Project Report

On

INSURANCE MANAGEMENT SYSTEM

ACKNOWLEDGEMENT

We take this occasion to thank God, almighty for blessing us with his grace and taking our endeavour to a successful culmination. We extend our sincere and heart felt thanks to our esteemed guide, Mr. XYZ for providing us with the right guidance and advice at the crucial junctures and for showing us the right way. We extend our sincere thanks to our respected head of the division Mr.XYZ, for allowing us to use the facilities available. We would like to thank the other faculty members also, at this occasion. Last but not the least, we would like to thank friends for the support and encouragement they have given us during the course of our work.

ABSTRACT

PROJECT TITLE : Insurance Management System

APPLICATION TYPE: Web Application

DESCRIPTION

The main purpose of this project is to estimate some important statistics of Micro insurance in the current market of India by gathering and analyzing data which is available. The statistics will be useful to prepare better insurance plans to boost up the Life Insurance industry in India.

MODULES

a) Admin module: This module allows only registered members to log in and manage the users and article.
b) users module: This module allows to known the premium details of the users and product details.

PROGRAMMING ENVIRONMENT

Operating System : Windows

Application type : Web Application

Frontend : PHP

Database : MySql

Server : Apache

INTRODUCTION

Purpose of the System

This web site is an online micro insurance Analysis and information management system that provides easy access of information regarding the people and resources of insurance. This site is not a static site but with wonderful dynamic facilities like search tools for insurance awareness articles, guidelines, illustrations through images for visitors. This site also provides several dynamic features.

Scope of the Developed System

The developed system is applicable to only MIRC to provide online services to its member organizations.

Insurance Management System

In this project we use PHP and Mysql database. It has two module

1.User Module

2.Admin Module.

User Module:

In this module there is Two Section

1.Insurance : In this section user can apply for policy and check his policy is approve or reject(which is done by admin).

2.Ticket: In this section user can raise ticket against any complain and see status of his/her ticket

User can also update his/ her profile, change password and recover password.

Admin Module

In this module there is seven sections

1.Dashboard: In this section admin can see all detail in brief.

2.Insurance Category: In this section admin can manage categories(add and update).

3.Insurance Subcategory: In this section admin can manage sub category(add and update).

4.Insurance Policy: In this section admin can manage insurance policy(add and update policy).

5.User Detail: In this section admin can manage all user detail.

6. Policy Holders: In this section admin can manage all insured policy. Admin can view policy on the basis of status(pending policy, approved policy and disapproved policy). Admin also can approve pending policy.

7. Tickets: In this section admin can view detail of issues raised by user and can also update remark on particular tickets.

Admin can also update his profile and change his password

SYSTEM ENVIRONMENT

Hardware Configuration

1. Pentium IV Processor
2. 512 MB RAM
3. 40GB HDD
4. 1024 * 768 Resolution Color Monitor

Note: This is not the "System Requirements".

Software Configuration

1. OS : Windows XP
2. PHP Triad (PHP5.6, MySQL, Apache, and PHPMyAdmin)

Software Features

PHP TRIAD

PHPTriad installs a complete working PHP/MySQL server environment on Windows platforms (9x/ NT). Installs PHP, MySQL, Apache, and PHPMyAdmin.

PHP

PHP is a scripting language originally designed for producing dynamic web pages. It has evolved to include a command line interface capability and can be used in standalone graphical applications. While PHP was originally created by Rasmus Lerdorf in 1995, the main implementation of PHP is now produced by **The PHP**

Group and serves as the *de facto* standard for PHP as there is no formal specification. PHP is free software released under the PHP License, however it is incompatible with the GNU General Public License (GPL), due to restrictions on the usage of the term *PHP*. It is a widely-used general-purpose scripting language that is especially suited for web development and can be embedded into HTML. It generally runs on a web server, taking PHP code as its input and creating web pages as output. It can be deployed on most web servers and on almost every operating system and platform free of charge. PHP is installed on more than 20 million websites and 1 million web servers.

PHP originally stood for Personal Home Page. It began in 1994 as a set of Common Gateway Interface binaries written in the C programming language by the Danish/Greenlandic programmer Rasmus Lerdorf. Lerdorf initially created these Personal Home Page Tools to replace a small set of Perl scripts he had been using to maintain his personal homepage. The tools were used to perform tasks such as displaying his résumé and recording how much traffic his page was receiving. He combined these binaries with his Form Interpreter to create PHP/FI, which had more functionality. PHP/FI included a larger implementation for the C programming language and could communicate with databases, enabling the building of simple, dynamic web applications.

Lerdorf released PHP publicly on June 8, 1995 to accelerate bug location and improve the code. This release was named PHP version 2 and already had the

basic functionality that PHP has today. This included Perl-like variables, form handling, and the ability to embed HTML. The syntax was similar to Perl but was more limited, simpler, and less consistent. Zeev Suraski and Andi Gutmans, two Israeli developers at the Technion IIT, rewrote the parser in 1997 and formed the base of PHP 3, changing the language's name to the recursive initialism *PHP: Hypertext Preprocessor*. The development team officially released PHP/FI 2 in November 1997 after months of beta testing. Afterwards, public testing of PHP 3 began, and the official launch came in June 1998. Suraski and Gutmans then started a new rewrite of PHP's core, producing the Zend Engine in 1999. They also founded Zend Technologies in Ramat Gan, Israel.

On May 22, 2000, PHP 4, powered by the Zend Engine 1.0, was released. On July 13, 2004, PHP 5 was released, powered by the new Zend Engine II. PHP 5 included new features such as improved support for object-oriented programming, the PHP Data Objects extension (which defines a lightweight and consistent interface for accessing databases), and numerous performance enhancements. The most recent update released by The PHP Group is for the older PHP version 4 code branch.

In 2008, PHP 5 became the only stable version under development. Late static binding has been missing from PHP and will be added in version 5.3. PHP 6 is under development alongside PHP 5. Major changes include the removal of register_globals, magic quotes, and safe mode. The reason for the removals was because register_globals had given way to security holes, and magic quotes had

an unpredictable nature, and was best avoided. Instead, to escape characters, Magic quotes may be substituted with the addslashes() function, or more appropriately an escape mechanism specific to the database vendor itself like mysql_real_escape_string() for MySQL.

PHP does not have complete native support for Unicode or multibyte strings; Unicode support will be included in PHP 6. Many high profile open source projects ceased to support PHP 4 in new code as of February 5, 2008, due to the GoPHP5 initiative, provided by a consortium of PHP developers promoting the transition from PHP 4 to PHP 5. It runs in both 32-bit and 64-bit environments, but on Windows the only official distribution is 32-bit, requiring Windows 32-bit compatibility mode to be enabled while using IIS in a 64-bit Windows environment. There is a third-party distribution available for 64-bit Windows.

Usage

PHP is a general-purpose scripting language that is especially suited for web development. PHP generally runs on a web server, taking PHP code as its input and creating web pages as output. It can also be used for command-line scripting and client-side GUI applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems. It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.

PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. It can automatically detect the language of the user. From PHP 4, the PHP parser compiles input to produce bytecode for processing by the Zend Engine, giving improved performance over its interpreter predecessor. Originally designed to create dynamic web pages, PHP's principal focus is server-side scripting,and it is similar to other server-side scripting languages that provide dynamic content from a web server to a client, such as Microsoft's Active Server Pages, Sun Microsystems' JavaServer Pages, and mod_perl. PHP has also attracted the development of many frameworks that provide building blocks and a design structure to promote rapid application develo-pment (RAD). Some of these include CakePHP, Symfony, CodeIgniter, and Zend Framework, offering features similar to other web application frameworks.

The LAMP architecture has become popular in the web industry as a way of deploying web applic-ations. PHP is commonly used as the *P* in this bundle alongside Linux, Apache and MySQL, although the *P* may also refer to Python or Perl.

As of April 2007, over 20 million Internet domains were hosted on servers with PHP installed, and PHP was recorded as the most popular Apache module. Significant websites are written in PHP including the user-facing portion of Facebook, Wikipedia (MediaWiki), Yahoo!, MyYearbook, , Digg, Wordpress and

Tagged. In addition to server-side scripting, PHP can be used to create stand-alone, compiled applications and libraries, it can be used for shell scripting, and the PHP binaries can be called from the command line.

Speed optimization

As with many scripting languages, PHP scripts are normally kept as human-readable source code, even on production web servers. In this case, PHP scripts will be compiled at runtime by the PHP engine, which increases their execution time. PHP scripts are able to be compiled before runtime using PHP compilers as with other programming languages such as C (the language PHP and its extensions are written in). Code optimizers aim to reduce the computational complexity of the compiled code by reducing its size and making other changes that can reduce the execution time with the overall goal of improving performance. The nature of the PHP compiler is such that there are often opportunities for code optimization, and an example of a code optimizer is the Zend Optimizer PHP extension.

Another approach for reducing overhead for high load PHP servers is using PHP accelerators. These can offer significant performance gains by caching the compiled form of a PHP script in shared memory to avoid the overhead of parsing and compiling the code every time the script runs.

Security

The National Vulnerability Database stores all vulnerabities found in computer software. The overall proportion of PHP-related vulnerabilities on the database amounted to: 12% in 2003, 20% in 2004, 28% in 2005, 43% in 2006, 36% in 2007, and 35% in 2008. Most of these PHP-related vulnerabilities can be exploited remotely: they allow hackers to steal or destroy data from data sources linked to the webserver (such as an SQL database), send spam or contribute to DOS attacks using malware, which itself can be installed on the vulnerable servers.

