# 1. Introduction

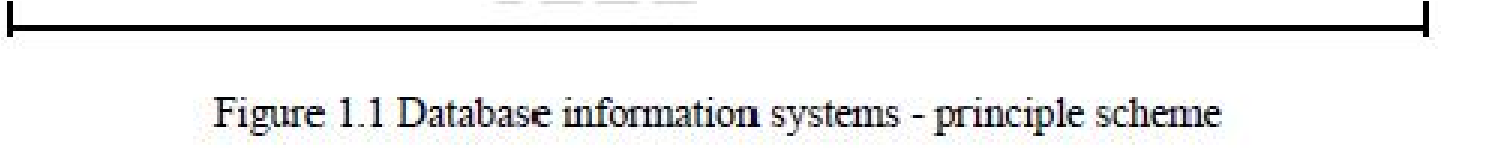
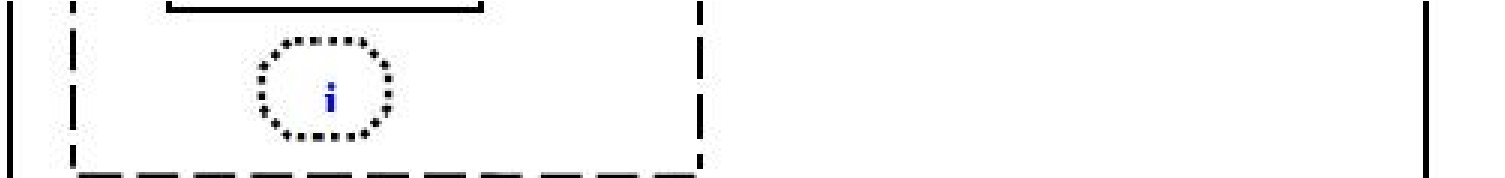
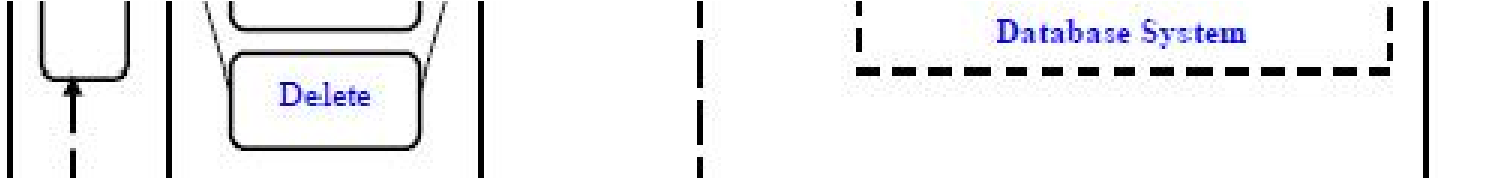
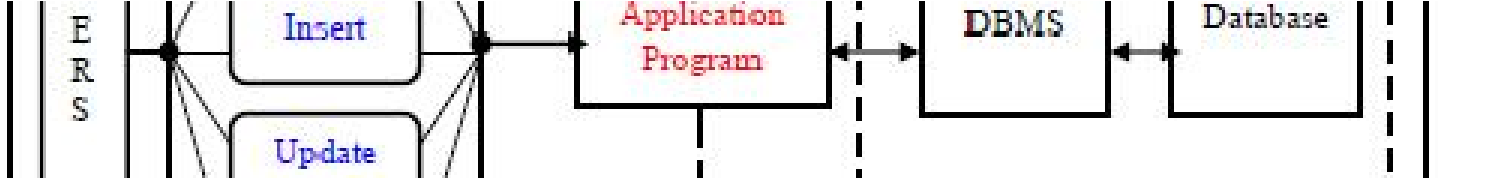
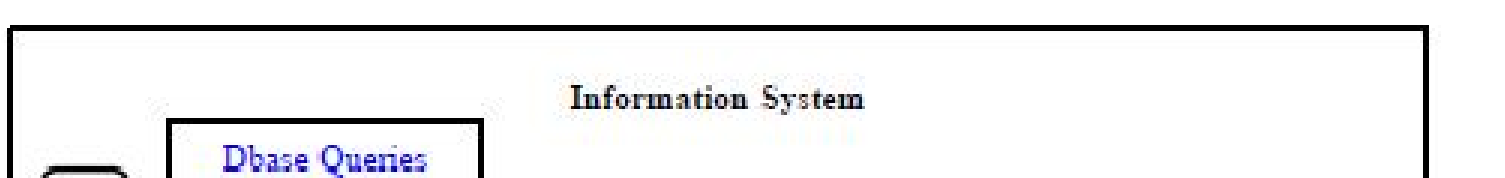
This chapter gives a brief theoretical preview upon the database information systems and goes through the essence of the problem that should be resolved.

**1.1 Background**

Most of the contemporary Information systems are based on the Database technology as a collection of logically related data, and DBMS as a software system allowing the users to define, create, maintain and control access to the database.

The process of constructing such kind of systems is not so simple. It involves a mutual development of application program and database. The application program is actually the bridge between the users and the database, where the data is stored. Thus, the well-developed application program and database are very important for the reliability, flexibility and functionality of the system. The so defined systems differentiate to each other and their development comprises a great variety of tasks to be resolved and implemented.

