SRM VALLIAMMAI ENGINEERING COLLEGE

(An Autonomous Institution)

SRM Nagar, Kattankulathur – 603 203

DEPARTMENT OF INFORMATION TECHNOLOGY

QUESTION BANK

Academic Year 2025 – 2026(EVEN SEMESTER)

VI SEMESTER

IT3661-FULL STACK DEVELOPMENT

Regulation – 2023

Prepared by

Mrs.S. Shenbagavadivu, AP (Sel.G)/IT

SRM VALLIAMMAI ENGINEERING COLLEGE

(An Autonomous Institution)

SRM Nagar, Kattankulathur – 603 203.

DEPARTMENT OF INFORMATION TECHNOLOGY

QUESTION BANK

SUBJECT: IT3661-FULL STACK DEVELOPMENT

SEM / YEAR: VI SEMESTER/ THIRD YEAR

UNIT-I: WEB DEVELOPMENT BASICS			
Web development Basics - HTML & Web servers, Shell - UNIX CLI Version control, Git &Github, HTML, CSS.			
PART-A			
Q.no	**Question**	**BTL**	**Competence**
1	Define web development.	BTL1	Remembering
2	Explain the role of web servers in web development.	BTL2	Understanding
3	What is HTML?	BTL1	Remembering
4	Describe the basic structure of an HTML document.	BTL2	Understanding
5	List any two HTML tags.	BTL1	Remembering
6	Explain the importance of HTML in web page creation.	BTL2	Understanding
7	Define a web server.	BTL1	Remembering
8	Explain how a web server processes a client request.	BTL2	Understanding
9	What is UNIX shell?	BTL1	Remembering
10	Describe the role of the UNIX command-line interface.	BTL2	Understanding
11	List any two UNIX commands.	BTL1	Remembering
12	Explain the use of basic UNIX commands in file handling.	BTL2	Understanding
13	Define version control.	BTL1	Remembering
14	Explain why version control systems are needed.	BTL2	Understanding
15	Name any two version control systems.	BTL1	Remembering
16	Compare centralized and distributed version control systems.	BTL2	Understanding
17	What is Git?	BTL1	Remembering
18	Explain how Git supports collaborative development.	BTL2	Understanding
19	Define GitHub.	BTL1	Remembering
20	Explain the purpose of repositories in GitHub.	BTL2	Understanding

21	List any two features of CSS.	BTL1	Remembering
22	Explain how CSS improves the presentation of web pages.	BTL2	Understanding
23	What is a webpage?	BTL1	Remembering
24	Describe the relationship between HTML and CSS.	BTL2	Understanding

PART-B				
Q.no	**Question**	**Mark**	**BTL**	**Competence**
1	Illustrate the complete process of web page delivery from client to server using a suitable diagram.	16	BTL3	Applying
2	Demonstrate the creation of a simple web page using HTML and explain the role of each tag used.	16	BTL3	Applying
3	Apply basic UNIX CLI commands to create, modify, and manage files and directories with examples.	16	BTL3	Applying
4	Examine the working of a web server in handling HTTP requests and responses.	16	BTL3	Applying
5	Analyze the architecture of a web application and classify its major components.	16	BTL4	Analyzing
6	Analyze the differences between static and dynamic web pages with suitable examples.	16	BTL4	Analyzing
7	Analyze the features of Git and classify them based on their role in version control.	16	BTL4	Analyzing
8	Analyze common challenges faced in web development and propose suitable solutions.	16	BTL4	Analyzing
9	Justify the need for version control systems in modern web development environments.	16	BTL5	Evaluating
10	Conclude the advantages of using GitHub over traditional file-sharing methods for project management.	16	BTL5	Evaluating
11	Justify the importance of CSS in improving the usability and maintainability of web pages.	16	BTL5	Evaluating
12	Conclude how UNIX CLI enhances developer productivity in web development.	16	BTL5	Evaluating
13	Design a simple website layout using HTML and CSS and explain the styling techniques used.	16	BTL6	Creating
14	Develop a web page using HTML and CSS that demonstrates responsive design principles.	16	BTL6	Creating
15	Create a Git repository and demonstrate the steps involved in committing and pushing changes to GitHub.	16	BTL6	Creating
16	Design and develop a basic website using HTML, CSS, and version control using Git.	16	BTL6	Creating
17	Create a complete web development workflow integrating HTML, CSS, UNIX CLI, Git, and GitHub.	16	BTL6	Creating

