



This course supports the assessment for Client-Server Application Development. The course covers 3 competencies and represents 3 competency units.

Introduction

Overview

We live in an interconnected world. Computer networks and their security are critical to most modern organizations of any size. This course introduces you to client/server application programming classes, structures, and concepts. The course covers networking and client/server, streams, threads, URLs, URIs, HTTP, and socket programming concepts.

Competencies

This course provides guidance to help you demonstrate the following 4 competencies:

- **Competency 4024.1.1: Networking Concepts**
The graduate explains basic concepts of networking, including the Internet.
- **Competency 4024.1.2: Streams and Threads**
The graduate explains the nature of streams and writes java code to implement and manipulate threads and streams.
- **Competency 4024.1.3: Internet Programming, URLs, URIs, and HTTP**
The graduate develops client/server applications that implement the Internet classes in Java, including proxies.
- **Competency 4024.1.4: Sockets**
The graduate implements client and server sockets, including secure sockets.

Course Instructor Assistance

As you prepare to successfully demonstrate competency in this subject, remember that course instructors stand ready to help you reach your educational goals. As subject matter experts, mentors enjoy and take pride in helping students become reflective learners, problem solvers, and critical thinkers. Course instructors are excited to hear from you and eager to work with you.

Successful students report that working with a course instructor is the key to their success. Course instructors are able to share tips on approaches, tools, and skills that can help you apply the content you're studying. They also provide guidance in assessment preparation strategies and troubleshoot areas of deficiency. Even if things don't work out on your first try, course instructors act as a support system to guide you through the revision process. You should expect to work with course instructors for the duration of your coursework, and you are encouraged to contact them as soon as you begin. Course instructors are fully committed to your success!

Preparing for Success

The information in this section is provided to detail the resources available for you to use as you complete this course.

Learning Resources



The learning resources listed in this section are required to complete the activities in this course. For many resources, WGU has provided automatic access through the course. However, you may need to manually enroll in or independently acquire other resources. Read the full instructions provided to ensure that you have access to all of your resources in a timely manner.

Automatically Enrolled Resources

You can access the learning resources listed in this section by clicking on the links provided throughout the course. You may be prompted to log in to the WGU student portal to access the resources.

uCertify

All of the course content will be found within the following uCertify course:

- [Client-Server Application Development](#)

The course is structured in such a way as to match your preassessment and objective assessment reports; each uCertify lesson corresponds to each assessment topic. That way, based on the results of your preassessment, you will know exactly which areas need the majority of your attention before taking the final exam.

Note: Once you have entered the uCertify course environment, it is not necessary to come back to this environment until you are ready to take your final assessment.

Complete the Preassessment

If you believe you have previous knowledge of some or all topics covered in this course, start by taking the preassessment before you begin and use its results to focus your studies.

- Complete the preassessment located in the Assessment tab.

Course instructors can help you develop a study plan based on your preassessment results.

Pacing Guide

The pacing guide suggests a weekly structure to pace your completion of learning activities. It is provided as a suggestion and does not represent a mandatory schedule. Follow the pacing guide carefully to complete the course in the suggested timeframe.

- [Pacing Guide: Client Server Application Development](#)

Note: This pacing guide does not replace the course. Please continue to refer to the course for a comprehensive list of the resources and activities.

Networking Concepts

To develop client/server network programs, it is essential to have an understanding of the

- basic network concepts, including network terms, protocol, standards, and technologies;
- technologies involved with communicating across the Internet;



- client/server model and the role of clients and servers in network communications; and
- basic operation of common routing protocols.

Network Concepts

This section includes an overview of basic networking terms and concepts. Also included are common networking standards and protocols, including the TCP model, TCP, IPv4, IPv6, DNS, and DHCP.

This topic addresses the following competency:

- **Competency 4024.1.1: Networking Concepts**

The graduate explains basic concepts of networking, including the Internet.

This topic highlights the following objectives:

- Describe basic networking terms and addresses.
- Identify the relationship between network layers and programming.
- Identify concepts associated with IP addresses and domain names.