These vulnerabilities are caused mostly by not following best practice programming rules: technical security flaws of the language itself or of its core libraries are not frequent. Recognizing that programmers cannot be trusted, some languages include taint checking to detect automatically the lack of input validation which induces many issues. However, such a feature is being developed for PHP

Hosting PHP applications on a server requires a careful and constant attention to deal with these security risks. There are advanced protection patches such as Suhosin and Hardening-Patch, especially designed for web hosting environments. Installing PHP as a CGI binary rather than as an Apache module is the preferred method for added security. With respect to securing the code itself, PHP code can be obfuscated to make it difficult to read while remaining functional.

Syntax

```
<html>
<head>
    <title>PHP Test </title>
</head>
<body>
    <?php echo “<p> Hello World </p>”; ?>
</body></html>
```

Note : - Code in bold letters shows the PHP code embedded within HTML

PHP only parses code within its delimiters. Anything outside its delimiters is sent

directly to the output and is not parsed by PHP. The most common delimiters are <?php and ?>, which are open and close delimiters respectively. <script language="php"> and </script> delimiters are also available. Short tags can be used to start PHP code, <? or <?= (which is used to echo back a string or variable) and the tag to end PHP code, ?>. These tags are commonly used, but like ASP-style tags (<% or <%= and %>), they are less portable as they can be disabled in the PHP configuration. For this reason, the use of short tags and ASP-style tags is discouraged. The purpose of these delimiters is to separate PHP code from non-PHP code, including HTML.

Variables are prefixed with a dollar symbol and a type does not need to be specified in advance. Unlike function and class names, variable names are case sensitive. Both double-quoted ("") and heredoc strings allow the ability to embed a variable's value into the string. PHP treats newlines as whitespace in the manner of a free-form language (except when inside string quotes), and statements are terminated by a semicolon. PHP has three types of comment syntax: /* */ serves as block comments, and // as well as # are used for inline comments. The echo statement is one of several facilities PHP provides to output text (e.g. to a web browser).

In terms of keywords and language syntax, PHP is similar to most high level languages that follow the C style syntax. *If* conditions, *for* and *while* loops, and

function returns are similar in syntax to languages such as C, C++, Java and Perl.

Data types

PHP stores whole numbers in a platform-dependent range. This range is typically that of 32-bit signed integers. Unsigned integers are converted to signed values in certain situations; this behavior is different from other programming languages. Integer variables can be assigned using decimal (positive and negative), octal, and hexadecimal notations. Floating point numbers are also stored in a platform-specific range. They can be specified using floating point notation, or two forms of scientific notation. PHP has a native Boolean type that is similar to the native Boolean types in Java and C++. Using the Boolean type conversion rules, non-zero values are interpreted as true and zero as false, as in Perl and C++. The null data type represents a variable that has no value.

The only value in the null data type is *NULL.* Variables of the "resource" type represent references to resources from external sources. These are typically created by functions from a particular extension, and can only be processed by functions from the same extension; examples include file, image, and database resources. Arrays can contain elements of any type that PHP can handle, including resources, objects, and even other arrays. Order is preserved in lists of values and in hashes with both keys and values, and the two can be intermingled. PHP also supports strings, which can be used with single quotes, double quotes, or heredoc syntax. The Standard PHP Library (SPL) attempts to solve standard problems and implements efficient data access interfaces and classes.

Functions

PHP has hundreds of base functions and thousands more from extensions. These functions are well documented on the PHP site, but unfortunately, the built-in library has a wide variety of naming conventions and inconsistencies. PHP currently has no functions for thread programming.

Version 5.2 and earlier

Functions are not first-class functions and can only be referenced by their name—directly or dynamically by a variable containing the name of the function. User-defined functions can be created at any time without being prototyped. Functions can be defined inside code blocks, permitting a run-time decision as to whether or not a function should be defined. Function calls must use parentheses, with the exception of zero argument class constructor functions called with the PHP new operator, where parentheses are optional. PHP supports quasi-anonymous functions through the create_function() function, although they are not true anonymous functions because anonymous functions are nameless, but functions can only be referenced by name, or indirectly through a variable $function_name();, in PHP.

Version 5.3 and newer

PHP gained support for first-class functions and closures. True anonymous functions are supported

function getAdder($x) using the following syntax :

```
function  getAdder($x)
    {
            return function ($y) use ($x) { return
            $x + $y;
            };
    }
    $adder  =  getAdder(8);
    echo  $adder(2);  //  prints  “10”
```

Here, getAdder() function creates a closure using parameter $x (keyword “use” forces getting variable from context), which takes additional argument $y and returns it to the caller. Such a function can be stored, given as the parameter to another functions, etc. For more details see Lambda functions and closures RFC.

Objects

Basic object-oriented programming functionality was added in PHP 3. Object handling was completely rewritten for PHP 5, expanding the feature set and enhancing performance. In previous versions of PHP, objects were handled like primitive types. The drawback of this method was that the whole object was copied when a variable was assigned or passed as a parameter to a method. In the new approach, objects are referenced by handle, and not by value. PHP 5 introduced private and protected member variables and methods, along with abstract classes and final classes as well as abstract methods and final methods. It also introduced a

standard way of declaring constructors and destructors, similar to that of other object-oriented languages such as C++, and a standard exception handling model.

Furthermore, PHP 5 added interfaces and allowed for multiple interfaces to be implemented. There are special interfaces that allow objects to interact with the runtime system. Objects implementing ArrayAccess can be used with array syntax and objects implementing Iterator or IteratorAggregate can be used with the foreach language construct. There is no virtual table feature in the engine, so static variables are bound with a name instead of a reference at compile time.If the developer creates a copy of an object using the reserved word *clone*, the Zend engine will check if a __clone() method has been defined or not. If not, it will call a default __clone() which will copy the object's properties. If a __clone() method is defined, then it will be responsible for setting the necessary properties in the created object. For convenience, the engine will supply a function that imports the properties of the source object, so that the programmer can start with a by-value replica of the source object and only override properties that need to be changed.

Resources

PHP includes free and open source libraries with the core build. PHP is a fundamentally Internet-aware system with modules built in for accessing FTP servers, many database servers, embedded SQL libraries such as embedded PostgreSQL, MySQL and SQLite, LDAP servers, and others. Many functions familiar to C programmers such as those in the stdio family are available in the standard PHP build. PHP has traditionally used features such as "magic_quotes_gpc" and

“magic_quotes_runtime” which attempt to escape apostrophes

(‘) and quotes (“) in strings in the assumption that they will be used in databases, to prevent SQL injection attacks. This leads to confusion over which data is escaped and which is not, and to problems when data is not in fact used as input to a database and when the escaping used is not completely correct. To make code portable between servers which do and do not use magic quotes, developers can preface their code with a script to reverse the effect of magic quotes when it is applied.

PHP allows developers to write extensions in C to add functionality to the PHP language. These can then be compiled into PHP or loaded dynamically at runtime. Extensions have been written to add support for the Windows API, process management on Unix-like operating systems, multibyte strings (Unicode), cURL, and several popular compression formats. Some more unusual features include integration with Internet Relay Chat, dynamic generation of images and Adobe Flash content, and even speech synthesis. The PHP Extension Community Library (PECL) project is a repository for extensions to the PHP language. Zend provides a certification exam for programmers to become certified PHP developers.

MY SQL

What is a database? Quite simply, it's an organized collection of data. A database management system (DBMS) such as Access, FileMaker Pro, Oracle or SQL Server provides you with the software tools you need to organize that data in a flexible manner. It includes facilities to add, modify or delete data from the database, ask questions (or queries) about the data stored in the database and produce reports summarizing selected contents.

MySQL is a multithreaded,multi-user SQL database management system(DBMS). The basic program runs as a server providing multi-user access to a number of databases.Originally financed in a similar fashion to the JBoss model, MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQLAB now a subsidiary of Sun Microsystem , which holds the copyright to most of the codebase. The project's source code is available under terms of the GNU General Public Licence, as well as under a variety of proprietory agreements.

MySQL is a database.The data in MySQL is stored in database objects called tables.A table is a collections of related data entries and it consists of columns and rows.Databases are useful when storing information categorically. A company may have a database with the following tables: "Employees", "Products", "Customers" and "Orders".

Database Tables

A database most often contains one or more tables. Each table is identified by a name (e.g. "Customers" or "Orders"). Tables contain records (rows) with data.

Queries

A query is a question or a request. With MySQL, we can query a database for specific information and have a record set returned.

Create a connection to a database

Before you can access data in a database, you must create a connection to the database.In PHP, this is done with the mysqli_connect() function.

Syntax

```
mysql_connect(servername,username,password);
```

Parameter	Description
servername	Optional. Specifies the server to connect to. Default value is "localhost:3306"
username	Optional. Specifies the username to log in with. Default value is the name of the user that owns the server process
password	Optional. Specifies the password to log in with. Default is ""

Example

In the following example we store the connection in a variable ($con) for later use in the script. The “die” part will be executed if the connection fails:

```
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
// some code
?>
```

Closing a Connection

The connection will be closed automatically when the script ends. To close the connection before, use the mysql_close() function:

```php
<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
// some code
mysql_close($con);
?>
```

Create a Database

The CREATE DATABASE statement is used to create a database in MySQL.

