1. INTRODUCTION

SNDP Karayogam Management System aims to develop a particular Karayogam for store their activity as online. The system is only for a particular karayogam. The karayogam is based on Taluk. In each karayogam there is a president, secretary, treasurer and members. In each karayogam has its own programs and financial details. Now the karayogam authorities store their financial details, members details, program scheduling details and other details as records. The aim of the project is reduced the time and work. That is the activities of SNDP karayogam is convert into online system. The main advantage of online system is store the financial details, program scheduling, members details as online and it is safe. Also it reduce the time and handworks. The SNDP Karayogam Management System (SNDPKMS) is technologically advanced features and user-friendly interface, SNDPKMS provide how the SNDP Karayogam works, their activity and other details about the Karayogam. The main activity of SNdP Karayogam is fund collection, loan and paraspara sahaya nidhi. Provide scholarship for students. Also the election is done online.

2. SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the facts to improve the system. Analysis is a detailed study of various operations performed by a system and their relationship within and outside of the system .This involves gathering information and using structured tools for analysis. System analysis is a step-by-step process used to identify and develop or acquire the software need to control the processing of specific application. System analysis is a continuing activity the stages of the systems development. System analysis is the process of gathering and interpreting facts, diagnosing problems and using the facts to improve the system. The outputs from the organization are traced through the various processing that the input phases through in the organization. This involves gathering information and using structured tools for analysis. A detailed study of this process must be made by various techniques like interviews; questionnaires etc. It is necessary to have such a good system analysis and then by a project development cycle so that the project can be completed in a strictly manner and able to finish with the desired time. The analyst must be so careful about his responsibilities. As the next step the current system analysis is done which identifies the real need of establishing our project in the environment, its opportunities and constraints etc. All of the steps discussed above are collectively known as the system analysis.

2.1. EXISTING SYSTEM

The existing system of SNDP Karayogam is manual. The all activities of SNDP Karayogam is store as records or files. For apply the loan, scholarship and paraspara sahaya nidhi, the member or relatives want to go the authorities and search the scheme and type final give a application for the loan, scholarship and paraspara sahaya nidhi. The monthly and weekly fund collection are done offline. Also the voting is done by offline. The authorities check the applications of member or relatives and grant it. SNDP Karagoyam keeps records and reports as files

LIMITATIONS OF EXISTING SYSTEM

- Records are store as files.
- More time and work.
- Report generation become difficult because all the data are stored in registers
- There is a chance for the loss of data
- Time consuming for apply loan, scholarship and paraspara sahaya nidhi.

2.2. PROPOSED SYSTSEM

All activities of SNDP Karayogam Management System is converted into online. Activities such as event scheduling, add a new member, election, scheme adding, paraspara sahaya nidhi, monthly and weekly fund collection and scholarship details.

There are 4 users in the system :

i. Admin
ii. Members
iii. Treasurer
iv. Guest

- Add a new member : The administrator is added the new member into the system.
- Add relatives : The member is add their relatives.
- Event Scheduling : Schedule the annual program for karayogam members and its manage its financial details by financial head.
- Monthly and weekly collection funds : Monthly and weekly collection funds add by the finance head and member or relatives pay online.
- Loan : Loan types are added by administrator and members apply through online and finance head accept or reject the application finally grand applications. Finance head generate a report of the loan applications.
- Paraspara sahaya nidhi : Scheme are added by administrator and members and relatives apply through online and finance head accept or reject the application finally grand the applications. Finance head generate a report of the Paraspara sahaya nidhi applications.

• Election

2.3. System Requirment Specification (SRS)

2.3.1. SOFTWARE SPECIFICATION

❖ Operating System : Windows XP Professional or Higher

❖ IDE : Visual Studio code

❖ Front-end : Python Django

❖ Scripting Language : HTML , Java Script

❖ Back-end : Microsoft SQL SERVER 2010

❖ Browser : Internet Explorer, Mozilla Firefox, Google Chrome

2.3.2. HARDWARE SPECIFICATION

Selection of hardware configuration is very important task related to the software development. The processor should be powerful to handle all the operations. The hard disk should have the sufficient capacity to solve the database and the application.

❖ Processor : Intel ® dual core ™ or higher

❖ System Bus : 64 BIT

❖ RAM : 4GB or higher

❖ Hard : 60GB or higher

2.3.3 Front End

- PYTHON

Python is a general purpose, dynamic, high-level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures. Python is easy to learn yet powerful and versatile scripting language, which makes it attractive for Application Development. Python's syntax anddynamic typing with its interpreted nature make it an ideal language for scripting and rapid application development.

Python supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles. Python is not intended to work in a particular area, such as web programming. That is why it is known as multipurpose programming language because it can be used with web, enterprise, 3D CAD, etc. We don't need to use data types to declare variable because it is dynamically typed so we can write a=10 to assign an integer value in an integer variable. Python makes the development and debugging fast because there is no compilation step included in Python development, and edit-test-debug cycle is very fast.

o Software Development
o Artificial Intelligence
o Web Applications
o Enterprise Applications
o 3D CAD Applications
o Machine Learning
o Computer Vision or Image Processing Applications.
o Speech Recognitions

• FEATURES OF PYTHON

Python provides many useful features which make it popular and valuable from the other programming languages. It supports object-oriented programming, procedural programming approaches and provides dynamic memory allocation. We have listed below a few essential features. 1) Easy to Learn and Use Python is easy to learn as compared to other programming languages. Its syntax is straightforward and much the same as the English language. There is no use of the semicolon or curly-bracket, the indentation defines the code block. It is the recommended programming language for beginners.

2) Expressive Language Python can perform complex tasks using a few lines of code. A simple example, the hello world program you simply type print("Hello World"). It will take only one line to execute, while Java or C takes multiple lines.

3) Interpreted Language into existence. It supports inheritance, polymorphism, and encapsulation, etc. The object oriented procedure helps to programmer to write reusable code and develop applications in less code. Python is an interpreted language; it means the Python program is executed one line at a time. The advantage of being interpreted language, it makes debugging easy and portable.

4) Cross-platform Language Python can run equally on different platforms such as Windows, Linux, UNIX, and Macintosh, etc. So, we can say that Python is a portable language. It enables programmers to develop the software for several competing platforms by writing a program only once.

5) Free and Open Source Python is freely available for everyone. It is freely available on its official website "www.python.org". It has a large community across the world that is dedicatedly working towards make new python modules and functions. Anyone can contribute to the Python community. The open-source means, "Anyone can download its source code without paying any penny."

6) Object-Oriented Language Python supports object-oriented language and concepts of classes and objects come.