Read and Complete: Chapter 1

As you complete the listed material below, pay attention to the following key points:

- The TCP/IP protocol is used by a plethora of devices to access the Internet.
- IPv6 was developed to solve the then imminent exhaustion of the IPv4 address block.
- IPv4 addresses were exhausted on February of 2011; every Internet address since then is assigned on IPv6.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 1 (“Basic Network Concepts”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

The Internet and the Client/Server Model

This section covers the structure of the Internet and many of the devices involved in communicating across networks and the Internet, including routers, proxy servers, and firewalls. Protocols and technologies involved in communicating across networks include Network Address Translation, filtering, FTP, SSH, and Telnet. The TCP and UDP protocols are covers, as connection-oriented and connectionless protocols. The client/server model is covered, including the role of both the client and the server in network communications. Common routing protocols covered include BGP, RIP, OSPF, and IGRP.

This topic addresses the following competency:

- **Competency 4024.1.1: Networking Concepts**

The graduate explains basic concepts of networking, including the Internet.



This topic highlights the following objectives:

- Explain the basic structure of the Internet, including the role of the devices used for communicating across the Internet.
- Compare the TCP and UDP protocols.
- Describe the client/server model in network communications.
- Identify communication routing protocols.

Read and Complete: Chapter 1

As you complete the listed material below, pay attention to the following key points:

- Pay attention to the differences between TCP and UDP protocols.
- Reflect on your everyday use of the Internet and try to identify the protocols you use more often.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 1 (“Basic Network Concepts”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Note: You may have completed this work in the previous activity. If so, you may wish to review and check your knowledge and understanding.

Streams and Threads

Streams and threads are important topics covered in network programming in order to program input and output operations in your programs.

The following are included within these topics:

- input, output, filter streams
- threads, synchronization, and scheduling

Input, Output, and Filter Streams

This section includes the purpose, classes, methods, and attributes of output, input, and filter streams. This section also includes the purpose, classes, methods, and attributes of reader and writer classes.

This topic addresses the following competency:

- **Competency 4024.1.2: Streams and Threads**
The graduate explains the nature of streams and writes java code to implement and manipulate threads and streams.

This topic highlights the following objectives:



- Describe the purpose of output, input, and filter streams from a programming perspective.
- Implement streams in Java.
- Describe the purpose and capabilities of readers and writers.
- Implement reader and writer classes in Java.

Read and Complete: Chapter 2

As you complete the listed material below, pay attention to the following key points:

- The Java I/O mentioned in this section is covered in-depth in the Software II course.
- Reader and writer classes are essential to the IPOS model, which most applications implement.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 2 (“Streams”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Threads, Synchronization, and Scheduling

This section covers the following:

- the purpose of threads, alternatives to threading, and the information that is being returned from threads
- factors that could prevent correct results from being retrieved, and the purpose of Futures, Callables, and Executors
- the purpose of synchronization and potential alternatives to synchronization, deadlocks and how to avoid them
- the purpose and need for thread scheduling and priorities, the starvation problem, and blocking, yielding, and sleeping

This topic addresses the following competency:

- **Competency 4024.1.2: Streams and Threads**
The graduate explains the nature of streams and writes java code to implement and manipulate threads and streams.

This topic highlights the following objectives:

- Describe the purpose of threads from a programming perspective.
- Identify the information that is being returned from threads.
- Explain the purpose of synchronization from a programming perspective.
- Describe deadlocks and how they impact programming.
- Describe the purpose and need for thread scheduling and priorities.
- Implement threads in Java.

Read and Complete: Chapter 3



As you complete the listed material below, pay attention to the following key points:

- Multithreading hardware is not new; it dates back to the 1950s. However, modern home operating systems which support multithreading have made multithreading processors the norm.
- The Java Virtual Machine makes multithreading possible in any platform that supports multithreading.
- Starvation is one of the problems a thread can encounter. Can you guess which resource a thread might be starved for?
- The Operating Systems for programmers course provides an in-depth inside to multithreading from the OS perspective.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 3 (“Threads”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Internet Programming, URLs, URIs, and HTTP

Internet programming, URLs, URIs, and HTTP classes, and associated methods are essential portions of Internet programming and retrieving information from websites. Programmers need to understand how to implement the

InetAddress and the NetworkInterface class.