Syntax

CREATE DATABASE database_name

To get PHP to execute the statement above we must use the mysql_query() function. This function is used to send a query or command to a MySQL connection.

Create a Table

The CREATE TABLE statement is used to create a table in MySQL

Syntax

CREATE TABLE table_name

(

column_name1

data_type,

column_name2

data_type,

column_name3

data_type,

....

)

MySQL Functions

mysql_affected_rows — Get number of affected rows in previous MySQL

operation mysql_change_user — Change logged in user of the active

connection mysql_client_encoding — Returns the name of the character

set

mysql_close — Close MySQL connection

mysql_connect — Open a connection to a MySQL

Server mysql_create_db — Create a MySQL database

mysql_data_seek — Move internal result pointer

mysql_db_name — Get result data

mysql_db_query — Send a MySQL query

mysql_drop_db — Drop (delete) a MySQL

database

mysql_errno — Returns the numerical value of the error message from previous

MySQL operation mysql_error — Returns the text of the error message from

previous MySQL operation mysql_escape_string — Escapes a string for use in a

mysql_query

mysql_fetch_array — Fetch a result row as an associative array, a numeric

array, or both mysql_fetch_assoc — Fetch a result row as an associative

array

mysql_fetch_field — Get column information from a result and return

as an object mysql_fetch_lengths — Get the length of each output in a

result mysql_fetch_object — Fetch a result row as an object

wsmysql_num_rows — Get number of rows in result

mysql_pconnect — Open a persistent connection to a

MySQL server

mysql_ping — Ping a server connection or reconnect if there is

no connection mysql_query — Send a MySQL query

mysql_result — Get result data

mysql_select_db — Select a MySQL

database mysql_set_charset — Sets the

client character set mysql_stat — Get

current system status mysql_tablename

— Get table name of field

mysql_thread_id — Return the current

thread ID

mysql_unbuffered_query — Send an SQL query to MySQL, without fetching and

buffering the result (*See Appendix 2 for more My_SQL Functions.*)

Macromedia Dreamweaver 8

Is a professional HTML editor for designing, coding, and developing websites, web pages, and web applications. Whether you enjoy the control of hand-coding HTML or prefer to work in a visual editing environment, Dreamweaver provides you with helpful tools to enhance your web creation experience.

The visual editing features in Dreamweaver let you quickly create pages without writing a line of code. You can view all your site elements or assets and drag them from an easy-to-use panel directly into a document. You can streamline your development workflow by creating and editing images in Macromedia Fireworks or another graphics application, then importing them directly into Dreamweaver, or by adding Macromedia Flash objects.Dreamweaver also provides a full-featured coding environment that includes code-editing tools (such as code coloring and tag completion) and language reference material on Cascading Style Sheets (CSS), JavaScript, and ColdFusion Markup Language (CFML), among others. Macromedia Roundtrip HTML technology imports your hand-coded HTML documents without reformatting the code; you can then reformat code with your preferred formatting style. Dreamweaver also enables you to build dynamic database-backed web

applications using server technologies such as CFML, ASP.NET, ASP, JSP, and PHP.

Dreamweaver and accessibility

Accessibility refers to making websites and web products usable for people with visual, auditory, motor, and other disabilities. Examples of accessibility features for software products and websites include screen reader support, text equivalents for graphics, keyboard shortcuts, change of display colors to high contrast, and so on. Dreamweaver provides tools that make the product accessible and tools that help you author accessible content:

Using Dreamweaver accessibility features For Dreamweaver web designers who need to use accessibility features, Dreamweaver offers screen reader support, keyboard navigation, and operating system accessibility support.For more information, see Using Dreamweaver accessibility features.

Authoring for accessibility For Dreamweaver web designers who need to create accessible content, Dreamweaver assists you in creating accessible pages that contain useful content for screen readers and comply with government guidelines.

Dreamweaver provides dialog boxes that prompt you to enter accessibility attributes when you insert page elements (see Optimizing the workspace for accessible page design).For example, the accessibility dialog box for images reminds you to add text equivalents for graphics. Then, when the image appears on a page for a user with visual disabilities, the screen reader reads the description.

Laying Out Pages with CSS

In Macromedia Dreamweaver 8, you can use CSS styles to lay out your page. You can either insert div tags manually and apply CSS positioning styles to them, or you can use Dreamweaver layers to create your layout. A layer in Dreamweaver is an HTML page element—specifically, a div tag, or any other tag—that has an absolute position assigned to it. Whether you use CSS, tables, or frames to lay out your pages, Dreamweaver has rulers and grids for visual guidance in your layout. Dreamweaver also has a tracing image feature, which you can use to re-create a page design that was created in a graphics application.

Client-side role of forms

Forms support the client side of the client-server relationship. When a visitor enters information into a form displayed in a web browser (the client) and clicks the submit button, the information is sent to the server where a server-side script or application processes it. Common server-side technologies used for processing form data include Macromedia ColdFusion, Microsoft Active Server Pages (ASP), and PHP. The server responds by sending requested information back to the user (or client), or performing some action based on the form's contents.

(Note : - See Appendix 1 for more about Macromedia Dreamweaver 8 and phpMyAdmin)

phpmydmin

phpMyAdmin is an open source tool written in PHP intended to handle the administration of MySQL over the World Wide Web. phpMyAdmin supports a wide range of operations with MySQL.Currently it can create and drop databases, create/drop/alter tables, delete/edit/add fields, execute any SQL statement, manage users and permissions, and manage keys on fields. while you still have the ability to directly execute any SQL statement. phpMyAdmin can manage a whole MySQL server (needs a super-user) as well as a single database. To accomplish the latter you'll need a properly set up MySQL user who can read/write only the desired database. It's up to you to look up the appropriate part in the MySQL manual.

phpMyAdmin can:

- browse and drop databases, tables, views, fields and indexes
- create, copy, drop, rename and alter databases, tables, fields and indexes
- maintenance server, databases and tables, with proposals on server configuration
- execute, edit and bookmark any SQL-statement, even batch-queries
- load text files into tables
- create and read dumps of tables
- export data to various formats: CSV, XML, PDF, ISO/IEC 26300 - OpenDocument Text and Spreadsheet, Word, Excel and L^AT_EX formats

- administer multiple servers
- manage MySQL users and privileges
- check referential integrity in MyISAM tables
- using Query-by-example (QBE), create complex queries automatically connecting required tables
- create PDF graphics of your Database layout
- search globally in a database or a subset of it
- transform stored data into any format using a set of predefined functions,

  like displaying BLOB-data as image or download-link
- support InnoDB tables and foreign keys
- support mysqli, the improved MySQL extension

A word about users:
Many people have difficulty understanding the concept of user management with regards to phpMyAdmin. When a user logs in to phpMyAdmin, that username and password are passed directly to MySQL. phpMyAdmin does no account management on its own (other than allowing one to manipulate the MySQL user account information); all users must be valid MySQL users.

1) phpMyAdmin can compress (Zip, GZip -RFC 1952- or Bzip2 formats) dumps and CSV exports if you use PHP with Zlib support (—with-zlib) and/or Bzip2 support (—with-bz2). Proper support may also need changes in php.ini.a phpMyAdmin screen appears as shown below.

Requirements

PHP

- You need PHP 5.6.0 or newer, with session support and the Standard PHP Library (SPL) extension.
- To support uploading of ZIP files, you need the PHP zip extension.
- For proper support of multibyte strings (eg. UTF-8, which is currently default), you should install mbstring and ctype extensions.
- You need GD2 support in PHP to display inline thumbnails of JPEGs (“image/jpeg: inline”) with their original aspect ratio
- When using the “cookie” authentication method, the mcrypt extension is strongly suggested for most users and is required for 64–bit machines. Not using mcrypt will cause phpMyAdmin to load pages significantly slower.

Apache Web server

Often referred to as simply *Apache*, a public-domain open source Web server developed by a loosely-knit group of programmers. The first version of Apache, based on the NCSA httpd Web server, was developed in 1995.

Core development of the Apache Web server is performed by a group of about 20 volunteer programmers, called the *Apache Group.* However, because the source code is freely available, anyone can adapt the server for specific needs, and there is a large public library of Apache add-ons. In many respects, development of Apache is similar to development of the Linux operating system.

The original version of Apache was written for UNIX, but there are now versions that run under OS/ 2, Windows and other platforms. The name is a tribute to the Native American Apache Indian tribe, a tribe well known for its endurance and skill in warfare. A common misunderstanding is that it was called Apache because it was developed from existing NCSA code plus various patches, hence the name *a patchy server*, or Apache server.

Apache consistently rates as the world's most popular Web server according to analyst surveys. Apache has attracted so much interest because it is full-featured, reliable, and free. Originally developed for UNIX™ operating systems, Apache has been updated to run on Windows, OS/2, and other platforms. One aspect of Apache that some site administrators find confusing — especially those unfamiliar with

UNIX-style software — is its configuration scheme. Instead of using a point-and-click graphic user interface (GUI) or Windows Registry keys as most other.

Configuration Files

Apache uses a system of three text files for managing its configuration data. All three of these files (almost always) appear in Apache's ./conf directory and are designed to be edited by system administrators:

1. httpd.conf for general settings
2. srm.conf for resource settings
3. access.conf for security settings

When Apache first starts, these files are processed in the order shown above. Originally, the initial installation of Apache included default entries within each of the three files. In the most recent versions of Apache, however, the default installation has changed. Now httpd.conf is treated as the "master" configuration file and it contains all of the settings. Both srm.conf and access.conf still exist in the installation, but they contain no settings and are empty except for some comments.

