SRM VALLIAMMAI ENGINEERING COLLEGE

(An Autonomous Institution)

SRM Nagar, Kattankulathur - 603 203

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

QUESTION BANK

VII SEMESTER

1922704 - WEB DEVELOPMENT FRAMEWORKS

Regulation – 2019

Academic Year 2025 – 2026 Odd Semester

Prepared by

R. Vaishnavi, Assistant Professor (O.G)

UNIT 1 FUNDAMENTALS OF WEB FRAMEWORK

Web framework-History-Types of framework architectures-Model–view–controller (MVC)- Three-tier organization-Introduction to frameworks-Framework applications -General-purpose website frameworks-Server-side-Client-side-Features

PART – A

Q.No	Question	Level	Competence
1.	What is a web framework?	BTL 1	Remember
2.	Name any two early web frameworks and their year of introduction.	BTL 1	Remember
3.	Mention one significant benefit web frameworks brought to web development.	BTL 1	Remember
4.	How did web frameworks evolve to support dynamic websites?	BTL 1	Remember
5.	What is meant by framework architecture?	BTL 2	Understand
6.	Name two common types of framework architectures.	BTL 1	Remember
7.	Distinguish between monolithic and modular framework architectures.	BTL 1	Remember
8.	What is the primary advantage of using a modular framework?	BTL 1	Remember
9.	Define the MVC architecture.	BTL 1	Remember
10.	What is the role of the 'Model' component in MVC?	BTL 2	Understand
11.	How does the 'Controller' help in MVC?	BTL 2	Understand
12.	Give an example of a framework that uses the MVC pattern.	BTL 1	Remember
13.	What are the three layers in a three-tier architecture?	BTL 1	Remember
14.	How is the business logic layer different from the data layer?	BTL 1	Remember
15.	Mention one advantage of using a three-tier architecture.	BTL 2	Understand
16.	How does the three-tier model improve application scalability?	BTL 2	Understand
17.	What is a software framework?	BTL 2	Understand
18.	List two characteristics of a typical software framework.	BTL 2	Understand
19.	How do frameworks differ from libraries?	BTL 1	Remember
20.	Give one reason why developers use frameworks in web development.	BTL 1	Remember
21.	Name two popular web applications built using frameworks.	BTL 1	Remember
22.	Mention one area where a framework can reduce development time.	BTL 1	Remember
23.	How do frameworks assist in enforcing coding standards?	BTL 2	Understand
24.	What is one security benefit offered by using web frameworks?	BTL 1	Remember

PART-B			
Q.No.	**Question**	**Level**	**Competence**
1.	Explain the evolution and history of web frameworks with examples.	BTL 4	Analyze
2.	Discuss the different types of web framework architectures with suitable diagrams.	BTL 4	Analyze
3.	Describe the Model–View–Controller (MVC) architecture with a neat labeled diagram and example.	BTL 3	Apply
4.	Compare and contrast MVC architecture with Three-tier architecture.	BTL 4	Analyze
5.	Explain in detail the three-tier architecture used in web applications.	BTL 4	Analyze
6.	Write a comprehensive note on the introduction and classification of frameworks.	BTL 3	Apply
7.	Describe the applications of web frameworks in modern software development.	BTL 3	Apply
8.	Explain how general-purpose website frameworks differ from domain-specific ones, with examples.	BTL 4	Analyze
9.	Compare server-side and client-side web frameworks with at least two examples of each.	BTL 4	Analyze
10.	Discuss the role and features of server-side frameworks in dynamic website development.	BTL 4	Analyze
11.	Discuss in detail the components and workflow of a client-side framework.	BTL 4	Analyze
12.	Write a detailed account of features common to modern web frameworks.	BTL 3	Apply
13.	Explain how MVC architecture is implemented in a real-world web framework like Django or Angular.	BTL 4	Analyze
14.	Describe the advantages and disadvantages of using web frameworks.	BTL 3	Apply
15.	How do web frameworks improve productivity and maintainability in software projects?	BTL 3	Apply
16.	Illustrate how a general-purpose framework like Laravel or Flask can be used to build a full-stack application.	BTL 4	Analyze
17.	Describe the layered architecture of a web application using both MVC and Three-tier architecture.	BTL 3	Apply
PART - C			
1.	Analyze and evaluate the impact of web frameworks on full-stack development using case studies or real-life examples.	BTL 5	Evaluate
2.	Discuss the evolution of web development from static websites to Web 2.0 and how frameworks facilitated this shift.	BTL 5	Evaluate
3.	Compare and contrast the roles of MVC and Three-tier architecture in enterprise web application development.	BTL 5	Evaluate

