

ITM 300 SYLLABUS

ITM 300 Communications in the Workplace

Hours: 3 credit hours / 45 contact hours

Instructor: Dan Tomal

Textbook, title, author, and year: *Workplace Communications, 6th ed.* Searles, George, 2013

Specific course information

- a. **Catalog description:** Review, analyze and practice verbal and written communication formats found in the workplace. Emphasis on developing skills in technical writing and oral presentations using electronic and traditional media. Credit not granted for both ITM 300 and COM 421. INTM 301 may be substituted for this course.
- b. **Prerequisites:** None

Specific goals for the course

- a. **Course Outcomes:**
 - Develop skills in writing quality technical proposals.
 - Develop skills in conducting technical presentations.
 - Understand the communications styles, ethics, and resumes writing.
 - Develop skills in effective written memos and letters.
 - Understand problem solving skills in effective group communications.
- c. **Course student outcomes:**
Upon completion of this course the student should be able to do the following:
 - Demonstrate proficiency in verbal communications by giving a technical presentation on a proposal.
 - Demonstrate proficiency in written communications by completing a comprehensive technical proposal.
 - Apply the principles of written communication by writing letters, resume, and reports.

Topics to be covered

- a. Introduction. Principles of Communications
- b. Communication Styles and Strategies
- c. Readability and Style
- d. Editing, Design, and Formats
- e. Technical Reports and Research Proposals
- f. Writing Letters and Resumes, Interviewing
- g. Group Communications and Efficiency
- h. Data Bases and library searches
- i. Conducting Technical Presentations
- j. Interpersonal, Verbal, and Nonverbal Communications
- k. Review of giving presentations and proposals
- l. Proposal Presentations and Final Proposal

ITM 301 SYLLABUS

ITM 301 Introduction to Contemporary Operating Systems and Hardware I

Hours: 3 credit hours / 60 contact hours; 30 hours lecture, 30 hours lab

Instructor: Vasilios "Billy" Pappademetriou

Textbook, title, author, and year:

The Official CompTIA A+ Core 1 Student Guide, Pam Taylor and James Pengelly, 2019

Specific course information

- a. **Catalog description:** Students study the basics of computer architecture and learn to use a contemporary operating system. Hardware requirements, hardware components, software compatibility, and system installation topics are covered along with post-installation, storage, security and system diagnosis, and repair. Topics also include discussion of current and future technology industry trends.
- b. **Prerequisites:** None
- c. **Required.**

Specific goals for the course

- a. **Program Educational Objectives**
 1. Problem solve and create innovative answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.
- b. **Course Outcomes:**
ITM301 is a foundation course in the basics of PC functioning from a hardware level to the upper level operating system. The intent is to serve as a basis for practical studies in other topics in IT. Upon completion, a student should be able to understand how a PC functions be able to troubleshoot and repair a PC and understand its workings in a networked environment from a hardware level to OS level. As well, as be comfortable understanding concepts of Linux, virtualization, servers, and tools for managing IT.
- c. **Course student outcomes:**
Upon completion of this course the student should be able to do the following
 - Recall and discuss the history of modern computing and the Internet
 - Describe fundamental concepts of electricity
 - Explain the operation and employment of power supplies
 - Recall and describe the components of a computer
 - Explain the functioning of processors
 - Describe the purpose and operation of motherboards, buses, architecture, and memory

- Explain the role and operation of storage, monitors, and other peripherals
- Assemble a computer from a set of components
- Demonstrate methods for troubleshooting hardware
- Describe fundamental concepts of networking including physical media, devices, protocols, standards
- Explain and demonstrate the installation and basic configuration of an operating system
- Explain and use operating system utilities
- Describe cloud computing concepts
- Recall and describe laws, regulations, and compliance frameworks that affect IT professionals
- Discuss current events in computing, especially related to security.
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline **(ABET Computing Criterion 3.2)**
- Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles **(ABET Computing Criterion 3.4)**

1. Topics to be covered
 - a. Introduction. IIT Banner, Class basics
 - b. Lab: Virtual Machine Lab
 - c. Installing and Configuring PC Components
 - d. Installing, Configuring, and Troubleshooting Display and Multimedia Devices
 - e. Installing, Configuring, and Troubleshooting Storage Devices
 - f. Installing, Configuring, and Troubleshooting Internal System Components
 - g. Midterm Presentations
 - h. Implementing Client Virtualization and Cloud Computing
 - i. Desktop Teardown & Reassemble (Full Lab Planned)
 - j. Network Infrastructure Concepts
 - k. Supporting and Troubleshooting Laptops
 - l. Supporting and Troubleshooting Mobile Devices
 - m. Installing, Configuring, and Troubleshooting Print Devices
 - n. Configuring and Troubleshooting Networks

ITM 303 SYLLABUS

ITM 303 Introduction to Contemporary Operating Systems and Hardware II

Hours: 3 credit hours / 60 contact hours; 30 hours lecture, 30 hours lab

Instructor: Vasilios "Billy" Pappademetriou

Textbook, title, author, and year:

The Official CompTIA A+ Core 2 Student Guide (Exam 220-1102), Pam Taylor and James Pengelly, 2019

Specific course information

- a. **Catalog description:** Introduces features of an advanced operating system, including basic commands, file and directory manipulation, security, and suitability for server applications. Popular and business-focused desktop and mobile device operating systems will be examined, as well as enterprise and open-source server implementations.
- b. **Prerequisites:** ITM 301
- c. **Optional.**

Specific goals for the course

- c. **Program Educational Objectives**
 1. Problem solve and create innovative answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.
- d. **Course Outcomes:**
ITM302 is a foundation course in the basics of computer, device, and server operating system. This serves as a basis for practical studies in other topics in IT. Upon completion, a student should be able to understand, use, and manage industry-standard operating systems.
- d. **Course student outcomes:**
Upon completion of this course the student should be able to do the following
 - Operating Systems
 - Compare and contrast common operating systems and their purposes.
 - Compare and contrast features of Microsoft Windows versions.
 - Summarize general OS installation considerations and upgrade methods.
 - Use appropriate Microsoft command line & GUI tools, features and Control Panel utilities.
 - Install and configure applications.
 - Configure Microsoft Windows networking on a client/desktop system.
 - Use features and tools of Mac OS and Linux client/desktop operating systems.
 - Security
 - Summarize the importance of physical security measures.
 - Explain logical security concepts.