The basic idea can be depicted on Figure 1.1 below:



information to users in an organization (for instance), as for the purposes of Data Information system suggests a computer technology to be used in order to provide transformation into useful information; computer hardware and software are designed and used [2]. A particular case is the Human Resources

Information System development. This kind of systems are responsible for storing data of the staff within an organization and generating reports upon request Such kind of system could be integrated with other Information systems or modules: Accounting Information System (AIS) – designed to transform financial data into information, or Management Information System (MIS) that provides decision oriented information to managers, and so on. Organizations depend on Information Systems in order to stay competitive. Productivity, which is crucial to staying competitive, can be increased through better Information Systems.” [2].

**1.2 Problem Statement**

This report’s documentation goes through the whole process of both application program and database development. It also comprises the development tools have been utilized for these purposes.

**1.3 Problem Discussion**

This system should consist of an application program, on one hand, and a database (repository of data) on the other. The program should perform the basic operations upon the database as retrieving, inserting, updating and deleting data. Any additional functionality is a goal of a further module development.

It is a kind of strategy to start the development from designing and constructing the database, as this structure will determine the further structure of the application program. The logical database model (tables, their content and the relationships between them) should respond to the given task and cover the basic requirements. The Interface of the program should be user-friendly, and the program should be as easy for use as it is possible. Both controls and forms should logically and functionally be related within the program and fully respond to the structure of the database. Another problem is establishing the connections with the database, every time, when a query is needed to be performed upon it. Exception handling should also be taken into an account during the system’s development due to eventual exceptions that may occur.

# 2. Proposed System

The Proposed Institute Information System provides all Institute documents online which can be accessed by all the departments in the Institute.

This project is based on carrying out various tasks which goes under college management. It will control all activities for a particular college.

As most of the work is done manually or it is based on paper work such as class attendance, internal assessment marks, fee dues, etc. These all process takes time. If we include all the work which will be based only on system, then it can reduce time and work.

Under this system it includes all the departments which comes a particular college. It will have billing system for the admission section, calculation of attendance, calculation of the average Internal marks of the students.

# 3. SYSTEM REQUIREMENTS

# 3.1 FUNTIONAL REQUIREMENTS

**HARDWARE REQUIREMENTS:**

PROCESSOR : Intel Pentium 4 or above

RAM : 512 MB Min

HARD DISK : 40 GB min

**SOFTWARE REQUIREMENTS**:

OPERATING SYSTEM : Windows 7 or higher

LANGUAGE(Server Side) : PHP

WEB TECHNOLOGY : HTML, CSS, JAVA SCRIPT

WEB SERVER : Apache

BACK END : MySQL

**3.2 NONFUNCTIONAL REQUIREMENTS :**

The non-functional requirement arise through user needs, because of budget constraints, because of organizational policies, with other software or hardware system. The non-functional requirements come from required characteristics of the software, the organization developing the software or the external sources. Following are the non-functional requirements

1. **Performance**: It step by step procedure and Response time is high.
2. **Scalability:** The application is a high performance, energy efficient, reliable and secure infrastructure.

**3. Maintainability:** Very low maintenance for this application as is built only for a particular stores. It works very faster with huge storage.

**4. Availability :** It supports all types of browsers and as it is online application, it is available for users anywhere and at any time

# System Design:-

**Data flow diagram:**

Office Info

Database

Student Info

Teaching Info

Non Teaching Info

The working procedure of the project is as follows:

1. The student must take admission in the Office Section. By entering all his details required by the Institution.
2. Once the student is admitted into the college. The Student information is once required to fill the details and then can store all his Academic details. Such as attendance, internal details , etc.
3. The System can also store the teaching details such as Name, Qualification, Designation, etc.
4. The system can store the non-teaching details such as name and work assigned.

**4.1 Schema Diagram:**

Office-Info

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Roll No | USN | Total fees | Fees paid | Pending fees |

Student-Info

|  |  |  |  |
| --- | --- | --- | --- |
| Name | USN | Roll No | SEM |

Attendance Details

|  |  |  |  |
| --- | --- | --- | --- |
| USN | Total classes | Classes Attended | Total Attendance |

Internal Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| USN | Subject | IA-2 | IA-3 | IA-3 | Average IA |

Teaching Info

|  |  |  |  |
| --- | --- | --- | --- |
| Name | E-mail | Qualification | Designation |

Non Teaching

|  |  |
| --- | --- |
| Name | Work Assigned |

**4.2 Entity Relationship Diagram:**

1 N

Admits in

Student

Office

1

Academics

1 N N

IA Marks

Attendance

Staff

N N

Non-Teaching

Teaching

**4.3 Table Description :**

**Office Info:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **size** | **Constraints** | **Description** |
| name | varchar | 100 | Null | Name of the student |
| usn | varchar | 10 | Null | USN of the student |
| roll\_no | integer | 10 | Null | Roll No student |
| total\_fees | integer | 100 | Null | Total fees paid |
| fees\_paid | integer | 100 | Null | Fees paid |
| Pending\_fees | integer | 100 | Null | Pending fees |

