LEARNING MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted by

AHAJAKANYA S

SANGHEETH SAHANA

SEETHALAKSHMI K

DIVYA R

Partial fulfilment of the requirements Internship

Submitted to:

UI DEVELOPER INTERNSHIP,

SOFTECH SYSTEM AND SOLUTION

BONAFIDE CERTIFICATE

Certified that this project report "**LEARNING MANAGEMENT SYSTEM** " is the bonafide work of " **S.AHAJAKANYA**, **D.SANGHEETH SAHANA**, K.**SEETHALAKSHMI**, **R.DIVYA** "who carried out the project work under my supervision.

SIGNATURE

M. Kannan ,

Professional Software Developer ,

Softech System & Solution,

Salem- 636009.

Submitted for the Project Work Viva-Voce Examination held on

_____

INTERNAL EXAMINER

ACKNOWLEDGEMENT

Working in the Softech system & solutions for Learning Management System was interesting , we learnt a lot on this project, especially on styling languages like CSS , Bootstrap etc.

We express our gratitude to our beloved Professor, **Kannan .M** for giving us a chance to complete our Internship in one of the reputed government institutions running under her magnificent leadership.

It is a great opportunity to express our sincere thanks to our parents, friends and all the people who have contributed to the successful completion of our project work through their support, encouragement and guidance.

ABSTRACT

In this paper, we propose a software application for the administration, tracking, reporting and delivery of educational courses or training programs. They help the instructor deliver material to the students and other assignments, track students progress and manage record- keeping. LMSs are focused on online learning delivery but support a range of uses, acting as a platform for fully online courses, as well as several hybrid forms, such as blended learning and flipped classrooms. LMSs can be complemented by learning technologies such as a training management system to manage instructor-led training or a Learning Record Store to store and track learning data. The intent of LMS was to enable administrators and mentors to manage the learning process . LMS use computer networks as a delivery mechanism and they allow students to take courses anywhere and anytime . Based on the cognitive science and artificial intelligence , they have proven their worth in multiple ways in multiple domains in education. Compared to existing system, we prove that the proposed system achieves the works excellent for online course management ,delivery, and tracking for formal learning.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER -1

INTRODUCTION

Learning and development is crucial to the success of all businesses, which is why so many companies invest in learning management systems. A learning management system (LMS) is a web-based platform that is designed to deliver and manage training content and resources. An LMS allows for easy reporting and tracking, ensuring talent managers have a good understanding of their workforces' progress. It provides a full and easy-to-access record of employee training. Here we will provide an explanation of learning management systems and how they are used in a corporate environment. We will outline the benefits, so you can see why an LMS is a worthwhile investment for your company.

An LMS helps companies to keep a record of any e-learning and training courses their employees have completed. Learning management systems also allow for testing to be carried out and can help managers produce effective training plans. More comprehensive learning management systems allow businesses to manage everything from competency to skills-gap analysis, succession planning and resource allocation. The type of LMS you invest in will depend on your learning and development requirements, as well as your budget.

Learning management systems (LMS) can be defined as "a suite of services designed to deliver, track, report on and administer learning content, student progress and student interactions". Information and communication technologies (ICT) such as Internet and the Web provide the perfect framework for current LMS. They are mostly based on courseware tools either commercial or coming from university research groups. The current work proposes a LMS model that is based on standard and open information technologies such as HTML, CSS[Bootstrap] and JavaScript. These documents have been used to represent multiple kinds of information and their application is particularly important in educational contexts. Before publishing test questions it is customary to get

it reviewed by admin. After going through its content either it gets approved or gets rejected. Modify the profile of other users registered in the system. Change user status from inactive to active.

EXPANSION OF LMS:

- **Learning**: An LMS handles the learning material delivery, administration, automation, and analytics. All of this is done for the purpose of one simple goal: fulfilling learning goals.
- **Management**: LMS holds an important role in documentation and database management. Without well-executed management of information, an LMS is doomed to fail.
- **System**: This refers to the LMS system as a whole that handles all the incoming and outgoing learning information within a single ecosystem. These three terms describe LMS in a very rudimentary manner. But, what's most important to take away is that an LMS is defined by its ability to facilitate learning.

1.2 FUNDAMENTALS OF LMS

An LMS, simply put, is a learning platform designed to enable companies to store and track the use of online training resources in order to make learning experiences more efficient, both for managers and employees. All information, such as test scores or time spent training, can be easily accessed to support and motivate learners who are participating in their organization's training scheme. Not only does an LMS allow learners to organize and access what can be an enormous amount of information all in one central place, but it can also help managers to confidently oversee and monitor their teams. It's one of the most essential tools for when you're ready to take your employees on the enriching journey of learning and upgrading their skills.

The four LMS fundamentals you need to know for LMS success are:

- Mining your LMS data goldmine
- Effectively engaging today's users
- Getting ahead on competency training

- Streamlining your administrative processes

1.2.1 FUNDAMENTAL 1: LMS DATA GOLDMINE

The system is continually collecting a wealth of information about your users and their activity. You can use that information to improve decision-making by taking a data-driven approach. As a result, you can achieve increased revenue, higher learner engagement, a better user experience, and an improved organizational process. Those are just a few possibilities.

Some basic types of information include:

- Learner name
- Email address
- Location
- Course name
- Grades

1.2.2 FUNDAMENTAL 2: ENGAGING OVERSTIMULATED USERS

The timeless method that really works is creating intrinsic motivation. Intrinsic motivation is built on three factors, which lead to better performance and personal satisfaction:

- Autonomy - desire to be self-directed
- Mastery - getting better (and being able to objectively gauge progress!)
- Purpose - having a challenge, plus mastery, reminds us of our purpose

Applying intrinsic motivation to your LMS will increase user engagement at a fundamental level.

1.2.3 FUNDAMENTAL 3: GETTING AHEAD ON COMPETENCY TRAINING

- Competency-based training focuses on learning concrete skills required to do their job, such as industry safety standards.
- Compliance training is focused on educating employees on laws, regulations, and company policies that apply to their daily job responsibilities. It helps employees-and your organization-avoid violations that could lead to legal liability and expensive fines. Compliance training often requires proof that your employees have
- completed the training.

1.2.4 FUNDAMENTAL 4: STREAMLINING ADMINISTRATIVE PROCESSES

Learning new LMS feature can take time, and it's tempting to avoid it because you're so busy. But streamlining your LMS administration is an essential component of successful LMS implementation. By learning about the time-saving features in your LMS, you can reduce your busyness and your tedious workload-for good. Whether you're managing users, creating courses, or ensuring your LMS is working effectively, you're probably encountering several administrative challenges that take more time than necessary. Common challenges include data entry, tedious maintenance tasks, and human error. But you can save time on these tasks by setting up integrations and troubleshooting.

1.3 ROLE OF LMS

In a nutshell, a well-built LMS can accomplish tasks such as:

- Delivery and tracking of assignments, quizzes, exams, and more.

- Managing of social learning forums, discussion boards, and other communication methods.
- Student progress tracking and learning analytics.
- Automatized sending of reminders to students.
- Extensive reporting based on learning goals.
- Delivery of both asynchronous and synchronous online course content.
- Automatized learning material recommendations based on student's profiles and skill levels.

1.4 BASIC FEATURES

- Create and update training modules
- Importing lessons and students
- Manage users and control their access levels
- Quizzes, activities, and assessments
- Report generation: assessment scores, student progress, and engagement
- Course calendar and due dates
- Certificates
- Drip content and module delivery
- Email or SMS reminders
- Social Learning features

1.5 OBJECTIVE

- Reduce time consumption.
- To maintain students education with quality.
- Automated Performance, Progress, and Schedule reports.
- Online Interaction between teachers and students.
- Centralized Database management.
- Easy operations for the user of the system.

CHAPTER – 2

SYSTEM SPECIFICATION

2.1 HARDWARE REQUIREMENTS :

- Processor : Pentium-V
- RAM : 8 GB
- Hard Disk :

2.2 SOFTWARE REQUIREMENTS:

- Operating System : Windows 10
- Language :

    Front End: HTML5, CSS3(Bootstrap 4), JavaScript

    Back End: C# (ASP.NET Framework), MySQL database

CHAPTER – 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing system is manual entry of up keeping of the details of the persons who are registered already. It is difficult for the students from far distance to reach the exam centre. This system is required to prepare registration to print a lot of number manually. The details of these students in a month by hand is very difficult. All the details of the student are maintained in a single record. So, searching and upgrading the details is a tedious task. Also, there is a chance of errors.

3.1.1 DISADVANTAGES OF EXISTING SYSTEM

- Need of extra manual effort.
- As current system is standalone normal employees cannot track their employment status.
- It is used to take much time to find any employee.
- Not very much accurate.
- System is compatible up to windows XP.
- Danger of losing the files in some cases.
- Certain required report is not available.
- It is much complex to interact with existing LMS.

3.2 PROPOSED SYSTEM

The purpose of learning management system is to automate existing manual system by the help of internet enabled computerized equipment's and full-fledged website, which is easily accessible. So, it could be easily accessed by any one whether student or faculty on anytime anywhere through the internet Connection. This reduces the most of the paper work as example instead of paper used in time tables, circulars, announcement; Now through website it could be easily accessed by the students.

3.2.1 MERITS OF PROPOSED SYSTEM

- Provide the information of student, quiz, assignments etc.
- Manage the information of student.
- Show the description and information of the course and faculty.
- It increases the efficiency of managing the course for student.
- It automates the existing manual system by the help of interest.
- Provide searching facilities & reduces paperwork.

CHAPTER-4

SOFTWARE DESCRIPTION

TECHNOLOGIES USED IN LMS

- HTML: Page layout has been designed in HTML.
- CSS: CSS page has been used for all the designing part.
- JavaScript: All the validations task and animation has been developed by JavaScript.
- SQL Server: SQL Server database has been used as a database for the project.
- Asp.Net: Used for Server Site.
- Bootstrap: For Form Designing.
- C#: Used for calling events and actions.

4.1 HTML5

HTML5 is the latest specification of the HTML language, and represented a major break with previous markup practices. The purpose of the profound changes to the language was to standardize the many new ways in which developers were using it, as well as to encourage a single set of best practices with regards to web development.

Earlier the version that was in use was HTML4 which was too complicated and it was not a child's play to design a perfect interactive website using it. But with the introduction of HTML5 now new opportunities arose and made the life of coders better. It has made the web life much more flexible. All the new features that have been introduced does not include JS. These inventions have changed the perspective of coders

towards HTML.HTML5 is introduced with new features. This version was released in October of 2014.

- Encouraging semantic (meaningful) mark up
- Separating design from content
- Promoting accessibility and design responsiveness
- Reducing the overlap between HTML, CSS, and JavaScript
- Supporting rich media experiences while eliminating the need for plugins.

4.1.1 NEW FEATURES OF HTML5

Video and Audio: Video and audio are the new tags which allow to embed a video in the website. YouTube also declare video embed by giving the code to embed for their videos. It helps the web to be more involved with multimedia. A new tag is also available in HTML5 and that is audio tag. Which is used to embed any audio in the web.

Nav: The nav element is used for the part of a internet site that links to different pages at the website. The hyperlinks can be organized a number of approaches. below, the hyperlinks are displayed inside paragraph factors. An unordered list can also be used.

Example:

```
<h1>HTML TAG</h1>
<nav>
<a href =”/html”>HTML</a>
<a href =”/css”>CSS</a>
</nav>
```

Canvass: canvas is a tag of HTML which is newly introduced in HTML5. It is used to draw the images on the fly. It can be used for visual images, rendering graphs, game graphics.

Example:

```
<!DOCTYPE HTML>
<html>
  <head>
    <style>
      #mycanvas{border:1px solid red;}
    </style>
  </head>
  <body>
    <canvas id = "mycanvas" width = "100" height = "100"></canvas>
  </body>
</html>
```

Figure and Fig caption: Earlier there was no way to of figure as well as give caption to that figure. But, with the introduction of figure as well as fig caption, it has become semantically possible to insert an image in a page with its caption.

Example:

```
<figure>
   <img src="image/image-1.jpg" alt="About ADMEC" />
      <figcaption>
         <p>This is our institute </p>
      </figcaption>
</figure>
```

Geolocation: HTML5 Geolocation API lets you share your location with your favourite web sites. A JavaScript can capture your latitude and longitude and can be sent to backed web server and do fancy location-aware things like finding local businesses or showing your location on a map.

4.2 CSS3

CSS3 is collaboration of CSS2 specifications and new specifications, we can called this collaboration is module.

The benefit of this functionality is that it allows the specification to be finalized and accept faster, as segments are finalized and accepted in portions. Also, this allows the browser to support segments of the specification.

4.2.1NEW FEATURES OF CSS3

Visibility: A property called *visibility* allows you to hide an element from view. You can use this property along with JavaScript to create very complex menu and very complex webpage layouts.

Values :-

Visible: The box and its contents are shown to the user.

Hidden: The box and its content are invisible to the user.

Collapse: This is for use only with dynamic table columns and row effects.

Pseudo classes: CSS pseudo-classes are used to add special effects to some selectors. You do not need to use JavaScript or any other script to use those effects. A simple syntax of pseudo-classes is as follows −

selector: pseudo-class {property: value}

Pseudo Elements: CSS pseudo-elements are used to add special effects to some selectors. You do not need to use JavaScript or any other script to use those effects. A simple syntax of pseudo-element is as follows −

selector: pseudo-element {property: value}

4.2.2 ADVANTAGES OF CSS3

CSS3 is the latest version of CSS. It holds many advantages over its predecessor. CSS3 started to gain popularity in early 2010. At that time, some of the browsers did not even use to support CSS3 and so it took a while for the web developer community to start implementing this in their projects. Here, we will see the advantages of CSS3 over CSS, and also discuss why CSS3 gained such huge popularity in a short amount of time.

- **Responsiveness:** CSS3 inherently supports responsive design, and is equipped to handle media queries. Media queries help in making a website responsive as we can apply personalized CSS properties for different screen widths. Thus, the website looks good in every device it is accessed from.
- **Colors:** New color formats like RGBA (Red, Green, Blue, Alpha), HSL (Hue, Saturation, Lightness), HSLA (Hue, Saturation, Lightness, Alpha) were added. It helps the web designers as it helps them in applying styling and different color effects into their pages. The gradient and opacity properties were also added.
- **Reduced the Alignment Problems:** The box-sizing has fixed some annoying alignment problems as now the developers can give appropriate size to the divisions using properties like padding, margin, etc. It helps in creating complex web page structures. The CSS grid helps in creating responsive web pages without using the table rows and columns. Thus, the alignment problems were solved and making the websites mobile friendly became easier.
- **Animations:** CSS requires the developer to create animation using a scripting language. On the other hand, CSS3 introduced animation features like text-shadow. It reduced the workload of the developers as applying such animations were

difficult earlier but with the introduction of these new CSS properties, it becomes much easier.

- **Compatibility:** As already discussed earlier, at the time of its release, CSS3 was not supported by many browsers and thus it took a while for the developers to use to get a hold of this. But nowadays, every modern browser supports CSS3 and that's the reason why every website that is built nowadays uses CSS3 instead of vanilla CSS.
- **JavaScript Independence Faster Loading:** As already discussed above, earlier the developers had to mix JavaScript and CSS to provide even simple decorations like text-shadow, etc. With the introduction of CSS3, the dependence of CSS on JS got reduced, as a result of which, the loading time of the web pages decreased.
- **Testing:** Earlier versions of CSS were a bit complex when it came to the testing part. However, CSS3 has been divided into smaller modules which makes it easier for the user to run compatibility tests and test the parts individually. Therefore, bug detection becomes easier and it saves a lot of time and energy for the hassle.
- **Platform Friendly:** CSS3 is compatible with all the platforms or devices like mobile phones, tablets, etc.
- **Border Radius:** The border-radius CSS property helps in making the borders of the images and the div elements rounded in shape, an effect that required a lot of formatting and photoshop work to achieve before. Another biggie is its support of rounded image corners. Other properties like shadowing have also been added.

4.3 JAVASCRIPT

JavaScript, also called JS script or JS, is a text-based programming language used to create interactive elements on a website. The language can be used for both client-side and server-side scripting. JavaScript is also used for developing mobile apps and online games. JavaScript is one of the three primary mark up and programming languages used in web development.

4.3.1 ADVANTAGES OF JAVASCRIPT:

- Regardless of where you host JavaScript, it always gets executed on client environment to save lots of a bandwidth and make execution process fast.
- In JavaScript, XML Http Request is an important object that was designed by Microsoft. The object call made by XMLHttpRequest as an asynchronous HTTP request to the server to transfer the data to both sides without reloading the page
- The biggest advantage to JavaScript having an ability to support all modern browsers and produce an equivalent result.
- Global companies support community development by creating projects that are important. An example is Google (created Angular framework) or Facebook (created the React.js framework).
- JavaScript is employed everywhere on the web.
- JavaScript plays nicely with other languages and may be utilized in an enormous sort of applications.

4.4 ASP.NET 4.5:

.NET is a developer platform made up of tools, programming languages, and libraries for building many different types of applications. The base platform provides components that apply to all different types of apps. Additional frameworks, such as ASP.NET, extend .NET with components for building specific types of apps. Here are some things included in the .NET platform:

- The **C#,** F**#,** and Visual Basic programming languages
- Base libraries for working with strings, dates, files/IO, and more

  Editors and tools for Windows, Linux, macOS, and Docker

4.4.1 FEATURES

- Enable Trace feature
- Strongly Typed Data Controls

- Performance Tuning in ASP.Net
- ASP.NET Web API
- Support for Web Sockets protocol
- Support for improved paging in ASP.NET 4.5 Grid View control
- Enhanced support for asynchronous programming
- Support for HTML5 form types.

The ASP.NET application codes can be written in any of the following languages: C#, Visual Basic.Net, Jscript, J#

ASP.NET is used to produce interactive, data-driven web applications over the internet. It consists of a large number of controls such as text boxes, buttons, and labels for assembling, configuring, and manipulating code to create HTML pages.

4.5 ASP.NET CORE

ASP.NET Core is the new web framework from Microsoft. It has been redesigned from the ground up to be fast, flexible, modern, and work across different platforms. Moving forward, ASP.NET Core is the framework that can be used for web development with .NET. If you have any experience with MVC or Web API over the last few years, you will notice some familiar features. At the end this tutorial, you will have everything you need to start using ASP.NET Core and write an application that can create, edit, and view data from a database.

4.5.1 ADVANTAGES OF .NET CORE

ASP.NET Core comes with the following advantages –

- ASP.NET Core has a number of architectural changes that result in a much leaner and modular framework.
- ASP.NET Core is no longer based on System.Web.dll. It is based on a set of granular and well factored NuGet packages.

- This allows you to optimize your app to include just the NuGet packages you need.
- The benefits of a smaller app surface area include tighter security, reduced servicing, improved performance, and decreased costs With ASP.NET Core, you can get the following improvements –
- Build and run cross-platform ASP.NET apps on Windows, Mac and Linux.
- Built on .NET Core, which supports true side-by-side app versioning.
- New tooling that simplifies modern Web development.
- Single aligned web stack for Web UI and Web APIs.
- Cloud-ready environment-based configuration.
- Built-in support for dependency injection.
- Tag Helpers which make Razor mark up more natural with HTML.
- Ability to host on IIS or self-host in your own process.

4.6 ASP.NET MVC

ASP.NET MVC is basically a web development framework from Microsoft, which combines the features of MVC (Model-View-Controller) architecture, the most up-to-date ideas and techniques from Agile development, and the best parts of the existing ASP.NET platform.

4.6.1 BENEFITS OF ASP.NET MVC

Following are the benefits of using ASP.NET MVC –

- Makes it easier to manage complexity by dividing an application into the model, the view, and the controller.
- Enables full control over the rendered HTML and provides a clean separation of concerns.

- Direct control over HTML also means better accessibility for implementing compliance with evolving Web standards.
- Facilitates adding more interactivity and responsiveness to existing apps.
- Provides better support for test-driven development (TDD).
- Works well for Web applications that are supported by large teams of developers and for Web designers who need a high degree of control over the application behaviour.

