

Table of Contents

1. Introduction	1
1.1 Objective	1
1.2 Users / Actors of System	1
1.3 Admin Features (privileges)	1
1.4 Student Features (Privileges)	2
2. About Front End	3
2.1 Architecture and Concepts	4
2.2 Software & Tools Used:	5
2.3 Reasons for using PHP	6
2.4 Basic PHP Syntax	7
2.5 HTML	8
2.6 HTML 5.....	9
2.7 CSS	9
2.8 JAVASCRIPT	11
3. About Back End.....	12
3.1 MySQL's Logical Architecture	12
3.2 Software's and tools used.....	14
3.3 PHP Main Features of MySQL.....	16
3.4 XAMPP	16
4. Software Requirement	18
5. Hardware Requirement	19
6. Software Development Process	20
6.1 SDLC	20
6.2 Requirement validation	20
6.3 Constraints and Limitations.....	22
7. Output	23
7.1 Main Dashboard.....	23
7.2 Registration Page.....	23
7.3 Student Login	24
7.4 Student Dashboard.....	24
7.5 After Search	25
7.6 Print Marksheet.....	25
7.7 Student Query.....	26
7.8 Admin Login	26
7.9 Admin Dashboard.....	27
7.10 Showing Student Query	27
8. Bibliography	28

INTRODUCTION

This project is useful for students and institutions for getting the results in simple manner. By a result analyser with subject status and marks is an application tool for displaying the results in secure way. The system is intended for the student. And the privileges that are provided to student are to read and execute his/her result by providing user name and password for secure login and in case of new student the registration is available.

The whole result analyser will be under the control of the administrator and the admin as the full privileges to read, write and execute the result.

And admin gives the privileges to the student to access the result. The student can share or download his / her result.

1.1 Objective

The main **objective** of the project is to provide the examination result to the student in a simple way.

1.2 Users / Actors of System

- Admin
- Student

1.3 Admin Features (privileges)

- He/she is a super user who enjoys all the privileges.
- Add Update Display Delete Semesters.
- Add Update Display Delete Subjects and their Full Marks.
- Create/Approve/student account.
- View Results of each student and their Percentage.
- Download Result Report as PDF format
- Update Records

1.4 Student Features (Privileges)

- He/she needs to create an account.
- Can view their Semester Results for All Subjects from anywhere around the globe.
- Generate/Download Report of Results and Percentage in PDF Format
- Generate/Download Report of all Results in PDF Format

ABOUT FRONT END

The front end is an interface between the user and the back end. The front and back ends may be distributed amongst one or more systems.

In network computing, front end can refer to any hardware that optimizes or protects network traffic. It is called application front-end hardware because it is placed on the network's outward-facing front end or boundary. Network traffic passes through the front-end hardware before entering the network.

In compilers, the front end translates a computer programming source code into an intermediate representation, and the back end works with the intermediate representation to produce code in a computer output language. The back end usually optimizes to produce code that runs faster. The front-end/back-end distinction can separate the parser section that deals with source code and the back end that generates code and optimizes.

These days, front-end development refers to the part of the web users interact with. In the past, web development consisted of people who worked with Photoshop and those who could code HTML and CSS. Now, developers need a handle of programs like Photoshop and be able to code not only in HTML and CSS, but also JavaScript or jQuery, which is a compiled library of JavaScript.

Most of everything you see on any website is a mixture of HTML, CSS, and JavaScript, which are all controlled by the browser. For example, if you're using Google Chrome or Firefox, the browser is what translates all of the code in a manner for you to see and with which to interact, such as fonts, colors, drop-down menus, sliders, forms, etc. In order for all of this to work, though, there has to be something to support the front-end; this is where the backend comes into play.

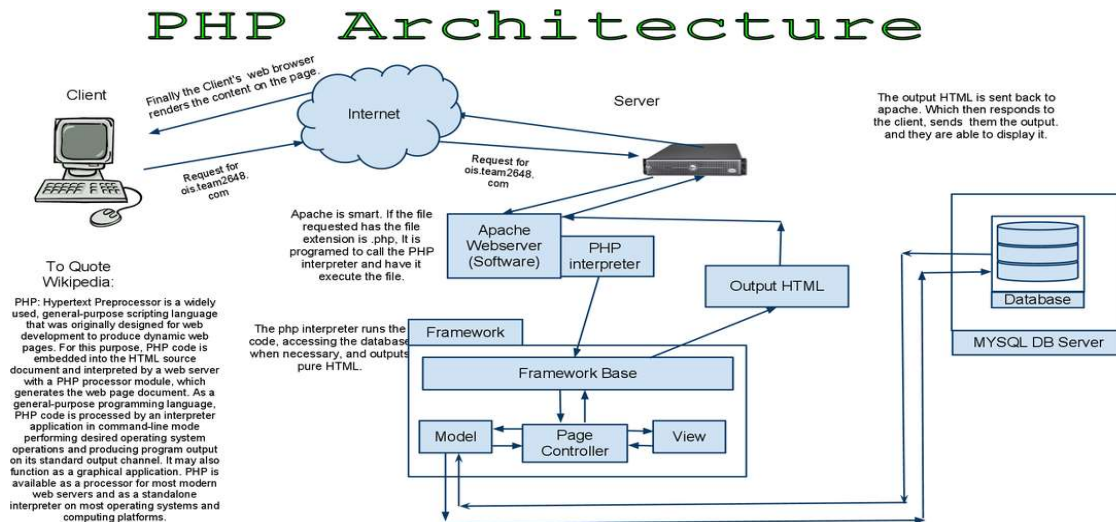


Fig. 2.1

2.1 Architecture and Concepts

The query cache plugin is implemented as a PHP extension. It is written in C and operates under the hood of PHP. During the start up of the PHP interpreter, it gets registered as a `mysqlnd` plugin to replace selected `mysqlnd` C methods. Hereby, it can change the behaviour of any PHP MySQL extension (`mysqli`, `PDO_MYSQL`, `mysql`) compiled to use the `mysqlnd` library without changing the extensions API. This makes the plugin compatible with each and every PHP MySQL application. Because existing APIs are not changed, it is almost transparent to use. Please, see the `mysqlnd` plugin API description for a discussion of the advantages of the plugin architecture and a comparison with proxy based solutions.

