

Chapter 1

Introduction to Web Development

Among web professionals, "web development" usually refers to the main non-design aspects of building web sites: writing markup and coding. Web development may use content management systems (CMS) to make content changes easier and available with basic technical skills.

Web development can range from developing a simple single static page of plain text to complex web-based internet applications (web apps), electronic businesses, and social network services. A more comprehensive list of tasks to which web development commonly refers, may include web engineering, web design, web content development, client liaison, client-side/server-side scripting, web server and network security configuration, and e-commerce development.

DEFINITION: **Web development** is the work involved in developing a web site for the Internet (World Wide Web) or an intranet (a private network).

For larger organizations and businesses, web development teams can consist of hundreds of people (web developers) and follow standard methods like Agile methodologies while developing websites. Smaller organizations may only require a single permanent or contracting developer, or secondary assignment to related job positions such as a graphic designer or information systems technician. Web development may be a collaborative effort between departments rather than the domain of a designated department.

1.1 WHY WEB DEVELOPMENT?

To answer this question, we should look at two issues:

- (1). What are the goals of Web Development
- (2). Why these goals are important and desirable

1.1.1 The Goals of Web Development

One of the greatest advances in technological tools has been the use of the Internet, due to its rapid reception, its transfer speeds and the effectiveness of resources, since access to its different applications is almost immediate, information of any content can be found , in any class and of any kind, it is important to set clear goals and objectives, in the creation of websites defining the service and use that will be given to the information, having a clear vision of the type of audience that will host these pages and will be potential users of them.

Currently the development and creation of websites is imposed on the world as a technological tool to unite the regions, create businesses, support companies, be displayed in the market and infinity of applications according to the perspectives of people and their scope, finding on the internet, a vital source of job creation, efficient and intelligent businesses and great help in achieving objectives and approaches.

Web development in recent decades has been transforming and driving the evolution of the Internet, every day greater progress on this issue, directly involving the ingenuity of the developer on the subject.

With the passing of time, new steps are being taken in search of a more stable, creative and efficient web environment where user interaction with the network is simple and practical for users.

1.1.2 Why the goals of Web-D are important and desirable.

It is self-evident that the goals of Web-D are important and desirable. However, we still give some more supporting argument to this issue.

First of all, the web entrepreneur must know deeply about the design of virtual spaces, having clear the minimum requirements for the creation of sites, where the priority is the information and the use of it creating rapid and efficient exchange methods, to carry out projects that comply with the required guidelines; One of the tasks is to create systems that generate solutions to problems related to communication, currently based on the use of the Internet.

In order for websites to be dynamic and efficient, very specific and relevant aspects must be taken into account for the proper use of this technology, both the technical part and the use of external resources, it is not the same to see a website in the city where data transfer resources are high to visualize it in rural environments where resources are limited or nonexistent, this is one of the great risks that are faced, which can lead to losing interest in it because it is not accessible , so the planning is vital for the good performance of the same in environments of different characteristics being just one of the problems that can affect the creation of a website.

The implementation of new updates, allow developers to be proactive in the solution and implementation of new designs more effective and competitive, leading by new directions in search of a final product according to the needs of each new user, where the marketing of These pages require web designers innovation in creation. You should have ideas that awaken in the users an interest when exploring these proposals.

The stages of development of current websites and their programming languages are very complex because of their extensive and varied code. Therefore, it is necessary to know tools that are currently available to dynamically develop web pages.

For the initiation of website design it is very important to have knowledge bases in the use of the programming language, both in HTML and other tools such as PHP, JAVA-SCRIPT, Dreamweaver, among others. They are a very interesting tool that induces joint work in the wonderful world of web page design, with the appropriation and correct handling of the tools, you can get to operate innovative and creative designs when marketing and offering the products made.

The network is a mechanism where information is stored and processed at great speeds, mostly handled through web pages, where daily users perform a number of operations according to their web management.

Chapter 2

Web Development Overview

There are two broad divisions of web development –

- (1) Front-end development (also called client-side development)
- (2) Back-end development (also called server-side development).

Front-end development refers to constructing what a user sees when they load a web application – the content, design and how you interact with it. This is done with three codes – HTML, CSS and JavaScript.

- **HTML** – All code in a web application is eventually translated to HTML. It's the language that web browsers understand and use to display information to users. A web developer's understanding of HTML is analogous to a carpenter's understanding of a screwdriver. It's so important and necessary that it's often assumed for employment.
- **CSS** – By itself, HTML is quite plain. HTML does provide some basic style options, but to build a good frontend, developers must have experience with CSS. CSS provides the paint, templates, glitter, buttons, tassel, lights, and many other things that can be used to improve the presentation of a web page. CSS is so commonly used that languages have been built to make writing CSS easier. These languages – like Sass and LESS – are also known as CSS precompilers, but they are simply used to write more efficient and manageable CSS code.
- **JavaScript** – If you could only learn one language in your lifetime, you'd be well-advised to choose JavaScript. Though it's not exclusively a frontend language, that's where it's most commonly used. JavaScript is a language that is run on a client machine, i.e. a user's computer. This means that JavaScript can be used to program fast, intuitive and fun user experiences, without forcing a user to refresh their web page. Drag-and-drop, infinite-scroll and videos that come to life on a web page can all be programmed with JavaScript. Frameworks like Angular, Ember, React and Backbone are all very widely used for JavaScript-heavy frontends.

Here's an example. Say you log into your Facebook account, and you are greeted with the latest updates in your News Feed. They're not going to be the same updates that you saw yesterday.

How did the page change? Did a Facebook employee manually edit the page to update your news feed? Of course not. A script on the Facebook back-end would have received the updates and re-generated the front-end accordingly.

Back-end scripts are written in many different coding languages and frameworks, such as...

- PHP
- Ruby on Rails
- ASP.NET
- Perl
- Java
- Node.js
- Python

Back-end Development is the term used for the behind-the-scenes activities that happen when performing any action on a website. It can be logging in to your account or purchasing a watch from an online store.

