Utah System of Higher Education

Software Development

FY2023 / 30 Credits (900 Clock-Hours)

Software Development			
Institutions: Davis, Ogden-Weber, Southwest, Tooele			
Certificate of Program Completion (Catalog Year: 2023, 30 Credits/900 Clock-Hours Required, CIP: 11.0201)			
Core (21 Credits/630 Clock-Hours)		**Credits**	**Clock-Hours**
TESD 1400	Computer Programming	4	120
TESD 1800	Software Development	4	120
TESD 1100	Client-side Web Development	4	120
TESD 1500	Database Development	4	120
TESD 1700	Server-side Web Development	4	120
TESD 1050	Job Seeking Skills	1	30
Electives (9 Credits/270 Clock-Hours)			
Davis Technical College			
TESD 1640	Mobile Development	4	120
TESD 1930	Introduction to DevOps	1	30
TESD 1410	C++ Programming I	4	120
TESD 1411	C++ Programming II	3	90
TESD 2840	Capstone Project	4	120
TESD 2914	Software Development Externship	4	180
TESD 2851	Special Project I	1	30
TESD 2852	Special Project II	2	60
TESD 2853	Special Project III	3	90
Ogden-Weber Technical College			
TESD 1030	Foundations of Computing	4	120
TESD 1040	Software Development Math	1	30
TESD 1135	JavaScript	2	60
TESD 2830	Capstone Project	2	60
Southwest Technical College			
TESD 1180	Advanced Web Development	4	120
TESD 1600	Android App Programming	4	120
TESD 1610	IOS App Programming	4	120
TESD 1420	Advanced Java Programming	4	120
TESD 1430	Python Programming	4	120
TESD 1620	Computer Game Programming	4	120
TESD 2860	Final Advanced Project	1	30
Tooele Technical College			
TESD 1810	Unit Testing and DevOps	2	60
TESD 2870	Capstone Project	2	60
TESD 1645	Mobile Development	4	120
TESD 1440	C# Programming I	3	90
TESD 1441	C# Programming II	3	90

UTAH SYSTEM OF HIGHER EDUCATION

Utah System of Higher Education

Software Development
FY2023 / 30 Credits (900 Clock-Hours)

PROGRAM DESCRIPTION

The Software Development program provides students with the opportunity to learn the languages and tools needed to start a career as a Software Developer. Languages and technologies taught will cover the spectrum from the front-end user interface to the back-end server, both desktop and mobile. Students will learn technologies and skills like Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript, User interface and User Experience (UI/UX), Application Programming Interfaces (API), Developer Operations (DevOps), Cloud Computing, Version Control, Leadership, and Soft Skills. Students will learn code documenting, testing, and debugging to create and maintain software applications.

Objectives:

- Explore modern software development
- Deliver software applications using current project management practices
- Demonstrate problem-solving skills with computer programming
- Practice workplace professionalism
- Demonstrate the use of techniques and tools necessary for Software Development

COURSE DESCRIPTIONS

Computer Programming **4 Credits/120 Clock-Hours**

In Computer Programming, students will use critical thinking, and problem-solving skills as they practice basic programming constructs including: selection, repetition, classes and methods, string processing, and array structures. Students will be introduced to version control on their code projects.

Objectives:

- Describe object-oriented programming
- Practice using procedures, methods and functions
- Create and use classes
- Apply structured programming techniques
- Utilize Version Control

Software Development **4 Credits/120 Clock-Hours**

In this course, students will explore the Software Development Life Cycle. Students will test code, practice searching, sorting, building data structures, using generic objects and collections, and asynchronous processing. Students will be exposed to modern project management styles.

Objectives:

- Practice project management techniques
- Design feature specification for software
- Apply Code Design patterns
- Design code using common data structures
- Explore the Software Development Lifecycle

UTAH SYSTEM OF HIGHER EDUCATION

Utah System of Higher Education

Software Development
FY2023 / 30 Credits (900 Clock-Hours)

Client-side Web Development — 4 Credits/120 Clock-Hours

Client-Side Web Development provides experience developing websites using current standards and technologies. Students will be exposed to modern scripting and the Document Object Model (DOM) of web pages. Students will create functional websites, making them interactive and dynamic.

Objectives:

- Implement common HTML tags in a functional coding format to create a Web site using current standards and technologies
- Demonstrate the ability to upload and publish a web page on a web server
- Investigate web scripting and the HTML DOM / Manipulate the DOM using web scripts
- Use web hosting service to deploy a website
- Explore best practices in modern responsive website design

Database Development — 4 Credits/120 Clock-Hours

Database Development provides students a fundamental introduction to database concepts and query languages used in database management systems. Students will design and implement simple databases, and utilize queries to retrieve, store, and update data in these databases.