Various storage handlers are supported to offer different scopes for cache entries. Different scopes allow for different degrees in sharing cache entries among clients.

- a) default (built-in): process memory, scope: process, one or more web requests depending on PHP deployment model used
- b) APC: shared memory, scope: single server, multiple web requests
- c) SQLite: memory or file, scope: single server, multiple web requests
- d) MEMCACHE: main memory, scope: single or multiple server, multiple web requests

- e) user (built-in): user-defined - any, scope: user-defined - any

Support for the APC, SQLite and MEMCACHE storage handler has to be enabled at compile time. The default and user handler are built-in. It is possible to switch between compiled-in storage handlers on a per query basis at run time. However, it is recommended to pick one storage handler and use it for all cache entries.

Built-in slam defence to avoid overloading:

To avoid overload situations the cache plugin has a built-in slam defence mechanism. If a popular cache entries expires many clients using the cache entries will try to refresh the cache entry. For the duration of the refresh many clients may access the database server concurrently. In the worst case, the database server becomes overloaded and it takes more and more time to refresh the cache entry, which in turn lets more and more clients try to refresh the cache entry. To prevent this from happening the plugin has a slam defense mechanism. If slam defines is enabled and the plugin detects an expired cache entry it extends the life time of the cache entry before it refreshes the cache entry. This way other concurrent accesses to the expired cache entry are still served from the cache for a certain time. The other concurrent accesses to not trigger a concurrent refresh. Ideally, the cache entry gets refreshed by the client which extended the cache entries lifespan before other clients try to refresh the cache and potentially cause an overload situation.

Unique approach to caching:

PECL/mysqld_qc has a unique approach to caching result sets that is superior to application based cache solutions. Application based solutions first fetch a result set into PHP variables. Then, the PHP variables are serialized for storage in a persistent cache, and then unsterilized when fetching. The mysqlnd query cache stores the raw wire protocol data sent from MySQL to PHP in its cache and replays it, if still valid, on a cache hit. This way, it saves an extra serialization step for a cache put that all application based solutions have to do. It can store the raw wire protocol data in the cache without having to serialize into a PHP variable first and desteralizing the PHP variable for storing in the cache.

2.2 Software & Tools Used:

1. PHP:-PHP is now officially known as “**PHP: Hypertext Pre-processor**”. It is a server-side scripting language usually written in an HTML context. Unlike an ordinary HTML page,

a PHP script is not sent directly to a client by the server; instead, it is parsed by the PHP binary or module, which is server-side installed. HTML elements in the script are left alone, but PHP code is interpreted and executed. PHP code in a script can query databases, create images, read and write files, talk to remote servers – the possibilities are endless. The output from PHP code is combined with the HTML in the script and the result sent to the user's web-browser, therefore it can never tell the user whether the web-server uses PHP or not, because the entire browser sees is HTML.

PHP's support for Apache and MySQL further increases its popularity. Apache is now the most-used web-server in the world, and PHP can be compiled as an Apache module. MySQL is a powerful free SQL database, and PHP provides a comprehensive set of functions for working with it. The combination of Apache, MySQL and PHP is all but unbeatable. That doesn't mean that PHP cannot work in other environments or with other tools. In fact, PHP supports an extensive list of databases and web-servers. While in the mid-1990s it was ok to build sites, even relatively large sites, with hundreds of individual hard-coded HTML pages, today's webmasters are making the most of the power of databases to manage their content more effectively and to personalize their sites according to individual user preferences.

2.3 Reasons for using PHP

There are some indisputable great reasons to work with PHP. As an open source product, PHP is well supported by a talented production team and a committed user community. Furthermore, PHP can be run on all the major operating systems with most servers.

Learning PHP is easy: Basic is easy any interpreted language should be easy to learn. Since you are isolated from the system (no pointers to use, no memory to allocate). The other advantage that all modern interpreted languages share is good associative array constructs.

Its Performance: While we can build an application that serves millions of pages a day on a server, when we really look at the performance of the language it sucks. We are still orders of magnitude from real performance. Not only that, but since PHP is designed around a single process model our ability to share data structures or connection pool resources is left to native code libraries.

The low cost:

There are many languages which are available at very less cost. There are some languages which are available at very less cost like below:

- a. PHP
- b. C
- c. C++ etc.

It's Open Source, We can modify it: We can modify it if you need a hole in your head! Technically the point is that it's an open source project and they release patches often. You're point is that the community is actively working out the bugs. So, what any active language is doing this... Unfortunately C, C++ and Perl have all "died" at this point and will pretty much remain static at their current functionality.

It's Portability:

C is portable; it's just the OS bits that aren't. A lot PHP isn't portable to Windows since people don't use the OS abstractions to avoid some problems.

It has interfaces to a large variety of database systems:

PHP supports a large variety of the database.

Support available:

Online Support is available for using PHP.

PHP Syntax:

You cannot view the PHP source code by selecting "View source" in the browser – you will only see the output from the PHP file, which is plain HTML. This is because the scripts are executed on the server before the result is sent back to the browser.

2.4 Basic PHP Syntax

A PHP scripting block always starts with `<? Php` and ends with `?>`. A PHP scripting block can be placed anywhere in the document. On servers with shorthand support enabled you can start a scripting block with `<?` And end with `?>`. However, for maximum compatibility, we

recommend that you use the standard form (`<? php`) rather than the shorthand form. A PHP file normally contains HTML tags, just like an HTML file, and some PHP scripting code.

2.5 HTML

HTML or **Hyper Text Mark-up Language** is the standard mark-up language used to create web pages. HTML was created in 1991 by Tim Berners-Lee at CERN in Switzerland. It was designed to allow scientists to display and share their research.

HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like `<html>`). HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although some tags represent *empty elements* and so are unpaired, for example ``. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags).

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a mark-up language rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as Java Script which affect the behaviour of HTML web pages. HTML is descriptive mark-up language. Library of various mark-up languages is defined in various browsers.

HTML Images - The `` Tag and the Src Attribute:

In HTML, images are defined with the `` tag.

The `` tag is empty, which means that it contains attributes only, and has no closing tag.

To display an image on a page, you need to use the `src` attribute. `Src` stands for "source". The value of the `src` attribute is the URL of the image you want to display.

Syntax for defining an image:

```

```

An HTML form can contain input elements like text fields, checkboxes, and radio-buttons, submit buttons and more. A form can also contain select lists, text area, field set, legend, and label elements.

Image tag (): To add an image to an HTML document, we just need to include an tag with a reference to the desired image. The tag is an empty element i.e. it doesn't require a closing tag and we can use it to include from small icons to large images.

Syntax:

2.6 HTML 5

HTML5 will be the new standard for HTML. The previous version of HTML, HTML 4.01, came in 1999. The web has changed a lot since then. HTML5 is still a work in progress. However, the major browsers support many of the new HTML5 elements and APIs. HTML5 is cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

WHATWG was working with web forms and applications, and W3C was working with XHTML 2.0. In 2006, they decided to cooperate and create a new version of HTML.

Some rules for HTML5 were established:

- a) New features should be based on HTML, CSS, DOM, and JavaScript
- b) Reduce the need for external plug-ins (like Flash)
- c) Better error handling
- d) More mark-up to replace scripting
- e) HTML5 should be device independent
- f) The development process should be visible to the public

2.7 CSS

CSS tutorial or CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

- a) CSS stands for Cascading Style Sheet.
- b) CSS is used to design HTML tags.
- c) CSS is a widely used language on the web.
- d) HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements 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 pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).

CSS can also allow the same mark-up page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified.

With plain HTML you define the colours and sizes of text and tables throughout your pages. If you want to change a certain element you will therefore have to work your way through the document and change it. With CSS you define the colours and sizes in "styles". Then as you write your documents you refer to the styles. Therefore: if you change a certain style it will change the look of your entire site. Another big advantage is that CSS offers much more detailed attributes than plain HTML for defining the look and feel of your site.

2.8 JAVASCRIPT

JavaScript (JS) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side network programming (with Node.js), game development and the creation of desktop and mobile applications. JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

The application of JavaScript in use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers. JavaScript was formalized in the ECMA Script language standard and is primarily used as part of a web browser (client-side JavaScript). This enables programmatic access to objects within a host environment. JavaScript is the most popular programming language in the world. It is the language for HTML, for the Web, for computers, servers, laptops, tablets, smart phones, and more.

You can use JavaScript to:

- Change HTML elements
- Delete HTML elements
- Create new HTML elements
- Copy and clone HTML elements

ABOUT BACK END

In a previous blog, we talked about how web programmers are concerned with launching websites, updates, and maintenance, among other things. All of that works to support the front-end of the website. The back-end has three parts to it: server, application, and database. To better explain how all of this works, let's use the example of a customer trying to purchase a plane ticket using a website. Everything that the customer sees on the webpage is the front-end, as we have explained before, but once that customer enters all of his or her information, such as their name, billing address, destination, etc. the web application stores the information in a database that was created previously on the server in which the website is calling for information. The web application creates, deletes, changes, renames, etc. items in the database. For example, when a customer purchases a ticket, that creates an item in the database, but when they have a change in their order or they wish to cancel, the item in the database is changed. In short, when a customer wants to buy a ticket, the backend operation is the web application communicating with the server to make a change in a database stored on said server. Technologies like PHP, Ruby, Python, and others are the ones backend programmers use to make this communication work smoothly, allowing the customer to purchase his or her ticket with ease.

3.1 MySQL's Logical Architecture

The topmost layer contains the services that aren't unique to MySQL. They're services most network-based client/server tools or servers need: connection handling, authentication, security, and so forth.

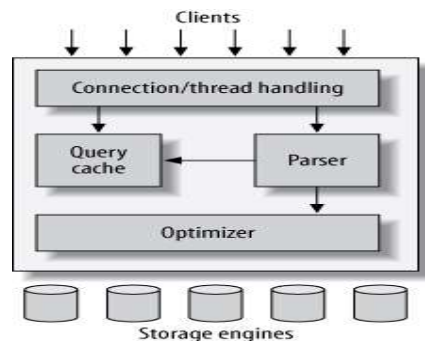


Fig. 3.1

The third layer contains the storage engines. They are responsible for storing and retrieving all data stored “in” MySQL. Like the various file systems available for GNU/Linux, each storage engine has its own benefits and drawbacks. The server communicates with them through the storage engine *API*. This interface hides differences between storage engines and makes them largely transparent at the query layer. The API contains a couple of dozen low-level functions that perform operations such as “begin a transaction” or “fetch the row that has this primary key.” The storage engines don’t parse SQL or communicate with each other; they simply respond to requests from the server.