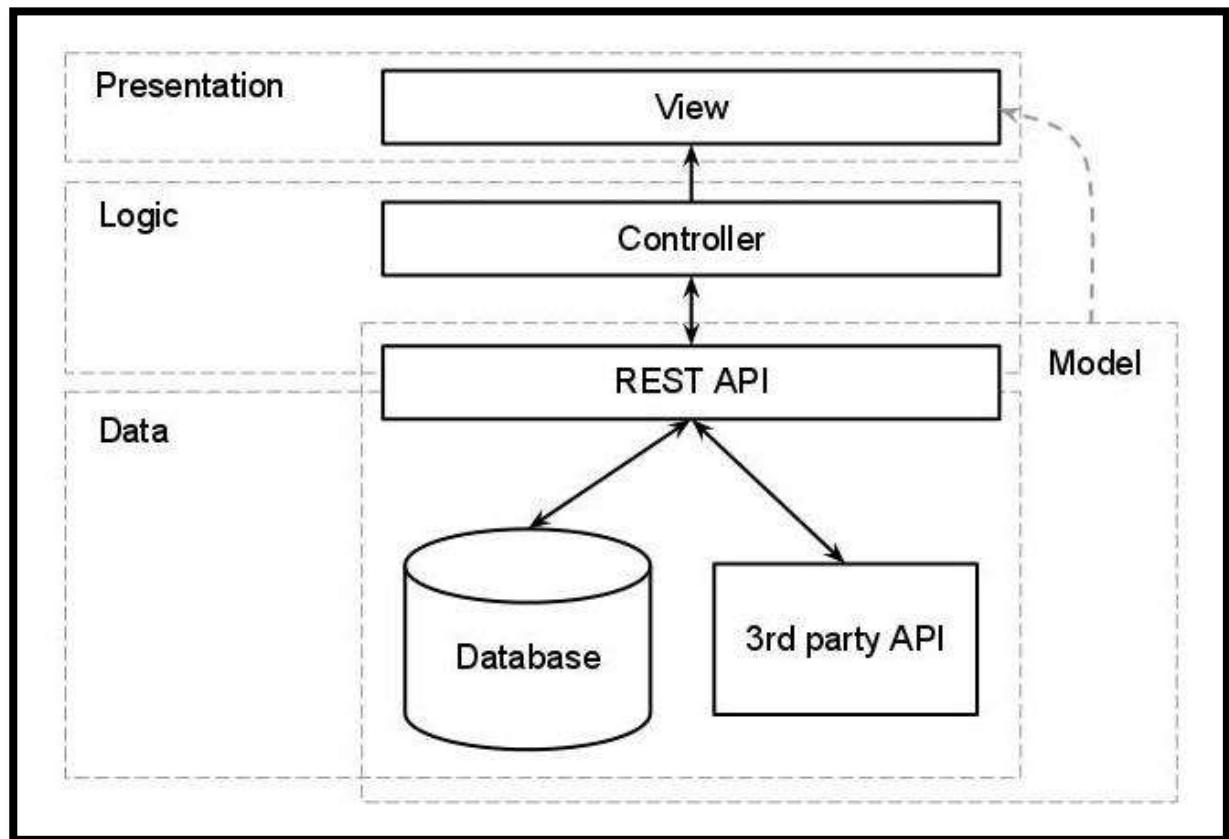
Backend developer focuses on databases, scripting, and the architecture of websites. Code written by back-end developers helps to communicate the database information to the browser. Most common example of Backend programming is when you are reading an article on the blog. The fonts, colors, designs, etc. constitute the frontend of this page. While the content of the article is rendered from a server and fetched from a database. This is the backend part of the application

The backend of a web application is an enabler for a frontend experience. An application's frontend may be the most beautifully crafted web page, but if the application itself doesn't work, the application will be a failure. The backend of an application is responsible for things like calculations, business logic, database interactions, and performance. Most of the code that is required to make an application

work will be done on the backend. Backend code is run on the server, as opposed to the client. This means that backend developers not only need to understand programming languages and databases, but they must have an understanding of server architecture as well. If an application is slow, crashes often, or constantly throws errors at users, it's likely because of backend problems.

Backend development is not all ones and zeros though. Much like frontend development, backend development has a human aspect to it as well. Since most of the code for an application is written on the backend, it should be easy to understand and work with. Most backend languages – like Ruby and Python – have standardized styles and idioms that make reading and writing code more efficient and enjoyable.

2.1 The Architecture of a PHP Web Application



INTRODUCTION TO HTML

HTML is a Hypertext Markup Language for describing web documents (web pages).

- HTML stands for Hyper Text Markup Language
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page

Basics of HTML

Syntax:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>
<h1>Heading</h1>
<p>Paragraph</p>
</body>
</html>
```

Syntax Explained:

- The `<!DOCTYPE html>` declaration defines this document to be HTML5
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the document
- The `<title>` element specifies a title for the document
- The `<body>` element contains the visible page content
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

Paragraphs Tag: The `<p>` tag defines a paragraph. Browsers automatically add some space (margin) before and after each `<p>` element. The margins can be modified with CSS (with the margin properties).

Line Breaks: The `
` tag inserts a single line break. The `
` tag is an empty tag which means that it has no end tag.

Headings: Headings are defined with the `<h1>` to `<h6>` tags. `<h1>` defines the most important heading. `<h6>` defines the least important heading.

Strong and Emphasized text: The `` tag is a phrase tag. It renders as emphasized text. The `` tag defines important text.

Html Formatting Tag: HTML uses tags like `` and `<i>` for formatting output, like **bold** and *italic* text respectively.

HTML Lists

Unordered HTML Lists: An unordered list starts with the `` tag. Each list item starts with the `` tag. The list items will be marked with bullets (small black circles).

Ordered HTML Lists: An ordered list starts with the `` tag. Each list item starts with the `` tag. The list items will be marked with numbers.

HTML Description Lists: A description list , is a list of terms, with a description of each term. The `<dl>` tag defines a description list. The `<dt>` tag defines the term (name), and the `<dd>` tag defines the data (description).

HTML Images: In HTML, images are defined with the `` tag. The `` tag is empty, it contains attributes only, and does not have a closing tag. The `src` attribute defines the url (web address) of the image

Syntax:- ``

HTML Links: HTML links are hyperlinks. A hyperlink is an element, a text, or an image that you can click on, and jump to another document.

Syntax:-link text

HTML Tables: Tables are defined with the <table> tag. Tables are divided into **table rows** with the <tr> tag. Table rows are divided into **table data** with the <td> tag. A table row can also be divided into **table headings** with the <th> tag.

HTML Forms

HTML forms are used to pass data to a server. An HTML form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain select lists, textarea, fieldset, legend, and label elements. The <form> tag is used to create an HTML form:

```
<form>
    input elements
</form>
```

The Input Element: The most important form element is the <input> element. The <input> element is used to select user information. An <input> element can vary in many ways, depending on the type attribute. An <input> element can be of type text field, checkbox, password, radio button, submit button, and more. The most common input types are described below:

<input type="text"> defines a one-line input field that a user can enter text into:

Example:

```
<form>
    First name: <input type="text" name="firstname"><br/>
    Last name: <input type="text" name="lastname">
</form>
```

Output:

First name:

Last name:

Password Field: <input type="password"> defines a password field:

Example:

```
<form>
    Password: <input type="password" name="pwd">
</form>
```

Output:

Password:

Radio Buttons: <input type="radio"> defines a radio button. Radio buttons let a user select ONLY ONE of a limited number of choices:

Example:

```
<form>
    <input type="radio" name="sex" value="male">Male<br/>
    <input type="radio" name="sex" value="female">Female
</form>
```

Output:

How the HTML code above looks in a browser:

- Male
- Female

Checkboxes Buttons: `<input type="checkbox">` defines a checkbox. Checkboxes let a user select ZERO or MORE options of a limited number of choices.

Example:

```
<form>
    <input type="checkbox" name="vehicle" value="Bike">I have a bike<br/>
    <input type="checkbox" name="vehicle" value="Car">I have a car
</form>
```

Output:

How the HTML code above looks in a browser:

- I have a bike
- I have a car

HTML Headings

HTML headings are defined with the `<h1>` to `<h6>` tags.

`<h1>` defines the most important heading. `<h6>` defines the least important heading:

Example:

```
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>
```

Output:

How the HTML code above looks in a browser

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

HTML Links

HTML links are defined with the `<a>` tag:

The link's destination is specified in the `href` attribute.

Attributes are used to provide additional information about HTML elements

Example:

```
<a href="#">This is a link</a>
```

Output:

How the HTML code above looks in browser

[This is a link](#)

- CSS stands for Cascading Style Sheets
- CSS describes **how HTML elements are to be displayed on screen, paper, or in other media**
- CSS **saves a lot of work.** It can control the layout of multiple web pages all at once
- External stylesheets are stored in **CSS files**

Why Use CSS?

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

HTML was NEVER intended to contain tags for formatting a web page!

HTML was created to **describe the content** of a web page, like:

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

When tags like ``, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS.

CSS removed the style formatting from the HTML page!

CSS is mainly of 3 types:-

1. Inline CSS.
2. Internal CSS.

3. External CSS.

Inline CSS

I have not mentioned them until now because in a way they defeat the purpose of using CSS in the first place. Inline styles are defined right in the (X)HTML file alongside the element you want to style.

See example below.

```
<html>
  <head>
  </head>
  <body>
    <p style="color:blue">hello</p>
    <h1 style="font-family:arial">Baby</h1>
  </body>
</html>
```

Internal Css

First we will explore the internal method. This way you are simply placing the CSS code within the `<head></head>` tags of each (X)HTML file you want to style with the CSS.

The format for this is shown in the example below.

```
<html>
  <head>
    <style>
      h1 {
        Color:blue;
        Font-family:Times new Roman;
      }
    </style>
  </head>
  <body>
    <h1>HTML stands for Hyper Text markup language</h1>
  </body>
</html>
```

External CSS

An external CSS file can be created with any text or HTML editor such as "Notepad" or "Dreamweaver". A CSS file contains no (X) HTML, only CSS. You simply save it with

the .css file extension. You can link to the file externally by placing one of the following links in the head section of every (X)HTML file you want to style with the CSS file.

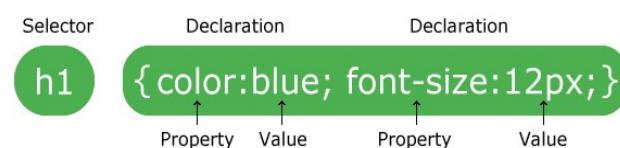
```
<link rel="stylesheet" type="text/css" href="Path To stylesheet.css" />
```

By using an external style sheet, all of your (X)HTML files link to one CSS file in order to style the pages. This means, that if you need to alter the design of all your pages, you only need to edit one .css file to make global changes to your entire website. Here are a few reasons this is better.

- (1) Easier Maintenance
- (2) Reduced File Size
- (3) Reduced Bandwidth
- (4) Improved Flexibility

CSS Syntax

A CSS rule-set consists of a selector and a declaration block:



The selector points to the HTML element you want to style.

The declaration block contains one or more declarations separated by semicolons.

Each declaration includes a CSS property name and a value, separated by a colon.

A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

Example

In this example all <p> elements will be center-aligned, with a red text color:

```
p {  
    color: red;  
    text-align: center;  
}
```

Output:



Hello World!

The above CSS would text-align to center

CSS Selectors

CSS selectors are used to "find" (or select) the HTML elements you want to style.

We can divide CSS selectors into five categories:

- Simple selectors (select elements based on name, id, class)
- Combinator selectors (select elements based on a specific relationship between them)
- Pseudo-class selectors (select elements based on a certain state)
- Pseudo-elements selectors (select and style a part of an element)
- Attribute selectors (select elements based on an attribute or attribute value)

Note: The universal selector (*) selects all HTML elements on the page.

Example The CSS rule below will affect every HTML element on the page:

```
* {  
    text-align: center;  
    color: blue;  
}
```

JavaScript was released by Netscape and Sun Microsystems in 1995. However, JavaScript is not the same thing as Java.

It is a programming language. It is an interpreted language. It is object-based programming. It is widely used and supported. It is accessible to the beginner.

Why JavaScript ?

JavaScript is one of the **3 languages** all web developers **must** learn:

1. **HTML** to define the content of web pages
2. **CSS** to specify the layout of web pages
3. **JavaScript** to program the behavior of web pages

Web pages are not the only place where JavaScript is used. Many desktop and server programs use JavaScript. Node.js is the best known. Some databases, like MongoDB and CouchDB, also use JavaScript as their programming language.

Uses of JavaScript

- Use it to add multimedia elements
With JavaScript you can show, hide, change, resize images, and create image rollovers.
You can create scrolling text across the status bar.
- Create pages dynamically
Based on the user's choices, the date, or other external data, JavaScript can produce pages that are customized to the user.
- Interact with the user
It can do some processing of forms and can validate user input when the user submits the form.

Writing JavaScript

JavaScript code is typically embedded in the HTML, to be interpreted and run by the client's browser. Here are some tips to remember when writing JavaScript commands.

- JavaScript code is case sensitive
- White space between words and tabs are ignored
- Line breaks are ignored except within a statement
- JavaScript statements end with a semi colon (;

What JavaScript can do?

1. JavaScript Can Change HTML Content

One of many JavaScript HTML methods is `getElementById()`.

This example uses the method to "find" an HTML element (with id="demo") and changes the element content (`innerHTML`) to "Hello JavaScript":