**Student Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **size** | **Constraints** | **Description** |
| name | varchar | 11 | Null | Name of the student |
| usn | varchar | 100 | Null | USN of the student |
| Roll no | integer | 100 | Null | Roll No of the student |
| sem | integer | 11 | Null | Semester |

**Teaching:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **type** | **size** | **Constraints** | **Description** |
| Name | varchar | 11 | Null | Name of the teacher |
| Email | varchar | 100 | Null | E-mail of the teacher |
| Qualification | varchar | 100 | Null | Qualification of the teacher |
| Designation | varchar | 11 | Null | Desination of the teacher |

**Non Teaching:**

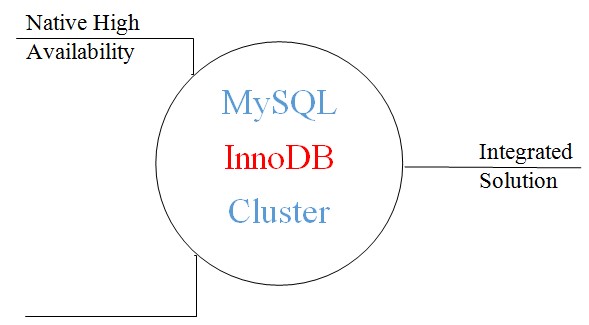
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **type** | **size** | **Constraints** | **Description** |
| Name | varchar | 100 | Null | Name |
| Works Assigned | varchar | 100 | Null | Works Assigned |

# 5. DETAILS OF TECHNOLOGIES AND LANGUAGES USED

**5.1 APACHE WEB SERVER:**

Apache HTTP Server, colloquially called Apache, is free and open-source cross-platform web server software, released under the terms of Apache License 2.0. ... Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled.Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. The Apache HTTP Server is cross-platform; as of 1 June 2017 92% of Apache HTTPS Server copies run on Linux distributions. Version 2.0 improved support for non-Unix operating systems such as Windows and OS/2.Old versions of Apache were ported to run on OpenVMS and NetWare. Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled. Apache played a key role in the initial growth of the World Wide Web, quickly overtaking NCSA HTTPd as the dominant HTTP server, and has remained most popular since April 1996. In 2009, it became the first web server software to serve more than 100 million websites. As of July 2016 was estimated to serve 46% of all active websites and 43% of the top million websites. Instead of implementing a single architecture, Apache provides a variety of Multi Processing Modules (MPMs), which allow Apache to run in a process-based, hybrid (process and thread) or event-hybrid mode, to better match the demands of each particular infrastructure. This implies that the choice of correct MPM and the correct configuration is important. Where compromises in performance need to be made, the design of Apache is to reduce latency and increase throughput, relative to simply handling more requests, thus ensuring consistent and reliable processing of requests within reasonable time-frames. For delivery of static pages, Apache 2.2 series was considered significantly slower than nginx and varnish. To address this issue, the Apache developers created the Event MPM, which mixes the use of several processes and several threads per process in an asynchronous event-based loop. This architecture, and the way it was implemented in the Apache 2.4 series, provides for performance equivalent or slightly better than event-based web servers, as is cited by Jim Jagielski and other independent sources. However, some independent, but significantly outdated, benchmarks show that it still is half as fast as nginx.

**5.2 MySQL:**



**5.3Easy To Use:**

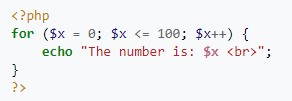
MySQL is an open-source relational database management system(RDBMS). The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality. MySQL is a central component of the LAMP open-source web application software stack. LAMP is an acronym for "Linux, Apache,

MySQL, Perl/PHP/Python". Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google, Face book, Twitter, Flickr, and YouTube. MySQL is written in C and C++. Its SQL parser is written in yacc, but it uses a home brewed lexical analyzer. MySQL works on many system platforms , including AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, macOS, Microsoft, Windows, NetBSD, Novell Netware, OpenBSD, Open Solaris, OS/2 Wrap, QNX, Oracle Solaris, Symbian, Sun OS, SCO OpenServer, SCO UnixWare, Sanos, Tru64. A port of MySQL to OpenVMS also exists. MySQL has received positive reviews, and reviewers noticed it "performs extremely well in the average case" and that the "developer interfaces are there, and the documentation is very, very good". It has also been tested to be a "fast, stable and true multi-user, multi-threaded sql database server" MySQL can be built and installed manually from source code, but it is more commonly installed from a binary package unless special customizations are required. On most Linux distributions, the package management system can download and install MySQL with minimal effort, though further configuration is often required to adjust security and optimization settings. Though MySQL began as a low-end alternative to more powerful proprietary databases, it has gradually evolved to support higher-scale needs as well. It is still most commonly used in small to medium scale single-server deployments, either as a component in a LAMP-based web application or as a standalone database server. Much of MySQL's appeal originates in its relative simplicity and ease of use, which is enabled by an ecosystem of open source tools such as phpMyAdmin. In the medium range, MySQL can be scaled by deploying it on more powerful hardware, such as a multi-processor server with gigabytes of memory.