Inside Httpd.conf

Traditionaly httpd.conf contained general settings such as the ServerName and Port number.These entries appear as follows in the file: ServerName compnetworking.about.com Port 80 The term "httpd" stands for *HTTP Daemon*. Recall that in a UNIX environment, the term *daemon* refers to a type of process designed to launch at system boot and continue running for very long periods of time. This file contains a number of other entries (technically called directives), but

for most of these,modifications are optional. Probably the most useful of these entries is ServerAdmin.

Access and Security Settings

It is recommended practice now for Apache administrators to manage their resource and security settings from httpd.conf. Administrators of older versions of Apache can simply cut their entries from srm.conf and access.conf and paste them into the master file. If an administrator wants to go one step further and delete the two empty files, they should also place the following entries in httpd.conf to prevent Apache from attempting to access them.

SYSTEM DESIGN

DATA DESIGN DIAGRAMS

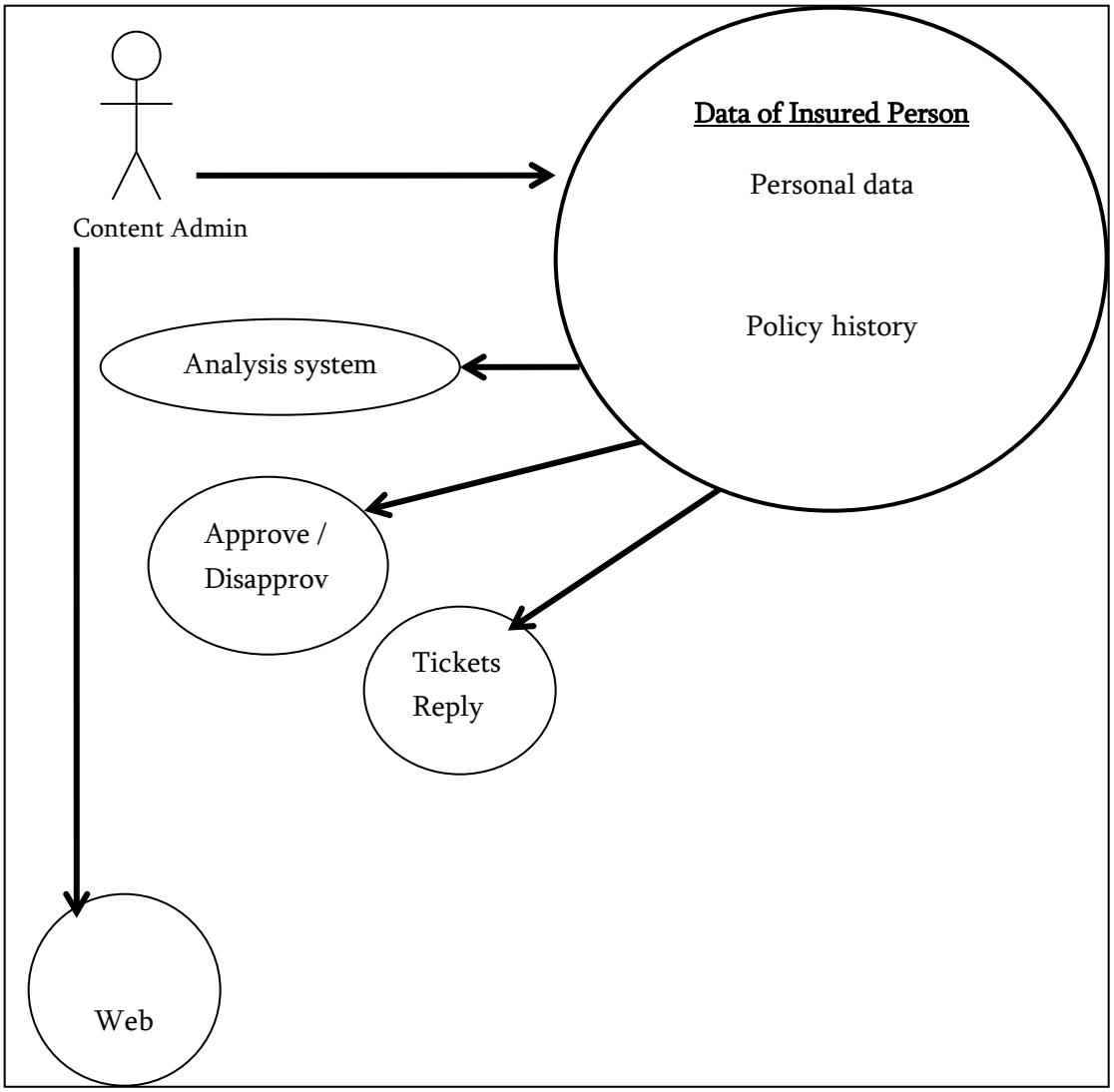
UML is a latest concept of presenting various diagrams of a project. UML stands for Unified Modeling Language.

The purpose of this section is to show the following type of conceptual/logical diagrams of the proposed project.

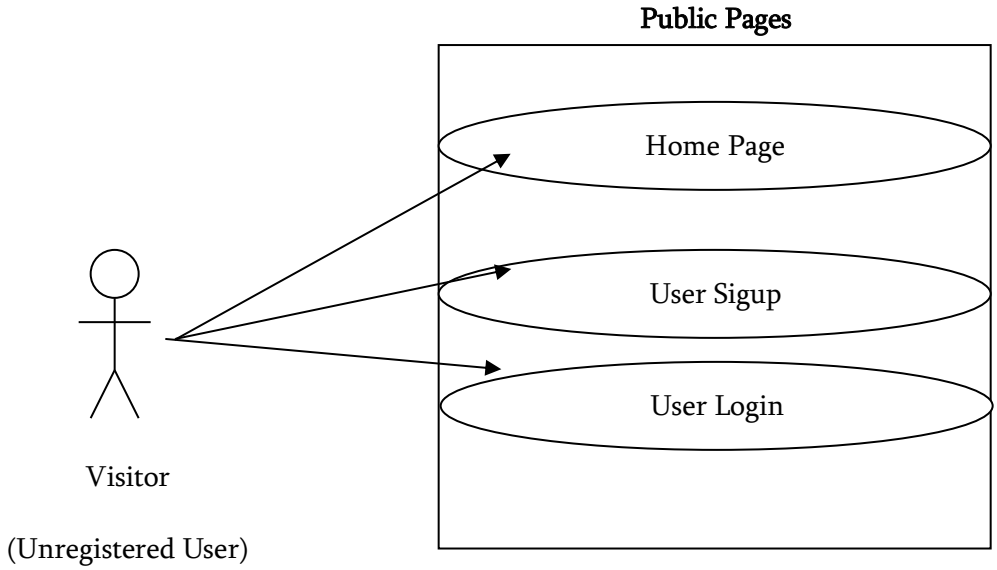
- Data Flow diagram of the project
- Use case diagrams
- Activity diagrams

Data flow diagram explains the overall view of utilization of user's data in different levels/modules of the project. Use case diagrams explain main entities and their features. Activity diagrams explain the main activities/processes present in the project**.**

