A Project Report On

**Project Management System**

**(Using PHP and MySQL)**

*In partial fulfillment for 5th semester degree of*

MASTER OF COMPUTER APPLICATION

Of

**BANGALORE UNIVERSITY**

By

**Raj Shrestha(13DVSCA008)**

**Rupesh Shrestha(13DVSCA010)**

Under the Guidance of

**Ms. S.VIJAYA RANI**



Brindavan College of MBA & MCA

Dwarkanagar, Baglur Main Road,

Yelahanka, Bangalore-63

Academic Year 2015-16

BrindavanCollege of MBA & MCA

(Formally NATIONAL FIRST GRADE COLLEGE)

Approved by AICTE, Affiliated to Bangalore University

Dwarkanagar, Yelahanka, Bangalore – 560063.



Brindavan College

MASTER OF COMPUTER APPLICATION

*Certificate*

This to certify that Mr Raj Shrestha**(13DVSCA008)** of 5th semester MCA has successfully completed the mini project entitled “**Project Management System”** for the practical **Software Engineering Lab(5MCA7)** in partial fulfillment of the requirement of MCA degree course of Bangalore University for the academic year 2015-16.

Project Guide Head of the Department

**Examiners:**

1. …………………………… Registration no………………….....

2. ……………………………............... Date of Examination ……………............

**Acknowledgement**

The project on **Project Management System** had profoundly reinforced our theoretical understanding of the concepts learnt. The project also provided us an avenue to understand the working of the system in general and software tools- HTML,PHP and MySql in particular. This we credit, the Bangalore University for its innovative and practical curriculum.

The satisfaction that accompanies the successful completion of any task would be incomplete without acknowledging those who have made it possible and to those who’s constant encouragement and guidelines has been a source of inspiration throughout the course of this project.

We sincerely thank Ms S.Vijaya Rani for her guidance. She went out of her way in guiding us, besides sorting out technical and logistic hiccups along the life cycle of the project.

We also thank lectures of the Department of MCA for their experiential sharing, concern and illuminating us with those concepts and precepts of building a project. Their clarifications and critical feedbacks accelerated the progress and quality of the project.

**Raj Shrestha**

# ABSTRACT

In a company where the hierarchy of employees spans over thousands, managing work with them is a difficult job. And in an environment where number of jobs is done simultaneously, picking the right person for the job is also a difficult task, as we are not aware of their availability. This application is designed for such an environment where the work is divided into group of employees and during the course of division the employees are selected to be part of the work in hand.

This software being a web based is easily accessible from any corner of the company as every machine is part of a LAN network. The reason why it is made as a web application rather than a window based application if for the same reason. The complete task is divided into two types of users the Teamleader and the Employee .This application provides most of the features required to manage the tasks to develop a project in a firm.

Project Management System is a dynamic process that utilises the appropriate resources of the organisation in a controlled and structured manner, to achieve some clearly defined objectives identified as needs.

It is necessary to Track or Measure the progress we have achieved towards a Goal that we wish to accomplish. We use Project Management to Aid us in **maximizing** and **optimizing** our resources to accomplish our goals.

# LITERATURE SURVEY

PHP is a server –side Scripting language designed for web development but also used as a general purpose programming language. PHP is now installed 224 million websites and 2.1 million web server. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by the PHP Group. While PHP originally stood for personal Home page, it now stand for PHP: Hypertext preprocessor, a recursive acronym.

PHP code is interpreted by a web server with PHP processor module which generated the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data, it to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical application.

PHP free software released under the PHP License, which is incompatible with the GNU General public license (GPL) due to restrictions on the usage of the term PHP.PHP can be deploying on most web server and also as a standalone shell on almost every operating system and platform, free of charge.

PHP is the web development language. PHP stands for PHP: preprocessor. PHP is a server-side scripting language, which can be embedded in HTML or used as a standalone binary.

Strictly speaking, PHP has little to do with layout, events, on the fly DOM manipulate, or really anything about what a web page looks and sounds like. In fact, most of what PHP does is invisible to the end user. Someone looking at a PHP page will not necessarily be able to tell that it was not written purely in HTML, because the result of PHP is HTML. The PHP preprocessor has two modes of operation, copy mode and interpret mode. It takes a PHP document file as input and produces an HTML document file. PHP is usually purely interpreted. The syntax and semantics of PHP are closely related to the syntax of JavaScript and Perl. It uses dynamic typing. PHP has extensive library of functions, making it a flexible and powerful tool for server-side software development.

