***Portfolio: Jose Alberto Salazar ICT - NMSU***

Class: CS 111 - Computer Science Principles.

### 1. **Introduction to Computational Thinking**

* **Problem Solving:** Breaking down complex problems into smaller, manageable parts.
* **Algorithms:** Understanding the step-by-step procedures for solving problems.
* **Abstraction:** Simplifying complex systems by focusing on the most important aspects.

### 2. **Programming Basics**

* **Programming Languages:** Introduction to a programming language
* **Control Structures:** Using loops, conditionals, and functions to control program flow.

### 3. **Data Representation**

* **Binary and Hexadecimal Systems:** Understanding how data is represented in computers.
* **Data Types:** Exploring different types of data such as integers, floats, strings, and booleans.

### 4. **Internet and Web Technologies**

* **How the Internet Works:** Basics of networking, IP addresses, and protocols like HTTP.

### 5. **Algorithms and Efficiency**

* **Algorithm Design:** Techniques for creating efficient algorithms.
* **Big-O Notation:** Understanding the efficiency and scalability of algorithms.

### 6. **Software Development**

* **Software Lifecycle:** Overview of the stages of software development from conception to deployment.

### 7. **Ethics and Impact of Computing**

* **Digital Citizenship:** Understanding the ethical implications of computing.
* **Privacy and Security:** Basics of data privacy and cybersecurity.
* **Social Impact:** Exploring how computing technologies affect society.

### Key Skills Developed:

* **Critical Thinking:** Analyzing problems and developing logical solutions.
* **Coding Proficiency:** Gaining hands-on experience with programming.
* **Collaboration:** Working with peers on projects and assignments.
* **Communication:** Presenting and explaining technical concepts clearly.

Class: CS 172 - Computer Science II (Java).

This course built on fundamental programming concepts, focusing on **object-oriented programming (OOP), data structures, and algorithm development** using **Java**.

1. **Fundamentals & Control Structures**
   1. **Loops and Branches** (if/else, for/while loops)
   2. **Fahrenheit to Celsius Conversion** (basic mathematical operations & user input handling)
2. **Object-Oriented Programming (OOP) Principles**
   1. **Creating Classes** (constructors, instance variables, methods)
   2. **Inheritance** (extending classes and code reuse)
   3. **Polymorphism** (method overriding and dynamic method dispatch)
3. **Data Structures & Game Development**
   1. **Tic-Tac-Toe Game** (likely using 2D arrays, loops, and conditionals)
   2. **Arrays & 2D Arrays** (handling game boards, lists of data)

This class would have been great preparation for **advanced Java programming, software development, and algorithm design**.

* Wrote **Java programs** that used loops, conditionals, and functions
* Built a **Tic-Tac-Toe game** using Java’s **object-oriented principles**
* Practiced **class creation, inheritance, and polymorphism** for modular code
* Worked with **arrays and 2D arrays** to store and manipulate data

This class has been great preparation for **advanced Java programming, software development, and algorithm design**.

Class: ICT 320 - Introduction to Internet Protocols.

### **1. Network Traffic Analysis & Packet Capturing**

**Wireshark Installation & Use**

* **Wireshark** is a popular network protocol analyzer used for **capturing and inspecting network traffic**.
* Helps in **troubleshooting, network security analysis, and learning how data flows** over a network.

**Packet Capturing**

* Capturing packets lets you see **source/destination addresses, protocols used, and data payloads** in real-time.
* Useful for **debugging network issues, detecting unauthorized access, and monitoring traffic patterns**.

### **2. Foundational Protocols & Network Services**

**RFC (Request for Comments)**

* **RFCs are official documents** that define internet standards and protocols (e.g., HTTP, TCP/IP).
* Created by **the Internet Engineering Task Force (IETF)** to ensure consistency in networking.

**NTP (Network Time Protocol)**

* Synchronizes **computer system clocks** across a network.
* Ensures that logs, timestamps, and transactions are accurate across devices.

**DHCP (Dynamic Host Configuration Protocol)**

* **Automatically assigns IP addresses** to devices in a network.
* Reduces manual configuration and prevents **IP conflicts**.

**ARP (Address Resolution Protocol)**

* Translates **IP addresses into MAC addresses** (used at the data link layer).
* Essential for devices to communicate within a **local network (LAN)**.

### **3. Domain Name System (DNS) & Network Management**

**DNS Zones**

* **DNS converts domain names (e.g., google.com) into IP addresses**.
* Zones are **segments of DNS** that help distribute name resolution across multiple servers:
  + **Primary (Master) Zone:** Stores the original DNS records.
  + **Secondary (Slave) Zone:** A read-only copy for redundancy.
  + **Reverse Lookup Zone:** Converts **IP addresses back into domain names**.

**SNMP (Simple Network Management Protocol)**

* Used for **monitoring and managing network devices** like routers, switches, and servers.
* Can **collect performance data, detect failures, and send alerts**.