• PYTHON FRAMEWORKS

Python is a highly demanded and effective programming language, and it contains vast libraries and frameworks which for almost every technical domain. The frameworks provide easiness while developing an application (or software) to developers. They offer automatic implementation of redundant tasks, reduce development time, and focus significantly on application logic rather than a common element. Below is the list of top Python frameworks.

a) Bottle

It is a micro framework that creates a single source of file for every developed application using it. This framework primarily focuses on developing APIs, and it is one of the finest Python web-framework. The Bottle doesn't require the other dependencies to create small web applications. It allows the developer to work closely with the hardware, and it is the most suited framework to make small apps and simplistic person-use apps.

b) CherryPy

CherryPy is an open-source, object-oriented micro-framework. It follows the minimalistic approach. This is one of the oldest Python frameworks that were introduced in 2002. We don't need to install the apache server to run cherryPy. This framework allows us to use any technology for data access.

c) Dash

It is a Python-based micro framework used to develop analytical web applications and popular among data scientists. It is suitable for those data scientists who aren't much into web development mechanics.

d) Flask

Flask is another popular Python micro-framework that comes under the BSD license. The Sinatra Ruby framework inspires it. This framework requires Jinja2 templates and the Werzeug.

• DJANGO FRAMEWORK

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Django makes it easier to build better web apps quickly and with less code. It is based on MVT (Model View Template) design pattern. The Django is very demanding due to its rapid development feature. It takes less time to build application after collecting client requirement. Django comes with the following design philosophies:

o Loosely Coupled

o Less Coding

o Don't Repeat Yourself (DRY)

o Fast Development

o Clean Design

Advantages of Django:

1) Object-Relational Mapping (ORM) Support: Django provides a bridge between the data model and the database engine, and supports a large set of database systems including MySQL,

Oracle, Postgres, etc. Django also supports NoSQL database through Django-nonrel fork. For now, the only NoSQL databases supported are MongoDB and google app engine.

2) Multilingual Support: Django supports multilingual websites through its built-in internationalization system. So you can develop your website, which would support multiple languages.

3) Framework Support: Django has built-in support for Ajax, RSS, Caching and various other frameworks.

4) Administration GUI: Django provides a nice ready-to-use user interface for administrative activities

2.3.4 BACK END

SQLITE

SQLite is a software library that implements a self-contained, serverless, zero configuration, transactional SQL database engine. SQLite is one of the fastest-growing database engines around, but that's growth in terms of popularity, not anything to do with its size. The source code for SQLite is in the public domain. It is a database, which is zero configured, which means like other databases you do not need to configure it in your system. SQLite engine is not a standalone process like other databases, you can link it statically or dynamically as per your requirement with your application. SQLite accesses its storage files directly.

o SQLite does not require a separate server process or system to operate (serverless). o SQLite comes with zero-configuration, which means no setup or administration needed.

o A complete SQLite database is stored in a single cross-platform disk file.

o SQLite is very small and light weight, less than 400KiB fully configured or less than 250KiB with optional features omitted.

o SQLite is self-contained, which means no external dependencies. o SQLite transactions are fully ACID-compliant, allowing safe access from multiple processes or threads. o SQLite supports most of the query language features found in SQL92 (SQL2) standard.
o SQLite is written in ANSI-C and provides simple and easy-to-use API. o SQLite is available on UNIX (Linux, Mac OS-X, Android, iOS) and Windows (Win32, WinCE, WinRT).

2.4.FEASIBILITY ANALYSIS

Feasibility is conducted to identify the best system that meets all requirements. It is both necessary and important to evaluate the feasibility of a project at the earliest possible time. feasibility study includes an identification description, an evaluation of proposed system and selection of the best system for the job. During the system is to be carried out this is to ensure that the proposed system is not A burden to the shop.

The feasibility study should be relatively cheap and quick. The results should inform the decision of whether to go ahead with a more detailed analysis, some understanding of the major requirements for the system is essential. Four key considerations involved in the feasibility analysis are,

- ➢ OPERATIONAL FEASIBILITY
- ➢ TECHNICAL FEASIBILITY
- ➢ ECONOMICAL FEASIBILITY
- ➢ BEHAVIORAL FEASIBILITY

OPERATIONAL FEASIBILITY

The purpose of the operational feasibility is to determine whether the new system will be used if it is developed and implemented and whether there will be resistance from users that will undermine the possible application benefits. The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the user solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive.

The proposed system is an upgrade version of the current systems new fields have been implemented according to the user need, hence it ensure all the aspects. The proposed system is very much user-friendly and the system is easily understood by simple training and it is operationally feasible to use by any users.

TECHNICAL FEASIBILITY

A study of function performance and constraints may improve the ability to create an acceptable system, technical feasibility is frequently the most difficult area to achieve at the stage of product. Engineering process technical feasibility is deals with the hardware as well as software requirements. The scope was whether the work for the project is done with the current equipments and the existing system technology has to be examined in the feasibility study. The result was found to be true. This feasibility is carried out to check the technical requirements of the system. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system. This is related to the technicality of the project.

This evaluation determines whether the technology needed for proposed system is available or not. it deals with hardware as well as software requirements. That is type of hardware, software and the methods required for running the system are analyzed. So it can be used in any windows os computer. This system requires very low system resources and it will work in almost all configurations. In the existing system all functions are doing manually. So if they get this designed software the problems can be avoided and thus the system will run smoothly.

ECONOMICAL FEASIBILITY

Economical feasibility is the most frequently used method for evaluating the effectiveness of the candidate system. It is very essential because the main goal of the proposed system is to have economically better result along with increased efficiency.

A cost evaluation is weighed against the ultimate income or product. Economic justification is generally the bottom-line consideration that includes cost benefit analysis, long term corporate income strategies and cost of resources needed for development and potential market growth. When compared to the advantage obtained from implementing the system its cost is affordable. Proposed system was developed with available resources since cost input for the software is almost nil the output of the software is always a profit. Hence software is economically feasible.

BEHAVIORAL FEASIBILITY

People are inherently resistant to change and computer is known for facilitating the changes an estimate should be made of how strongly the user, staff reacts towards the development of the computerized system. In the existing system more manpower is required and time factor is more. The more manpower for managing many files for dynamic data replication and more time for search through these files is needed. But in the proposed system, both manpower and time factors are reduced and also unnecessary burden is reduced. Thus the remaining people are made to engage in some other important work. Also there is no need to wait in case of downloading the data for the users therefore , the system is behaviorally feasible.

2.5. DATA FLOW DIAGRAM

A DFD, also known as a “bubble chart” has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. A DFD consists of a series of bubbles joined by lines. The bubbles represents data transformations and the lines represents data flow in the system.

A data flow diagram may be used to represent a system or software at any level of abstraction. A DFD is a diagram that describes the flow of data and the processes that change or transform data throughout a system. It is a structured analysis and design tool that can be used or flowcharting in place of,or in association with, information oriented and process oriented system flowchart. When analyst prepare the DFD, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources.