MYSQL is an open source. MYSQL is a free, highly efficient, widely used database system that implements SQL. There are a plethora of tools, both in MYSQL itself and available from third parties, to make this job even easier. MYSQL isn’t a database until you give it some structure and form.

**Initial Investigation**

During system development process,many organizations don't analyze the system. The initial investigation should be carried out in general to know about the existing system,the work procedure,information flow,problems in existing system and so on.The initial investigation helps to provide the base to build the new system.The initial investigation is done to know the user requirements, functional and non-functional requirements, etc.For initial investigation, the following approaches were used:

* Observation
* Detailed Study
* Interview

## Problem Definition

In this system,the teamleader delegates tasks to employees.The employees perform the tasks.If solve issue is there a ticket is generated.The generated ticket contains the detail of the issue.The appropriate person resolves the issue and notifies the teamleader.

## Evaluation of Existing System

* The current system is a manual one where in the company maintains all the information in the form of records. There by collecting necessary information with require a manual search in the record(s).
* Transfer of information between different sections of the enterprise is in the form of documents or letters. Drafting letters will take time.
* Selection of a person for a task is done by manually approaching the person and confirming the availability of the person.
* Unavailability of proper information to different levels of employees within the firm.

**Proposed System**

The development of the new system contains the following activities, which try to automate the entire process keeping in view of the database integration approach.

* User friendliness is provided in the application with various controls.
* The system makes the overall project management much easier and flexible.
* Readily upload the latest updates, allows user to download the alerts by clicking the URL.
* There is no risk of data mismanagement at any level while the project development is under process.
* It provides high level of security with different level of authentication.

## Software Selection

In order for the proposed system to be technically sound, choosing the proper implementation tools are extremely Important as well. This is the first step in the process of implementing the designed solution. When selecting the development platform it includes the selection of hardware and software, which are needed to build the system. Many aspect need to be the consider when selecting tools, to make sure that requirement are met. Therefore this selection explains the necessary concerns regarding the system of development tools in order to build a good system. First the operating system will be selected and the development language and database were selected depending on pre-selected operating system.

### OS Selection

The performance of every product depends on its operating system. Thus a selecting of operating system has to handle in extreme care. Mainly used operating system; windows 2000/XP/7. Selection of operating system is mainly based o user’s familiarity with them. The operating system is selected under the following criteria.

To handle in extreme care. Mainly used operating system; windows 2000/XP/7. Selection of operating system is mainly based o user’s familiarity with them. The operating system is selected under the following criteria.

#### Reliability and Scalability

For any system to be performed the operations accurately, reliability and stability of operating system.

#### Security

Security is one of the key features in any operating system. Since information would be used for strategic decision making, secrecy of data and knowledge is an essential requirement.

**Familiarity and Popularity**

When selecting the operating system familiarity and popularity of the operating system not only be advantage to the developers, but also it helps end-users when using them.

**Performance and Resource Management**

The selected operating system has to be capable of handling available resource with the maximum usage to generated good output performance. Therefore Performance support and accurate resource management should be considered.

**The Final Choice**

By considering above mention criteria windows XP/vista/7 were selected as the best suited operating system for this software.

**Language Selection For Business Logic Components**

Proper development language is the essence of developing a successful product. Selection of implementation language is evaluated under a several feature. C#.NET,PHP, HTML, SQL SERVER, could be used develop a business logic components.

**Execution Spent and Efficiency**

Components are the essence of the proposed system. Should efficiently fetch the data from the database and analyses them and present it to user in appropriate manner.

**DEVELOPMENT TOOLS:**

Since product needs to be delivered in very limited time period, rapid development is essential.

**EASE DATABASE CONNECTIVITY**:

Proposed system involves high degree of database manipulation. Those databases should be carried out effective and speedily. Therefore the selected development language should support easy database connectivity and manipulation.

**DATABASE SELECTION:**

Data storing is the major issue we should be considered at the system development time, we have used MySql as the database software.