```
document.getElementById("demo").innerHTML = "Hello  
JavaScript";
```

2. JavaScript Can Change HTML Styles (CSS)

Changing the style of an HTML element, is a variant of changing an HTML attribute:

```
document.getElementById("demo").style.fontSize = "35px";
```

3. JavaScript Can Hide HTML Elements

Hiding HTML elements can be done by changing the `display` style:

```
document.getElementById("demo").style.display = "none";
```

The <script> Tag

In HTML, JavaScript code must be inserted between `<script>` and `</script>` tags.

```
<script>  
document.getElementById("demo").innerHTML = "My First JavaScript";  
</script>
```

JavaScript in <head> or <body>

You can place any number of scripts in an HTML document.

Scripts can be placed in the <body>, or in the <head> section of an HTML page, or in both.

JavaScript in <head>

In this example, a JavaScript function is placed in the <head> section of an HTML page.

The function is invoked (called) when a button is clicked:

```
<!DOCTYPE html>
<html>
<head>
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph After.";
}
</script>
</head>
<body>
<h1>A Web Page</h1>
<p id="demo">Paragraph Before</p>
<button type="button" onclick="myFunction()">Try it</button>
</body>
</html>
```

Output:

JavaScript in Head Paragraph before <input type="button" value="Try it"/>	JavaScript in Head Paragraph after <input type="button" value="Try it"/>
--	---

External JavaScript

Scripts can also be placed in external files:

External file: myScript.js

```
function myFunction() {  
    document.getElementById("demo").innerHTML = "Paragraph changed.";  
}
```

External scripts are practical when the same code is used in many different web pages. JavaScript files have the file extension **.js**. To use an external script, put the name of the script file in the `src` (source) attribute of a `<script>` tag:

Example

```
<script src="myScript.js"></script>
```

You can place an external script reference in `<head>` or `<body>` as you like. The script will behave as if it was located exactly where the `<script>` tag is located. External scripts cannot contain `<script>` tags.

External JavaScript Advantages

Placing scripts in external files has some advantages:

- It separates HTML and code
- It makes HTML and JavaScript easier to read and maintain
- Cached JavaScript files can speed up page loads

To add several script files to one page - use several script tags:

Example

```
<script src="myScript1.js"></script>  
<script src="myScript2.js"></script>
```

External References

External scripts can be referenced with a full URL or with a path relative to the current web page. This example uses a full URL to link to a script:

Example

```
<script src="https://www.w3schools.com/js/myScript1.js"></script>
```

Chapter-6

PHP (*HyperText Pre-Processor*)

What is PHP?

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

PHP is an amazing and popular language. It is powerful enough to be at the core of the biggest blogging system on the web (WordPress). It is deep enough to run the largest social network (Facebook). It is also easy enough to be a beginner's first server side language.

What is a PHP File?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have extension ".php"

What Can PHP Do?

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even flash movies. You can also output any text, such as XHTML and XML.

Why PHP?

- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is free. Download it from the official PHP resource: www.php.net
- PHP is easy to learn and runs efficiently on the server side

What's new in PHP 7

- PHP 7 is much faster than the previous popular stable release (PHP 5.6)
- PHP 7 has improved Error Handling
- PHP 7 supports stricter Type Declarations for function arguments
- PHP 7 supports new operators (like the spaceship operator: <=>)

Instead of lots of commands to output HTML (as seen in C or Perl), PHP pages contain HTML with embedded code that does "something" (in this case, output "Hi, I'm a PHP script!"). The PHP code is enclosed in special start and end processing instructions <?php and ?> that allow you to jump into and out of "PHP mode." What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server, generating HTML which is then sent to the client. The client would receive the results of running that script, but would not know what the underlying code was. You can even configure your web server to process all your HTML files with PHP, and then there's really no way that users can tell what you have up your sleeve.

The best things in using PHP are that it is extremely simple for a newcomer, but offers many advanced features for a professional programmer. Don't be afraid reading the long list of PHP's features. You can jump in, in a short time, and start writing simple scripts in a few hours. Although PHP's development is focused on server-side scripting, you can do much more with it.

A PHP script is executed on the server, and the plain HTML result is sent back to the browser.

Basic PHP Syntax

A PHP script can be placed anywhere in the document. A PHP script starts with `<?php` and ends with `?>`:

```
<?php  
// PHP code goes here  
?>
```

The default file extension for PHP files is ".php". A PHP file normally contains HTML tags, and some PHP scripting code. Below, we have an example of a simple PHP file, with a PHP script that uses a built-in PHP function "echo" to output the text "Hello World!" on a web page:

Example

```
<!DOCTYPE html>  
<html>  
<body>  
  
<h1>My first PHP page</h1>  
  
<?php  
echo "Hello World!";  
?>  
  
</body>  
</html>
```

Note: PHP statements end with a semicolon (;).

PHP Case Sensitivity

In PHP, NO keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are case-sensitive. In the example below, all three echo statements below are equal and legal:

Example

```
<!DOCTYPE html>
<html>
<body>
<?php
ECHO "Hello World!<br>";
echo "Hello World!<br>";
EcHo "Hello World!<br>";
?>
</body>
</html>
```

Look at the example below; only the first statement will display the value of the \$color variable! This is because \$color, \$COLOR, and \$coLOR are treated as three different variables:

Example

```
<!DOCTYPE html>
<html>
<body>
<?php
$color = "red";
echo "My car is " . $color . "<br>";
echo "My house is " . $COLOR . "<br>";
echo "My boat is " . $coLOR . "<br>";
?>