### **4. File Transfer & Remote Access Protocols**

**FTP (File Transfer Protocol) & TFTP (Trivial File Transfer Protocol)**

* **FTP:** Transfers files between a client and a server, **requires authentication**.
* **TFTP:** A **simpler, less secure** version of FTP, used for **quick file transfers**, often in network booting.

**SFTP (Secure File Transfer Protocol) & SCP (Secure Copy Protocol)**

* **SFTP:** Secure version of FTP that **encrypts file transfers using SSH**.
* **SCP:** A command-line tool for **securely copying files over SSH**.

**RDP (Remote Desktop Protocol) & SSH (Secure Shell)**

* **RDP:** Allows remote **graphical desktop access** to a Windows machine.
* **SSH:** Secure command-line access to **remote servers**, commonly used in **Linux administration**.

### **5. Email Communication Protocols**

**SMTP (Simple Mail Transfer Protocol)**

* The protocol responsible for **sending emails between mail servers**.
* Works alongside **IMAP/POP3**, which retrieve emails for users.
* Uses **port 25 (unencrypted) or 465/587 (encrypted)** for secure email transmission.

Summary:

* **Installed and used Wireshark** for packet analysis.
* **Captured and analyzed network traffic** to understand how protocols work.
* **Configured and tested DHCP, ARP, and DNS zones** in a lab environment.
* **Set up and secured file transfers** with FTP, TFTP, SFTP, and SCP.
* **Used RDP and SSH** for remote access to different systems.

This class built a strong foundation in **networking, security, and IT troubleshooting**, essential for careers in **network administration, cybersecurity, and IT support**.

Class: ICT 339 - Digital Forensics.

This course provided a comprehensive introduction to the principles and practices of digital forensics, equipping me with the skills to investigate and analyze digital evidence for legal and investigative purposes. Key areas of study included:

* **Digital Evidence Handling:** Understanding the collection, preservation, and analysis of digital evidence, including data recovery and chain of custody protocols.
* **Forensic Tools and Techniques:** Hands-on experience with industry-standard tools like EnCase and FTK for imaging, hashing, and analyzing digital media.
* **File System and OS Forensics:** Examining file systems (e.g., NTFS, FAT32) and operating system artifacts to uncover critical evidence.
* **Network and Mobile Forensics:** Analyzing network traffic for suspicious activity and extracting data from mobile devices for investigations.
* **Cybercrime and Incident Response:** Investigating cybercrimes such as hacking and phishing, and developing incident response strategies.
* **Legal and Ethical Considerations:** Gaining knowledge of the legal frameworks governing digital forensics, including privacy laws and expert testimony requirements.

Through practical labs, case studies, and a capstone project, with this class we developed strong analytical, technical, and reporting skills, preparing me for roles in law enforcement, corporate security, legal consulting, and cybersecurity. This course emphasized attention to detail, critical thinking, and the ability to present findings clearly in legal and professional contexts.

Class: ICT 360 - Operating Systems.

This course provided a comprehensive understanding of operating systems (OS), focusing on their design, functionality, and management of hardware and software resources. Key topics and practical experiences included:

* **OS Fundamentals:** Explored the core functions of operating systems, including process management, memory management, file systems, and device management.
* **Process and Thread Management:** Studied process scheduling algorithms (e.g., Round Robin, Priority Scheduling), thread synchronization, and multitasking.
* **Memory Management:** Learned about memory allocation, paging, segmentation, and virtual memory to optimize system performance.
* **File Systems:** Gained hands-on experience with file organization, storage, and directory structures, including commands for managing files and directories in both GUI and CLI environments.
* **Network and OS Management:** Examined how operating systems handle network resources and interact with distributed systems.
* **User Interfaces:** Worked with both **Graphical User Interfaces (GUI)** like the **GNOME desktop** and **Command-Line Interfaces (CLI)** to perform system tasks, manage files, and execute commands.
* **Applications and System Interaction:** Explored how applications interact with the OS, including system calls, APIs, and resource allocation.
* **Security and Protection:** Studied OS-level security mechanisms, including user authentication, access control, and file permissions.

### **Hands-On Experience:**

* **File and Directory Commands:** Practiced using CLI commands (e.g., ls, cd, mkdir, rm) to navigate and manage files and directories in a Linux environment.
* **GNOME Desktop:** Utilized the GNOME desktop environment to perform system management tasks and understand the role of GUIs in modern operating systems.
* **System Monitoring and Troubleshooting:** Analyzed system performance, monitored processes, and resolved common OS issues using both CLI and GUI tools.

### **Key Skills Developed:**

* **System-Level Programming:** Wrote scripts and programs to interact with OS APIs and manage system resources.
* **Command-Line Proficiency:** Gained fluency in using CLI for system administration and file management.
* **Problem-Solving:** Debugged and optimized OS performance issues, including memory and process management.
* **Technical Knowledge:** Developed a deep understanding of OS architecture, file systems, and user interfaces.
* **Critical Thinking:** Analyzed trade-offs in OS design, such as performance vs. security and GUI vs. CLI efficiency.

