COWLEY COLLEGE
& Area Vocational Technical School

COURSE PROCEDURE FOR

ASP.NET PROGRAMMING
CIS1865 3 Credit Hours

Student Level:
This course is open to students on the college level in either the Freshman or Sophomore year.

Catalog Description:

CIS1865 – ASP.NET PROGRAMMING (3 hrs)
This course will prepare students to write web applications using the ASP.NET framework. The topics will include: basic components, web technologies, data controls, server controls, master pages, state conditions, components for accessing data in a database, data sources, security, and deployment.

Prerequisites:
CIS1864 C# Programming

Co-requisites:
None

Controlling Purpose:
This course is designed to prepare students to write web applications. The student will learn the various techniques to harden the web application into a secure application suitable for industry using proper data handling, security, and authentication.

Learner Outcomes:
Upon completion of the course, the student will be able to write web applications that include the following ASP.NET components: basic components, web technologies, data controls, server controls, master pages, state conditions, components for accessing data in a database, data sources, security, and deployment.

Units Outcomes and Clock Hours of Instruction for Core Curriculum:
The following outline defines the minimum core content not including the final examination period. Instructors may add other material as time allows.

Evaluation Key:

A	=	All major and minor goals have been achieved and the achievement level is considerably above the minimum required for doing more advanced work in the same field.

B = All major goals have been achieved, but the student has failed to achieve some of the less important goals. However, the student has progressed to the point where the goals of work at the next level can be easily achieved.

C = All major goals have been achieved, but many of the minor goals have not been achieved. In this grade range, the minimum level of proficiency represents a person who has achieved the major goals to the minimum amount of preparation necessary for taking more advanced work in the same field, but without any major handicap of inadequacy in his background.

D = A few of the major goals have been achieved, but the student's achievement is so limited that he is not well prepared to work at a more advanced level in the same field.

F = Failing, will be computed in GPA and hours attempted.

N = No instruction or training in this area.

UNIT 1: Introduction to ASP.NET Development

Outcomes: Demonstrate knowledge of the basics of the ASP.NET development environment as well as working with a basic web form

A	B	C	D	F	N	Specific Competencies
						Describe the ASP.NET development environment including the .NET framework
						Explain how an ASP.NET application works
						Create a web form using Visual Studio
						Add validation controls to a form
						Add C# code to a form
						Test a web application

UNIT 2: HTML5, CSS3, and BootStrap with ASP.NET

Outcomes: Demonstrate the knowledge necessary write HTML, HTML5, CSS, and utilize Bootstrap for web design

A	B	C	D	F	N	Specific Competencies:
						Code basic HTML, HTML5, and CSS elements
						Incorporate HTML and CSS in Visual Studio
						Use BootStrap for responsive web design

UNIT 3: Multi-page Web Application and Debugging

Outcomes: Implement techniques to use multiple pages in a web application as well as using testing and debugging techniques in finding errors

A	B	C	D	F	N	Specific Competencies
						Create multi-page web applications
						Design and use data sources
						Use session state
						Test an ASP.NET web application
						Use the debugger and trace feature

UNIT 4: Standard Server Controls and Validation Controls

Outcomes: Implement programs that use the standard server controls and validation controls

A	B	C	D	F	N	Specific Competencies
						Describe the purpose of the standard server controls
						Use the common server controls, button controls, and list controls
						Implement the validation controls
						Discuss validation techniques

UNIT 5: State, cookies, URL encoding, and Master Pages

Outcomes: Write simple programs to keep track of the state of a web application using view states, session states, application states, cookies, and URL encoding. As well as deploy master and content pages

A	B	C	D	F	N	Specific Competencies:
						Use the view, session, and application states
						Implement cookies and URL encoding
						Create master pages
						Develop content pages
						Customize content pages

UNIT 6: Bootstrap in ASP.NET, Friendly URLS, and Routing

Outcomes: Implement programs that use Bootstrap, Friendly URLS, and routing to simplify interfaces with the web

A	B	C	D	F	N	Specific Competencies:
						Work with the Bootstrap CSS classes
						Use the Bootstrap components
						Implement Bootstrap themes
						Control the rendered HTML
						Use the FriendlyURLS feature
						Implement ASP.NET routing
						Combine ASP.NET routing with FriendlyURLS

UNIT 7: Database Programming and SQL Data Sources

Outcomes: Write basic SQL to view, update, insert and delete data in a database. Implement an SQL data source in an application using an SQL data source

A	B	C	D	F	N	Specific Competencies:
						Use SQL to work with data in a relational database
						Work with database objects
						Describe and implement ADO.NET
						Create a SQL data source
						Use custom statement and stored procedures
						Implement data binding and the DataList control

UNIT 8: GridView, DetailView, FormView, ListView, and DataPager Controls

Outcomes: Write programs that use a GridView, DetailView, FormView, ListView, and DataPager controls to view and update data in a database

A	B	C	D	F	N	Specific Competencies:
						Customize the GridView control
						Update GridView data
						Work with template fields
						Update the data in a DetailsView control