Objectives:

- Recognize core database concepts
- Describe database objects: data types, views and stored procedures
- Utilize basic SQL to interact with databases
- Explain data storage concepts: normalization; primary, foreign, and composite keys; and indexes
- Use basic information assurance and database security concepts

Server-side Web Development — 4 Credits/120 Clock-Hours

Server-side programming explores delivering a customized user experience. This course combines the skills of programming, client-side development, and relational database management to create and manage dynamic web-based content. Students will be exposed to using, creating, and testing web APIs.

Objectives:

- Implement server-side programming to serve the client-side development
- Demonstrate proper syntax, patterns, data structures, and functional usage of server-side language
- Connect and utilize database
- Develop controls and event-handling procedures
- Apply server-side concepts and techniques to create, manage, and use dynamic web pages
- Employ API testing and development

Job Seeking Skills — 1 Credit/30 Clock-Hours

Job Seeking Skills explores how to prepare and successfully apply to potential career opportunities. During this course, you will be presented with essential job-seeking skills needed to find gainful employment.

Objectives:

- Create a professional resume, cover letter and reference sheet
- Utilize online tools successfully to create an e-portfolio

- Expand and develop networking skills
- Utilize online resources effectively to find job openings
- Demonstrate the ability to fill out job applications in a professional manner
- Perform successfully in a job interview
- Demonstrate appropriate follow-up procedures

NON-ALIGNED (ELECTIVE) COURSES
Davis Technical College

Mobile Development **4 Credits/120 Clock-Hours**

This course introduces students to programming technologies, design and development related to mobile applications. Topics include accessing device capabilities, industry standards, operating systems, and programming for mobile applications. Students will work on multiple projects producing professional-quality mobile applications.

Objectives:

- Use a modern programming framework to create mobile apps
- Use navigation between screens in an app
- Utilize native features like the camera and maps
- Add style to objects and components on multiple screens in apps
- Work with local storage, and connect with databases for long term storage

Introduction to DevOps **1 Credit/30 Clock-Hours**

The Introduction to DevOps course will expose students to DevOps's cultural mindset and tools. Students will practice creating containers for software applications to run in for fast deployment and frequent releases. Students will practice managing multiple containers with current industry technology during this course. Students will explore how Continuous Integration and Continuous Delivery are integral parts of the software application life cycle.

Objectives:

- Explore how DevOps (Development Operations) fits in the Software Development Life Cycle
- Practice creating and using software containers
- Experience orchestrating multiple containers with container management software
- Explore Continuous Integration and Continuous Delivery (CI/CD)

C++ Programming I **4 Credits/120 Clock-Hours**

This course introduces students to the C++ language and object-oriented programming. Students will practice critical thinking, problem solving skills, and basic testing processes as they are introduced to basic C++ programming constructs including selection and repetition controls, classes and methods, string processing, array structures, input and output, searching, sorting, and vectors.

Objectives:

- Create and use basic C++ programs
- Incorporate selection and repetition controls in C++ programs
- Practice using functions, classes and objects, and vectors
- Demonstrate searching and sorting algorithms in arrays and vectors

- Include pointers in C++ programs
- Demonstrate use of C-strings and strings in C++ programs

C++ Programming II **3 Credits/120 Clock-Hours**

This course gives students more experience in intermediate C++ programming language and object-oriented programming constructs, such as data abstraction and inheritance. As students complete this course, they will develop competency in pointers, virtual functions, abstract classes, templates, exception handling, recursion, and data structures

Objectives:

- Create and use classes with inheritance, aggregation, and composition
- Include virtual functions, abstract classes, and pure virtual functions in C++ programs
- Practice advanced file and I/O operations with random access files
- Include recursion in C++ programs
- Use template functions to handle multiple data types
- Practice exception handling
- Include functions in the Standard Template Library in C++ programs
- Incorporate pointers in C++ programs
- Demonstrate the use of linked lists, stacks, queues, and binary trees

Capstone Project **4 Credits/120 Clock-Hours**

The capstone course allows students to demonstrate how the knowledge and skills learned through the Software Development program can be applied to solving real-world business problems. Individually or in a small group, students will find a real-world business problem to solve. Students will research and understand the business case. Creating a scope for the project and setting the timeline for the deliverables. Then develop the solution according to the plan. At the completion of the project, the individual or team will present their capstone project to the Software Development class.