This course provided a strong foundation in operating system principles and practical skills, preparing me for roles in system administration, software development, and IT support. The hands-on experience with both GUI and CLI environments, file management, and system troubleshooting has equipped me to work effectively in diverse computing environments.

### Class: ICT 362 - SOFTWARE TECH II

This course focused on advancing software development skills through hands-on programming in Python, emphasizing problem-solving, efficient coding practices, and the implementation of core programming concepts. Key topics and activities included:

* **Python Programming Fundamentals:**
  + **Control Structures:** Mastered **if statements**, **if-else constructs**, and **nested conditionals** to create decision-making logic in programs.
  + **Boolean Expressions:** Used logical operators (and, or, not) to evaluate conditions and control program flow.
  + **Loops:** Implemented **for loops** and **while loops** to iterate over data and automate repetitive tasks.
* **Data Structures in Python:**
  + **Lists:** Created, manipulated, and iterated over lists to store and process collections of data.
  + **Dictionaries:** Used key-value pairs to organize and retrieve data efficiently.
  + **Nested Structures:** Combined lists and dictionaries to model complex data relationships.
* **Problem-Solving and Algorithm Design:**
  + **Algorithm Development:** Designed algorithms to solve real-world problems using Python.
  + **Code Efficiency:** Learned to write clean, efficient, and readable code.
* **Software Development Practices:**
  + **Debugging and Testing:** Identified and fixed errors in code using debugging tools and wrote test cases to validate program functionality.
  + **Version Control:** Used tools like **Git** to manage code versions and collaborate with peers.
* **Hands-On Projects:**
  + **Programming Assignments:** Completed numerous Python programming assignments to reinforce concepts like loops, conditionals, and data structures.
  + **Capstone Project:** Designed and implemented a Python-based application, integrating all learned concepts to solve a practical problem.

### **Key Skills Developed:**

* **Python Proficiency:** Gained advanced skills in Python programming, including control structures, data manipulation, and algorithm design.
* **Problem-Solving:** Developed the ability to break down complex problems and implement solutions using Python.
* **Code Optimization:** Learned to write efficient and maintainable code.
* **Collaboration:** Worked with peers on coding projects, using version control and collaborative tools.
* **Debugging and Testing:** Improved skills in identifying and resolving errors in code.

### **Hands-On Experience:**

* **Python Programming:** Wrote and executed Python scripts to solve problems involving conditionals, loops, lists, and dictionaries.
* **Data Manipulation:** Used Python to process and analyze data, including nested data structures.
* **Project Development:** Applied Python skills to build a functional application as part of the course’s capstone project.

This course strengthened my programming foundation and prepared me for roles in software development, data analysis, and automation. The emphasis on Python programming, problem-solving, and practical projects has equipped me with the skills to tackle real-world challenges in the tech industry.

### Class: ICT 364 - WINDOWS ENTERPRISE ADMIN.

This course provided hands-on training in administering and managing Windows-based enterprise environments, with a focus on deploying, configuring, and maintaining critical server infrastructure. Key topics and practical experiences included:

#### **Core Topics Covered:**

1. **Active Directory (AD) Management:**
   1. Installed and configured **Active Directory Domain Services (AD DS)** to manage users, computers, and resources in a domain environment.
   2. Created and managed **AD objects**, including user accounts, groups, organizational units (OUs), and group policies.
   3. Implemented **Group Policy Objects (GPOs)** to enforce security settings, automate configurations, and manage user and computer behavior across the network.
2. **Networking Services:**
   1. Configured **DHCP (Dynamic Host Configuration Protocol)** for both **IPv4** and **IPv6** to automate IP address allocation.
   2. Set up and managed **DNS (Domain Name System)** to resolve hostnames and support domain operations.
   3. Installed and configured **IIS (Internet Information Services)** to host and manage websites, including setting up **FTP (File Transfer Protocol)** for file sharing.
   4. Implemented **SSL/TLS certificates** to secure web services and ensure encrypted communication.
3. **File and Storage Services:**
   1. Managed **file storage solutions**, including shared folders, NTFS permissions, and **Storage Spaces** for scalable and resilient storage.
   2. Configured **disk quotas** and implemented **file classification** to organize and control data storage.
4. **PowerShell and Automation:**
   1. Used **PowerShell** to automate administrative tasks, such as managing AD objects, configuring network settings, and generating reports.
   2. Wrote and executed scripts to streamline repetitive tasks and improve system management efficiency.
5. **Monitoring and Troubleshooting:**
   1. Utilized **Event Viewer** and **event logs** to monitor system activity, diagnose issues, and ensure system reliability.
   2. Performed **backup and recovery operations** to protect data and ensure business continuity.
6. **Core Server Administration:**
   1. Worked with **Windows Server Core**, a minimal server installation option, to deploy and manage servers with reduced overhead and enhanced security.
   2. Configured server roles and features to support enterprise operations.