**5.4 PHP:**

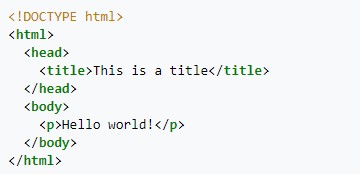
Generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP dynamic web page content or dynamic images used on websites or elsewhere.PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Preprocessor.PHP code may be embedded into HTML or HTML5 mark-up, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server software combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications. The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge The PHP interpreter only executes PHP code within its delimiters. Anything outside its delimiters is not processed by PHP, although non-PHP text is still subject to control structures described in PHP code. The most common delimiters are <?php to open and ?> to close PHP sections. The shortened form <? also exists. This short delimiter makes script files less portable, since support for them can be disabled in the local PHP configuration and it is therefore discouraged. The first form of delimiters, <?php and ?>, in XHTML and other XML documents, creates correctly formed XML processing instructions. This means that the resulting mixture of PHP code and other mark-up in the server-side file is itself well formed XML. Variables are prefixed with a dollar symbol, and a type does not need to be specified in advance. PHP 5 introduced type hinting that allows functions to force their parameters to be objects of a specific class, arrays, interfaces or callback functions. However, before PHP 7.0, type hints could not be used with scalar types such as integer or string.[53] Unlike function and class names, variable names are case sensitive. Both double-quoted ("") and heredoc strings provide the ability to interpolate a variable's value into the string.[96]PHP treats newlines as whitespace in the manner of a free-form language, and statements are terminated by a semicolon. PHP has three types of comment syntax: /\* \*/ marks block and inline comments; // as well as # are used for one-line 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 the C style syntax. if conditions, for and while loops, and function returns are similar in syntax to languages such as C, C++, C#, Java and Perl.

The following is an example of PHP for loop:



**5.5 HTML:**

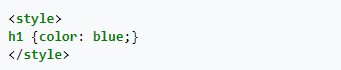
Hypertext Mark-up Language (HTML) is the standard mark-up language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as forms, may be embedded into the rendered page. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img>and<input >introduce content into the page directly. Others such as<p> ... </p>surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page. HTML can embed programs written in a scripting language such as JavaScript which affect the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML. The following is an example of the classic "Hello, World!" program, a common test employed for comparing programming languages, scripting languages and mark-up languages.



**5.6 CASCADING STYLE SHEET:**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a mark-up language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging web pages, user interfaces for web applications, and user interfaces for many mobile applications. CSS is designed primarily to enable the separation of presentation and content, including aspects such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .Css file, and reduce complexity and repetition in the structural content. Separation of formatting and content makes it possible to present the same mark-up page in different styles for different rendering methods, such as on-screen, in print, by voice, and on Braille-based tactile devices. It can also display the web page differently depending on the screen size or viewing device. Readers can also specify a different style sheet, such as a CSS file stored on their own computer, to override the one the author specified. Changes to the graphic design of a document (or hundreds of documents) can be applied quickly and easily, by editing a few lines in the CSS file they use, rather than by changing mark-up in the documents. The CSS specification describes a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities are calculated and assigned to rules, so that the results are predictable.

The following example shows the style element that gives red colour to fonts:



**5.7 JAVASCRIPT:**JavaScript often abbreviated as JS, is a high-level, dynamic, weakly typed, prototype based, multi-paradigm, and interpreted programming language. Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Web content production. It is used to make web pages interactive and provide online programs, including video games. The majority of websites employ it, and all modern web browsers support it without the need for plug-ins by means of a built-in JavaScript engine. Each of the many JavaScript engines represent a different implementation of JavaScript, all based on the ECMA Script specification, with some engines not supporting the spec fully, and with many engines supporting additional features beyond ECMA. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has an API for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded. Initially only implemented client side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets. Although there are strong outward similarities between JavaScript and Java, including language name, syntax, arespective standard libraries, the two languages are distinct and differ greatly in design; JavaScript was influenced by programming languages such as Self and Scheme.

# CONCLUSION

The project “INSTITUTE INFORMATION SYSTEM” manages all the information of the institution of all the departments such as Office-Info, Student-Info, Teaching-Info, Non-Teaching-Info. In the office dept the project stores all the information about the student information and auto calculates the pending fees from the total fees and the fees paid. In the student Info it maintains all the student information and the attendance details, IA details and auto calculates the average of all the three internals. In teaching information it maintains the teaching details and in non-teaching its maintains the non-teaching information.

**Future Enhancement:**

Even though the project was completed to the original requirements, there are a few features that we could add or improve on, if time allowed. The file system implementation should be revised. we could look at other DBMS file systems to get a better idea on how to design a more efficient and space conserving system.

