**4.1 Introduction to System Development**

The primary objective of development are to translate the most promising design approach into a stable, interoperable, producible and cost effective design, validate the manufacturing and production process and demonstrate system capabilities through testing.

Although much of the activities in the development addresses the computer programs that make up the system, this phase also puts in place the hardware, software and communication environment of the overall system. At the end of development, the system will be ready for activities of the integration and testing.

**4.2 Tool/Language Selection**

There are hundreds of tools available in market for development of Different kinds of software. Tools and language selection is very important in software development. Developer has to choose which tool is best for the software to be developed. Software development performance depends on tools that are used. Software performance is increased if proper tools selection is done. If appropriate tools are not selected then software performance is decreased.

Programming languages are used to develop software to solve problems of different people belonging to different fields of life. According to user requirements proper programming language selection is very important. Developer should choose that programming language which provides all features according to user requirements and which provides higher performance. Such language should choose that provides reliable solution modules for all types of users.

There are hundreds of tools available in market for development of Different kinds of software. Tools and language selection is very important in software development. Developer has to choose which tool is best for the software to be developed. Software development performance depends on tools that are used. Software performance is increased if proper tools selection is done. If appropriate tools are not selected then software performance is decreased.

Programming languages are used to develop software to solve problems of different people belonging to different fields of life. According to user requirements proper programming language selection is very important. Developer should choose that programming language which provides all features according to user requirements and which provides higher performance. Such language should choose that provides reliable solution modules for all types of users.

**4.2.1 Tools Used**

I have used following tools in my project:

* Dreamweaver
* XAMP Server

**4.2.2 Programming Languages Used**

I have used following programming languages in my project:

* PHP
* HTML, HTML5 and CSS
* JavaScript and JQuery.

**4.2.2 Features of Tools Used**

**4.2.2.1 Features of WAMP Server**

WAMP is a free, pre-compiled and pre-configured software package containing the Apache HTTP Server, MySQL database and tools to run PHP and Perl web applications. It was conceived to be fast, flexible and easy to install and run. Also, it contains a number of useful packages that make easy things like generating traffic reports, accelerating PHP content, administering MySQL database through a browser with phpMyAdmin, uploading and downloading files with ProFTPD FTP server and so on. Moreover, it will install on rather many Linux distributions, as well as on Windows and Solaris.

The WAMP package includes:

* Apache
* MySQL
* phpMyAdmin
* WAMP Control Panel

**4.2.2.1.1 Apache**

The Apache HTTP Server, colloquially called Apache, is the world's most widely-used web server software. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Apache supports a variety of features, many implemented as compiled modules which extend the core functionality. These can range from server-side programming language support to authentication schemes. A sample of other features include Secure Sockets Layer and Transport Layer Security support, a proxy module, a URL rewriter, custom log files, and filtering support.

**4.2.2.1.2MySQL**

MySQL is an open source relational database management system (RDBMS) based on Structured Query Language (SQL). It is used to store databases, tables and records of different tables. It also allows to manage relationships among all tables.

**4.2.2.1.3 PhpMyAdmin**

PhpMyAdmin is a free and open source tool written in PHP intended to handle the administration of MySQL with the use of a web browser. It can perform various tasks such as creating, modifying or deleting databases, tables, fields or rows; executing SQL statements; or managing users and permissions.

**4.2.2.1.4 WAMP Control Panel**

WAMP control panel provides GUI for managing all features of WAMP package. User can turn on or off any feature using WAMP control panel. User can install or uninstall any service. User can view log files which contain error history. User can modify php core files using WAMP control panel.

**4.2.3 More about PHP**

**PHP: Hypertext Preprocessor** is a widely used, general-purpose scripting language that was originally designed for web development to produce dynamic web pages. For this purpose, PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document. As a general-purpose programming language, PHP code processed by an interpreter application in command-line mode performing desired operating system operations and producing program output on its standard output channel. It may also function as a graphical application. PHP is available as a processor for most modern web server and as a standalone interpreter on most operating system and computing platforms.

**4.3 Hardware for the System**

Hardware required to run this system will be:

* Pentium series computer preferably Pentium iv
* VGA graphics card and color monitor.
* Windows compatible mouse, keyboard
* 1 GB RAM

**4.4 Software Development & Implementation**

The development plan is planned according to development methodology which is RAD Rapid Application Development. The architecture of is as under in fig.