CHAPTER-5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

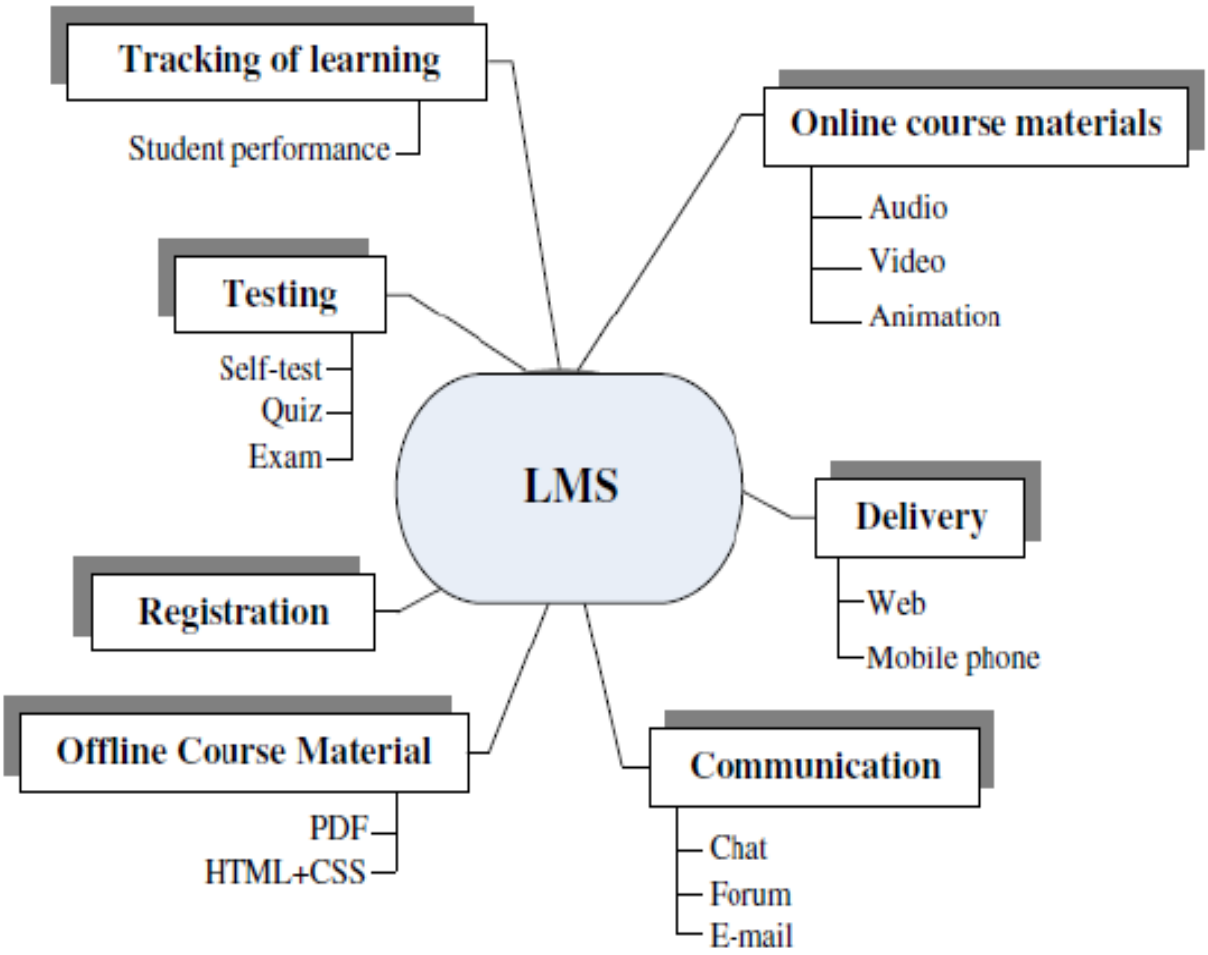


Fig 5.1 SYSTEM ARCHITECTURE

5.2UML DIAGRAMS

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows: Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.

- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modelling language.
- Encourage the growth of OO tools market.

- Support higher level development concepts such as collaborations, frameworks, patterns and components.
- Integrate best practices.

5.2.1 USE CASE DIAGRAM:

A use case diagram in the Unified Modelling Language (UML) is a type of behavioural diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

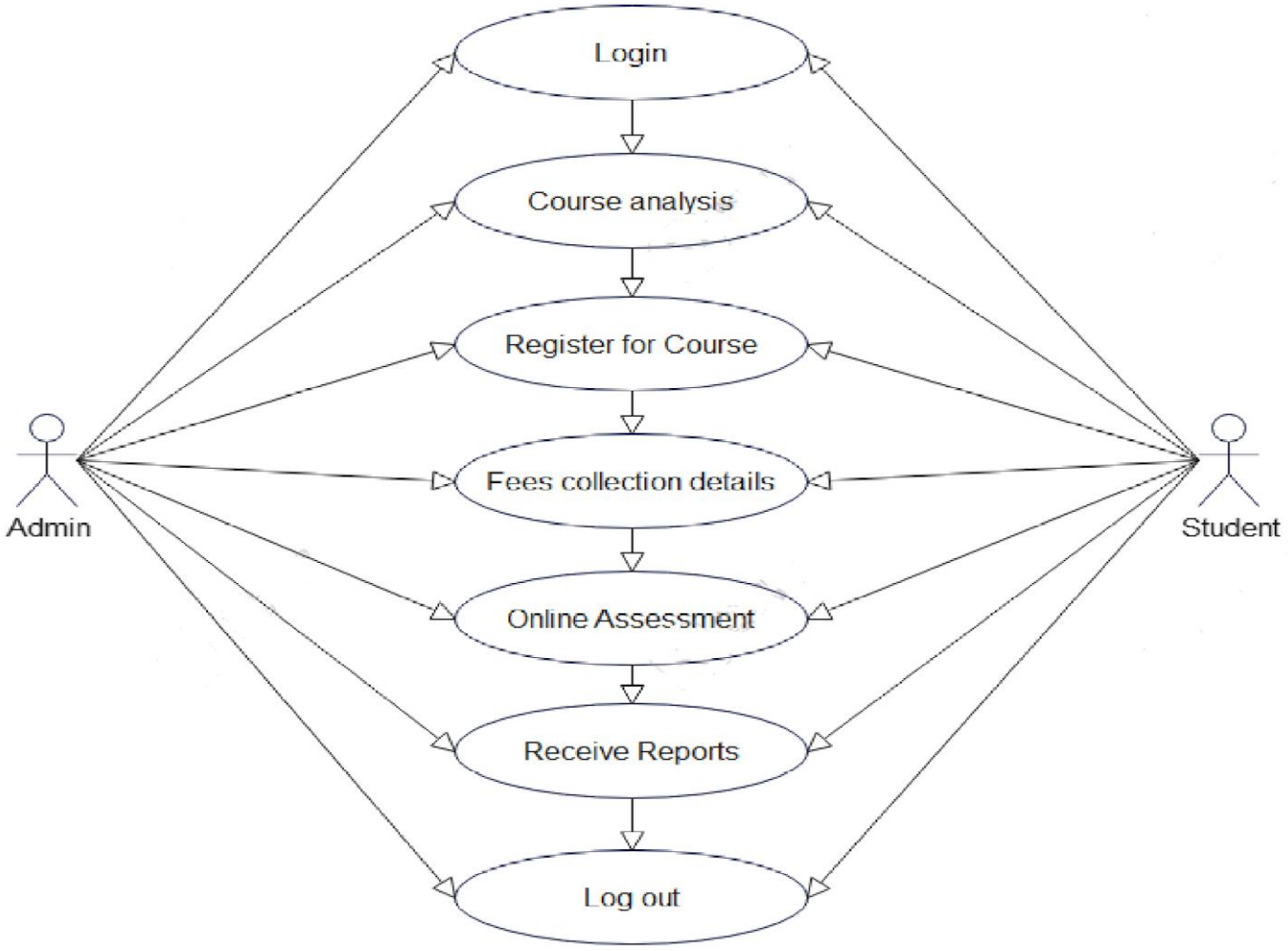


Fig 5.2.1 Use Case Diagram

5.2.2COMPONENT DIAGRAM

A component diagram is used to break down a large object-oriented system into the smaller components, so as to make them more manageable. It models the physical view of a system such as executables, files, libraries, etc. that resides within the node.

It visualizes the relationships as well as the organization between the components present in the system. It helps in forming an executable system. A component is a single unit of the system, which is replaceable and executable. The implementation details of a component are hidden, and it necessitates an interface to execute a function. It is like a black box whose behaviour is explained by the provided and required interfaces.

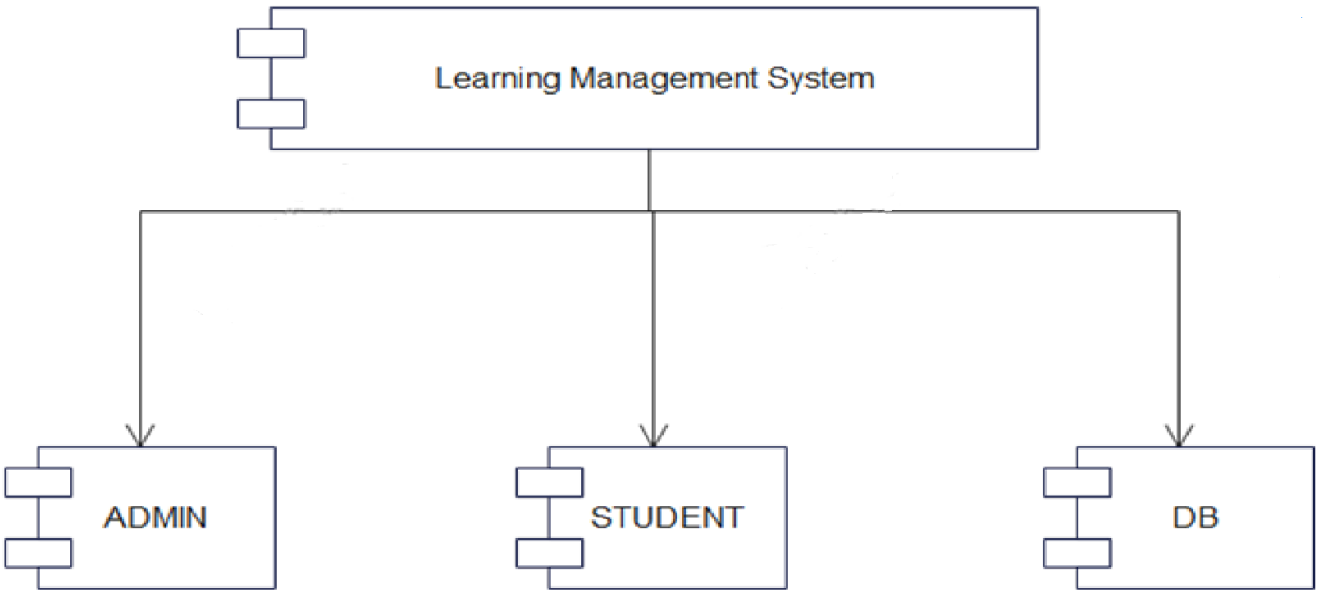


Fig 5.3.1 Component Diagram

5.3 DATA FLOW DIAGRAM:

- The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
- The data flow diagram (DFD) is one of the most important modelling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
- DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
- DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

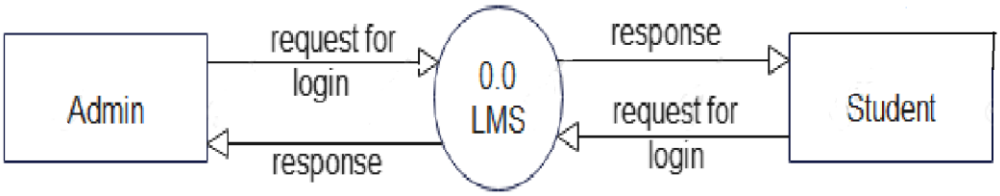


Fig 5.3.1 Dataflow 0th Level

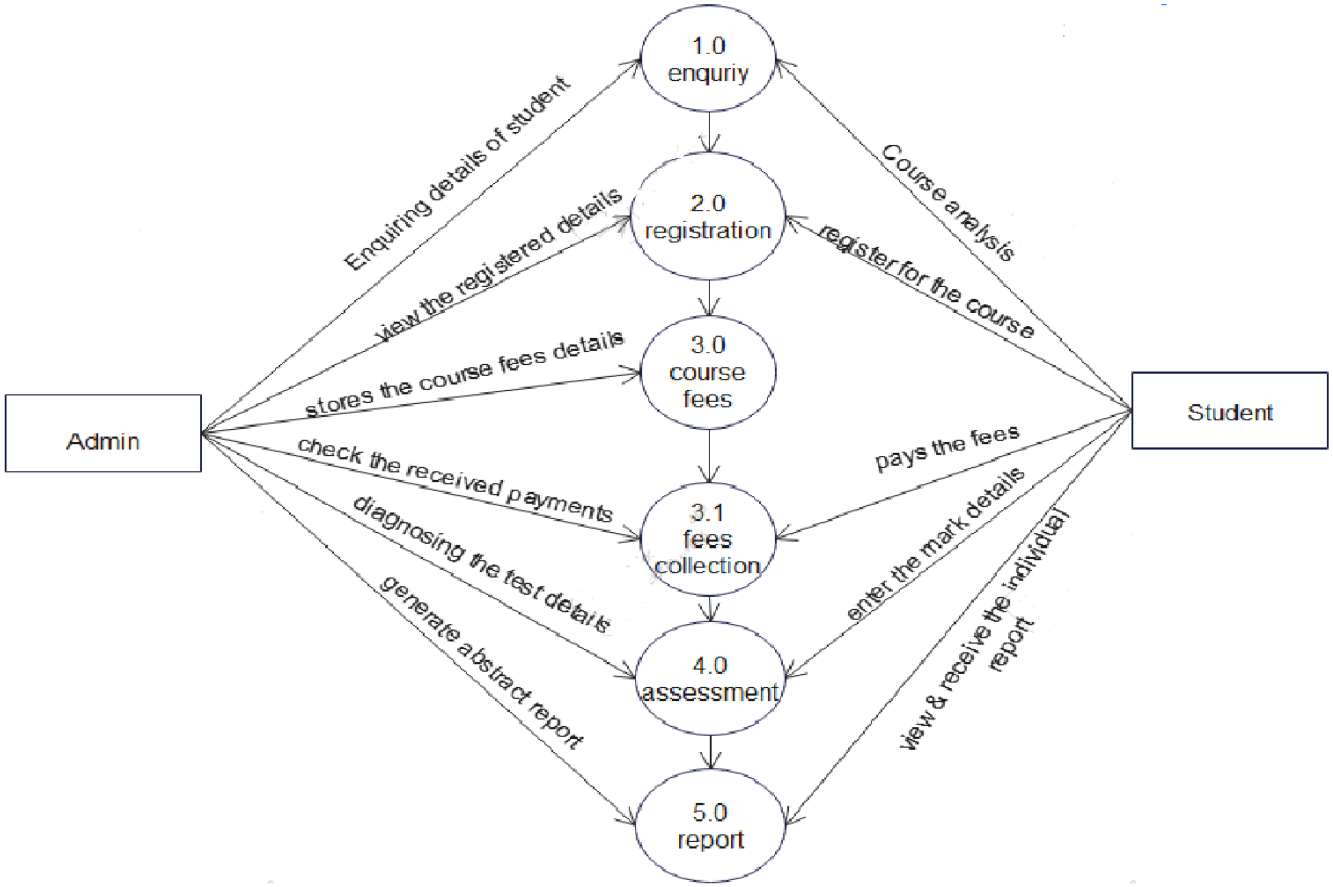


Fig 5.3.2 Level 1 Dataflow

5.4 ER DIAGRAM

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.

At first look, an ER diagram looks very similar to the flowchart. However, ER Diagram includes many specialized symbols, and its meanings make this model unique. The purpose of ER Diagram is to represent the entity framework infrastructure.

Prime reasons for using the ER Diagram

- Helps you to define terms related to entity relationship modelling.
- Provide a preview of how all your tables should connect, what fields are going to be on each table.
- Helps to describe entities, attributes, relationships.
- ER diagrams are translatable into relational tables which allows you to build databases quickly.
- ER diagrams can be used by database designers as a blueprint for implementing data in specific software applications.
- The database designer gains a better understanding of the information to be contained in the database with the help of ERP diagram.
- ERD Diagram allows you to communicate with the logical structure of the database to users.

ER Diagrams Symbols & Notations

Entity Relationship Diagram Symbols & Notations mainly contains three basic symbols which are rectangle, oval and diamond to represent relationships between elements, entities and attributes. There are some sub-elements which are based on main elements in ERD Diagram. ER Diagram is a visual representation of data that describes how data is related to each other using different ERD Symbols and Notations.

Following are the main components and its symbols in ER Diagrams:

- **Rectangles:** This Entity Relationship Diagram symbol represents entity types
- **Ellipses :** Symbol represent attributes
- **Diamonds:** This symbol represents relationship types
- **Lines:** It links attributes to entity types and entity types with other relationship types
- **Primary key:** attributes are underlined
- **Double Ellipses:** Represent multi-valued attributes

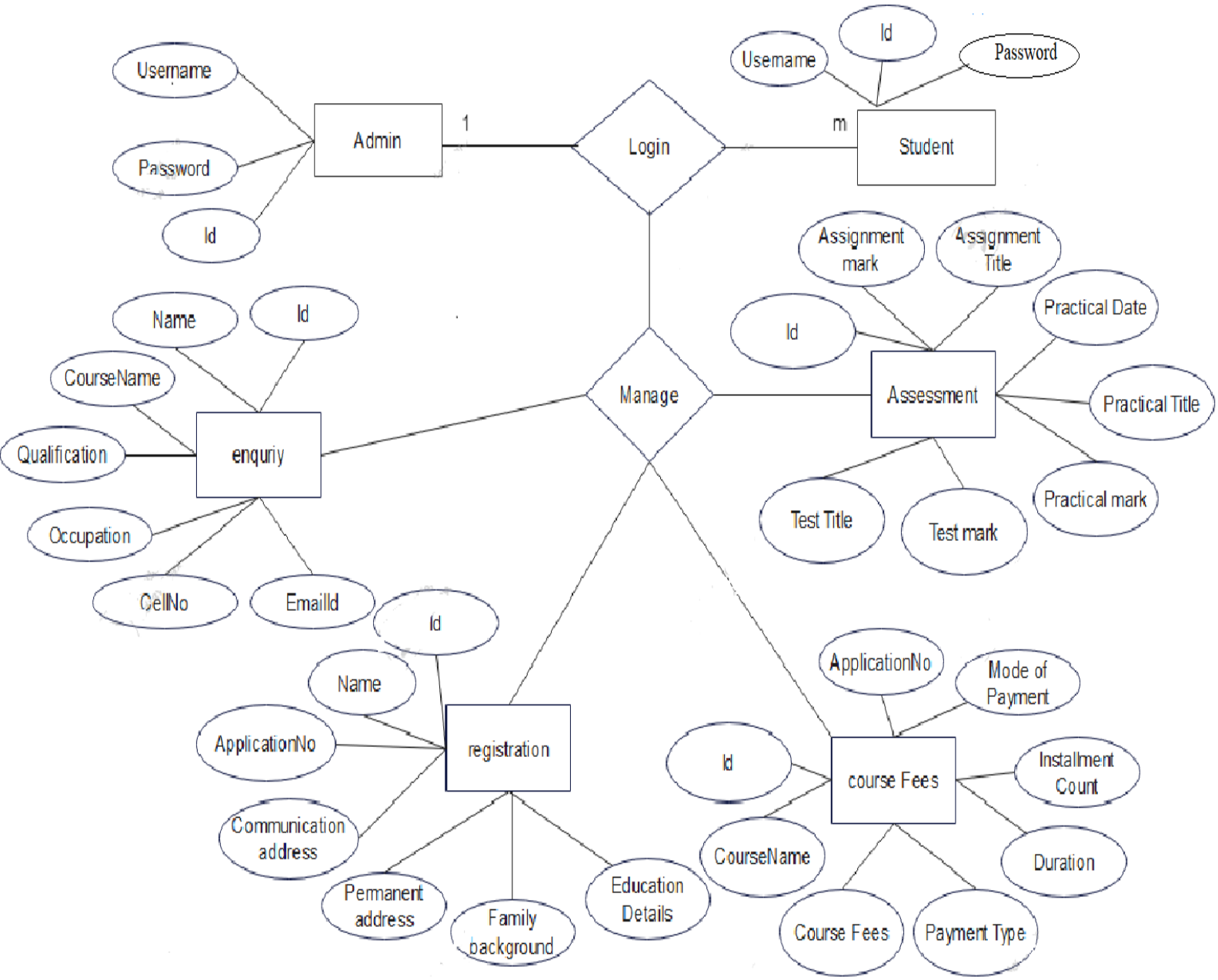


Fig 5.4 ER Diagram

CHAPTER-6

MODULE DESCRIPTION

MODULES

- Login
- Sign Up
- Enquiry Module
- Registration Module
- Assessment Module
- Course Fee Module
- Fees Collection

6.1 Sign Up Module: Registration process must be initiated whenever a visitor clicks on the ‘Sign Up’ located on any public page. Clicking ‘Sign Up’ button must invoke a pop-up Sign Up form. Sign Up form consists of two registration options that must be toggled by using tabs.