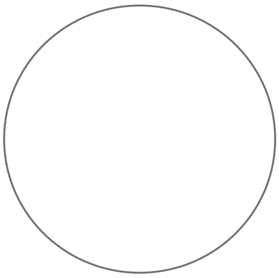
This network is constructed by using a set of symbols that do not imply a physical implementation. The DFD reviews the current physical system, prepare input and output specification, specifies the implementation plan etc.

Basic data flow diagrams symbols are:

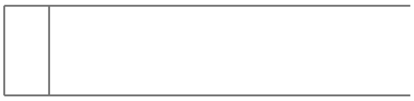
➢ A “rectangle” defines a source or destination of a system data

➢ An “arrow” identifies data flow. It is a pipeline through which information flows.

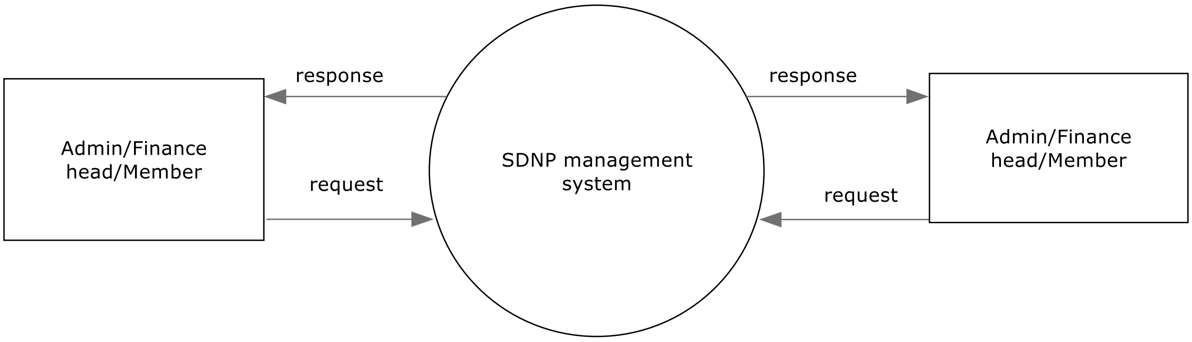
➢ A “circle “ represents” a process that transforms incoming data flow(s) into outgoing data flow(s).



