

**SEMESTER LEARNING PLAN
(RENCANA PEMBELAJARAN SEMESTER)**

No. Doc: SLP/ IT /I322-2BITMA

Course Name	: Server-side Web Programming
Course Code	: I322-2BITMA
Credit	: 4 SKS/ 6 ECTS
Semester	: 2
Course Status	: Mandatory
Prerequisite/s	: -
Faculty	: Engineering & Information Technology
Study Program	: Information Technology
Concentration	: -
Degree	: Bachelor
Instructor(s)	: Joseph Andreas
Delivery Mode	: On-Site
Learning Method	: Synchronous
Total Activity Time¹	: 180 hours
Scheduled Session²	: 58.4 hours
Independent Study	: 121.6 hours

¹ 1 credit equals to 45 hours of learning activities (including guided, structured assignment, & independent study) per semester

² total sessions x total SKS x 50 minutes



<p>Course Description</p>	<p>This course introduces monolithic web development based on server-side programming using HTML, CSS, PHP, JQuery AJAX, and JSON. Students will also be able to create dynamic web pages, which will allow user input data to be stored in a database server and the developed application will be able to retrieve data stored on the database server. By the end of the semester, students should have a working final project.</p>
<p>Learning Outcomes</p>	<p>*Graduate Learning Outcomes (CPL) assigned to this course</p> <p>CPL1: Apply knowledge of computing fundamentals, mathematics, and programming to solve computing problems CPL2: Design, implement, and evaluate computer-based systems or processes that meet desired needs CPL3: Demonstrate proficiency in web and software development tools and techniques CPL4: Work effectively as an individual or as a member of a team in developing software applications CPL5: Communicate effectively in written and oral forms in both technical and non-technical contexts CPL6: Recognize the need for and engage in continuous learning to adapt to emerging technologies</p> <p>*Course Learning Outcomes (CPMK)</p> <p>CPMK1: Explain the structure and workflow of monolithic web applications CPMK2: Develop static web pages using HTML and CSS following modern web standards CPMK3: Implement dynamic client-side functionality using JavaScript and jQuery CPMK4: Build server-side logic using PHP to handle user interaction and data processing CPMK5: Integrate front-end and back-end using AJAX for asynchronous data exchange CPMK6: Design and manipulate databases using MySQL to support dynamic web content CPMK7: Demonstrate the ability to work in a team to develop a simple monolithic web application CPMK8: Communicate the design and functionality of a developed web system clearly in documentation and presentations CPMK9: Reflect on technological advancements in web development and apply continuous improvement principles</p> <p>*Planned Final Ability (Sub-CPMK, if any)</p> <p>Sub-CPMK X: Sub-CPMK X:</p>

	Sub-CPMK X:
References	Primary Reference: PHP & MySQL Server-side Web Development ISBN: 9781119149217 Wiley
	Additional Reference: L. Welling and L. Thomson, PHP and MySQL Web Development, 5th ed. Boston, MA, USA: Addison-Wesley, 2016

Session	Topic/ Subtopics	Learning Outcomes	Assessment Criteria		Student Learning Experiences/Activities	Learning Methods and Modalities	Allocation Time		Learning Materials
			Indicator	Weight (%)			Scheduled (@50 Min)	Independent Study (@Hours)	
1	- Basic PHP syntax - Data types and its lifetime	Students are able to understand the basic PHP syntax, Data types and its lifetime	- Clarity in explaining the basic PHP syntax - Successfully in writing error-free code	5%	- Introduction to PHP syntax (e.g., variables, operators) and data types - Hands-on coding exercises: Write basic PHP programs - Peer discussion on data types and variable scope	- Interactive lecture on basic PHP syntax - Guided coding exercises - Peer discussion for better understanding of data types	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	PHP manual, coding exercises, IDE setup (VS Code, XAMPP)
2	- Form, GET AND POST - Input and form data validation	- Students are able to identify the type of form for data submission - Students are able to validate the content of form	- Clarity in explaining differentiating the method used in Form Writing code that successfully validate the user data sent from a form or input	5%	- Analyze form submission methods (GET vs POST) - Validate user input in forms using PHP - Code validation scripts for forms, work on error handling - Group activity to identify security risks in form handling	- Hands-on form creation with GET and POST methods - Group discussion on data validation principles - Pair programming for form validation	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	PHP and HTML form examples, validation cheat sheet, W3Schools documentation