6.2 Login Module: The Login Module is a portal module that allows users to type a user name and password to log in. You can add this module on any module tab to allow users to log in to the system. If you allow users to create accounts and turn on Portal Direct Entry, a Create Account link appears in the Login Module.

6.3 Enquiry Module: Information are collected from the individual and stored in database tables. From this information collection they are moved to the companies. Here you can edit your information or export it on your computer. Some validation process also applied to the form. The objective of the Enquiry Letter is to make a request to the recipient. The field Names kept are :

ID, Name, Course, Degree, Department, College Name, Year of Passing, Occupation, Cell No, WhatsApp No, Email Id

6.4 Registration Module: The student selects from various courses available. Once registered the student gets an Enrolment Number through message. Collects the user data and those data are stored in the database. Photos are stored and converted as binary datatype. The photo size should be 10kb to upload it in a form. Here webcam technologies are also provided.

App No, Course Name, Curriculum, Name, Permanent Address, Communication Address, Location, City, Postal, Contact, DOB, Age.

6.5 Assessment Module: Creating test questions for the course, test questions will reside in a draft area. Modifying the questions, delete the entire test browse through the tests that students have submitted, just as a student would., view the results of those students that have taken test for his courses. Replying back to the messages from students. The students have to enter the assignment mark, practical mark, test mark for the course he/she applied. Once he submitted the form goes to course fee form. The AppNo, course, Name fields are coming from the enquiry form. The required fields are

Assignment No, Assignment Date, Assignment Title, Assignment mark, Practical No, Practical Date, Practical mark, Test date, Test No, Test Title, Test Mark.

6.6 Course Fee Module :For the course Fee module, we have set the instalment option, course Name, tax ,and kept search button so that we can easily get the course name that is entered in the register module .Level of instalment are also given the student can pay the amount through instalment also .You can edit the content /Add /Delete the content in the course fee form .Data grid view is inserted at the last to write some instructions .The required fields are:

Course Name, Duration, Fees, Tax, Instalment No, Instal1, Instal2, Instal3.

6.7 Fees Collection: Here the payment type, Mode of payment, Receipt No, Instalment No, Reference id fieldnames are given. we can pay the fees either through online or offline that option is available here. These modules get connected to course fee model for instalment details. validation concept is also included for each form.

App No, Course Name, Course fees, Payment Details, Receipt No, Mode Of payment, Date, Receipt No

CHAPTER-7

TABLE OF STRUCTURE

Table Name: EnquiryTable

Table Description : It collects and maintains the student details.

FIELDNAME	TYPE	FIELDSIZE	DESCRIPTION
Id	Int		Uniquely identify the student
EName	nvarchar		Student name
EDate	nvarchar		Current date of enquiry
ECourse	nvarchar		Choose the course
EDegree	nvarchar(MAX)		Degree studied
EDepartment	nchar		Department
ECollege	nvarchar		College
EyearOfPassing	nvarchar		Year passed out
EOccupation	nvarchar		Current occupation
ECellNo	nchar		Student cell no
EWhatsApp	nchar		Student what's App
EEmailId	nvarchar		Student emailid

Table Name: ApplicationTable

Table Description : It stores the details of the registered students.

FIELDNAME	TYPE	FIELDSIZE	DESCRIPTION
Id	int		Identifies the student
AppFor	nvarchar		Student selects the courses
AppNo	nchar		Student AppNo is given
Course Name	nvarchar		Choosing course name to learn
Curriculam	nvarchar		Select the curriculam
Student Name	nvarchar		Student Name
CAddress	nvarchar		Address of the Temporary
CLocation	nvarchar		Location near by
CCity	nvarchar		City
CPostalCode	nchar		Postal code
CPhoneNumber	nchar		Phone Number
CWhatsAppNumber	nchar		WhatsApp
PAddress	nvarchar		Permanent Address

PLocation	nvarchar		Location
PCity	nvarchar		City
PPostalCode	nchar		Postalcode
PPhoneNumber	nchar		Phone Number
PWhatsAppNumber	nchar		WhatsApp
Age	nchar		Student Age
DOB	nvarchar		Student Dob
FName	nvarchar		Father’s Name
Fage	int		Father’s Age
FEducation	nchar		Father’s Education
FOccupation	nvarchar		Father’s Occupation
MName	nvarchar		Mother’s Name
Mage	int		Mother’s Age
MEducation	nchar		Mother’s Education
MOccupation	nvarchar		Mother’s Occupation
S1Name	nvarchar		Sibling Name
S1Age	int		Sibling Age
S1Education	nchar		Sibling Education
S1Occupation	nvarchar		Sibling Occupation
S2Name	nvarchar		Sibling Name
S2Age	int		Sibling Age
S2Education	nchar		Sibling Education
S2Occupation	nvarchar		Sibling Occupation
S3Name	nvarchar		Sibling Name
S3Age	int		Sibling Age
S3Education	nchar		Sibling Education
S3Occupation	nvarchar		Sibling Occupation
E1Duration	nchar		Duration of course
E1Institute	nvarchar		Institute
E1Authority	nvarchar		Affiliated to Authority
E1YearOfPassing	int		Year Passed out
E1Percentage	int		Percentage of marks
E2CourseName	nvarchar		Refers group
E2Duration	nchar		Duration of course
E2Institute	nvarchar		Institute
E2Authority	nvarchar		Affiliated to Authority
E2YearOfPassing	int		Year Passed out
E2Percentage	int		Percentage of marks
E2CourseName	nvarchar		Refers group
E2Duration	nchar		Duration of course
E2Institute	nvarchar		Institute
E2Authority	nvarchar		Affiliated to Authority

E2YearOfPassing	int		Year Passed out
E2Percentage	int		Percentage of marks
E3CourseName	nvarchar		Refers group
E3Duration	nchar		Duration of course
E3Institute	nvarchar		Institute
E3Authority	nvarchar		Affiliated to Authority
E3YearOfPassing	int		Year Passed out
E3Percentage	int		Percentage of marks
E4CourseName	nvarchar		Refers group
E4Duration	nchar		Duration of course
E4Institute	nvarchar		Institute
E4Authority	nvarchar		Affiliated to Authority
E4YearOfPassing	int		Year Passed out
E4Percentage	int		Percentage of marks
E5CourseName	nvarchar		Refers group
E5Duration	nchar		Duration of course
E5Institute	nvarchar		Institute
E5Authority	nvarchar		Affiliated to Authority
E5YearOfPassing	int		Year Passed out
E5Percentage	int		Percentage of marks

Table Name: CourseFeesTable

Table Description : It stores the course details and fees details.

FIELDNAME	TYPE	FIELDSIZE	DESCRIPTION
Id	int		Identify the courses
Course Name	nvarchar		Available Courses given
Duration	nvarchar		Course Duration
Fees	nvarchar		Course Fees
Tax	nvarchar		Tax
Install Count	nvarchar		Number of instalments
Instal 1	nvarchar		First Instalment
Install2	nvarchar		Second Instalment
Install3	nvarchar		Third Instalment

Table Name: FeesCollectionTable

Table Description: It stores the details of fees collected from the student.

FIELDNAME	TYPE	FIELDSIZE	DESCRIPTION
Id	int		Unique id

AppNo	nvarchar		Student Application Number
Course Name	nvarchar		Course Name
Course Fees	nvarchar		Course Fees
Payment Type	nvarchar		Select Payment Type
InstallmentNo	Nvarchar		Instalment Number
ReceiptNo	nvarchar		Receipt Number
Date	nvarchar		Fees Collected Date
Fees	nvarchar		Fees
ModeOfPayment	nvarchar		Online/Cash
ReferrenceNumber	nvarchar		Reference Number
Total	nvarchar		Total amount paid

Table Name: Assessment Table

Table Description : It stores the marks of the student.

FIELDNAME	TYPE	FIELDSIZE	DESCRIPTION
Id	int		Unique name
AppNo	nvarchar		Student Application Number
Assignment No	nvarchar		Assignment Number
Assignment Date	nvarchar		Assignment Date
Assignment Title	nvarchar		Assignment Title
Assignment Mark	nvarchar		Assignment Mark
Practical No	nvarchar		Practical Number
Practical Date	nchar		Practical Date
Practical Title	nvarchar		Practical Title
Practical Mark	nvarchar		Practical Mark
Test No	nvarchar		Test Number
Test Date	nvarchar		Test Date
Test Title	nvarchar		Test Title
Test Mark	nvarchar		Test Mark

Table Name: Login /Sign Up Table

Table Description : It stores the student username and password.

FIELDNAME	TYPE	FIELDSIZE	DESCRIPTION
Id	int		Unique id
Username	nvarchar		Username
Password	nvarchar		Password

CHAPTER-8

TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

TYPES OF TESTS

8.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

TEST STRATEGY AND APPROACH

Field testing will be performed manually and functional tests will be written in detail.

TEST OBJECTIVES

- All field entries must work properly.
- Pages must be activated from the identified link.

- The entry screen, messages and responses must not be delayed.

FEATURES TO BE TESTED

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

8.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

8.3FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centred on the following items:

8.3.1 Valid Input : identified classes of valid input must be accepted.

8.3.2 Invalid Input : identified classes of invalid input must be rejected.

8.3.3 Functions : identified functions must be exercised.

8.3.4 Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

8.4 SYSTEM TESTING

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

8.5 WHITE BOX TESTING

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

8.6 BLACK BOX TESTING

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document.

It is a testing in which the software under test is treated, as a black box. You cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

8.7 ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER-9

CONCLUSION

LMS is a fast-developing educational model that is likely to replace the traditional learning. This model will change the methods of instructions, as it is more learner-centred relative to the traditional model. Learners will access information via computers, interact via computers, and be evaluated via computers. Various tools such as computer software and hardware will be required in e-learning. The role of the learner will change from a passive receiver of processed information to an active processer and manager of self and information. The tutor changes from an all-time supervisor and processor to an instructor. Finally, effectiveness of e-learning can be evaluated through process appraisal or result assessment.

CHAPTER-10

APPENDICES

10.1 SOURCE CODE

ADMIN LOGIN MODULE:

```
<html class="no-js" lang="en">
<head>
  <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
  <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet"></link>
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-
validator/0.11.9/validator.min.js"></script>

  <style>
    body {
      background: #EECDA3;
      background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
      background: linear-gradient(to top, #EF629F, #EECDA3);
    }

    .container {
      max-width: 750px;
     max-height:500px;
    }

    .has-error label,
    .has-error input,
    .has-error textarea {
      color: red;
      border-color: red;
    }

    .list-unstyled li {
      font-size: 13px;
      padding: 4px 0 0;
      color: red;
    }
  </style>
</head>

<body>
  <div class="container mt-5">
    <div class="card">
      <h5 class="card-header text-center">Login Form</h5>
      <div class="card-body">
        <form role="form" data-toggle="validator">
```

```
                    <div class="form-group">
                      <label>UserName</label>
                       <input type="text" class="form-control" maxlength="10" minlength="3"
pattern="^[a-zA-Z0-9_.- ]*$" id="UName" placeholder="username" required />
                    <div class="help-block with-errors"></div>
                    </div>
                     <div class="form-group">
                      <label>Password</label>
                      <input type="Password" class="form-control"  maxlength="10" minlength="5"
                         pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}"                id="Password"
placeholder="password" required>

                      <!-- Error -->
                      <div class="help-block with-errors"></div>
                    </div>
                    <div class="form-group">
                      <button type="login" class="btn btn-primary btn-block">Login</button>
                      <button type="Sign up" class="btn btn-primary btn-block">Sign Up</button>
                    </div>
                    </form>
                  </div>
                </div>
              </div>
                    </body>

              </html>
```

Sign Up Module:

```
        <html class="no-js" lang="en">
        <head>
          <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
          <link          href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet"></link>
          <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
          <script                              src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-
validator/0.11.9/validator.min.js"></script>

          <style>
            body {
              background: #EECDA3;
              background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
              background: linear-gradient(to top, #EF629F, #EECDA3);
            }

            .container {
               max-width: 750px;
              max-height:500px;
            }

            .has-error label,
            .has-error input,
            .has-error textarea {
```

```
            color: red;
            border-color: red;
        }

        .list-unstyled li {
            font-size: 13px;
            padding: 4px 0 0;
            color: red;
        }
    </style>
</head>

<body>
    <div class="container mt-5">
        <div class="card">
            <h5 class="card-header text-center">SignUp Form</h5>
            <div class="card-body">
                <form role="form" data-toggle="validator">
                    <div class="form-group">
                        <label>Name</label>
                        <input type="text" id="TxtName" placeholder="Name" pattern="^[a-zA-Z0-9_.-
]*$" minlength="3" maxlength="10" class="form-control" required />
                        <div class="help-block with-errors"></div>
                    </div>
                    <div class="form-group">
                        <label>Password</label>
                        <input type="Password" class="form-control"  maxlength="10" minlength="5"
                            pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}"                id="Password"
placeholder="password" required>


                        <div class="help-block with-errors"></div>
                    </div>
                    <div class="form-group">
                        <label>Confirm Password</label>
                        <input type="Password" class="form-control"  maxlength="10" minlength="5"
                            pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}"                id="Password"
placeholder="password" required>


                        <div class="help-block with-errors"></div>
                    </div>
                    <div class="form-group">
                        <label>Email</label>
                        <input type="email" class="form-control" maxlength="20" minlength="5"
pattern="/^(([^<>()\[\]\\.,;:\s@]+(\.[^<>()\[\]\\.,;:\s@]+)*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-
9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;" placeholder="email" required />
                    <div class="help-block with-errors"></div>
    </div>
                <div class="form-group">
                    <button type="Register" class="btn btn-primary btn-block">Register</button>
                            </div>
```

```
            </form>
              </div>
            </div>
          </div>
        </body>

          </html>
```

Enquiry Form:

```
<html class="no-js" lang="en">
<head>
  <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
  <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet"></link>
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-
validator/0.11.9/validator.min.js"></script>

  <style>
    body {
      background: #EECDA3;
      background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
      background: linear-gradient(to top, #EF629F, #EECDA3);
    }

    .container {
      max-width: 750px;
    }

    .has-error label,
    .has-error input,
    .has-error textarea {
      color: red;
      border-color: red;
    }

    .list-unstyled li {
      font-size: 13px;
      padding: 4px 0 0;
      color: red;
    }
  </style>
</head>

<body>
  <div class="container mt-5">
    <div class="card">
      <h5 class="card-header text-center">Enquiry Form</h5>
      <div class="card-body">
        <form role="form" data-toggle="validator">
          <div class="form-group">
            <label>Name</label>
```

```
                    <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                        pattern="^[a-zA-Z0-9_.-]*$" id="name" placeholder="Name" required>
                    <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label>Course</label>
                  <select class="custom-select"  id="dropcourse" pattern=" /^(!?--\s*[Ss]elect\s*--)$/"
required >
                  <option></option>
                   <option>IT FOUNDATION</option>
                  <option>UI DEVELOPER</option>
                  <option>CLOUD DEVELOPER</option>
                  <option>FULLSTACK DEVELOPER</option>
                  <option>CCC</option>
                  <option>INTERNSHIP</option>
                </select>
                      <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label><b>Qualification</b></label>
                  </div>

                  <div class="form-group">
                    <label>Degree</label>
                    <select class="custom-select" id="dropdegree"  pattern=" /^(!?--\s*[Ss]elect\s*--
)$/" required >
                  <option></option>
                  <option>BE</option>
                  <option>BTECH</option>
                  <option>ME</option>
                  <option>MTECH</option>
                  <option>BSC</option>
                  <option>BCA</option>
                  <option>MSC</option>
                  <option>MCA</option>
                  <option>MBA</option>
                  <option>OTHERS</option>
                </select>

                    <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label>College Name</label>
                    <input type="text" class="form-control" name="ColName" maxlength="18"
minlength="10"
                        pattern="^[a-zA-Z0-9_.-]*$"  id="Colname"  placeholder="ColegeName"
required />
                    <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label>Department</label>
```

```
                    <input type="text" class="form-control" name="Deprt" maxlength="12"
minlength="3"
                      pattern="^[a-zA-Z0-9_.-]*$" id="Dept" placeholder="Department" required>
                    <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label>Year Of Passing</label>
                    <input type="number" class="form-control" name="Year" maxlength="3"
minlength="2"
                      pattern="/^[0-9]+$/" id="Year" placeholder="Year" required>
                     <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label><b>Occupation</b></label>
                     <div class="form-check form-check-inline" required>
                   <input class="form-check-input" type="radio"  id="Radio"/>
                  <label class="form-check-label" for="radio">Student</label>
                  </div>
                 <div class="form-check form-check-inline">
                   <input class="form-check-input" type="radio"  id="Radio1"/>
                  <label class="form-check-label" for="radio1">Jobseeker</label>
                  </div>
                 <div class="form-check form-check-inline">
                   <input class="form-check-input" type="radio"  id="Radio2"/>
                  <label class="form-check-label" for="radio2">Employee</label>
                  </div>
              <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label>CellNo</label>
                    <input type="number" class="form-control" name="ATitle" maxlength="10"
minlength="10"
                      pattern="/^[0-9]+$/" id="CellNo" placeholder="CellNo" required>
                  <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <label>Whats AppNo</label>
                    <input type="text" class="form-control" name="PMark" maxlength="10"
minlength="10"
                      pattern="/^[0-9]+$/" id="Wno" placeholder="WhatsAppNo" required>
                    <div class="help-block with-errors"></div>
                  </div>

                 <div class="form-group">
                    <label>Email</label>
                    <input type="email" class="form-control" id="inputEmail" placeholder="Email"
required>
                    <div class="help-block with-errors"></div>
                  </div>
                  <div class="form-group">
                    <button type="submit" class="btn btn-primary btn-block">Submit</button>
                  </div>
```

```
            </form>
          </div>
        </div>
      </div>
    </body>
    </html>
```

