**CHAPTER 1**

**GENERAL INTRODUCTION**

Information technology (IT) is the application of computers and telecommunications equipment to store, retrieve, transmit and manipulate data, often in the context of a business or other field.

Based on the storage and processing technologies employed. The technologies associated with the ubudehe management in the way of recording, collecting, and viewing data and classifying citizens in their economic –class.

**Background of the Study**

Ubudehe is process of working together to solve problems in Rwanda.The most fundamental problem that faces Rwanda today is poverty. The Government of Rwanda's response lies within trance 2020 and the Poverty Reduction Strategy. This problem is both in terms of the human taking away and weakness that poverty results in and also because it is the greatest obstacle to the sustainable economic development of Rwanda. A central plan of Government of Rwanda's policy is physical and financial recapitalization, Poverty analysis has shown that the irresistible majority of the poor in Rwanda are living in the rural continuation economy.

A group of households join together to mine their fields; acting collectively to share the load of the work and make sure that everyone is readyin time for the planting season. The concept of ubudehe is very inclusive, covering men andWomen and all social groups including the Batwa. It can also extended to those who are tooPoor or incapacitated to take part in the collective action. After the group has completed theirfields they move on to the fields of those who have not been able to participate directly.

The reach of the quality service to the Ubudehe, enhance the base for minimizing the processing costs, time taken by citizens in order to get their group of economic class, increase transparency and reduce the circulation of the citizens .

This process is performed manually. On the day of classifying citizens must be in their village and then commit of the village distribute the question paper which will help us to know their group .When the classification process is done, committee has to collect all data of each household. After collection of those paper or data the following task isto go to the cell to report and cell report to the sector after the completion of reporting, results are published to the sector in unknown time.

**Statement of the Problem**

In the current system, committee sits together with all citizens of the village and records and studies their status based on the terms of Ubudehe. Unfortunately the way this process is done currently comes with specific problems are mentioned below:

1. **Time Consuming:**This current system takes a lot of time when they are grouping people and time taken by the leaders to bring data from the cell to the sector and then to the district.
2. **Errors:** In real life humans can commit mistakes. In this current system people commit mistakes in capturing and recording data of the citizens on the forms.
3. **Inconveniencing and Costly:**In the existing system the citizens are sometimes forced to travel from their home to the office of sector to look for their economic classes.
4. **Access to Information:** In case an organization needs to know the economic class of someone it is not easy to access it. The citizen is required to go to sector’s office for the information.
5. **Loss of Documents:** the forms that contain the records of the citizens can be misplaced or lost.
6. **Corruption:**since the process involves people it is possible that a citizen may be wrongly classified in a different economic group. The committee can sometimes be subjective and influenced.
7. **Reporting:**It is currently very difficult for LODA and other users to get a comprehensive report
8. **Wasting resources:** If there is anyone from the district who needs some information about their village, they must give him/her a transport to the one of village -staff in order to bring it to the District, this is the waste of resources, time and it fatigues also.

**Choice and Motivation of the Study**

The implementation of Ubudehe Online Management Information System (UOMIS) the following interests arise:

Apply the knowledge acquired in academic studies and training in different fields.

This research provides valuable knowledge and information for the citizens and also to Gasabo District and then to the LODA.

and it provides better understanding of using a web based system implementation and the way it can be a source of increasing the delivery of better services to the citizens and also to Gasabo District and then to the LODA .

Moreover, the UOMIS web based system is going to help the citizens and also to Gasabo District and then to the LODA. To deliver better services.

Through this research an attempt to show how web based system implementation could be a valuable tool for implementing of the Ubudehe Category performance.

**Objectives of the Study**

**General objectives**

Developing web based system for Gasabo district that can help them in the classifying recording retrieving data and group publication.

### Specific Objectives

1. Analyze the ubudehe process in GASABO district and to know why people are claiming.
2. Develop a system which will automate classifying citizens according their property in each village of Gasabo District.
3. Studying, identifying and specifying the requirements of a new web based system to be implemented.
4. Creating a database to store and retrieve relevant data of the system, and a user friendly interface which will help user in Ubudehe. Creating an interface that will help user to record people.

**Scope of the Study**

Our study is going to be limited to Gasabo properties inubudehe and by technology used as we could have done an ubudehe system to be used with a telephone but will deliver ubudehe on the web.