# FEASIBILITY REPORT

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

* Technical Feasibility
* Operation Feasibility
* Economical Feasibility

## Technical Feasibility

The technical issue usually raised during the feasibility stage of the investigation includes the following:

* Does the necessary technology exist to do what is suggested?
* Do the proposed equipments have the technical capacity to hold the data required to use the new system?
* Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
* Can the system be upgraded if developed?
* Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of ‘Secure Infrastructure Implementation System’. The current system developed is technically feasible. It is a web based user interface for audit workflow at NIC-CSD. Thus it provides an easy access to the users. The database’s purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security.

## Operational Feasibility

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization’s operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

* Is there sufficient support for the management from the users?
* Will the system be used and work properly if it is being developed and implemented?
* Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits.

The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

## Economic Feasibility

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC, There is nominal expenditure and economical feasibility for certain.

# HARDWARE AND SOFTWARE SPECIFICATION

## SOFTWARE REQUIREMENT

|  |  |
| --- | --- |
| FRONT END | HTML,CSS |
| BACK END | MySql |
| SERVER SIDE SCRIPTING | PHP |
| OPERATING SYSTEM | Windows xp or above |

## HARDWARE REQUIREMENT

|  |  |
| --- | --- |
| PROCESSOR | PENTIUM DUAL CORE OR ABOVE |
| RAM | MIN 512MB |
| KEYBOARD | STANDARD |
| HARD DISK | Min 40GB |

**SOFTWARE REQUIREMENT SPECIFICATION**

**Software Scope:**

**Features of Basic PHP**

PHP is a server –side Scripting language designed for web development but also used as a general purpose programming language. PHP is now installed 224 million websites and 2.1 million web server. PHP code is interpreted by a web server with PHP processor module which generated the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command – line interface capability and can be used in standalone graphical application.

• Reduce the time to create large websites.

• Create a customized user experience for visitors based on information that you have gathered from them.

• Open up thousands of possibilities for online tools. Check out PHP -Hot Scripts for examples of the great things that are possible with PHP.

• Allow creation of shopping carts for e-commerce websites.

The HTTP Authentication hooks in PHP are only available when it is running as an Apache module and is hence not available in the CGI version. In an Apache module PHP script, it is possible to use the **header()** function to send an "Authentication

Required" message to the client browser causing it to pop up a Username/Password input window. Once the user has filled in a username and a password, the URL containing the PHP script will be called again with the predefined variables PHP\_AUTH\_USER, PHP\_AUTH\_PW, and AUTH\_TYPE set to the user name, password and authentication type respectively.These predefined variables are found in the $\_SERVER and $HTTP\_SERVER\_VARS arrays. Only "Basic" authentication is supported. See the **header()** function for more information.

**Cookies**

PHP transparently supports HTTP cookies. Cookies are a mechanism for storing data in the remote browser and thus tracking or identifying return users. You can set cookies using the **setcookie()** function. Cookies are part of the HTTP header, so

**setcookie ()** must be called before any output is sent to the browser. This is the same limitation that **header()** has. You can use the output buffering functions to delay the script output until you have decided whether or not to set any cookies or sendany headers.Any cookies sent to you from the client will automatically be turned into a PHP variable just like GET and POST method data, depending on the register\_globals and variables \_order configuration variables. If you wish to assign multiple values to a single cookie, just add *[]* to the cookie name.

**POST method uploads**

PHP is capable of receiving file uploads from any RFC-1867 compliant browser (which includes Netscape Navigator 3 or later, Microsoft Internet Explorer 3 with a patch from Microsoft, or later without a patch). This feature lets people upload both text and binary files. With PHP's authentication and file manipulation functions, you have full control over who is allowed to upload and what is to be done with the file once it has been uploaded.

**Related Configurations Note:** See also the file\_uploads, upload\_max\_filesize, upload\_tmp\_dir, and post\_max\_size directives in php.ini

Note that PHP also supports PUT-method file uploads as used by Netscape Composer and W3C's Amaya clients. See the PUT Method Support for more details.

**PUT method support**

PUT method support has changed between PHP 3 and PHP 4. In PHP 4, one should use the standard input stream to read

the contents of an HTTP PUT.

