**CHAPTER THREE**

**Methodology and Design**

**3.1 Introduction**

A methodology is a rigorous study or inquiry, particularly to unearth new facts or information; thus, research methodology should be good enough to enable the achievement of the specified objectives, which are achievable using specific components, such as data collection and design procedures, and system modeling (use case, activity, and class diagrams). This chapter provides the input/output specifications as well as the system requirements for the Central E-Birth Registration and Certificate Issuing System under construction.

**3.2 Methods of Data Collection**

Before developing any system, collecting data and facts about the existing system is critical to understand the existing system better. This research was carried out using two methods.

i. Observation of the processes involved in the existing system

ii. Documentation

**3.2.1 Observation of the Work Environment**

This method was employed to acquire information and data for this study by monitoring how the manual system worked. The most evident flaws in the existing system were discovered via detailed inspection. Using the observational approach, will give a firsthand detailed information about the existing system.

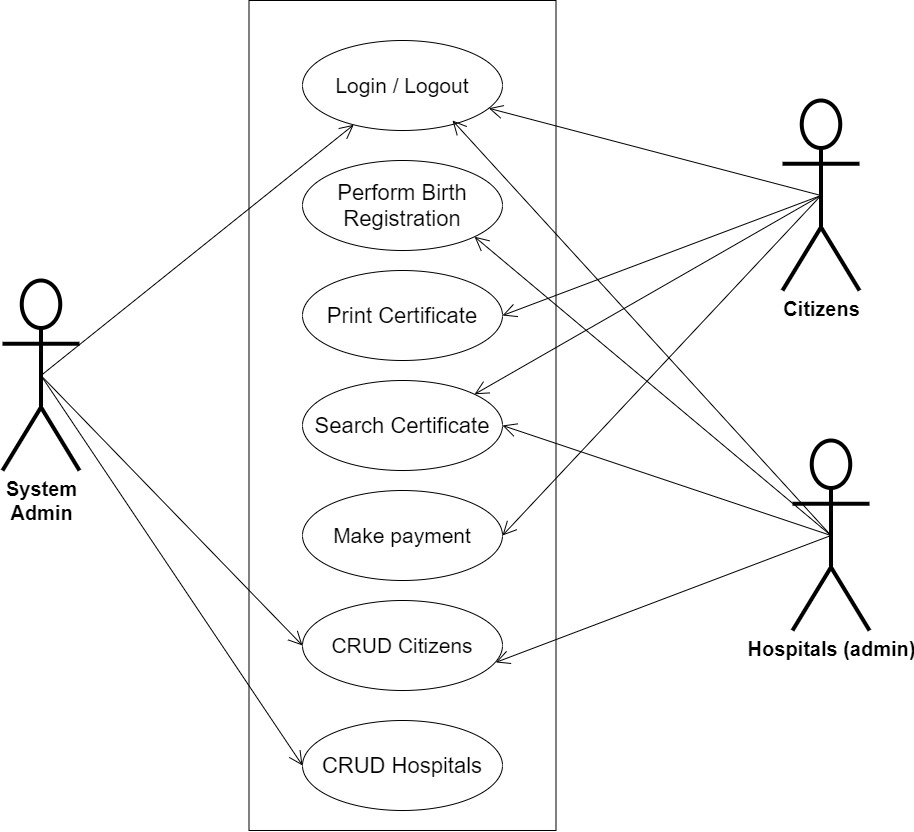
**3.2.2 Documentation**

Documentation is a type of secondary data collection. This method makes use of journals, manuals, past work, publications, and other sources. This method of data collection is used because it allows for comparison with past studies. This includes the internet, which is a data collection tool. The internet was used to find information on difficult or ambiguous issues.

**3.3 System Modeling**

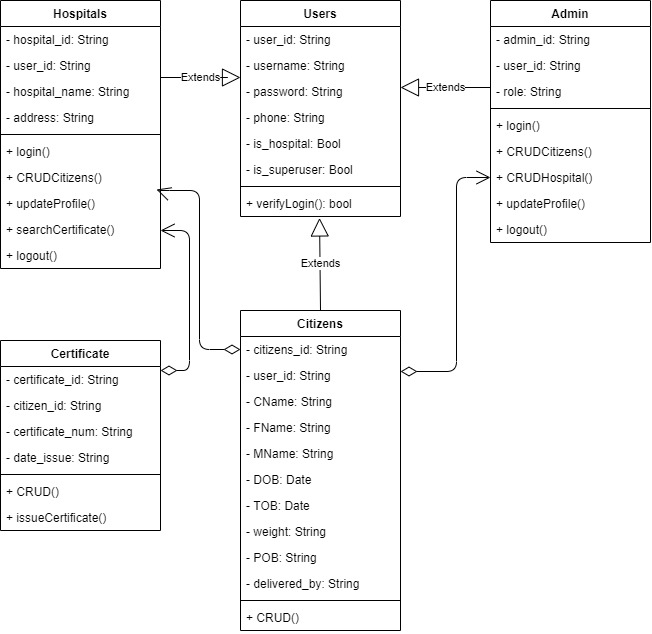
A system model is a conceptual model, which is used to describe and represent a system. It portrays the interaction between a set of components that work together to achieve a common purpose. Visual models of object-oriented software-intensive systems may be created utilizing a set of visual notation techniques included in the Unified Modeling Language, which is used in the development of this modern system. UML diagrams utilized in this new design include use case diagrams, class diagrams, and activity diagrams.