**Methodology and Techniques Used in the Study**

**Methodology**

Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system. This was the methodology used to conduct requirements analysis and logical design of the Ubudehe Online Management Information System

**Data Collection Techniques**

Many facets are involved in conducting a research. One very essential factor is collecting data. Data can be gathered from a number of sources, which include documents, the workplace, the Internet surveys, focus groups, field notes, questionnaires and interviews.

To collect sufficient data ones need to use different methods as each tool has its specificity, but complete one another. Following are techniques that were used in collecting data.

**Observation Technique**

It is a technique commonly used in sciences. Its main advantage is that subjective partiality is eliminated if observation is done accurately. The information obtained under this method relates to what is currently happening.

We used it in order to gain information on functionalities of classification citizens this helps the leaders to know the class of each citizen and to help citizens who are poor.

**Documentation Technique**

The documentation is the reflection of the organization and the existing system. It allows to get information that none of the other tools would permit to get. The consultation of the documents on definition of tasks and responsibilities are very useful as no system development project would know how to be led to good without thoroughness documentation consultation, study and analysis.

For the purpose of gaining information, documents and books from various sources were consulted as a mean to help us in our finding.

**Interview Technique**

To know the existing system, the analyst must be going to look for information on site which means in the user population because there is information that we cannot find in written documents, records, the Internet, surveys and so forth.

Interview Technique involves presentation of oral-verbal stimuli and reply in terms of oral-verbal responses. This method can be used through personal interviews and if possible, through telephone interviews.

In order to attain highly personalized information data about the operation of the existing system and to better understand the Ubudehe management problems, we had an interview with Mrs. Mayor of Gasabo District, lawyer and other persons such as the Local Administrative Entities Development Agency (LODA).

**Expected Results**

The solution that will address the problems in the current system will be development of a web-based system (Online Ubudehe Management Information System) that will have the following features:

1. Online forms to register family and members, for indicators, questions and answers
2. A classification tool for citizens based on information given
3. A database to capture and store records, long-term
4. Easy reporting: the new system will facilitate easy generation of reports to the system users.

**Organization of the Report**

Our work contains five chapters. Chapter one provides basic information on ubudehe and presents problems which face people in Rwandan ubudehe services, interest of the work, objectives, delimitation of the study, expected results, and organization of the report.

The Second chapter focuses on the analysis of the existing system where we describe the operations in the existing ubudeheways, totally discover their issues, and suggest solutions on those problems. This chapter also describes the environment in which the new system will be implemented and data collection techniques used.

Chapter three is the logical conception of the new system. It portrays the intangible process of the solutions proposed to solve the problems of the existing system and discusses also the methods used.

The fourth chapter emphasizes on the technical realization of the application, the interpretation of the results and how it has been regarded, as well as the technologies used to build the software while chapter five concludes our project as well as recommendations for future development.

**CHAPTER 2**

**ANALYSIS OF THE EXISTING SYSTEM**

**Introduction**

A good understanding of the existing system is the key for the beginning of a new system. When you do not know the weaknesses and strengths of baptized system, you cannot know what to do.

Systems analysis and design is a proven methodology that helps the citizens and administrative (leaders) to achieve the rewards of utilizing information to its full capability.

**Background on Gasabo District**

Gasabo is a district in [Kigali Province](https://en.wikipedia.org/wiki/Kigali_Province,_Rwanda), [Rwanda](https://en.wikipedia.org/wiki/Rwanda). Its capital is in Kacyiru Sector, the town neighboring the presidential office and number of Rwanda different ministry offices. Gasabo district is divided into 15 [sectors](https://en.wikipedia.org/w/index.php?title=Sector_%28Rwanda%29&action=edit&redlink=1) : Bumbogo, Gatsata, Jali, Gikomero, Gisozi, Jabana, Kinyinya, Ndera, Nduba, Rusororo, Rutunga, Kacyiru, Kimihurura, Kimironko and Remera.

Gasabo District has a superficies of 430.30 Km2 and in 2012 has the population of530.907.

**Description of the Current System**

It is a traditional Rwandan practice and cultural value of working together to solve problems.Ubudehe is targeted to the village (*umudugudu)* level composed of about one hundred households, and small enough to promote collective action.

Ubudehe is a program using by Rwanda government in other to fit with the poverty.A group of households join together to work in their fields; acting collectively to share the trouble of the work and make sure that everyone is ready in time for the planting season.