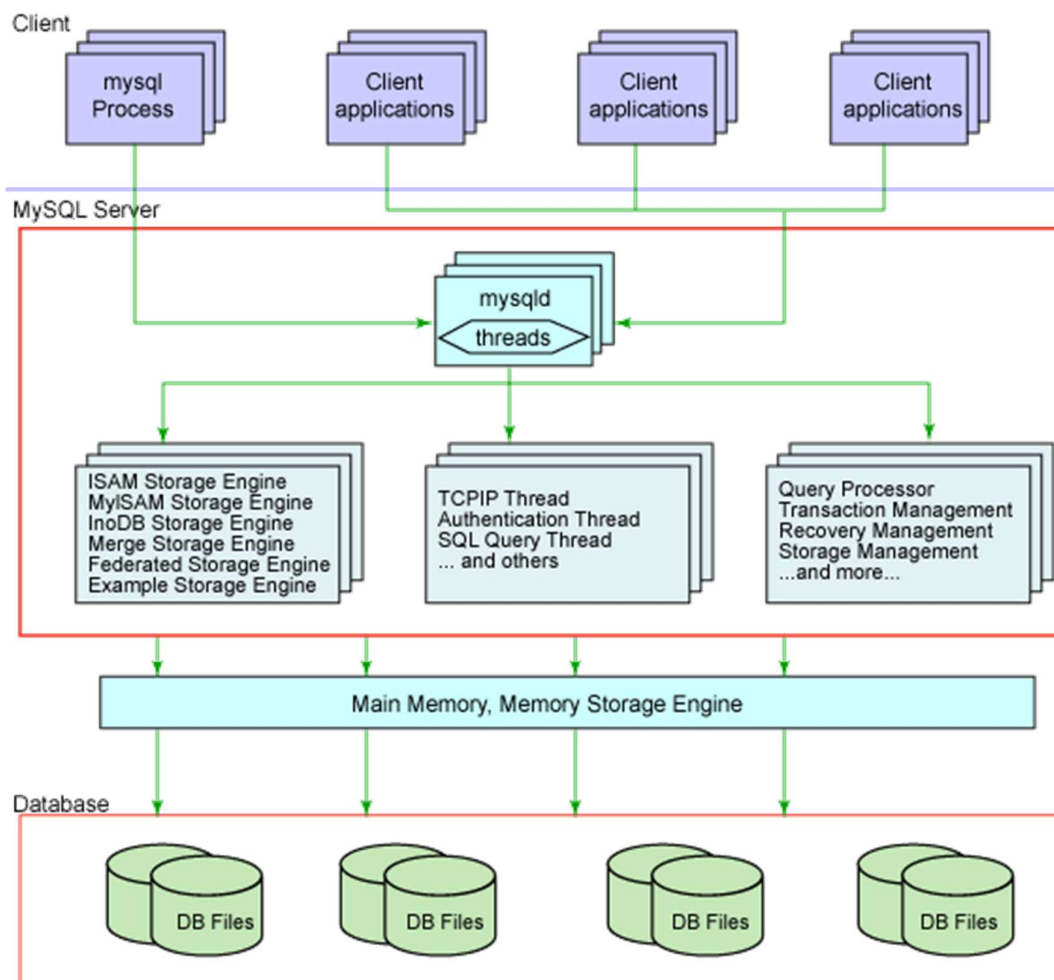


Fig. 3.2

3.2 Software's and tools used

My Sql:



The database has become an integral part of almost every human's life. Without it, many things we do would become very tedious, perhaps impossible tasks. Banks, universities, and libraries are three examples of organizations that depend heavily on some sort of database system. On the Internet, search engines, online shopping, and even the website naming convention would be impossible without the use of a database. One of the fastest SQL (Structured Query Language) database servers currently on the market is the MySQL server, developed by T.c.X. DataKonsultAB. MySQL, available for download at www.mysql.com, offers the database programmer with an array of options and capabilities rarely seen in other database servers. MySQL is free of charge for those wishing to use it for private and commercial use. Those wishing to develop applications specifically using MySQL should consult MySQL's licensing section, as there is charge for licensing the product.

These capabilities range across a number of topics, including the following:

- a) Ability to handle an unlimited number of simultaneous users.
- b) Capacity to handle 50,000,000+ records.
- c) Very fast command execution, perhaps the fastest to be found on the market.
- d) Easy and efficient user privilege system.

Reasons to Use MySQL:

- a) Scalability and Flexibility

The MySQL database server provides the ultimate in scalability, sporting the capacity to handle deeply embedded applications with a footprint of only 1MB to running massive data warehouses holding terabytes of information. Platform flexibility is a stalwart feature of MySQL with all flavours of Linux, UNIX, and Windows being supported.

- b) High Performance

A unique storage-engine architecture allows database professionals to configure the MySQL database server specifically for particular applications, with the end result being amazing performance results.

- C) High Availability

Rock-solid reliability and constant availability are hallmarks of MySQL, with customers relying on MySQL to guarantee around-the-clock uptime. MySQL offers a variety of high-availability options from high-speed master/slave replication configurations, to specialized Cluster servers offering instant failover, to third party vendors offering unique high-availability solutions for the MySQL database server.

d) Robust Transactional Support

MySQL offers one of the most powerful transactional database engines on the market. Features include complete ACID (atomic, consistent, isolated, durable) transaction support, unlimited row-level locking, distributed transaction capability, and multi-version transaction support where readers never block writers and vice-versa.

e) Web and Data Warehouse Strengths

MySQL is the de-facto standard for high-traffic web sites because of its high-performance query engine, tremendously fast data inserts capability, and strong support for specialized web functions like fast full text searches.

f) Strong Data Protection

Because guarding the data assets of corporations is the number one job of database professionals, MySQL offers exceptional security features that ensure absolute data protection. In terms of database authentication, MySQL provides powerful mechanisms for ensuring only authorized users have entry to the database server, with the ability to block users down to the client machine level being possible.

g) Management Ease

MySQL offers exceptional quick-start capability with the average time from software download to installation completion being less than fifteen minutes. This rule holds true whether the platform is Microsoft Windows, Linux, Macintosh, or UNIX.

3.3 PHP Main Features of MySQL

- The MySQL Server design is multi-layered with independent modules.
- Fully multi-threaded using kernel threads. It can easily use multiple CPUs if they are available.
- Provides transactional and non-transactional storage engines.
- Uses very fast B-tree disk tables with index compression.
- Relatively easy to add other storage engines. This is useful if you want to provide an SQL interface for an in-house database.
- A very fast thread-based memory allocation system.
- Very fast joins using an optimized one-sweep multi-join.
- In-memory hash tables, which are used as temporary tables.
- SQL functions are implemented using a highly optimized class library and should be as fast as possible. Usually there is no memory allocation at all after query initialization.
- The server is available as a separate program for use in a client/server networked environment.