UNIT II - FRONTEND DEVELOPMENT

JavaScript Basics, Object-Oriented Programming (OOP) Aspects of JavaScript, Memory Usage and Functions in JavaScript, AJAX for Data Exchange with the Server, jQuery Framework, jQuery Events and UI Components, and JSON Data Format.

PART-A

Q.no	Question	BTL	Competence
1	Define JavaScript.	BTL1	Remembering
2	Explain the purpose of JavaScript in web development.	BTL2	Understanding
3	What is meant by Object-Oriented Programming in JavaScript?	BTL1	Remembering
4	Explain how objects are created in JavaScript.	BTL2	Understanding
5	Define a JavaScript function.	BTL1	Remembering
6	Describe the role of functions in JavaScript programs.	BTL2	Understanding
7	What is memory allocation in JavaScript?	BTL1	Remembering
8	Explain how JavaScript manages memory during program execution.	BTL2	Understanding
9	Define AJAX.	BTL1	Remembering
10	Explain how AJAX enables asynchronous communication with the server.	BTL2	Understanding
11	What is jQuery?	BTL1	Remembering
12	Describe the advantages of using jQuery in web applications.	BTL2	Understanding
13	Define jQuery events.	BTL1	Remembering
14	Explain how event handling is performed using jQuery.	BTL2	Understanding
15	What are jQuery UI components?	BTL1	Remembering
16	Describe the role of jQuery UI in improving user interfaces.	BTL2	Understanding
17	Define JSON.	BTL1	Remembering
18	Explain the structure of JSON data format with an example.	BTL2	Understanding
19	What is meant by data exchange in web applications?	BTL1	Remembering
20	Explain the role of JSON in client–server communication.	BTL2	Understanding
21	List any two features of JavaScript.	BTL1	Remembering
22	Explain how JavaScript enhances interactivity in web pages.	BTL2	Understanding
23	What is an AJAX request?	BTL1	Remembering
24	Explain the difference between synchronous and asynchronous communication.	BTL2	Understanding

PART-B

Q.no	Question	Mark	BTL	Competence
1	Illustrate the basic structure of a JavaScript program with suitable examples.	16	BTL3	Applying
2	Demonstrate the use of JavaScript functions to perform form validation.	16	BTL3	Applying

3	Apply object-oriented programming concepts in JavaScript with a suitable example.	16	BTL3	Applying
4	Examine how JavaScript handles memory allocation and garbage collection.	16	BTL3	Applying
5	Analyze the role of functions and scope in JavaScript execution.	16	BTL4	Analyzing
6	Analyze the differences between procedural and object-oriented approaches in JavaScript.	16	BTL4	Analyzing
7	Analyze the working of AJAX for asynchronous data exchange between client and server.	16	BTL4	Analyzing
8	Classify jQuery features based on DOM manipulation, events, and effects.	16	BTL4	Analyzing
9	Justify the need for AJAX in modern web applications.	16	BTL5	Evaluating
10	Conclude the advantages of using jQuery over plain JavaScript.	16	BTL5	Evaluating
11(a)	Justify the importance of JSON as a data interchange format in web applications.	16	BTL5	Evaluating
11(b)	Conclude how JSON improves data exchange efficiency between client and server.	16	BTL5	Evaluating
12(a)	Explain how JavaScript enhances interactivity in web pages.	8	BTL5	Evaluating
12(b)	Justify the role of event handling in improving user experience.	8	BTL5	Evaluating
13	Design a dynamic web page using JavaScript and jQuery to handle user events.	16	BTL6	Creating
14	Develop an AJAX-based application to fetch and display data from a server.	8	BTL6	Creating
15(a)	Design a JavaScript program using OOP principles to manage user data.	8	BTL6	Creating
15(b)	Develop suitable methods to perform CRUD operations.	16	BTL6	Creating
16	Design and develop a web interface using jQuery UI components.	16	BTL6	Creating
17(a)	Create a frontend module integrating JavaScript and AJAX	8	BTL6	Creating
17(b)	Design JSON data formats for client–server communication.	8	BTL6	Creating

