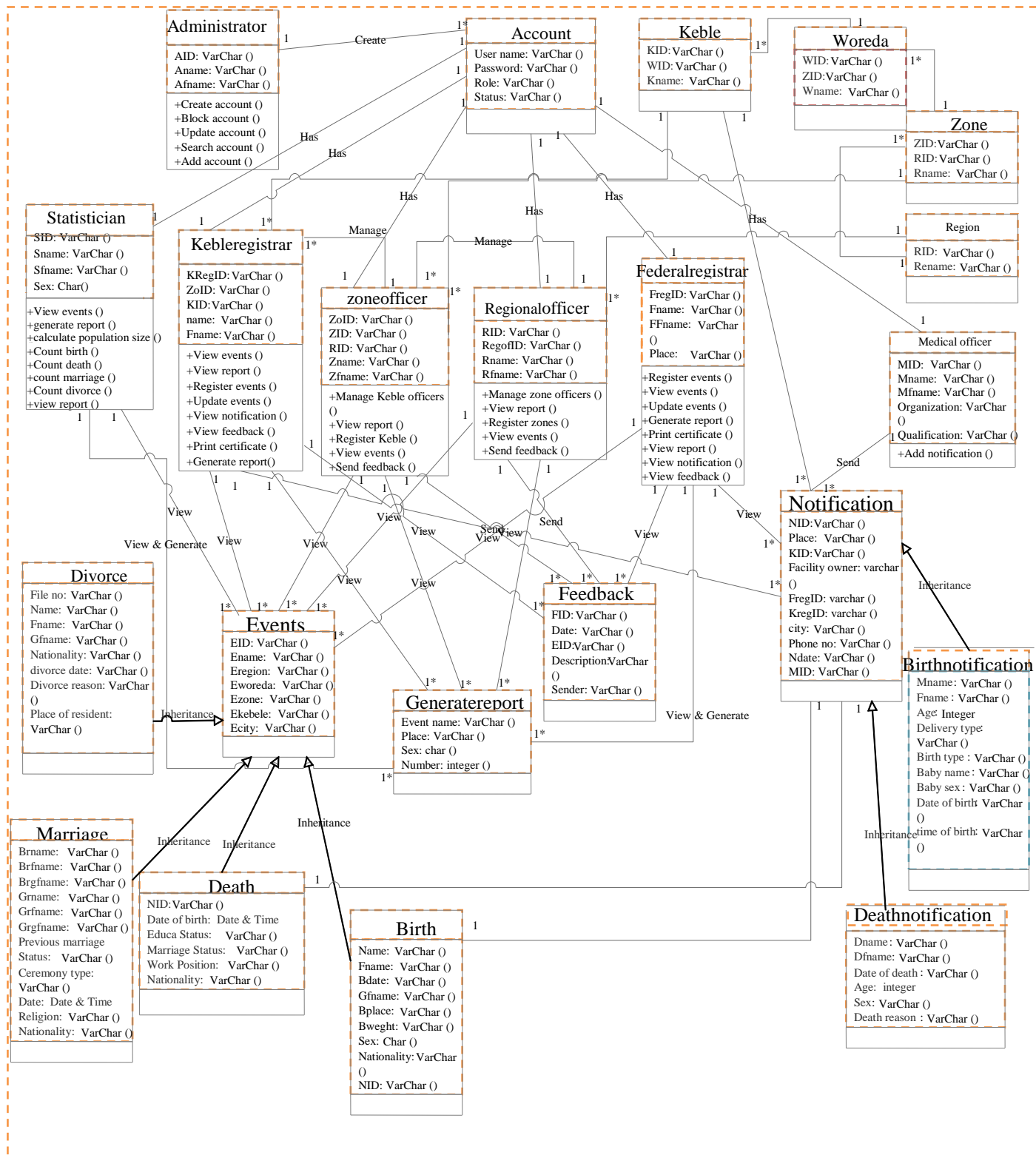


Figure 14: Design class diagram

3.2. Database design

3.2.1. Physical data model

A physical data model (or database design) is a representation of a data design as implemented, or intended to be implemented, in a database management system. In the lifecycle of a project it typically derives from a logical data model, though it may be reverse-engineered from a given database implementation. A complete physical data model will include all the database artifacts required to create relationships between tables or to achieve performance goals, such as indexes, constraint definitions, linking tables, partitioned tables or clusters. Analysts can usually use a physical data model to calculate storage estimates; it may include specific storage allocation details for a given database system.

A physical database model shows all table structures, including column name, column data type, column constraints, primary key, foreign key, and relationships between tables.

- ✓ Entities are tables.
- ✓ Attributes are columns.
- ✓ Unique identifiers are columns that are not allowed to have NULL values.
- ✓ Relationships are modeled by foreign keys.