**Connection handling**

Internally in PHP a connection status is maintained. There are 3 possible states:

• 0 - NORMAL

• 1 - ABORTED

• 2 - TIMEOUT

When a PHP script is running normally the NORMAL state, is active. If the remote client disconnects the ABORTED state flag is turned on. A remote client disconnect is usually caused by the user hitting his STOP button. If the PHP-imposed time limit (see **set\_time\_limit()**) is hit, the TIMEOUT state flag is turned on.

You can decide whether or not you want a client disconnect to cause your script to be aborted. Sometimes it is handy to always have your scripts run to completion even if there is no remote browser receiving the output. The default behaviour ishowever for your script to be aborted when the remote client disconnects. This behaviour can be set via the ignore\_user\_abort php.ini directive as well as through the corresponding "php\_valueignore\_user\_abort" Apache .conf directive or with the **ignore\_user\_abort()** function. If you do not tell PHP to ignore a user abort and the user aborts, your script will terminate. The one exception is if you have registered a shutdown function using **register\_shutdown\_function()**.

With a shutdown function, when the remote user hits his STOP button, the next time your script tries to output something, PHP will detect that the connection has been aborted and the shutdown function is called. This shutdown function will also get called at the end of your script terminating normally, so to do something different in case of a client disconnect you can

use the **connection aborted()** function. This function will return TRUE if the connection was aborted.

Your script can also be terminated by the built-in script timer. The default timeout is 30 seconds. It can be changed using the max execution\_time php.ini directive or the corresponding "php\_valuemax\_execution\_time" Apache .conf directive as well as with the **set\_time\_limit()** function. When the timer expires the script will be aborted and as with the above client disconnect case, if a shutdown function has been registered it will be called. Within this shutdown function you can check to see if a timeout caused the shutdown function to be called by calling the **connection\_timeout()** function. This function will return TRUE if a timeout caused the shutdown function to be called.

One thing to note is that both the ABORTED and the TIMEOUT states can be active at the same time. This is possible ifyou tell PHP to ignore user aborts. PHP will still note the fact that a user may have broken the connection, but the script will keep running. If it then hits the time limit it will be aborted and your shutdown function, if any, will be called. At this point you will find that **connection\_timeout()** and **connection\_aborted()** return TRUE. You can also check both states in a single call by using the **connection\_status()**. This function returns a bit field of the active states. So, if both states are active it would return 3, for example.

**Persistent Database Connections**

Persistent connections are SQL links that do not close when the execution of your script ends. When a persistent connectionis requested, PHP checks if there's already an identical persistent connection (that remained open from earlier) - and if it exists, it uses it. If it does not exist, it creates the link. An 'identical' connection is a connection that was opened to the same

host, with the same username and the same password (where applicable).

**Note:** There are other extensions that provide persistent connections, such as the IMAP extension.

The first method is to use PHP as a CGI "wrapper". When run this way, an instance of the PHP interpreter is created and destroyed for every page request (for a PHP page) to your web server. Because it is destroyed after every request, any resources that it acquires (such as a link to an SQL database server) are closed when it is destroyed. In this case, you do not gain anything from trying to use persistent connections -- they simply don't persist.

The second, and most popular, method is to run PHP as a module in a multiprocessor web server, which currently only includes Apache. A multiprocessor server typically has one process (the parent) which coordinates a set of processes (its children) who actually do the work of serving up web pages. When each request comes in from a client, it is handed off to one of the children that is not already serving another client. This means that when the same client makes a second request to the server, it may be serviced by a different child process than the first time. What a persistent connection does for you in this case it make it so each child process only needs to connect to your SQL server the first time that it serves a page that makes use of such a connection. When another page then requires a connection to the SQL server, it can reuse the connection that child established earlier.

The last method is to use PHP as a plug-in for a multithreaded web server. Currently PHP 4 has support for ISAPI, WSAPI,and NSAPI (on Windows), which all allow PHP to be used as a plug-in on multithreaded servers like Netscape Fast-track (iPlanet), Microsoft's Internet Information Server (IIS), and O'Reilly's Website Pro. The behavior is essentially the same as for the multiprocessor model described before. Note that SAPI support is not available in PHP 3.