</body>
</html>
```

Chapter-7 ***Project Discussion and Snapshots***

Today ***Online Examination System*** has become a fast growing examination method because of its speed and accuracy. It is also needed less manpower to execute the examination. Almost all organizations now-a-days, are conducting their objective exams by online examination system, it saves students time in examinations. Organizations can also easily check the performance of the student that they give in an examination. As a result of this, organizations are releasing results in less time. It also helps the environment by saving paper.

According to today's requirement, ***online examination project in php*** is very useful to learn it.

What is an Online Examination System?

In an online examination system examine get their user id and password with his/her admit card. This id is already saved in the examination server. When examine login to the server he/she get his/her profile already register. On the certain time examine gets the message to start the examination. All answers given by examine are saved into the server with his/her profile information. Online examination system also allows to correct the answer if the examine needed to change any answer in the examination time duration, however, after the time duration any change will not allow. This also makes c checking the answer easy and error proof as computers are more accurate than man and provide fast results too. Php is a web base language so we can create an ***online examination system in PHP***.

Administrator of Online Examination has multiple features such as Add, Delete, Update Topics and Question.

To Login as Admin put inside your browser "www.applicationname/admin"

The user will automatically get the updated version by logging using the user ID and Password provided at the time of registration.

No need of reprinting, appearance, vigilance and the job is done.

Online examination system features

1. Login system must be present and secured by password.
2. Ability to save the answer given by the candidate along with the question.
3. Answer checking system should be available.
4. Could Update Profile
5. Log out after the over.
6. Admin Panel

Project objective:

Online examination system is a non removable examination pattern of today's life. We need more time saving and more accurate examination system as the number of applicants is increasing day by day. For all IT students and professionals, it is very important to have some basic understanding about the online examination system.

TECHNOLOGIES USED

1. Tools to Be Used

- Database design:- MySQL
- Website design:- Bootstrap with Custom Designing using CSS3, Wordpress
- Coding(logic):- PHP and JavaScript
- Server:- XAMPP
- Platform:- Windows
- Application:-Notepad++

2. Requirements and setting up system for PHP development

- **What we need to know:**

Designing part of the website is done with the help of Bootstrap 4.0 and CSS3 and for the database designing we use My SQL

- **What we need to have (System Requirements):**

To run Website we need a browser and to code we need application like Notepad++, atom, etc.

SOFTWARE REQUIREMENTS

1. Software Requirements

Initially we need to have a development machine that is running any of the following operating systems:

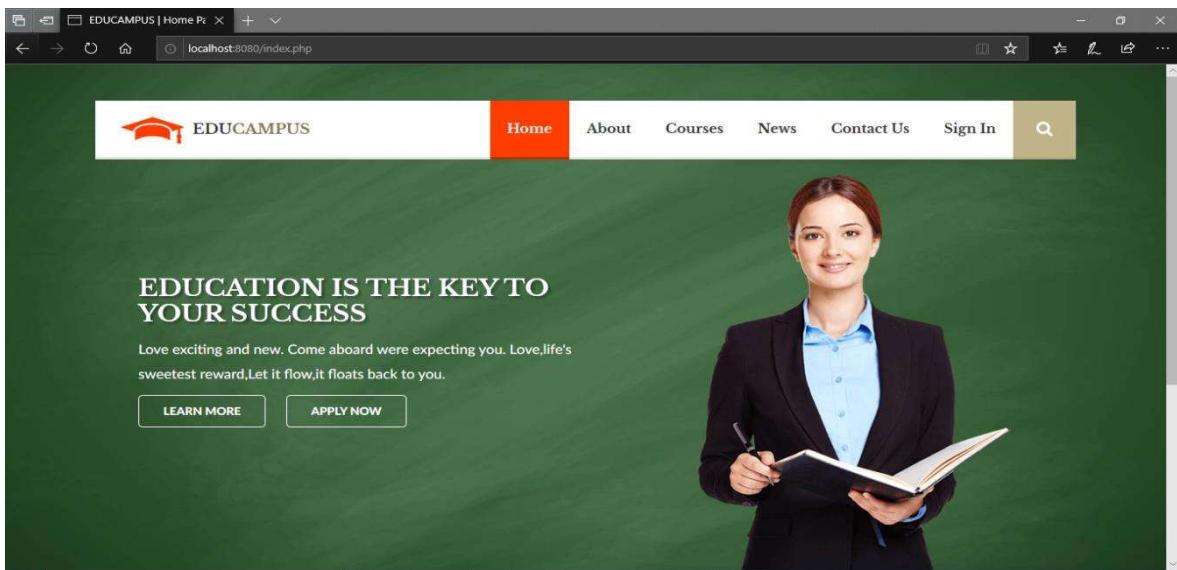
- Windows XP, Vista, Windows 7, 8
- Development environment (Notepad++).
- XAMPP server.

- 2. Software Requirement Analysis:** The software requirement specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional description, a representation of system behavior, an indication of performance requirement and design constraints appropriate validation criteria, and other information pertinent to requirement.

The introduction to software requirements specification states the goals and objectives of the software, describing it in the context of the computer based system. The Information Description provides a detailed description of the problem that the software must solve. Information content, flow and structure are documented.

SNAPSHOTS

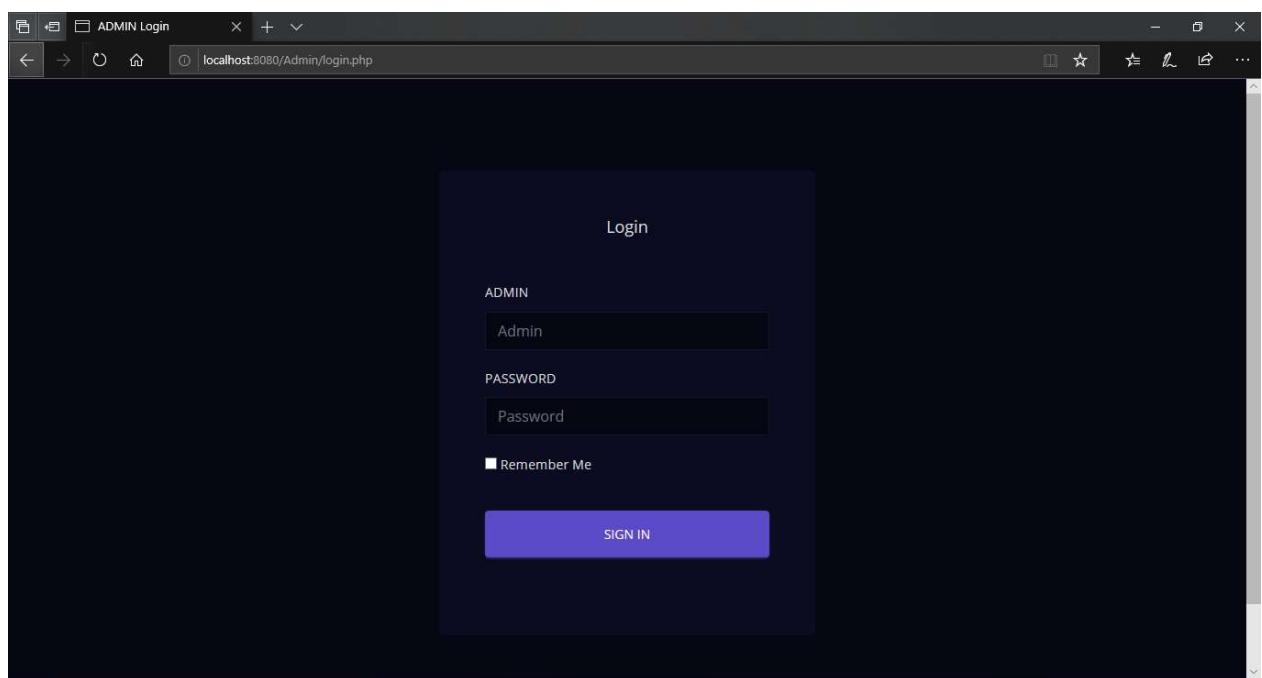
1) Home Page



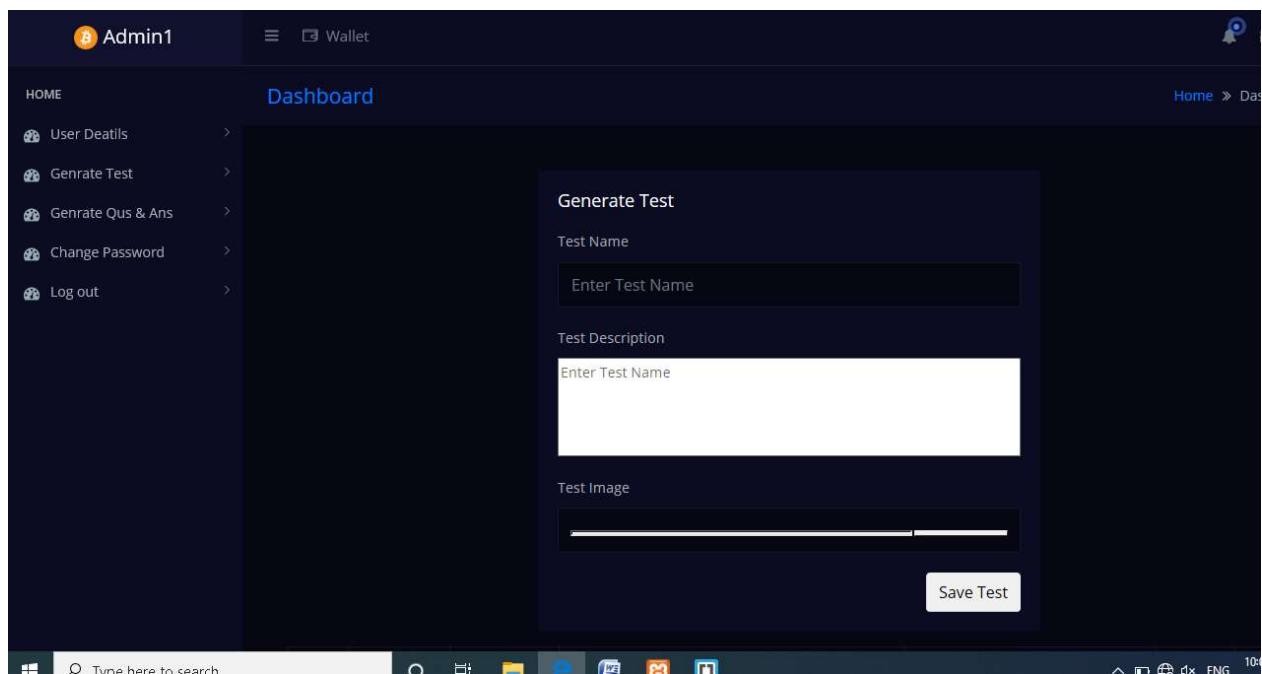
2) Registration Page

A screenshot of a web browser showing the EDUCAMPUS Sign In Page. The page has an orange header with the text "Sign In" on the left and "Home > Sign In" on the right. Below the header, there are two buttons: "Login" and "Register". The main form area contains fields for "Username" and "Password", a "Remember Me" checkbox, and a "LOG IN" button. A link for "Forgot Password?" is also present.

3) LogIn page for an Adminstrator



4) Dashboard for creating the test with Test_name and Description about the test