4.	Design a sample web application and explain how both client-side and server-side frameworks would be applied in its development.	BTL 6	Create
5.	Critically evaluate the features of at least three modern web frameworks and justify which is most suitable for a startup e-commerce application.	BTL 5	Evaluate

UNIT II JAVA WEB FRAMEWORK

Java Web Frameworks-Struts-The Struts Framework- The Struts Tag Libraries- - Struts Configuration Files- Applying Struts

PART – A

Q.No	Question	Level	Competence
1.	What is the Struts framework?	BTL 1	Remember
2.	Define Java web framework.	BTL 1	Remember
3.	What is the role of `struts-config.xml`?	BTL 1	Remember
4.	Name any two tag libraries provided by Struts.	BTL 2	Understand
5.	What is the purpose of the `ActionForm` class in Struts?	BTL 1	Remember
6.	Mention any two configuration files used in Struts.	BTL 1	Remember
7.	List the core components of the Struts framework.	BTL 2	Understand
8.	Name any two features of the Struts framework.	BTL 2	Understand
9.	What is the full form of MVC in Struts?	BTL 1	Remember
10.	Identify two built-in tags in Struts HTML Tag Library.	BTL 1	Remember
11.	What is web.xml used for in a Struts application?	BTL 2	Understand
12.	Name the main package where Struts classes are stored.	BTL 2	Understand
13.	Define the purpose of Struts tag libraries.	BTL 2	Understand
14.	How do Struts framework supports the MVC architecture.	BTL 1	Remember
15.	Define the function of the ActionServlet in Struts.	BTL 2	Understand
16.	Compare struts-config.xml and web.xml.	BTL 1	Remember
17.	How does the ActionMapping component work in Struts?	BTL 1	Remember
18.	What is the use of validate() method in ActionForm.	BTL 2	Understand
19.	Differentiate between HTML and Bean tag libraries in Struts.	BTL 1	Remember
20.	List the flow of control in a Struts-based web application.	BTL 1	Remember
21.	How do Struts handles form data submission.	BTL 2	Understand
22.	How error messages are handled in Struts using properties files.	BTL 2	Understand

23.	How are business logics separated from the presentation layer in Struts?	BTL 1	Remember
24.	Mention the process of deploying a Struts application on a web server.	BTL 1	Remember
PART-B			
Q.No.	**Question**	**Level**	**Competence**
1.	Explain the architecture of the Struts framework with a neat diagram.	BTL 4	Analyze
2.	Describe the key features of the Struts framework that make it suitable for enterprise-level web development.	BTL 3	Apply
3.	Discuss the MVC pattern implementation in the Struts framework with an example.	BTL 3	Apply
4.	Explain the life cycle of a request in the Struts framework.	BTL 4	Analyze
5.	Describe the role of Action, ActionForm, and ActionMapping in Struts.	BTL 3	Apply
6.	Explain the configuration process of a Struts application using `struts-config.xml` and `web.xml`.	BTL 4	Analyze
7.	Discuss in detail the Struts tag libraries and their role in simplifying view development.	BTL 4	Analyze
8.	Write notes on the HTML and Bean tag libraries in Struts with usage examples.	BTL 3	Apply
9.	Explain the validation mechanism in Struts using validation.xml and ValidatorForm.	BTL 4	Analyze
10.	Describe the use and purpose of the Struts Tiles framework.	BTL 3	Apply
11.	Compare the functionality of custom tags and standard tags in Struts.	BTL 3	Apply
12.	Describe how to handle form input and data binding in Struts with relevant code examples.	BTL 3	Apply
13.	Explain the directory structure and deployment process of a Struts-based web application.	BTL 4	Analyze
14.	Illustrate the use of exception handling and error messages in Struts applications.	BTL 5	Evaluate
15.	How does ActionServlet function in the Struts framework? Explain with an example.	BTL 3	Apply
16.	Discuss the advantages and limitations of the Struts framework in modern Java web development.	BTL 4	Analyze
17.	Write a detailed note on the integration of Struts with other technologies like Hibernate or Spring.	BTL 3	Apply
PART-C			
1.	Develop a complete Struts application to perform user login and explain each component used.	BTL 6	Create
2.	Compare Struts 1.x and Struts 2.x in terms of architecture, performance, and ease of use.	BTL 5	Evaluate