Registration form:

```
    <%@ Page Language="C#" AutoEventWireup="true" Code Behind="bootform1.aspx.cs" Inherits="LMS.WebForm3" %>

    <!DOCTYPE html>

    <html lang="en">

    <head>

    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <meta name="author" content="Saurav">

    <link href="css/bootstrap.min.css" rel="stylesheet">

    <title>Responsive Registration Form Template</title>

      <script src="bootstrap.bundle.min.js"></script>

       <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

      <script src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-validator/0.11.9/validator.min.js"></script>

     <scripttype="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"></script>

    <script src="WebCam.js" type="text/javascript"></script>

    <script type="text/javascript">

      $(function () {

        Webcam.set({

          width: 320,
```

```
            height: 240,
            image_format: 'jpeg',
            jpeg_quality: 90
        });
        Webcam.attach('#webcam');
        $("#btnCapture").click(function () {
            Webcam.snap(function (data_uri) {
                $("#imgCapture")[0].src = data_uri;
                $("#btnUpload").removeAttr("disabled");
            });
        });
        $("#btnUpload").click(function () {
            $.ajax({
                type: "POST",
                url: "CS.aspx/SaveCapturedImage",
                data: "{data: '" + $("#imgCapture")[0].src + "'}",
                contentType: "application/json; charset=utf-8",
                dataType: "json",
                success: function (r) { }
            });
        });
    });
</script>
    <style>
        body {
```

```
            background: #EECDA3;
            background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
            background: linear-gradient(to top, #EF629F, #EECDA3);
        }
        .container {
            max-width: 950px;
        }
        .has-error label,
        .has-error input,
        .has-error textarea {
            color: red;
            border-color: red;
        }
        .list-unstyled li {
            font-size: 13px;
            padding: 4px 0 0;
            color: red;
        }
         td .fv-plugins-icon {
                right: 10px;
                top: 10px;
            }
    </style>
    </head>
<body>
```

```
<div class="container mt-5">
  <div class="card">
    <h5 class="card-header text-center">Register Form</h5>
    <div class="card-body">
     <form role="form" data-toggle="validator" id="demoForm">
    <div class="py-5 text-center">
   <img class="d-block mx-auto mb-4" src="image/Softech.jpg"  alt="" width="243" height="68">
      <h2 class="text-danger bg-info">Registration Form Template</h2>
    </div>
  <table border="0" cellpadding:"0" cellspacing:"0">
    <tr>
      <th align:"center"><u>Live Camera</u></th>
      <th align:"center"><u>Captured image</u></th>
    </tr>
    <tr>
      <td><div id="webcam"></div></td>
      <td><img id="imgCapture" src="..."/></td>
</tr>
    <tr>
      <td align:"center">
        <input type="button" id="btnCapture" value="Capture" />
      </td>
       <td align:"center">
        <input type="button" id="btnUpload" value="Upload"  />
      </td>
```

```
                <td align:"center"><input type="file" id="btnload" name="filename"></td>
                <td><div><img align:"center" src="..." id="imgLoad" /></div></td>
                <td align:"center"><input type="button" id="btnupload" value="Upload" /></td>
              </tr>
             </table>
            <br />
           <div class="form-group">
            <label for="" class="form-label">Name</label>
              <input type="text" class="form-control"  maxlength="10" minlength="5"
                     pattern="^[a-zA-Z0-9_.-]*$" id="name" placeholder="Name" required>
                  <!-- Error -->
                  <div class="help-block with-errors"></div>
        </div>
           <br />
          <div class="form-group position-relative">
            <label for="Age">
             Age</label>
             <input type="number"  class="form-control" id="NumAge" maxlength="3" minlength="2" pattern="/^[0-9]+$/"  required />
             <div class="help-block with-errors"></div>
            </div>
           <br />
           <div class="form-group position-relative">
           <label for="exampleInputDOB">
            DOB</label>
              <input type="number"  class="form-control" id="TxtDOB" maxlength="10"
```

```
minlength="8" pattern="/^[0-9]+$/" placeholder="DOB" required />
                    <div class="help-block with-errors"></div>
                </div>
                    <br />
                <div class="row">
                    <div class="col">
                <div class="form-group">
                <label for="exampleInputAppFor">AppFor</label>
                <select class="form-control  d-block w-100" id="AppFor" required>
                    <option value="">Choose...</option>
                    <option>DOEACC</option>
                    <option>CSI</option>
                   <option>NET</option>
                    <option>NCST</option>
                    <option>TANCET</option>
                    <option>GATE</option>
                    <option>IETE</option>
                     <option>IES</option>
                    <option>MICROSOFT</option>
                     <option>SUN</option>
                    <option>PROJECT</option>
                    <option>OTHERS</option>
                </select>
                   <div class="help-block with-errors"></div>
                </div>
```

```
            </div>
            </div>
          <br />
          <div class="form-group">
         <label for="exampleInputAppNo">AppNo</label>
         <input type="Number" class="form-control" id="TxtAppNo" pattern="/^[0-9 ]+$/" maxlength="10" minlength="1" required />
            <div class="help-block with-errors"></div>
            </div>
        <br />
        <div class="form-group">
          <label for="exampleInputCName">CourseName</label>
          <input type="text" class="form-control" id="TxtCName" pattern="^[a-zA-Z0-9_.- ]*$" placeholder="CName" maxlength="10" minlength="3" required />
            <div class="help-block with-errors"></div>
            </div>
        <br />
            <div class="col">
            <h6 class="mb-0 me-4">Curriculam: </h6>
            <div class="form-check form-check-inline mb-0 me-4">
                  <input
                    class="form-check-input"
                    type="radio"
                    name="inlineRadioOptions"
                    id="RdoState"
                    value="option1" required
```

```
                />
                <label class="form-check-label" for="State">State</label>
              </div>
              <div class="form-check form-check-inline mb-0 me-4">
                <input
                  class="form-check-input"
                  type="radio"
                  name="inlineRadioOptions"
                  id="RdoNational"
                  value="option2" required
                />
                <label class="form-check-label" for="National">National</label>
              </div>
              <div class="form-check form-check-inline mb-0">
                <input
                  class="form-check-input"
                  type="radio"
                  name="inlineRadioOptions"
                  id="RdoInternational"
                  value="option3" required />
                <label class="form-check-label" for="International">International</label>
                  </div>
             <div class="help-block with-errors"></div>
              </div>
<div class ="form-group">
```

```
<div class="text-center mb-3 pb-2 mt-3">
<h4 style="color: #495057 ;">Communication Address</h4>
</div>
</div>
<div class="form-group">
<table>
<tr>
<td><labelclass="form-label" for="form1Example1">Address</label></td>
<td style="width:50%"><input type="text" id="TxtCAddress" class="form-control input-custom" pattern="^[a-zA-Z0-9_.- ]*$" maxlength="60" minlength="20" required/></td>
<td><label class="form-label" for="form1Example2">Location</label></td>
<td style="width:50%;"><input type="text" id="TxtCLocation" class="form-control input-custom" maxlength="10" minlength="5" pattern="^[a-zA-Z0-9_.- ]*$" required/>
</td>
</tr>
<tr><td></td></tr>
<tr><td></td></tr>
<tr><td></td></tr>
<tr>
<td><label class="form-label" for="form1Example3">City</label></td>
<td> <input type="text" id="TxtCCity" class="form-control input-custom" maxlength="10" minlength="5" pattern="^[a-zA-Z0-9_.- ]*$" required />
</td>
<td> <label class="form-label" for="form1Example4">PostalCode</label></td>
<td style="width:50%;"><input type="number" id="TxtCPCode" class="form-control
```

```
input-custom" pattern="^[a-zA-Z0-9_.- ]*$" minlength="6" maxlength="7" required/>
                </td>
                    </tr>
                        <tr><td></td></tr>
                        <tr><td></td></tr>
                        <tr><td></td></tr>
                        <tr>
                            <td><label class="form-label" for="form1Example5">WhatsAppNumber</label></td>
                            <td style="width:50%;"><input type="number" id="TxtCWNo" class="form-control input-custom" pattern="^[0-9 ]+$" minlength="10" maxlength="10" required/></td>
                            <td><label class="form-label" for="form1Example6">PhoneNumber</label></td>
                            <td style="width:50%;"><input type="number" id="TxtCPNo" class="form-control input-custom" pattern="^[0-9 ]+$" minlength="6" maxlength="7" required/>
            </td>
                        </tr>
                            </table>
                        <div class="help-block with-errors"></div>
                </div>
              <div class="form-group">
             <div class="text-center mb-3 pb-2 mt-3">
               <h4 style="color: #495057 ;">Permanent Address</h4>
                <div class="form-check">
               <input class="form-check-input" type="checkbox" id="check1" name="option1" value="something" >
```

```
            <label class="form-check-label">Both addresses are same</label>
        </div>
            </div>
          </div>
              <div class="form-group">
                  <table>
                     <tr>
                        <td><label class="form-label" for="form1Example1">Address</label></td>
                    <td style="width:50%"><input type="text" id="TxtPAddress" class="form-control input-custom" pattern="^[a-zA-Z0-9_.- ]*$" maxlength="60" minlength="20" required/></td>
                    <td><label class="form-label" for="form1Example2">Location</label></td>
                    <td style="width:50%;"><input type="text" id="TxtPLocation" class="form-control input-custom" maxlength="10" minlength="5" pattern="^[a-zA-Z0-9_.- ]*$" required/>
                        </td>
                       </tr>
                    <tr><td></td></tr>
                    <tr><td></td></tr>
                    <tr><td></td></tr>
                    <tr>
                        <td><label class="form-label" for="form1Example3">City</label></td>
                  <td> <input type="text" id="TxtPCity" class="form-control input-custom" maxlength="10" minlength="5" pattern="^[a-zA-Z0-9_.- ]*$" required />
                </td>
                        <td><labelclass="form-label" for="form1Example4">PostalCode</label></td>
```

<td style="width:50%;"><input type="number" id="TxtPPCode" class="form-control input-custom" pattern="^[a-zA-Z0-9_.-]*$" minlength="6" maxlength="7" required/>

</td>

</tr>

<tr><td></td></tr>

<tr><td></td></tr>

<tr><td></td></tr>

<tr>

<td><label class="form-label" for="form1Example5">WhatsAppNumber</label></td>

<td style="width:50%;"><input type="number" id="TxtCPWNo" class="form-control input-custom" pattern="^[0-9]+$" minlength="10" maxlength="10" required/></td>

<td><label class="form-label" for="form1Example6">PhoneNumber</label></td>

<td style="width:50%;"><input type="number" id="TxtPPNo" class="form-control input-custom" pattern="^[0-9]+$" minlength="6" maxlength="7" required/>

</td>

</tr>

</table>

<button type="Next" class="btn btn-secondary">Next</button>

<button type="Save" class="btn btn-secondary">Save</button>

```
        </form>
                    </div>
                </div>
                </div>
                </body>
            </html>
```

FAMILY PROFILE MODULE

```
        <html class="no-js" lang="en">
<head>
    <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet"></link>
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-validator/0.11.9/validator.min.js"></script>
    <style>
        body {
            background: #EECDA3;
            background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
            background: linear-gradient(to top, #EF629F, #EECDA3);
        }
        .container {
            max-width: 750px;
        }
```

```
        .has-error label,
        .has-error input,
        .has-error textarea {
            color: red;
            border-color: red;
        }
        .list-unstyled li {
            font-size: 13px;
            padding: 4px 0 0;
            color: red;
        }
    </style>
</head>
<body>
    <div class="container mt-5">
        <div class="card">
            <h5 class="card-header text-center">Family Profile</h5>
            <div class="card-body">
                <form role="form" data-toggle="validator">
                    <div class="form-group">
                      <label style="color:darkblue"> <b>Father's Detail</b></label>
                    </div>
                     <div class="form-group">
                      <label>Name</label>
                      <input type="text" class="form-control"  maxlength="10" minlength="5"
```

```
                pattern="^[a-zA-Z0-9_.-]*$" id="FName" placeholder="Name" required>
            <!-- Error -->
            <div class="help-block with-errors"></div>
        </div>
        <div class="form-group">
            <label>Age</label>
            <input type="number" class="form-control"  maxlength="10" minlength="2"
                pattern="/^[0-9]+$/" id="FAge"  required>
            <!-- Error -->
            <div class="help-block with-errors"></div>
        </div>
        <div class="form-group">
            <label>Education</label>
            <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                pattern="^[a-zA-Z0-9_.-]*$" id="FEducation" placeholder="Education"
required>
            <!-- Error -->
            <div class="help-block with-errors"></div>
        </div>
        <div class="form-group">
            <label>Occupation</label>
            <input type="text" class="form-control"  maxlength="10" minlength="5"
                pattern="^[a-zA-Z0-9_.-]*$" id="FOccupation" placeholder="Occupation"
required>
```

```
<!-- Error -->
<div class="help-block with-errors"></div>
</div>
<div class="form-group">
<label style="color:darkblue"><b>Mother's Detail</b></label>
</div>
<div class="form-group">
<label>Name</label>
<input type="text" class="form-control"  maxlength="10" minlength="5"
pattern="^[a-zA-Z0-9_.-]*$" id="MName" placeholder="Name" required>
<!-- Error -->
<div class="help-block with-errors"></div>
</div>
<div class="form-group">
<label>Age</label>
<input type="number" class="form-control"  maxlength="10" minlength="2"
pattern="/^[0-9]+$/" id="MAge"  required>
<!-- Error -->
<div class="help-block with-errors"></div>
</div>
<div class="form-group">
<label>Education</label>
<input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
pattern="^[a-zA-Z0-9_.-]*$" id="MEducation" placeholder="Education"
required>
```

```
<!-- Error -->
<div class="help-block with-errors"></div>
</div>
<div class="form-group">
<label>Occupation</label>
<input type="text" class="form-control" name="name" maxlength="10" minlength="5"
pattern="^[a-zA-Z0-9_.-]*$" id="MOccupation" placeholder="Occupation" required>
<!-- Error -->
<div class="help-block with-errors"></div>
</div>
<div class="form-group">
<label style="color:darkblue"><b>Sibling's Detail</b></label>
</div>
<div class="form-group">
<label>Name</label>
<input type="text" class="form-control" maxlength="10" minlength="5"
pattern="^[a-zA-Z0-9_.-]*$" id="S1Name" placeholder="Name" required>
<!-- Error -->
<div class="help-block with-errors"></div>
</div>
<div class="form-group">
<label>Age</label>
<input type="number" class="form-control" maxlength="10" minlength="2"
pattern="/^[0-9]+$/" id="S1Age" required>
```

```
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Education</label>
                <input type="text" class="form-control" name="name" maxlength="10" minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="S1Education" placeholder="Education" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Occupation</label>
                <input type="text" class="form-control" name="name" maxlength="10" minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="S1Occupation" placeholder="Occupation" required>
<!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
             <div class="form-group">
                <label>Name</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="S2Name" placeholder="Name" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
```

```
            </div>
            <div class="form-group">
               <label>Age</label>
               <input type="number" class="form-control"  maxlength="10" minlength="2"
                  pattern="/^[0-9]+$/" id="S2Age"  required>
               <!-- Error -->
               <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
               <label>Education</label>
               <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                  pattern="^[a-zA-Z0-9_.-]*$" id="S2Education" placeholder="Education"
required>
               <!-- Error -->
               <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
               <label>Occupation</label>
               <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                  pattern="^[a-zA-Z0-9_.-]*$" id="S2Occupation" placeholder="Occupation"
required>
               <!-- Error -->
               <div class="help-block with-errors"></div>
            </div>
             <div class="form-group">
```

```
                <label>Name</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="S3Name" placeholder="Name" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Age</label>
                <input type="number" class="form-control"  maxlength="10" minlength="2"
                    pattern="/^[0-9]+$/" id="S3Age"  required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Education</label>
                <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="S3Education" placeholder="Education"
required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Occupation</label>
                <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
```

```
                    pattern="^[a-zA-Z0-9_.-]*$"   id="S3Occupation"   placeholder="Occupation"
required>
                    <!-- Error -->
                    <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                    <button type="Save" class="btn btn-primary btn-block">Save</button>
                    <button type="Submit" class="btn btn-primary btn-block">Submit</button>
                </div>
            </form>
        </div>
    </div>
</div>
</body>
</html>
```

EDUCATIONAL PROFILE MODULE

```
<html class="no-js" lang="en">
<head>
    <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
    <link        href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet"></link>
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
    <script                          src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-
validator/0.11.9/validator.min.js"></script>
    <style>
        body {
```

```
            background: #EECDA3;
            background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
            background: linear-gradient(to top, #EF629F, #EECDA3);
        }
        .container {
            max-width: 750px;
        }
        .has-error label,
        .has-error input,
        .has-error textarea {
            color: red;
            border-color: red;
        }
        .list-unstyled li {
            font-size: 13px;
            padding: 4px 0 0;
            color: red;
        }
    </style>
</head>
<body>
    <div class="container mt-5">
        <div class="card">
            <h5 class="card-header text-center">Educational Profile</h5>
            <div class="card-body">
```

```
<form role="form" data-toggle="validator">
  <div class="form-group">
    <label style="color:darkblue"><b>SSLC</b></label>
  </div>
   <div class="form-group">
    <label>Duration</label>
    <input type="text" class="form-control"  maxlength="10" minlength="5"
      pattern="^[a-zA-Z0-9_.-  ]*$"  id="FName"  placeholder="Duration"
required>
    <!-- Error -->
    <div class="help-block with-errors"></div>
  </div>
  <div class="form-group">
    <label>School/Institute</label>
    <input type="text" class="form-control"  maxlength="10" minlength="5"
      pattern="/^[a-zA-Z0-9_.- ]*$/" id="School" placeholder="School/Institute"
required>
    <!-- Error -->
    <div class="help-block with-errors"></div>
  </div>
  <div class="form-group">
    <label>Board</label>
    <input  type="text"  class="form-control"  name="name"  maxlength="10"
minlength="5"
      pattern="^[a-zA-Z0-9_.-]*$" id="Board" placeholder="Board" required>
```

```
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
              <div class="form-group">
                <label>Year Of Passing</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                  pattern="^[a-zA-Z0-9_.-]*$" id="Year Of Passing" placeholder="Year Of
passing" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
               <div class="form-group">
                <label>Percentage</label>
                <input type="number" class="form-control"  maxlength="10" minlength="2"
                  pattern="/^[0-9]+$/" id="Percent"  required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
             <div class="form-group">
               <label style="color:darkblue"> <b>HSLC</b></label>
              </div>
               <div class="form-group">
                <label>Duration</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                  pattern="^[a-zA-Z0-9_.-  ]*$"  id="HDuration"  placeholder="Duration"
required>
```

```
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>School/Institute</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                    pattern="/^[a-zA-Z0-9_.-                ]*$/"                id="HSchool"
placeholder="School/Institute" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Board</label>
                <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="HBoard" placeholder="Board" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Year Of Passing</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="HYear Of Passing" placeholder="Year Of
passing" required>
```

```
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
              <div class="form-group">
                <label>Percentage</label>
                <input type="number" class="form-control"  maxlength="10" minlength="2"
                  pattern="/^[0-9]+$/" id="HPercent"  required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
             <div class="form-group">
               <label style="color:darkblue"> <b>UG</b></label>
             </div>
              <div class="form-group">
                <label>Duration</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                  pattern="^[a-zA-Z0-9_.-  ]*$"  id="UDuration"  placeholder="Duration"
required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
              <div class="form-group">
                <label>Institute</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                  pattern="/^[a-zA-Z0-9_.-  ]*$/"  id="UInstitute"
placeholder="School/Institute" required>
```

```
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Board</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="UBoard" placeholder="Board" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
                <label>Year Of Passing</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                    pattern="^[a-zA-Z0-9_.-]*$" id="UYear Of Passing" placeholder="Year Of
passing" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
             <div class="form-group">
                <label>Percentage</label>
                <input type="number" class="form-control"  maxlength="10" minlength="2"
                    pattern="/^[0-9]+$/" id="UPercent"  required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
```

```
                <label style="color:darkblue"><b>Diplomo</b></label>
              </div>
               <div class="form-group">
                <label>Duration</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                   pattern="^[a-zA-Z0-9_.-  ]*$"  id="DDuration"  placeholder="Duration"
required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
              <div class="form-group">
                <label>Institute</label>
                <input type="text" class="form-control"  maxlength="10" minlength="5"
                   pattern="/^[a-zA-Z0-9_.-  ]*$/"  id="DInstitute"  placeholder="Institute"
required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
              <div class="form-group">
                <label>Board</label>
                <input  type="text"  class="form-control"  name="name"  maxlength="10"
minlength="5"
                   pattern="^[a-zA-Z0-9_.-]*$" id="DBoard" placeholder="Board" required>
                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
```

```
<div class="form-group">
  <label>Year Of Passing</label>
  <input type="text" class="form-control"  maxlength="10" minlength="5"
    pattern="^[a-zA-Z0-9_.-]*$" id="DYear Of Passing" placeholder="Year Of passing" required>
  <!-- Error -->
  <div class="help-block with-errors"></div>
</div>
 <div class="form-group">
  <label>Percentage</label>
  <input type="number" class="form-control"  maxlength="10" minlength="2"
    pattern="/^[0-9]+$/" id="DPercent"  required>
  <!-- Error -->
  <div class="help-block with-errors"></div>
</div>
<div class="form-group">
  <label style="color:darkblue"> <b>PG</b></label>
</div>
 <div class="form-group">
  <label>Duration</label>
  <input type="text" class="form-control"  maxlength="10" minlength="5"
    pattern="^[a-zA-Z0-9_.-  ]*$"  id="PDuration"  placeholder="Duration" required>
  <!-- Error -->
  <div class="help-block with-errors"></div>
</div>
```

```
<div class="form-group">
   <label>Institute</label>
   <input type="text" class="form-control"  maxlength="10" minlength="5"
      pattern="/^[a-zA-Z0-9_.-  ]*$/"  id="PInstitute"  placeholder="Institute"
required>
   <!-- Error -->
   <div class="help-block with-errors"></div>
</div>
<div class="form-group">
   <label>Board</label>
   <input type="text" class="form-control"  maxlength="10" minlength="5"
      pattern="^[a-zA-Z0-9_.-]*$" id="PBoard" placeholder="Board" required>
   <!-- Error -->
   <div class="help-block with-errors"></div>
</div>
<div class="form-group">
   <label>Year Of Passing</label>
   <input type="text" class="form-control"  maxlength="10" minlength="5"
      pattern="^[a-zA-Z0-9_.-]*$" id="PYear Of Passing" placeholder="Year Of
passing" required>
   <!-- Error -->
   <div class="help-block with-errors"></div>
</div>
 <div class="form-group">
   <label>Percentage</label>
   <input type="number" class="form-control"  maxlength="10" minlength="2"
```

```
                    pattern="/^[0-9]+$/" id="PPercent"  required>

                  <!-- Error -->

                  <div class="help-block with-errors"></div>

                </div>

                <div class="form-group">

                  <button type="Previous" class="btn btn-primary btn-block">Previous</button>

                    <button type="Submit" class="btn btn-primary btn-block">Submit</button>

                </div>

              </form>

            </div>

          </div>

        </div>

    </body>

    </html>
```