5) Creating Multiple Choice Questions

The screenshot shows a web interface titled "Generate Ques. & Ans." on the right side of a sidebar. The sidebar contains a "HOME" section with links: "User Details", "Generate Test", "Generate Qus & Ans", "Change Password", and "Log out". The main area has a "Select test" dropdown set to "java", a "Questions" section with a text input placeholder "Type the Question.", an "Answers" section with four options ("Option 1", "Option 2", "Option 3", "Option 4"), and a "Correct Answer" dropdown set to "1".

6) Giving the Online Test as an registered user

The screenshot shows a web page titled "EDUCAMPUS" with a navigation bar including "Home", "About", "Courses", "News", "Contact Us", "Aakash", and "Logout". The user is welcome "AAKASH". There are three questions displayed:

- Question 1:** A list of four options (1, 2, 3, 4) with the first one selected.
- Question 2: $2+2^4$:** A list of four options (1, 2, 3, 4) with the fourth one selected.
- Question 3: What would be the output of:** A list of two options (1, 2) with the first one selected.

7) Result Page after giving the online test

The screenshot shows a web browser window for the EDUCAMPUS website. The URL is `localhost:8080/result.php`. The page has a header with the EDUCAMPUS logo and navigation links for Home, About, Courses, News, Contact Us, and Logout. A user profile for "Akash" is visible. The main content area is titled "Result" and displays the following statistics:

- Total Question : 3
- No of Attempt : 3
- Correct Answer : 3
- Wrong Answer : 0

8) Variety of tests to choose from

The screenshot shows a web browser window for the EDUCAMPUS website. The URL is `localhost:8080/courses.php`. The page has a header with the EDUCAMPUS logo and navigation links for Home and Courses. The main content area is titled "Courses" and features three course cards:

- C & C++**: An image of a classroom lecture. Description: Once you start coding in C, you'll be introduced to one of the most commonly used programming languages in the world. [APPLY NOW >](#)
- .NET**: An image of three students working together. Description: Visual Basic.NET is a multi-paradigm, object-oriented programming language, implemented on the .NET Framework. [APPLY NOW >](#)
- PHP**: An image of four students in a study group. Description: PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. [APPLY NOW >](#)

WEEKLY REPORT

WEEK: 1

Description of activity, task, duty or responsibility	Performed with team	Performed Alone	Time Spent
Introduction to HTML		Yes	1 lecture
Basic Tags and HTML Lists		Yes	1 lecture
HTML Tables and Attributes		Yes	1 lecture
Introduction to Forms and Form Validation		Yes	1 lecture

List one thing that went particularly well this week (area of improvement, new task, etc.)

=>HTML overview and basics of Front-End Web Development

List one thing that was the most challenging this week (issue, problem, difficulty, etc.)

=>It's all related to basics

List one way you can improve your performance:

=>By practicing HTML

WEEK: 2

Description of activity, task, duty or responsibility	Performed with team	Performed Alone	Time Spent
CSS Basics		Yes	1 lecture
Floating		Yes	1 lecture
Styling Texts		Yes	1 lecture
Text and Fonts using Google Fonts		Yes	1 lecture

List one thing that went particularly well this week (area of improvement, new task, etc.)

=>Floating is quite interesting

List one thing that was the most challenging this week (issue, problem, difficulty, etc.)

=>There was some issue in understanding up the float property

List one way you can improve your performance:

=>By practicing CSS

WEEK: 3

Description of activity,task,duty or responsibility	Performed with team	Performed Alone	Time Spent
Introduction To Javascript Section		Yes	1 lecture
Manipulating Styles With Javascript		Yes	1 lecture
If else and Loops		Yes	1 lecture
Functions and External JS		Yes	1 lecture

List one thing that went particularly well this week (area of improvement, new task, etc.)

=>JavaScript Functions was pretty interesting topic

List one thing that was the most challenging this week (issue, problem, difficulty, etc.)

=>There were some difficulties in Control Flow in JavaScript

List one way you can improve your performance:

=>By optimizing the algorithms in JS

WEEK: 4

Description of activity,task,duty or responsibility	Performed with team	Performed Alone	Time Spent
Introduction To jQuery Section		Yes	1 lecture
Changing Website Content		Yes	1 lecture
Animating Content		Yes	1 lecture
Regular Expressions and Form Validation		Yes	1 lecture

List one thing that went particularly well this week (area of improvement, new task, etc.)

=>Animation was a pretty new concept