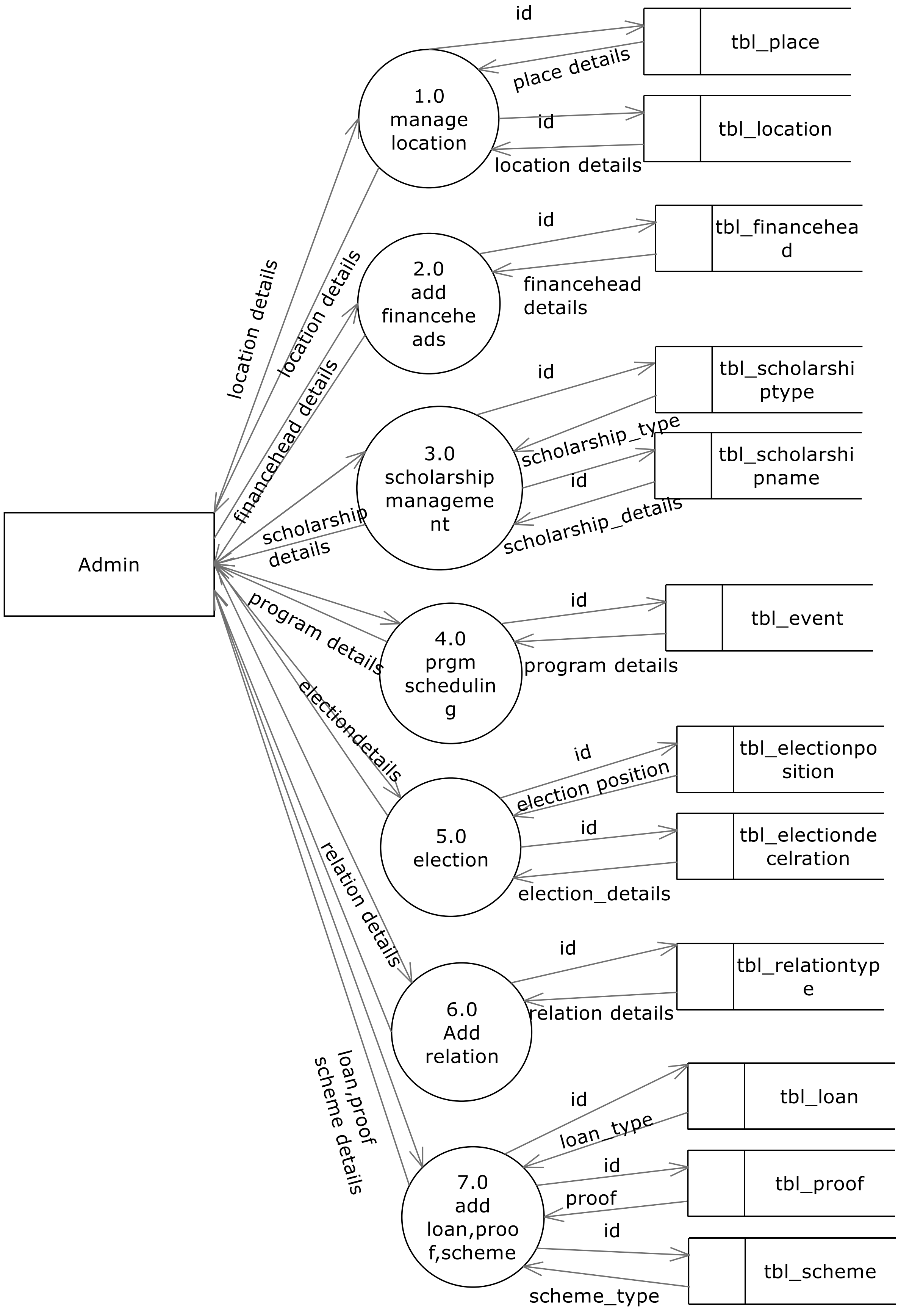
➢ An ”open rectangle” is a data store

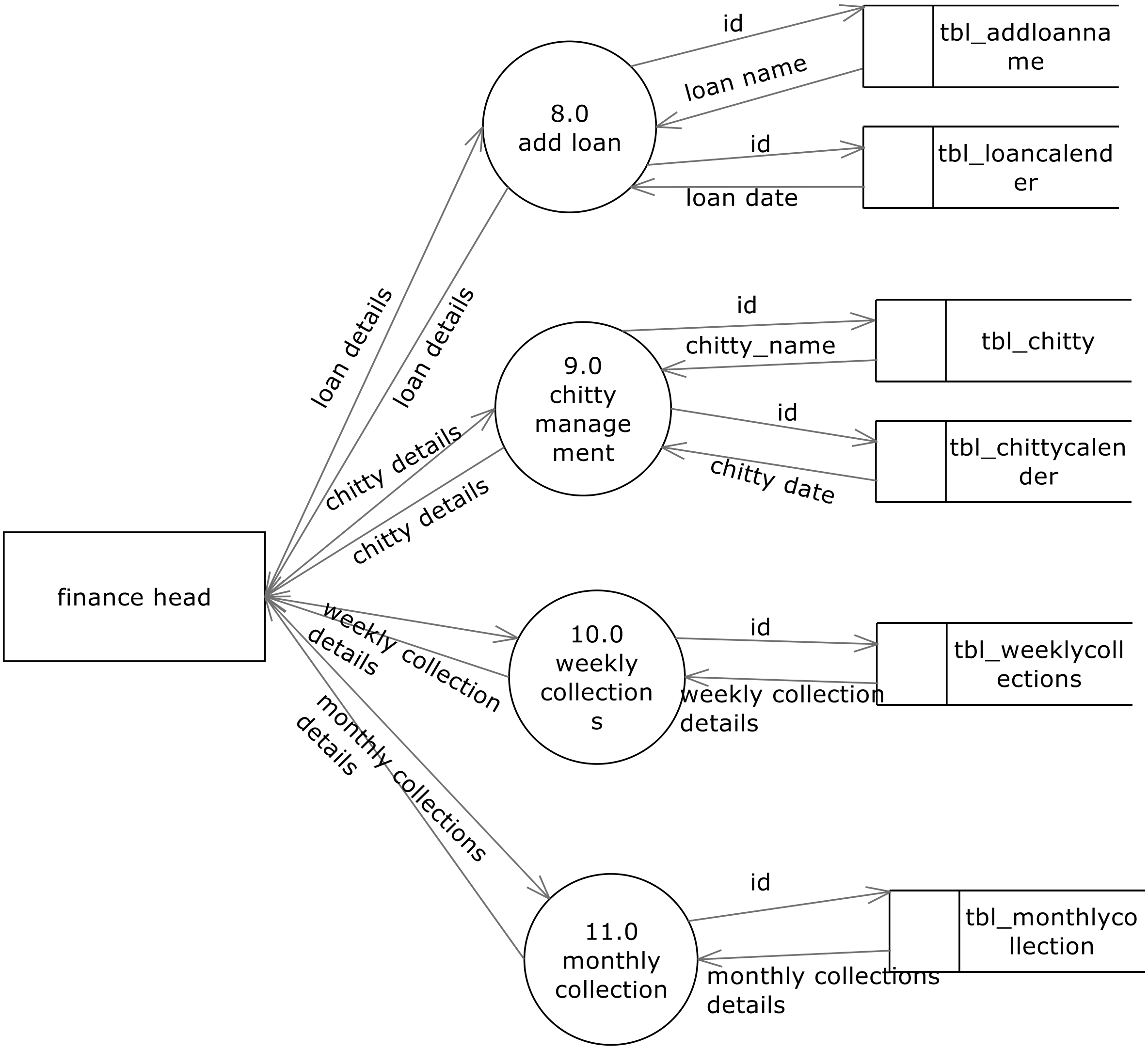


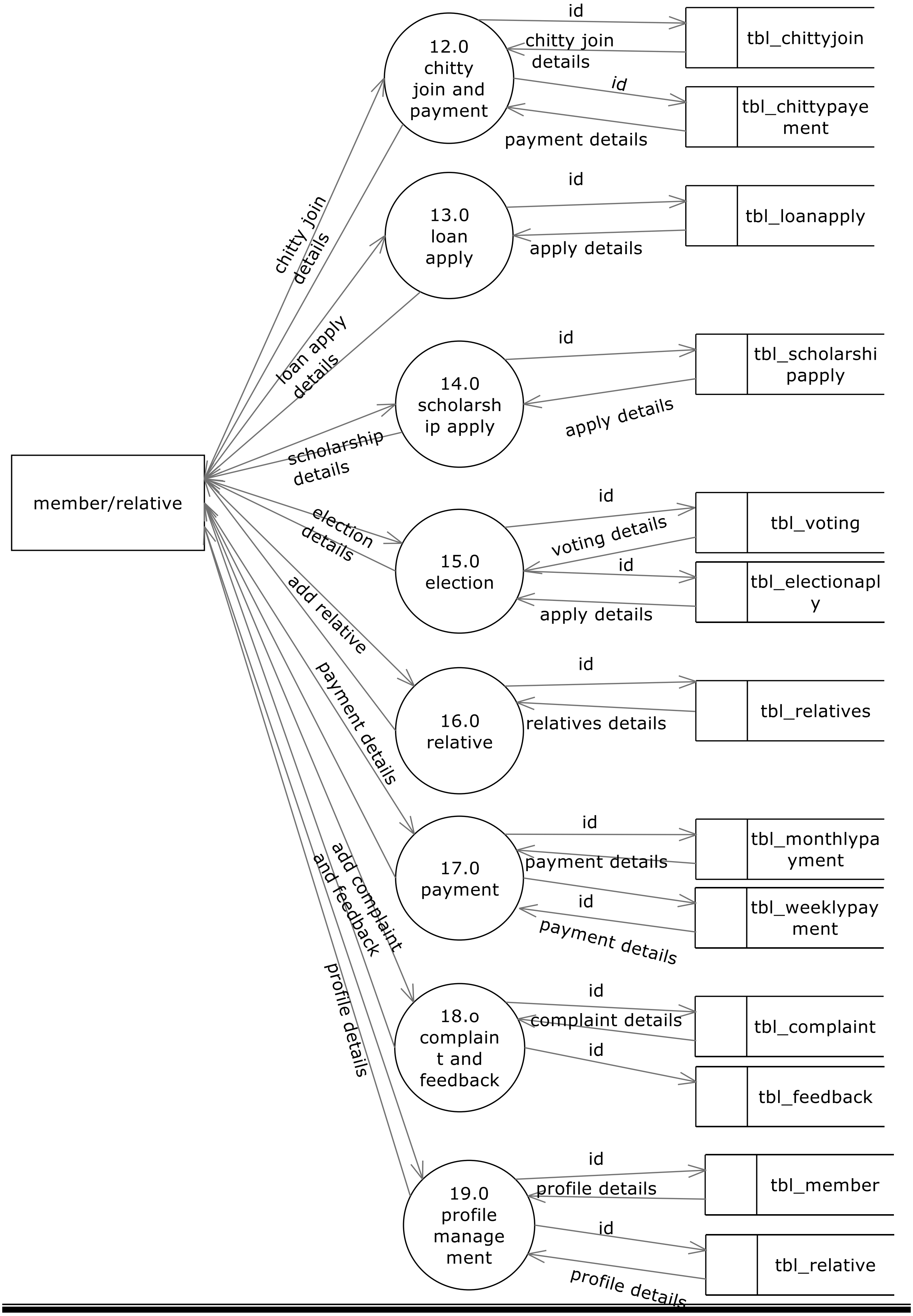
Level 0



Level 1


Admin
1.0 manage location
2.0 add financeheads
3.0 scholarship management
4.0 prgm scheduling
5.0 election
6.0 Add relation
7.0 add loan,proof,scheme
tbl_place
tbl_location
tbl_financehead
tbl_scholarshiptype
tbl_scholarshipname
tbl_event
tbl_electionposition
tbl_electiondecelration
tbl_relationtype
tbl_loan
tbl_proof
tbl_scheme
id
place details
location details
financehead details
scholarship_type
scholarship_details
scholarship details
program details
election position
election_details
electiondetails
relation details
loan_type
proof
scheme_type
loan,proof scheme details


id
tbl_addloanname
loan name
8.0
add loan
id
tbl_loancalender
loan date
id
tbl_chitty
chitty_name
loan details
loan details
9.0
chitty
management
id
tbl_chittycalender
chitty date
chitty details
chitty details
finance head
weekly collection
details
10.0
weekly
collections
id
tbl_weeklycollections
weekly collection
details
monthly collections
details
id
11.0
monthly
collection
tbl_monthlycollection
monthly collections
details


member/relative
12.0 chitty join and payment
id
chitty join details
tbl_chittyjoin
id
payment details
tbl_chittypayement
chitty join details
13.0 loan apply
id
apply details
tbl_loanapply
loan apply details
14.0 scholarship apply
id
apply details
tbl_scholarshipapply
scholarship details
15.0 election
id
voting details
tbl_voting
id
apply details
tbl_electionaply
election details
16.0 relative
id
relatives details
tbl_relatives
add relative
17.0 payment
id
payment details
tbl_monthlypayment
id
payment details
tbl_weeklypayment
payment details
18.o complaint and feedback
id
complaint details
tbl_complaint
id
tbl_feedback
add complaint and feedback
19.0 profile management
id
profile details
tbl_member
id
profile details
tbl_relative
profile details

3.SYSTEM DESIGN

The term design describes the final system and the way in which it is developed. The system design is a solution, how to approach to the new system. This important phase is composed of several steps. An emphasis is on translating the performance requirements of our proposed system into design specification.

Design goes through logical and physical stage of development. In the design phase the physical design producing the working system by defining a particular specification that helps to knowing exactly what the new system must do. The logical design determines the information flow into and of the system and require database. Design is a multistep process that focuses on data structure, software, architecture, procedural details, and interface between modules. The design process translates the requirements into the representation of the software. Computer software design changes continually because new methods, better analysis and broader understanding evolved.

It provides the understanding and procedure details necessary for implementing the proposed system .an emphasis is on translating the performance requirement of our proposed system into design specification. Design goes through logical and physical stage. The system design is the last phase that indicate the final system and process of design phase. In the designed phase of maintenance management system the database tables, input screens and output reports are designed. In table designing, redundancy is avoided.

Design is the only way that we can accurately translate a system requirement into a software product. In our production management system, the all input screens are designed as user friendly and understandable.

3.1. INPUT DESIGN

Input design is the link that ties the information system into the world of its users. The input design involves determining what the input is, how the data should be performed, how to validate data, how to minimize data entry and how to provide a multi user facility, inaccurate input data are the most common cause of errors in data processing. Errors entered by data entry operator can be controlled by input design. Input design is the process of converting user originated input to a computer-based format.