COURSE FEES MODULE

```
        <html class="no-js" lang="en">
        <head>
          <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
          <link          href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet" />
          <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
          <script                              src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-
validator/0.11.9/validator.min.js"></script>

          <style>
            body {
              background: #EECDA3;
              background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
              background: linear-gradient(to top, #EF629F, #EECDA3);
            }
            .container {
              max-width: 750px;
            }
            .has-error label,
            .has-error input,
            .has-error textarea {
```

```
            color: red;
            border-color: red;
        }

        .list-unstyled li {
            font-size: 13px;
            padding: 4px 0 0;
            color: red;
        }
    </style>
</head>

<body>
    <div class="container mt-5">
        <div class="card">
            <h5 class="card-header text-center">CourseFees Details Form</h5>
            <div class="card-body">
                <form role="form" data-toggle="validator">
                    <div class="form-group">
                    <table>
                        <tr>
                            <td>Course</td>
                        </tr>
                       <tr>
                            <td style="width:90%"><input type="text" class="form-control" name="" minlength="3" pattern="^[a-zA-Z0-9_.- ]*$" placeholder ="CourseName" required /></td>
                          <td></td>
                            <td><button type="button" class="btn btn-primary">Search</button></td>
                          </tr>
                        </table>
                       <div class="help-block with-errors"></div>
                        </div>

                    <br />
                    <div class="form-group">
                        <label>Duration</label>
                        <input type="text" class="form-control" name="" maxlength="18" minlength="10"
                            pattern="^[a-zA-Z0-9_.-]*$" id="Duration" placeholder="Duration" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                    </div>
                     <div class="form-group">
                        <label>Fees</label>
                       <input type="text" class="form-control" maxlength="18" minlength="5" pattern="^[a-zA-Z0-9_.-]*$" placeholder="Fees" required />
                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                    </div>

                    <div class="form-group">
```

```
                        <label>Tax</label>
                        <input type="text" class="form-control"  maxlength="3" minlength="3"
                          pattern="^[0-9]+$" id="Tax" placeholder="Tax" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                      </div>
                      <div class="form-group">
                        <label>InstallmentNo</label>
                        <input type="number" class="form-control"  maxlength="3" minlength="2"
                          pattern="/^[0-9]+$/" id="InstallNo" placeholder="InstallmentNo" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                      </div>
                     <div class="form-group">
                        <label>InstallNo1</label>
                        <input type="number" class="form-control" name="" maxlength="10"
minlength="10"
                          pattern="/^[0-9]+$/" id="Install1" placeholder="InstallmentNo1" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                      </div>

                    <div class="form-group">
                        <label>InstallmentNo2</label>
                        <input type="number" class="form-control" id="Instal2"
placeholder="InstallmentNo2" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                      </div>
                       <div class="form-group">
                        <label>InstallmentNo3</label>
                        <input type="number" class="form-control" id="Instal3"
placeholder="InstallmentNo3" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                      </div>
                      <div class="form-group">
                        <button type="Add" class="btn btn-primary btn-block">Add</button>
                        <button type="Edit" class="btn btn-primary btn-block">Edit</button>
                        <button type="Submit" class="btn btn-primary btn-block">Submit</button>
                      </div>
                    </form>
                  </div>
                </div>
              </div>
            </body>
</html>
```

FEES COLLECTION MODULE

```
<html class="no-js" lang="en">
<head>
  <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
  <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet"></link>
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-
validator/0.11.9/validator.min.js"></script>
  <style>
    body {
      background: #EECDA3;
      background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
      background: linear-gradient(to top, #EF629F, #EECDA3);
    }
    .container {
      max-width: 750px;
    }
    .has-error label,
    .has-error input,
    .has-error textarea {
      color: red;
      border-color: red;
    }
    .list-unstyled li {
      font-size: 13px;
      padding: 4px 0 0;
      color: red;
    }
  </style>
</head>
<body>
  <div class="container mt-5">
    <div class="card">
      <h5 class="card-header text-center">Fees Details Form</h5>
      <div class="card-body">
        <form role="form" data-toggle="validator">
          <div class="form-group">
            <label>AppNo</label>
            <input type="number" class="form-control" name="num" maxlength="10"
minlength="5"
              pattern="/^[0-9]+$/" id="AppNo" placeholder="Number" required>
            <!-- Error -->
            <div class="help-block with-errors"></div>
          </div>
          <div class="form-group">
            <label>CourseName</label>
          <input type="text" class="form-control" name="Cname" maxlength="10" minlength="5"
              pattern="^[a-zA-Z0-9_.-]*$" id="CName" placeholder="Course" required>
            <div class="help-block with-errors"></div>
          </div>
```

```
            <div class="form-group">
              <label>Course Fee</label>
              <input type="text" class="form-control" name="ColName" maxlength="18"
minlength="10"
                pattern="^[a-zA-Z0-9_.-]*$" id="Colname" placeholder="ColegeName" required>
              <!-- Error -->
              <div class="help-block with-errors"></div>
            </div>
             <div class="form-group">
              <label>Payment Type</label>
              <select  class="custom-select"  id="dropdegree"   pattern="  /^(!?--\s*[Ss]elect\s*--)$/"
required >
            <option>FullPay</option>
            <option>Installment</option>
          </select>
              <!-- Error -->
              <div class="help-block with-errors"></div>
            </div>

            <div class="form-group">
              <label>InstallmentNo</label>
              <input type="text" class="form-control"  maxlength="3" minlength="3"
                pattern="/^[0-9]+$/" id="InstalNo" placeholder="InstallmentNo" required>

              <!-- Error -->
              <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
              <label>ReceiptNo</label>
              <input type="number" class="form-control"  maxlength="3" minlength="2"
                pattern="/^[0-9]+$/" id="Receipt" placeholder="ReceiptNo" required>

              <!-- Error -->
              <div class="help-block with-errors"></div>
            </div>
            <div class="form-group">
              <label><b>Mode Of payment</b></label>
               <div class="form-check form-check-inline" required>
             <input class="form-check-input" type="radio"  id="Radio"/>
            <label class="form-check-label" for="radio">Cash</label>
            </div>
           <div class="form-check form-check-inline">
             <input class="form-check-input" type="radio"  id="Radio1"/>
            <label class="form-check-label" for="radio1">Online</label>
            </div>

              <!-- Error -->
              <div class="help-block with-errors"></div>
            </div>

            <div class="form-group">
              <label>ReferrenceNo</label>
```

```
                <input type="text" class="form-control" name="" maxlength="10" minlength="10"
                  pattern="/^[0-9]+$/" id="Rno" placeholder="ReferrenceNo" required>

                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>

            <div class="form-group">
                <label>Total</label>
                <input type="number" class="form-control" id="Total" placeholder="Total" required>

                <!-- Error -->
                <div class="help-block with-errors"></div>
              </div>
              <div class="form-group">
                <button type="submit" class="btn btn-primary btn-block">Submit</button>
              </div>
            </form>
          </div>
        </div>
    </div>
</body>

</html>
```

ASSESSMENT MODULE

```
    <html class="no-js" lang="en">
    <head>
      <title> Bootstrap 4 Form Validation with Validator.js Example | positronx.io</title>
      <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet"></link>
      <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
      <script src="https://cdnjs.cloudflare.com/ajax/libs/1000hz-bootstrap-
validator/0.11.9/validator.min.js"></script>

      <style>
        body {
          background: #EECDA3;
          background: -webkit-linear-gradient(to top, #EF629F, #EECDA3);
          background: linear-gradient(to top, #EF629F, #EECDA3);
        }

        .container {
          max-width: 750px;
        }

        .has-error label,
        .has-error input,
        .has-error textarea {
          color: red;
          border-color: red;
        }
```

```
            .list-unstyled li {
               font-size: 13px;
               padding: 4px 0 0;
               color: red;
            }
         </style>
      </head>

      <body>
         <div class="container mt-5">
            <div class="card">
               <h5 class="card-header text-center">AssessmentForm</h5>
               <div class="card-body">
                  <form role="form" data-toggle="validator">


                     <div class="form-group">
                        <label>AppNo</label>
                        <input type="number" class="form-control" data-error="You must have a name."
id="AppNo" placeholder="AppNo" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                     </div>

                     <div class="form-group">
                        <label>Name</label>
                        <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                           pattern="^[a-zA-Z0-9_.-]*$" id="name" placeholder="Name" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                     </div>
                     <div class="form-group">
                        <label>Course</label>
                        <input type="text" class="form-control" name="name" maxlength="10"
minlength="5"
                           pattern="^[a-zA-Z0-9_.-]*$" id="Course" placeholder="Course" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
                     </div>
                     <div class="form-group">
                        <label>AssignmentNo</label>
                        <input type="number" class="form-control" name="name" maxlength="3"
minlength="2"
                           pattern="/^[0-9]+$/" id="ANo" placeholder="No" required>

                        <!-- Error -->
                        <div class="help-block with-errors"></div>
```

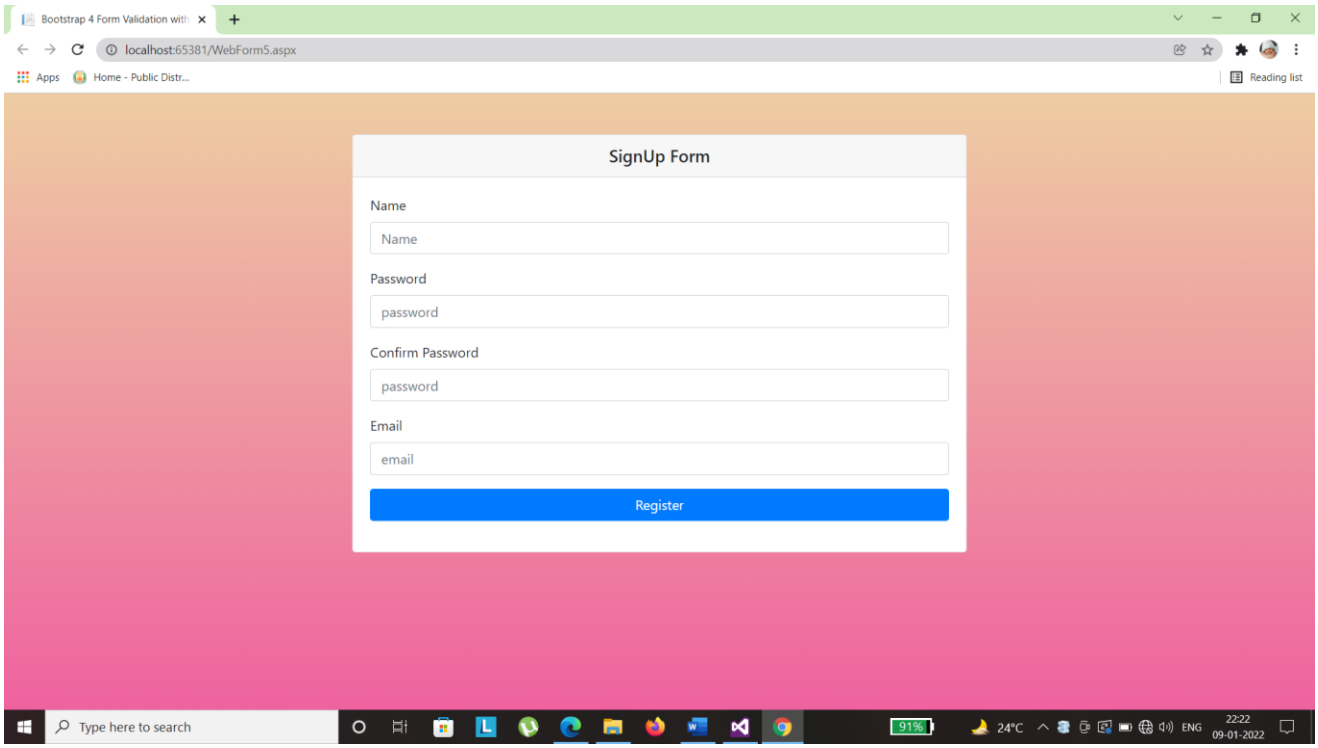
```
                </div>
                <div class="form-group">
                  <label>Assignment Date</label>
                  <input type="date" class="form-control" name="date" maxlength="10"
minlength="8"
                    pattern="^([0-2][0-9]|(3)[0-1])(\/)(((0)[0-9])|((1)[0-2]))(\/)\d{4}$" id="Adate"
placeholder="dd/mm/yy" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Assignment Title</label>
                  <input type="text" class="form-control" name="ATitle" maxlength="18"
minlength="10"
                    pattern="^[a-zA-Z0-9_.-]*$" id="ATitle" placeholder="Title" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Assignment Mark</label>
                  <input type="text" class="form-control" name="AMark" maxlength="3"
minlength="3"
                    pattern="^[a-zA-Z0-9_.-]*$" id="AMark" placeholder="Mark" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Practical No</label>
                  <input type="number" class="form-control" name="No" maxlength="3"
minlength="2"
                    pattern="/^[0-9]+$/" id="PNo" placeholder="Number" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Practical Date</label>
                  <input type="date" class="form-control" name="date" maxlength="10"
minlength="3"
                    pattern="^([0-2][0-9]|(3)[0-1])(\/)(((0)[0-9])|((1)[0-2]))(\/)\d{4}$" id="Pdate"
placeholder="dd/mm/yy" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Practical Title</label>
                  <input type="text" class="form-control" name="ATitle" maxlength="10"
minlength="3"
```

```
                    pattern="^[a-zA-Z0-9_.-]*$" id="PTitle" placeholder="Title" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Practical Mark</label>
                  <input type="text" class="form-control" name="PMark" maxlength="3"
minlength="2"
                    pattern="^[a-zA-Z0-9_.-]*$" id="PMark" placeholder="Mark" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
              <div class="form-group">
                  <label>Test No</label>
                  <input type="number" class="form-control" name="no" maxlength="10"
minlength="3"
                    pattern="/^[0-9]+$/" id="TNo" placeholder="No" required>
                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Test Date</label>
                  <input type="date" class="form-control" name="date" maxlength="10"
minlength="3"
                    pattern="^([0-2][0-9]|(3)[0-1])(\/)(((0)[0-9])|((1)[0-2]))(\/)\d{4}$" id="Tdate"
placeholder="dd/mm/yy" required>
                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Test Title</label>
                  <input type="text" class="form-control" name="TTitle" maxlength="18"
minlength="8"
                    pattern="^[a-zA-Z0-9_.-]*$" id="TTitle" placeholder="Title" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>
                <div class="form-group">
                  <label>Test Mark</label>
                  <input type="text" class="form-control" name="TMark" maxlength="2"
minlength="3"
                    pattern="^[a-zA-Z0-9_.-]*$" id="TMark" placeholder="Mark" required>

                  <!-- Error -->
                  <div class="help-block with-errors"></div>
                </div>


                <div class="form-group">
```

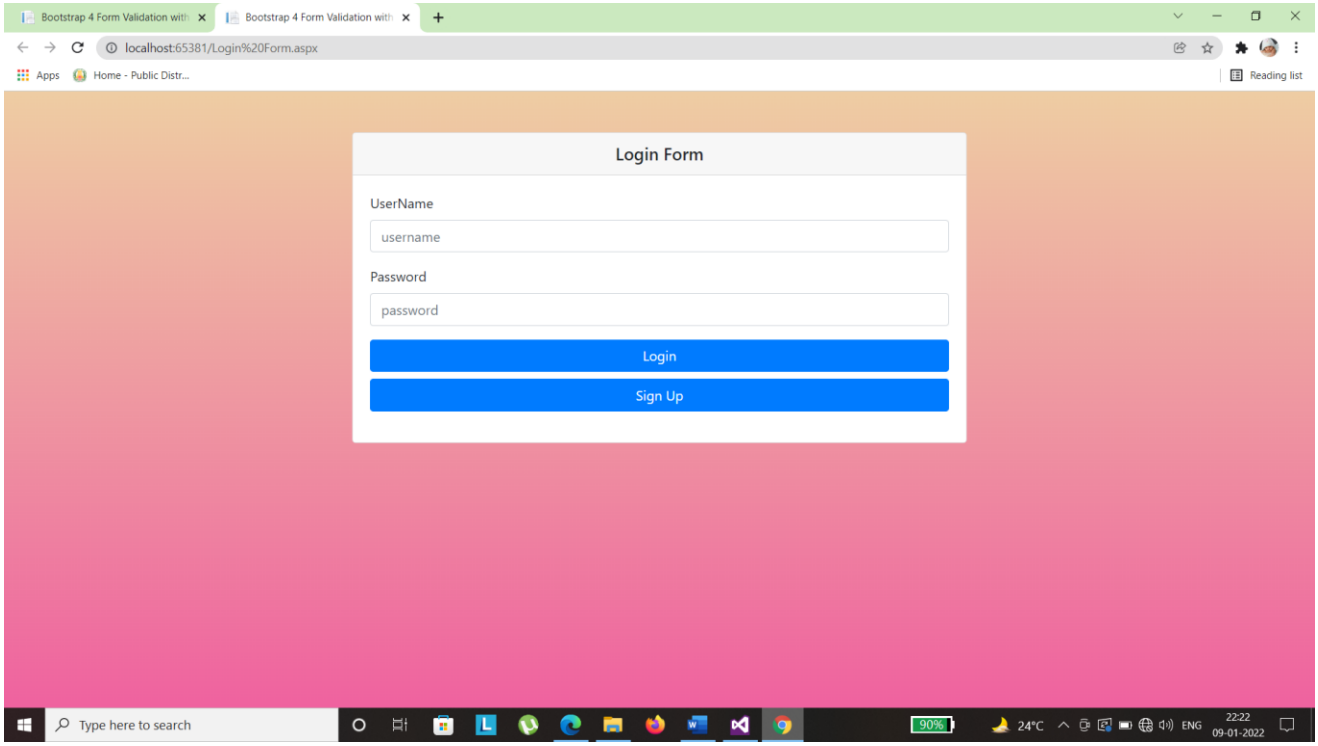
```
<button type="submit" class="btn btn-primary btn-block">Submit</button>
                </div>
            </form>
        </div>
    </div>
</body>
</html>
```