3.	Analyze the role of tag libraries in Struts and demonstrate their implementation in form creation and data display.	BTL 5	Evaluate
4.	Explain how a typical Struts application is initialized and processed, covering configuration, actions, results, and view.	BTL 5	Evaluate
5.	Design and implement a form-based registration module using Struts and explain validation, input processing, and result display.	BTL 6	Create

UNIT III STRUTS2

Struts and Agile Development -Basic Configuration.-Actions and Action Support.-Results and Result Types.-OGNL, the Value Stack, and Custom Tags-Form Tags-Form Validation and Type Conversion-Exceptions and Logging-Getting Started with JavaScript-Advanced JavaScript, the DOM, and CSS-Themes and Templates-Rich Internet Applications

PART – A

Q.No.	Question	Level	Competence
1.	What is Agile development in the context of web applications?	BTL 1	Remember
2.	How does Struts support Agile development practices?	BTL 1	Remember
3.	Name two benefits of integrating Struts with Agile methodologies.	BTL 2	Understand
4.	How can Struts help in iterative and incremental development?	BTL 1	Remember
5.	What is the purpose of the `struts.xml` configuration file?	BTL 1	Remember
6.	Define the role of the `web.xml` file in a Struts application.	BTL 2	Understand
7.	What is an interceptor in Struts configuration?	BTL 2	Understand
8.	Mention two configuration properties used in `struts.properties`.	BTL 2	Understand
9.	What is an `Action` class in Struts?	BTL 1	Remember
10.	How does `ActionSupport` simplify the development process in Struts?	BTL 1	Remember
11.	What is the purpose of the `execute()` method in a Struts action class?	BTL 1	Remember
12.	Differentiate between `Action` and `ActionSupport`.	BTL 2	Understand
13.	What is a result in Struts?	BTL 2	Understand
14.	Name two default result types provided by Struts.	BTL 2	Understand
15.	What is the use of the `dispatcher` result type?	BTL 1	Remember
16.	How can you configure multiple results for a single action?	BTL 2	Understand
17.	What is OGNL in Struts?	BTL 2	Understand
18.	Define the value stack and its purpose in Struts.	BTL 1	Remember
19.	How do custom tags access data from the value stack?	BTL 1	Remember

20.	Give an example of an OGNL expression used in Struts.	BTL 2	Understand
21.	What are Struts form tags used for?	BTL 1	Remember
22.	How does Struts perform form validation?	BTL 2	Understand
23.	Explain the concept of type conversion in Struts.	BTL 2	Understand
24.	Mention one built-in validator used in Struts form validation.	BTL 1	Remember

PART-B

Q.No.	Question	Level	Competence
1.	Explain how Struts supports Agile development. Include advantages and practices.	BTL 4	Analyze
2.	Describe the basic configuration files used in a Struts application.	BTL 4	Analyze
3.	Explain the roles of Action and ActionSupport classes in the Struts framework.	BTL 4	Analyze
4.	Discuss the types of results and result types available in Struts with examples.	BTL 4	Analyze
5.	Describe OGNL and explain how it works with the value stack in Struts.	BTL 5	Evaluate
6.	How are custom tags implemented and used in a Struts application?	BTL 3	Apply
7.	Explain the use of form tags in Struts and how they simplify HTML form creation.	BTL 4	Analyze
8.	Illustrate the process of form validation using Validator Framework in Struts.	BTL 5	Evaluate
9.	Explain type conversion in Struts with a suitable example.	BTL 4	Analyze
10.	Describe the mechanism of exception handling and logging in Struts.	BTL 3	Apply
11.	Write a short note on how JavaScript enhances interactivity in Struts-based applications.	BTL 3	Apply
12.	Describe the use of JavaScript DOM and CSS in modern web UI development.	BTL 4	Analyze
13.	Explain how themes and templates are applied in Struts to support UI customization.	BTL 4	Analyze
14.	Compare the functionality of built-in and custom themes in Struts.	BTL 3	Apply
15.	Discuss the advantages and features of Rich Internet Applications (RIAs).	BTL 3	Apply
16.	How does advanced JavaScript improve client-side functionality in web applications?	BTL 4	Analyze
17.	Explain the integration of CSS with JavaScript for enhanced UI development in Struts.	BTL 4	Analyze

PART – C

1.	Develop and explain a login form using Struts with form validation and error handling.	BTL 5	Evaluate

2.	Design a mini application using Struts, demonstrating the flow from Action class to Result with proper configuration.	BTL 6	Create
3.	Compare traditional web applications and Rich Internet Applications in terms of user experience, architecture, and tools.	BTL 5	Evaluate
4.	Explain the entire process of applying themes, CSS, and templates in a dynamic form-based Struts application.	BTL 5	Evaluate
5.	Analyze the importance of combining OGNL, custom tags, and value stack in managing dynamic data flow in Struts apps.	BTL 6	Create