Input data are collected and organized into groups of similar data. Once identified, appropriate input media are selected for processing all the input data re validated in the order and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions then it is transferred to the appropriate table in the database.

A form is designed to enter the details should be user friendlier so that authorized user with even less knowledge can enter the data. The form is designed using v b tools like command boxes, text boxes, labels, option buttons, combo boxes etc.

3.2. OUTPUT DESIGN

Output design is very important concept in the computerized system, without reliable output the user may feel the entire system unnecessary and avoids using it. The proper output design is important in any system and facilitates effective decision making. The output design of this system includes various reports. Output requirements are designed during system analysis. An application is successful only when it can provide efficient and effective reports.

The goal of the output design is to capture the output and get the data into a format suitable for the computer. It is very helpful to produce the clear, accurate and speedy information for end users. A major form of the output is the harder copy from the pointer and screen reports. Printouts are designed around the output requirements of the user. Allowing the user to view the sample screen is important because the user is the ultimate judge of the quality of output. Output of this project is provided in the form of reports created using crystal report tool.

3.3. DATABASE DESIGN

Database is a collection of interrelated data stores with minimum the overall objective in the development of the database technology has been to treat data as an organizational resource and has an integrated whole. Database management system allows data to be protected and organized separately from other resources. Database is an integrated collection of data. This is the difference between logical and physical data. The general objective is to make information access easy, quick, inexpensive and flexible for users.

The database approach to system design places greater emphasis on the integration, integrity and independence of data. This involves the separation of logical and Physical storage and vice versa. Databases are normally implemented by using a package called DBMS.

3.4. TABLE DESIGN

Database design is the logical form of design of data storage in the form of records in a particular structure in the form of tables with fields which is not transparent to the normal user but it actually acts as the backbone of the system.

As we know database is a collection of data, which helps the system to manage and store data called database management system. Data base management system builds some form of constraints like integrity constraints, i.e, the primary key/ unique key and referential integrity which help to keep data structure storage and access of data from tables efficiently and accurately and take necessary steps to concurrent access of data and avoid redundancy of data in tables by normalization criterions.

Normalization is the method of breaking down complex table structures into simple table structures by using certain rules thus reduce redundancy and inconsistency and disk space usage and thus increase the performance of the system or application which is directly linked to the database design and also solve the problems of anomalies.

The Data Base design of new system's is in second normal form and the second normal form defines that all non-key fields of the table are fully dependent on the whole key.

Each field of the tables is made to depend only on key attributes and dependency on non-key fields is eliminated.