3.4 XAMPP

XAMPP is the title used for a **compilation of free software**. The name is an acronym, with each letter representing one of the five key components. The software packet contains the web server **Apache**, the relational database management system **MySQL** (or **Maria DB**), and the scripting languages **Perl** and **PHP**. The initial **X** stands for the operating systems that it works with: Linux, Windows, and Mac OS X.

- **Apache:** Open source web server Apache is the most widely used server worldwide for delivery of web content. The server application is made available as a free software by the Apache Software Foundation.
- **MySQL/Maria DB:** in MySQL, XAMPP contains one of the most popular relational database management systems in the world. In combination with the web server Apache and the scripting language PHP, MySQL offers data storage for web services.

Current XAMPP versions have replaced MySQL with Maria DB (a community-developed fork of the MySQL project, made by the original developers).

- **PHP:** the server-side programming language PHP enables users to create dynamic websites or applications. PHP can be installed on all platforms and supports a number of diverse database systems.
- **Perl:** the scripting language Perl is used in system administration, web development, and network programming. Like PHP, Perl also enables users to program dynamic web applications.

Application areas:

An XAMPP server can be installed and used with a single executable file quickly and easily, functioning as a local test system for Linux, Windows, and Mac OS X. The software packet contains the same components that are found on common web servers. Developers have the chance to test out their projects locally and to transfer them easily to productive systems. But XAMPP isn't suitable to use as a public server, because many safety features have been deliberately left out to simplify and speed up the system for testing.

SOFTWARE REQUIREMENT

- Sublime Editor
- MS Sql Server
- Window 7/8/8.1/10

HARDWARE REQUIREMENT

- Processor : Intel Core Duo 2.0 GHz or more
- RAM : 1 GB or More
- Hard disk : 80GB or more
- Monitor : 15” CRT, or LCD monitor
- Keyboard : Normal or Multimedia
- Mouse : Compatible mouse

SOFTWARE DEVELOPMENT PROCESS

6.1 SDLC

The Systems Development Life Cycle (SDLC) is a conceptual model used in project management that describes the stages involved in an information system development project from an initial feasibility study through maintenance of the completed application. Various SDLC methodologies have been developed to guide the processes involved including the waterfall model (the original SDLC method), rapid application development (RAD), joint application development (JAD), the fountain model and the spiral model. Mostly, several models are combined into some sort of hybrid methodology.

Feasibility Study:

The first step is to study whether or not it is practical to develop the software or whether or not it is worth carrying on with the requirement analysis. The following questions should be answered in this step:

- Does the software contribute to the overall objective of the organization?
- Can the software be implemented using current technology and within given cost and schedule constraints?
- Can the software be integrated with other software in the application domain.

A widely used model to document requirement is called use case, which describes the interaction of one or several scenarios. A use case usually consists of the following parts. A description of what the software and users expect when the scenario starts.

- A description of the normal flow of events in the scenario.
- A description of what can go wrong and how this is handled and
- A description of the system state when the scenario finishes.

6.2 Requirement validation

This is the process to show that the requirements actually refine the software that the users want.

Economic Analysis:

Among the most important information contained in feasibility study is Cost Benefit Analysis and assessment of the economic justification for a computer based system project. Cost Benefit Analysis delineates costs for the project development and weighs them against

tangible and intangible benefits of a system. Cost Benefits Analysis is complicated by the criteria that vary with the characteristics of the system to be developed, the relative size of the project and the expected return on investment desired as part of company's strategic plan. In addition, many benefits derived from a computer-based system are intangible (e.g. better design quality through iterative optimization, increased customer satisfaction through programmable control etc.) As this is an in-house project for the company, to be used for its own convenience and also it is not that big a project. So neither it requires a huge amount of money nor any costly tools or infrastructure need to be set up for it.

Technical Analysis:

During technical analysis, the technical merits of the system are studied and at the same time collecting additional information about performance, reliability, maintainability and predictability. Technical analysis begins with an assessment of the technical viability of the proposed system.

What technologies are required to accomplished system function and performance?

How will these obtained from technical analysis form the basis for another go/no-go decision on the test system? If the technical risk is severe, if models indicate that the desired function cannot be achieved, if the pieces just won't fit together smoothly-it's back to the drawing board.

As the software is very much economically feasible, then it is really important for it to be technically sound.

System Analysis:

System analysis is the process of studying the business processors and procedures, generally referred to as business systems, to see how they can operate and whether improvement is needed. This may involve examining data movement and storage, machines and technology used in the system, programs that control the machines, people providing inputs, doing the processing and receiving the outputs.

6.3 Constraints and Limitations

The constraints and limitation within a system are the drawbacks that occur during the implementation of the system. These limitations and constraints can crop up in almost every system; the most important fact is to find a way to overcome these problems.

Software design is the first of three technical activities – design, code generation, and test that are required to build and verify the software. Each activity transforms information in manner that ultimately results in validated computer software. The design task produces a data design, an architectural design, an interface design and component design.

The design of an information system produces the details that clearly describe how a system will meet the requirements identified during system analysis. The system design process is not a step by step adherence of clear procedures and guidelines. When I started working on system design, I face different types of problems; many of these are due to constraints imposed by the user or limitations of hardware and software available. Sometimes it was quite difficult to enumerate that complexity of the problems and solutions thereof since the variety of likely problems is so great and no solutions are exactly similar however the following consideration I kept in mind during design phased.