(MINALOC: Ubudehe to Fight Poverty, 2007). The government of Rwanda has resurrected this traditional supportivemethod as the model for a program designed to advance collective action at the community level, rebuild trust and improve poverty (Dunbar 2004, 4). It draws on the understanding that the poor regularly best understand the problems they face and know their priorities, but do not have sufficient information or resources to design effective solutions.

. The word Ubudehe refers to the long-standing Rwandan practice and culture of collective action and mutual support to solve problems within a community. Today, the concept has been translated into a home grown development plan whereby citizens are placed into different categories. These categories inform the level of support families receive through government social protection program.

Ubudehe has much service but my research project is concern on the categorization of ubudehe in this program, is where citizens seat together and make each people in relative standing-economic class According to his own properties which called ubudehe categorization, the staff of umudugudu is the one who is in charge of this action in the village.

The process aims to encourage poor families’ living standards and improve social welfare.

In2014; the Local Administrative Entities Development Agency created new Ubudehe categories. Under the program, households are put in categories based on their social-economic status, and their property – in terms of land and other belongings – and what the families’ breadwinners do to earn a living. The categories are as follows:

* **Category 1:** Families who do not own a house and can hardly afford basic needs.
* **Category 2:**Those who have a residence of their own or are able to rent one but rarely get full time jobs.
* **Category 3:** Those who have a job and farmers who go beyond subsistence farming to produce an excess which can be sold. The latter also includes those with small and medium enterprises who can provide employment to dozens of people.
* **Category 4:**Those who own large-scale business, individuals working with international organizations and industries as well as public servants.

The new Ubudehe categorization process involves local leaders and communities. The community gathers and a representative from each household gives details on the families’ social and economic status. The details are provided through to a questionnaire designed by the Ministry of Local Government. After each household has finished responding thequestions related to their properties and there family, the community gathers at the cell level to crosscheck the accuracy of the information. When the community approves the information as accurate, the categorization process begins. The data collected is sent to the Sectorlevel and also check the accurate of that information then send it to the district and also check up the validity of the information which sends it to LODA and verify the accurate of the information given us then after check the validity of the information return it to the District, the District return to village

Then the village collects the errors and confirms the new one and continues to the cells level until the District then after receiving it will send to the province (KC) Kigali city, the province check it and mention the errors or confirm and return to the District if there is no errors the District send it to the LODA, LODA public ate the categories .when there is an errors the process start to village and continue until the District.

**Analysis of the Current System**

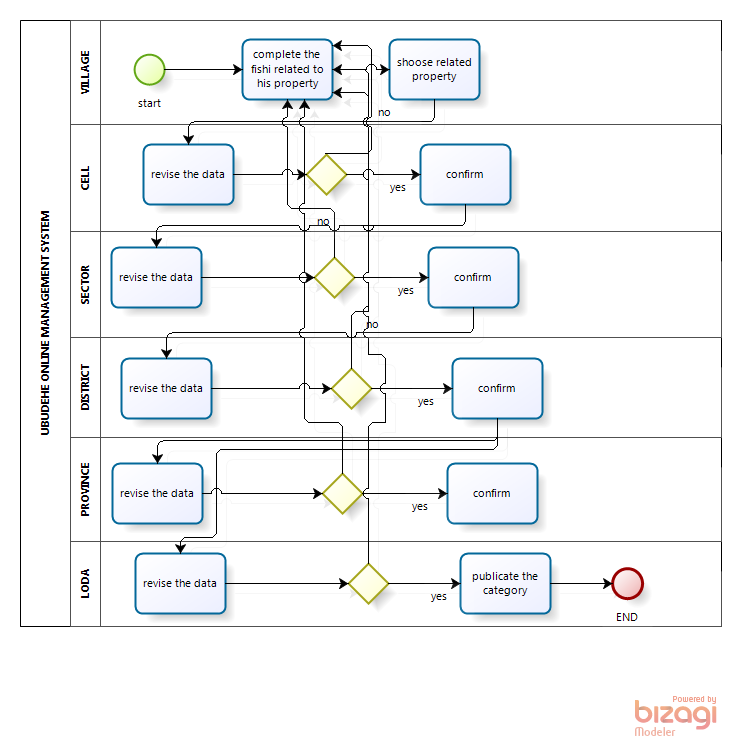
****

Figure 1: Activity Diagram of the Current System

Based on the analysis of the current system from the diagram above, it is clear that there are some inefficiencies in the way the district is conducting the Ubudehe program. The current system has problems described in the next section.