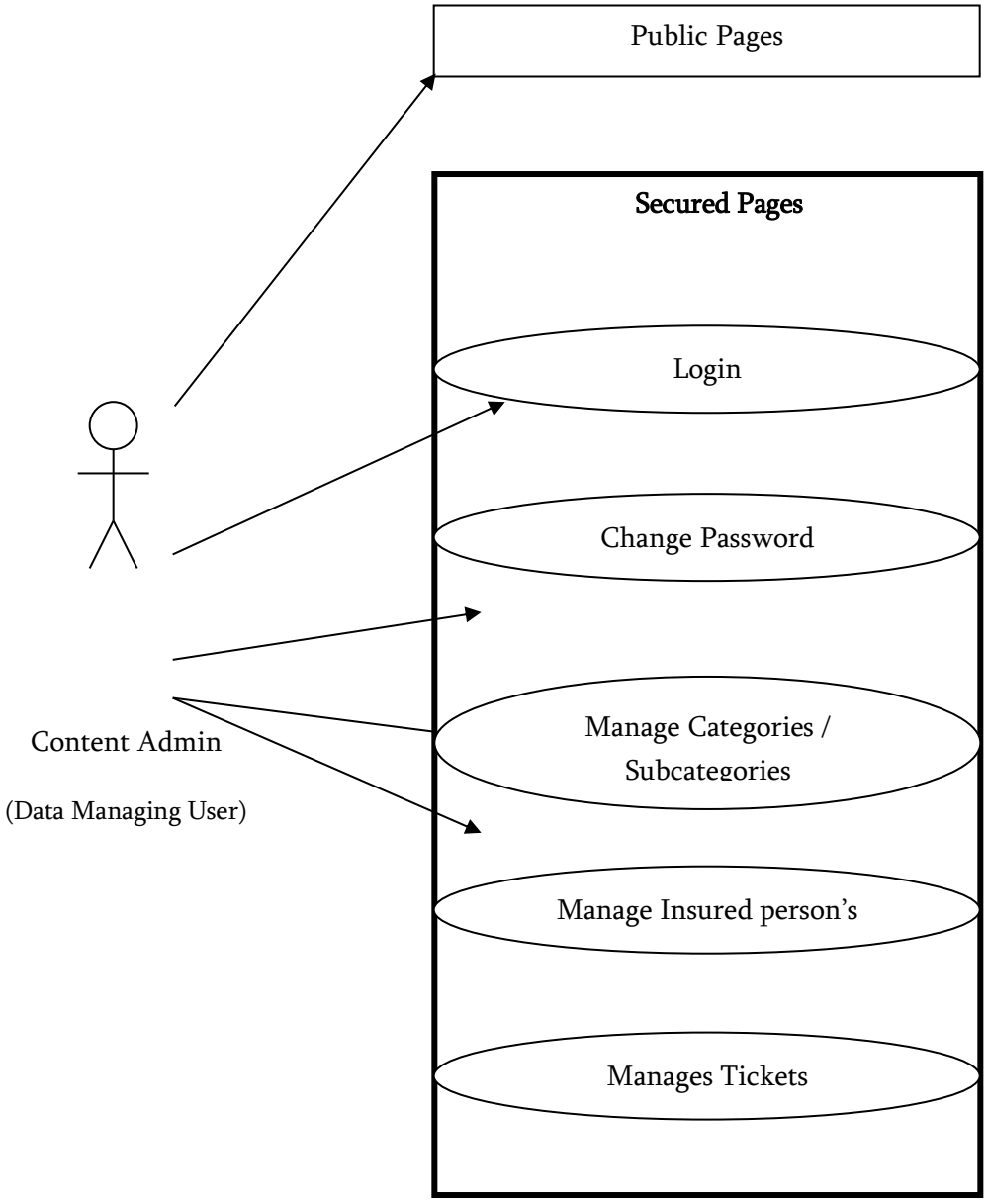
DATA FLOW DIAGRAM



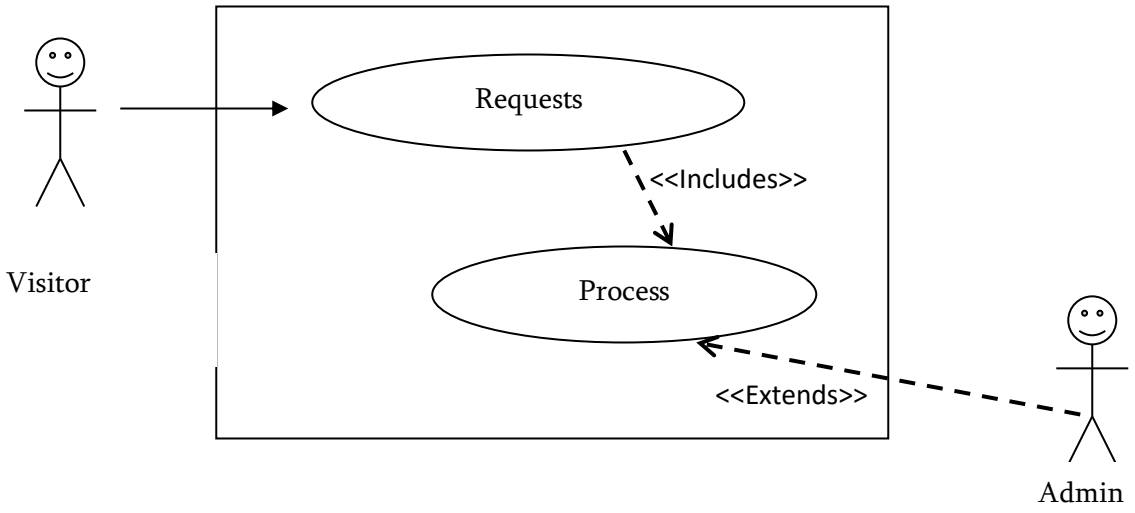
USE CASE DIAGRAMSUse Case Diagram: Visitor and Facilities

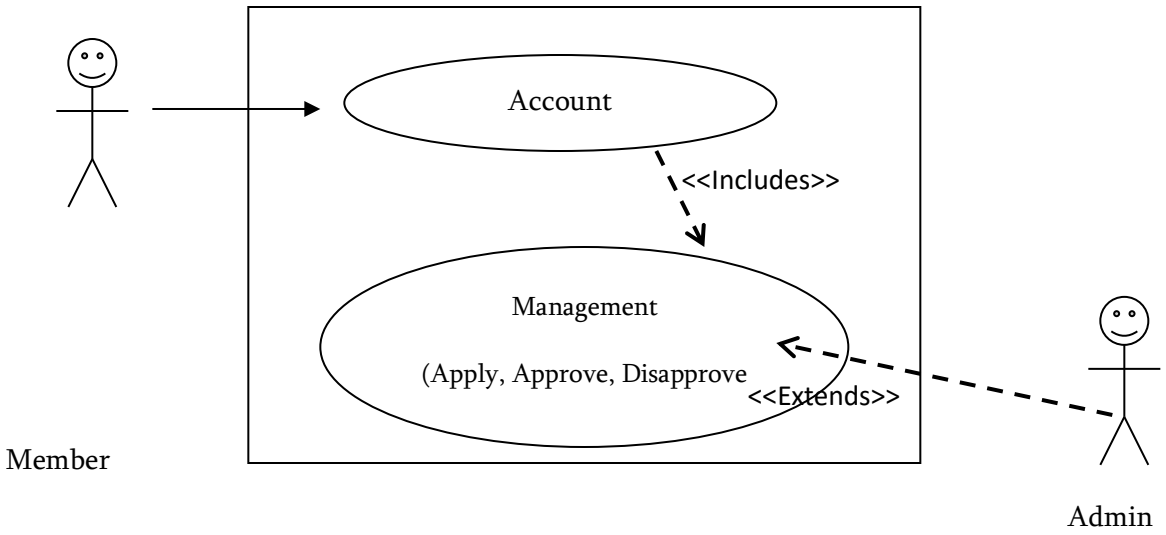


Use Case Diagram: Online Registered Members and Facilities



USE CASE RELATIONSHIP DIAGRAMS


Requests
<<Includes>>
Process
Visitor
<<Extends>>
Admin


Account
<<Includes>>
Management
(Apply, Approve, Disapprove
<<Extends>>
Member
Admin

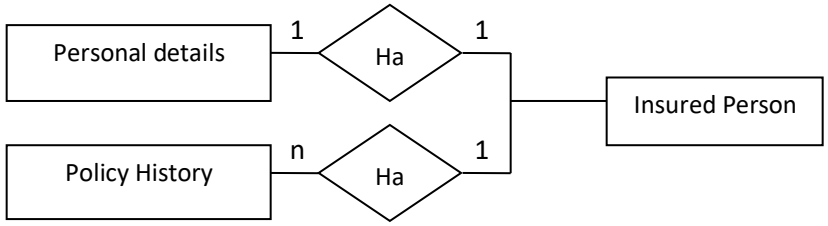
Entity Relation (ER) Diagrams

ER (Entity Relation) diagrams

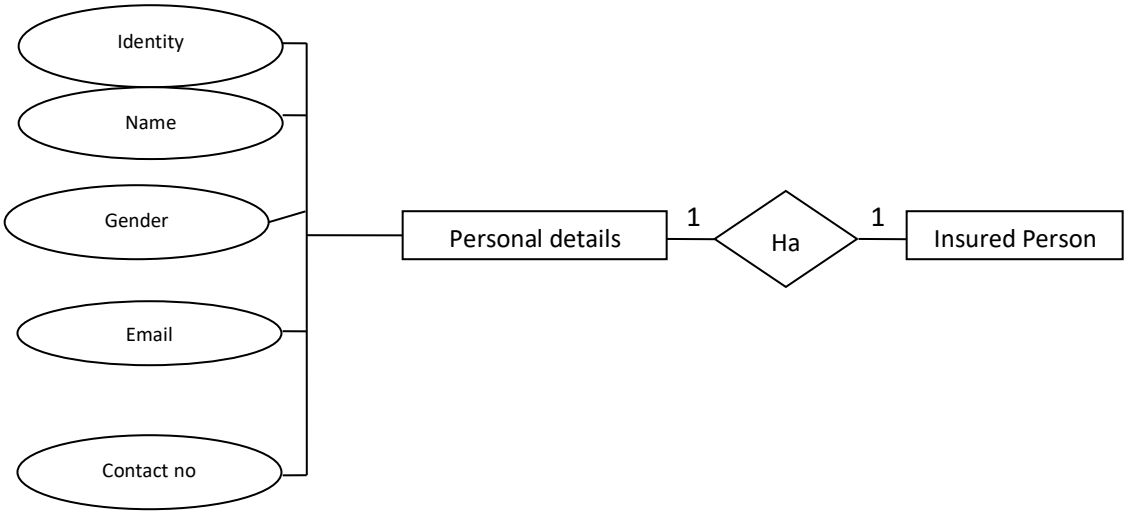
An ER diagram gives a clear idea of logical relation between two or more entities. These diagrams help us to understand the technical design of database involved in the project.

The following relational diagrams show the logical relation between the entities.