Output

7.1 Main Dashboard

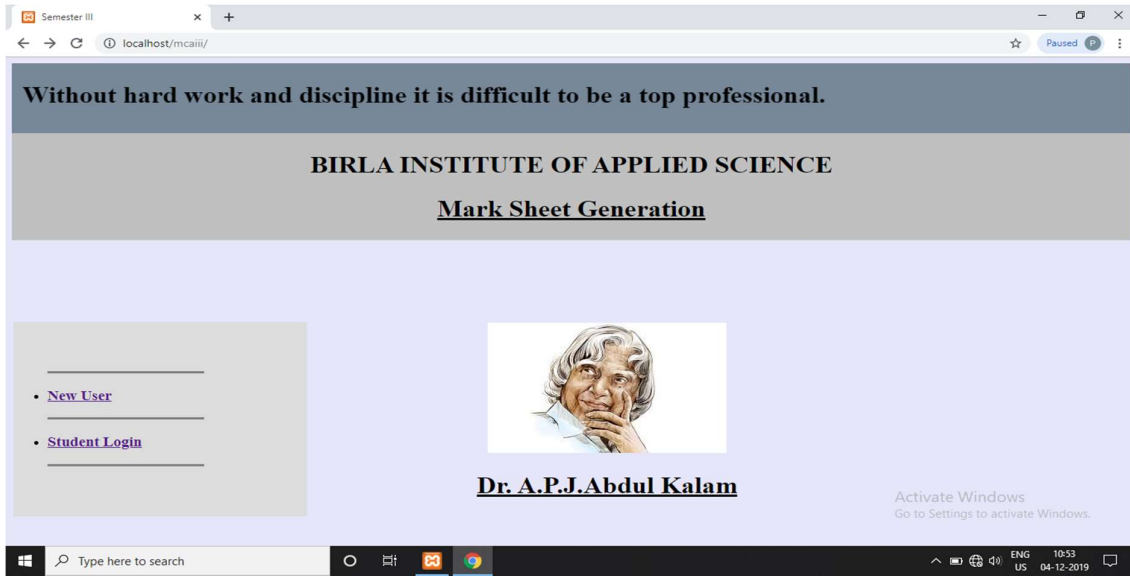


Fig. 7.1

7.2 Registration Page



Fig. 7.2

7.3 Student Login

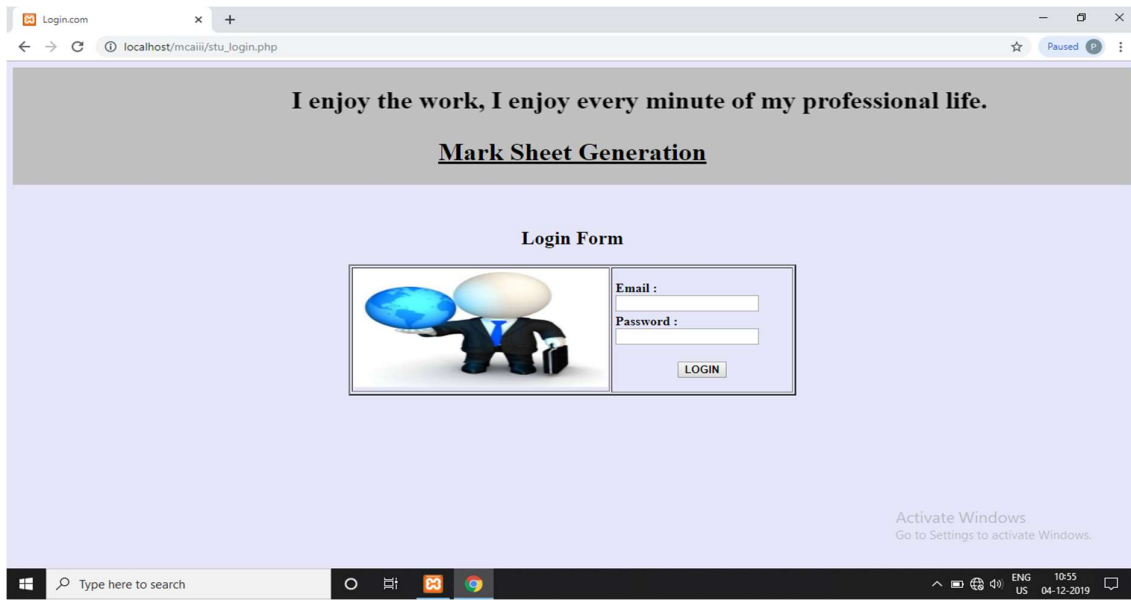


Fig. 7.3

7.4 Student Dashboard

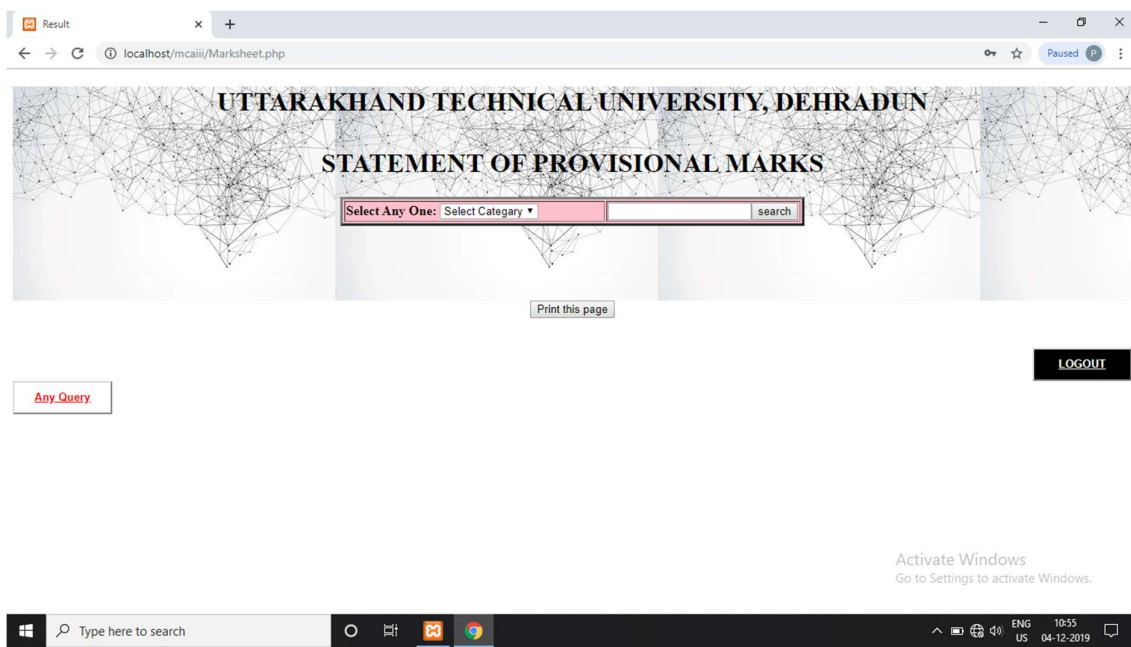


Fig. 7.4

7.5 After Search

The screenshot displays a web browser window with the URL `localhost/mcaiii/Marksheet.php`. The page shows the 'STATEMENT OF PROVISIONAL MARKS' for the MCA III SEMESTER at Uttarakhand Technical University, Dehradun. The student's name is MAANAS PRATAP SINGH, and the roll number is 120050102028. The father's name is VEERESH PRATAP SINGH. The table below lists the subjects, maximum marks, and marks obtained.

Subject Code & Name	Maximum Marks		Total	Marks Obtained		Total
	Ext.	Sess.		Ext.	Sess.	
Engineering Mathematics-I, TMA101	100	50	150	30	30	60
Basic Technical Communication, THM101	100	50	150	45	32	77
Engineering Chemistry, TCY101	100	50	150	73	42	115
Basic Mechanical Engineering, TME 101	100	50	150	55	41	96
Fundamentals of Electronic Engineering, TEC 101	100	50	150	40	34	74
Chemistry Lab, PCY 101	25	25	50	21	21	42
Basic Mechanical Engineering Lab, PME 101	25	25	50	23	24	47
Fundamentals of Electronic Engineering Lab, PEC 101	25	25	50	17	17	34
Engineering Drawing Lab, PED 101	25	25	50	21	23	44
Grand Total	950			589		
			Percentage of Marks	62		

Below the table, it states: * Marks as per University Cross list. There is a 'Print this page' button and an 'Activate Windows' watermark.