UNIT - III - REACT JS			
Introduction to React, React Router and Single Page Applications, React Forms, Flow Architecture, Introduction to Redux, More Redux, and Client–Server Communication.			
Q. No.	**Questions**	**BT Level**	**Competence**
1	Define React.	BTL1	Remembering
2	Explain the purpose of React in web application development.	BTL2	Understanding
3	What is a component in React?	BTL1	Remembering
4	Explain the role of components in React applications.	BTL2	Understanding
5	Define JSX.	BTL1	Remembering
6	Explain how JSX improves UI development in React.	BTL2	Understanding

7	What is meant by Single Page Application (SPA)?		BTL1	Remembering
8	Explain how React supports Single Page Applications.		BTL2	Understanding
9	Define React Router.		BTL1	Remembering
10	Explain the need for React Router in React applications.		BTL2	Understanding
11	What are React forms?		BTL1	Remembering
12	Explain controlled components in React forms.		BTL2	Understanding
13	Define unidirectional data flow in React.		BTL1	Remembering
14	Explain the concept of flow architecture in React.		BTL2	Understanding
15	Define Redux.		BTL1	Remembering
16	Explain the role of Redux in state management.		BTL2	Understanding
17	What is a Redux store?		BTL1	Remembering
18	Explain the interaction between actions and reducers in Redux.		BTL2	Understanding
19	Define client–server communication.		BTL1	Remembering
20	Explain how React applications communicate with servers using REST APIs.		BTL2	Understanding
21	What is meant by middleware in Redux?		BTL1	Remembering
22	Explain the purpose of middleware in Redux architecture.		BTL2	Understanding
23	Define REST API.		BTL1	Remembering
24	Explain the role of REST APIs in React-based applications.		BTL2	Understanding
PART – B				
1	Illustrate the basic features of React and explain its advantages over traditional JavaScript frameworks.	16	BTL3	Applying
2	Demonstrate the creation of a simple React component and explain its lifecycle.	16	BTL3	Applying
3	Apply the concept of JSX to build reusable UI components in React with examples.	16	BTL3	Applying
4	Examine how React handles state and props in component-based architecture.	16	BTL3	Applying
5	Illustrate the working of React Router in building Single Page Applications.	16	BTL3	Applying
6.	Analyze the architecture of a Single Page Application and classify its major components.	16	BTL4	Analyzing
7	Analyze the differences between traditional multi-page applications and React-based SPAs.	16	BTL4	Analyzing
8	Analyze the flow of data in React applications using unidirectional data flow.	16	BTL4	Analyzing
9	Analyze the working of React forms and controlled components with suitable examples.	16	BTL4	Analyzing
10	Analyze the Redux architecture and explain the interaction between store, actions, and reducers.	16	BTL4	Analyzing
11	Examine how Redux improves state management in large React applications.	16	BTL3	Applying
12	Analyze client–server communication in React applications using REST APIs.	8	BTL4	Analyzing
	Examine the role of HTTP methods in React-based client–server interaction.	8	BTL4	Analyzing
13	Justify the need for Redux in complex React applications.	16	BTL5	Evaluating
14	Conclude the advantages of using React Router for navigation in	16	BTL5	Evaluating

