IDIR

Local Idir managing application based on basic java and database SQL

Goals

1. The program will try to achieve the ease access of the Idir system which is exercised locally in Ethiopia.

Traditionally the system and all the data are managed through multiple hand-written large record books. Which can lead to repeated data, multiple unnecessary record books which get full every time $ must be replaced with new ones continuously which also may cause loss of data. As well as it is not very efficient for accessing and retrieving any information.

Thus our program will try to overcome these problems. Basically what our program:-

* Have an interactive GUI that can easily be understood and used by the wide society.
* Have a database for the data of the members of the Idir and can easily retrieve, manipulate, access and add information in the database

1. We will try to work on it this summer and finalize it by the end of this year or if takes long as far as meskel.

And we will use Java for the application program with java swing for the GUI and SQl for the database (probably with MySQL) and also will work on through GitHub for version control and collaboration.

End Users

The program is designed to be used by many Idirs from small to large based on their respective needs.

The end user are the ones who are at the writing desk of Idir houses who receive monthly payments from members and give compassionate (ማስተዛዘኛ) money for the affected, and for those who manage the Idir’s properties like tents, chairs, tables, etc.

Also the persons who audit, calculate incomes and spending (ሂሣብ ያዥ)

\*(other roles in the Idir to be identified)

Plan – Steps

Back end

1. Identifying the entities in the Idir system and build based on the relational database model and build everything from identifying entities to making ER-Diagrams to mapping to relational tables (1st week)
2. After making the theoretical work of the database then build the database by using SQL in MySQL database management system.

(Which we have to learn from scratch btw) (2nd week)

Front end

1. Then we start off by building the basic skeletal structure of the application program by using java
2. Identifying what the objects in the GUI of the program do and how the function (3rd week)

Connecting

1. Connecting the java program with the database making the objects in the GUI function properly and effectively and finalizing. (4th week)
2. Testing (5th week)

Conceptual Design

The Idir is organized of many members and official selected from those members and the idir also have many properties like offices, tents, chairs, etc.

This Idir has 6 entities

1. **MEMBER**

An entity type MEMBER with attributes

* Full name – Composite (first name, father’s name, grandfather’s name), not null
* Member ID – **primary key**, a number assigned for members in order of their joining of the idir starting from the first member as 1 and automatically increased as more members join
* Address – composite (wereda, kebele, house No)
* Phone No – atomic the cell phone number of the member
* Age - atomic
* Occupation – atomic can be null
* Religion – atomic
* Photo

Domains

Full name – each 3 components will have a varchar of 15 characters

Member ID – a char of 3 numbers like ‘012’, ‘221’

Address – a string of ‘wereda, kebele, house no’ like ‘shenkor, 10, 551’ a varchar of 30 characters

Phone No – a char of 10 numbers like ‘0912345678’

Age – int

Occupation – varchar of 20 letters

Religion – varchar of 15 letters

Photo – a file location to the location of the image a varchar of 100 characters

1. **THE IDIR (the organization itself)**

The organization itself has many attributes thus we take it as an entity itself.

* **Official name** – the name of the Idir in establishment

Atomic and not null but unique from any other. Primary Key

* **Bank account No**- an account no that the idir uses to store the money collected.
* **Office Address** – the address where the idir basically resides and where the money is collected and many more
* **Store Address** – the address where the properties of the idir are stored
* **Chairman** – the member ID of the person who serves as a chairman of the idir. The chairman is the person who oversees and represents the idir as whole
* **Vice Chairman** - the member ID of the person who serves as a vice chairman. This person serves as supportive of the chairman and as placeholder in case of the absence of the chairman
* **Secretary** - the member ID of the person who serves as secretary. The secretary is the one who prepares meeting titles and takes notes on these meetings and reports.
* **Math personnel** (ሂሣብ ሹም) - the member ID of the person who serves as math personnel. This person controls all income receipts and all money movements
* **Main money holder** - the member ID of the person who serves as money holder. This person receives money from the math personnel and stores it in the Idir’s bank account.
* **Daily money collector** - the member ID of the person who serves as money collector. This person collects money from members in monthly or other payments and gives the money to the money holder through the math personnel
* **Property buyer** - the member ID of the person who serves as property holder. This person buys things that the Idir needs
* **Shift supervisors** – multivalued – 3 - the member IDs of the persons who serves as shift supervisors. These persons who supervise funerals and many arrangements in the funeral and other ceremonies
* **Auditors** – multivalued - 2 - the member IDs of the persons who serves as auditors. These persons are the ones who audit the workings of the other officials and mainly the money income/outcome of the idir at least twice a year
* **Starting date** - current officials starting date
* **Rules & Regulations –** this is a word or text file where inside it are the rules and regulations with which the Idir is based and works.