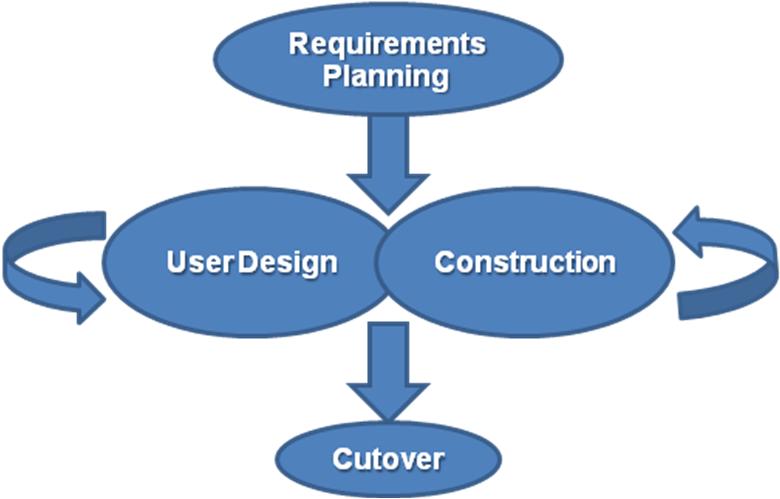


Figure 4.1 Development Plan

As show in figure it has mostly four phases.

* Requirements Planning Phase
* User Design Phase
* Construction Phase
* Cutover Phase

**4.4.1 Requirements planning phase**-combines elements of the system planning and systems analysis phases of the System Development Life Cycle (SDLC). Users, managers, and IT staff members discuss and agree on business needs, project scope, constraints, and system requirements. It ends when the team agrees on the key issues and obtains management authorization to continue.

**4.4.2 User design phase**-during this phase, users interact with systems analysts and develop models and prototypes that represent all system processes, inputs, and outputs. The RAD groups or subgroups typically use a combination of Joint Application Development (JAD) techniques and CASE tools to translate user needs into working models. User Design is a continuous interactive process that allows users to understand, modify, and eventually approve a working model of the system that meets their needs.

**4.4.3 Construction phase**-focuses on program and application development task similar to the SDLC. In RAD, however, users continue to participate and can still suggest changes or improvements as actual screens or reports are developed. Its tasks are programming and application development, coding, unit-integration and system testing.

**4.4.4 Cutover phase**-resembles the final tasks in the SDLC implementation phase, including data conversion, testing, changeover to the new system, and user training. Compared with traditional methods, the entire process is compressed. As a result, the new system is built, delivered, and placed in operation much sooner. Its tasks are data conversion, full-scale testing, system changeover, user training.

Above all are professionally using technique. I tried to follow basic things while making the software.

Implementation stage of a product defines the success and failure of product. In implementation stage product is installed in user environment or released in market. There may be some errors in product’s working in user environment. To solve those problems some modifications are made in product functionality. This modification may be in the modules of product or in the form of patches. Patches may be solution of problems occurred in product working or may be solution of new requirement of user. These patches are then installed on product in user environment to resolve errors. Modification is performed again and again until product does not work according to user needs.

Implementation stage of a product defines the success and failure of product. In implementation stage product is installed in user environment or released in market. There may be some errors in product’s working in user environment. To solve those problems some modifications are made in product functionality. This modification may be in the modules of product or in the form of patches. Patches may be solution of problems occurred in product working or may be solution of new requirement of user. These patches are then installed on product in user environment to resolve errors. Modification is performed again and again until product does not work according to user needs.

**4.4.2 Software Testing**

Testing is the process of evaluating a system or its component(s) with the intent to find that whether it satisfies the specified requirements or not. This activity results in the actual, expected and difference between their results. In simple words testing is executing a system in order to identify any gaps, errors or missing requirements in contrary to the actual desire or requirements.

According to ANSI/IEEE 1059 standard, Testing can be defined as “A process of analyzing a software item to detect the differences between existing and required conditions (that is defects/errors/bugs) and to evaluate the features of the software item”.

**4.4.2.1 Who Does Testing?**

It depends on the process and the associated stakeholders of the project. In the IT industry, large companies have a team with responsibilities to evaluate the developed software in the context of the given requirements. Moreover, developers also conduct testing which is called Unit Testing. In most cases, following professionals are involved in testing of a system within their respective capacities:

* Software Tester
* Software Developer
* Project Lead/Manager
* End User

Different companies have difference designations for people who test the software on the basis of their experience and knowledge such as Software Tester, Software Quality Assurance Engineer, and QA Analyst etc.

**4.4.2.2 Testing Types**

There are hundreds of testing types according to the nature of software. Few testing types are mentioned here.