**3.3.1 Use Case Diagrams**



**Fig 3.1 System Use Case Diagram**

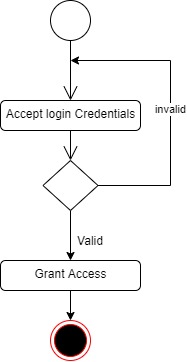
**3.3.2 Class Diagrams**



**Fig 3.2 System Class Diagram**

**3.3.3 Activity Diagrams**

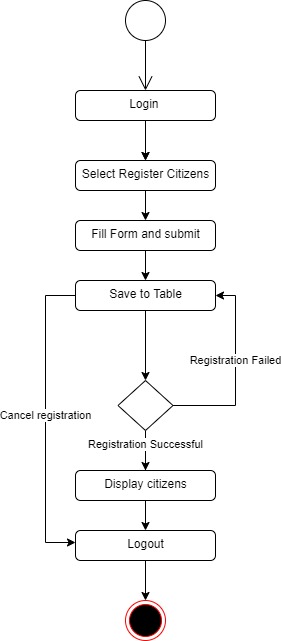
**Login**

The process for gaining access to the system is depicted in the diagram below; the email address and password must be accurate to gain access.

**Fig 3.3 System Login Activity Diagram**

**Register Citizens**

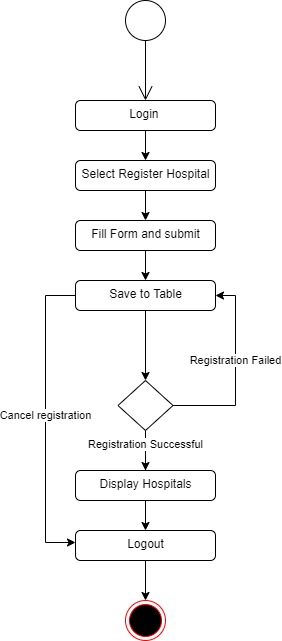
The process for registering citizens for birth certificate issuance is depicted in the diagram below; The system ensures that the user is authenticated using their registered email and password before they can be authorized to perform the registration.



**Fig 3.4 Register Citizens Activity Diagram**

**Registering Hospitals**

The process for registering hospitals is depicted in the diagram below; The system ensures that the user is authenticated and authorized to perform the registration for validity purposes.



**Fig 3.5 Hospital Registration Activity Diagram**

**3.4 Database Design**

The following are some of the input specifications used in this project work.

1. Users Table: contains the generic information of all system users.
2. Citizens Table: contains every system registered citizen’s information.

**Table 3.1 UsersInput Specification Table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Null** | **Key** | **Length** | **Description** |
| user\_id | Varchar | No | PK | 32 | Unique string for identifying users |
| username | Varchar | No |  | 20 | Unique name for users |
| password | Varchar | No |  | 128 | User Password |
| phone | Varchar | No |  | 11 | User phone number |
| pics | Varchar | No |  | 100 | User profile picture |
| is\_hospital | Varchar | No |  | 5 | Boolean to verify if a user is of hospital\_type |
| is\_superuser | Varchar | No |  | 5 | Boolean to verify if a user is an administrator |

**Table 3.2 CitizenInput Specification Table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Null** | **Key** | **Length** | **Description** |
| citizens\_id | Varchar | No | PK | 32 | Unique string for identifying citizens |
| user\_id | BigInt | No |  | - | Reference to the User table |
| CName | Varchar | No |  | 60 | Citizen full name |
| FName | Varchar | No |  | 60 | Citizen father name |
| MName | Varchar | No |  | 60 | Citizen mother name |
| DOB | Date | No |  | 20 | Citizen’s Date of birth |
| TOB | Date | No |  | 20 | Citizen Time of birth |
| weight | Double | No |  | 10 | Citizen weight at birth |
| POB | Varchar | No |  | 60 | Citizen’s place of birth |
| delivered\_by | String | No |  | 60 | The hospital that made the registration |

**3.5 Output Design**

This declares and displays the outcome of the given input. This system's output is dependent on its input. The output specification is listed below.