- Compare and contrast wireless security protocols and authentication methods.
- Detect, remove, and prevent malware using appropriate tools and methods.
- Compare and contrast social engineering, threats, and vulnerabilities.
- Compare and contrast the differences in basic Microsoft Windows OS security settings
- Implement security best practices to secure a workstation, mobile device, SOHO wired or wireless devices.
- Implement appropriate data destruction and disposal methods.
- Software Troubleshooting
 - Troubleshoot:
 - Microsoft Windows OS.
 - and resolve PC Security issues.
 - Mobile OS and Application issues.
 - Mobile OS and Application Security Issues.
 - Use best practices for malware removal.
- Operational Procedures
 - Compare and contrast best practices associated with types of documentation.
 - Implement:
 - Change management best practices.
 - Basic disaster prevention and recovery methods.
 - Explain:
 - Common safety procedures/practices.
 - Environmental impacts and appropriate controls.
 - Describe processes for addressing prohibited content/activity, and privacy, licensing, and policy concepts.
 - Use proper communication techniques and professionalism.
 - Identify the basics of scripting.
 - Use remote access technologies.
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline **(ABET Computing Criterion 3.2)**

Topics to be covered

- a. Introduction. IIT Banner, Class basics Book and Lab setup
- b. Lab - Virtual Machine Lab
- c. Support Operating Systems
- d. Install, Configure, and Maintain an OS
- e. Maintain and Troubleshoot MS Windows
- f. Configure and Troubleshoot Networks
- g. Open Topic/Midterm
- h. Manage Users, Workstations & Shared Resources
- i. Security Concepts
- j. Secure Workstation and Data
- k. Troubleshoot Workstation Security Issues
- l. Support and Troubleshoot Mobile Devices
- m. Implement Operational Procedures

ITM 311 SYLLABUS

ITM 311 Introduction to Software Development

Hours: 3 credit hours / 60 contact hours; 30 hours lecture, 30 hours lab

Instructor: Katherine Papademas

Text book, title, author, and year: *Java Programming*, by Joyce Farrell, 9th Edition

Specific course information

- a. Catalog description: A broad introduction to object-oriented programming and the related knowledge necessary to program in a contemporary programming language. This would include coverage of an Application Development Kit, a standard integrated Development environment, and the use of GUI components.
- b. Prerequisites: None
- c. Required.

Specific goals for the course

- a. Program Educational Objectives:
 2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing of relevant policy and best practices as appropriate.
- b. **Course Outcomes:**
 - Develop the ability to write and resolve programming problems using Java Language
 - Build Java Applications and Java Applets
 - Identify Java standard libraries and classes
 - Write, compile, execute and troubleshoot Java programming
 - Understand and utilize Java Graphical User Interface in the program writing
 - Understand Java programming syntax, control structures and Java programming concepts
 - Understand, locate and Use Help Resources
 - Build the confidence in “speaking” and writing programs in Java
 - Be acquainted with the various IDEs used for Java Application Programming
- e. **Course student outcomes:**
Upon completion of this course the student should be able to do the following:
 - Recall and describe software application and development theory and concepts
 - Write, compile, execute, troubleshoot, and resolve problems using the Java programming language
 - Build Java Applications and Java Applets
 - Develop with and use Object-Oriented Programming and Methodology in program development

- Identify and use important Java standard libraries and classes
- Locate and use Help Resources
- Develop and write Object Oriented Java Programs
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions **(ABET Computing Criterion 3.1)**
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program’s discipline **(ABET Computing Criterion 3.2)**
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems **(ABET IT Criterion 3.6)**

Topics to be covered

- a. Using data. Java I/O, Scanner class, data types, variable naming conventions & declarations
- b. Methods
- c. More Object concepts.
- d. Overloaded methods, Constructors. Blocks & Scopes, static fields.
- e. Decision making. Boolean Logic.
- f. Looping techniques. Finite loops. For, while, do-while loops.
- g. Loops continued. Loop performance. 2 Nested loops. Pre/post fix incrementing.
- h. Characters, strings, String builder class
- i. Arrays. Subscripting, parallel arrays.
- j. Advance Arrays. Sorts, ragged arrays, multidimensional arrays.
- k. Introduction to Inheritance
- l. Advance Inheritance. Using packages.
- m. Advanced inheritance continued.
- n. Exception Handling
- o. File I/O. Sequential and Random file 5 processing.
- p. Introduction to Swing Components- Java GUI. Event driven programming
- q. Advanced GUI topics, JAVAFX Intro
- r. Graphics
- s. Applets, Images and Sound
- t. JAVA, JSON, JQUERY, JAVASCRIPT, AngularJS

ITM 312 SYLLABUS

ITM 312 Introduction to Systems Software Programming (C++ Programming)

Hours: 3 credit hours / 60 contact hours; 30 hours lecture, 30 hours lab

Instructor: TBD

Textbook, title, author, and year: *Starting Out With C++, 8th Edition*, Tony Gaddis, 2013

Specific course information

- a. **Catalog description:** Introduces basic concepts of systems programming. Students learn to apply basic programming concepts toward solving problems, create source files and implement header files, work with and effectively use basic data types, abstract data types, control structures, code modularization and arrays. Students will be introduced to object paradigm including, classes, inheritance, and polymorphism applications.
- b. **Prerequisites:** None

Specific goals for the course

- a. **Program Educational Objectives:**
 2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
- b. **Course Outcomes:**
 - Introduces basic concepts of systems programming.
 - Students learn to write computer programs in C++ that solve problems; solving problems with programs is the focus.
 - Students learn to create source files and implement header files, work with and effectively use basic data types, compile source code into binary executable files, and understand the use of project management.
- c. **Course student outcomes:**

Upon completion of this course the student should be able to do the following:

 - Design an algorithm to solve a problem.
 - Translate that algorithm into a C++ program.
 - Debug, test, and verify that the program solves the original problem.
 - Apply Object-Oriented-Programming methods in developing a program.
 - Use C++ standard libraries and classes.
 - Analyze programs for efficiency; change techniques and data structures to improve efficiency.

- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions **(ABET Computing Criterion 3.1)**
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline **(ABET Computing Criterion 3.2)**
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems **(ABET IT Criterion 3.6)**

Topics to be covered

- a. Using Introduction to Computers and Programming, Introduction to C++ Expressions and Interactivity, Making Decisions
- b. Loops, Making Decisions
- c. Loops
- d. Functions
- e. Arrays
- f. Arrays Part 2
- g. Introduction to Classes
- h. Continuation of Classes
- i. Pointers
- j. Inheritance, Polymorphism, and Virtual Functions
- k. Recursion
- l. Exceptions, Templates, and the Standard Template Library (STL)

ITM 313 SYLLABUS

ITM 313 Introduction to Open Source Application Development

Hours: 3 credit hours / 60 contact hours; 30 hours lecture, 30 hours lab

Instructor: Sheikh “Sam” Shamsuddin

Textbook, title, author, and year: *Introduction to Programming Using Python*, Daniel Y., 2013.

Specific course information

- a. **Catalog description:** Introduces basic concepts of systems programming using a modern open source language. Students learn to apply basic programming concepts toward solving problems, writing pseudocode, working with and effectively using basic data types, abstract data types, control structures, code modularization and arrays. They will learn to detect errors, work with variables and loops, and discover how functions, methods, and operators work with different data types. Students will be introduced to the object paradigm including classes, inheritance, and polymorphism.
- b. **Prerequisites:** None
- c. **Required**

Specific goals for the course

- a. **Program Educational Objectives:**
 2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
- b. **Course Outcomes:**
 - Gain experience skills to write, compile, execute, troubleshoot, analyze, evaluate, and resolve simple problems through program coding using Python computer language.
 - Acquire the ability to develop, synthesize, and identify important language standard libraries and utilities.
 - Build experience in software application and development theory and concepts
- c. **Course Student Outcomes:**

Upon successful completion of the course the student should be able to:

 - Recall and describe software application and development theory and concepts
 - Write, compile, execute, troubleshoot, analyze, evaluate, and resolve simple problems through program coding using Python computer language.
 - Develop, synthesize, and identify important language standard libraries and utilities.
 - Apply data transfer techniques between modules using parameters and return values.
 - Construct applications to use simple files for input and output.
 -

- Implement arrays as structures to contain data.
- Use a higher-level programming language to code, test, and debug software designs.
- Implement concepts of Object Oriented Programming (OOP), inheritance and polymorphism.
- Describe integration of Graphical User Interfaces (GUIs) and event driven programming.
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions **(ABET Computing Criterion 3.1)**
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program’s discipline **(ABET Computing Criterion 3.2)**
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems **(ABET IT Criterion 3.6)**

Topics to be covered

- a. Data Types. Arithmetic operation. Variable declarations. Flowcharts, Pseudo Codes, Simple programming. Software Development.
- b. Learning a programming language. Input/Output. Decision Structure.
- c. Continue on Decision Structure and Flowcharting.
- d. Repetition Structure (looping).
- e. Continue on Repetition Structure. Pseudo codes, Flowcharts.
- f. Debugging programs
- g. Functions and flowchart Array/List File I/O
- h. Objects and classes Object and classes – private data fields
- i. Inheritance
- j. Graphical User Interface (GUI).

ITM 497 SYLLABUS

ITM 497 Independent Study

Hours: Variable, but normally 1-3 credit hours / 15-45 contact hours

Instructor: TBD

Textbook, title, author, and year: Will vary based on specific course content.

Specific course information

- a. **Catalog description:** Special Projects.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will be defined by student course proposal or prospectus.
- b. **Course Student Outcomes:** Will be defined by student course proposal or prospectus.

Topics to be covered: Will be defined by student course proposal or prospectus.

ITM 498 SYLLABUS

ITM 498 Undergraduate Research Immersion: Team

Hours: 3 credit hours / 90 contact hours; 90 hour lab

Instructor: Steven Szmurlo

Textbook, title, author, and year: Will vary based on specific course content.

Specific course information

- a. **Catalog description:** This course provides a faculty-mentored immersive research experience as a part of a student team. Research topics are determined by faculty mentor's area of research.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will vary based on course content.
- b. **Course Student Outcomes:** Will vary based on course content.

Topics to be covered: Will vary based on course content.

ITMD 321 SYLLABUS

ITMD 321 Data Modeling and Applications

Hours: 3 credit hours / 45 contact hours

Instructor: Peter Fales

Textbook, title, author, and year:

Learning SQL: Mastering SQL Fundamentals, 2nd Edition, Alan Beaulieu, 2009.

Creating your MySQL Database: Practical Design Tips and Techniques, Marc Delisle, 2016.

Specific course information:

- a. **Catalog description:** Basic data modeling concepts are introduced. Hands-on database design, implementation, and administration of single-user and shared multi-user database applications using a contemporary relational database management system.
- b. **Prerequisites:** None
- c. **Required.**

Specific goals for the course

- a. **Program Education Objective:**
 1. Problem solve and create innovative answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.
- b. **Course Outcomes:**
At the conclusion of this course, each successful student will able to:
 - Describe the theoretical and physical concepts of a relational database.
 - Employ design methodology for databases and verify their structural correctness.
 - Use query languages, primarily SQL, and database management software.
 - Design and build a simple database management system and perform fundamental tasks involved with modeling, designing, and implementing a DBMS.
 - Demonstrate knowledge of essential DBMS concepts: database security, high availability, backup and recovery and SQL database tuning.
- c. **Course student outcomes:**
At the conclusion of this course, each successful student will able to:
 - Describe the theoretical and physical concepts of a relational database.
 - Explain the design methodology for databases and verify their structural correctness.
 - Use query language, primarily SQL, and their database related supported software.
 - Implement the theory behind the various database models and query languages.

- Design and build a simple database management system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.
- Outline essential DBMS concepts: database security, high availability, backup and recovery and SQL database tuning
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions (**ABET Computing Criterion 3.1**)
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline (**ABET Computing Criterion 3.2**)
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems (**ABET IT Criterion 3.6**)

Topics to be covered

- a. Introduction to databases DBMS Concepts
- b. Introduction to Virtualization
- c. Relational Model ER – Modeling Intro to SQL
- d. Database Design More SQL Intro
- e. Data Definition Language Data Types and Constraints Normalization
- f. Schema Definition Table Creation SQL Queries
- g. SQL Intro
- h. MySQL Workbench Introduction Filtering (WHERE clause)
- i. MySQL Workbench MySQL for Excel
- j. Joins
- k. Sets
- l. Case Studies MySQL Server
- m. Server Administration
- n. Subqueries Grouping and Aggregates
- o. Advanced Joins
- p. Conditional Logic
- q. Transactions
- r. Indexes Constraints
- s. Views Metadata
- t. Programming/APIs SQLite
- u. Database auditing and Security
- v. Database Backup and Disaster Recovery
- w. Introduction to NoSQL databases Big Data
- x. Deeper dive into DDL, DCL, TCL
- y. Deeper dive into Cassandra and CQL