1- Table Name : tbl_adminlogin
Primary key : id
Description : Used to store administrator details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Admin id
email	varchar	50	Admin user name
password	varchar	50	Admin password

2- Table Name : tbl_place
Primary key : id
Description : Used to store place details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Place id
place_name	varchar	50	Place name

3- Table Name : tbl_location
Primary key : id
Foreign Key : place_id
Description : Used to store location details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Location id
Place_id	int		Place id
location_name	varchar	50	Location name

4- Table Name : tbl_financehead
Primary key : id
Description : Used to store finance head details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Finance head id
financehead_name	varchar	50	Finance head name
user_name	varchar	50	User name of finance head
password	varchar	50	Password of finance head
fdate	varchar	50	Starting date of finance head
sdate	varchar	50	Ending date of finance head

5- Table Name : tbl_loan
Primary key : id
Description : Used to store loan details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Loan id
loan_name	varchar	50	Loan name

6- Table Name : tbl_proof
Primary key : id
Description : Used to store proof type

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Proof id

proof_name	varchar	50	Proof type name

7- Table Name : tbl_scheme

Primary key : id

Description : Used to store scheme

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Scheme id
scheme_name	varchar	50	Scheme name

8- Table Name : tbl_memberadding

Primary key : id

Foreign Key : location_id

Description : Used to store member details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Member id
member_name	varchar	50	Member name
age	int		Age of member
contact	varchar	50	Contact of member
email	varchar	50	Email of member
address	varchar	100	Address of member
gender	varchar	50	Gender of member
password	varchar	50	Password of member account
location_id	int		Location id

photo	varchar	50	Photo of member

9- Table Name : tbl_relationtype
Primary key : id
Description : Used to store relation type

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Relation type id
relation_type	varchar	50	Relation Type

10- Table Name : tbl_scholarshiptype
Primary key : id
Description : Used to store scholarship type

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of scholarship type
scholarship_type	varchar	50	Name scholarship type

11- Table Name : tbl_addloan
Primary key : id
Foreign Key : head_id, loan_type_id
Description : Used to store loan name

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of loan name
loan_name	varchar	50	Name of loan
loan_details	varchar	50	Loan details

head_id	int		Id of finance head
loan_type_id	int		Id of lon type

12- Table Name : tbl_scholarshipname
Primary key : id
Foreign Key : proof_name_id, scholarship_type_id
Description : Used to store scholarship name

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of scholarship name
scholarship_name	varchar	50	Name of scholarship
scholarship_details	varchar	100	Scholarship details
proof_name_id	int		Id of proof type
scholarship_type_id	int		Id of scholarship type

13- Table Name : tbl_chitty
Primary key : id
Foreign Key : head_id, scheme_id
Description : Used to store chitty name and details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of chitty name
chitty_name	varchar	50	Name of chitty
chitty_details	varchar	100	Chitty details

head_id	int		Id of finance head
scheme_id	int		Id of scheme type

14- Table Name : tbl_chittycalendar
Primary key : id
Foreign Key : head_id, chitty_name_id
Description : Used to store chitty calendar

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of chitty calendar
amount	varchar	50	Amount of chitty
startdate	varchar	50	Start date of chitty
enddate	varchar	50	End date of chitty
no_installment	int		Number of installment
chitty_name_id	int		Id of chitty name
head_id	int		Id of finance head

15- Table Name : tbl_loancalendar
Primary key : id
Foreign Key : head_id, loan_name_id
Description : Used to store loan calendar

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of loan calendar
amount	int		Amount of loan

startdate	varchar	50	Start date of loan
enddate	varchar	50	End date of loan
no_installment	int		Number of installment
loan_name_id	int		Id of loan name
head_id	int		Id of finance head

16- Table Name : tbl_monthlycollection
Primary key : id
Foreign Key : head_id
Description : Used to store monthly collection details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of monthly collection
amount	int		Amount of monthly collection
head_id	int		Id of finance head

17- Table Name : tbl_weeklycollection
Primary key : id
Foreign Key : head_id
Description : Used to store weekly collection details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of weekly collection
amount	int		Amount of weekly collection
head_id	int		Id of finance head

18- Table Name : tbl_chittyjoin
Primary key : id
Foreign Key : memberdata_id, relative_id, proof_name_id, chittydata_id Description : Used to store chitty apply

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of loan calendar
status	int		Status of chitty application
document	varchar	100	Identity document
chittydata_id	int		Id of chitty name
memberdata_id	int		Id of member
proof_name_id	integer		Id of proof
relative_id	integer		Id of relative
apply_date	varchar	50	Date of apply

19- Table Name : tbl_loanapply
Primary key : id
Foreign Key : head_id, loan_name_id, member_name_id, proof_name_id Description : Used to store loan apply

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of loan calendar
status	int		Status of loan application
document	varchar	100	Identity document

loan_name_id	int		Id of loan name
member_name_id	int		Id of member
proof_name_id	int		Id of proof
apply_date	varchar	50	Date of apply

20- Table Name : tbl_monthlycollectionpayment
Primary key : id
Foreign Key : monthlycollection_id, member_name_id
Description : Used to store monthly collection payment details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of monthly collection
mothlycollection_id_id	int		Id of monthly collection
member_name_id	int		Id of member
Payment_date	varchar	50	Payment date of monthly collection

21- Table Name : tbl_paymentchitty
Primary key : id
Foreign Key : chitty_apply_id, member_name_id, relative_id
Description : Used to store chitty payment details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Payment id of chitty
chitty_apply_id	int		Id of chitty application
member_name_id	int		Id of member
relative_id	int		Id of relative

repayment_date	varchar	50	Repayment date of chitty

22- Table Name : tbl_relative
Primary key : id
Foreign Key : member_name_id, relation_type_id, refer_id
Description : Used to store relative details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Relative id
relative_name	varchar	50	Relative name
age	int		Age of relative
contact	varchar	50	Contact of relative
email	varchar	50	Email of relative
gender	varchar	50	Gender of relative
password	varchar	50	Password of relative account
photo	varchar	50	Photo of member
member_name_id	int		Id of member
relation_type_id	int		Id of relation type
refer_id	int		Id of refence

23- Table Name : tbl_repaymentloan
Primary key : id
Foreign Key : chitty_apply_id, member_name_id
Description : Used to store loan repayment details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION

id	int		Payment id of loan
loanapply_id	varchar	50	Id of loan application
member_name_id	varchar	50	Id of member
repayment_date	varchar	50	Repayment date of loan

24- Table Name : tbl_scholarshipapply
Primary key : id
Foreign Key : scholarship_name_id, member_name_id, relative_name_id Description : Used to store scholarship apply

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of scholarship application
status	int		Status of scholarship application
document	varchar	100	Identity document
scholarship_name_id	int		Id of scholarship name
member_name_id	int		Id of member
relative_name_id	int		Id of relative
apply_date	varchar	50	Date of apply

25- Table Name : tbl_weeklycollectionpayment
Primary key : id

Foreign Key : weeklycollection_id, member_name_id, relative_name_id Description : Used to store weekly collection payment details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of weeklycollection
weeklycollection_id_id	int		Id of weeklycollection
member_name_id	int		Id of member
relative_name_id	int		Id of relative
payment_date	varchar	50	Payment date of monthly collection