#### **Hands-On Experience:**

* **Active Directory Lab:** Installed and configured AD DS, created OUs, and managed user and group accounts.
* **Networking Lab:** Set up DHCP, DNS, and IIS, including configuring websites and FTP services.
* **PowerShell Lab:** Wrote scripts to automate tasks like user creation, GPO deployment, and system monitoring.
* **File Storage Lab:** Configured shared folders, implemented NTFS permissions, and set up Storage Spaces.
* **Group Policy Lab:** Designed and deployed GPOs to enforce security policies and automate system configurations.

#### **Key Skills Developed:**

* **Active Directory Expertise:** Gained proficiency in managing AD objects, GPOs, and domain services.
* **Networking Knowledge:** Configured and maintained critical networking services like DHCP, DNS, and IIS.
* **PowerShell Scripting:** Automated administrative tasks and improved system management efficiency.
* **Security Implementation:** Secured web services with SSL/TLS certificates and enforced security policies through GPOs.
* **Problem-Solving:** Diagnosed and resolved issues using event logs and monitoring tools.

#### **Capstone Project:**

* Designed and implemented a comprehensive Windows-based enterprise solution, integrating AD, DHCP, DNS, IIS, file storage, and group policies to simulate a real-world administrative scenario.

This course equipped me with the technical skills and practical experience needed to manage and secure Windows enterprise environments. The hands-on labs and capstone project provided real-world experience, preparing me for roles in system administration, network management, and IT support.

### Class: ICT 377 Computer Networking 1.

This course provided a comprehensive introduction to computer networking, focusing on the design, implementation, and management of network infrastructures using both theoretical knowledge and hands-on practice.

#### **Core Topics Covered:**

1. **Networking Fundamentals:**
   1. **Network Types:** Explored LANs (Local Area Networks), WANs (Wide Area Networks), and MANs (Metropolitan Area Networks).
   2. **OSI and TCP/IP Models:** Studied the layers of the OSI and TCP/IP models and their roles in network communication.
   3. **IP Addressing and Subnetting:** Mastered **IPv4** and **IPv6** addressing, including **subnetting** techniques to design efficient and scalable networks.
2. **Network Devices and Configuration:**
   1. **Cisco Packet Tracer:** Used **Cisco Packet Tracer** to simulate and configure network topologies, including routers, switches, and end devices.
   2. **Command-Line Interface (CLI):** Gained proficiency in using the CLI to configure and manage Cisco devices, including routers and switches.
   3. **VLANs (Virtual Local Area Networks):** Configured and managed VLANs to segment network traffic and improve performance and security.
3. **Routing Protocols:**
   1. **Static Routing:** Configured static routes to manually define paths for network traffic.
   2. **Dynamic Routing Protocols:**
      1. **RIP (Routing Information Protocol):** Implemented RIP for basic dynamic routing.
      2. **EIGRP (Enhanced Interior Gateway Routing Protocol):** Configured EIGRP for efficient routing in medium to large networks.
      3. **OSPF (Open Shortest Path First):** Set up OSPF for advanced dynamic routing in complex networks.
4. **Network Services:**
   1. **DNS and DHCP:** Configured and managed DNS and DHCP services to automate IP address allocation and hostname resolution.
   2. **Network Address Translation (NAT):** Implemented NAT to allow private IP addresses to access public networks.
5. **Network Security and Troubleshooting:**
   1. **Access Control Lists (ACLs):** Configured ACLs to control traffic flow and enhance network security.
   2. **Troubleshooting Tools:** Used tools like **ping**, **traceroute**, and **Wireshark** to diagnose and resolve network issues.

#### **Hands-On Experience:**

* **Cisco Packet Tracer Labs:**
  + Designed and simulated network topologies, including routers, switches, and end devices.
  + Configured VLANs, routing protocols (RIP, EIGRP, OSPF), and network services (DNS, DHCP).
* **Subnetting Practice:**
  + Performed numerous subnetting exercises to design efficient IP addressing schemes for various network sizes.
* **CLI Configuration:**
  + Used the CLI to configure static routes, dynamic routing protocols, and VLANs on Cisco devices.
* **Troubleshooting Labs:**
  + Diagnosed and resolved network issues using tools like Wireshark and built-in Cisco IOS commands.

#### **Key Skills Developed:**

* **Network Design and Implementation:** Gained the ability to design, configure, and manage network infrastructures.
* **Subnetting Expertise:** Mastered IP addressing and subnetting for efficient network design.
* **Routing Protocol Knowledge:** Configured and optimized static and dynamic routing protocols (RIP, EIGRP, OSPF).
* **CLI Proficiency:** Became skilled in using the CLI to configure and troubleshoot Cisco devices.
* **Troubleshooting:** Learned to diagnose and resolve network issues using advanced tools and techniques.
* **Security Awareness:** Implemented ACLs and other security measures to protect network traffic.