ITMD 361 SYLLABUS

ITMD 361 Fundamentals of Web Development

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- Eloquent JavaScript, 2nd ed.* Haverbeke, M., 2014
- HTML5 for Web Designers 2nd ed.* Keith, J. & Andrew, R., 2016
- Responsive Web Design, 2nd ed.* Marcotte, E., 2014
- On Web Typography.* Santa Maria, J., 2014
- Pragmatic Version Control Using Git.* Swicegood, T., 2009

Specific course information:

- Catalog description: This course covers the creation and deployment of modern, standards-compliant web pages written in HTML, CSS, and JavaScript in the context of the client-server architecture of the web. Students create and deploy a website with multiple, structured pages cross-linked by a site navigation structure.
- Prerequisites:** None.
- Required.**

Specific goals for the course

a. Course Outcomes:

Students completing this course will learn to:

- Recognize HTML, CSS, and JavaScript markup and code in a web page/application
- Select the proper mark-up tags or code to achieve a particular result
- Identify improperly used markup and code
- Produce modern standards compliant web pages
- Deploy web pages to a public server
- Thoughtfully evaluate and adopt only the most standards-compliant documentation, libraries, and development techniques
- Write valid, well-formed semantic HTML; error-free, backward- and forward-compatible CSS; and error-free, progressively enhanced JavaScript over HTML pages that continue to function in the absence of JavaScript
- Effectively comment on and format source code for maximum readability
- Track development of a project over time and collaborate with others using version control.

c. Course student outcomes:

At the conclusion of this course, successful students will be able to:

- Recognize HTML, CSS, and JavaScript markup and code in a web page/application
- Select the proper mark-up tags or code to achieve a particular result
- Identify improperly used markup and code
- Produce modern standards compliant web pages

- Deploy web pages to a public server
- Thoughtfully evaluate and adopt only the most standards-compliant documentation, libraries, and development techniques
- Write valid, well-formed semantic HTML; error-free, backward- and forward-compatible CSS; and error-free, progressively enhanced JavaScript over HTML pages that continue to function in the absence of JavaScript
- Effectively comment on and format source code for maximum readability
- Track development of a project over time and collaborate with others using version control
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions **(ABET Computing Criterion 3.1)**
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline **(ABET Computing Criterion 3.2)**
- Communicate effectively in a variety of professional contexts **(ABET Computing Criteria 3.3)**

Topics to be covered:

- GIT; HTML history, syntax & fundamentals
- Well formed, valid semantic HTML; global attributes
- Responsive web design overview + HTML foundations
- Course refresher; work day
- CSS syntax & fundamentals; units of measure
- CSS: typography
- CSS: page layout (fluid grids), feature detection
- Responsive images, accessible media
- Responsive images; Javascript: syntax, fundamentals, data structures
- JavaScript: syntax and fundamentals; functional programming style
- DOM Scripting: Javascript + the Document Object Model; JQuery and alternatives
- Unobtrusive JavaScript & progressive enhancement
- JavaScript: non-blocking asynchronous patterns, page performance
- HTTP 1.1, HTTP/2, and REST architecture; server-side web development
- Parting material: Preprocessors

ITMD 362 SYLLABUS**ITMD 362 Human-Computer Interaction**

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *The New CSS Layout*. Andrew, Rachel, 2017
- b. *Designing for Touch*. Clark, Josh, 2015
- c. *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability, 3rd ed.* Krug, Steven, 2014
- d. *Thinking with Type: A Critical Guide for Designers, Writers, Editors, & Students. 2nd ed.* Lupton, Ellen, 2010
- e. *Responsive Design: Patterns and Principles*. Marcotte, Ethan, 2015

Specific course information

- a. **Catalog description:** Students in this course will learn the importance of human-computer interaction design and the effectiveness of user-centered design. The course will cover a survey of methods frequently used in the HCI profession, such as usability testing and prototyping, as well as general design principles and the use design guidelines. A particular emphasis will be placed on usability for website engineering. Students will apply knowledge from the field in the design and construction of user-centered websites.
- b. **Prerequisites:** ITMD 361
- c. **Required**

Specific goals for the course

- a. **Program Educational Objectives:**
 2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate
- b. **Course Outcomes:** At the conclusion of this course, successful students will be able to:
 - Describe the diversity of information system users and tasks, and their impact on design.
 - Describe the core concepts, applicability, and cost benefits of user-centered design.
 - Demonstrate how user-centered concerns can be incorporated into system development life cycles.
 - Explain the need to evaluate system usability.
 - Recall and apply general principles of design.
 - Describe and execute touch-friendly, mobile-first responsive web design.
 - Understand and apply core theories from human-computer interaction to web design and development.

c. Course student outcomes:

Students completing this course will be able to:

- Recall, describe and apply principles of user-centered design.
- Conduct task analysis & apply the information to user-centered design.
- Evaluate user interface designs with human subjects.
- Recall, explain, and apply the design principles of alignment, contrast, proximity, and repetition.
- Design and build a user-centered website applying HCI methods and good principles of design.
- Apply color and typography in web design to optimize the interface.
- Engage in agile, iterative web design and development individually and in teams, supported by version control.
- Write useful, descriptive messages attached to granular commits in a version control system.
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions **(ABET Computing Criterion 3.1)**
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline **(ABET Computing Criterion 3.2)**
- Communicate effectively in a variety of professional contexts **(ABET Computing Criteria 3.3)**
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems **(ABET IT Criterion 3.6)**

Topics to be covered

- a. Course Overview; Valid, Semantic HTML
- b. Mobile-First, Relative-Unit CSS and Media Queries
- c. Unobtrusive JavaScript; Form Elements, Data, and Events
- d. Typography: Content, User Interface
- e. The New CSS Layout
- f. Color and Material Design Systems
- g. Git in Team Settings; Catching Up
- h. Designing for Touch; Motion and Effects
- i. Activity Theory
- j. Usability: Guiding Principles, HCI Guidelines
- k. Ethics and Dark Patterns
- l. JavaScript-Backed Prototyping
- m. Traditional Usability Testing Methods; Remote Testing, Experimental Builds
- n. Parting Material: Assessing Front-End Libraries and Frameworks