UNIT IV PYTHON WEB FRAMEWORKS

Introduction to Python Frameworks-Web 2.0, Python, and Frameworks-The Role of AJAX in Web 2.0-Web 2.0 with Traditional Python-Introducing the Frameworks-Web Application Frameworks-MVC in Web Application Frameworks-Common Web Application Framework Capabilities

PART – A

Q.No.	Question	Level	Competence
1.	What is a Python web framework?	BTL 1	Remember
2.	List any two popular Python web frameworks.	BTL 1	Remember
3.	Mention one benefit of using frameworks in Python-based web development.	BTL 1	Remember
4.	How do Python frameworks help in rapid application development?	BTL 1	Remember
5.	Define Web 2.0.	BTL 2	Understand
6.	How is Web 2.0 different from traditional web?	BTL 2	Understand
7.	Explain the role of Python in building Web 2.0 applications.	BTL 1	Remember
8.	Mention any two characteristics of Web 2.0 applications.	BTL 2	Understand
9.	What is AJAX?	BTL 1	Remember
10.	How does AJAX enhance the user experience in Web 2.0 applications?	BTL 2	Understand
11.	Mention two technologies used to implement AJAX.	BTL 1	Remember
12.	How is AJAX supported in Python web frameworks?	BTL 2	Understand
13.	What are the limitations of traditional Python in Web 2.0 development?	BTL 1	Remember
14.	How can traditional Python scripts be integrated into modern web apps?	BTL 2	Understand
15.	Explain the need for frameworks over standalone Python scripts.	BTL 2	Understand

16.	Give one example of using Python CGI for basic web interaction.	BTL 1	Remember
17.	What are micro frameworks in Python?	BTL 2	Understand
18.	Differentiate between full-stack and micro frameworks.	BTL 2	Understand
19.	Give an example of a full-stack Python framework.	BTL 1	Remember
20.	Give an example of a micro-framework in Python.	BTL 2	Understand
21.	Define the MVC architecture in the context of web frameworks.	BTL 1	Remember
22.	What is the role of the View in MVC?	BTL 1	Remember
23.	How does the Controller interact with the Model in MVC?	BTL 2	Understand
24.	List two Python web frameworks that follow the MVC pattern.	BTL 1	Remember

PART-B

Q.No.	Question	Level	Competence
1.	Explain the evolution of Python web frameworks with examples.	BTL 4	Analyze
2.	Describe the concept of Web 2.0. How has it influenced Python web development?	BTL 4	Analyze
3.	Discuss how AJAX plays a crucial role in Web 2.0 applications.	BTL 4	Analyze
4.	Explain how traditional Python scripting methods were adapted for Web 2.0 environments.	BTL 4	Analyze
5.	Describe the key components of a modern Python web application framework.	BTL 4	Analyze
6.	Write notes on any two Python frameworks and compare their use cases.	BTL 3	Apply
7.	Explain the features and purpose of the Model-View-Controller (MVC) architecture.	BTL 4	Analyze
8.	How is MVC implemented in Python frameworks like Django or Flask? Provide a brief example.	BTL 4	Analyze
9.	List and explain any four common Python web frameworks used today.	BTL 4	Analyze
10.	Describe the features of Web 2.0 and explain how Python supports these features.	BTL 5	Evaluate
11.	Explain the limitations of using traditional Python for web applications and how frameworks solve them.	BTL 4	Analyze
12.	Describe how AJAX enhances responsiveness in Python-based web applications.	BTL 3	Apply
13.	Write a note on the server-side vs client-side logic in Python web development.	BTL 3	Apply
14.	Discuss the benefits of using a full-stack framework compared to a micro framework.	BTL 3	Apply
15.	Explain how Python web frameworks simplify database connectivity and CRUD operations.	BTL 4	Analyze

16.	Write the common features provided by most Python web frameworks?	BTL 4	Analyze
17.	Describe a real-world application where a Python framework (e.g., Django) is effectively used.	BTL 3	Apply
PART-C			
1.	Develop a basic web application using any Python framework and explain how MVC architecture is followed.	BTL 6	Create
2.	Compare traditional web development in Python with modern framework-based approaches.	BTL 6	Create
3.	Analyze the importance of AJAX in enhancing user interaction in Python-based Web 2.0 applications.	BTL 6	Create
4.	Design and describe the structure of a Web 2.0 application using Django, highlighting the key components.	BTL 6	Create
5.	Evaluate the strengths and weaknesses of at least three common Python web frameworks with respect to scalability and ease of use.	BTL 5	Evaluate