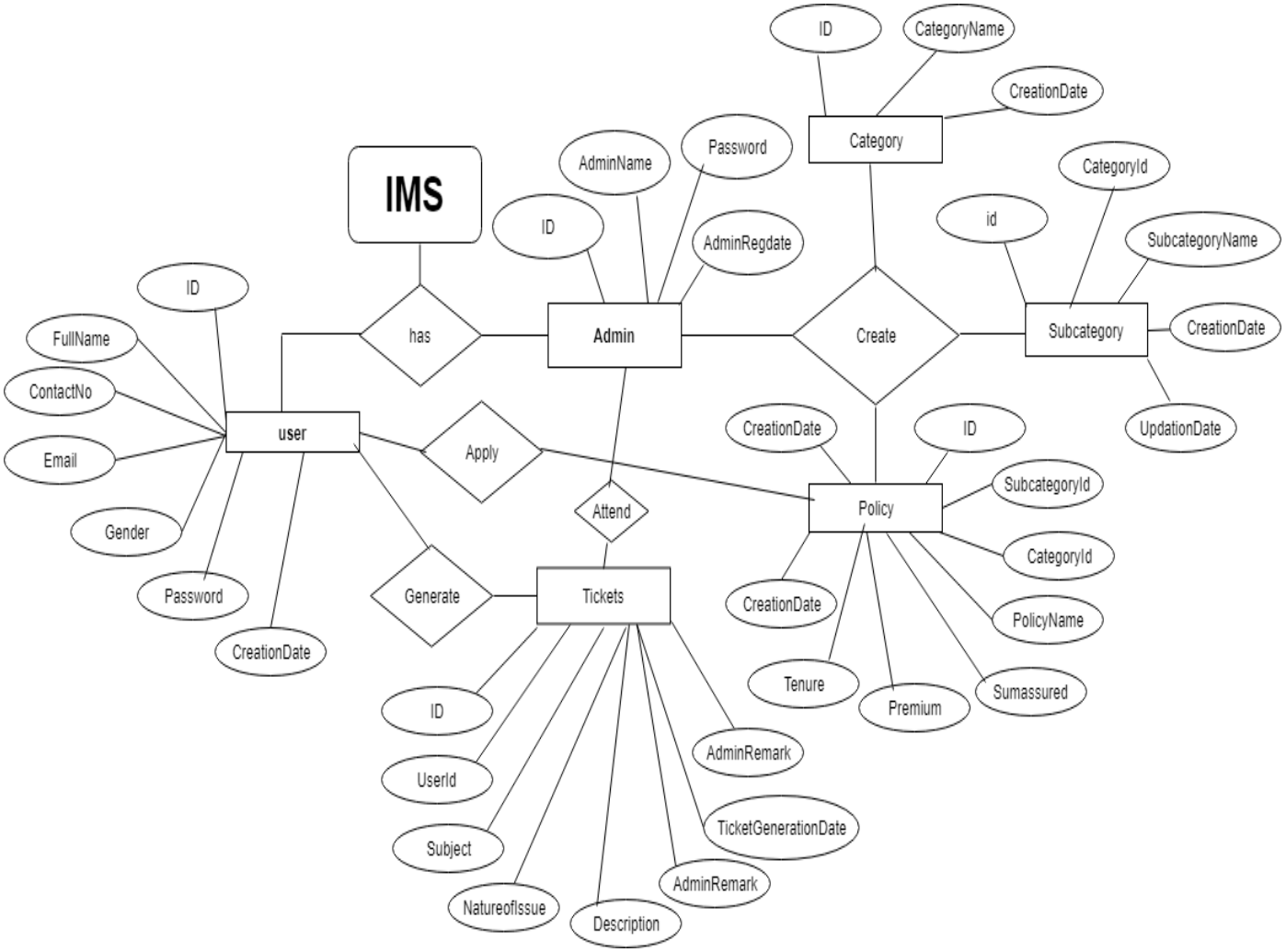
Insured Person's details diagram



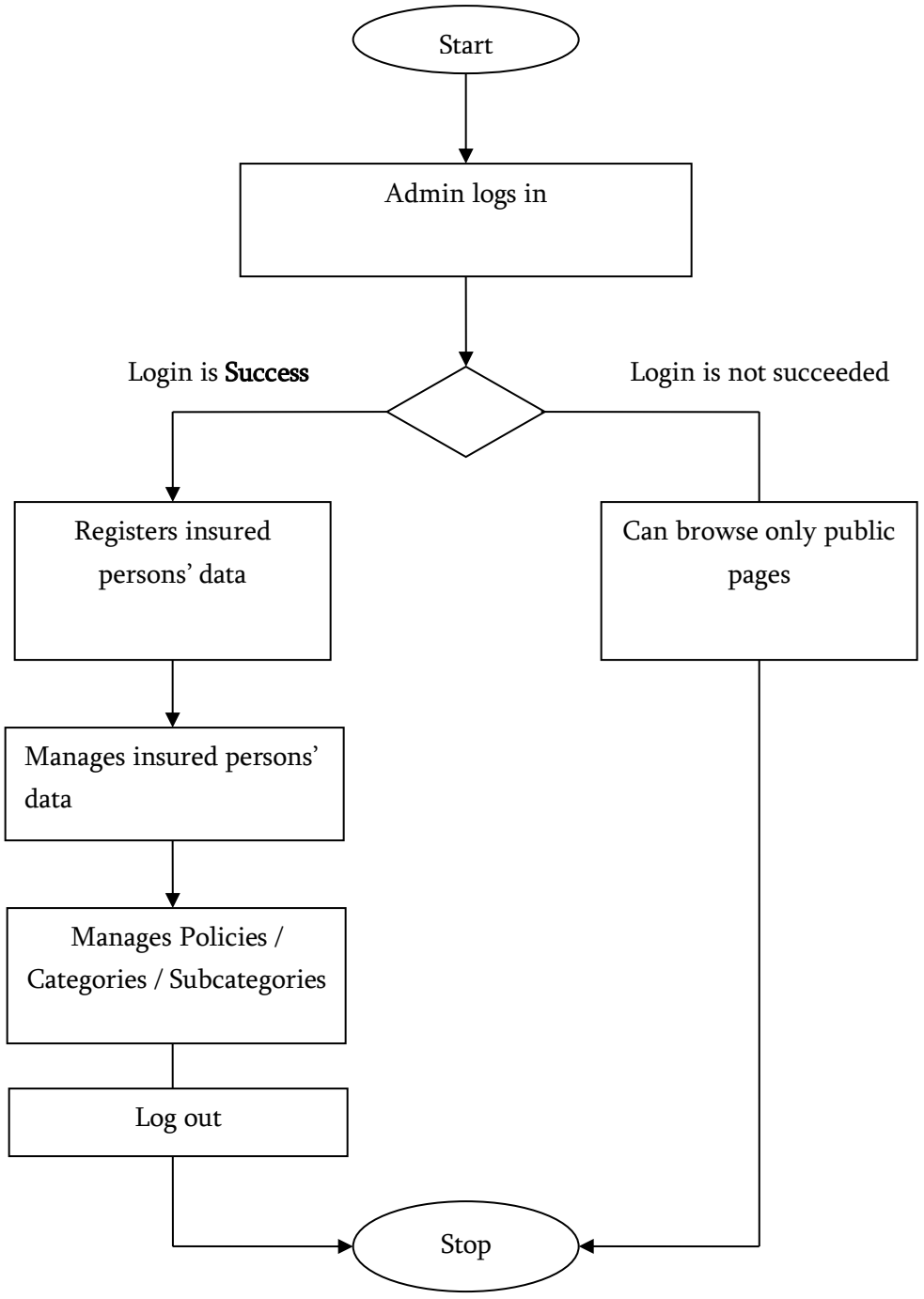
Insured Person – Personal details relation



ER- Diagram


IMS
Admin
AdminName
Password
ID
AdminRegdate
has
user
ID
FullName
ContactNo
Email
Gender
Password
CreationDate
Apply
Attend
Generate
Tickets
ID
UserId
Subject
NatureofIssue
Description
AdminRemark
TicketGenerationDate
AdminRemark
Category
ID
CategoryName
CreationDate
Create
Subcategory
id
CategoryId
SubcategoryName
CreationDate
UpdationDate
Policy
CreationDate
ID
SubcategoryId
CategoryId
PolicyName
Sumassured
CreationDate
Tenure
Premium