3	MySQL Input & Select MySQL data validation	- Students are able to identify the SQL data manipulation commands Students are able to verify a registered user login	Clarity in explaining the MySQL DML Successfully validate data on the database server	5%	- Lecture on MySQL basic operations (INSERT, SELECT) - Hands-on practice: Create and query a MySQL database - Perform validation for user login data - Collaborative debugging of SQL queries	- Lecture on MySQL DML (Data Manipulation Language) - Hands-on practice with SQL queries and database connections - Group activity on database validation	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	MySQL documentation, sample database schema, SQL queries reference
4	MySQL delete, Edit CRUD operations	- Students are able to identify the SQL data manipulation commands - Students are able to perform CRUD on database server	Clarity in writing the CRUD operation in performing the desired result	10%	- Code exercises on DELETE, UPDATE, and INSERT SQL commands - Hands-on CRUD operation on a sample database - Group work on building a small CRUD application	- Interactive SQL practice on CRUD operations - Pair work on implementing database edits - Teacher-led demonstration on using PHP with MySQL for CRUD	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	CRUD operation examples, MySQL query debugging tools
5	Cookie, session	Students are able to understand cookie and session and able to write code to create session	- Clarity in explaining cookie and its relation with session - Successfully create session and cookie and in client and server communication	5%	- Explanation of session management and cookies - Lab activity: Create a login system using sessions and cookies - Group work: Discuss session management best practices	- Demonstration of cookies and sessions in PHP - Hands-on lab work on creating sessions and cookies - Group discussion on session security best practices	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	PHP session and cookie management tutorial, session best practices

6	Final project materials	Students are able to understand what is expected from them for the final project	Clarity in defining the tasks of the final project	5%	<ul style="list-style-type: none"> - Introduction to final project requirements - Brainstorming session: Students define project scope - Independent research on project ideas and components 	<ul style="list-style-type: none"> - Individual brainstorming session on final project scope - Instructor-guided discussion of project requirements 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	Project guidelines, project planning template, sample project ideas
7	File handling and Object in PHP	Students are able to understand about file handling and Object-oriented programming in PHP	Clarity in identifying object and successfully creating profile picture	5%	<ul style="list-style-type: none"> - Explanation and demonstration of file handling in PHP - Code exercises on uploading, reading, and writing files - Introduction to Object-Oriented Programming in PHP (OOP) - Group discussions on object creation and methods 	<ul style="list-style-type: none"> - Demonstration of file handling in PHP - Hands-on exercises to work with file uploads and file systems - Introduction to OOP concepts through examples 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	PHP file handling documentation, OOP PHP examples, online file storage APIs
8	JQuery AJAX	Students are able to understand about JQuery AJAX. Students are able to work with JQuery AJAX	Clarity in programming with JQuery AJAX and produce a functional code	5%	<ul style="list-style-type: none"> - Introduction to AJAX and its use in PHP applications - Practice using jQuery to make asynchronous requests - Lab work: Build an AJAX-driven form submission system 	<ul style="list-style-type: none"> - Explanation of AJAX and jQuery basics - Practice coding AJAX requests in PHP - Problem-solving session to build an interactive web form 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	jQuery and AJAX tutorial, AJAX request examples, browser developer tools
9	Rest API	<ul style="list-style-type: none"> - Students are able to understand about Rest API - Students are able to work with Rest API 	Clarity in programming with Rest API and produce product catalog	5%	<ul style="list-style-type: none"> - Lecture on REST API principles and architecture - Hands-on work: Connect to a public API and retrieve data - Group activity: Discuss how APIs can be used in web applications 	<ul style="list-style-type: none"> - Lecture and demonstration on integrating with REST APIs - Hands-on activities using external APIs (e.g., JSONPlaceholder) - Group discussion on REST architecture 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	REST API documentation (Postman, Swagger), sample public APIs