ITMD 411 Syllabus

ITMD 411 Intermediate Software Development

Hours: 3 credit hours / 45 contact hours

Instructor: James Papademas

Textbook, title, author, and year:

Starting Out with Java: From Control Structures to Data Structures, 4th Edition, Tony Gaddis, 2020

Specific course information

- a. **Course Description:** This course covers object-oriented programming concepts in the Java Standard Edition platform. Employing the latest software development kit, the student considers software development topics in data structures, stream I/O, serialization, concurrency and graphical clients. Software engineering topics including packaging, deployment, debugging and unit testing. Hands on exercises reinforce concepts gained throughout the course. A final project integrates course topics into a contemporary Graphical User Interface client application.
- b. **Prerequisites:** (ITM 311 or CS 116 or CS 201) and (ITM 312 or ITM 313 or CS 331)
- c. **Required.**

Specific goals for the course:

- a. **Program Educational Objective:**
 1. Problem solve and create innovative answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.
- b. **Course Outcomes:**
Students completing this course will be able to:
 - Understand basic Object-Oriented programming concepts.
 - Apply Test Driven Development methodologies.
 - Understand packaging and deployment Java SE applications.
 - Describe Software development terminology.
- b. **Course student outcomes:**
Students completing this course will be able to:
 - Write Object Oriented Java SE code.
 - Create a Java based Graphical User Interface.
 - Locate application functionality from a large programmer API.
 - Author well-constructed code and software documentation.
 - Utilize an IDE to develop, test and debug Java SE code.

- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions **(ABET Computing Criterion 3.1)**
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline **(ABET Computing Criterion 3.2)**
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems **(ABET IT Criterion 3.6)**

Topics to be covered

- a. Java SE Essentials, algorithm development, the Java compiler/Eclipse IDE
- b. Object-Oriented Programming (OOP) concepts / OOAD (OOP Designs)
- c. Arrays and Array lists
- d. Abstraction, Inheritance, Polymorphism
- e. Stream I/O
- f. Exception Handling
- g. Generics
- h. Collections
- i. Stream API
- j. Linked Lists
- k. Stacks, Queues
- l. Databases
- m. GUI's – Swing
- n. Serialization
- o. Networking
- p. Regular Expressions
- q. Packaging and Deployment
- r. Concurrency
- s. JUnit testing / TDD

ITMD 412 SYLLABUS

ITMD 412 Advanced Structured and Systems Programming

Hours: 3 credit hours / 45 contact hours

Instructor: Sheikh "Sam" Shamsuddin

Textbook, title, author, and year:

Starting Out with C++: From Control Structures to Objects, 9th Edition, by Tony Gaddis, 2017.

Specific course information

- a. Catalog description: Structured programming continues with advanced concepts including strings, arrays, pointers, data structures, file manipulation, and dynamic memory management. Students create more complex applications that work with user input, manipulate user supplied text or text obtained from a file, apply standard library routines for working with literal text, use pointers to store complex structures within arrays, and read and write data from files, the console, and the terminal. The object-oriented programming (OOP) paradigm is covered in depth including the philosophy of OOP, classes and objects, inheritance, template classes, and making use of class libraries. Current technologies included in this course include the C++ programming language.
- b. Prerequisites: ITM 312

Specific goals for the course

- a. Course Outcomes:
 - Implement Object Oriented Programming Methodology
 - Integrate language Standard Libraries in program design
 - Implement data structure and algorithms
 - Apply programming language Templates features
 - Improve program Design and Analysis
- b. Course Student Outcomes:

Upon successful completion of the course the student should be able to do the following:

 - Recall and apply Data Structure Algorithms such as Stack, Linked-List, Queues and Binary Trees
 - Create programs and apply the C++ Standard Template Library (STL)
 - Mine the language additional Standard Library
 - Analyze, evaluate and implement the reuse Programming Methodology
 - Distinguish between Object Oriented Programming and Structured Programming
 - Develop and enhance Object Oriented Programming skills

Topics to be covered

- a. C++ Review
- b. Advanced File IO
- c. Classes and Friends
- d. Operator Overload and Copy CON
- e. Inheritance
- f. Polymorphism and Virtual Function
- g. Function Template
- h. Error Exception
- i. STL
- j. Linked-list, STL list
- k. Stack, STL Stack
- l. STL dequeue, Queue Containers
- m. Recursive
- n. Binary Tree Searching

ITMD 413 SYLLABUS

ITMD 413 Open Source Programming

Hours: 3 credit hours / 45 contact hours

Instructor: Sheikh "Sam" Shamsuddin

Textbook, title, author, and year: *Starting Out with Python, 4th Ed.*, by Tony Gaddis, 2018.

Specific course information

- a. **Catalog description:** Contemporary open-source programming languages and frameworks are presented. The student considers design and development topics in system, graphical user interface, network, and web programming. Dynamic scripting languages are covered using object-oriented, concurrent, and functional programming paradigms. Concepts gained throughout the course are reinforced with numerous exercises which will culminate in an open-source programming project.
- b. **Prerequisites:** ITMD 411
- c. **Required.**

Specific goals for the course

- a. **Course Outcomes:**
 - Learn how to write computer programming using Python language.
 - Learn the Python language, its structure, syntax concepts, libraries and application.
 - Learn Input/output, functions, data types, control structures, and lists/arrays.
 - Demonstrate Object Oriented Programming using Python.
 - Become confident in developing and writing Object Oriented Programs.
 - Test, design and solve problems using Python Programming Language
- b. **Course Student Outcomes:**

Upon successful completion of the course the student should be able to do the following:

 - Write, compile, execute, troubleshoot, and resolve problems using the Python Programming Language and its features.
 - Demonstrate Object Oriented Programming methodology in program development.
 - Identify important Python ample libraries.
 - Outline the fundamentals of Data Science.
 - Locate and use Help Resources.
 - Demonstrate implementation of a Graphical User Interface (GUI).
 - Analyze and evaluate software application and development theory and concepts.

Topics to be covered

- a. Data Class and course introduction.
 - i. Downloading using and Python.
 - ii. Learning simple Python programs.
- b. Simple I/O. Decision Structures.
- c. Repetition Structures. Functions.
- d. Lists and Tuples.
- e. File I/O and Exceptions.
- f. Creating Graphs with Matplotlib and Strings in Python.
- g. Data Analysis with Numpy.
- h. Statistical Data Analysis with Pandas.
- i. Data Visualization with Python seaborn package.
- j. Dictionary, and Sets
- k. Classes and Object-Oriented Programming.
- l. Object Oriented Programming.
- m. Inheritance.
- n. Recursions and Graphical User Interface (GUI).