#### **Capstone Project:**

* Designed and implemented a multi-site network using Cisco Packet Tracer, integrating VLANs, static and dynamic routing (RIP, EIGRP, OSPF), and network services (DNS, DHCP).
* Tested and troubleshooted the network to ensure optimal performance, scalability, and security.

This course provided a strong foundation in computer networking, preparing me for roles in network administration, IT support, and cybersecurity. The hands-on experience with **Cisco Packet Tracer**, **CLI**, and routing protocols has equipped me with the skills to design, configure, and manage modern network environments effectively.

### Class: ICT 450 - INTRO TO HACKING AND PENETRATION TESTING.

This course provided a hands-on introduction to ethical hacking and penetration testing, focusing on identifying, exploiting, and mitigating vulnerabilities in systems and networks. Key topics and practical experiences included:

#### **Core Topics Covered:**

1. **Ethical Hacking Fundamentals:**
   1. **Introduction to Ethical Hacking:** Learned the role of ethical hackers in securing systems and the importance of penetration testing in cybersecurity.
   2. **Legal and Ethical Considerations:** Explored the legal frameworks and ethical guidelines governing penetration testing and hacking activities.
2. **Hacking Environment Setup:**
   1. **Kali Linux VM:** Set up and configured **Kali Linux**, a penetration testing platform, as the primary tool for ethical hacking.
   2. **Metasploitable and DVWA:** Used **Metasploitable** (a vulnerable virtual machine) and **DVWA (Damn Vulnerable Web Application)** to practice exploitation techniques in a controlled environment.
3. **Penetration Testing Methodologies:**
   1. **Pre-Engagement Phase:** Created **Engagement Drafts** and **Engagement Scoping Documents** to define the scope, objectives, and rules of engagement for penetration tests.
   2. **Reconnaissance:** Used tools like **Nmap**, **Whois**, and **Recon-ng** to gather information about target systems and networks.
   3. **Scanning and Enumeration:** Conducted network scans and enumerated services, users, and shares using tools like **Nmap** and **Netcat**.
4. **Vulnerability Assessment and Exploitation:**
   1. **Vulnerability Scanning:** Used **Nessus** to identify vulnerabilities in systems and applications.
   2. **Exploitation with Metasploit:** Conducted **Metasploit demos** to exploit vulnerabilities, including gaining access, privilege escalation, and maintaining persistence.
   3. **Documentation:** Created **Metasploit documentation** to record exploitation steps, findings, and remediation recommendations.
5. **Web Application Attacks:**
   1. **DVWA Lab:** Exploited vulnerabilities in **DVWA**, such as **SQL injection**, **cross-site scripting (XSS)**, and **file inclusion vulnerabilities**.
   2. **Burp Suite:** Used Burp Suite to analyze and manipulate web traffic for vulnerability discovery.
6. **Social Engineering and Post-Exploitation:**
   1. **Social Engineering Toolkit (SET):** Simulated social engineering attacks using **SET** to manipulate individuals into divulging sensitive information.
   2. **Armitage:** Used **Armitage**, a graphical interface for Metasploit, to visualize and execute attacks.
   3. **Post-Exploitation Techniques:** Learned to maintain access, exfiltrate data, and cover tracks on compromised systems.

#### **Hands-On Experience:**

* **Kali Linux VM:** Set up and used Kali Linux as the primary platform for penetration testing.
* **Metasploitable and DVWA Labs:** Exploited vulnerabilities in Metasploitable and DVWA to practice real-world attack scenarios.
* **Reconnaissance Tools Lab:** Used tools like Nmap, Netcat, and Recon-ng for information gathering and scanning.
* **Nessus Scanning:** Conducted vulnerability scans and analyzed results to identify exploitable weaknesses.
* **Metasploit and Armitage:** Exploited vulnerabilities and visualized attacks using Metasploit and Armitage.
* **Social Engineering Lab:** Simulated phishing and other social engineering attacks using SET.

#### **Key Skills Developed:**

* **Penetration Testing Methodologies:** Gained experience in planning, scoping, and executing penetration tests.
* **Tool Proficiency:** Became skilled in using tools like **Kali Linux**, **Metasploit**, **Nessus**, **Nmap**, **Netcat**, **SET**, and **Armitage**.
* **Vulnerability Exploitation:** Learned to identify, exploit, and document vulnerabilities in systems and applications.
* **Social Engineering:** Understood the techniques and tools used in social engineering attacks.
* **Reporting and Documentation:** Developed the ability to create detailed reports, including **Engagement Scoping Documents** and **Metasploit documentation**.

#### **Capstone Project:**

* Conducted a full penetration test on a simulated environment, including reconnaissance, vulnerability scanning, exploitation, and post-exploitation activities.
* Prepared a comprehensive report documenting findings, exploitation steps, and recommendations for remediation.

This course provided a strong foundation in ethical hacking and penetration testing, preparing me for roles in cybersecurity, such as penetration tester, security analyst, or ethical hacker. The hands-on experience with tools like **Kali Linux**, **Metasploit**, **Nessus**, and **DVWA** has equipped me to identify and address security vulnerabilities effectively.

