Software Engineering Syllabus

Jeremy Ouellette, Instructor • ouellettej@hartfordschools.net • 802-295-8630 x2223

Course Description:

The Software Engineering Program at the HACTC will give students the foundational skills necessary to pursue a career in software development, dev ops, or cloud platform management. Students will be given a thorough introduction to Python, one of the world's most popular programming languages, and learn valuable software development skills along the way.

Also, with the HACTC's status as an Amazon Web Services (AWS) Academy, students will receive up-to-date training in the world's largest cloud services platform. Lastly, students undertake a substantial project where they will have the opportunity to learn a new technology or framework, taking an idea from requirements to implementation.

Classroom Expectations and Goals:

A successful student in the Software Engineering program will be able to:

- manage multiple deadlines on simultaneous activities
- focus on tasks for up to 30 minutes at a time, often in front of a computer screen
- maintain a clean, safe, and professional work environment
- communicate professionally with peers, community members, and supervisors
- work well in small groups
- give and accept constructive feedback
- solve challenging problems, often experiencing failure
- learn new skills that may be out of their comfort zone

Concurrent Enrollments Offered:

The program offers concurrent enrollment through River Valley Community College:

- CSCI-176R: Introduction to Python.

Industry Recognized Credentials:

- Amazon Web Services Certified Cloud Practitioner
- WorkKeys National Career Readiness Certificate

Common Standards Assessed in Every Program:

Communication: ESS02.01 Select and employ appropriate reading and communication strategies to learn and use technical concepts and vocabulary in practice.

Leadership and Teamwork: ESS07.03 Employ teamwork skills to achieve collective goals and use team members' talents effectively.

Technical Skill: ESS10.01 Employ information management techniques and strategies in the workplace to assist in decision-making.

Problem Solving/Critical Thinking: ESS03.01 Employ critical thinking skills independently and in teams to solve problems and make decisions (e.g., analyze, synthesize and evaluate).

Units of Study:

- Career Education & Readiness
- Introduction to Python
  - Input and output
  - Variables and data types
  - Conditionals
  - Loops
  - Functions
  - Object-oriented programming
  - Graphical user interfaces
  - IDE Usage
  - Source code control
  - Software test design
- Amazon Web Services Certified Cloud Practitioner
  - AWS service definitions, pricing models and global infrastructure
  - Security and compliance measures (IAM)
  - Virtual Private Cloud creation
  - Compute services (EC2, Lambda, Elastic Beanstalk)
  - Storage solutions (S3, EBS, EFS, S3 Glacier)
  - Database services (DS, DynamoDB, Redshift, Aurora)
  - AWS architectural principles
  - Load balancing and autoscaling
- Independent Project
- Digital Assessment Tool

Assignment Policy:

All assignments will be posted on the Canvas learning management system to provide all students with clear due dates and expectations.

Embedded Credit:

Students enrolled in this program receive 1 credit of Technology, 0.5 credits of Math, and 1.5 elective credits.

Grading Categories and Weights:

Problem Solving/Critical Thinking	30%
Technical Skill	30%
Communication	25%
Leadership & Teamwork	15%
Total	100%

Standards Unique to Information Technology:

Graduates of the Software Engineering program will be able to:

- Write computer programs to solve targeted problems.
- Use an integrated development environment to write code efficiently.
- Utilize version control systems to manage source code repositories.
- Utilize a debugger to diagnose errors in a program.
- Design test cases to verify aspects of program functionality.
- Follow best coding practices in areas such as commenting, variable names, and program structure.
- Demonstrate knowledge of the different stages of the software development life cycle.