ITMD 415 SYLLABUS

ITMD 415 Advanced Software Programming

Hours: 3 credit hours / 45 contact hours

Instructor: Scott Spyrison

Textbook, title, author, and year:

Java EE 8 Recipes A Problem-Solution Approach, Josh Juneau, 2018

The Java EE 8 Tutorial. Oracle Corporation, 2017
<https://javaee.github.io/tutorial/toc.html>

Specific course information

- a. **Catalog description:** Contemporary open-source programming languages and frameworks are presented. The student considers design and development topics in system, graphical user interface, network, and web programming. Dynamic scripting languages are covered using object-oriented, concurrent, and functional programming paradigms. Concepts gained throughout the course are reinforced with numerous exercises which will culminate in an open-source programming project.
- b. **Prerequisites:** ITMD 411

Specific goals for the course

- a. **Course Outcome:** At the completion of the course, each student will have designed, produced, and documented projects using the Java EE platform, culminating in a comprehensive and multi-tiered final project that builds cumulatively on prior work. Students will have deployed enterprise applications to modern application server environments. Students will have demonstrated knowledge of Java EE specifications, APIs, architectures and techniques, including security, database persistence, business components, web services and presentation components.
- b. **Course Student Outcomes:**
Upon successful completion of the course the student should be able to do the following:
 - Write, Use and administer modern Java EE application server
 - Explain the benefits and best practices associated with multi-tier/multi-layer enterprise applications
 - Persistence Layer (a.k.a. Information or Database Tier)
 - Business and/or Service Layer (a.k.a. Business Tier)
 - Presentation Layer (a.k.a. Web Tier)

- Describe the concepts of Declarative Programming, Inversion of Control and Configuration by Exception, and correlate these concepts with CDI and other Java EE specifications
- Explain the use of design patterns within the Java EE platform
- Describe the origins, benefits and weaknesses of JSP and JSTL technology
- Create JSP and JSTL pages consisting of several standard tags
- Produce Servlet code to process HTTP requests
- Differentiate between JDBC and JPA
- Implement Java code that uses common JPA annotations and JPA Query Language
- Use associations and inheritance to demonstrate Object-Relational Mapping
- Create business components with EJB technology and expose related web services
- Create web service producers based on EJB business components
- Differentiate between EL, JSF, JSP, JSTL and Servlet technologies
- Produce MVC web applications
- Explain how JSF relates to the MVC architecture
- Differentiate between authentication and authorization as security mechanisms
- Contrast Java EE with other modern frameworks
- Explain the benefits of Platform as a Service

Topics to be covered

- a. Introduction and Setup N/A
- b. Platform Basics (Git, Maven, Junit), JDBC and Validation
- c. Introduction to Web Applications with Servlet and JSP
- d. ORM/JPA I
- e. ORM/JPA II
- f. ORM/JPA III
- g. EJB/Service Layer
- h. Web Application Security/JSF I
- i. Web Application Security/JSF II
- j. JSF/MVC (Putting it all together) I
- k. JSF/MVC (Putting it all together) II
- l. Web Services/MVC or JSF/MVC

ITMD 419 SYLLABUS

ITMD 419 Topics in Software Development

Hours: Variable, but normally 3 credit hours / 45 contact hours

Instructor: TBD

Textbook, title, author, and year: Will vary based on specific course content.

Specific course information

- a. **Catalog description:** This course will cover a particular topic in software development, varying from semester to semester, in which there is particular student or staff interest. This course may be taken more than once but only 9 hours of ITMD 419/519 credit may be applied to a degree.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will vary based on specific course content.
- b. **Course Student Outcomes:** Will vary based on specific course content.

Topics to be covered: Will vary based on specific course content.

ITMD 422 SYLLABUS**ITMD 422 Advanced Topics in Data Management**

Hours: 3 credit hours / 45 contact hours

Instructor: Luke Papademas

Textbook, title, author, and year: *Database Systems: Design, Implementation & Management*, Carlos Coronel, Steven Morris, 2019

Specific course information

- a. **Catalog description:** Advanced topics in database management and programming including client server application development are introduced. Students will learn the use of Structured Query Language in a variety of application and operating system environments. Expands knowledge of data modeling concepts and introduces object - oriented data modeling techniques with specific attention to the use of database management systems in response to defined business problems.
- b. **Prerequisites:** ITMD 321.

Specific goals for the course

- a. **Course Outcomes:** Upon completion of this course, students will be able to use a Data Base Management System (DBMS) to create and manage files of data on a microcomputer system. The data structures for specific business applications will be created; enter, manipulate, and organize the data; issue data queries; use a report generator; restructure files; use the screen for input and output; and establish relationships between multiple files. The student will become aware of the need for back-up procedures as an integral part of data integrity.
- b. **Course Student Outcomes:**
 - Explain and use Data Modeling
 - Describe and use ERDs and DFDs
 - Recall SQL basics including
 - Datatypes
 - Character Functions
 - Date Functions
 - Aggregate Functions
 - Joins
 - Subqueries
 - Set Operators
 - Describe and employ database objects
 - Explain and use DML
 - Explain and use a Data Dictionary
 - Outline and describe essential elements of database security
 - Demonstrate PL / SQL Programming Basics
 - Describe and employ error handling, cursors, and triggers

Topics to be covered:

- a. RDBMS Systems and Concepts; Database Terminology; The Language of Database Systems; Database Set Theory; MS Access; OLAP (Online Analytical Processing)
- b. Data Modeling ; ERDs, UML; Business Requirements / Business Rules
- c. The Relational Database Model; A Review of SQL; SQL Statements
- d. Using Database Tools; Entity Relationship (ER) Modeling
- e. More on SQL / Data Analytics and Data Management; Advanced Data Modeling
- f. Topics in PL / SQL; Topics in Database Table Normalization
- g. More Topics in PL / SQL
- h. Topics in Data Analytics; Decision Making; Forecasting; Advanced SQL Concepts
- i. Object - Oriented Database Practices; Database Design
- j. Topics; Web Applications of Database Systems (ASP and the Web / PHP and the Web); Transaction Management
- k. Intelligent - Based Database Systems (Search Engines / Cortana); Database Performance Tuning / Optimization
- l. Big Data; Data Mining; Data Science; Distributed Database Management Systems
- m. Queuing Theory for Oracle Professionals; Business Intelligence and Data Warehouses
- n. Eliminating Data Redundancy; Big Data Analytics and NoSQL
- o. General Course Review; Database Connectivity; Database Administration and Security