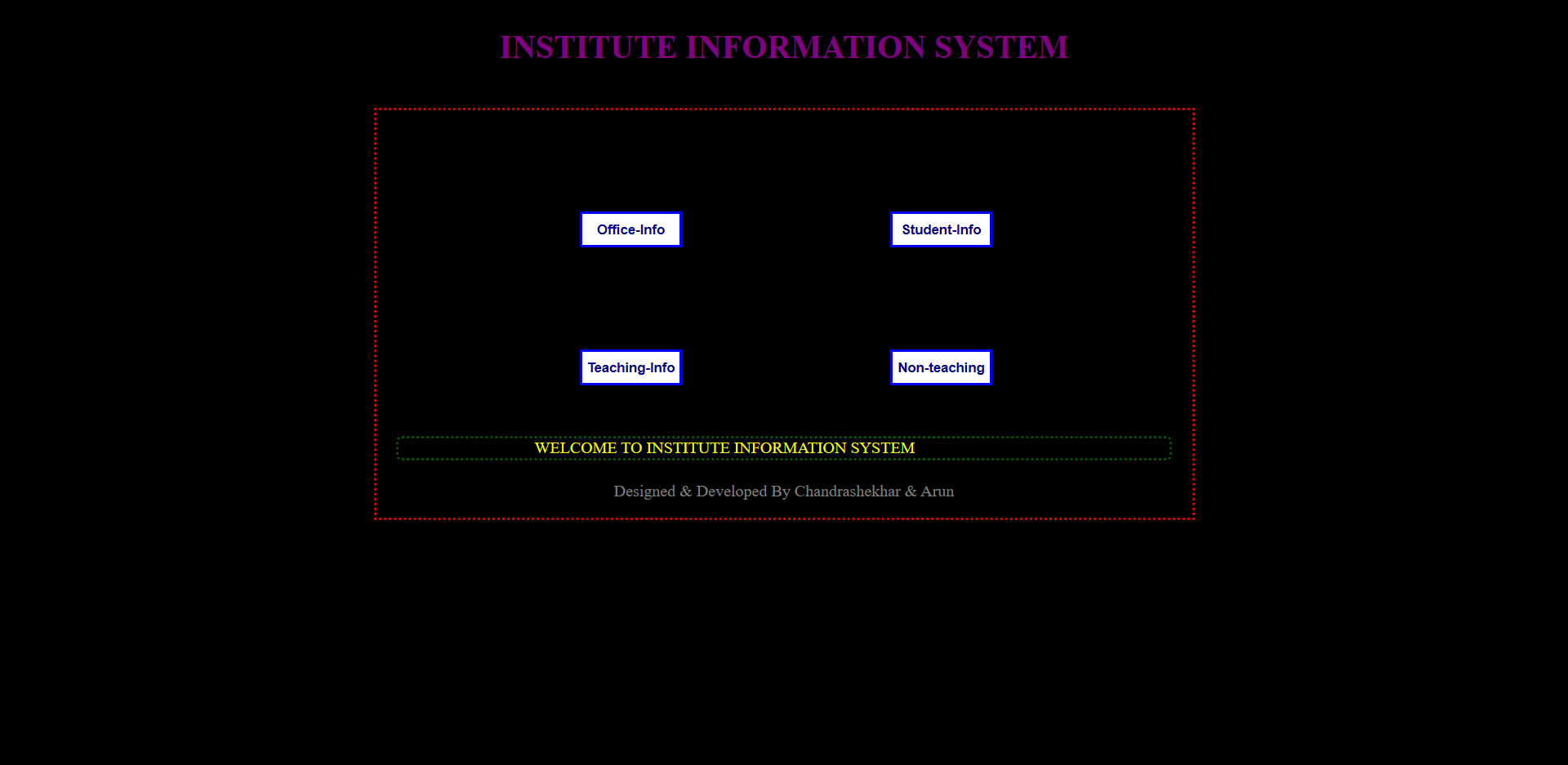
Security is the most important feature of the system. In this some security measures are taken to avoid the mishandling of the database. So, we can add more powerful security to the system.