Problems of the Current System

According to the existing system analysis we found the following problems:

**It takes a long time**

This current system takes a long time in order for the people to know their current economic class

The process is to vast which means that the process begin to the village until to the District but when there is an error the superior institution will start the process until the information will be an accurate .

**Problem of running time:** This current system takes a lot of time when they are grouping citizens and time taken to the leaders to bring data from the cell, sector and districtand to the province.

**Shared Information and communication:** The current system it is difficult get information and to communicate the top level administrative and also difficult for the public institution or private when they need information .It takes a long process or long journey.

**Wasting resources:** If there is anyone from the district who needs some information about their village, they must give him/her a transport to the one of village -staff in order to bring it to the District, this is the waste of resources, time and it tires also.

**Entrusted Information:** To control what is happening to the field is difficult when they are in action to classify the citizens in different category. Because when they finished doing it they can modify or complete on other form to replace the first.

**Problem of accessibility of data:** It difficult to get data or information for the superior administrative institution for example like district and also local

**It is easy to lose data:** As long as they have no software and database to solve the problem of process information and keeping their data in security, it is easy to lose them. If the form in which they write information of each citizens is damaged this result in losing data.

**Corruption:** Since the process involves people it is possible that a citizen may be wrongly classified in a different economic group. The committee can sometimes be subjective and influenced.

**Reporting:** It is currently very difficult for LODA and other users to get a comprehensive report .when the action of classifying citizens is running, so it is difficult for the higher institution of the village to get the report because the current system work manual and that the reason why they will wait until the whole village being finished and also they need the presence of person and money for the transport.

**Proposed Solutions**

When problems have been revealed out, it is the beginning of their solutions. According to the problems said above we are proposing to give the following solutions:

* I will make software that will decrease time it takes to get the needed information by using a web based application.
* The new system will decrease the cost using by the citizens in the time of going to the sector to see he/she is Economic-class and also the money spend by the Leaders to report to the higher public institution .
* It will reduce the error in Ubudehe system because there is a database and a web based application it is very easy to correct it.
* The new system which I will make it will help the public and private institution to get the report at any time he/she is needed and also anywhere he/she is.
* The new application software will help the Gasabo district and LODA to keep data permanently.
* The new system will categorize the citizens according to their property.

**CHAPTER 3**

**REQUIREMENT ANALYSIS AND DESIGN OF THE NEW SYSTEM**

**Introduction**

Good system design is impossible without careful, perfect system analysis .The design phase typically cannot begin until the analysis work is complete .It is better to complete the analysis phase before moving on to systems design.

The development of a system is a work which requires much effort and attention. Actually, the main goal of a new system is to satisfy the needs of its users by solving problems they face with the existing system. Deep analysis of users’ needs will most of the time lead to a useful software development as a system might give perfect result.

System development can generally be thought of as having two major components:

* System Analysis and,
* System Design

The System AnalysisIncludes the output that must be produced by the system, the input needed by the system, and the process that must be produced by the system without regard to how tasks will be accomplished physically.SystemDesignis a plan for the actual implementation of the system. The system design is built on the system’s analysis design and describes a specific implementation; much like a working blue print describes the actual construction of a building.

In this chapter, we analyzed requirements then came out with a design that will form the blue print to the actual solution to the problem in hand. But before getting into this in details, let us start by discussing the techniques and tools that were used for that end.

### Unified Modeling Language (UML)

UML is a graphical language for capturing the artifacts of software developments. It represents a collection of best engineering practices that have proved success in the modeling of large and complex systems. The language provides us with the notations to produce models and, is explicitly designed to be implemented by computer-assisted software engineering (CASE) tools.

Naturally, some aspects of methodology are implied by the elements that comprise a UML model, but UML itself just provides a visual syntax that we can use to construct models. It does not give us any kind of modeling methodology.

### Models of UML

A **model** is a subjective and pertinent representation of the reality. It is easier to refer on a model than to refer on the reality because the model represents just essential aspects of the reality and ignores the useless aspects.

The UML provides many different models for a system such as use case diagram, class diagram, sequence diagram, etc. Following is a list by Ariadne Training Limited (Ariadne Training Limited, 2001) of UML models with a one sentence summary of the purpose of the model:

* **Use Cases** - How will our system interact with the outside world?
* **Class Diagram** - What objects do we need? How will they be related?
* **Collaboration Diagram** - How will the objects interact?
* **Sequence Diagram** - How will the objects interact?
* **State Diagram** - What states should our objects be in?
* **Package Diagram** - How are we going to modularize our development?
* **Component Diagram** - How will our software components be related?
* **Deployment Diagram** - How will the software be deployed?