ITMD 441 SYLLABUS

ITMD 441 Web Application Foundations

Hours: 3 credit hours / 45 contact hours

Instructor: Brian Bailey

Textbook, title, author, and year:

- a. None; online material is used

Specific course information:

- a. **Catalog description:** In this course students examine core web technologies that are integral in the creation of web-based applications typically delivered in a browser. The course will cover fundamental web protocols, web application architectures, markup, and scripting languages. A focus will be placed on writing modern, standards-compliant JavaScript and how it is used to interact with HTML and CSS to enable rich user interfaces and communication with other services. Current frameworks, libraries, and tools will also be explored.
- b. **Prerequisites:** ITMD 361

Specific goals for the course

- a. **Course Outcomes:** Students completing this course learn about the core technologies used to develop contemporary web applications. This includes the primary protocols, tools, software, markup, and scripting languages used in modern web development and how they have progressed over time. Each student successfully completing this course will demonstrate a strong foundational knowledge in the design and development of web-based internet applications. Students will use standards compliant HTML and CSS to create responsive user interfaces targeting modern browsers. Additionally, students learn about the JavaScript language, debugging techniques, and JavaScript APIs and how to effectively utilize them.
- b. **Course student outcomes:**
At the conclusion of this course, successful students will be able to:
 - Recognize HTML, CSS, and JavaScript markup and code in a web page/application
 - Describe the differences between server and client-side technology as it relates to applications delivered through internet browsers
 - Explain the history, role, and use of enabling technologies in modern web-based applications, including JavaScript, Cascading Style Sheets (CSS), and HTML including their impact
 - Outline strategies for turning interface mockups into working code prototypes
 - Describe the role of the HTTP and HTTPS protocol and the request/response cycle

- Use valid standards compliant HTML, CSS, and JavaScript to build web pages and applications
- Use modern browser developer tools to inspect and debug web applications
- Describe basic responsive design principles and techniques and explain the advantages of responsive design
- Discuss differences in ES5 and ES6+ JavaScript language and syntax
- Develop native ES6+ JavaScript for use in the browser
- Describe terminology and functionality afforded by advanced JavaScript programming paradigms including: Prototype, Object Oriented Programming (OOP), JavaScript Object Notation (JSON), Namespacing, Modules
- Compare AJAX techniques for asynchronously loading content
- Do Document Object Model (DOM) manipulation in native JavaScript
- Describe and utilize HTML5 APIs
- Outline the potential security threats posed to internet applications and their users and strategies for mitigating these risks
- Use tools to manage project dependencies and source code management tools for version control in a project

Topics to be covered:

- a. JavaScript, Cascading Style Sheets (CSS), HTML
- b. Standard Compliant markup and code
- c. HTML5 features, markup, and APIs
- d. JavaScript language syntax
- e. JavaScript version differences
- f. History of Internet applications and web technologies
- g. HTTP / HTTPS Protocol
- h. Client/Server architecture
- i. AJAX techniques for asynchronously loading content
- j. Creating working code prototypes from interface mockups
- k. Object Oriented Programming (OOP)
- l. JavaScript Object Notation (JSON)
- m. Namespacing
- n. ES6 Modules
- o. Security threats posed to internet applications
- p. DOM manipulation
- q. Responsive design principles
- r. Introduction to command line tools and version control (Git)
- s. Frameworks
- t. Online API documentation

ITMD 442 SYLLABUS**ITMD 442 Full-Stack Development**

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *Agile Development with Rails 7*. Ruby, S. with D. Thomas, 2022.

Specific course information:

- a. Catalog description: This course covers the fundamental concepts and techniques of full-stack web development, focusing on server-delivered front-end content such as server-rendered HTML or JSON and its integration with back-end architectures and data stores.
- b. **Prerequisites:** ITMD 441.
- c. **Required.**

Specific goals for the course**a. Course Outcomes:**

Students completing this course will learn to:

- Understand the connections between MVC architecture and full-stack web development
- Leverage an object-relational mapper (ORM) for data-driven application design and database-neutral implementation
- Become conversant with documentation and release notes in order to properly implement and stay current with ongoing development to a full-stack web framework
- Evaluate the design and architecture of a web or mobile system, including issues such as design patterns (including MVC), layers, tradeoffs between redundancy and scalability, state management, and search engine optimization.

c. Course student outcomes:

At the conclusion of this course, successful students will be able to:

- Analyze specific computing problems of information storage and dissemination, and articulate their requirements and appropriate solution in object-oriented languages
- Design, implement, and evaluate a Ruby on Rails web application that meets specific, desired user needs
- Understand and articulate how full-stack frameworks adhere, or fail to adhere, to open standards for networking and the web
- Read and interpret documentation and release notes for languages, libraries, and frameworks, and adjust professional practice based on the contents of that material
- Describe how authentication, secure certificates, and secure communication can be used in web sessions.

Topics to be covered:

- a. Ruby Fundamentals
- b. Ruby on Rails Fundamentals
- c. The Structure of Full-Stack Web Applications
- d. The Structure of Application Data
- e. Models
- f. Unit Testing
- g. Controllers
- h. Functional Testing
- i. Views
- j. Integration and Regression Testing

- k. Session Data and Users
- l. Third-Party Integrations (OAuth)
- m. Continuous Integration/Continuous Deployment
- n. Refactoring
- o. Tuning for Performance: Compression, Minifications

ITMD 443 SYLLABUS

ITMD 443 Front-End Development

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *Practical SVG*, Coyier, C. 2016
- b. *Going Responsive*, McGrane, K. 2015
- c. *On Web Typography*, Santa Maria, J., 2014
- d. *Cross-Cultural Design*, Akpem, S., 2020.

Specific course information:

- a. Catalog description: This course emphasizes front-end, browser-based components of web application development. It includes a robust survey of Web APIs in addition to advanced coverage of visual design executed in leading-edge CSS.
- b. **Prerequisites:** ITMD 442.

Specific goals for the course

- a. **Course Outcomes:**
Students completing this course will learn to:
 - Design a responsive web application utilizing a web framework and presentation technologies in support of a diverse online community.
 - Apply principles of UXD to enhance the user experience of a web site or mobile application.
 - Apply leading-edge web standards specifications in a backward-compatible way
- c. **Course student outcomes:**
At the conclusion of this course, successful students will be able to:
 - Compare characteristics such as color depth, compression, codec, and server requirements for media file formats.
 - Propose a graphic file type for a given set of image characteristics.
 - Engage in robust test coverage for all facets of front-end development.