**4.4.2.2.1 Unit Testing**

Software verification and validation method in which a programmer tests whether individual units of source code are fit for use or not is known as unit testing. It is usually conducted by the development team. It is done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code.

**4.4.2.2.2 Integration Testing**

Integration testing is a software testing in which individual software modules are combined and tested as a group. Testing of integrated modules to verify combined functionality after integration. Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems. It is usually conducted by testing teams.

**4.4.2.2.3 Acceptance Testing**

It is formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system. The aim of validation is to ensure that the product is made according to the requirements of the client, and fulfills the intended purpose. It is usually performed by the customer.

**4.4.2.2.4 Recovery Testing**

Recovery testing is basically done in order to check how fast and better the application can recover against any type of crash or hardware failure etc. Type or extent of recovery is specified in the requirement specifications. It is basically testing how well a system recovers from crashes, hardware failures or other catastrophic problems. It is performed by the testing teams.

**4.4.2.2.5 Security Testing**

A process to determine that an information system protects data and maintains functionality as intended. Can system be penetrated by any hacking way? Testing how well the system protects against unauthorized internal or external access. It is checked whether system, database is safe from external attacks or not. It can be performed by testing teams or by specialized security-testing companies.

**4.4.2.2.6 White Box Testing**

This testing is based on knowledge of the internal logic of an application’s code. Also known as Glass box Testing. Internal software and code working should be known for this type of testing. Tests are based on coverage of code statements, branches, paths, conditions. It is performed by software developers.

**4.4.2.2.7 Black Box Testing**

A method of software testing that verifies the functionality of an application without having specific knowledge of the application's code/internal structure. Typically, when performing a black box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.Tests are based on requirements and functionality. It is performed by QA teams.

**4.4.2.2.8 Gray Box Testing**

A combination of Black Box and White Box testing methodologies is known as Gray Box Testing. In this testing a piece of software against its specification is tested but using some knowledge of its internal workings. It can be performed by either development or testing teams.

**4.4.2.2.9 End-to-End Testing**

Similar to system testing, involves testing of a complete application environment in a situation that mimics real-world use, such as interacting with a database, using network communications, or interacting with other hardware, applications, or systems if appropriate. It is performed by QA teams.

**4.5 Code/Algorithms of important modules**

Main module of my project is **Get Student Admission**

**Code**

<?php

error\_reporting(0);

ini\_set('display\_errors',0);

ob\_start();

?>

<?php

include("db.php");

$a\_id="";

$s\_name="";

$f\_name="";

$gender="";

$cnic="";

$class="";

$roll\_no="";

$section="";

$session="";

$btVal="Add Student";

$ed="";

if(isset($\_GET['a\_id']))

{

$a\_id=$\_GET['a\_id'];

$btVal="Update";

$ed="true";

}

if(isset($\_GET['s\_name']))

{

$s\_name=$\_GET['s\_name'];

}

if(isset($\_GET['f\_name']))

{

$f\_name=$\_GET['f\_name'];

}

if(isset($\_GET['fath\_name']))

{

$fath\_name=$\_GET['fath\_name'];

}

if(isset($\_GET['gender']))

{

$gender=$\_GET['gender'];

}

if(isset($\_GET['cnic']))

{

$cnic=$\_GET['cnic'];

}

if(isset($\_GET['class']))

{

$class=$\_GET['class'];

}

if(isset($\_GET['roll\_no']))

{

$roll\_no=$\_GET['roll\_no'];

}

if(isset($\_GET['section']))

{

$section=$\_GET['section'];

}

if(isset($\_GET['session']))

{

$session=$\_GET['session'];

}

$submit=$\_POST['submit'];

$a\_id=$\_GET['a\_id'];

$ed=$\_GET['ed'];

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>New Student</title>

<style type="text/css">

<!--

body {

margin-top: 0px;

}

-->

</style>

<link href="style.css" rel="stylesheet"/>

</head>

<body background="logos/bg\_blue.png">

<table width="664" height="342" border="0" align="center" cellpadding="0" cellspacing="0">

<tr align="center">

<td width="664" colspan="4" >

<?php include('head.php'); ?>

</td>

</tr>

<tr>

<?php include('navi\_main.html'); ?>

</tr>

<!--

<tr bgcolor="#FFFFFF">

<td height="26" colspan="4" align="left"><font color="#006699">Main Page > User > New User </font></td>

</tr>

-->

<tr bgcolor="#FFFFFF">

<td height="26" colspan="4" align="center"><font size="+2" color="#000000"><b>Enter Student's Data</b></font></td>