Actually, it is not required to make all the models for a system; instead, it is advised to model the ones that are to visualize sufficiently the system.

When we model a system, we must not only identify the things that form the vocabulary of the system but also model how these things stand in relation to one another.

### Relationships

Relationships allow us to show on a model how two or more things relate to each other. The role that relationships play in UML models is to allow us to capture meaningful (semantic) connections between things. Relationships apply to the structural and grouping things in a model and are depicted in the figure 2 below by Jim and Ila, 2002.

* **Association** is a structural relationship that specifies that objects of one thing are connected to objects of another. Given an association connecting two classes, objects of one class can be related to objects of the other class. It may be refined into an aggregation relationship or a stronger form of aggregation known as the composition aggregation or simply composition relationship.
* **Aggregation** – this is a loose relationship between objects – an example might be a computer and its peripherals. Its illustration is as shown in figure 3.



* **Composition** – this is a very strong type of relationship between objects – it is like a tree and its leaves. Its illustration is as shown in figure 4.



* **Dependency** is a relationship that states that one thing uses the information and services of another thing, but not necessarily the reverse. A dependency exists between two elements; changes to the definition of one element may cause changes to the other.



* **Generalization** is a relationship between a general kind of thing (called the superclass or parent) and a more specific kind of thing (called the subclass or child). An object of the child class may be used for a variable or parameter typed by the parent, but not the reverse. A child inherits the properties of its parents, especially their attributes and operations.

### Multiplicity (Cardinality)

Kinds of multiplicity are the symbols which indicate the number of instances of one class linked to one instance of the other class and are placed at the end of relationships. It is written as an expression with a minimum and maximum value, two dots are used to separate the minimum and maximum values. (Briefly is the number of objects that participate in the relationship).

### Unified Process (UP)

Unified process (**UP**) is a software development process (SDP) also known as software engineering process (SEP), SDP is the process in which we turn user requirements into software. It tells the workers, activities, and artifacts that is needed to utilize, perform or create in order to model a software system (It defines who, what, when, and how of a software development).

UML is not tied to any specific kind of software development process, and indeed it is capable of being used with all existing development processes. UP uses UML as its underlying visual modeling syntax and we can therefore think of UP as being the preferred process for UML, as it is the best adapted to it , but UML itself can provide the visual modeling support for other process like Water fall process, prototype model, V model, spiral process and so on.

One of its characteristics is that UP is an iterative and incremental development process. Iterative aspect means that the project has to be broken into small subprojects (called iterations) which are easier to manage and to complete successfully. UP is incremental because each iteration generates a baseline that comprises a partially complete version of the final system and any associated project documentation.

UP phases (Inception, Elaboration, Construction and Transition) are divided into a series of time boxed iterations. Each iteration results in an increment, which is of the system that contains added or improved functionality compared with the previous release.

### Phases of UP

The Unified Process consists of cycles that may repeat over the long-term life of a system. A cycle consists of four phases: Inception, Elaboration, Construction and Transition. Each cycle is concluded with a release, there are also releases within a cycle. Let's briefly review the four phases in a cycle:

Design of the New System – Diagrams

After getting all customer needs or requirements we have to analyze them, doing so different models or diagrams are used. At this particular point we will focus on use case diagram. Use case diagram describes the functionality provided by a system in terms of actors, their goals represented as use cases, and relationships between actors and use cases. The followings are components of a use case model:

USE CASE DIAGRAM



|  |  |
| --- | --- |
| **Description** | **Shape** |
| **An actor:**   * Is a Person or system that derives benefit from and is external to the subject. * Is depicted as either a stick figure (default) or if a nonhuman Acton is involved, as a rectangle with « actor » in it (alternative). * Can be associated with other actors using a specialization super class association, denoted by an Arrow with a hollow arrow head. |  |
| **A use case:**   * Represents a major piece of system functionality * Can extend another use case * Can include another use case. * Is placed inside the system boundary * Is labeled with a descriptive verb–noun phrase. |  |
| **A boundary:**   * It is a box drawn around the use case to denote the edge or boundary of the system being modeled. * Includes the name of the subject inside or on top. * Represents the scope of the subject, e.g., a system or an individual business process. |  |
| **An association relationship:**   * Links an actor with the use case(s) with which it interacts. |  |
| **An include relationship:**   * Represents the inclusion of the functionality of one use case within * Has an arrow drawn from the base use case to the used use case? |  |
| **An extend relationship:**   * Represents the extension of the use case to include optional behavior. * Has an arrow drawn from the extension use case to the base use case? |  |
| **A generalization relationship:**   * Represents specialized use case to a more generalized one. * Has an arrow drawn from the specialized use case to the base use case? |  |