### Class: ICT 460 - MULTIMEDIA TOOLS AND SUPPORT

This course provided a comprehensive introduction to web development, focusing on designing, building, and deploying modern web applications. Key topics and hands-on activities included:

#### **Core Topics Covered:**

1. **Front-End Development:**
   1. **HTML5:** Learned to structure web content using semantic HTML5 elements.
   2. **CSS3:** Styled web pages using CSS3, including layouts, animations, and responsive design techniques.
   3. **JavaScript:** Added interactivity to web pages using JavaScript, including DOM manipulation and event handling.
2. **Back-End Development:**
   1. **PHP:** Used PHP to handle server-side logic and dynamically generate web content.
   2. **MySQL:** Integrated MySQL databases to store and retrieve data for web applications.
   3. **APIs:** Designed and consumed **RESTful APIs** to enable communication between front-end and back-end systems.
3. **Multimedia Tools:**
   1. **Audacity:** Used Audacity for audio editing and integrating sound into web applications.
   2. **Gimp:** Created and edited images using Gimp for use in web design.
   3. **Blender:** Designed 3D models and animations using Blender to enhance web applications.
4. **Web Application Deployment:**
   1. **Version Control:** Used **Git** and **GitHub** for version control and collaboration.
   2. **Hosting and Deployment:** Deployed web applications to platforms like **Heroku**, **Netlify**, or **AWS**.
   3. **Performance Optimization:** Learned techniques to optimize web applications for speed and scalability.
5. **Web Security:**
   1. **Common Vulnerabilities:** Studied vulnerabilities like **SQL injection**, **cross-site scripting (XSS)**, and **cross-site request forgery (CSRF)**.
   2. **Secure Coding Practices:** Implemented security best practices to protect web applications from attacks.

#### **Hands-On Experience:**

* **Front-End Projects:**
  + Built responsive and interactive web pages using HTML5, CSS3, and JavaScript.
  + Created dynamic user interfaces and integrated multimedia content.
* **Back-End Projects:**
  + Developed server-side logic using PHP and integrated MySQL databases.
  + Designed and consumed RESTful APIs to enable communication between front-end and back-end systems.
* **Multimedia Projects:**
  + Edited audio files using Audacity for use in web applications.
  + Created and edited images using Gimp for web design.
  + Designed 3D models and animations using Blender to enhance web applications.
* **Deployment Projects:**
  + Deployed web applications to cloud platforms like Heroku or Netlify.
  + Used Git for version control and collaboration.

#### **Key Skills Developed:**

* **Front-End Development:** Gained proficiency in HTML5, CSS3, and JavaScript for building responsive and interactive web pages.
* **Back-End Development:** Learned to build server-side logic using PHP, integrate MySQL databases, and design RESTful APIs.
* **Multimedia Editing:** Developed skills in using Audacity, Gimp, and Blender to create and edit multimedia content for web applications.
* **Web Security:** Understood common vulnerabilities and implemented secure coding practices.
* **Problem-Solving:** Enhanced critical thinking and problem-solving skills in web development scenarios.

#### **Capstone Project:**

* Designed and developed a full-stack web application, integrating front-end and back-end technologies.
* Incorporated multimedia content created using Audacity, Gimp, and Blender.
* Deployed the application to a cloud platform and implemented user authentication and authorization.
* Prepared documentation and presented the project to demonstrate functionality and design choices.

This course provided a strong foundation in web development, preparing me for roles in front-end, back-end, or full-stack development. The hands-on experience with modern web technologies, multimedia tools, and frameworks has equipped me to design, build, and deploy dynamic and secure web applications.

### Class: 462 Linux System Administration

This course provided comprehensive training in administering and managing Linux-based systems, focusing on the tools, techniques, and best practices for maintaining secure, efficient, and scalable server environments. Key topics and hands-on activities included:

#### **Core Topics Covered:**

1. **Linux Fundamentals:**
   1. **Linux Architecture:** Explored the architecture of Linux systems, including the kernel, shell, and file system hierarchy.
   2. **Command-Line Interface (CLI):** Mastered essential CLI commands for file management, process control, and system monitoring.
2. **System Installation and Configuration:**
   1. **Linux Distributions:** Installed and configured popular Linux distributions such as **Ubuntu**, **CentOS**, and **Fedora**.
   2. **Package Management:** Used package managers like **APT** (Advanced Package Tool) and **YUM** (Yellowdog Updater, Modified) to install, update, and remove software.
3. **User and Permission Management:**
   1. **User Accounts:** Created and managed user accounts, groups, and permissions.
   2. **File Permissions:** Configured file and directory permissions using **chmod**, **chown**, and **chgrp**.
   3. **Sudo and Root Access:** Managed superuser access and privileges.
4. **Networking and Services:**
   1. **Network Configuration:** Configured network interfaces, IP addresses, and routing using tools like **ifconfig**, **ip**, and **nmcli**.
   2. **Network Services:** Set up and managed services like **SSH** (Secure Shell), **Apache**, **Nginx**, and **FTP**.
   3. **Firewall Configuration:** Implemented firewall rules using **iptables** and **firewalld**.
5. **System Monitoring and Maintenance:**
   1. **Process Management:** Monitored and managed system processes using commands like **ps**, **top**, and **htop**.
   2. **Log Management:** Analyzed system logs using **journalctl** and **syslog** for troubleshooting and auditing.
   3. **Backup and Recovery:** Implemented backup strategies and disaster recovery plans using tools like **rsync** and **tar**.
6. **Scripting and Automation:**
   1. **Shell Scripting:** Wrote and executed **Bash scripts** to automate routine administrative tasks.
   2. **Cron Jobs:** Scheduled tasks using **cron** and **anacron**.
7. **Security and Hardening:**
   1. **Security Best Practices:** Implemented security measures such as **SELinux** (Security-Enhanced Linux) and **AppArmor**.
   2. **Vulnerability Scanning:** Used tools like **Lynis** to scan for vulnerabilities and harden the system.

#### **Hands-On Experience:**

* **Lab Exercises:**
  + Installed and configured Linux distributions in a lab environment.
  + Managed user accounts, permissions, and file systems.
  + Configured network interfaces and services like SSH and Apache.
  + Wrote and executed Bash scripts to automate tasks.
* **System Monitoring and Troubleshooting:**
  + Monitored system performance and analyzed logs to diagnose and resolve issues.
  + Implemented backup and recovery strategies.
* **Security Labs:**
  + Configured firewalls and implemented security measures to harden Linux systems.
  + Conducted vulnerability scans and applied patches.

#### **Key Skills Developed:**

* **Linux System Administration:** Gained proficiency in installing, configuring, and managing Linux-based systems.
* **Networking and Services:** Learned to configure and manage network services and firewalls.
* **Scripting and Automation:** Developed skills in writing Bash scripts and automating administrative tasks.
* **Security and Hardening:** Implemented security best practices to protect Linux systems.
* **Problem-Solving:** Enhanced critical thinking and problem-solving skills in system administration scenarios.

#### **Capstone Project:**

* Designed and implemented a secure and efficient Linux server environment, including user management, network configuration, and service setup.
* Conducted a security audit and implemented hardening measures to protect the system.
* Prepared documentation and presented the project to demonstrate functionality and security measures.

This course provided a strong foundation in Linux system administration, preparing me for roles in server administration, DevOps, and cybersecurity. The hands-on experience with Linux distributions, networking, scripting, and security has equipped me to manage and secure Linux-based systems effectively.

### Class: 463 - ENTERPRISE NETWORK ADMINISTRATION

This course provided advanced training in managing and automating enterprise network environments, focusing on the tools and techniques used to deploy, configure, and maintain scalable and efficient network infrastructures. Key topics and hands-on activities included:

#### **Core Topics Covered:**

1. **Network Automation with Ansible:**
   1. **Ansible Basics:** Learned to write and execute Ansible playbooks for automating network configuration and management tasks.
   2. **Error Handling in Ansible:** Troubleshot and resolved errors in Ansible playbooks to ensure reliable automation.
   3. **SSH Data Streams:** Used SSH for secure communication and data transfer in automated workflows.
2. **Containerization and Orchestration:**
   1. **Docker:** Explored containerization using Docker, including creating and managing Docker containers.
   2. **Docker Swarm and Stacks:** Configured and managed Docker Swarm for container orchestration and deployed multi-container applications using Docker stacks.
   3. **Gitea Container:** Set up and managed Gitea, a self-hosted Git service, within a Docker container.
3. **Version Control and Collaboration:**
   1. **GitHub:** Used GitHub for version control, collaboration, and managing code repositories.
   2. **Gitea:** Explored Gitea as an alternative to GitHub for self-hosted version control.
4. **Virtualization and Environment Management:**
   1. **VirtualBox:** Used VirtualBox to create and manage virtual machines for testing and development.
   2. **Vagrant:** Automated the creation and configuration of virtualized environments using Vagrant.
5. **Scripting and Automation:**
   1. **PHP Scripting:** Wrote PHP scripts for server-side tasks and automation.
   2. **Bash Scripting:** Developed Bash scripts to automate routine administrative tasks.
6. **Network Configuration and Management:**
   1. **SSH Configuration:** Configured and managed SSH for secure remote access and data transfer.
   2. **Network Services:** Set up and managed network services like web servers and databases.

#### **Hands-On Experience:**

* **Ansible Labs:**
  + Wrote and executed Ansible playbooks to automate network configuration tasks.
  + Troubleshot and resolved errors in Ansible playbooks.
* **Docker Labs:**
  + Created and managed Docker containers for various applications.
  + Configured Docker Swarm and deployed multi-container applications using Docker stacks.
  + Set up and managed Gitea within a Docker container.
* **Virtualization Labs:**
  + Used VirtualBox to create and manage virtual machines.
  + Automated environment setup using Vagrant.
* **Version Control Labs:**
  + Used GitHub and Gitea for version control and collaboration.
* **Scripting Labs:**
  + Wrote PHP and Bash scripts to automate tasks and manage server configurations.