List one thing that was the most challenging this week (issue, problem, difficulty, etc.)

=> There were some difficulties in Draggables & Resizables

List one way you can improve your performance:

=>By analyzing AJAX

WEEK: 5

Description of activity,task,duty or responsibility	Performed with team	Performed Alone	Time Spent
Introduction To PHP		Yes	1 lecture
GET and POST variables		Yes	1 lecture
Loops in PHP		Yes	1 lecture
Introduction to phpMyAdmin		Yes	1 lecture

List one thing that went particularly well this week (area of improvement, new task, etc.)

=> phpMySQL was easy

List one thing that was the most challenging this week (issue, problem, difficulty, etc.)

=> There were some difficulties in GET and POST variables

List one way you can improve your performance:

=>By analyzing the problems in GET and POST variables

WEEK: 4

Description of activity,task,duty or responsibility	Performed with team	Performed Alone	Time Spent
Connecting and Retrieving Data from A Database		Yes	1 lecture
Inserting And Updating Data		Yes	1 lecture
Session Variables and Cookies		Yes	1 lecture
Project-eCommerce Website		Yes	1 lecture

List one thing that went particularly well this week (area of improvement, new task, etc.)

=>SQL was easy

List one thing that was the most challenging this week (issue, problem, difficulty, etc.)

=> There were some difficulties with Session Variables and Cookies

List one way you can improve your performance:

=>By practicing by making projects.

CONCLUSION AND FUTURE SCOPE

From this summer training and the project, we are able to experience a technology which is Web Development. We are confident enough to tack the real-life problems or task based on the supervised and unsupervised learning. A website is the representation of the business online. These days, every business realizes the need for having a website and are putting in efforts to design and develop the best site for taking their products or services online. This is where we can see a great deal of scope for web development and design

With the tremendous progress in the launch of websites, people who can create exemplary designs and platforms for their online presence is what businesses are searching for. Web developers and designers are bringing in all the best of their technical expertise and skills to develop and unique design websites that are and capable of pulling the crowd.

Most web developers and designers are familiar with the latest tools, techniques, technologies, and frameworks that will help them get beautiful sites up in a short span of time. The development time of websites has come down with the agile technologies that make development easier. Here, we will cover the scope and future of web development.

REFERENCES

1. Kamal Acharya. School management system project report. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172254873.34023165/v1>
2. Kamal Acharya. A CASE STUDY OF CINEMA MANAGEMENT SYSTEM PROJECT. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172254873.30191075/v1>
3. Kamal Acharya. A CASE STUDY ON ONLINE TICKET BOOKING SYSTEM PROJECT. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172254872.26972790/v1>
4. Kamal Acharya. Web chatting application project report management system. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172254871.18588592/v1>
5. Kamal Acharya. RETAIL STORE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172254871.14590154/v1>
6. Kamal Acharya. SUPERMARKET MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172252491.19145062/v1>
7. Kamal Acharya. SOCIAL MEDIA MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172252491.11210579/v1>
8. Kamal Acharya. Online music portal management system project report. Authorea. August 01, 2024. DOI: <https://doi.org/10.22541/au.172252488.89734698/v1>
9. Kamal Acharya. COLLEGE BUS MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: <https://doi.org/10.22541/au.172245277.70798942/v1>
10. Kamal Acharya. AUTOMOBILE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: <https://doi.org/10.22541/au.172245276.67982593/v1>
11. Kamal Acharya. Ludo management system project report. Authorea. July 31, 2024. DOI: <https://doi.org/10.22541/au.172243999.98091616/v1>
12. Kamal Acharya. Literature online quiz system project report. Authorea. July 31, 2024. DOI: <https://doi.org/10.22541/au.172243825.53562953/v1>
13. Kamal Acharya. Avoid waste management system project. Authorea. July 29, 2024. DOI: <https://doi.org/10.22541/au.172228528.85022205/v1>
14. Kamal Acharya. CHAT APPLICATION THROUGH CLIENT SERVER MANAGEMENT SYSTEM PROJECT. Authorea. July 29, 2024. DOI: <https://doi.org/10.22541/au.172228527.74316529/v1>
15. Acharya, Kamal, Online Job Portal Management System (May 5, 2024). Available at SSRN: <https://ssrn.com/abstract=4817534> or <http://dx.doi.org/10.2139/ssrn.4817534>