**Use-case description**

Use Case description details what a use case do, and what it requests in order to be well executed. Each use case looks like this:

* **Name :** a name of a use case
* **Description:** what a system intends to do
* **Actor:** the actor involved in the use case
* **Pre-condition:** the system state before the use case can begin
* **Post-condition:** the system state when the use case is over
* **Normal flow:** the actual steps of the use case
* **Alternative flow**: steps which may happen in case a normal flow fails.

|  |
| --- |
| **Main normal flow:**   1. The Administrator logs in the system with correct credentials 2. The system gives the page 3. The Administrator register people(s) |
| **Alternative flow:**  1.A. if Administrator fills an existing people  1. The system gives the message that the people already exist. |

Table 1

Creating account Use Case Description

|  |
| --- |
| **Use case:** Creating users |
| **Goal:** Help Gasabo to give access on data to users |
| **Actor:** The person who interact with the system |
| **Pre-condition:** the admin must have all identifications of the people or the user to add |
| **Post-condition:** the system should display the admin page which shows the menu of how to use the web application. |
| **Main normal flow:**  1.Admin requests System to bring pages that allow to insert new branch or new user  2. System displays a form to fill the identification information,  3. Admin fills in the requested data and controls the whole system,  4. System validates data,  5. System adds data in database  6. The user(s) have username to log in the system |
| **Alternative flow:**  1.A. if Administrator fills an existing username  1. The system gives the message that the user already exist. |

Table 2

|  |
| --- |
| **Use case:** user login |
| **Goal:** Help user to give access on data |
| **Actor:** The person who interact with the system |
| **Pre-condition:** the Vice Mayor in charge of Economic must have all identifications of the user to login and interact with the system |
| **Post-condition:** the system should display the admin page which shows the menu of how to use the web application. |

Table 3

|  |
| --- |
| **Use case:** register ubudehe\_category |
| **Goal:** Help user to record the category of ubudehe in the database |
| **Actor:** The person who interact with the system |
| **Pre-condition:** the cell staff must login first and start to register people |
| **Post-condition:** the system should display the ubudehe Category page which shows the interface of entering the category exist in ubudehe data. |

**Table 4**

|  |
| --- |
| **Use case:** register Family |
| **Goal:** Help user to record the identification of the family. |
| **Actor:** The person who interact with the system |
| **Pre-condition:** the cell staff must login first and start to register family |
| **Post-condition:** the system should display the family page which shows the interface of family where help to record all the identification. |

**Table 5**

|  |
| --- |
| **Use case:** Record members |
| **Goal:** Help user to record the members of the family |
| **Actor:** The person who interact with the system |
| **Pre-condition:** the cell staff must login first and start to register people |
| **Post-condition:** the system should display the members page which shows the interface of members where help to record all the member of the family. |

**Table 6**

|  |
| --- |
| **Use case:** Register questions. |
| **Goal:** Help user to register questions |
| **Actor:** The person who interact with the system |
| **Pre-condition:** the cell staff must login first and start to register questions |
| **Post-condition:** the system should display the ubudehe category page which shows the interface of ubudehe category where help to record all the ubudehe category of the family. |

**Table 7**

|  |
| --- |
| **Use case:** Register ubudehe Data. |
| **Goal:** Help user to register the ubudehe data of the family |
| **Actor:** The person who interact with the system |
| **Pre-condition:** the cell staff must login first and start to register the family and the related data. |
| **Post-condition:** the system should display the the report page of the family which shows the interface of ubudehe category of the family. |

**CLASS DIAGRAM**



**Class diagram description**

**Class No 1 indicators**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| indicatorID | String | Not null | Unique identification of indicator |
| indicator | string | Not null | name of indicator |

**Class No2 indicator\_category**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| ubuidID | String | Not null | Unique identification of indicator\_category |
| indid | String | Not null | Unique identification of indicator\_category |
| significance | String | Not null | Signification of indicator |