Extremely simple -- efficiency. Persistent connections are good if the overhead to create a link to yourSQL server is high. Whether or not this overhead is really high depends on many factors. Like, what kind of database it is, whether or not it sits on the same computer on which your web server sits, how loaded the machine the SQL server sits on is and so forth. The bottom line is that if that connection overhead is high, persistent connections help you considerably. They cause the child process to simply connect only once for its entire lifespan, instead of every time it processes a page that requires connecting to the SQL server. This means that for every child that opened a persistent connection will have its own open persistent connection to the server. For example, if you had 20 different child processes that ran a script that made a persistent connection to your SQL server, you'd have 20 different connections to the SQL server, one from each child.

**Security and Safe Mode**

**Table 22.1. Security and Safe Mode Configuration Directives**

**Name Default Changeable**

safe\_mode"0"PHP\_INI\_SYSTEM

safe\_mode\_gid "0" PHP\_INI\_SYSTEM

safe\_mode\_include\_dir NULLPHP\_INI\_SYSTEM

safe\_mode\_exec\_dir ""PHP\_INI\_SYSTEM

safe\_mode\_allowed\_env\_vars PHP\_ PHP\_INI\_SYSTEM

safe\_mode\_protected\_env\_vars LD\_LIBRARY\_PATH PHP\_INI\_SYSTEM

open\_basedir NULL PHP\_INI\_SYSTEM

disable\_functions "" PHP\_INI\_SYSTEM

disable\_classes "" PHP\_INI\_SYSTEM

**safe\_modeBoolean**

Whether to enable PHP's safe mode. Read the Security and chapter for more information.

**safe\_mode\_gidBoolean**

By default, Safe Mode does a UID compare check when opening files. If you want to relax this to a GID compare, then

turn on safe\_mode\_gid. Whether to use UID (FALSE) or GID (TRUE) checking upon file access.

**safe\_mode\_include\_dirstring**

UID/GID checks are bypassed when including files from this directory and its subdirectories (directory must also be in include\_path or full path must including).

As of PHP 4.2.0, this directive can take a semi-colon separated path in a similar fashion to the include\_path directive, rather than just a single directory.

The restriction specified is actually a prefix, not a directory name. This means that "safe\_mode\_include\_dir = /dir/incl" also allows access to "/dir/include" and "/dir/incls" if they exist. When you want to restrict access to only the specifieddirectory, end with a slash. For example: "safe\_mode\_include\_dir = /dir/incl/"

**safe\_mode\_exec\_dirstring**

If PHP is used in safe mode, **system()** and the other functions executing system programs refuse to start programs that

are not in this directory.

**safe\_mode\_allowed\_env\_varsstring**

Setting certain environment variables may be a potential security breach. This directive contains a comma-delimited list

of prefixes. In Safe Mode, the user may only alter environment variables whose names begin with the prefixes supplied

here. By default, users will only be able to set environment variables that begin with PHP\_ (e.g. PHP\_FOO=BAR).

**Note:** If this directive is empty, PHP will let the user modify ANY environment variable!

**safe\_mode\_protected\_env\_varsstring**

This directive contains a comma-delimited list of environment variables that the end user won't be able to change using **putenv()**. These variables will be protected even if safe\_mode\_allowed\_env\_vars is set to allow to change them.

**open\_basedirstring**

Limit the files that can be opened by PHP to the specified directory-tree, including the file itself. This directive is *NOT* affected by whether Safe Mode is turned On or Off.

When a script tries to open a file with, for example, fopen or gzopen, the location of the file is checked. When the file is outside the specified directory-tree, PHP will refuse to open it. All symbolic links are resolved, so it's not possible to avoid this restriction with a symlink.The special value .indicates that the directory in which the script is stored will be used as base-directory. Under Windows, separate the directories with a semicolon. On all other systems, separate the directories with a colon. As an Apache module, open\_basedir paths from parent directories are now automatically inherited. The restriction specified with open\_basedir is actually a prefix, not a directory name. This means that "open\_basedir = /dir/incl" also allows access to "/dir/include" and "/dir/incls" if they exist. When you want to restrict access to only the specified directory, end with a slash. For example: "open\_basedir = /dir/incl/"

**Proposed System**

Proposed system has been developed under two tier technology with PHP as the front end and MY SQL as the Back end. The system was developed under windows 7 operating system.

System helped to analyze the present system in its full dimension. It recommended the need for a few systems that could handle the inefficiencies of the present system.