**Table 3.3 Users output design table**

**CENTRAL e-BIRTH MANAGEMENT SYSTEM**

**List of the System Registered Users**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **User\_id** | **Username** | **Phone** | **Phone** | **Is\_hospital** | **Is\_superuser** |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |

**Table 3.4 Allocationoutput design table**

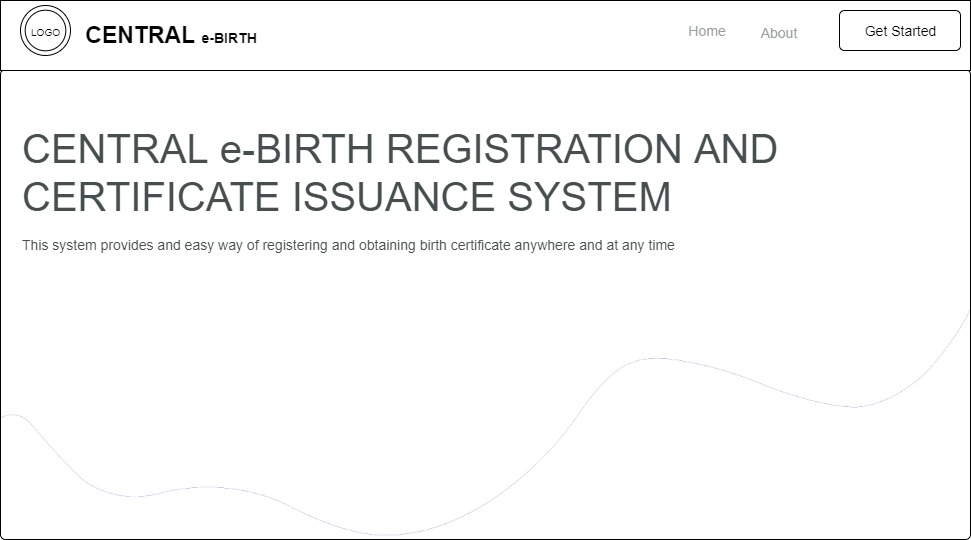
**CENTRAL e-BIRTH MANAGEMENT SYSTEM**

**List of the System Registered Citizens**

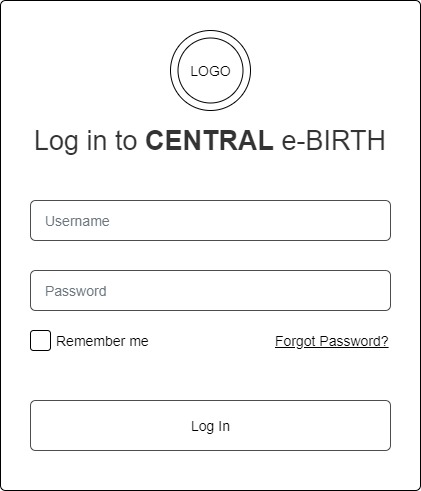
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **citizens\_id** | **user\_id** | **CName** | **FName** | **MName** | **DOB** | **TOB** | **Weight** | **POB** | **Delivered\_by** |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXX | XX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXX | XX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXX | XX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXX | XX | XXXX | XXXX | XXXX |

**3.6 Input & User Interface Design**

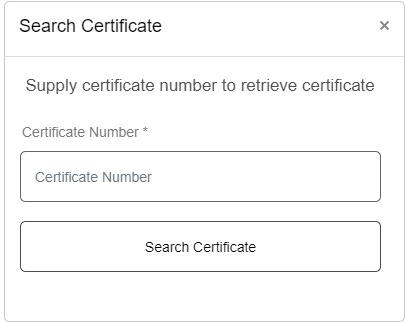
This is a graphic depiction of the system interface; it will be designed to be user-friendly, responsive, and visually beautiful. Furthermore, it will be appropriately safeguarded, thus authorization will be required to see certain levels of the information. To help with the designs, a mid-fidelity wireframing program called Draw.io is employed.



**Fig 3.6Home Page**



**Fig 3.7 Login Form**



**Fig 3.8Search Certificate**

**3.7 System Requirement**

Every software system built has a stated system requirement on which it is meant to execute for best performance. The system requirements, on the other hand, are the bare minimum of hardware and software required for the system to work properly.

**3.7.1 Hardware Requirement**

System Hardware Requirement Include:

1. Minimum of 2 GB of RAM (Random Access Memory).
2. Minimum of Intel Dual core processor.
3. Minimum of 250GB HDD (Hard Disk Drive).

**3.7.2 Software Requirement**

The software requirements include:

1. At least windows 7 OS (Operating System).
2. Vs. Code IDE installation.
3. Browsers include Chrome and Firefox.

**3.8 Choice of Programming Language**

This research project will be a web-based application built on a relational database architecture (SQLite). For frontend development, HTML (hypertext markup language), CSS (cascading style sheet), and JavaScript will be used, while Django (Python) will be used for backend programming.