</tr>

<tr bgcolor="#FFFFFF" align="center">

<?php include('navi\_result.html'); ?>

</tr>

<tr bgcolor="#FFFFFF"><td height="257">

<table width="567" border="0" align="center" bgcolor="#CCCCCC" class="mini\_tab">

<tr>

<td colspan="4" align="right" bgcolor="#33CC99"><font color="#FF0000">\* Mandatory Fields</font></td>

</tr>

<form method="post" action="add\_stud.php?a\_id=<?php echo $a\_id; ?>&ed=<?php echo $ed; ?>" name="stad\_form" enctype="multipart/form-data" onsubmit="return validateForm()">

<tr>

<td width="127" height="24" class="font">&nbsp;&nbsp;Student Name</td>

<td width="144" height="24"><input type="text" value="<?php echo $s\_name; ?>" name="s\_name" class="sub\_input"/></td>

<td width="134" height="24" >&nbsp;&nbsp;Father's Name:</td>

<td width="144" height="24"><input type="text" value="<?php echo $f\_name; ?>" name="f\_name" class="sub\_input" /></td>

</tr>

<tr>

<td width="127" height="24" class="font">&nbsp;&nbsp;Roll No</td>

<td width="144" height="24"><input type="text" value="<?php echo $roll\_no; ?>" name="roll\_no" class="sub\_input"/></td>

<td width="134" height="24" class="font">&nbsp;&nbsp;Father's CNIC:</td>

<td width="144" height="24"><input type="text" value="<?php echo $cnic; ?>" name="cnic" class="sub\_input" /><br />

<font color="#FF0000">eg: 3630255913203</font></td>

</tr>

<tr>

<td width="127" height="24" class="font">&nbsp;&nbsp;Gender</td>

<td width="144" height="24"><input type="radio" name="gender" value="Male"<?php if($gender=="Male") echo "selected";?> /> Male <input type="radio" name="gender" value="Female"<?php if($gender=="Female") echo "selected";?> /> Female </td>

<td width="134" height="24">&nbsp;&nbsp;Class </td>

<td width="144" height="24"><select name="class" class="sub\_input">

<?php

$sql="select \* from class\_res order by C\_id ASC";

$resultss=mysql\_query($sql)or die(mysql\_error());

while($row=mysql\_fetch\_array($resultss))

{

$cname=$row['Class'];

$C\_id=$row['C\_id'];

?>

<option value="<?php echo $cname;?>"<?php if($class=="$cname") echo "selected";?>>

<?php

echo $cname;

?>

</option><?php } ?>

</select></td>

</tr>

<tr>

<td width="134" height="26">&nbsp;&nbsp;Session: </td>

<td width="144" height="26"><input type="text" value="<?php echo $session; ?>" name="session" class="sub\_input"/></td>

</tr>

<tr align="center">

<td colspan="4"><input type="submit" value="<?php echo $btVal; ?>" name="submit" class="loginButton" />

<input type="hidden" name="ed" value="<?php echo $ed;?>" />

<input type="hidden" name="a\_id" value="<?php echo $a\_id;?>" />

<input type="hidden" name="submit" value="<?php echo $submit; ?>" /></td>

</tr></form>

<tr>

<td colspan="4">&nbsp;</td>

</tr>

</table>

</td></tr>

<tr bgcolor="#FFFFFF">

<td>&nbsp;</td>

</tr>

</table>

<script type="text/javascript">

function validateForm(frm)

{

var s\_name=document.forms["stad\_form"]["s\_name"].value

var roll\_no=document.forms["stad\_form"]["roll\_no"].value

var cnic=document.forms["stad\_form"]["cnic"].value

var session=document.forms["stad\_form"]["session"].value

if ((s\_name==null) || (s\_name==""))

{

alert("Please enter Student Name");

return false;

//}

if (s\_name!="")

{

var filter=/^[a-zA-Z \_]+$/;

var test\_bool = filter.test(s\_name);

if(test\_bool==false)

{

alert('Please enter only alphabets in Student Name');

return false;

}

if (trim(s\_name)== "")

{

alert('Please enter Student Name in correct format');

return false;

}

} }

if (roll\_no == "" || roll\_no == "null")

{

alert("Please enter Roll No#");

return false;

}

if (cnic == "" || cnic == "null")

{

alert("Please enter CNIC #");

return false;

}

if (session == "" || session == "null")

{

alert("Please enter Session");

return false;

}

}

</script>

</body>

</html>