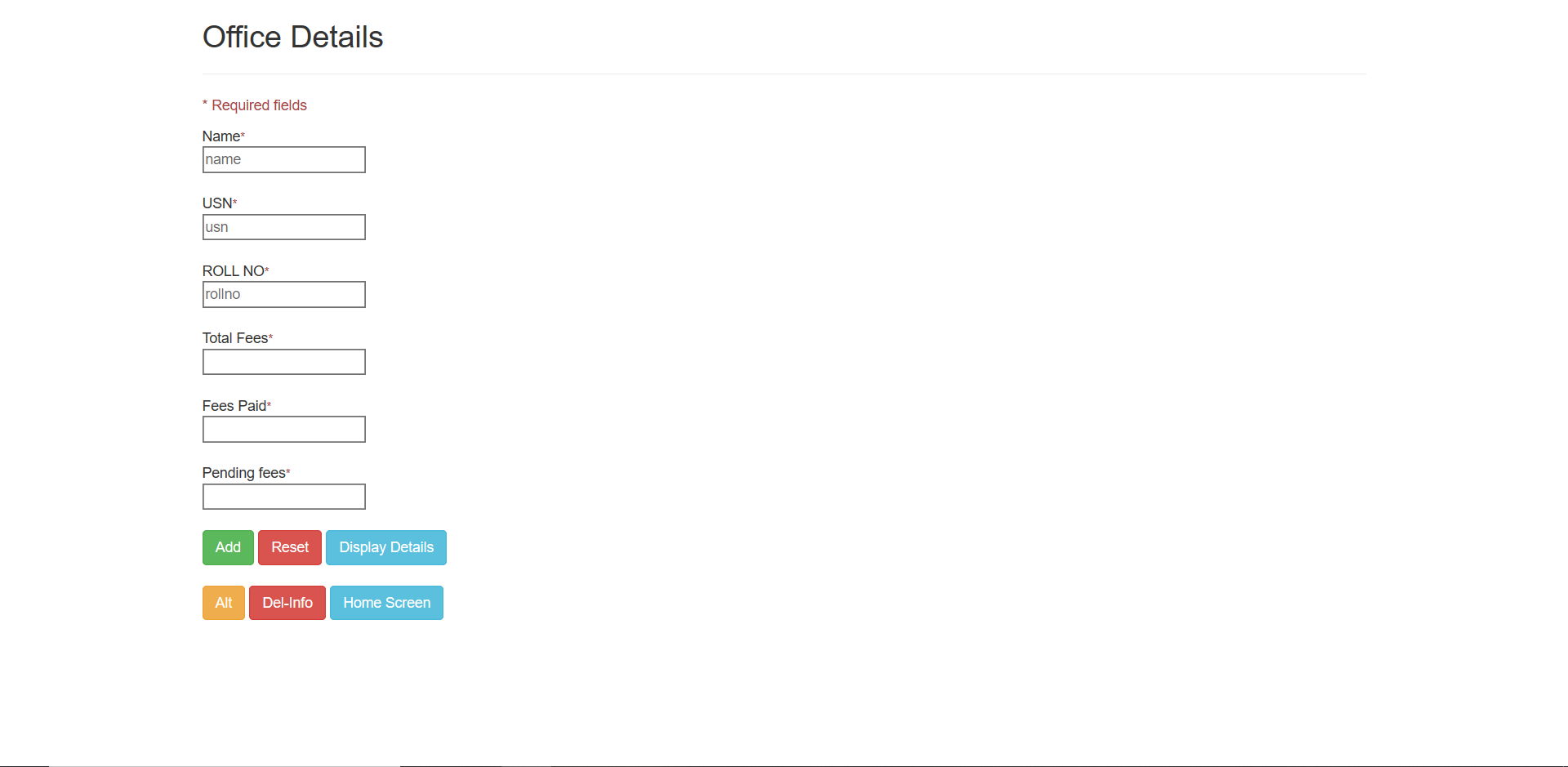
**LIMITATIONS:**

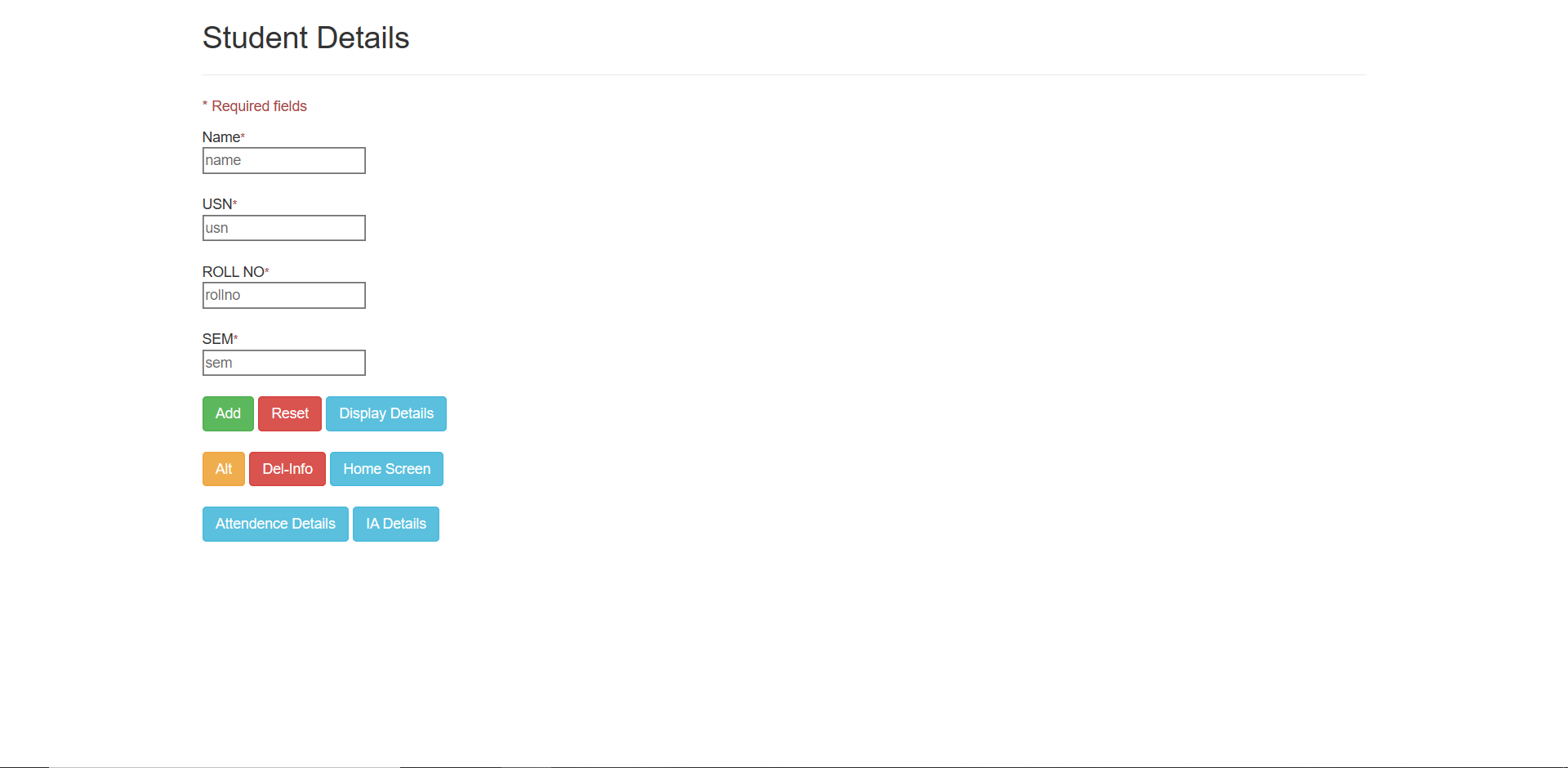
1.Some security Issues.