Included in the InetAddress class are

- the methods and attributes of the InetAddress class,
- the getByByName() method,
- using the InetAddress class for obtaining DNS lookups,
- and the differences between the Inet4Address and Inet6Address classes.

Included in the NetworkInterface class are

- the classes, methods, and attributes of the NetworkInterface class, and
- the correct code to invoke a factory methods and getter methods.

This topic addresses the following competency:

This topic highlights

Read and Complete: Chapter 4



As you complete the listed material below, pay attention to the following key points:

- Java provides multiple methods for testing features of an Internet address. All their names start with *is* and all of them return Boolean values.
- The Internet Address class makes it possible to write client-server applications that transmit data across private and public networks.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 4 (“Internet Addresses”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

URLs, URIs, HTTP, and Connections

URLs, URIs, HTTP, and Connections are a major section within network programming. This section covers many topics including

- URL and URI methods,
- proxies,
- server-side programming,
- methods and classes available for accessing password protected sites,
- HTTP methods,
- establishing URL connections,
- and reading a cache response and making a cache request.

Also included are

- relative URLs,
- the `passwordAuthentication` class,
- the `JPasswordField` class, and
- HTTP response codes.

This topic addresses the following competency:

- **Competency 4024.1.3: Internet Programming, URLs, URIs, and HTTP**
The graduate develops client/server applications that implement the Internet classes in Java, including proxies.

This topic highlights the following objectives:

- Identify differences between URLs and URIs.
- Describe the URL class and its methods.
- Describe the URI class and its methods.
- Explain the purpose of proxies and the methods for using the proxy class.
- Explain the methods for communicating with server-side programs.
- Describe the methods and classes available for accessing password-protected sites.



- Explain the HTTP protocol, HTTP methods, and associated programming concepts.
- Explain the methods used for opening, reading data from, and configuring URL connections.
- Identify concepts associated with caches in Java.
- Implement web connections.

Read and Complete: Chapter 5

As you complete the listed material below, pay attention to the following key points:

- The terms URI and URL are often used interchangeably. In reality, they both point to network resources, but they have different structures. Get familiar with the structure of each of them.
- The HTTP and HTTPS protocol are most commonly used to access the WWW. Reflect on the protocol you are using to access this course. Can you think of a reason why that is the right protocol for this application?
- Cookies are used by many web applications. The way cookies are stored for web applications depends on the browser.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 5 (“URLs and URIs”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Read and Complete: Chapter 6

As you complete the listed material below, pay attention to the following key points:

- The terms URI and URL are often used interchangeably. In reality, they both point to network resources, but they have different structures. Get familiar with the structure of each of them.
- The HTTP and HTTPS protocol are most commonly used to access the WWW. Reflect on the protocol you are using to access this course. Can you think of a reason why that is the right protocol for this application?
- Cookies are used by many web applications. The way cookies are stored for web applications depends on the browser.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 6 (“HTTP”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Read and Complete: Chapter 7

As you complete the listed material below, pay attention to the following key points:

- The terms URI and URL are often used interchangeably. In reality, they both point to



network resources, but they have different structures. Get familiar with the structure of each of them.

- The HTTP and HTTPS protocol are most commonly used to access the WWW. Reflect on the protocol you are using to access this course. Can you think of a reason why that is the right protocol for this application?
- Cookies are used by many web applications. The way cookies are stored for web applications depends on the browser.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 7 (“URLConnections”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Sockets

Network programmers need to understand and program client sockets and server sockets. Additionally, with the increased attention given to security issues over the Internet, programmers need to know how to implement secure sockets in their programs.