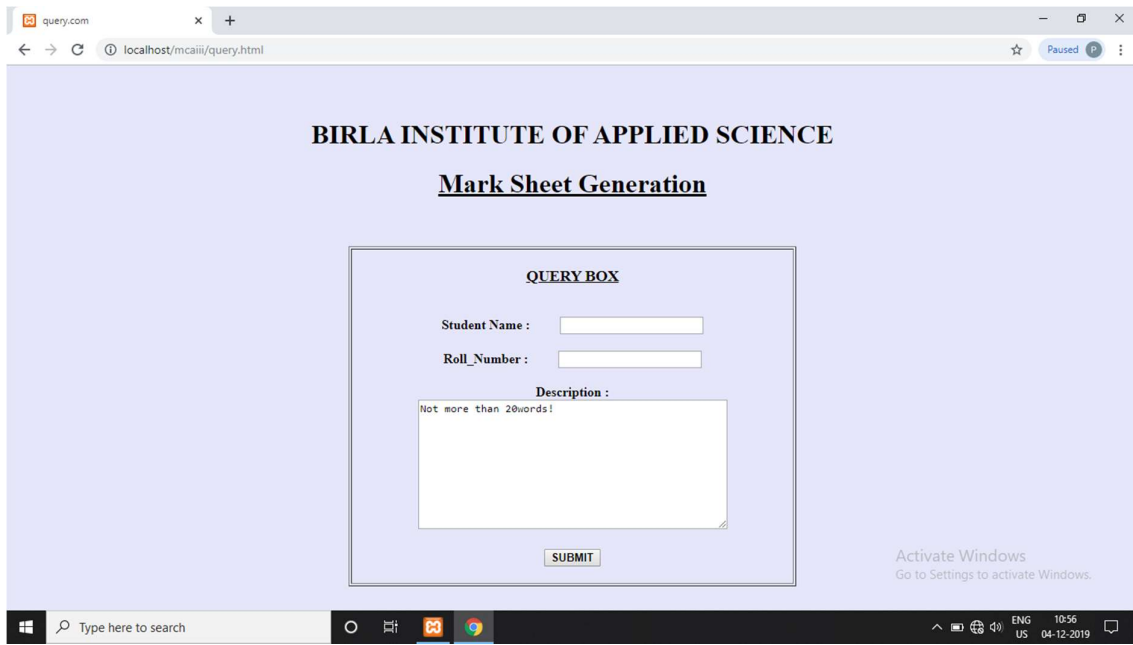
Fig. 7.5

7.6 Print Marksheet

The screenshot shows the same marksheet as in Fig. 7.5, but with the print dialog open. The print settings are configured for 2 sheets of paper, Microsoft Print to PDF as the destination, All pages, Portrait layout, and Color. The 'Print' button is highlighted in blue. The 'LOGOUT' button is also visible. The 'Any Query' button is at the bottom left.

Fig. 7.6

7.7 Student Query



The screenshot shows a web browser window with the address bar displaying 'localhost/mcaiii/query.html'. The page has a light blue background. At the top, the text 'BIRLA INSTITUTE OF APPLIED SCIENCE' is centered, followed by the title 'Mark Sheet Generation' in a bold, underlined font. Below this is a 'QUERY BOX' containing three input fields: 'Student Name', 'Roll_Number', and 'Description'. The 'Description' field has a placeholder text 'Not more than 20words!'. A 'SUBMIT' button is located at the bottom of the form. In the bottom right corner of the page, there is a watermark that says 'Activate Windows Go to Settings to activate Windows.' The Windows taskbar is visible at the bottom of the screen.

BIRLA INSTITUTE OF APPLIED SCIENCE

Mark Sheet Generation

QUERY BOX

Student Name :

Roll_Number :

Description :
Not more than 20words!

Activate Windows
Go to Settings to activate Windows.