UNIT V TURBOGEARS WEB FRAMEWORK

Introduction to TurboGears -TurboGears History-Main TurboGears Components-Alternate Components-MVC Architecture in TurboGears-Creating an Example Application-The Controller and View-Introduction to Django-Django History-Django Components-Alternate Components-MVC Architecture in Django-Creating an Example Application

PART – A

Q.No.	Question	Level	Competence
1.	What is TurboGears?	BTL 1	Remember
2.	Who developed TurboGears and in which year was it introduced?	BTL 1	Remember
3.	Mention any two goals of TurboGears during its development.	BTL 2	Understand
4.	List two differences between TurboGears 1.x and 2.x versions.	BTL 1	Remember
5.	Name any two main components of TurboGears.	BTL 1	Remember
6.	What is the function of the Genshi template engine in TurboGears?	BTL 2	Understand
7.	List any two alternate components that can be used with TurboGears.	BTL 2	Understand
8.	What role does SQLAlchemy play in TurboGears?	BTL 1	Remember
9.	Define the MVC pattern as used in TurboGears.	BTL 2	Understand
10.	Explain the role of the "Controller" in the TurboGears MVC architecture.	BTL 2	Understand
11.	What is the use of the "Model" in TurboGears?	BTL 2	Understand
12.	How does TurboGears render views in its architecture?	BTL 2	Understand
13.	What command is used to create a new TurboGears project?	BTL 1	Remember

14.	Describe how routing is handled in a TurboGears application.	BTL 2	Understand
15.	How are templates linked with controllers in TurboGears?	BTL 2	Understand
16.	What is the role of the `start()` function in a TurboGears application?	BTL 1	Remember
17.	What is Django?	BTL 2	Understand
18.	Who created Django and in which context was it developed?	BTL 1	Remember
19.	List two major versions of Django and their release years.	BTL 2	Understand
20.	Mention one key reason for Django's popularity.	BTL 2	Understand
21.	Name any two core components of Django.	BTL 1	Remember
22.	List two alternative template engines that can be used with Django.	BTL 1	Remember
23.	What is the function of the Django ORM?	BTL 2	Understand
24.	Explain how middleware enhances functionality in Django.	BTL 1	Remember

PART-B

Q.No.	Question	Level	Competence
1.	Explain the history and evolution of the TurboGears web framework.	BTL 4	Analyze
2.	Describe the main components of the TurboGears framework with examples.	BTL 3	Apply
3.	Write notes on alternate components that can be integrated into TurboGears.	BTL 4	Analyze
4.	Describe the MVC architecture implementation in TurboGears.	BTL 4	Analyze
5.	Explain how to create a basic web application using TurboGears.	BTL 4	Analyze
6.	Discuss the role and functionality of Controllers and Views in TurboGears.	BTL 4	Analyze
7.	Outline the key features that differentiate TurboGears from other Python frameworks.	BTL 4	Analyze
8.	Describe the history and design philosophy behind Django.	BTL 4	Analyze
9.	Explain the major components of Django and their responsibilities.	BTL 4	Analyze
10.	Write a note on alternate or pluggable components used in Django.	BTL 4	Analyze
11.	Compare the MVC pattern of Django and TurboGears with examples.	BTL 3	Apply
12.	Explain how Django follows MVC (or MTV) architecture.	BTL 4	Analyze
13.	Describe the steps involved in creating a simple application using Django.	BTL 3	Apply
14.	Explain how the controller (views) and templates work together in Django.	BTL 4	Analyze
15.	How does TurboGears simplify full-stack development? Illustrate with its component interactions.	BTL 4	Analyze

16.	Compare Django and TurboGears in terms of flexibility, ease of use, and application structure.	BTL 4	Analyze
17.	Discuss the routing and URL mapping mechanism in Django with suitable examples.	BTL 3	Apply
PART - C			
1.	Develop a sample TurboGears application and explain how MVC is maintained throughout.	BTL 6	Create
2.	Build a Django application that includes models, views, templates, and URL configuration. Explain each step.	BTL 6	Create
3.	Analyze and compare the architectural strengths and weaknesses of Django and TurboGears frameworks.	BTL 5	Evaluate
4.	Design an online student portal using Django or TurboGears and describe how components interact in MVC.	BTL 6	Create
5.	Evaluate the suitability of Django and TurboGears for large-scale enterprise applications.	BTL 5	Evaluate