All the attributes from chairman down (except starting date & rules and regulations) are foreign keys from the MEMBER entity

Domains

Official Name – a varchar of 50 characters

Bank account No – a varchar of 15 numbers

Office & store address - a string of ‘wereda, kebele, house no’ like ‘shenkor, 10, 551’ a varchar of 30 characters

Starting Date – Date in GC

Rules & regulations – a file location to text file a varchar of 100

All the rest are foreign keys from the MEMBER entity a char of 3 numbers

1. **PROPERTY**

* Type –the type of the property like chair, table, tent etc. primary key
* Number of items – the quantity of the item
* Number of items in store – the quantity of items which are currently left in the store and not given to members
* Individual price – price of one piece of that type, in case of breaking or losing of that item by a member to be paid of that amount

Domain

Type – varchar of 15 characters

Number of items – int

Number of items in store – int

Individual price – decimal with 6 digits before decimal point and 2 after

1. **RECEIPT**

* Receipt No – primary key- no generated from the 1st receipt as 1 and continuously increased
* Date – the date of receipt issued
* Payer ID – the member ID of the payer - foreign key
* Reason for payment – the reason for payment in less than 3 words
* Amount – the amount of money paid
* Money Receiver – The collector the money the receipt paid for

Domain

Receipt No – a char of 6 characters like 000001, 123103 or 001230

Date – Date

Payer ID – foreign key from the MEMBER a char of 3 numbers

Reason for payment – a varchar of 30 characters

Amount - decimal with 4 digits before decimal point and 2 after

Money Receiver - foreign key from the MEMBER a char of 3 numbers

1. **FAMILY**

The families of the member where the member gets compassion money in case one of the families decease. A weak entity

* Full name – composite – the name of the family. Partial key
* Relationship – the relationship the family have with the member (Mother, Father, Child or Siblings and include in-laws or spouse’s Mother, Father or Siblings)
* Member ID – the member in which it depends on – foreign key
* Phone No – the Phone No of the family

Domain

Full name – each 3 components will have a varchar of 15 characters

Relationship – a varchar of 20 characters

Member ID - foreign key from the MEMBER a char of 3 numbers

Phone No - a char of 10 numbers like ‘0912345678’

1. **Agenda**

It is defined as the agendas or notes that are taken in meetings and other gatherings

* Agenda No – number generated from 1 that can be used as an identifier for the agenda – Primary Key
* Date – the date when the agenda was taken
* Title – title for the meeting where the agenda was taken
* Writer – the person who wrote the agenda – Foreign key
* Text – the word or text file of the agenda written (the main text)

Domains

Agenda No – int

Date – Date

Title – a varchar of 20 characters

Writer - foreign key from the MEMBER a char of 3 numbers

Text – a text file

**Relationships**

There are relationships

1. SERVES – a relationship between a MEMBER and IDIR where the member serves as some official.

This relationship has an attribute - Role Name – which specifies the role the member plays in the Idir in the SERVES relationship.

The relationship is created by importing the primary key of the MEMBER - Member ID – in to the IDIR entity

1. HAS\_FAMILY – a relationship between MEMBER and FAMILY where the families are the family of the member.

The relationship is created by importing the primary key of the MEMBER - Member ID – in to the FAMILY entity

1. PAYES – a relationship between MEMBER and RECEIPT where the member pays the money issued in the receipt

The relationship is created by importing the primary key of the MEMBER - Member ID – in to the RECEIPT entity.

1. HAS\_PROPERTY – a relationship between IDIR and PROPERTY where properties are what the Idir possesses.

The relationship is created by importing the primary key of the IDIR – Official Name– in to the PROPERTY entity.

1. HAS\_AGENDA – a relationship between IDIR and Agenda where an agenda is taken in an Idir.

The relationship is created by importing the primary key of the IDIR – Official Name– in to the AGENDA entity.

1. WRITES - a relationship between MEMBER and Agenda where an agenda is written by a member.

The relationship is created by importing the primary key of the MEMBER – member ID – in to the AGENDA entity.

1. Receives – a relationship between MEMBER & RECEIPT entity where the member receives the money issued in the receipt.

The relationship is created by importing the primary key of the MEMBER - Member ID – in to the RECEIPT entity.

1. Monthly Payment History

This will be a description of the history of a member’s payment throughout months & years. This will be a relationship between MEMBER & RECEIPT entities. And there will be a different table

Relationship Constraints

1. Cardinality ratio – MEMBER:IDIR – 1:1

Participation constraint – MEMBER participation is partial while the IDIR is total participation

1. Cardinality ratio – MEMBER:FAMILY – 1:N

Participation constraint – both entities participation is total.

1. Cardinality ratio – MEMBER:RECEIPT – 1:N

Participation constraint – both entities participation is total.

1. Cardinality ratio – IDIR:PROPERTY – 1:N

Participation constraint – both entities participation is total.

1. Cardinality ratio – IDIR:AGENDA – 1:N

Participation constraint – both entities participation is total.

1. Cardinality ratio – MEMBER:AGENDA – 1:N

Participation constraint – AGENDA entity participation is total while the participation of MEMBER is partial.

1. Cardinality ratio – MEMBER:RECEIPT – 1:N

Participation constraint – MEMBER entities participation is partial while RECEIPT is total.

ER-diagram to relational table mapping

MEMBER

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Member ID | Full Name | Address | Phone No | Age | Occupation | Religion |

IDIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Official Name | Office  Address | Store Address | Bank Account | Chairman | Vice  Chairman | Secretary | Math Personnel | Main  Money  Holder | Daily Money Collector | Property Buyer | Shift Supervisors  (1,2,3) | Auditors  (1,2) | Starting Date | Rules & Regulations |

PROPERTY

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Number of Items | Number of Items in store | Individual price |

RECEIPT

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Receipt No | Payer ID | Date | Reason | Amount | Money Receiver |

FAMILY

|  |  |  |  |
| --- | --- | --- | --- |
| Full Name | Member ID | Relationship | Phone No |

AGENDA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Agenda No | Date | Title | Writer | text |

Monthly Payment History

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Member ID | Sep  2016 | Oct  2016 | Nov  2016 | Dec  2016 | Jan  2016 | Feb  2016 | Mar  2016 | Apr  2016 | May …  2016 |

#each value will be the Receipt No for the payment of that month.