Foundational Courses

TESD 1050 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.

TESD 1100 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.

TESD 1400 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.

TESD 1500 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.

TESD 1700 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.

TESD 1800 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.

Supplemental Courses Varies by Institution

Bridgerland

TESD 1000 Frontend Framework **3 Credits / 90 Clock-Hours**

The Frontend Framework course equips students with the skills and knowledge necessary for proficient development of modern web applications using cutting-edge frontend frameworks. Frontend Framework focuses on empowering students with the tools and techniques essential for building dynamic, responsive, and user-friendly interfaces.

Objectives:
• Identify the role of frontend frameworks in website development.
• Create responsive user interfaces.
• Utilize component-based architecture.
• Demonstrate proper use of state management.
• Implement optimization and security best practices in a frontend framework.

TESD 1010 Backend Framework **3 Credits / 90 Clock-Hours**

The Backend Framework course empowers students with the expertise required to build robust and scalable server-side applications. This course delves into the intricacies of backend frameworks, providing students with the skills and knowledge needed to architect, implement, and maintain powerful backend systems.

Objectives:
• Identify the role of backend frameworks in website development.
• Integrate and manage databases using a backend framework.
• Create & consume Application Programming Interfaces (APIs).
• Construct authentication and authorization processes to verify users using a backend framework according to industry standards.
• Implement optimization and security best practices in a backend framework.

TESD 1110 Prototyping and Design **3 Credits / 90 Clock-Hours**

The Prototyping and Design course empowers students with the skills and knowledge needed to create compelling, user-centric designs and functional prototypes. In this course, students learn industry-standard design principles, tools, and methodologies to transform conceptual ideas into visually appealing and interactive software solutions.

Objectives:
• Prototype user experience (UX) for web and mobile development.
• Create a site map for project planning.
• Implement graphic design principles including color, typography, and layout.
• Create a cohesive website design using creative thinking and brainstorming methodologies.
• Utilize industry-standard software used for graphic design to create website mock-ups.

TESD 1510 Content Management Systems Essentials **3 Credits / 90 Clock-Hours**

The Content Management System Essentials course equips students with the essential skills to create, customize, and manage dynamic websites using a content management system (CMS). From basic website setup to advanced customization, students gain hands-on experience to become proficient CMS users.

Objectives:
• Properly install a content management system program.
• Create a website using a content management system platform.
• Utilize plugins and themes to improve site functionality.
• Secure a website using content management system security features.
• Develop a website using content management system themes.

TESD 1900 Fullstack Integration **3 Credits / 90 Clock-Hours**

The Fullstack Integration course bridges the gap between frontend and backend development, creating well-rounded fullstack developers. This course empowers students with the expertise to seamlessly integrate frontend and backend components, resulting in cohesive and fully functional web applications.

Objectives:
• Identify the integration of frontend and backend framework's role in website development.
• Integrate and manage databases across the stack.
• Create & consume Application Programming Interfaces (APIs).
• Construct authentication and authorization across the stack.
• Implement optimization and security best practices across the stack.

TESD 2835 Software Development Capstone **3 Credits / 90 Clock-Hours**

The Software Development Capstone course combines all of the components from the student’s chosen track into a capstone project that can be used in their portfolio. Students who complete this course are able to complete a simple project from the proposal stage all the way to presenting their finished product.

Objectives:
• Create a proposal for a project of the student’s choice.
• Establish and meet project deadlines.
• Organize and Manage project resources, such as files, media, APIs, or databases.
• Build on previous knowledge by utilizing additional learning resources.
• Develop a project with instructor input.

TESD 2840 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.

TEDA 1031 Python Fundamentals **3 Credits / 90 Clock-Hours**

This course teaches fundamental Python skills tailored for data analysis, encompassing Python's core syntax, data structures, and procedural programming techniques. Students perform data cleaning, data manipulation, and exploratory analysis using industry-standard libraries, fostering expertise in managing, analyzing, and visualizing data. Through practical projects, learners refine their abilities, gaining confidence to proficiently handle, analyze, and present data using Python. This course cultivates real-world application skills and sharpens proficiency in data project documentation, serving as a strong foundation for future data science endeavors.

Objectives:
• Describe core Python syntax, data structures, and procedural programming concepts within a Python Integrated Development Environment (IDE).
• Identify specific features and functionalities of the Python IDE, becoming proficient in executing tasks and optimizing workflows within the development environment.
• Clean, manipulate, and analyze data using industry-standard libraries within a Python IDE.
• Illustrate statistical analysis techniques by incorporating loops, joins, functions and decision-making skills.
• Characterize relevant business conclusions as revealed by the data using plots, data frames and aggregation.
• Document and present the project's findings using standard documentation practices.

Davis

TESD 1640 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.

TESD 2050 Version Control **1 Credit / 30 Clock-Hours**

Description: This course introduces students to the necessity of version control in software development. During this course, students will use GitHub to control the source of their coding projects and manage versions of their software during development.

Objectives:
• Analyze information to develop critical thinking and problem-solving skills.
• Use a variety of programming software applications.
• Recognize general software development processes.
• Use standard Git features to track software source code.

TESD 2851 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.

TESD 2852 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.