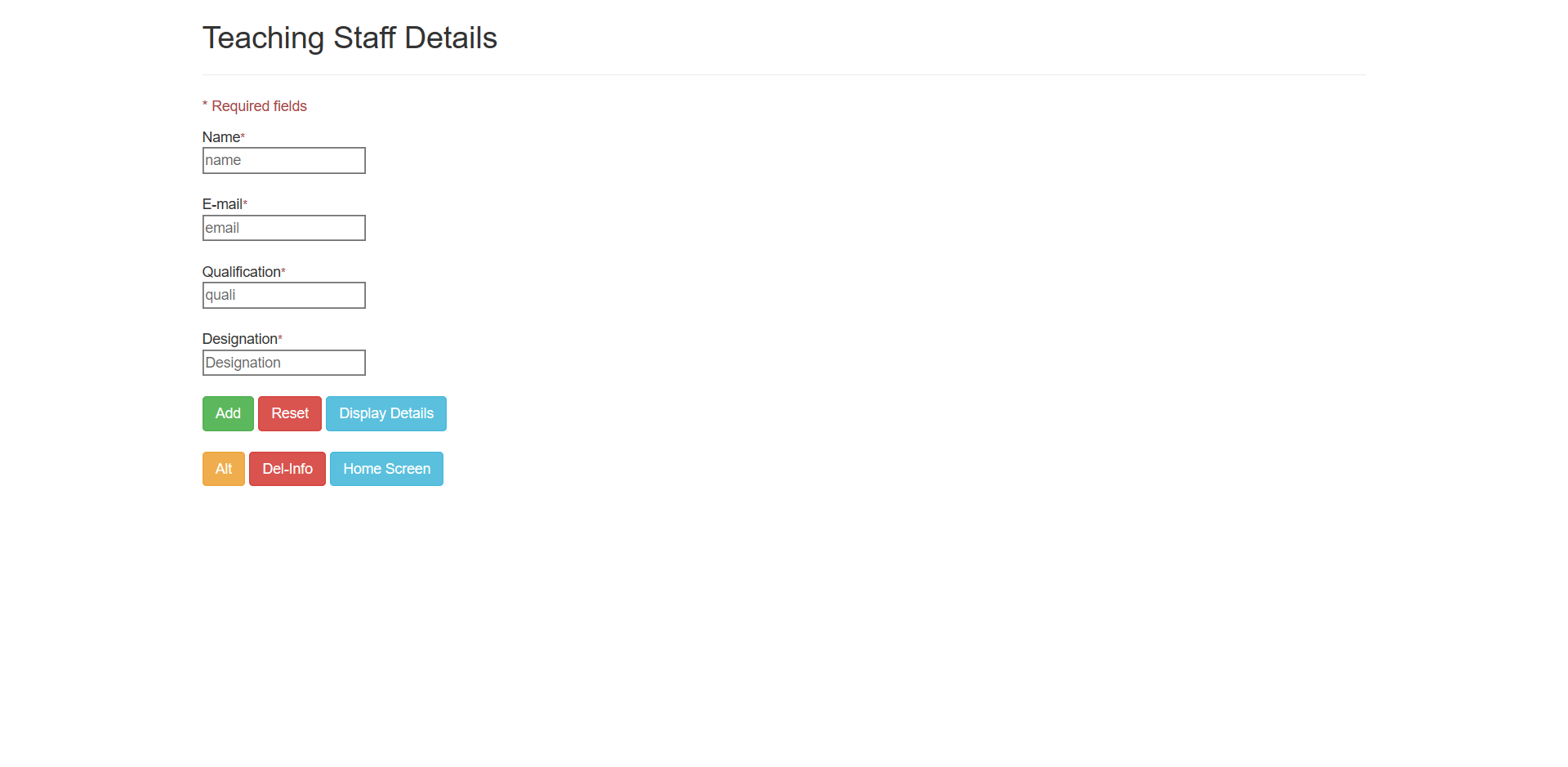
2.Data analysing.