3.2.2. Normalized table

Table 12: Birth table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|------------------|-------------|-------------|-------------|------------|
| Eid | Varchar(30) | yes | | yes |
| KregID | Varchar(20) | | yes | |
| ZoID | Varchar(20) | | yes | |
| RegofID | Varchar(20) | | yes | |
| City | Varchar(20) | | | |
| Baby name | Varchar(20) | | | |
| Father name | Varchar(20) | | | |
| Grandfather name | Varchar(20) | | | |
| Birth date | Date & time | | | |

| | | | | |
|-------------|-------------|--|--|--|
| Birth place | Varchar(20) | | | |
| weight | Integer | | | |
| sex | char | | | |
| Age | integer | | | |
| Nationality | Varchar(20) | | | |

Table 13: Notification table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|----------------|---------------|-------------|-------------|------------|
| Nid | Varchar(30) | yes | | yes |
| KID | Varchar(20) | | yes | |
| FregID | Varchar(20) | | yes | |
| MID | Varchar(20) | | yes | |
| place | Varchar(30) | | | |
| Facility owner | Varchar(20) | | | |
| Facility name | Varchar(20) | | | |
| city | Varchar(20) | | | |
| Phone no | Varchar(13) | | | |
| Date of sent | Date and time | | | |

Table 14: Event table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|-------------|-------------|-------------|-------------|------------|
| Eid | Varchar(30) | yes | | yes |
| RegID | Varchar(20) | | yes | |
| ZoID | Varchar(20) | | yes | |
| city | Varchar(20) | | | |
| kregID | Varchar(20) | | yes | |

Table 15: Birth notification table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|-------------|-------------|-------------|-------------|------------|
| Nid | Varchar(20) | yes | | |
| KID | Varchar(20) | | yes | |

| | | | | |
|-----------------|---------------|--|-----|--|
| FregID | Varchar(20) | | yes | |
| KregID | Varchar(20) | | yes | |
| MID | Varchar(20) | | yes | |
| Place | Varchar(30) | | | |
| Facility owner | Varchar(20) | | | |
| Facility name | Varchar(20) | | | |
| City | Varchar(20) | | | |
| Phone no | Varchar(13) | | | |
| Date of sent | Date and time | | | |
| Mname | Varchar(20) | | | |
| MFname | Varchar(20) | | | |
| Age | Integer(2) | | | |
| Birth type | Varchar(20) | | | |
| Delivery type | Varchar(20) | | | |
| Baby first name | Varchar(20) | | | |
| Baby last name | Varchar(20) | | | |
| Date of birth | Date & time | | | |
| Time of birth | Varchar(20) | | | |

Table 16: Medical officer table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|--------------------|-------------|-------------|-------------|------------|
| MID | Varchar(30) | yes | | yes |
| Mname | Varchar(20) | | | |
| MFname | Varchar(20) | | | |
| Organization/place | Varchar(20) | | | |
| Qualification | Varchar(20) | | | |

Table 17: Federal registrar table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|-------------|-------------|-------------|-------------|------------|
| FregID | Varchar(30) | yes | | yes |

| Fname | Varchar(20) | | | |
|--------------------|-------------|-------------|-------------|------------|
| Lname | Varchar(20) | | | |
| Organization/place | Varchar(20) | | | |
| Column name | Data type | Primary key | Foreign key | Uniqueness |

Table 18: Kebele registrar table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|-------------|-------------|-------------|-------------|------------|
| KregID | Varchar(30) | yes | | yes |
| ZoID | Varchar(30) | | yes | |
| KID | Varchar(30) | | yes | |
| Fname | Varchar(20) | | | |
| Lname | Varchar(20) | | | |

Table 19: Zone officer table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|-------------|-------------|-------------|-------------|------------|
| ZoffiID | Varchar(30) | yes | | yes |
| ZID | Varchar(30) | | yes | |
| RegofID | Varchar(30) | | yes | |
| Fname | Varchar(20) | | | |
| Lname | Varchar(20) | | | |

Table 20: Regional officer table

| Column name | Data type | Primary key | Foreign key | Uniqueness |
|-------------|-------------|-------------|-------------|------------|
| RID | Varchar(30) | | yes | |
| RegofID | Varchar(30) | yes | | yes |
| Fname | Varchar(20) | | | |
| Lname | Varchar(20) | | | |

Table 21: Statistician table

| | | | | |
|-------|-------------|-----|--|-----|
| SID | Varchar(30) | yes | | yes |
| Fname | Varchar(20) | | | |
| Lname | Varchar(20) | | | |

3.1. User interface design



Figure 15: Home page of the system

Figure 16: registrar's page of the system

Figure 17: event registration forms

3.3 system architecture

3.3.1. Deployment diagram

Deployment diagram is a structure diagram which shows architecture of the system as deployment or distribution of software artifacts to deployment targets. Deployment diagrams model the physical architecture of a system. It also shows the relationship between the software and hardware. A deployment diagram shows how and where the system is to be deployed; that is, its execution architecture.

