4. Apply Partitioning Methods, Hierarchical Methods, Density-Based Methods for clustering on a data set and compare the performance of the obtained results using different metrics
5. Create an ensemble using Random Forest and show the impact of bagging and boosting on the performance
6. Use Naive bayes, K-nearest, and Decision tree classification algorithms and build classifiers on any two datasets. Divide the data set into training and test set. Compare the accuracy of the different classifiers under the following situations:
   I. a) Training set = 75% Test set = 25% b) Training set = 66.6% (2/3rd of total), Test set = 33.3%
   II. Training set should be chosen by i) hold out method ii) Random subsampling iii) Cross-Validation. Compare the accuracy of the classifiers obtained.
   III. Data should be scaled to the standard format.

Project

Students should be promoted to take up one project on any UCI/kaggle/data.gov.in or on a dataset verified by the teacher. Preprocessing steps and at least one data mining technique should be shown on the selected dataset. This will allow the students to have practical knowledge of how to apply the various skills learned in the subject to a single problem/project.

DISCIPLINE SPECIFIC Elective (DSE 03b): Web Design and Development

Semester 5

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 03b Web Design and Development	**4**	**3**	**0**	**1**	Pass in Class XII	Knowledge of Structured Query Language (SQL)

Learning Objectives

The course aims at introducing the basic concepts and techniques of complete website-based programming. The student shall be able to develop simple, interactive, and dynamic websites using HTML, Javascript, PHP and database.

Learning outcomes

On successful completion of this course, the student will be able to:

1. Build interactive and dynamic websites.
2. Use the client-side validation techniques using Javascript.
3. Write the server-side programming techniques with PHP for accessing the contents to/from the server.
4. Use GET and POST methods for exchanging data between client and server.
5. Learn to connect PHP with databases, save and retrieve data dynamically.

SYLLABUS OF DSE 03b

Unit 1. Introduction (2 hours)

Introduction to internet and web design. Basic concepts of web architecture.

Unit 2. HTML (11 hours)

Introduction to hypertext mark-up language (HTML), creating web pages, lists, elements of HTML, hyperlinks, tables, forms, inserting images.

Unit 3. Basics of Javascript (12 hours)

Document object model, data types and variables, functions, methods, and events, controlling program flow, client-side form validation.

Unit 4. Introduction to PHP (8 hours)

Basic syntax, defining variables and constants, data types, operators and expression, decision making statements, loop constructs, functions.

Unit 5. Handling HTML Form with PHP (6 hours)

Connecting an HTML form with PHP, submitting data to the server using GET and POST methods, GET vs POST methods.

Unit 6. Database Connectivity (6 hours)

Connectivity with database, database creation, creating tables, insertion and retrieval of the data from the database, data manipulation.

Essential/recommended readings

Suggested References

1. Nixon, R. **Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5,** 2018, O'Rielly.
2. Powell, T.A. (2017). **HTML & CSS: The Complete Reference**. 5th edition, 2017, Tata McGrawHill.
3. Duckett, J. **JavaScript and JQuery: Interactive Front-End Web Development**, 2014, Wiley.
4. Murach J., **Murach's PHP and MySQL,** 2nd Edition, 2014, Mike Murach & Associates.
5. Holzner S. **PHP: The Complete Reference,** 2017, McGraw Hill**.**
6. Ivan Bayross, **Web Enabled Commercial Application Development Using Html, Dhtml, Javascript, Perl CGI**, 2010, BPB Publications.

Suggested Practical List (If any): (30 Hours)

HTML

1. Create an HTML document with following formatting – Bold, Italics, Underline, Colors, Headings, Title, Font and Font Width, Background, Paragraph, Line Brakes, Horizontal Line, Marquee text.
2. Create an HTML document with Ordered and Unordered lists, Images, Internal and External linking.
3. Create an HTML document for displaying the current semester's timetable.
4. Create a student registration form using HTML which has the following controls:
    a. Text Box

b. Dropdown box
c. Option/radio button
d. Check boxes
e. Reset and Submit button.

Javascript

1. Write event-driven programs in JavaScript for the following:
    a. Enter a number and on click of a button print its factors.
    b. Print the smallest of five numbers entered by the user.
    c. Find the factorial of a number entered by the user.
    d. Take a number in an input text box, on click of a button, display its multiplication table.
2. Create a student registration form. Create functions to perform the following checks:
    a. Student id is a 10-digit alphanumeric value
    b. Name should be an alphabetical value (String)
    c. Non-empty fields like Age
3. Create a form containing various HTML elements and perform appropriate validations on each of them while submitting the form.

PHP

1. Write a PHP script to print the sum of even digits of a number.
2. Write a script in PHP to display a Multiplication Table.
3. Design a Student Registration form, using appropriate input fields consisting of following:
    a. Roll Number
    b. First Name
    c. Last Name
    d. Gender
    e. Department
    f. DOB

    Submit, retrieve the form data using the $_POST variable and display it.
4. Write PHP Code to create a database, connect to it, create tables, insert and access their contents.
5. Write PHP code to insert, delete, and retrieve data from the database. Create proper forms for performing the above operations. Display the messages such as "The record is added in the database!" when data is inserted into the database, "A record is deleted from the database" when data is deleted from the database. Use appropriate button names such as Add Data, Delete Data, and Display Data.