#### **Key Skills Developed:**

* **Network Automation:** Gained proficiency in using Ansible for automating network configuration and management tasks.
* **Containerization:** Learned to create and manage Docker containers and orchestrate them using Docker Swarm.
* **Version Control:** Developed skills in using GitHub and Gitea for version control and collaboration.
* **Virtualization:** Became skilled in using VirtualBox and Vagrant for creating and managing virtualized environments.
* **Scripting and Automation:** Wrote PHP and Bash scripts to automate routine tasks and manage server configurations.
* **Problem-Solving:** Enhanced critical thinking and problem-solving skills in network administration scenarios.

#### **Capstone Project:**

* Designed and implemented an automated enterprise network environment using Ansible, Docker, and Vagrant.
* Set up and managed a self-hosted Git service using Gitea within a Docker container.
* Prepared documentation and presented the project to demonstrate functionality and automation workflows.

This course provided a strong foundation in enterprise network administration, preparing me for roles in network automation, DevOps, and system administration. The hands-on experience with tools like Ansible, Docker, Vagrant, and GitHub has equipped me to manage and automate enterprise network environments effectively.

### Class: 477 - COMPUTER NETWORKING II

This advanced course focused on designing, configuring, and managing complex network infrastructures, with an emphasis on routing protocols, VLANs, trunking, and network traffic analysis. Key topics and hands-on activities included:

#### **Core Topics Covered:**

1. **VLAN Configuration and Trunking:**
   1. **Static VLANs:** Configured static VLANs to segment network traffic and improve performance and security.
   2. **Trunking:** Implemented trunking protocols like **802.1Q** to allow multiple VLANs to communicate across switches.
2. **Routing Protocols:**
   1. **Static Routes:** Configured static routes to manually define paths for network traffic.
   2. **RIPv2 (Routing Information Protocol version 2):** Implemented RIPv2 for dynamic routing in small to medium-sized networks.
   3. **EIGRP (Enhanced Interior Gateway Routing Protocol):** Configured EIGRP for efficient and scalable routing in enterprise networks.
   4. **Route Redistribution:** Learned to redistribute routes between different routing protocols (e.g., RIPv2 and EIGRP) to enable communication across diverse network environments.
3. **Network Traffic Analysis:**
   1. **Wireshark:** Used Wireshark to capture, analyze, and interpret network traffic, identifying protocols, troubleshooting issues, and detecting anomalies.
   2. **Traffic Optimization:** Analyzed network traffic patterns to optimize performance and identify bottlenecks.
4. **Network Design and Troubleshooting:**
   1. **Network Topologies:** Designed and implemented complex network topologies, integrating VLANs, routing protocols, and trunking.
   2. **Troubleshooting Tools:** Used tools like **ping**, **traceroute**, and **Wireshark** to diagnose and resolve network issues.

#### **Hands-On Experience:**

* **VLAN and Trunking Labs:**
  + Configured static VLANs and implemented trunking using 802.1Q on Cisco switches.
  + Tested VLAN communication across multiple switches.
* **Routing Protocol Labs:**
  + Configured static routes and dynamic routing protocols (RIPv2 and EIGRP) on Cisco routers.
  + Implemented route redistribution between RIPv2 and EIGRP.
* **Wireshark Labs:**
  + Captured and analyzed network traffic using Wireshark to identify protocols, troubleshoot issues, and detect anomalies.
* **Network Design Projects:**
  + Designed and implemented a multi-site network, integrating VLANs, routing protocols, and trunking.
  + Tested and optimized network performance.

#### **Key Skills Developed:**

* **VLAN and Trunking Expertise:** Gained proficiency in configuring and managing VLANs and trunking protocols.
* **Routing Protocol Knowledge:** Learned to configure and optimize static routes, RIPv2, and EIGRP.
* **Route Redistribution:** Developed the ability to redistribute routes between different routing protocols.
* **Network Traffic Analysis:** Became skilled in using Wireshark to analyze and troubleshoot network traffic.
* **Problem-Solving:** Enhanced critical thinking and problem-solving skills in complex network scenarios.

#### **Capstone Project:**

* Designed and implemented a multi-site network, integrating VLANs, trunking, and routing protocols (static routes, RIPv2, and EIGRP).
* Conducted network traffic analysis using Wireshark to optimize performance and troubleshoot issues.
* Prepared documentation and presented the project to demonstrate functionality and design choices.

This course provided a strong foundation in advanced networking concepts, preparing me for roles in network administration, network engineering, and cybersecurity. The hands-on experience with VLANs, routing protocols, and Wireshark has equipped me to design, configure, and troubleshoot complex network infrastructures effectively.