10	Working Catalog	Students are able to create Catalog	Clarity in producing product catalog	10%	<ul style="list-style-type: none"> - Build a product catalog with PHP, MySQL, and basic front-end design - Code exercises: CRUD operations for catalog items - Group project to design and implement a product catalog 	<ul style="list-style-type: none"> - Build product catalog in teams, focusing on CRUD functionality - Peer reviews and improvement suggestions - Hands-on work with MySQL and PHP 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	Product catalog examples, CRUD implementation guides, design templates
11	JSON Dynamic	Students are able to combine JQuery and JSON Dynamic search	Clarity in producing JSON object from the JQuery	5%	<ul style="list-style-type: none"> - Introduction to JSON and its role in dynamic web applications - Hands-on activity: Create a dynamic search interface using JSON and JQuery - Debugging and testing JSON functionality 	<ul style="list-style-type: none"> - Hands-on exercise: Integrating JQuery with JSON for dynamic searches - Debugging and testing code for dynamic interactions 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	jQuery JSON tutorial, dynamic search example code, JSON formatting tools
12	Project progress	Students are able to show their project progress	Clarity in explaining the progress	5%	<ul style="list-style-type: none"> - Progress presentations from students on final project - Peer feedback and self-assessment - Discussion on challenges faced and solutions implemented 	<ul style="list-style-type: none"> - Presentation of project progress to peers and instructors - Self-assessment and peer feedback session 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	Project progress update template, peer feedback forms, self-assessment sheets
13	Lumens Micro framework	Students are able to work with Lumens Micro framework	Clarity in defining Lumens Micro framework and successfully allow lightweight installation	5%	<ul style="list-style-type: none"> - Introduction to the Lumen micro-framework - Hands-on coding: Setting up a basic Lumen project - Discuss Lumen's role in lightweight APIs and web apps 	<ul style="list-style-type: none"> - Demonstration and hands-on practice with the Lumen framework - Discussion of framework features and advantages - Pair programming for API integration with Lumen 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1 SKS * 2 hours)	8.7 hours	Lumen documentation (Official Lumen Docs), Lumen setup guides, video tutorials on micro-frameworks
14	Rest API Lumens Micro framework	Students are able to combine Rest API with	Successfully combining Rest API with Lumens Micro framework	5%	<ul style="list-style-type: none"> - Learn to integrate a REST API with Lumen - Build a small application using Lumen and a REST API - Debug and test the API 	<ul style="list-style-type: none"> - Guided activity combining REST API and Lumen framework - Collaborative work on small projects 	150 mins theory (3 SKS * 1 hour) + 100 mins lab (1	8.7 hours	Lumen and REST API integration examples, Postman API testing tools, Lumen deployment guides



		Lumens Micro framework			connections	integrating both technologies	SKS * 2 hours)		
Final Exam	Students present the project that they create	Final Project presentation	Clarity in explaining the final result of the project and how to defend it	30%	- Final presentations of projects - Peer reviews and critiques - Self-reflection on learning outcomes	- Student-led project presentations - Q&A and defense of project choices and challenges faced - Peer evaluation of projects	120 minutes		Final project rubric, project presentation guidelines, peer evaluation sheets

Notes:

Prepared by:	Approved by	Acknowledged by:
Kho I Eng	James Purnama M.Sc	Dr. Maulahikmah Galinium S.Kom, M.Sc
Head of Editorial Team	Head of BIT Department	Dean of FEIT