						Use the FormView control
						Implement the ListView control
						Update ListView data
						Use the DataPager control

UNIT 9: Object Data Sources with ADO.NET, Model Binding and the Entity Framework

Outcomes: Implement different methods of access data including ADO.NET, model binding, LINQ, and an Entity Data Model

A	B	C	D	F	N	Specific Competencies:
						Describe object data sources
						Create a data access class
						Use paging and sorting with object data sources
						Create an Entity Data Model
						Use LINQ to Entities
						Implement model binding to display, update, insert, and delete data
						Utilize data annotations to validate data

UNIT 10: Web Security, Authentication, and Authorization

Outcomes: Explain the process of implementing various security protocols in a web application including authentication and authorization

A	B	C	D	F	N	Specific Competencies:
						Describe TLS/SSL
						Use a secure connection
						Define the basics of authentication
						Create a web application that authenticates users
						Authorize users
						Customize users

UNIT 11: Email, Custom Error Pages, and Back-Button Control in an Application

Outcomes: Explain the process of implementing email, custom error pages, and back-button control in a web application

A	B	C	D	F	N	Specific Competencies:
						Send an email from a form
						Create custom error handling
						Handle the back-button problem

UNIT 12: ASP.NET Ajax, WCF, Web API, and MVC

Outcomes: Explain the process of implementing various web APIS including Ajax, WCF, Web API, and MVC

A	B	C	D	F	N	Specific Competencies:
						Describe the purpose of ASP.NET Ajax
						Use the ASP.NET Ajax server controls
						Define web services
						Create a WCF service
						Build a web application that consumes a WCF service
						Create a Web API service
						Implement a web application that consumes a Web API service
						Describe MVC and ASP.NET MVC
						Work with views
						Work with controls and postbacks

UNIT 13: Design of the Application Architecture

Outcomes: Explain the process of designing the architecture of an application considering application layers, role life cycle, state management, caching strategy, WebSocket strategy, HTTP modules and handlers

A	B	C	D	F	N	Specific Competencies:
						Plan the application layers
						Design a distributed application
						Design and implement the Windows Azure role life cycle
						Configure state management
						Design a caching strategy
						Plan and implement a WebSocket strategy
						Design HTTP modules and handlers

UNIT 14: Design of the User Experience

Outcomes: Explain the process of developing the UI layout to maximize the user experience

A	B	C	D	F	N	Specific Competencies:
						Apply the user interface design for a web application
						Design and implement UI behavior
						Compose the UI layout of an application
						Enhance application behavior and style based on browser feature detection
						Plan an adaptive UI layout

UNIT 15: Development of the User Experience

Outcomes: Explain the process of planning for search engine optimization, accessibility, globalization, localization, MVC controllers, routes, and network bandwidth considerations

A	B	C	D	F	N	Specific Competencies:
						Plan for search engine optimization and accessibility
						Design and implement globalization and localization
						Plan and implement MVC controllers and actions
						Build and implement routes
						Control application behavior by using MVC extensibility points
						Reduce network bandwidth

UNIT 16: Troubleshoot and Debug Web Applications

Outcomes: Explain the process troubleshooting and debugging web applications and Windows Azure applications

A	B	C	D	F	N	Specific Competencies:
						Describe how to prevent and troubleshoot runtime issues
						Design an exception handling strategy
						Test a web application
						Debug a Windows Azure application

UNIT 17: Design and Implement Security

Outcomes: Explain how to implement security using authentication, data integrity, and site considerations

A	B	C	D	F	N	Specific Competencies:
						Configure authentication
						Configure and apply authorization
						Design and implement claims-based authentication across federated identity store
						Manage data integrity
						Implement a secure site with ASP.NET

Projects Required:

Varies, refer to syllabus.

Textbook:
Contact Bookstore for current textbook.

Materials/Equipment Required:
None

Attendance Policy:
Students should adhere to the attendance policy outlined by the instructor in the course syllabus.

Grading Policy:
The grading policy will be outlined by the instructor in the course syllabus.

Maximum class size:
Based on classroom occupancy

Course Time Frame:
The U.S. Department of Education, Higher Learning Commission and the Kansas Board of Regents define credit hour and have specific regulations that the college must follow when developing, teaching and assessing the educational aspects of the college. A credit hour is an amount of work represented in intended learning outcomes and verified by evidence of student achievement that is an institutionally-established equivalency that reasonably approximates not less than one hour of classroom or direct faculty instruction and a minimum of two hours of out-of-class student work for approximately fifteen weeks for one semester hour of credit or an equivalent amount of work over a different amount of time. The number of semester hours of credit allowed for each distance education or blended hybrid courses shall be assigned by the college based on the amount of time needed to achieve the same course outcomes in a purely face-to-face format.

Refer to the following policies:

402.00 Academic Code of Conduct
263.00 Student Appeal of Course Grades
403.00 Student Code of Conduct

Disability Services Program:
Cowley College, in recognition of state and federal laws, will accommodate a student with a documented disability. If a student has a disability which may impact work in this class and which requires accommodations, contact the Disability Services Coordinator.