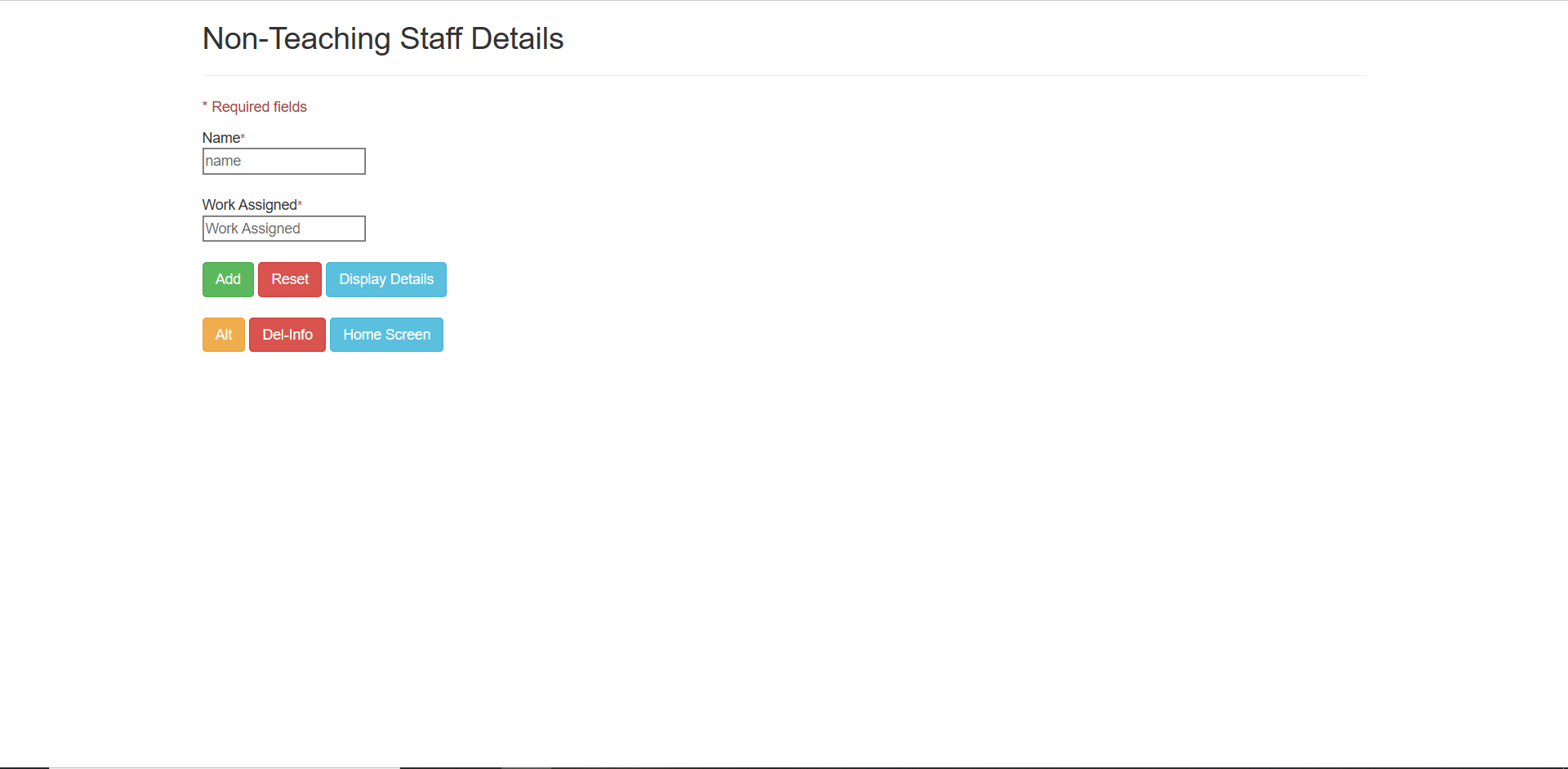
**7. SNAPSHOTS :**











# REFERENCES

**BOOKS REFERRED**

* Programming the World Wide Web – Robert W. Sebesta, 4th Edition, Pearson Education, 2008.
* PHP and MySQL Web Development by Luke Welling & Laura Thompson, Sam Publishing, ISBN 0-672-32525-X, First print 2003, Pages: 912.
* PHP – Writing PHP scripts & using Variables R. Berdan.
* HTML the definite guide – OREILLY, By Chuck Musciano & Bill Kennedy; ISBN 1-56592492-4, Third Edition, August 1998, Pages: 576.

**WEB LINK**

1. http://php-guide-lines.blogspot.in/
2. http://www.codingforums.com/
3. http://www.php.happycodings.com/
4. http://www.programmingfacts.com/
5. http://911-need-code-help.blogspot.in/
6. http://en.wikipedia.org/wiki/
7. http://www.dreamincode.net
8. http://www.a1vbcode.com