16. Acharya, Kamal, Employee leave management system. (May 7, 2024). Available at SSRN: <https://ssrn.com/abstract=4819626> or <http://dx.doi.org/10.2139/ssrn.4819626>
17. Acharya, Kamal, Online electricity billing project report. (May 7, 2024). Available at SSRN: <https://ssrn.com/abstract=4819630> or <http://dx.doi.org/10.2139/ssrn.4819630>
18. Acharya, Kamal, POLICY MANAGEMENT SYSTEM PROJECT REPORT. (December 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4831694> or <http://dx.doi.org/10.2139/ssrn.4831694>
19. Acharya, Kamal, Software testing for project report. (May 16, 2023). Available at SSRN: <https://ssrn.com/abstract=4831028> or <http://dx.doi.org/10.2139/ssrn.4831028>
20. Acharya, Kamal, ONLINE CRIME REPORTING SYSTEM PROJECT. (August 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4831015> or <http://dx.doi.org/10.2139/ssrn.4831015>
21. Acharya, Kamal, Burger ordering system project report. (October 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4832704> or <http://dx.doi.org/10.2139/ssrn.4832704>
22. Acharya, Kamal, Teachers Record Management System Project Report (December 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4833821> or <http://dx.doi.org/10.2139/ssrn.4833821>
23. Acharya, Kamal, Dairy Management System Project Report (December 20, 2020). Available at SSRN: <https://ssrn.com/abstract=4835231> or <http://dx.doi.org/10.2139/ssrn.4835231>
24. Acharya, Kamal, Electrical Shop Management System Project (December 10, 2019). Available at SSRN: <https://ssrn.com/abstract=4835238> or <http://dx.doi.org/10.2139/ssrn.4835238>
25. Acharya, Kamal, Online book store management system project report. (February 10, 2020). Available at SSRN: <https://ssrn.com/abstract=4835277> or <http://dx.doi.org/10.2139/ssrn.4835277>
26. Acharya, Kamal, Paint shop management system project report. (January 10, 2019). Available at SSRN: <https://ssrn.com/abstract=4835441> or <http://dx.doi.org/10.2139/ssrn.4835441>
27. Acharya, Kamal, Supermarket billing system project report. (August 10, 2021). Available at SSRN: <https://ssrn.com/abstract=4835474> or <http://dx.doi.org/10.2139/ssrn.4835474>
28. Acharya, Kamal, Online taxi booking system project report. (March 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4837729> or <http://dx.doi.org/10.2139/ssrn.4837729>
29. Acharya, Kamal, Online car servicing system project report. (March 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4837832> or <http://dx.doi.org/10.2139/ssrn.4837832>
30. Acharya, Kamal, School management system project report. (July 10, 2021). Available at SSRN: <https://ssrn.com/abstract=4837837> or <http://dx.doi.org/10.2139/ssrn.4837837>
31. Acharya, Kamal, Furniture Showroom Management System Project Report (March 21, 2021). Available at SSRN: <https://ssrn.com/abstract=4839422> or <http://dx.doi.org/10.2139/ssrn.4839422>
32. Acharya, Kamal, Online Vehicle Rental System Project Report (March 21, 2019). Available at SSRN: <https://ssrn.com/abstract=4839429> or <http://dx.doi.org/10.2139/ssrn.4839429>

33. Acharya, Kamal, *Fruit Shop Management System Project Report* (August 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4841048> or <http://dx.doi.org/10.2139/ssrn.4841048>
34. Acharya, Kamal, *Hall Booking Management System Project Report* (December 21, 2023). Available at SSRN: <https://ssrn.com/abstract=4841055> or <http://dx.doi.org/10.2139/ssrn.4841055>
35. Acharya, Kamal, *Lundry Management System Project Report* (October 21, 2023). Available at SSRN: <https://ssrn.com/abstract=4841059> or <http://dx.doi.org/10.2139/ssrn.4841059>
36. Acharya, Kamal, *Web Chatting Application Management System Project Report* (April 25, 2022). Available at SSRN: <https://ssrn.com/abstract=4842771> or <http://dx.doi.org/10.2139/ssrn.4842771>
37. Acharya, Kamal, *Automobile management system project report* (May 25, 2022). Available at SSRN: <https://ssrn.com/abstract=4846917> or <http://dx.doi.org/10.2139/ssrn.4846917>
38. Acharya, Kamal, *College bus management system project report* (April 25, 2023). Available at SSRN: <https://ssrn.com/abstract=4846920> or <http://dx.doi.org/10.2139/ssrn.4846920>
39. Acharya, Kamal, *Courier management system project report* (May 25, 2023). Available at SSRN: <https://ssrn.com/abstract=4846922> or <http://dx.doi.org/10.2139/ssrn.4846922>
40. Acharya, Kamal, *Event management system project report* (April 25, 2021). Available at SSRN: <https://ssrn.com/abstract=4846927> or <http://dx.doi.org/10.2139/ssrn.4846927>
41. Acharya, Kamal, *Library management system project report II* (May 25, 2020). Available at SSRN: <https://ssrn.com/abstract=4848857> or <http://dx.doi.org/10.2139/ssrn.4848857>