FUNCTIONAL FLOW DIAGRAM



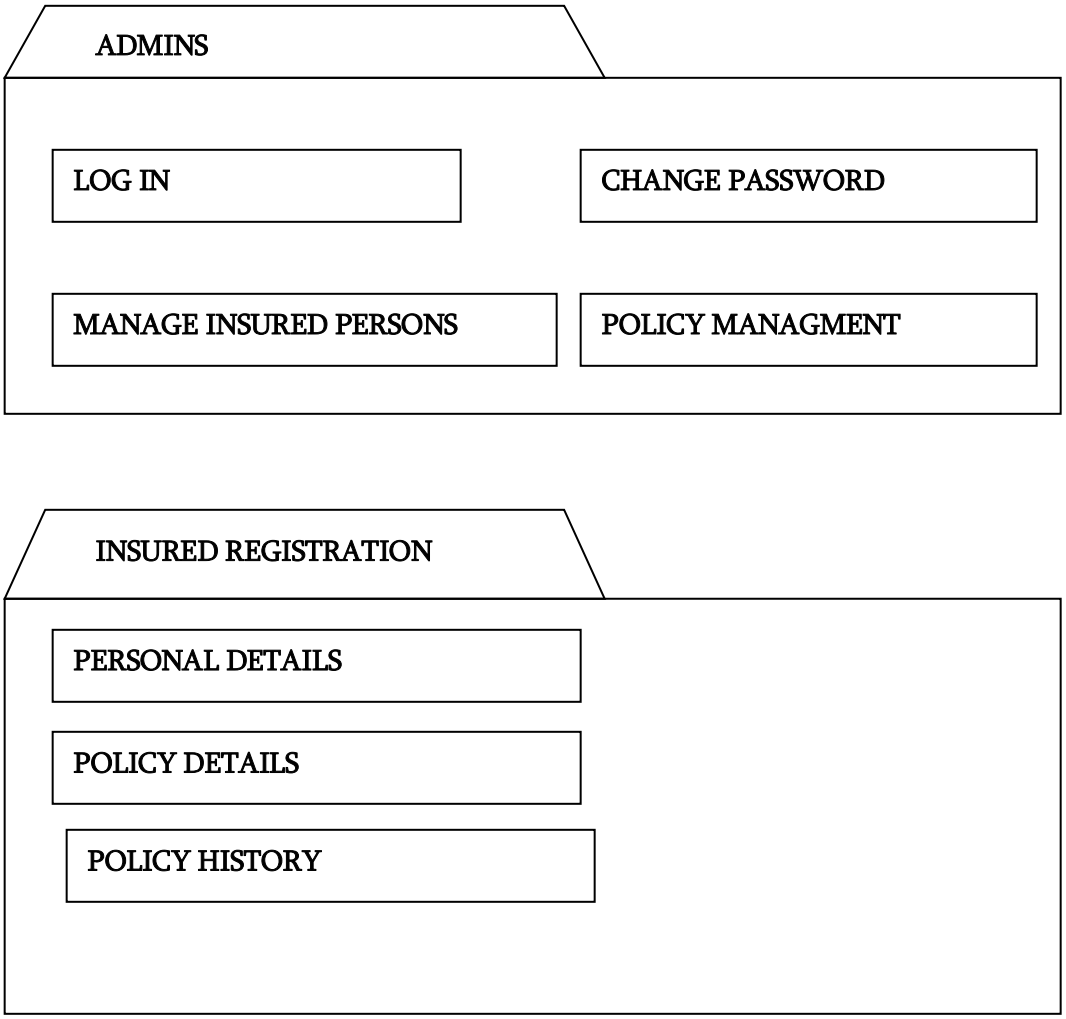
Subsystem decomposition

This section shows the all-functional objects in the project and their functionality.

The following list of functional object as identified in the project.

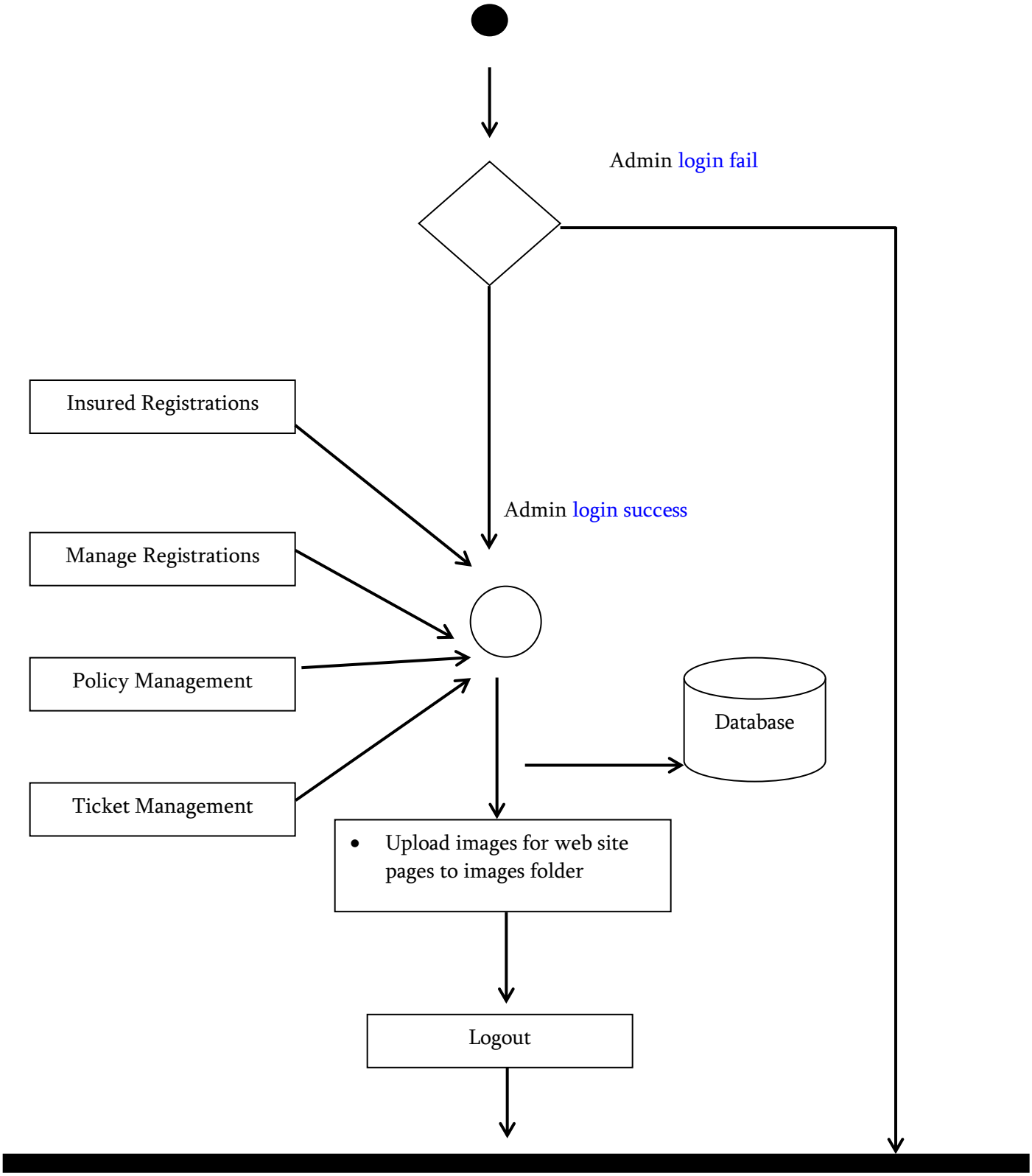
- Login
- Manage Categories / Subcategories / Policies
- Manages Insure Persons data
- Manages Articles

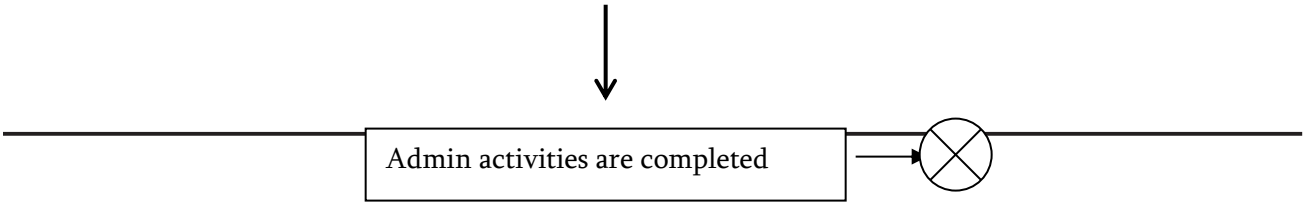
Insured person's



Activity diagrams

ACTIVITY DIAGRAM OF ADMIN ACTIVITIES



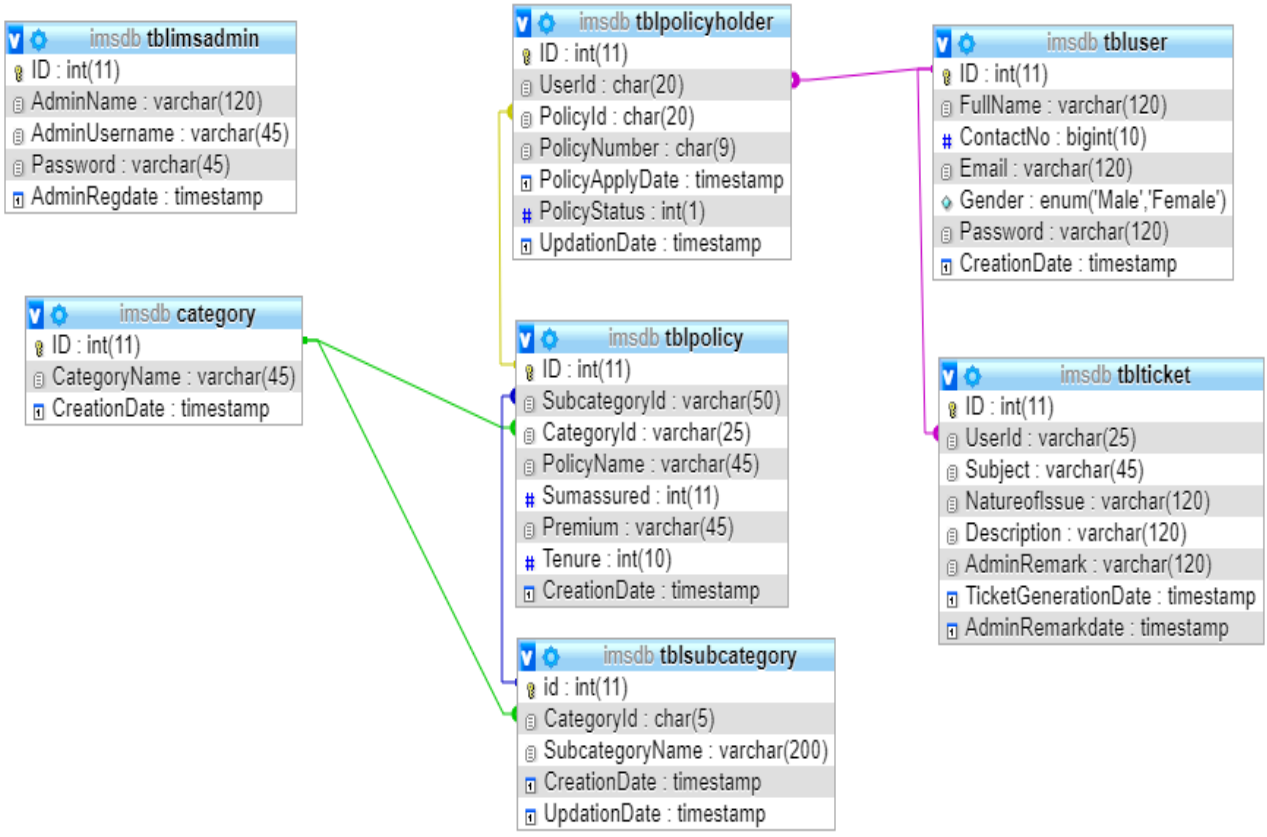

Admin activities are completed

Database Design

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MS Access database has been chosen for developing the relevant databases.