10.2 SCREENSHOTS

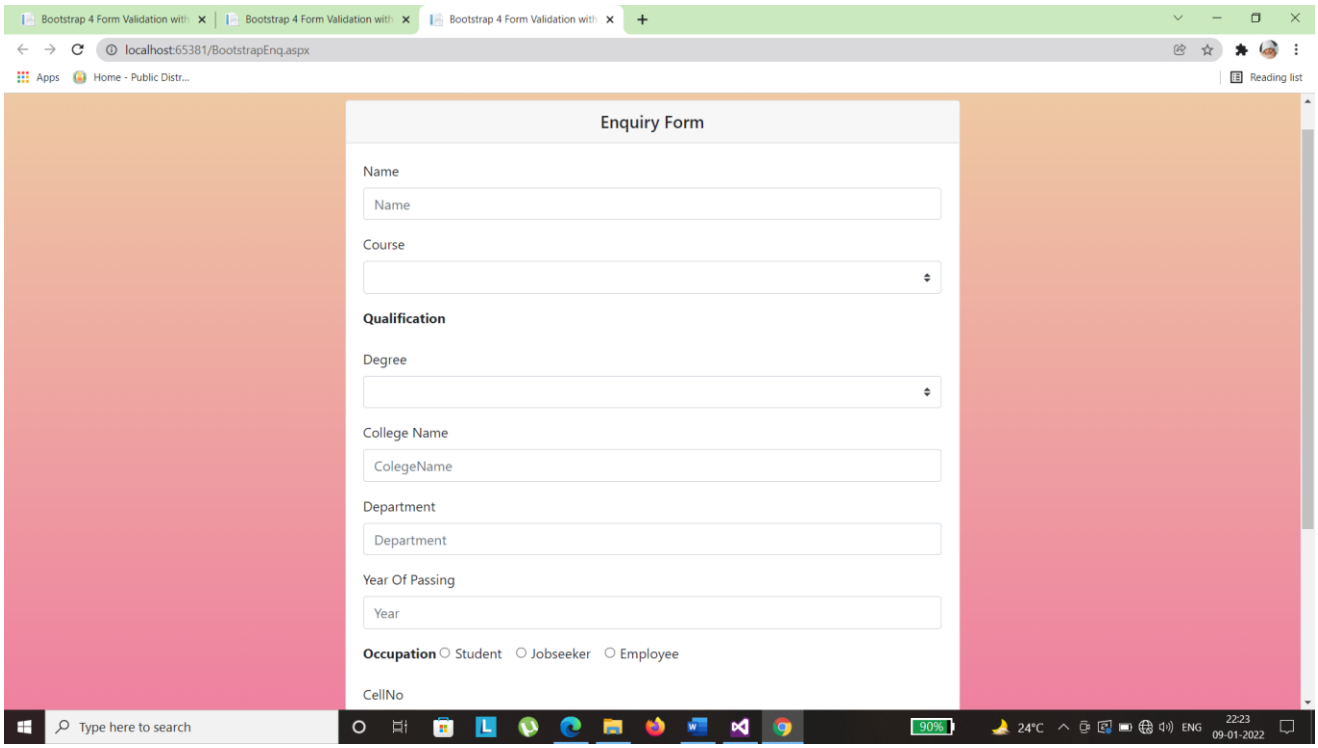
SIGN UP FORM

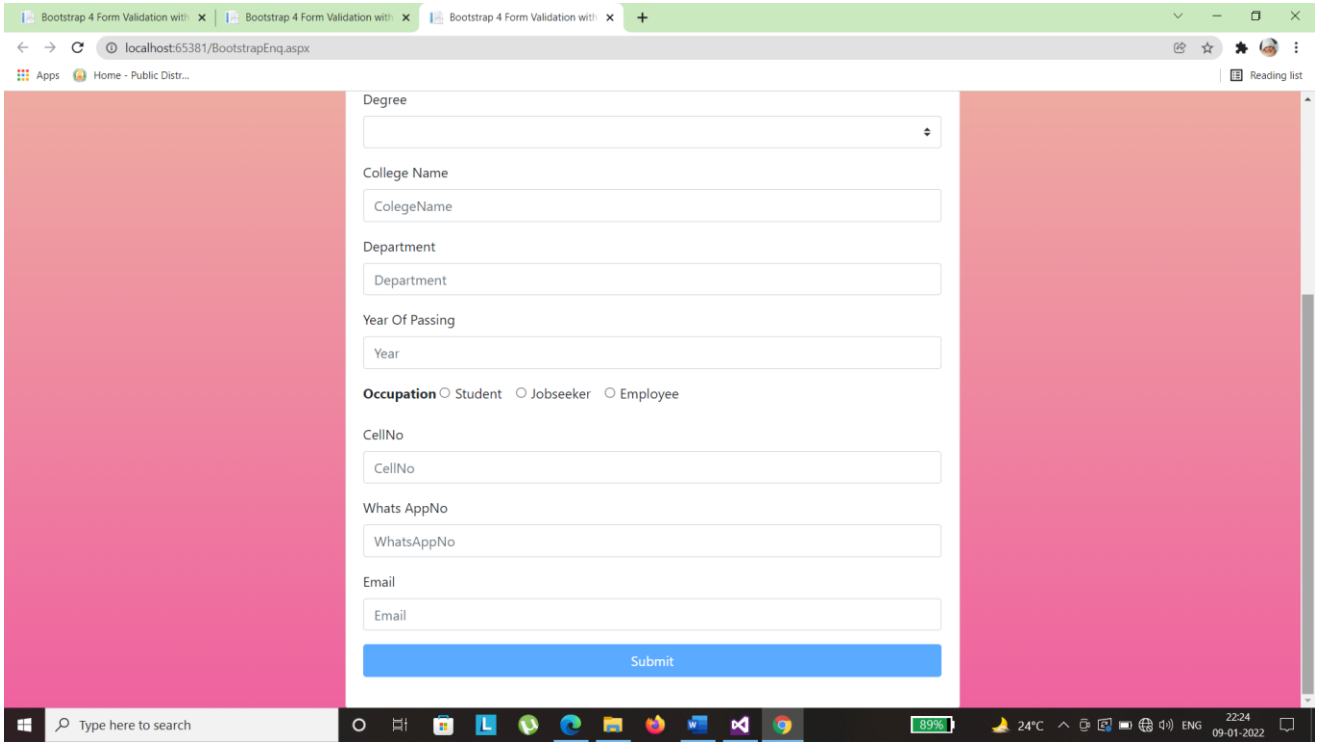


LOGIN FORM

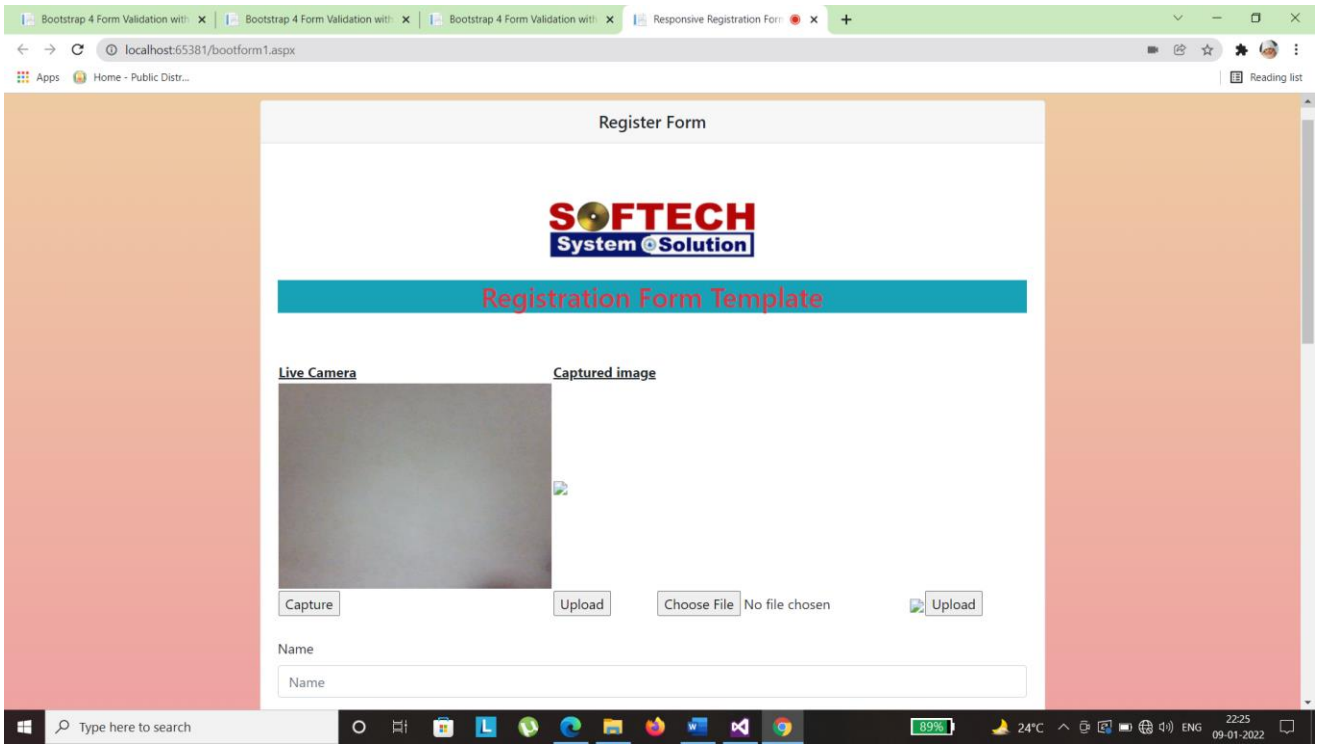


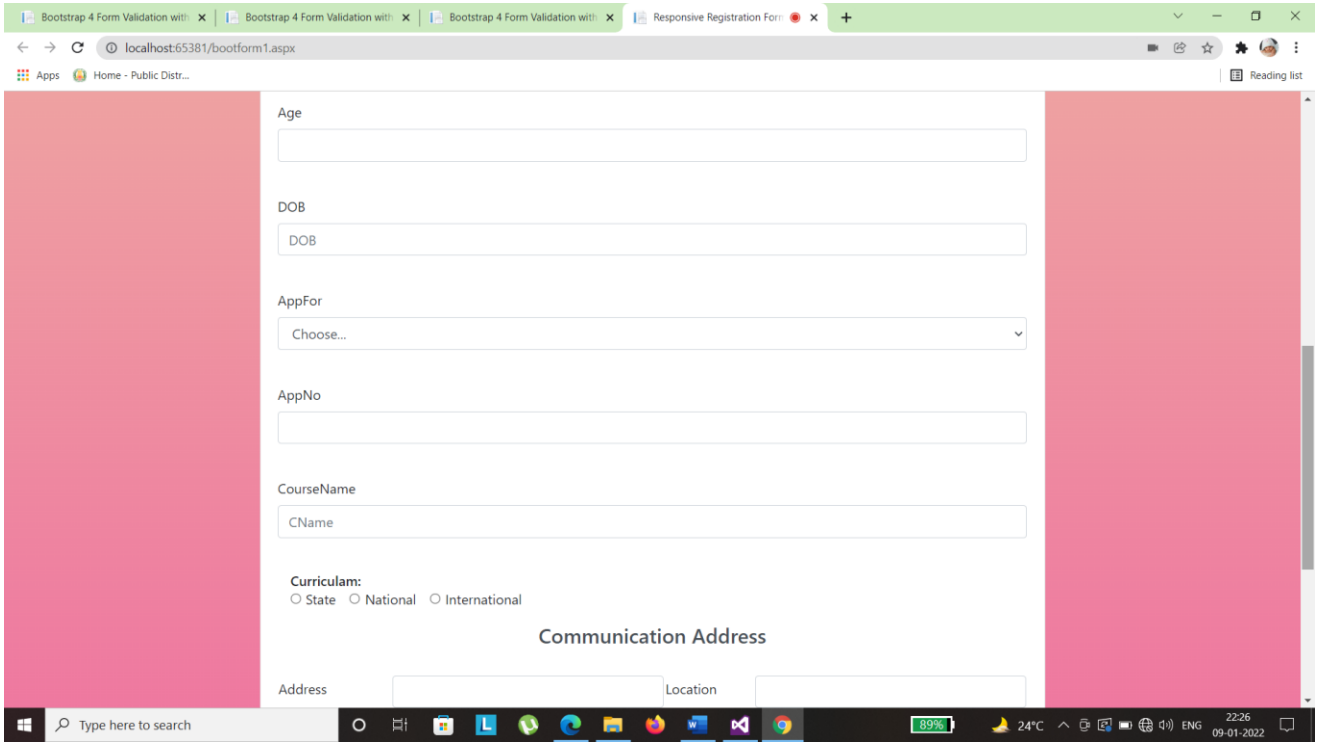
ENQUIRY FORM

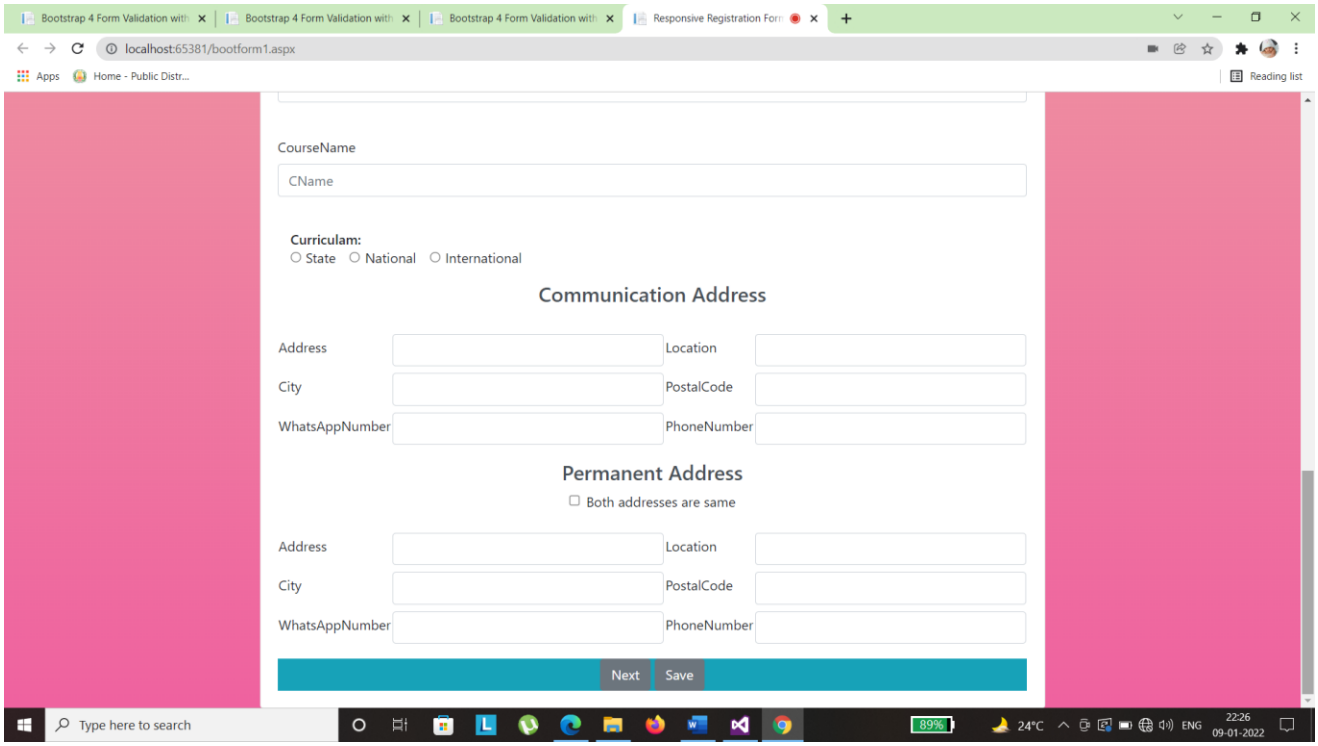




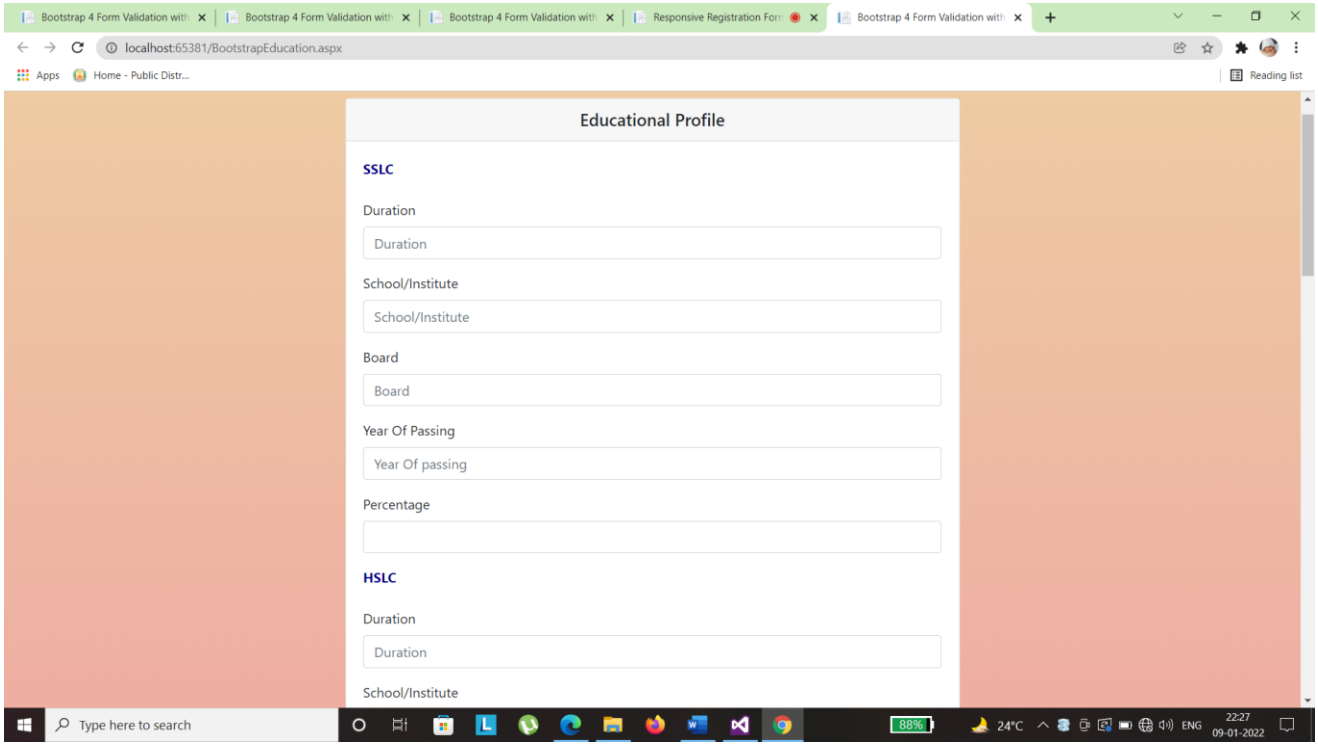
REGISTRATION FORM

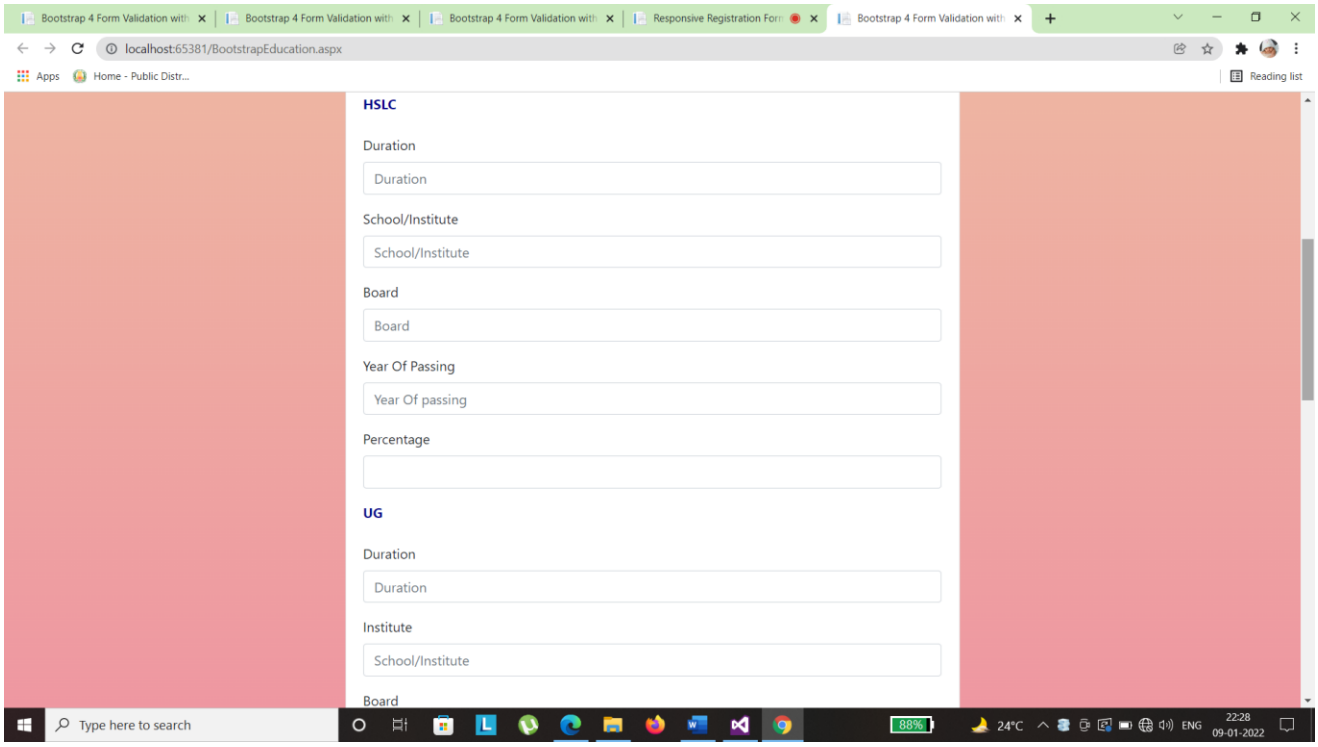


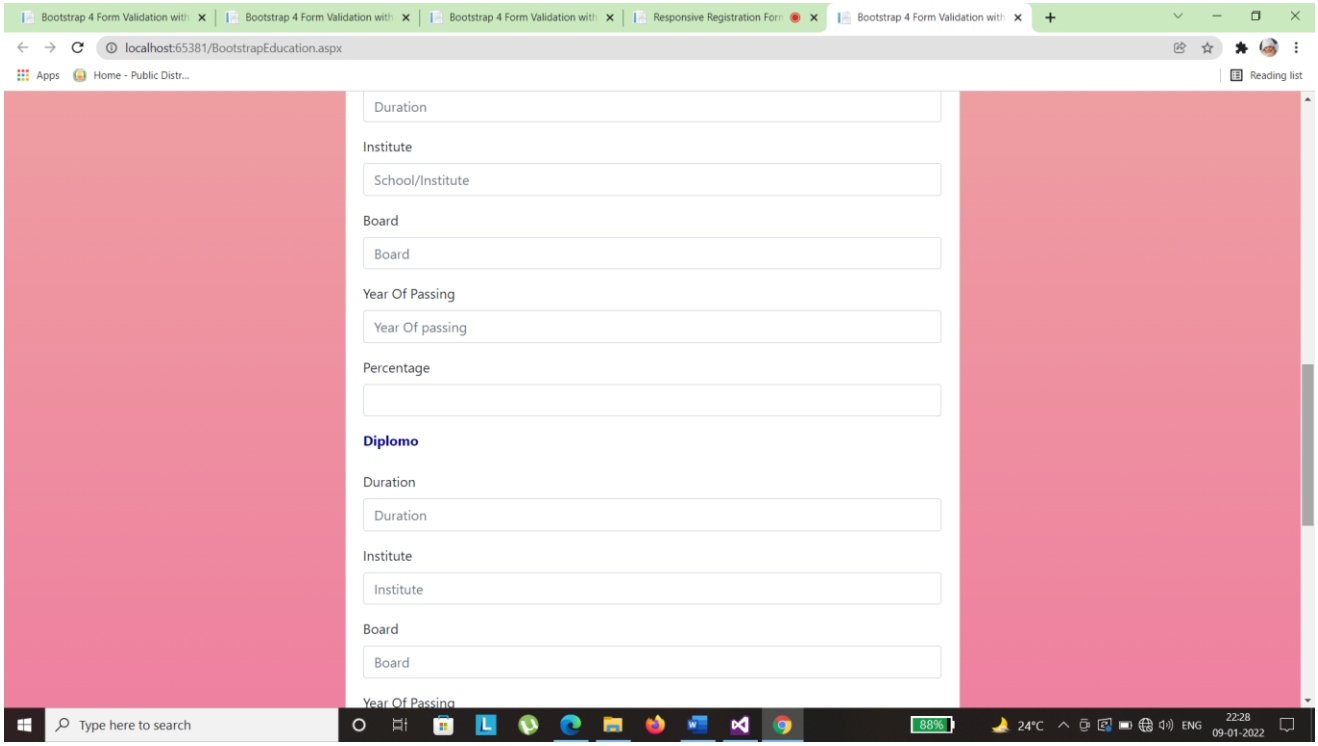

Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Responsive Registration Form
localhost:65381/bootform1.aspx
Apps
Home - Public Distr...
Reading list
Age
DOB
DOB
AppFor
Choose...
AppNo
CourseName
CName
Curriculam:
State
National
International
Communication Address
Address
Location
Type here to search
89%
24°C
ENG
22:26
09-01-2022

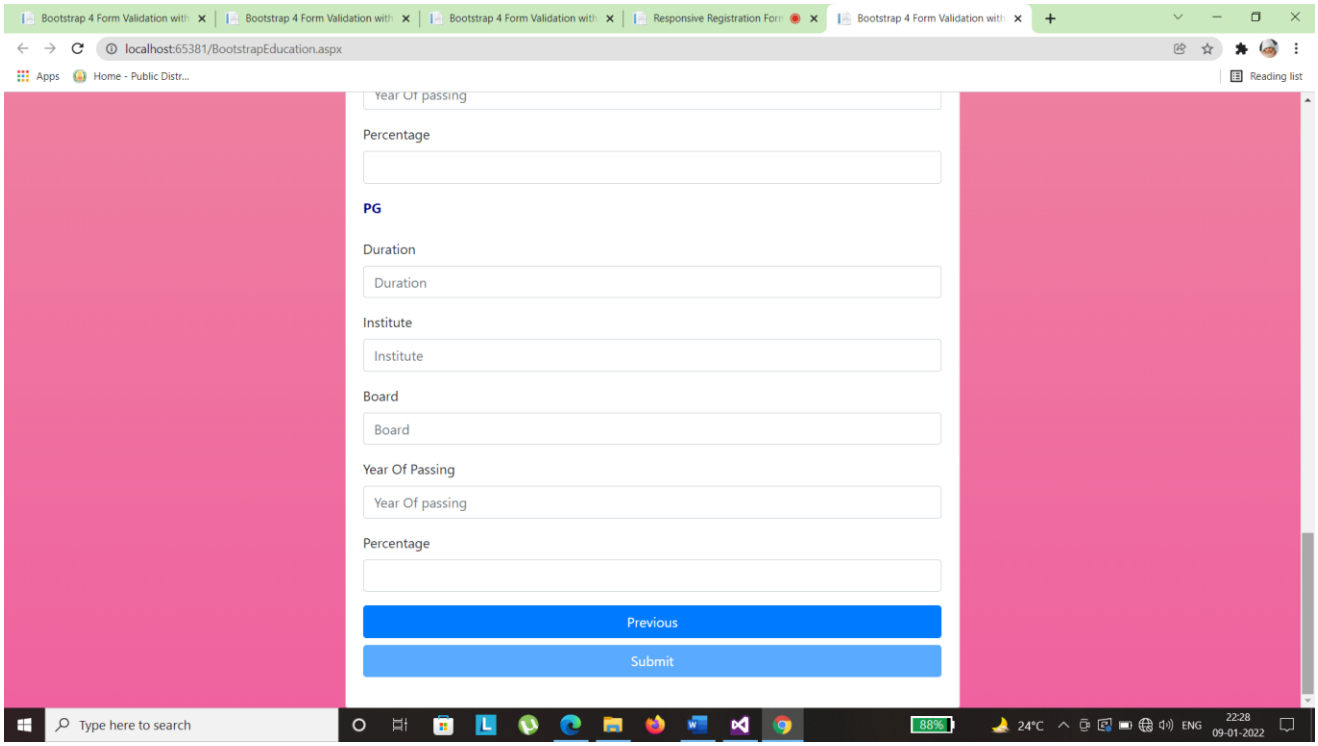

Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Responsive Registration Form
localhost:65381/bootform1.aspx
Apps
Home - Public Distr...
Reading list
CourseName
CName
Curriculam:
State
National
International
Communication Address
Address
Location
City
PostalCode
WhatsAppNumber
PhoneNumber
Permanent Address