**Class No3 Family**

|  |  |  |  |
| --- | --- | --- | --- |
| fid | String | Not null | Unique identification of family |
| Headfamily | String | Not null | The representative of the family |
| sector | string | Not null | Sector which that family live in |
| village | String | Not null | The village of that family |
| district | String | Not null | The District in which that family live |
| Province | String | Not null | The province in which that family live |
| cell | String | Not null | The cell of the family |
| headnid | String | Not null | The nation id of the representative of the family |
| regdate | Date | Not null | Registration date |
| ubuid | String | Not null | The unique identification of his ubudehe\_category |

**Class No4Indicator\_questions**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| qid | String | Not null | Unique identification of indicator\_questions |
| Indid | String | Not null | Unique identification of indicator\_questions |
| truevalue | String | Not null | The status of indicator |

**Class No5ubudehe\_category**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| ubuid | String | Not null | Unique identification of Ubudehe\_category |
| category | String | Not null | Name of the category |

**Class No6members**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| memberid | String | Not null | Unique identification of member |
| relationship | String | Not null | Relationship has in his family |
| gender | String | Not null | The gender him/her |
| dob | String | Not null | The date of birth |
| mstatus | String | Not null | The marital status |
| nidstatus | String | Not null | The nation id status |
| nid | String | Not null | The nation id |
| job | String | Not null | He /she has a job |
| notjobreason | String | Not null | Why he/she doesn’t have a job |
| fid | String | Not null | Unique identification of his family |

**Class No7user**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| userid | String | Not null | Unique identification of user |
| name | String | Not null | Name of user |
| cell | string | Not null | Cell of user |
| type | String | Not null | Role of user |
| username | String | Not null | Username of the user |
| password | String | Not null | Password of the user |

**Class No8 ubudehe\_data.**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| userid | String | Not null | Unique identification of user which register ubudehedata |
| fid | String | Not null | The unique identification of the which assign to ubudehe data |
| ubuid | string | Not null | The related ubudehe category |
| opid | String | Not null | The option of him/her |
| indid | String | Not null | The indicator id of him/her |
| qid | String | Not null | The question id of him/her (response) |
| optionStatus | String | Not null | The status of ubudehedata |

**Class No9 Option.**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| opid | String | Not null | Unique identification of option |
| qid | String | Not null | Question unique identification |
| option | String | Not null | Option name |
| indicator Status | String | Not null | The status of indicator |

**Class No10 Indicator question.**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTRIBUTE NAME** | **DATA TYPE** | **CONSTRAINT** | **DESCRIPTION** |
| qid | String | Not null | Unique identification of question |
| name | String | Not null | The name of the question |

**Sequence Diagram**

A Sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

The symbols below are used in sequence diagram:

* **Object**



* **Activation**

 : Activation

* **Messages**

or



## References

1. [**Jump up^**](https://en.wikipedia.org/wiki/Sequence_diagram#cite_ref-1) OMG (2011). [OMG Unified Modeling Language (OMG UML), Superstructure, V2.4.1](http://www.omg.org/spec/UML/2.4.1/Superstructure/PDF/), p. 507.
2. [**Jump up^**](https://en.wikipedia.org/wiki/Sequence_diagram#cite_ref-2) OMG (2008). [OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2](http://www.omg.org/spec/UML/2.1.2/Superstructure/PDF), p. 485.
3. [**Jump up^**](https://en.wikipedia.org/wiki/Sequence_diagram#cite_ref-3) OMG (2008). [OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2](http://www.omg.org/spec/UML/2.1.2/Superstructure/PDF). p. 467.

**Database Diagram**



**System architecture diagram**

System architecture diagram is a drawing of a system, in which the main parts or functions are symbolized by blocks connected by lines that show the communications of the blocks.