Database Table Relationship



MySQL Table and their Structure

tblimsdmin table Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	AdminName	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
3	AdminUsername	varchar(45)	latin1_swedish_ci		Yes	*NULL*		
4	Password	varchar(45)	latin1_swedish_ci		Yes	*NULL*		
5	AdminRegdate	timestamp			Yes	CURRENT_TIMESTAMP		

category table Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	CategoryName	varchar(45)	latin1_swedish_ci		Yes	*NULL*		
3	CreationDate	timestamp			Yes	CURRENT_TIMESTAMP		

Tblsubcategory table Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	*None*		AUTO_INCREMENT
2	CategoryId	char(5)	latin1_swedish_ci		No	*None*		
3	SubcategoryName	varchar(200)	latin1_swedish_ci		No	*None*		
4	CreationDate	timestamp			Yes	CURRENT_TIMESTAMP		
5	UpdationDate	timestamp		on update CURRENT_TIMESTAMP	Yes	*NULL*		ON UPDATE CURRENT_TIMESTAMP

Tblpolicy table Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	SubcategoryId	varchar(50)	latin1_swedish_ci		No	*None*		
3	CategoryId	varchar(25)	latin1_swedish_ci		Yes	*NULL*		
4	PolicyName	varchar(45)	latin1_swedish_ci		Yes	*NULL*		
5	Sumassured	int(11)			Yes	*NULL*		
6	Premium	varchar(45)	latin1_swedish_ci		Yes	*NULL*		
7	Tenure	int(10)			Yes	*NULL*		
8	CreationDate	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP		ON UPDATE CURRENT_TIMESTAMP

Tbluser table Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	FullName	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
3	ContactNo	bigint(10)			Yes	*NULL*		
4	Email	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
5	Gender	enum('Male', 'Female')	latin1_swedish_ci		Yes	*NULL*		
6	Password	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
7	CreationDate	timestamp			No	CURRENT_TIMESTAMP		

Tblpolicyholder table Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	UserId	char(20)	latin1_swedish_ci		Yes	*NULL*		
3	PolicyId	char(20)	latin1_swedish_ci		Yes	*NULL*		
4	PolicyNumber	char(9)	latin1_swedish_ci		Yes	*NULL*		
5	PolicyApplyDate	timestamp			Yes	CURRENT_TIMESTAMP		
6	PolicyStatus	int(1)			Yes	*NULL*		
7	UpdationDate	timestamp		on update CURRENT_TIMESTAMP	No	0000-00-00 00:00:00		ON UPDATE CURRENT_TIMESTAMP

tblticket table Structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	UserId	varchar(25)	latin1_swedish_ci		Yes	*NULL*		
3	Subject	varchar(45)	latin1_swedish_ci		Yes	*NULL*		
4	NatureofIssue	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
5	Description	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
6	AdminRemark	varchar(120)	latin1_swedish_ci		Yes	*NULL*		
7	TicketGenerationDate	timestamp			Yes	CURRENT_TIMESTAMP		
8	AdminRemarkdate	timestamp		on update CURRENT_TIMESTAMP	Yes	*NULL*		ON UPDATE CURRENT_TIMESTAMP

Output Design

Designing computer output should proceed in an organized, well throughout manner; the right output element is designed so that people will find the system whether or executed. When we design an output we must identify the specific output that is needed to meet the system. The usefulness of the new system is evaluated on the basis of their output.

Once the output requirements are determined, the system designer can decide what to include in the system and how to structure it so that the require

output can be produced. For the proposed software, it is necessary that the output reports be compatible in format with the existing reports. The output must be concerned to the overall performance and the system's working, as it should. It consists of developing specifications and procedures for data preparation, those steps necessary to put the inputs and the desired output, ie maximum user friendly. Proper messages and appropriate directions can control errors committed by users.

The output design is the key to the success of any system. Output is the key between the user and the sensor. The output must be concerned to the system's working, as it should.

Output design consists of displaying specifications and procedures as data presentation. User never left with the confusion as to what is happening without appropriate error and acknowledges message being received. Even an unknown person can operate the system without knowing anything about the system.

SYSTEM ANALYSIS

Feasibility study

- **Functional Feasibility**

The proposed/developed system is a web application and does not require any additional software requirements except hosting. A web hosting company hosts the web application in their web servers. Any client (end-user) with an internet connection and a web browsing software (at client side) can immediately start accessing/using the system. So, the system does not need any installation/setup procedure. Hence, the system is functionally feasible.

- **Economic Feasibility**

Author (project stakeholder) does not need to purchase any software and hardware to host the developed system. Web hosting companies purchase and maintain all necessary hardware and software for hosting the websites. Author has to pay only hosting charges to the hosting company. When it comes to the end-user, he/she does not need to purchase any software to use the website. Now-a-days every computer's operating system is providing built in web browser and so many web browsing software products are available in the market for free.

Only the cost that the author should bare is cost of development. Before starting the development, the development team estimates cost of development depends on the features asked by the client. If any new features to be added as per the new requirements of the client, cost of new additions are submitted to the client and after getting the confirmation from the client the new features will be added by the development team. Development team will take care of in-time delivery of the project to avoid excess cost of development due to delay.

Hence, the proposed system is economically feasible.

- **Technical feasibility**

This is about acceptance of the new system by the existing end-users and employees of the system and author. The website is aiming at employees and now-a-days all of them are aware of web browsing. If any person does not have any idea of web browsing, he/she will be trained to use the system with in one hour time. So, the system can easily be accepted by any kind of end-user. Hence the proposed system is technically feasible.

SYSTEM TESTING

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is the process of executing the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. The ultimate aim is quality assurance.

Tests are carried out and the results are compared with the expected document. In the case of erroneous results, debugging is done. Using detailed testing strategies a test plan is carried out on each module. The various tests performed in "**Network Backup System**" are unit testing, integration testing and user acceptance testing.

Unit Testing

The software units in a system are modules and routines that are assembled and integrated to perform a specific function. Unit testing focuses first on modules, independently of one another, to locate errors. This enables, to detect errors in coding and logic that are contained within each module. This testing includes entering data and ascertaining if the value matches to the type and size supported by java. The various controls are tested to ensure that each performs its action as required.

Integration Testing

Data can be lost across any interface, one module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integration testing is a systematic testing to discover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here the Server

module and Client module options are integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data.

User Acceptance Testing

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the system users at time of developing and making changes whenever required.

IMPLEMENTATION

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over, an evaluation of change over methods. Apart from planning major task of preparing the implementation are education and training of users. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. In network backup system no additional resources are needed.

Implementation is the final and the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it is found to be working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

7.1 User Training

After the system is implemented successfully, training of the user is one of the most important subtasks of the developer. For this purpose user manuals are

prepared and handled over to the user to operate the developed system. Thus the users are trained to operate the developed system. Both the hardware and software securities are made to run the developed systems successfully in future. In order to put new application system into use, the following activities were taken care of:

- Preparation of user and system documentation
- Conducting user training with demo and hands on
- Test run for some period to ensure smooth switching over the system

The users are trained to use the newly developed functions. User manuals describing the procedures for using the functions listed on menu are circulated to all the users. It is confirmed that the system is implemented up to users need and expectations.

Security and Maintenance

Maintenance involves the software industry captive, typing up system resources .It means restoring something to its original condition. Maintenance follows conversion to the extend that changes are necessary 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's operation. Maintenance is also done based on fixing the problems reported, changing the interface with other software or hardware enhancing the software.

Any system developed should be secured and protected against possible hazards. Security measures are provided to prevent unauthorized access of the database at various levels. An uninterrupted power supply should be so that the power failure or voltage fluctuations will not erase the data in the files.

CONCLUSION

The developed project fulfills the website facilities estimated for pahse-1 development and as per all the currently addressed requirements of the client (stakeholder).

Development team will provide

- Uploading and trail running of the website.
- Plan to avoid/handle unexpected damages
- Probable list of modifications that will guide the pahse-2 development of the project.

BIBILIOGRAPHY

1. www.w3schools.com

*2. in.**php**.net*

*3. en.wikipedia.org/wiki/**PHP***

4 . www.hotscripts.com/category/**php**/

5. www.**apache**.org/

6. www.**mysql**.com/click.php?e=35050