### CIT 4302: WEB APPLICATION DEVELOPMENT

**Contact Hours:** 45 hours

**Prerequisites:** CIT 4103 Introduction to Web Programming, CCI 4201 Database Systems

**Purpose:** This unit provides students with the skills necessary to build secure, responsive, dynamic and database driven web applications by combining server-side programming with database integration, while adhering to the MVC (Model-View-Controller) architecture.

**Expected Learning Outcomes**

By the end of this unit of study the student should be able:

1. **Explain** server-side web development using a server-side language and database to build dynamic websites.
2. **Demonstrate** proficiency in procedural and object-oriented web programming language for clean, maintainable web code.
3. **Integrate** a web framework with Bootstrap for a workable front end and back end web development
4. **Develop** a secure web application with a framework and database, following security best practices.

**Course Content**

**Introduction to Server-Side Web Development**: Overview of server-side web development: fundamental concepts, including server-client architecture and web protocols; role of server-side technologies in building dynamic web applications; **Design Patterns and MVC Frameworks**: Purpose, structure, and application of common software design patterns; benefits, trade-offs, and architectural principles of the MVC design pattern; role of MVC in separating concerns within software systems; strengths and limitations of various MVC frameworks; **Web Services Design and RESTful APIs**: Principles of web service design and REST architecture; RESTful web services: design principles, concepts, and implementation; role of HTTP methods, status codes, and headers in RESTful communication; web APIs and interaction using JSON, XML, and HTTP; best practices and security considerations in RESTful API design; **Introduction to a Server-Side Language**: Fundamentals of a selected server-side language (PHP, Java, C#, Python, etc.); variables, data types, arrays, functions, and object-oriented programming (OOP) concepts; writing simple and object-oriented code to handle data processing, manipulate variables, and design basic applications; **Web Frameworks: Introduction and Setup**: Overview of popular web frameworks: Django (Python), Laravel (PHP), .NET (C#), and Spring (Java); installation, configuration, and directory structure; built-in tools for efficient application development; **Routes, Views, and Controllers**: Managing routes for different HTTP requests; dynamic views and templating using Blade (Laravel), Thymeleaf (Spring), Razor (.NET), and Jinja2 (Django);**Integration with Bootstrap Templates**: Customizing and integrating templates with a chosen framework; asset management, partials, layouting, and templating; **Database Migrations and Seeders**: Creating and testing migrations and seeders; ensuring database integrity through automated setup processes; **User Authentication and Security**: Implementing registration and login functionality; authentication and authorization mechanisms; securing user data in web applications.

**CRUD Operations in Web Frameworks**: Implementing Create, Read, Update, and Delete (CRUD) operations; understanding the significance of CRUD in dynamic web applications.

**Mode of Delivery**

Face to Face Lectures, Blended learning, tutorials, practical’s.

**Instructional Materials/Equipment**

* **Computer Laboratory**:  
  Equipped with modern computers or laptops capable of running web development software for server-side technologies, including PHP, C#, and Python.
* **Web Authoring Tools**:  
  Integrated Development Environments (IDEs) and text editors for web development, such as PhpStorm, Visual Studio, Visual Studio Code, PyCharm, Sublime Text, and other tools tailored for PHP, C#, and Python development.
* **Web Server Software(XAMPP)**:  
  Apache, Nginx, IIS, or other web server software suitable for deploying PHP, C#, or Python applications. This may include local server environments like XAMPP or WAMP for PHP, IIS for .NET, or Flask/Django development servers for Python.
* **Web Application Development Frameworks**:  
  Frameworks for building robust web applications, such as Spring(Java), **ASP.NET** (.NET), **Laravel** (PHP), and **Django** (Python). These frameworks provide the necessary structure for building dynamic, secure, and scalable enterprise applications.

**Assessment**

Type Weighting (%)

Examination 70%

Continuous Assessment 30%

Total 100%

**Core Text Books**

1. Nixon, R. (2020). Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 (6th ed.). O'Reilly Media.  
   ISBN-13: 978-1492051309,ISBN-10: 1492051304
2. Freeman, A. (2021). Pro ASP.NET Core 5 (1st ed.). Apress.ISBN-13: 978-1484261202  
   ISBN-10: 1484261200
3. Walls, C. (2020). Spring in Action (6th ed.). Manning Publications.ISBN-13: 978-1617297182, ISBN-10: 1617297180

**Core Journals**

1. *Journal of Web Semantics. ISSN: 1570-8268.*
2. *Journal of Web Development and Web Designing. ISSN: 1554-1045.*
3. *Internet Research. ISSN: 1066-2243.*

**Recommended Textbooks**

1. Williams, L. (2020). *PHP for the Web: Visual QuickStart Guide* (5th ed.). Peachpit Press.  
   ISBN-13: 978-0134444345,ISBN-10: 0134444345
2. Sang, J. (2021). *Core Java Volume I—Fundamentals* (11th ed.). Prentice Hall.ISBN-13: 978-0135166307, ISBN-10: 0135166306
3. Wrox, J. (2020). *Beginning C# and .NET* (9th ed.). Wiley.ISBN-13: 978-1119794694, ISBN-10: 1119794697

**Recommended Journals**

1. *International Journal of Web Engineering. ISSN: 1741-9212.*
2. *International Journal of Web Technology. ISSN:2278-2389.*
3. *ACM Transactions on the Web. ISSN: 1559-1131.*