Objectives:

- Demonstrate knowledge and skills learned in the Software Development program
- Work collaboratively with a team or user to develop a software project
- Solve a real-world problem
- Research and understand a business case that software can solve
- Practice program management
- Present project to an audience

Software Development Externship **4 Credits/180 Clock-Hours**

This course provides an opportunity for Software Development students to gain professional exposure to the technologies learned in the program through internship, externship, or job-shadowing, as determined by employer-college relationships.

Objectives:

- Experience a real-world software development
- Practice developing software as a team
- Report to a project manager with task progress
- Practice time management
- Work in a live production environment

Special Project I **1 Credit/30 Clock-Hours**

This course provides students the opportunity to continue their learning in a specific area, explore an area which is not currently available as part of their current training plan, or otherwise not covered by the ongoing Software Development program course offerings. Coursework will consist of instructional activity and/or a significant professional project that is logically consistent with the content of the student's program of study or a special project guided and evaluated by a member of the Software Development faculty.

Objectives:

- Explore advanced skills used in the Software Development workplace via a special project and instruction related to a student's career goals.

Special Project II **2 Credits/60 Clock-Hours**

This course provides students the opportunity to continue their learning in a specific area, explore an area which is not currently available as part of their current training plan, or otherwise not covered by the ongoing Software Development program course offerings. Coursework will consist of instructional activity and/or a significant professional project that is logically consistent with the content of the student's program of study or a special project guided and evaluated by a member of the Software Development faculty.

Objectives:

- Explore advanced skills used in the Software Development workplace via a special project and instruction related to a student's career goals.

Special Project III **3 Credits/90 Clock-Hours**

This course provides students the opportunity to continue their learning in a specific area, explore an area which is not currently available as part of their current training plan, or otherwise not covered by the ongoing Software Development program course offerings. Coursework will consist of instructional activity and/or a significant professional project that is logically consistent with the content of the student's program of study or a special project guided and evaluated by a member of the Software Development faculty.

Objectives:

- Explore advanced skills used in the Software Development workplace via a special project and instruction related to a student's career goals.

Ogden-Weber Technical College

Foundations of Computing **4 Credits/120 Clock-Hours**

This course provides a solid foundation in computer science topics with industry applications. In this course there is no expectation of a computing background. It introduces concepts such as ethical issues in computing, networks, operating systems, databases, problem solving and programming.

Objectives:

- Describe historical implications of computing
- Describe social implications of computing
- Explain computing security

- Explain computing ethics
- Identify computer architectures
- Explain networks
- Describe the internet
- List database fundamentals
- Describe numbering systems
- Explain data representations
- Identify data structures
- List operating systems
- Identify file structures
- Define the human-computer interface
- Explain problem solving
- Describe debugging
- Explain software engineering

Software Development Math **1 Credit/30 Clock-Hours**

This course teaches students the essential concepts of mathematics including algebra that Software Developers use. Using the skills developed through this course, students will be able to face the logical and mathematical challenges that programming represents.

Objectives:

- Demonstrate algebra concepts and explain their use in programming

JavaScript **2 Credits/60 Clock-Hours**

This course covers more advanced uses of JavaScript both client-side and server-side.

Objectives:

- Demonstrate consuming Web Services
- Use Node, Express, and React

Capstone Project **2 Credits/60 Clock-Hours**

This course provides an opportunity to complete a significant programming project from the design phase through implementation with minimal instructor support. Emphasis is placed on project definition, testing, presentation, and implementation. This course will explore how to complete a project from the definition phase through implementation

Objectives:

- Demonstrate time management principles
- Demonstrate the ability to meet deadlines for regular deliverables
- Demonstrate appropriate customer relations with regard to project changes including scheduled updates, revisions, etc.
- Presentation the capstone project using applicable presentation skills

Southwest Technical College

Advanced Web Development **4 Credits/120 Clock-Hours**

This course covers advanced web development and helps students understand some common frameworks available to web developers. Advanced layouts and styling will be taught to help websites look more professional. Students will create a few different types of websites to help them understand how these different websites are used by companies worldwide.

Objectives:

- Use advanced CSS to create responsive websites
- Explain common frameworks used in web development
- Implement several different types of websites commonly used by companies
- Practice deploying websites on live web servers

Android App Programming **4 Credits/120 Clock-Hours**

Android apps are used constantly by mobile users throughout the world. Understanding how these apps work and are programmed is a highly sought-after skill in today's job market. Our Android App Programming course covers core concepts to help students create working Android apps. Building a reliable app that uses several Android App API's will give students the job ready skills they need.

Objectives:

- Develop a user interface using different types of controls
- Explore user input, variables, and operations
- Use lists, arrays, and Web browsers in an Android app
- Include audio such as music in Android apps
- Create an Android app that requests, stores, and modifies data for multiple activities

IOS App Programming **4 Credits/120 Clock-Hours**

IOS apps are used by many users throughout the world today. Using Swift and Xcode, students will learn how to build working IOS apps. This course will also help students debug and test their IOS apps as they learn layouts, controllers, and functions that explore the Apple devices these apps will be deployed on.