TESD 2853 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.

TESD 2914 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.

Dixie

TESD 1520 TypeScript Programming **3 Credits / 90 Clock-Hours**

TypeScript Programming introduces students to strongly-typed programming for the web using TypeScript. This course focuses on writing safe, predictable code by leveraging TypeScript's static type system and modern JavaScript features. Students will learn to write modular, maintainable, function-based code using TypeScript syntax and best practices.

Objectives:
• Explain the benefits of using TypeScript over JavaScript.
• Use TypeScript to define and enforce types for variables, functions, arrays, tuples, and objects.
• Write and organize modular function-based code using TypeScript.
• Use modern JavaScript features (e.g., destructuring, arrow functions) in TypeScript.
• Debug and test TypeScript code effectively.
• Follow best practices for readable, maintainable, and strictly-typed code.

TESD 1815 Capstone Prototyping **3 Credits / 90 Clock-Hours**

Capstone Prototyping prepares students to begin their final capstone project by designing and testing the core functionality of a planned application. Students will use rapid prototyping methods to explore user experience, technical feasibility, and project scope. Emphasis is placed on iteration, feedback, and building a working proof-of-concept.

Objectives:
• Propose a capstone project idea that solves a real-world problem.
• Define clear project goals, users, and use cases.
• Design wireframes or mockups to outline core user flows.
• Create a functional prototype that demonstrates the project’s main features.
• Evaluate feasibility through testing and technical research.
• Incorporate peer and instructor feedback into prototype revisions.
• Document prototype functionality and development decisions.

TESD 1820 Capstone **3 Credits / 90 Clock-Hours**

In the Capstone course, students complete a comprehensive, self-directed project that demonstrates their cumulative knowledge and technical skills. Building on the Capstone Prototyping course, students develop, test, and deploy a full application or system. The course emphasizes project management, documentation, and professional presentation.

Objectives:
• Plan and manage a full-scale software or web development project.
• Implement a functional, complete application or system from prototype.
• Use version control to track progress and collaborate effectively.
• Test, debug, and refine the final product based on user feedback.
• Write clear technical documentation for end users and developers.
• Present and demonstrate the completed project to a professional audience.

Ogden-Weber

TESD 2830 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.

TESD 1950 AI for Developers **2 Credits / 60 Clock-Hours**

This course introduces students to the practical use of Artificial Intelligence (AI) tools and local AI agents in software development workflows. Students learn to integrate AI for writing, debugging, testing, documenting, and managing projects. Emphasis is placed on configuring local AI environments for privacy and reliability, and on understanding the ethical use of AI-generated code. Topics include AI-assisted code generation, automated unit test creation, code review, documentation generation, and project mentoring. By the end of the course, students will have a functioning set of AI agents that support the complete software lifecycle, from code authoring to testing and documentation.

Objectives
- Explain ethical and practical uses of AI in software development.
- Configure and run local AI agents for secure, offline development.
- Use AI to generate, refactor, and debug code.
- Create and improve unit tests using AI assistance.
- Apply AI tools to review code for quality and best practices.
- Generate documentation with AI.
- Use AI agents as mentors and assistants for projects and learning support.
- Integrate AI workflows across the full software development lifecycle.

Southwest

TESD 1180 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.

TESD 1420 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.

TESD 1430 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.

TESD 1600 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.

TESD 1610 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.

TESD 1620 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.

TESD 2860 Final Advanced Project **4 Credits / 120 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

TESD 1530 Web Frameworking **3 Credits / 90 Clock-Hours**

This course examines the core processes involved in building web applications with modern frameworks. Topics include project planning, feature specification, architectural design, version control strategy, implementation workflows, testing practices, and iterative improvement. The course also addresses time management and deployment considerations within a complete development cycle.

Objectives:
• Identify major components of project planning and feature specification in web framework development.
• Outline architectural approaches, design considerations, and version control strategies.
• Explain general processes involved in implementation, testing, and quality assurance within a framework environment.
• Summarize iteration cycles, feedback loops, and revision stages support project refinement. Recognize principles of time management, milestone planning, and risk mitigation in software development.
• Organize project timelines, milestones, and risk-mitigation plans using appropriate management tools.

TESD 2010 DevOps Fundamentals **2 Credits / 60 Clock-Hours**

This course provides an introduction to using Jest for building, running, and managing code tests. It covers the unit-testing lifecycle, the role of testing in modern development practices, and practical methods for mocking and validating core business logic. The course also introduces additional tooling and integrations commonly used in DevOps environments.

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

TESD 2120 Cloud Development **2 Credits / 60 Clock-Hours**

This course introduces the core concepts of cloud technologies and how they compare to traditional local server environments. Topics include cloud service models, industry best practices, and an overview of major platforms such as AWS, Microsoft Azure, and Google Cloud. The course also covers coding and deployment across different operating systems as well as emerging trends in hybrid, multi-cloud, and serverless computing.

Objectives:
• Compare the differences between cloud computing and local servers
• Demonstrate coding for different operating systems
• Explain emergent trends in cloud computing
• Recognize differences in hybrid, multi-cloud, and serverless computing
• Complete an architecture cloud computing final project

TESD 2870 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.