Fig. 7.7

7.8 Admin Login



The screenshot shows a web browser window with the address bar displaying 'localhost/mcaiii/admin.php'. The page has a light blue background. At the top, the text 'BIRLA INSTITUTE OF APPLIED SCIENCE' is centered, followed by the title 'Mark Sheet Generation' in a bold, underlined font. Below this is an 'Admin Login' section. On the left side of this section is an image of a 3D character in a suit holding a globe. On the right side is a login form with two input fields: 'Email' and 'Password', and a 'LOGIN' button. In the bottom right corner of the page, there is a watermark that says 'Activate Windows Go to Settings to activate Windows.' The Windows taskbar is visible at the bottom of the screen.

BIRLA INSTITUTE OF APPLIED SCIENCE

Mark Sheet Generation

Admin Login



Email :

Password :

Activate Windows
Go to Settings to activate Windows.

Fig. 7.8

7.9 Admin Dashboard

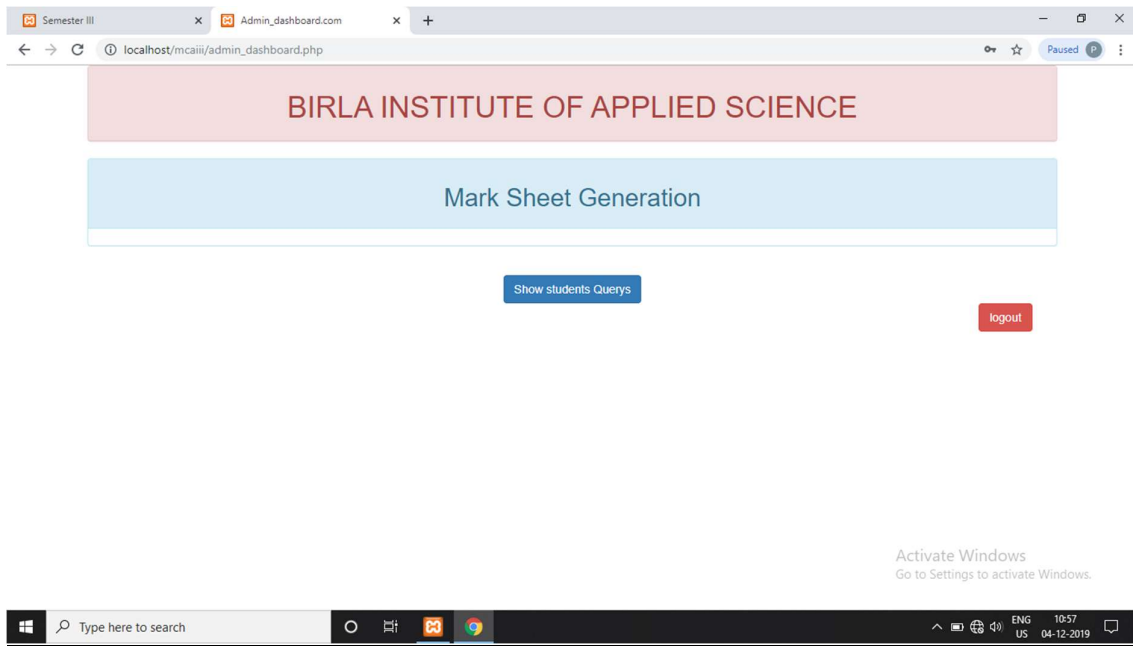


Fig. 7.9

7.10 Showing Student Query

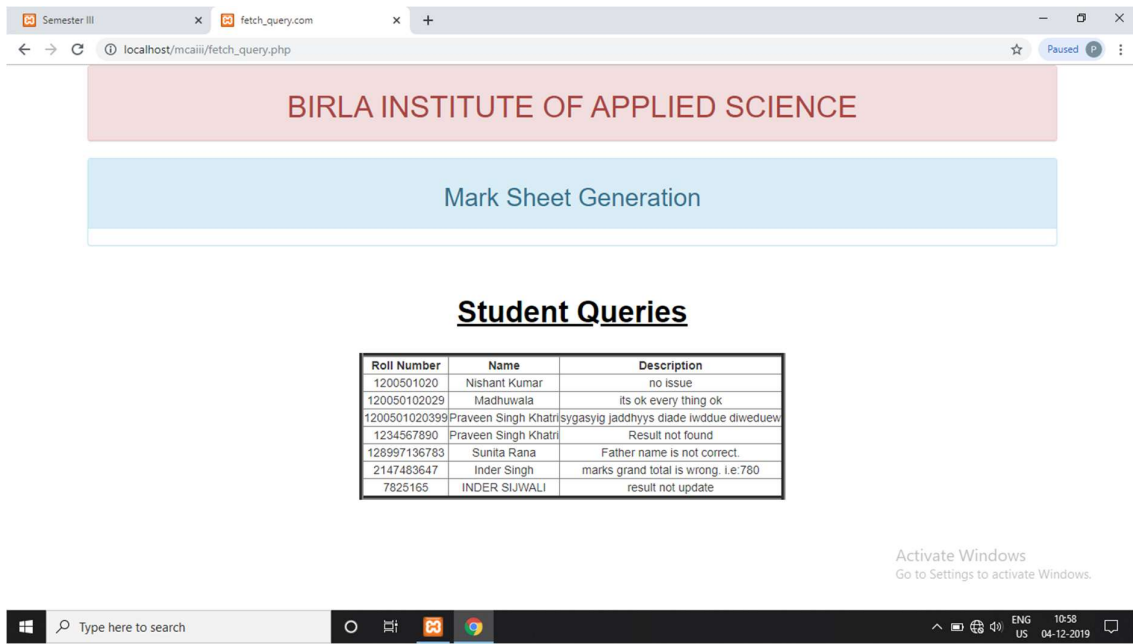


Fig. 7.10

Bibliography

<https://www.w3schools.com/php/>

<https://www.javatpoint.com/php-tutorial>

<https://stackoverflow.com/>

<https://www.quora.com/>

<https://www.apachefriends.org/>

<https://www.phptpoint.com/php-tutorial/>

<https://www.youtube.com/>