26- Table Name : tbl_electionposition
Primary key : id
Description : Used to store election position details

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of election position
election_position	varchar	50	Election position

27- Table Name : tbl_electiondeclaration
Primary key : id
Description : Used to store scholarship apply

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of election declaration
title	varchar	50	Title of election
details	varchar	50	Details of election

nomination_date	varchar	50	Date of nomination
verified_date	varchar	50	Date of verification
election_date	varchar	50	Date of election
result_date	varchar	50	Date of result
posting_date	varchar	50	Date of posting

28- Table Name : tbl_electionapply
Primary key : id
Foreign Key : election_name_id, election_position_id, member_name_id Description : Used to store election apply

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of election declaration
status	int		Status of election application
election_name_id	int		Id of election title
election_postion_id	int		Id of election position
member_name	int		Id of member

29- Table Name : tbl_voting
Primary key : id
Foreign Key : election_apply_id, relative_name_id, member_name_id Description : Used to store voting details.

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of voting

election_apply_id	int		Id of election apply
relative_name_id	int		Id of relative
member_name_id	int		Id of member

30- Table Name : tbl_events
Primary key : id
Description : Used to store event details.

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of event
name	varchar	50	Name of event
details	varchar	50	Details of event
event_date	varchar	50	Event date

31- Table Name : tbl_feedback
Primary key : id
Foreign Key : relative_name_id, member_name_id
Description : Used to store feedback.

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of event
content	varchar	50	Name of event
member_name_id	int		Id of member
relative_name_id	int		Id of relative

32- Table Name : tbl_complaint
Primary key : id
Foreign Key : relative_name_id, member_name_id

Description : Used to store complaint.

FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
id	int		Id of event
title	varchar	50	Title of the complaint
status	int		Status of the complaint
content	varchar	50	Content of the complaint
member_name_id	int		Id of member

4.SYSTEM TESTING & IMPLEMENTATION

4.1 SYSTEM TESTING

Testing is the process of examining the software to compare the actual behaviour with that of the excepted behavior. The major goal of software testing is to demonstrate that faults are not present. In order to achieve this goal, the tester executes the program with the intent of finding errors. Though testing cannot show absence of errors but by not showing their presence it is considered that these arenot present.

System testing is defined as the process by which one detects the defects in the software. Any softwaredevelopment organization or team has to perform several processes. Software testing is one among them. It is the final opportunity of any programmer to detect and rectify any defects that may have appeared during the software development stage. Testing is a process of testing a program with the explicit intention of finding errors that makes the program fail. In short system testing and quality assurance is a review in software products and related documentation for completion, correctness, reliability and maintainability.

System testing is the first stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct and thegoal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptance testing.

The testing steps are:

- Unit Testing
- Integration Testing
- Validation testing
- Output Testing
- Acceptance Testing

System Testing provides the file assurance that software once validated mast combined with all other system elements. System testing verifies whether all elements nave been combined properly and that overall system function and performance is achieved. FA the integration of modules, the validation test was carried out over the system. lt was that all the modules work well together and meet the overall system function and performance.

Unit Testing

Unit testing is carried out screen-wise, each screen being identified as an object. Attention is diverted to individual modules, independently to one another to locate errors. This has enabled the detection oferrors in coding and logic.

Various test cases are prepared. For each module these test cases are implemented and it is checked whether the module is executed as per the requirements and outputs the desired result. In this test eachservice input and output parameters are checked.

In unit testing:

- Module interface was tested to ensure that information properly flows into and out of theprogram under test.
- Boundary condition was tested to ensure that module operates properly at boundariesestablished to limit or restrict processing.
- All independent paths through the control structures were executed to ensure that all statementsin the modules have been executed at least once.
- Error handling paths were also tested.

Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing.Unit tested module were taken and a single program structure was built that has been dictated by the design. Incremental integration has been adopted here.The modules are tested separately lor accuracy and modules are integrated too.th tn. using bottom up integration i.e., by integrating from moving from bottom to the toon the system is checked and errors found during integration are rectified.

Validation Testing

Validation testing is done to ensure complete assembly of the error-free software. Validation can be termed successful only if it functions in manner. Reasonably expected by the student under validationis alpha and beta testing. The student-side validation is done in this testing phase. It is checked whetherthe data passed to each student is valid or not. Entering incorrect values does the validation testing andit is checked whether the errors are being considered. Incorrect values are to be discarded. The errors are rectified.

In “University result portal" verifications are done correctly. So, there is no chance for users to enter incorrect values. It will give error messages by using different validations. The validation testing is done very clearly and found it is error free.

Output Testing

After performing the validation testing the next step is output testing of the proposed system, since nosystem could be useful if it does not produce the required output in a specific format.

The output format on the screen was found to be correct as the format was designed in the system design phase according to the user needs. For the hard copy also, the output comes out as specified requirement by the user. Hence output testing does not result in any Correction in the system. output This project is developed based on the user choice. It is user friendly. The output format is very clear to user. Output testing is done on Smart builders correctly.

Acceptance testing

Acceptance involves running a suite of tests on the completed system. Each individual test, known as a Case, exercise particular operating condition of the operating condition of the user's environment orfeature of the system, and will result in a pass fail.

4.2 SYSTEM IMPLEMENTATION

The implementation is the final state and it is an important phase. It involves the invalid programming system testing. user training and the operational running of developed proposed system that constitutesthe application subsystems. A major task of preparing for implementation is education of users. whichshould really have been taken place much carrier in the project when they were belong involved in theinvestigation and design work. During the implementation phase system actually take physical shape.In order to develop a system implemented planning is very essential.

The implementation phase of the software development is concerned with translating design specification into source code. The user tests the developed system and changes are made according totheir needs. Our system has been successfully implemented.

Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period of time.

The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from old system to new system. The system canbe implemented only after testing is done and is found to be working to specifications. The implementation stage is a systems project in its own right.

The implementation stage involves following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of method to achieve change over
- Evaluation of the changeover method.

In the case of this project all the screens are designed first. For making it to be executable, codes are written on each screen and performs the implementation by creating the database and connecting to theserver. After that the system, is Checked, whether it performs all the transactions Correctly. Then databases are cleared and made it to be usable to the technicians.

5.SECURITY TECHNOLOGIES AND POLICTES

The protection of computer-based resources that includes hardware, software, data procedures and people against unauthorized use or natural. Disaster is known as System Security. System Security can be divided into four related issues:

- Security
- Integrity
- Privacy
- Confidentiality

SYSTEM SECURITY refers to the technical innovations and procedures applied to the hardware and operation systems to protect against deliberate or accidental damage from a defined threat.