The proposed system support easy calculation and updating. It generates all the information about the feedback details. The main objective of the processed system is to overcoming the drawbacks in the manual system. It provides faster and updates information. The proposed system user is provided with the choice of menus, which are very user friendly and mechanical. Errors in data entry are minimized through validation check by interrupting with appropriate messages.

**FEATURES OF THE PROPOSED SYSTEM:**

1. Less manual interaction.
2. Faster information retrieval.
3. Integrated security.
4. Faster processing
5. Record report system.
6. Long term data storage.

**Non Functional Requirements and Constraints**

**1. USER FRIENDINESS:**

System is users friendly**.** Its functionality remains similar to the existing system functionality. It improves the user friendliness of the system.

**2. SECURITY:**

Unauthorized access of data should the access to the stored information. User levels are clearly defined and their security levels defined to secure the information.

**3. PORTABILITY:**

User can use the software on the system on the system with minimum hardware and software requirements. It can be used in different platform.

**4.EASE OF USE:**

Since the actual users of this system are not computer professional. It is kept simple and to understand. A system guide is provided with the system to use in any case difficulties.

**SYSTEM DEFINITION**

At system analysis level a use case diagram were drawn and the requirements are gathered. In the preliminary state of design phase,system was viewed in two layered structure, which is also named as vertical visualization. Then the system is further divided into separate logical and functional areas called subsystem, which gives the horizontal visualization of the system. Each subsystem will be further described using activity.

The main benefit of this technique is that it is extensible; enhancement to the system can be added where necessary because each subsystem is independent line of communication between subsystem must be formally define to ensure the integrity of data and the execution of operations each sub system can be tested independently, accessed the parallel implementation and testing of different subsystem.

**7.1 SYSTEM ARCHITECTURE**

High level view of the system can be shown as a two-tiered architecture. The presentation layer and the storage layer.

Layers are the two components, which will provide the conceptual view of the system in highly summarized manner. At this stage system should be looked in an implementation of view point; hence it is possible to view this in a more descriptive manner. As it is described in the below figure 5-1, clear separations between the layer can be identified. There by it

|  |
| --- |
| Provide an overall undertaking of the system  C:\Users\Raj Shrestha\AppData\Local\Microsoft\Windows\INetCache\Content.Word\digital-adult-at-computer.jpguser interface storage tier |

**ARCHITECTURE REPRESENTATION**

**Presentation Layer**

The application running in the presentation layer is responsible for presenting the user interface, that the user work with and for interacting with the backend storage.

**Business logic tier (application server)**

The logic tier pulled out from the presentation tier and, as its own layer; it controls an application’s functionality by performing details processing.

**Data access layer**

The data access layer contains the data access component, which is responsible for retrieving data from the data base as well as inserting, updating knows what database to use, how to connect to the database and which a stored procedures and view to execute to return the information. It has a basic class that all other classes in the data access component inherit.

**Architectural description**

|  |  |
| --- | --- |
| PRESENTATION LAYER HTML,PHP   |  | | --- | | PROJECT MANAGEMENT SYSTEM | |

|  |
| --- |
| DATA ACCESS LAYER |
| |  | | --- | | COMPANY DETAILS |  |  | | --- | | TEAMLEADER DETAILS |  |  | | --- | | EMPLOYEE DETAILS |   PROJECT DETAILS |

|  |
| --- |
| DATA BASE (SQL SERVER) |

**Module design**

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer’s goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word “Quality”. Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer’s view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

**DATA FLOW DIAGRAMS**

A data flow diagram is graphical tool used to describe and analyze movement of data through a system. These are the central tool and the basis from which the other components are developed. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow diagrams. The physical data flow diagrams show the actual implements and movement of data between people, departments and workstations. A full description of a system actually consists of a set of data flow diagrams. Using two familiar notations Yourdon, Gane and Sarson notation develops the data flow diagrams. Each component in a DFD is labeled with a descriptive name. Process is further identified with a number that will be used for identification purpose. The development of DFD’S is done in several levels. Each process in lower level diagrams can be broken down into a more detailed DFD in the next level. The lop-level diagram is often called context diagram. It consists a single process bit, which plays vital role in studying the current system. The process in the context level diagram is exploded into other process at the first level DFD.

The idea behind the explosion of a process into more process is that understanding at one level of detail is exploded into greater detail at the next level. This is done until further explosion is necessary and an adequate amount of detail is described for analyst to understand the process.