Both addresses are same
Address
Location
City
PostalCode
WhatsAppNumber
PhoneNumber
Next
Save
Type here to search
89%
24°C
ENG
22:26
09-01-2022

Educational Profile

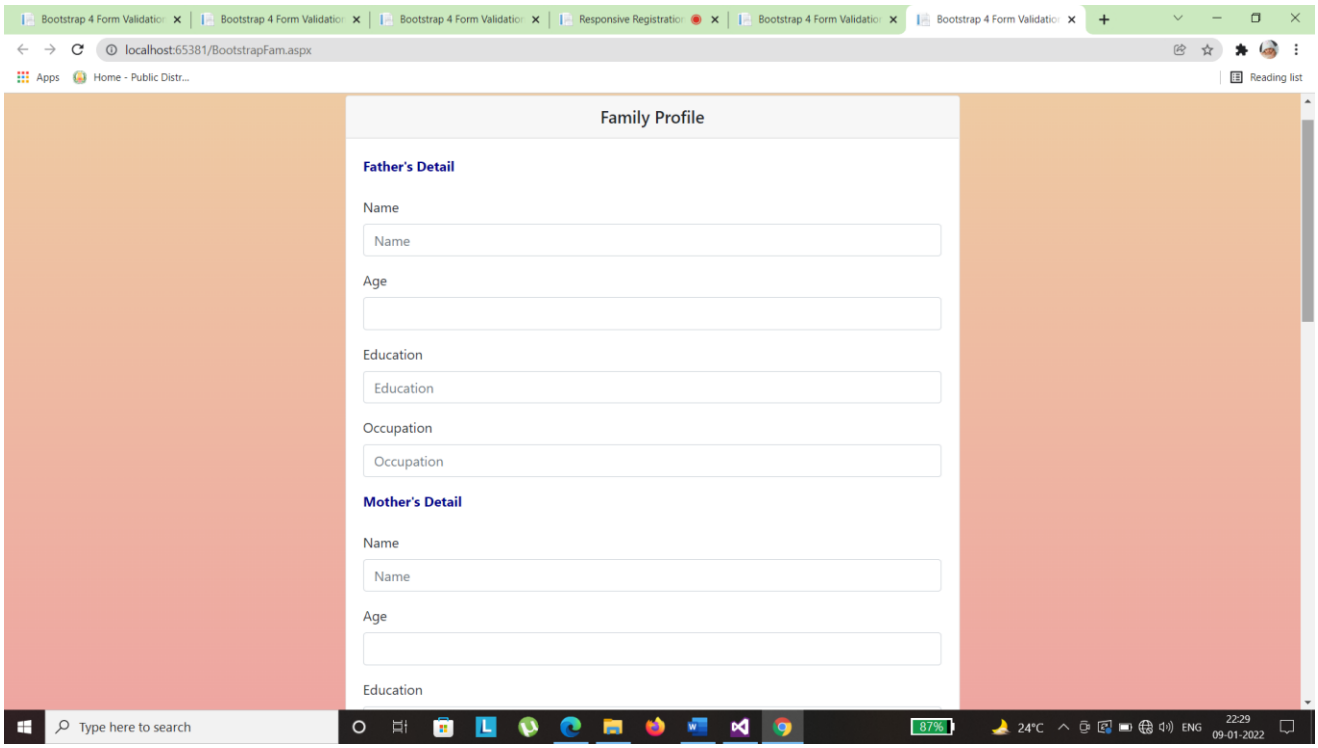


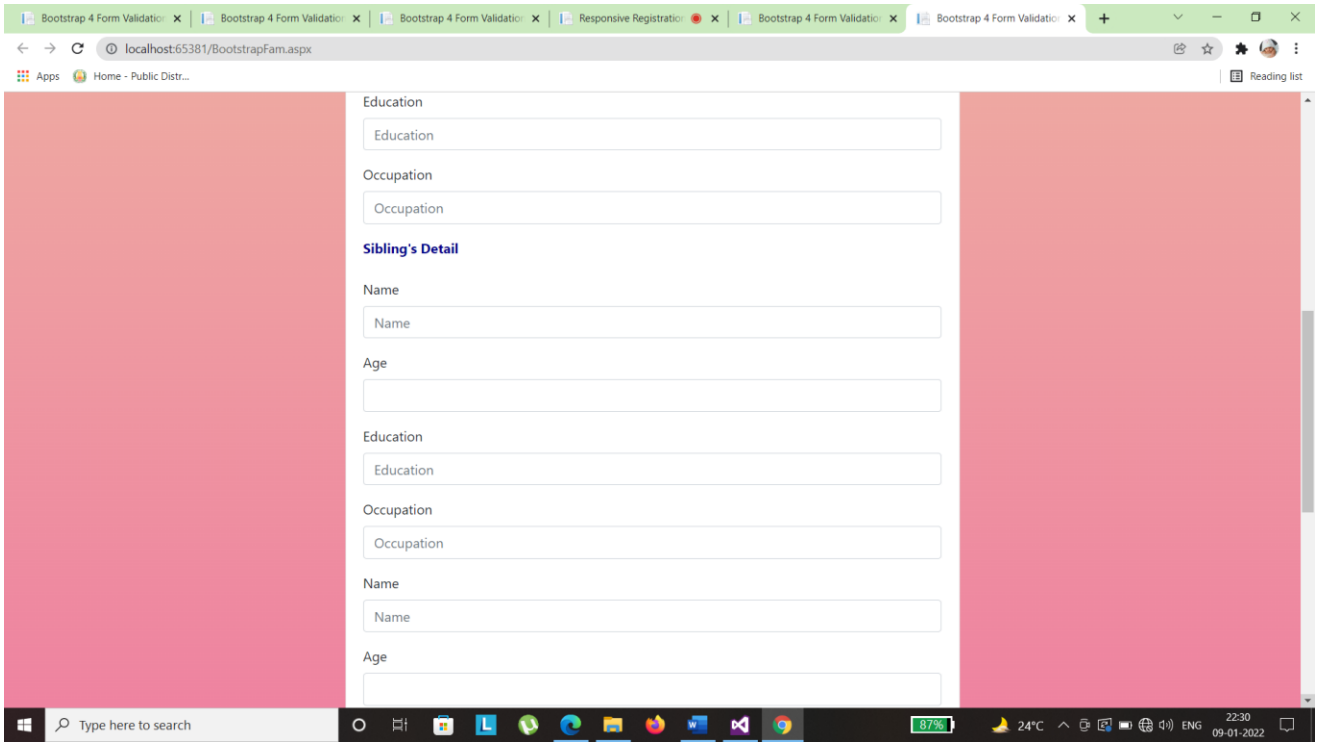


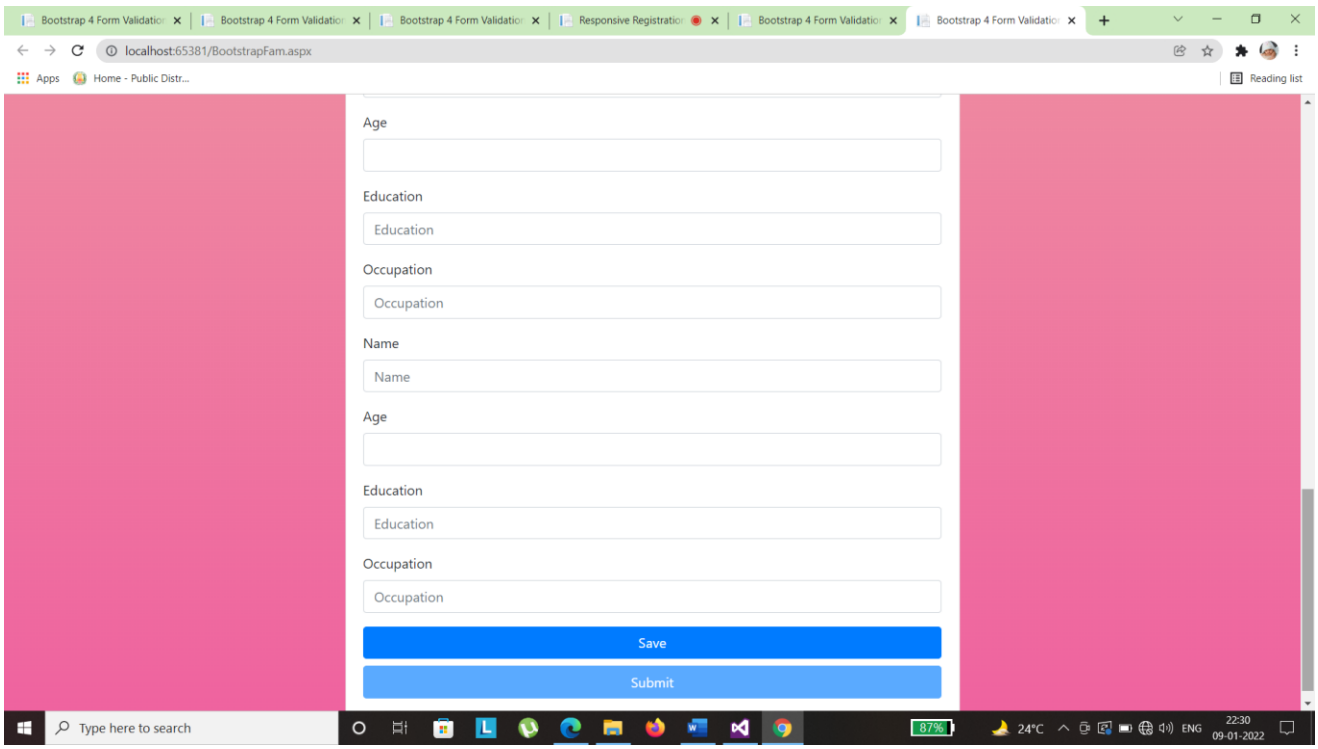

Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Responsive Registration Form
Bootstrap 4 Form Validation with
localhost:65381/BootstrapEducation.aspx
Apps
Home - Public Distr...
Reading list
Duration
Institute
School/Institute
Board
Board
Year Of Passing
Year Of passing
Percentage
Diplomo
Duration
Duration
Institute
Institute
Board
Board
Year Of Passing
Type here to search
88%
24°C
ENG
22:28
09-01-2022


Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Bootstrap 4 Form Validation with
Responsive Registration Form
Bootstrap 4 Form Validation with
localhost:65381/BootstrapEducation.aspx
Apps
Home - Public Distr...
Reading list
Year Of passing
Percentage
PG
Duration
Duration
Institute
Institute
Board
Board
Year Of Passing
Year Of passing
Percentage
Previous
Submit
Type here to search
88%
24°C
ENG
22:28
09-01-2022