Objectives:

- Discuss design and color theories
- Use Sketch to build app designs
- Navigate Xcode
- Design in Swift Playground and Xcode Storyboard
- Explain different controllers and how to use them

Advanced Java Programming **4 Credits/120 Clock-Hours**

This course covers advanced Java programming concepts such as generics, data structures, search trees, and advanced JavaFX design and implementation. Students will gain a sound understanding of in-depth Java programming and will use their skills to create an advanced JavaFX layout utilizing these concepts.

Objectives:

- Explain generics and how they are used

- Explore common data structures
- Work with search trees and common data related to them
- Build a useful JavaFX layout using advanced Java programming concepts

Python Programming **4 Credits/120 Clock-Hours**

Python Programming integrates your previous programming experience with the Python syntax. While in this course, you will create programs involving graphics, image manipulation, GUIs, simple networked client/server applications, and stacks.

Objectives:

- Explain procedural abstraction in function definitions
- Manipulate graphics and image processing
- Implement networks and client/server programming
- Use events and event-driven programming
- Create and apply stacks and lists

Computer Game Programming **4 Credits/120 Clock-Hours**

Our computer game programming course covers a wide range of skills used by computer game programmers every day. We explore many different aspects of computer games including working with images, using databases in games, animations, and developing an efficient game loop. Students will develop a working game using many programming skills already learned throughout their courses such as objects, functions, and loops.

Objectives:

- Explain game loops and how to efficiently set them up
- Work with assets such as images and sounds
- Build effective animations
- Fine tune the user experience and create a working game

Final Advanced Project **1 Credit/30 Clock-Hours**

Students will plan out a project of their choosing utilizing their development skills. The project will cover the concepts they have learned throughout their courses. A presentation will be made to a group to show others the finalized project

Objectives:

- Plan a development project that uses previously taught skills
- Develop a website/program/app to be used by a user/company
- Present the finalized project to a group

Tooele Technical College

Unit Testing and DevOps **2 Credits/60 Clock-Hours**

The goal of this course is to give the students a basic understanding of how to build, run, and use Jest to test their code. It will also teach students how the life cycle of unit testing works and why we use it in modern code practices today. The other goal is to help them mock and test core business logic in a practical way. This will also include some important material on how to include additional tooling using DevOps.

Objectives:

- Install and run a unit test using Jest.
- Explain the unit test life cycle of unit testing
- Write good tests and explain what bad tests look like
- Test the integration of the code
- Test the DOM and frontend

Capstone Project **2 Credits/60 Clock-Hours**

The goal of this course is to allow students to demonstrate the skills they have learned from the core content and apply it to a real-world project. This project will give guidelines for students to put together their real-world project. It will have minimal instructor support alongside the emphasis on time management, testing, and presentation.

Objectives:

- Demonstrate programming skills learned
- Demonstrate time management skills
- Present a real-world solution to a real-world problem
- Display schedule planning skills with updates, revisions, versioning and more

Mobile Development **4 Credits/120 Clock-Hours**

The outcomes for this course will be to teach you how to read, write, and build mobile applications using React-Native. This course will include best practices for industry standards in mobile development, device capabilities, managing the difference between android and iOS platforms, and programming complex projects.

Objectives:

- Use a modern framework to build apps
- Demonstrate writing and understanding operating system differences
- Build complex applications using time management
- Use navigation between screens in an app
- Add style to objects and components on multiple screens in apps
- Work with local storage, and connect with databases for long term storage

C# Programming I **3 Credits/90 Clock-Hours**

This course introduces students to the C# language and object-oriented programming. Students will practice critical thinking, problem solving skills, and basic testing processes as they are introduced to basic C# programming constructs including selection and repetition controls, classes and methods, string processing, array structures, input and output, searching, sorting, and vectors.

Objectives:

- Create and use basic C# programs
- Incorporate selection and repetition controls in C# programs
- Practice using functions, classes and objects, and enums
- Demonstrate searching and sorting algorithms in arrays

C# Programming II **3 Credits/90 Clock-Hours**

This course will help students understand more complex uses of the C# language, building their skills in many diverse projects, utilizing frameworks, object-oriented programming, alongside some personal projects that students will build to display competencies in using the C# language.

Objectives:

- Explain advanced object-oriented programming
- Build a project using frameworks and software tools
- Build complex projects with many moving parts
- Use C# in the web development
- Create a CRUD application using C#