DATA SECURITY is the protection of data from loss, disclosure, modification and destruction.

SYSTEM INTEGRITY refers to the power functioning of hardware and programs, appropriate physical security and safety against external threats such as caves dropping and wiretapping

PRIVACY defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.

CONFIDENTIALITY is a special status given to sensitive information in a database to minimize the possible invasion of privacy. It is an attribute of information that characterizes it needs for protection.

SECURITY IN SOFTWARE System security refers to various validations on data in form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered and only valid operations are performed on the systems.

The system employees two types check and controls:

CLIENT-SIDE VALIDATION Various client-side validations are used to ensure on the client side that only valid data is entered.Client-side validation savesserver time and load to handle invalid data.Some checks imposed are:

- ❖ Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the client side to save the server time and load.
- ❖ Tab-indexes are set according to the need and taking into account the ease of user while working with the system.

SERVER-SIDE VALIDATION Some checks cannot be applied at client side. Server-side checks are necessary to save the system from failing and intimating the user that some invalid operation has been performed or the performed operation is restricted. Some of the server-side checks imposed is:

- ❖ Server-side constraint has been imposed to check for the validity of primary key and foreign key. A primary key value cannot be duplicated. Any attempt to duplicate the primary value results into a message intimating the user about those values through the forms using foreign key can be updated only of the existing foreign key values.
- ❖ User is intimating through appropriate messages about the successful operations or exceptions occurring at server side.
- ❖ Various Access Control Mechanisms have been built so that one user may not agitate upon another. Access permissions to various types of users are controlled according to the organizational structure. Only permitted users can log on to the system and can have access according to their category. User name, passwords and permissions are controlled over the server side.
- ❖ Using server-side validation, constraints on several restricted operations are imposed

6 .MAINTENANCE

Software maintenance is the modification of a software product aner delivery to correct faults, to improve performance or other attributes. Maintenance is the ease with which a program can be corrected if any error is encountered, adapted if its environment changes or enhanced if the customer desires a change in requirement. Maintenance follows conversation to extend that changes arenecessary to maintain satisfactory operations relative to changes in the user's environment.

Maintenance often includes minor enhancements or corrections to problems that surface in the system'soperation. Maintenance is also done based on fixing the problems reported, changing the interface withother software or hardware enhancing the software.

CATEGORIES OF MAINTENANCE

Corrective Maintenance

Corrective maintenance is the most commonly used maintenance approach, but it is easy to see its limitations. When equipment fails, it often leads to downtime in production, and sometimes damages other parts. In most cases, this is expensive. Also, if the equipment needs to be replaced, the cost of replacing it alone can be substantial. Reliability of systems maintained by this type of maintenance is unknown and cannot be measured. Corrective maintenance is possible since the consequences of failureor wearing out are not significant and the cost of this maintenance is not great

Adaptive Maintenance

Modification of a software product performed after delivery to keep a are product usable m a changed or changing environment. Adaptive maintenance includes any work initiated as a consequence of moving the software to a different hardware or software platform. It is a change driven by the need to accommodate modifications in the enviroment of software system. The environment in this context refers to the totality of all conditions and influences which act from outside upon the system.

Perfective Maintenance

Modification of a software product alter delivery to improve performance or maintainability. This term is used to describe changes undertaken to expand the existing requirements of the system. A successful piece or software lends to be subjected to a the Succession of changes resulting in an increase in us requirements. This is based an premise that as the software becomes useful, the user experiment with new cases beyond the of Scope for which it was initially developed. Vxpansi01 n requirements can takethe form enhancement of existing system functionality and improvement in computational efficiency.

Preventive Maintenance

Preventive maintenance is a schedule of planned maintenance actions aimed at the prevention of breakdowns and failures. The primary goal of preventive maintenance is to prevent the failure of equipment before it actually occurs. It is designed to preserve and enhance equipment reliability by replacing worn components before they actually fail. Preventive maintenance activities include equipment checks, partial or complete overhauls at specified periods.

Long-term benefits of preventive maintenance include:

- Improved system reliability.
- Decreased cost of replacement.
- Decreased system downtime.

7.SCOPE FOR FUTURE ENHANCEMENT

This system will reduce the manual operation. Maintain all the records. And also generates the various reports of loan, scholarship and paraspara sahaya nidhi. Apply for loan, scholarship and paraspara sahaya nidhi through online. Election is done through the website. This project can be further enhanced to conduct the event activities such as registration of events and financial details through online. This project is very useful for members ,relatives and authorities of SNDP Karayogam. They can use this application anytime, anywhere at a low cost.

The system has been designed in such a way that it can be modified with very little effort when such needs arise in the future. New features can be added with slight modifications of software which make it easy to expand the scope of this project. Though the system is working on various assumptions, it can be modified easily to any kind of requirements.

- Conduct event registration through online and its finance details.

8.CONCLUSION

It has been a great pleasure for me to work on this challenging project. The project proved good for me as it provide knowledge of not only programming but also about all handling procedure related with SNDP KARAYOGAM MANAGEMENT SYSTEM. It also provide knowledge about latest technology used in developing web enabled application and client server technology that will be great demand in future. The performance of the system is provided to be efficient. All the customers receive on the overall benefits through the system. The system provides flexibility for incorporating new features, which may be necessary in future keeping in mind need of the end user, attempts has been made to cover as much as possible within stipulated time limits. SNDP Karayogam Management System aims to develop a particular Karayogam for store their activity as online. The system is only for a particular karayogam. The karayogam is based on Taluk. In each karayogam there is a president, secretary, treasurer and members. In each karayogam has its own programs and financial details. Now the karayogam authorities store their financial details, members details, program scheduling details and other details as records. The aim of the project is reduced the time and work. That is the activities of SNDP karayogam is convert into online system. The main advantage of online system is store the financial details, program scheduling, members details as online and it is safe. Also it reduce the time and handworks.

9.BIBLIOGRAPHY

1. **Software Engineering A Practioner's Approach,Roger S Pressman,** McGraw hill International,Edition,Sixth Edition.

2. **Software Engineering,K K Agarwal and Yogesh Singh**, New age international,Third Edition.

3. **Object Oriented modelling andDesign with UML,Micheal Blaha,James Rumbaugh,Person**, Second Edition.

4. **Craig Larman,"Applying UML and patterns:An international to object",Oriented Analysis andDesign and Unified Process**, Third Edition,Person Education,2007.

5**. Raghu Ramakrishnan and Johannes Gehrke,Database Management System**, Third Edition,McGraw Hill Companies, 2003.

6. **Programming and problem Solving with Python,Ashok Namdev Kamthane &Amit Ashok.**

WEBSITES

- http://chat.openai.com/
- http://tutorialspoint.com/
- http://github.com/
- http://www.w3schools.com/