	Single Page Applications.			
15	Design a React form to handle user input and validation.	8	BTL6	Creating
	Develop suitable techniques to manage form state effectively in React.	8	BTL6	Creating
16	Develop a simple React application integrated with Redux for state management.	16	BTL6	Creating
17	Design and develop a React-based client–server application using RESTful services.	16	BTL6	Creating

UNIT - IV : JAVA WEB DEVELOPMENT			
Java Programming Basics, Model–View–Controller (MVC) Pattern, MVC Architecture Using Spring, RESTful API Using Spring Framework, and Building an Application Using Maven.			
Q. No.	**Questions**	**BT Level**	**Competence**
1	Define Java programming.	BTL1	Remembering
2	Explain the features of Java programming language.	BTL2	Understanding
3	What is meant by Object-Oriented Programming in Java?	BTL1	Remembering
4	Explain the importance of OOP concepts in Java.	BTL2	Understanding
5	Define Model–View–Controller (MVC) pattern.	BTL1	Remembering
6	Explain the role of MVC pattern in web applications.	BTL2	Understanding
7	What is a Model in MVC architecture?	BTL1	Remembering
8	Explain the responsibilities of View and Controller in MVC.	BTL2	Understanding
9	Define Spring Framework.	BTL1	Remembering
10	Explain the need for Spring Framework in Java web development.	BTL2	Understanding
11	What is Spring MVC?	BTL1	Remembering
12	Explain how Spring MVC implements MVC architecture.	BTL2	Understanding
13	Define RESTful API.	BTL1	Remembering
14	Explain the principles of REST architecture.	BTL2	Understanding
15	What are HTTP methods used in RESTful services?	BTL1	Remembering
16	Explain the role of HTTP methods in REST API design.	BTL2	Understanding
17	Define Maven.	BTL1	Remembering
18	Explain the purpose of Maven in Java projects.	BTL2	Understanding
19	What is a Maven repository?	BTL1	Remembering
20	Explain the role of dependencies in Maven.	BTL2	Understanding
21	Define REST controller in Spring.	BTL1	Remembering
22	Explain how Spring handles REST requests and responses.	BTL2	Understanding
23	What is meant by build automation?	BTL1	Remembering
24	Explain how Maven supports build automation in Java applications.	BTL2	Understanding
PART – B			
1	Illustrate the basic features of Java programming with suitable examples.	16	BTL3	Applying
2	Demonstrate the use of packages and access modifiers in Java programs.	16	BTL3	Applying
3	Apply object-oriented concepts such as inheritance and polymorphism in Java with examples.	16	BTL3	Applying

4	Examine the working of the Model–View–Controller (MVC) design pattern in Java applications.	16	BTL3	Applying
5	Analyze the architecture of an MVC-based web application and classify its components.	16	BTL4	Analyzing
6.	Analyze the role of Spring Framework in implementing MVC architecture.	16	BTL4	Analyzing
7	Analyze the flow of control between Model, View, and Controller in Spring MVC.	16	BTL4	Analyzing
8	Analyze the working of RESTful web services in Java-based applications.	16	BTL4	Analyzing
9	Examine how HTTP methods are mapped to RESTful APIs using Spring Framework.	16	BTL3	Applying
10	Analyze the advantages of using Spring MVC over traditional Java web technologies.	16	BTL4	Analyzing
11	Apply Java annotations in Spring MVC controllers.	8	BTL3	Applying
12	Examine their role in request handling.	8	BTL3	Applying
	Analyze the design principles of RESTful APIs.	8	BTL4	Analyzing
13	Examine how Spring supports REST API development.	8	BTL4	Analyzing
14	Justify the need for using MVC architecture in large-scale Java web applications.	16	BTL5	Evaluating
15	Conclude the benefits of using Maven for project management and build automation.	16	BTL5	Evaluating
	Design a simple Spring MVC application structure for a web-based system.	16	BTL6	Creating
16	Develop a RESTful API using Spring Framework for a sample application.	16	BTL6	Creating
17	Design and develop a Java web application using MVC architecture and Maven build tool.	16	BTL6	Creating

