

## COMPUTER SCIENCE III: Software Development

*Computer Science III: Software Development* focuses on gaining knowledge and acquiring competencies in the processes, techniques and tools used to develop production quality software. The course framework aligns with professional standards and situates software development within the context of a software project, providing focus on requirements development and management; project scheduling; project success metrics; code design, development and review principles; testing procedures; release and revision processes; and project archival. An additional topic provides exposure to career opportunities within the software development field. The final product of this capstone experience is a working software product that adheres to industry standards.

- DOE Code: 5249
- Recommended Grade Level: Grade 11-12
- Required Prerequisite: Computer Science I and Computer Science II
- Credits: 2 semester course, 2 semesters required, 1-3 credit per semester, maximum of 6 credits
- Counts as an elective Science credit or Directed Elective for all diplomas

### Implementation Guidance

Computer Science III: Software Development builds on skills developed in Computer Science I and Computer Science II to create a mobile, desktop, or web application. This course is designed to be taught in any object oriented programming language, meaning it is language neutral. However, the instructor should focus on one language throughout the duration of the course. It would be very beneficial for an instructor to work with an industry partner throughout the duration of the course. Additionally, all domains are an intrinsic part of the software development lifecycle. All domains are expected to be covered by the end of the second semester.

### Dual Credit

This course provides the opportunity for dual credit for students who meet postsecondary requirements for earning dual credit and successfully complete the dual credit requirements of this course. The Dual Credit crosswalk can be accessed [here](#).

### Career and Technical Student Organizations (CTSOs)

Career and Technical Student Organizations are considered a powerful instructional tool when integrated into Career and Technical Education programs. They enhance the knowledge and skills students learn in a course by allowing a student to participate in a unique program of career and leadership development. Students should be encouraged to participate in Business Professional of America, DECA, or Future Business Leaders of America, the CTSOs for this area.

## Content Standards

### Domain – Software Development Overview

**Core Standard 1** Students will create a plan to develop a software project.

#### Standards

- CSIII-1.1 Understand the phases of the software development life cycle.
- CSIII-1.2 Identify the different models for software development projects (example: Agile).
- CSIII-1.3 Understand the human and political considerations in steering a project.
- CSIII-1.4 Synthesize project requirements according to user/customer needs.
- CSIII-1.5 Compare and contrast different mobile platforms, systems software, and applications software, their associated tools and development process.

### Domain – Team Management and Collaboration

**Core Standard 2** Students collaborate to complete various tasks.

#### Standards

- CSIII-2.1 Design a solution to a problem by working in a team.
- CSIII-2.2 Explore technologies that can be used to collaborate with others of various cultures and career fields.
- CSIII-2.3 Utilize a problem solving approach to develop a solution using technology.
- CSIII-2.4 Analyze the work of peers and provide feedback.
- CSIII-2.5 Program a solution to a problem using pair programming or other methods.
- CSIII-2.6 Understand the different stages of team formation: leadership styles, decision-making approaches, and conflict resolution techniques.

### Domain – Project Management and Metrics

**Core Standard 3** A student will be able to construct a high level software development schedule using current industry practices.

#### Standards

- CSIII-3.1 Estimate the effort, resources, and time required to execute a project.
- CSIII-3.2 Develop requirements analysis and specification for project goals.
- CSIII-3.3 Conduct a feasibility study to determine project scope.
- CSIII-3.4 Develop an implementation plan.
- CSIII-3.5 Create scheduling visualizations (Examples: PERT and GANTT chart).
- CSIII-3.6 Report on project metrics, including completion, cost, and quality.

## **Domain – Code Design and Development**

**Core Standard 4** Students will design and develop a software solution.

- CSIII-4.1 Construct proper programming statements syntactically correct using variables, constants, character strings, arithmetic operators, expressions and statements.
- CSIII-4.2 Use control structures and methods in programs.
- CSIII-4.3 Implement classes using encapsulation, inheritance, polymorphism, and abstraction.
- CSIII-4.4 Design and implement user-defined classes and their methods and attributes for satisfying stated programming objectives.
- CSIII-4.5 Demonstrate usage of multi-dimensional arrays, array lists, queues, stacks, pointers and classes.
- CSIII-4.6 Demonstrate and compare iteration and recursion.
- CSIII-4.7 Create and access data files using sequential and random access I/O functions and streams.
- CSIII-4.8 Demonstrate how to create and utilize user defined data structures.
- CSIII-4.9 Describe and utilize searching and sorting techniques.
- CSIII-4.10 Understand and work with the implementation of Linked List data structures.
- CSIII-4.11 Develop applications using audio, images and animation.
- CSIII-4.12 Develop applications using persistent data.
- CSIII-4.13 Create web based databases applications.
- CSIII-4.14 Demonstrate an understanding of the basic concepts of good user-interface design.
- CSIII-4.15 Integrate exception handling where appropriate.
- CSIII-4.16 Use memory appropriately, including proper allocation/deallocation procedures.
- CSIII-4.17 Demonstrate usage of API's and standard libraries.

## **Domain – Testing and Release**

**Core Standard 5** Students will test and debug software.

### **Standards**

- CSIII-5.1 Test and debug use cases.
- CSIII-5.2 Execute black box testing to validate that the programming artifact meets customer requirements.
- CSIII-5.3 Execute white box testing to validate coding best practices and operation.
- CSIII-5.4 Optimize programming code for efficiency utilizing timers, event handlers, and messaging in software applications.
- CSIII-5.5 Publish applications to web or mobile platforms.

## **Domain – Maintenance, Updates, and Versioning**

**Core Standard 6** Students will manage maintenance and versioning of software.

- CSIII-6.1 Recommend a software release or revision as needed.
- CSIII-6.2 Implement version control.
- CSIII-6.3 Create a disaster recovery plan including backups.
- CSIII-6.4 Plan the expected retirement of a program.

## **Domain – Documentation**

**Core Standard 6** Students document their computing artifacts as part of the software development lifecycle.

### **Standards**

- CSIII-7.1 Describe the function of a computing artifact (for example, code or design).
- CSIII-7.2 Identify the purposes of a computing artifact.
- CSIII-7.3 Explain concepts related to a computing artifact
- CSIII-7.4 Describe how to use a computing artifact
- CSIII-7.5 Explain cause/effect by interpreting input and output.
- CSIII-7.6 Create documentation for computing artifact such as comments or user manual/readme.

## **Domain – Social, Legal, and Ethical Issues**

**Core Standard 8** Students will consider a variety of social, legal, and ethical issues in software engineering.

### **Standards**

- CSIII-8.1 Explain the privacy concerns related to the collection and generation of data through implicit and explicit processes.
- CSIII-8.2 Evaluate the social and emotional implications of privacy in the context of safety, law, and ethics.
- CSIII-8.3 Give examples to illustrate how sensitive data can be affected by malware and other attacks.
- CSIII-8.4 Discuss the concepts and justifications for using secure design techniques.
- CSIII-8.5 Discuss the laws surrounding intellectual property.
- CSIII-8.6 Perform a software risk evaluation to identify and mitigate risk.
- CSIII-8.7 Design accessible user interfaces that meet state and federal government disability standards.

## **Domain – Careers**

**Core Standard 9** Students will investigate various careers within the field of computer science.

### **Standards**

- CSIII-9.1 Identify computer science occupations and the roles and responsibilities of each.
- CSIII-9.2 Report job outlook, demand, and projected wages for computer science careers.
- CSIII-9.3 Explore the job opportunities that are available in computer science.
- CSIII-9.4 Investigate post-secondary training opportunities and industry certifications that are available.