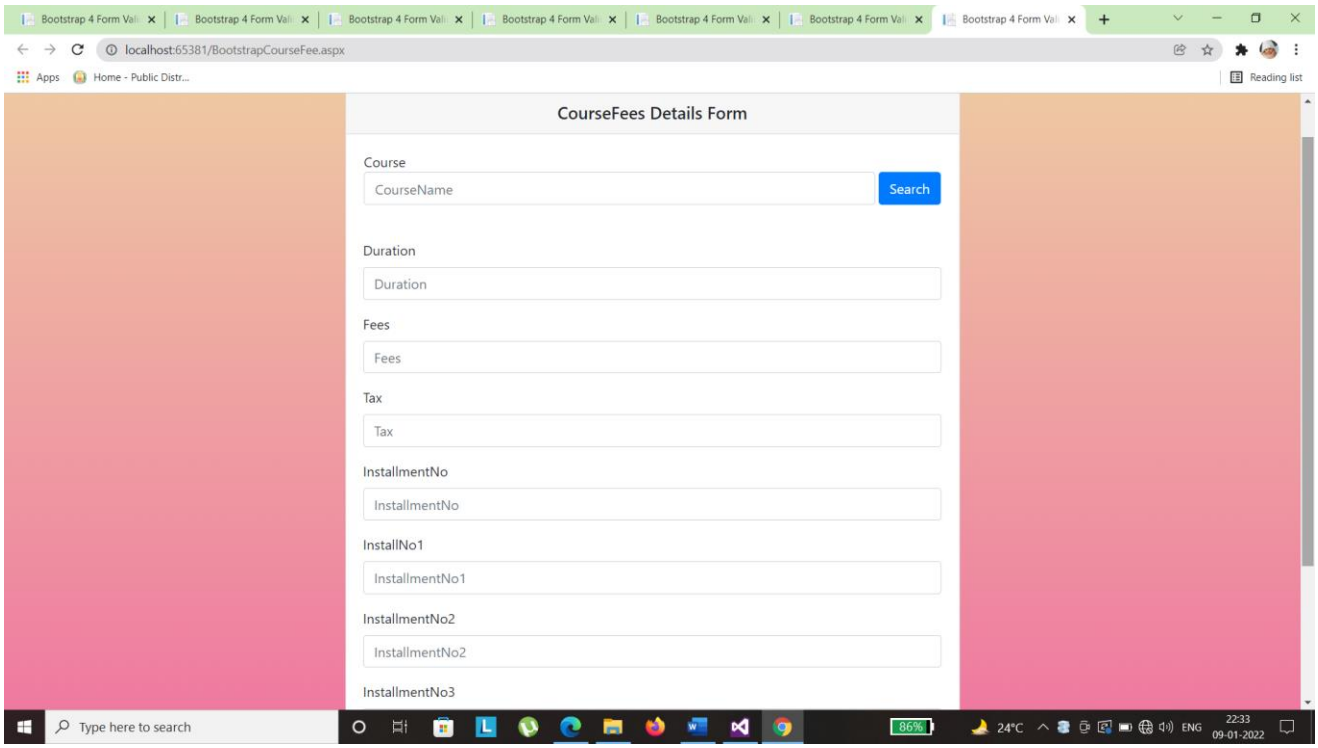
FAMILY PROFILE

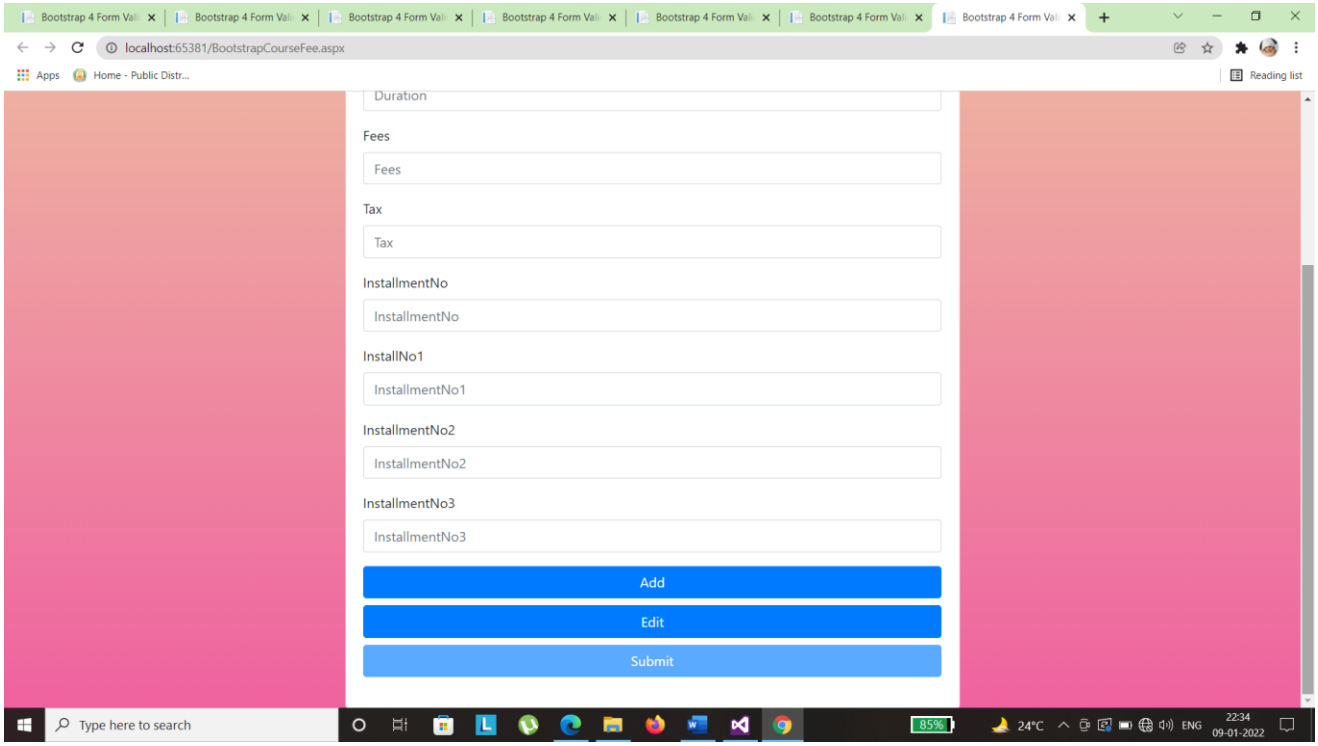




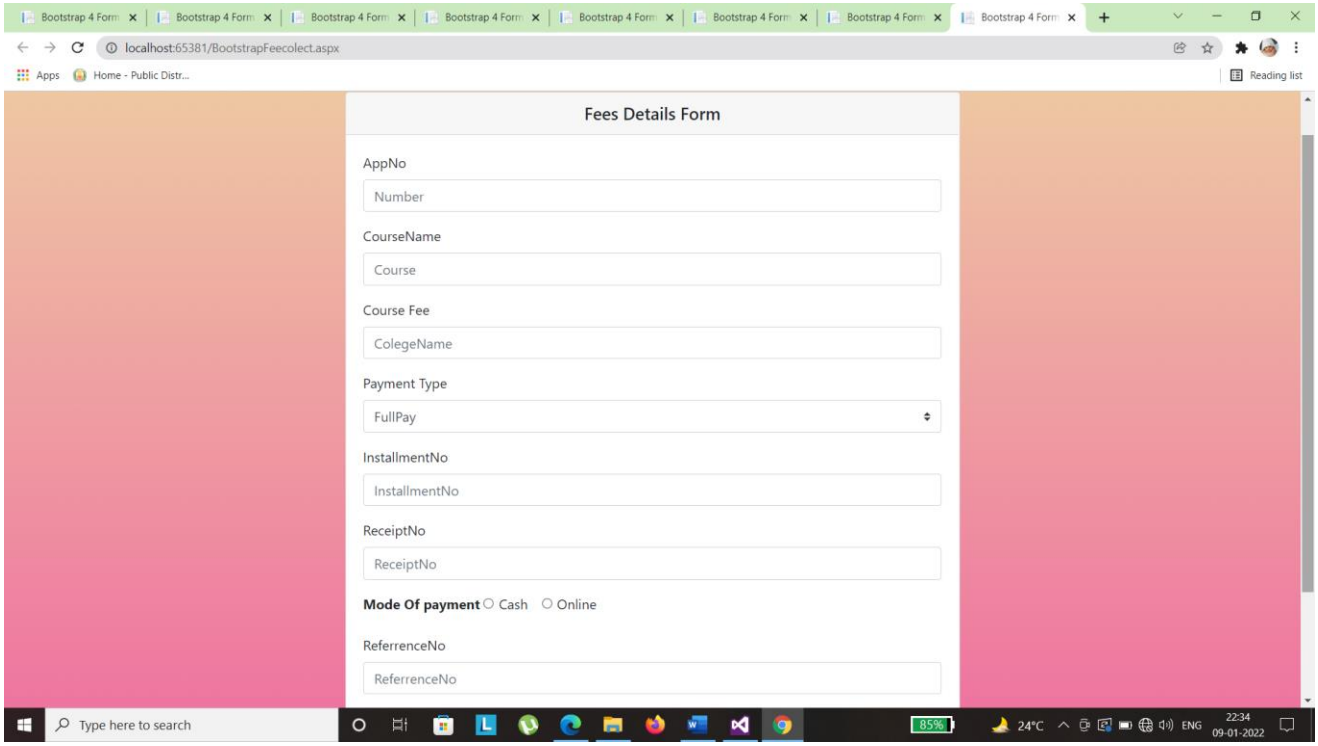

Bootstrap 4 Form Validation
localhost:65381/BootstrapFam.aspx
Apps
Home - Public Distr...
Reading list
Age
Education
Education
Occupation
Occupation
Name
Name
Age
Education
Education
Occupation
Occupation
Save
Submit
Type here to search
87%
24°C
ENG
22:30
09-01-2022

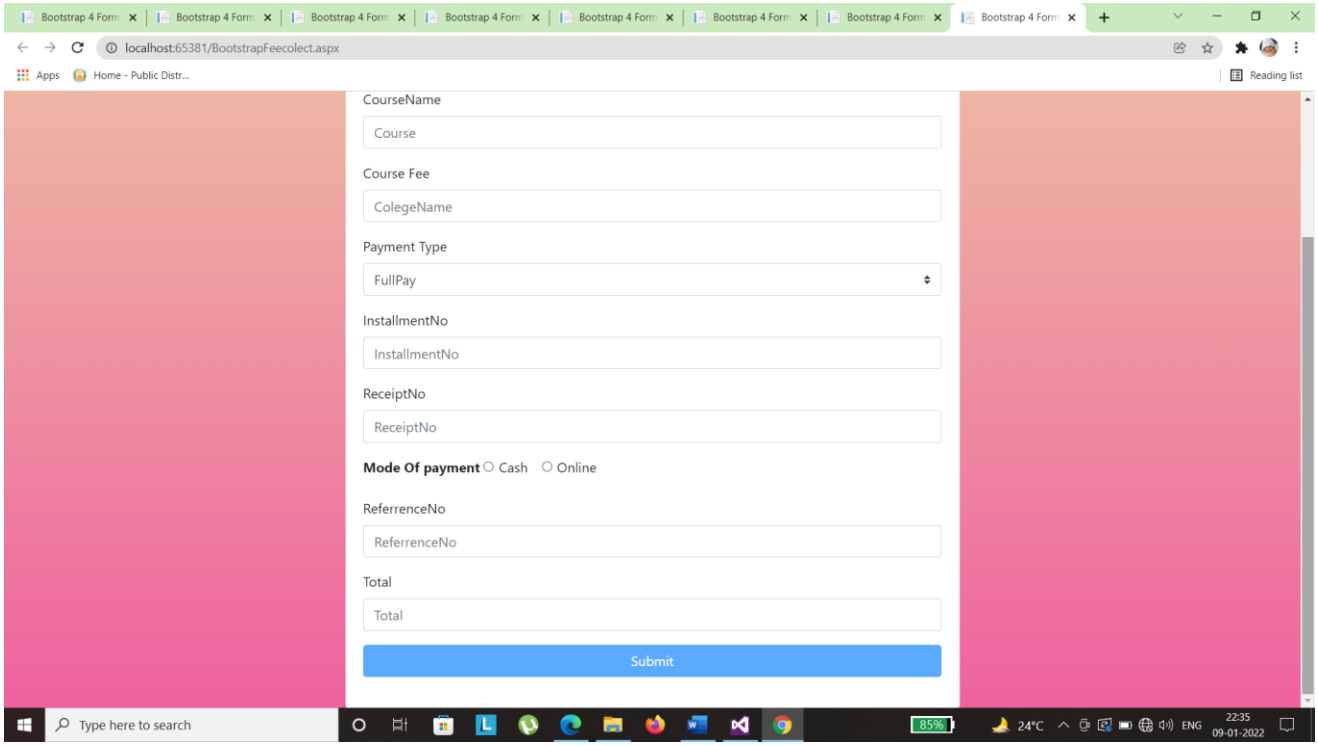
COURSEFEES


Bootstrap 4 Form Validation
localhost:65381/BootstrapCourseFee.aspx
Apps
Home - Public Distr...
Reading list
CourseFees Details Form
Course
CourseName
Search
Duration
Duration
Fees
Fees
Tax
Tax
InstallmentNo
InstallmentNo
InstallNo1
InstallmentNo1
InstallmentNo2
InstallmentNo2
InstallmentNo3
Type here to search
86%
24°C
ENG
22:33
09-01-2022

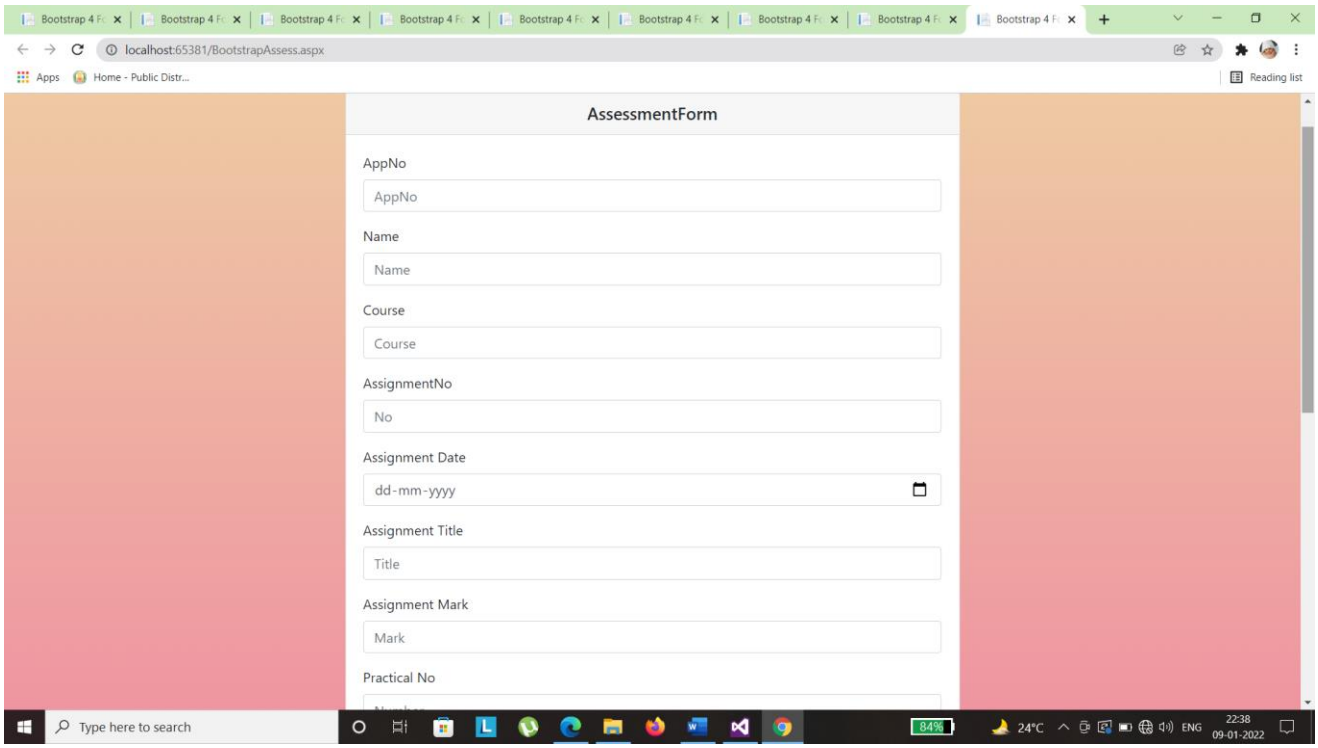


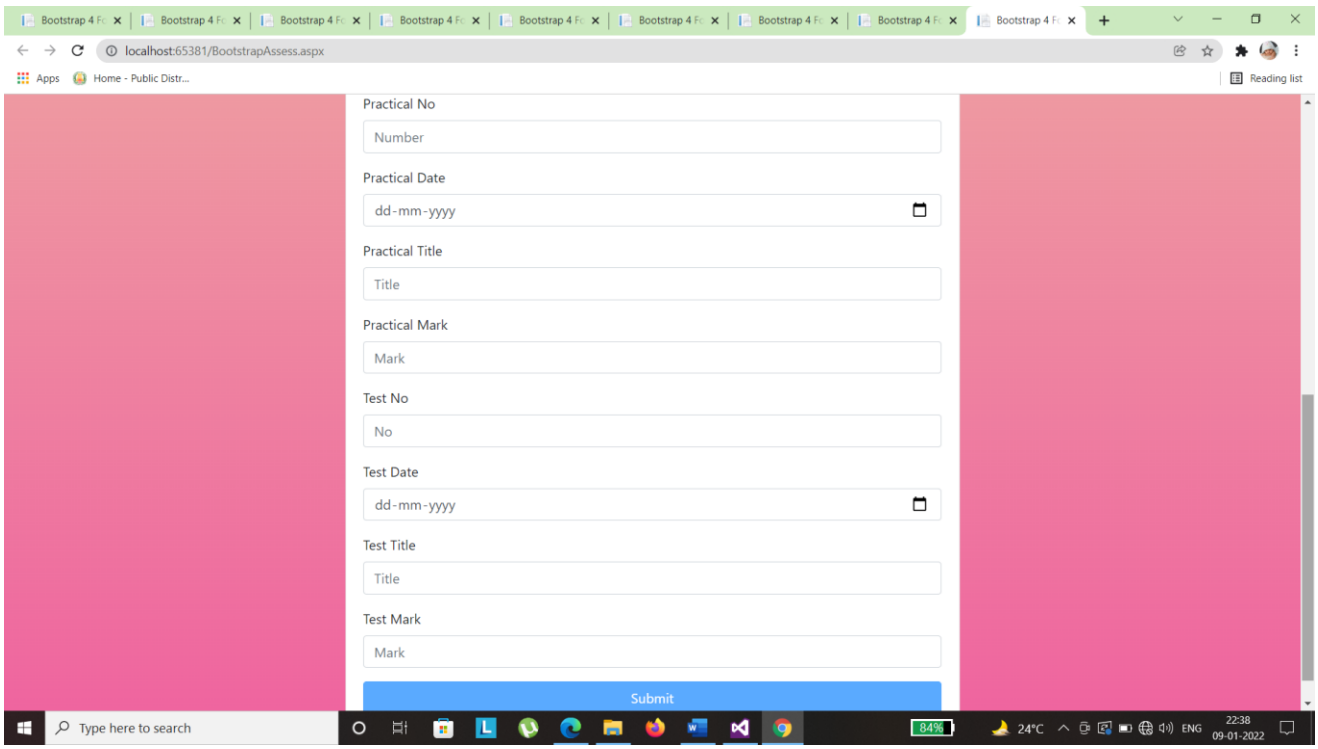
FEES COLLECTION




localhost:65381/BootstrapFeecolect.aspx
CourseName
Course
Course Fee
ColegeName
Payment Type
FullPay
InstallmentNo
InstallmentNo
ReceiptNo
ReceiptNo
Mode Of payment Cash Online
ReferrenceNo
ReferrenceNo
Total
Total
Submit

ASSESSMENT


localhost:65381/BootstrapAssess.aspx
AssessmentForm
AppNo
AppNo
Name
Name
Course
Course
AssignmentNo
No
Assignment Date
dd-mm-yyyy
Assignment Title
Title
Assignment Mark
Mark
Practical No


localhost:65381/BootstrapAssess.aspx
Practical No
Number
Practical Date
dd-mm-yyyy
Practical Title
Title
Practical Mark
Mark
Test No
No
Test Date
dd-mm-yyyy
Test Title
Title
Test Mark
Mark
Submit
Type here to search

CHAPTER-11

BIBLIOGRAPHY

1. Raza SA, Qazi W, Khan KA, Salam J (April 2021). "Social Isolation and Acceptance of the Learning Management System (LMS) in the time of COVID-19 Pandemic: An Expansion of the UTAUT Model". Journal of Educational Computing Research. **59** (2): 183–208. doi:10.1177/073563312096042. ISSN 0735-6331. PMC 7509242.
2. Prat, Joana; Llorens, Ariadna; Salvador, Francesc; Alier, Marc; Amo, Daniel (6 May 2021). "A Methodology to Study the University's Online Teaching Activity from Virtual Platform Indicators: The Effect of the Covid-19 Pandemic at Universitat Politècnica de Catalunya". Sustainability. **13**: 5177 – via MDPI Open Access Journals.
3. Toquero, Cathy Mae (16 April 2020). "Challenges and Opportunities for Higher Education amid the COVID-19 Pandemic: The Philippine Context". Pedagogical Research. **5** (4): em0063. doi:10.29333/pr/7947. ISSN 2468-4929.
4. Jones, Kyle M. L. (2 July 2019). "Learning analytics and higher education: a proposed model for establishing informed consent mechanisms to promote student privacy and autonomy". International Journal of Educational Technology in Higher Education.
5. Lin, Sandi (16 November 2015). "SaaS Learning Management System: Is your LMS Truly SaaS? - eLearning Industry". E-Learning Industry. Retrieved 4 February 2017.
6. Ellis, Ryann K. (2009), Field Guide to Learning Management, ASTD Learning Circuits, archived from the original on 24 August 2014, retrieved 5 July 2012

.