Using Sockets for Clients and Servers

Describe

- the purpose of sockets,
- socket operations,
- methods and attributes of the socket class,
- the use of TelNet to connect to a server, and
- the methods and attributes used to construct and control client sockets and server sockets, including
 - writing a class for a multithreaded servers,
 - closing server sockets, and
 - server logs.

This topic addresses the following competency:

- **Competency 4024.1.4: Sockets**
The graduate implements client and server sockets, including secure sockets.

This topic highlights the following objectives:

- Describe the purpose of sockets from a programming perspective.
- Identify the socket operations.
- Identify the methods used to construct and control client sockets.
- Identify the methods used to construct and control server sockets.
- Implement client and server sockets in Java.

Read and Complete: Chapter 8

As you complete the listed material below, pay attention to the following key points:



- Sockets are used to communicate across a network.
- Utilizing sockets, a client application can read and write from a server and vice-versa.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 8 (“Sockets for Clients”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Read and Complete: Chapter 9

As you complete the listed material below, pay attention to the following key points:

- Sockets are used to communicate across a network.
- Utilizing sockets, a client application can read and write from a server and vice-versa.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 9 (“Sockets for Servers”).
- Complete all learning activities in the chapter.
- Quiz yourself to check your understanding.

Secure Sockets

This section covers programming secure communications in networking, encryption, the differences between synchronous and asynchronous encryption, and the differences between public and private keys. Also included are the methods used to construct and configure secure client and server sockets, the implementations of Cipher Suites and creating and configuring secure sessions between client and server.

This topic addresses the following competency:

- **Competency 4024.1.4: Sockets**
The graduate implements client and server sockets, including secure sockets.

This topic highlights the following objectives:

- Describe the purpose of secure communications in networking from a programming perspective.
- Describe the methods used to construct and configure secure client and server sockets.
- Implement secure client and server sockets.

Read and Complete: Chapter 10

As you complete the listed material below, pay attention to the following key points:

- Communication between secure sockets is encrypted.
- Can you think of a reason not to utilize secure sockets in every single application?
- Synchronous and Asynchronous encryption, referenced in this section, is covered in



detail in the Network and Security - Foundations course.

Complete the following in the [Client-Server Application Development](#) uCertify course:

- Read chapter 10 (“Secure Sockets”)
- Complete all learning activities in the chapter
- Quiz yourself to check your understanding

Final Steps

Congratulations on completing the activities in this course! This course has prepared you to complete the assessment associated with this course. If you have not already been directed to complete it, schedule and complete the assessment now. Remember that when you go to take your assessment, that you are required to have your whiteboard, markers, and eraser with you.

Complete the Preassessment

The preassessment can help you determine your level of preparation. It is highly recommended that you pass the preassessment before attempting the final exam.

- Complete the preassessment located in the Assessment tab.

Course instructors can help you develop a review plan based on your preassessment results.

First Attempt Checklist

One of the many things that makes WGU unique is its competency-based education model. If you know the material, all you have to do is prove it by passing the exam. If you can do this, you can accelerate the receipt of your degree.

To make sure you have the best chance possible to pass the exam on your first attempt, the following steps should be completed successfully before you take it:

1. Complete the reading for the assigned chapters in UCertify and take detailed notes.
2. Complete the activities within each chapter and ensure you have a solid understanding of the concepts presented in the activities.
3. Complete each end of chapter quiz. Go back and review the material for any quiz where you scored less than 100%. Take the quiz again, striving for 100% because you understand the concepts. Be careful to avoid accidentally memorizing the answers.
4. Go back through the course and review each section, ensuring you can provide a full explanation for any of the concepts outlined as key points. If you cannot provide a full explanation for one of those points, go back to that chapter and review the content. Reach out to the course instructor for assistance with anything that is unclear after your review.
5. Take the preassessment. View the Coaching Report to see your results and review each individual topic with a score less than 85%.

If you have completed the steps above and you feel comfortable with all of the concepts presented, you are most likely ready to attempt the exam.



If you fail your first attempt, you will be required to contact the course instructor to see what went wrong and how you can prepare to ensure a successful second attempt. After determining you are ready, your course instructor will approve your request once to make another exam attempt.