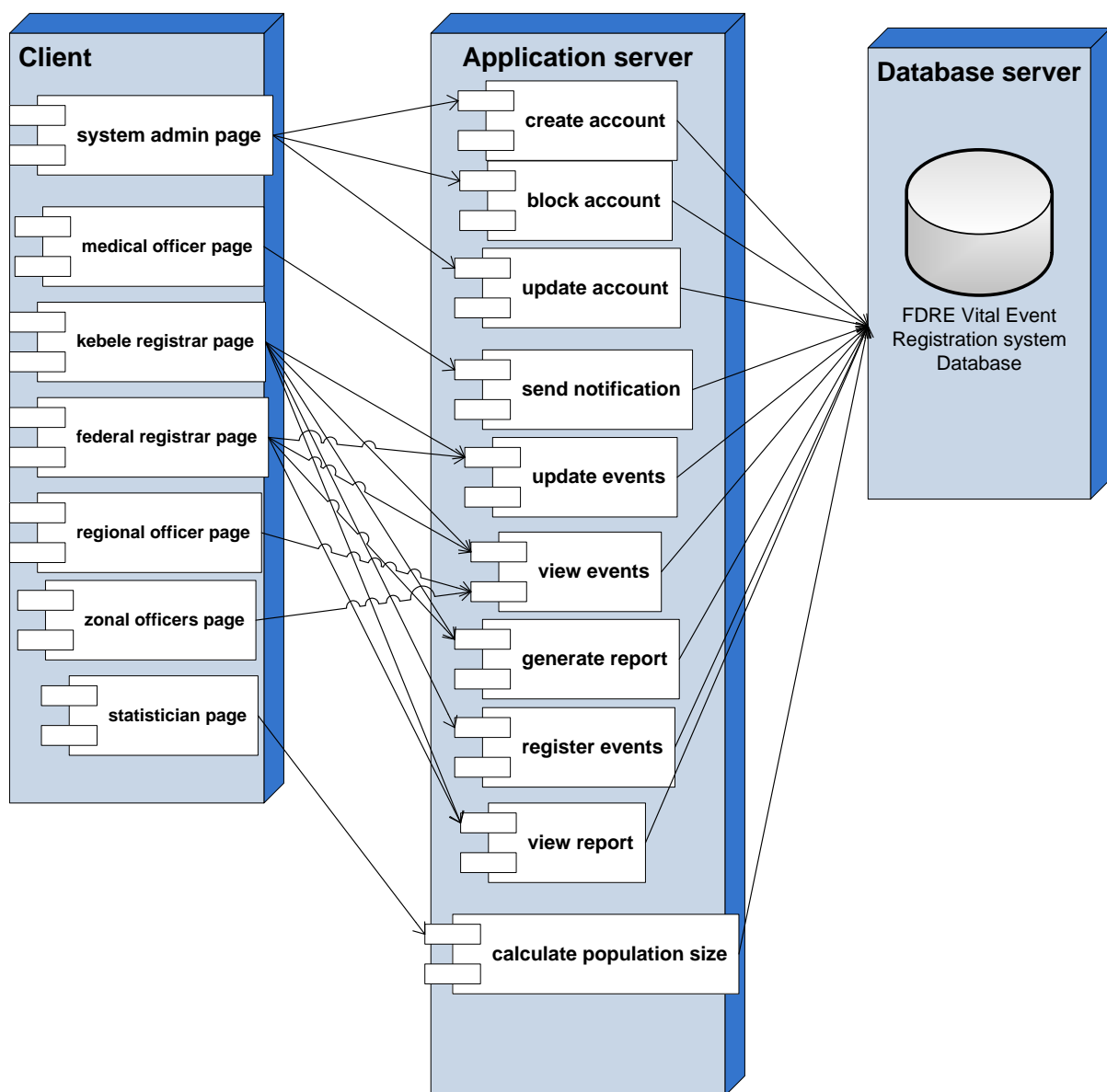


Figure 16 Deployment diagram

References

- [1]. Technical Feasibility, Economic Feasibility, Operational Feasibility, and Legal Feasibility: <http://www.freetutes.com/systemanalysis/sa3-technical-economic-operational-legal.html>
- [2]. Definition of functional requirement: <http://www.ops.fhwa.dot.gov/functional-requirement>
- [3] Definition of functional requirement & non-functional requirement: <https://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/>
- [4]. Activity diagram: https://en.wikipedia.org/wiki/Activity_diagram
- [5]. Class diagram: https://en.wikipedia.org/wiki/Class_diagram
- [6]. Physical Data Model: <https://www.1keydata.com/datawarehousing/physical-data-model.html>
- [7]. about FDRE vital events registration system: www.vera.gov.et