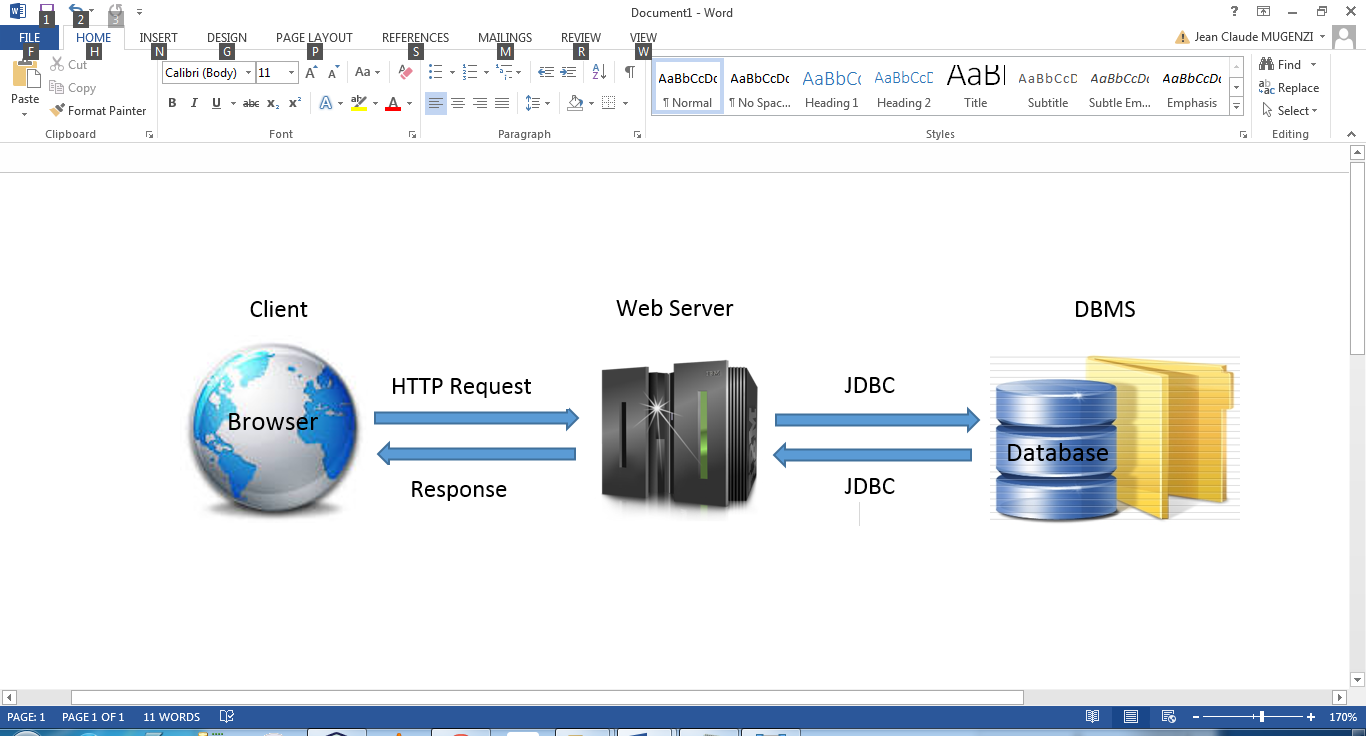


Figure 6: Model view controller architecture

This structure can be explained by the MVC model (model view controller). Model view controller (MVC) is a very useful and popular design pattern. Since we’re writing software, we should know it. Fortunately it’s also one of the simplest to truly understand

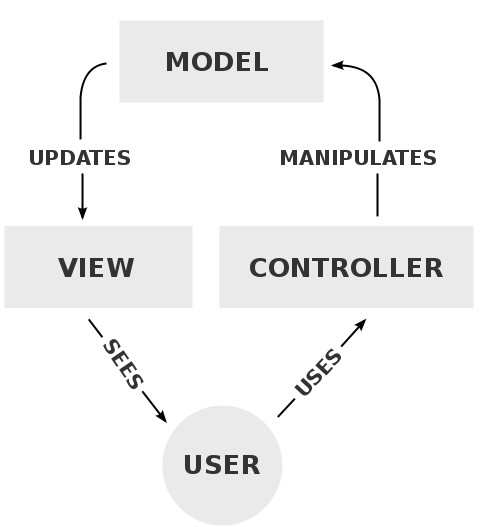
****

Figure 7: Model view controller architecture

1. The **model** represents the data, and does nothing else. The model does NOT depend on the controller or the view.
2. The **view** displays the model data, and sends user actions (e.g. button clicks) to the controller. The view can:

* be independent of both the model and the controller; or
* Actually be the controller, and therefore depend on the model.

3. The **controller** provides model data to the view, and interprets user actions such as button clicks. The controller depends on the view and the model. In some cases, the controller and the view are the same object.

To understand well let’s take an address book application as an example. The model is a list of family objects, the view is a GUI window that displays the list of family, and the controller handles actions such as “Delete family”, “Add member” etc. The following example does not use MVC because the model depends on the view.