UNIT - V : DATABASES & DEPLOYMENT			
Relational Schemas and Normalization, Structured Query Language (SQL), Data Persistence Using Spring JDBC, Agile Development Principles, and Deploying Applications in the Cloud.			
Q. No.	**Questions**	**BT Level**	**Competence**
1	Define a relational database.	BTL1	Remembering
2	Explain the characteristics of relational databases.	BTL2	Understanding
3	What is a relational schema?	BTL1	Remembering
4	Explain the importance of relational schemas in database design.	BTL2	Understanding
5	Define normalization.	BTL1	Remembering
6	Explain the need for normalization in databases.	BTL2	Understanding
7	What is First Normal Form (1NF)?	BTL1	Remembering
8	Explain the purpose of Second Normal Form (2NF).	BTL2	Understanding
9	Define Structured Query Language (SQL).	BTL1	Remembering
10	Explain the role of SQL in database management systems.	BTL2	Understanding
11	What is a primary key?	BTL1	Remembering
12	Explain the importance of primary and foreign keys.	BTL2	Understanding

13	Define Spring JDBC.		BTL1	Remembering
14	Explain how Spring JDBC supports data persistence.		BTL2	Understanding
15	What is meant by data persistence?		BTL1	Remembering
16	Explain the role of DAO in Spring JDBC.		BTL2	Understanding
17	Define Agile development.		BTL1	Remembering
18	Explain the core principles of Agile development.		BTL2	Understanding
19	What is continuous integration?		BTL1	Remembering
20	Explain the importance of continuous integration in Agile projects.		BTL2	Understanding
21	Define cloud computing.		BTL1	Remembering
22	Explain the benefits of deploying applications in the cloud.		BTL2	Understanding
23	What is meant by cloud deployment?		BTL1	Remembering
24	Explain the role of cloud platforms in modern application deployment.		BTL2	Understanding
PART – B				
1	Illustrate the concept of relational schemas with suitable examples.	16	BTL3	Applying
2	Apply normalization techniques to convert a given table into Third Normal Form (3NF).	16	BTL3	Applying
3	Demonstrate the use of SQL commands for data definition and data manipulation with examples.	16	BTL3	Applying
4	Apply SQL queries to retrieve and manipulate data from multiple tables using joins.	16	BTL3	Applying
5	Analyze the need for normalization and explain its impact on database performance.	16	BTL4	Analyzing
6.	Analyze the differences between various normal forms with suitable examples.	16	BTL4	Analyzing
7	Analyze the architecture of a database-driven application using Spring JDBC.	16	BTL4	Analyzing
8	Analyze the working of Spring JDBC for data persistence in enterprise applications.	16	BTL4	Analyzing
9	Examine how SQL constraints help in maintaining data integrity.	16	BTL3	Applying
10	Analyze the role of DAO pattern in Spring JDBC–based applications.	16	BTL4	Analyzing
11	Justify the use of Agile development principles in modern software projects.	8	BTL5	Evaluating
12	Conclude the benefits of iterative and incremental development in Agile methodology.	8	BTL5	Evaluating
	Examine the role of cloud platforms in scalable application deployment.	16	BTL3	Applying
13	Analyze the challenges involved in deploying applications to the cloud and suggest suitable solutions.	16	BTL4	Analyzing
14	Design a relational database schema for a real-world application.	8	BTL6	Creating
15	Develop suitable SQL queries to support CRUD operations.	8	BTL6	Creating
	Develop SQL queries to support CRUD operations for a given database schema.	16	BTL6	Creating
16	Analyze the impact of Agile practices on database design and deployment.	16	BTL4	Analyzing
17	Design and deploy a simple database-driven application using Spring JDBC in a cloud environment.	16	BTL6	Creating