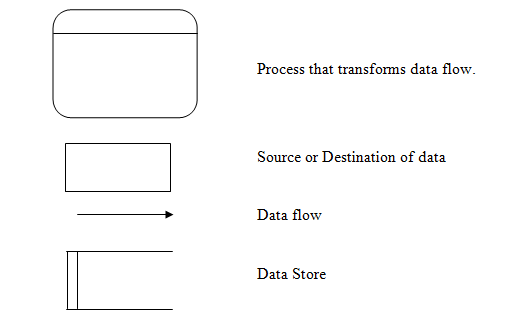
Larry Constantine first developed the DFD as a way of expressing system requirements in a graphical from, this lead to the modular design.

A DFD is also known as a “bubble Chart” has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is the starting point of the design to the lowest level of detail.

**DFD SYMBOLS:**

In the DFD, there are four symbols-

1. A square defines a source(originator) or destination of system data.
2. An arrow identifies data flow. It is the pipeline through which the information flows.
3. A circle or a bubble represents a process that transforms incoming data flow into outgoing data flows.
4. An open rectangle is a data store, data at rest or a temporary repository of data.



**CONSTRUCTING A DFD:**

Several rules of thumb are used in drawing DFD’S:

1. Process should be named and numbered for an easy reference. Each name should be representative of the process.
2. The direction of flow is from top to bottom and from left to right. Data traditionally flow from source to the destination although they may flow back to the source. One way to indicate this is to draw long flow line back to a source. An alternative way is to repeat the source symbol as a destination. Since it is used more than once in the DFD it is marked with a short diagonal.
3. When a process is exploded into lower level details, they are numbered.
4. The names of data stores and destinations are written in capital letters. Process and dataflow names have the first letter of each work capitalized

Questionnaires should contain all the data elements that flow in and out. Missing interfaces redundancies and like is then accounted for often through interviews.

**SAILENT FEATURES OF DFD’S**

1. The DFD shows flow of data, not of control loops and decision are controlled considerations do not appear on a DFD.
2. The DFD does not indicate the time factor involved in any process whether the dataflow take place daily, weekly, monthly or yearly.
3. The sequence of events is not brought out on the DFD.

**TYPES OF DATA FLOW DIAGRAMS**

1. Current Physical
2. Current Logical
3. New Logical
4. New Physical

**CURRENT PHYSICAL:**

In Current Physical DFD process label include the name of people or their positions or the names of computer systems that might provide some of the overall system-processing label includes an identification of the technology used to process the data. Similarly data flows and data stores are often labels with the names of the actual physical media on which data are stored such as file folders, computer files, business forms or computer tapes.

**CURRENT LOGICAL**

The physical aspects at the system are removed as mush as possible so that the current system is reduced to its essence to the data and the processors that transform them regardless of actual physical form.

**NEW LOGICAL**

This is exactly like a current logical model if the user were completely happy with he user were completely happy with the functionality of the current system but had problems with how it was implemented typically through the new logical model will differ from current logical model while having additional functions, absolute function removal and inefficient flows recognized.

**NEW PHYSICAL**

The new physical represents only the physical implementation of the new system.

**RULES GOVERNING THE DFD’S PROCESS**

1. No process can have only outputs.
2. No process can have only inputs. If an object has only inputs than it must be a sink.
3. A process has a verb phrase label.

**DATA STORE**

1. Data cannot move directly from one data store to another data store, a process must move data.
2. Data cannot move directly from an outside source to a data store, a process, which receives, must move data from the source and place the data into data store
3. A data store has a noun phrase label.

**SOURCE OR SINK**

The origin and /or destination of data.

1. Data cannot move direly from a source to sink it must be moved by a process
2. A source and /or sink has a noun phrase land

**DATA FLOW**

1. A Data Flow has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. The later is usually indicated however by two separate arrows since these happen at different type.
2. A join in DFD means that exactly the same data comes from any of two or more different processes data store or sink to a common location.
3. A data flow cannot go directly back to the same process it leads. There must be atleast one other process that handles the data flow produce some other data flow returns the original data into the beginning process.
4. A Data flow to a data store means update (delete or change).
5. A data Flow from a data store means retrieve or use.

**Level 0 Diagram:**