Topics to be covered:

- a. ARIA Roles and ARIA in HTML
- b. Function-Syntax CSS
- c. Intrinsic Design
- d. Visual Regression Testing
- e. End-to-End Testing
- f. Variable Fonts
- g. SVG for Illustration
- h. SVG for User-Interface Components
- i. Cross-Cultural Design
- j. Internationalization (i18n)
- k. Service Workers
- l. Web Notifications
- m. Progressive Web Apps
- n. Front-End Preprocessors
- o. Auditing Front-End Assets

ITMD 444 SYLLABUS

ITMD 444 Back-End Development

Hours: 3 credit hours / 45 contact hours

Instructor: Brian Bailly

Textbook, title, author, and year: [Getting MEAN with Mongo, Express, Angular, and Node 2nd Edition](#).
Holmes, Simon and Harber, Clive. 2019

Specific course information

- a. **Catalog description:** This course emphasizes back-end, server-side components of web application development. It provides broad coverage of server-side data stores and languages, and surveys multiple leading server-side web development frameworks.
- b. **Prerequisites:** ITMD 321 AND ITMD 442

Specific goals for the course

- a. **Course Outcomes:**
Students learn to interact with back-end portion of web application development. This includes deploying standard back-end frameworks and deploying opensource datastores. The class will cover a single back-end development framework and focus on deploying a complete working web application
- b. **Course student outcomes:**
 - Explain the client and server architecture of the Internet and related web technologies.
 - Compare standard open source application back-end frameworks
 - Deploy a standard open source application back-end framework
 - Deploy back-end systems to communicate with frontends via APIs
 - Explain the benefits of using API layers for communication with datastores
 - Explain the benefits of full-stack development
 - Compare relational and non-relational (NoSQL) datastores fitness for deployment in applications
 - Implement and describe security concerns for back-end frameworks
 - Explain back-end security concerns and common security mitigation strategies

- Describe the role of HTTP and HTTPS in the context of web applications
- Evaluate back-end authentication standards
- Develop a web application that validates data inputs on the client- and server-side as appropriate
- Describe various frameworks, open source applications and their use cases
- Evaluate the design and architecture of a web system, including issues such as design patterns, layers, tradeoffs between redundancy and scalability, state and management
- Use version control tools to work with a team on a web development
- Use industry-standard tools and technologies for web development

Topics to be covered

- a. Introductions / Syllabus Review / Overview
- b. Introduction to commandline tooling and version control (Git)
- c. Introducing Node.js and Express
- d. Introducing MongoDB
- e. Discussing common MEAN Architectures
- f. Designing a flexible MEAN Architectures
- g. Setting up MEAN projects
- h. Defining Express routes
- i. Using Mongoose with MongoDB
- j. Exposing MongoDB via the application API
- k. Managing User Authentication
- l. Advanced Topics

ITMD 445 SYLLABUS**ITMD 445 Web Real-Time Communications**

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *Programming WebRTC: Build Real-Time Streaming Applications for the Web*. Stolley, K., 2022.

Specific course information:

- a. Catalog description: This course covers the WebRTC specification's set of protocols, architectures, and APIs designed to enable browser-to-browser real-time communication of voice, video, and data. Students will learn to apply basic technologies including WebSockets, HTTP, HTML5, Web Sockets, NAT, STUN, TURN, and ICE to ensure two-way real-time communication. Students will use JavaScript and development environments to create basic data and media applications based on WebRTC technologies and will analyze the impact of their applications on the performance and behavior of the networks that carry them.
- b. **Prerequisites:** ITMD 441.

Specific goals for the course

- a. **Course Outcomes:**
Students completing this course will learn to:
 - Describe the architecture and function of WebRTC's APIs in the browser
 - Reference the contents of the WebRTC specification
 - Build forward- and backward-compatible streaming applications in WebRTC according to the perfect-negotiation pattern
 - Analyze complex computing problems and apply principles of computing and other relevant disciplines to identify solutions
 - Participate in the creation, execution, and ongoing revision of an effective project plan
 - Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements
 - Communicate effectively in a variety of professional contexts
 - Function effectively as a member or leader of a team engaged in advanced web development
 - Identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems
 - Apply security principles and practices to maintain operations in the presence of risks and threats
- c. **Course student outcomes:**
At the conclusion of this course, successful students will be able to:
 - Command the suite of technologies and APIs that comprise WebRTC in the browser
 - Leverage a signaling server and the perfect-negotiation pattern to connect modern browsers over WebRTC
 - Explore automated testing to ensure code quality on WebRTC applications
 - Deploy WebRTC applications in production, including configuring a private STUN/TURN server

- Problem-solve and create innovative answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.
- Perform requirements analyses, design, and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
- Apply current technical and mathematical concepts and practices in the core information technologies and recognize the need to engage in continuing professional development.

Topics to be covered:

- a. WebRTC as a Web Standard
- b. Non-blocking Real-Time Foundations
- c. WebRTC Foundations I: HTTPS, Media Permissions
- d. WebRTC Foundations II: Signaling, Connection, Perfect Negotiation
- e. Alternative Signaling Servers
- f. WebRTC Data Channels
- g. Browser Compatibility and "Perfected" Negotiation
- h. Cross-Browser Feature Detection
- i. Binary Data
- j. Multipeer Connections and Mesh-Network Topography
- k. Multipeer Calls and Scale
- l. STUN/TURN Servers
- m. Testing WebRTC
- n. WebRTC in Production
- o. Monitoring with RTCStats

ITMD 446 SYLLABUS

ITMD 446 Web Microservices and APIs

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *Design and Build Great Web APIs*, Admundsen, M, 2020.

Specific course information:

- a. Catalog description: This course covers fundamental principles and methods for programmatically accessing and parsing data returned by internet-available data APIs. The course guides students in carefully examining the structure of API endpoints expressed as URLs and the conventions of RESTful architecture.
- b. **Prerequisites:** ITMD 441.

Specific goals for the course

a. Course Outcomes:

Students completing this course will learn to:

- Develop familiarity with the key principles of Application Programming Interfaces, for both programming languages and data/software services, and as both a consumer and producer of APIs
- Develop critical reading skills for navigating and comprehending the documentation for APIs
- Understand and apply progressive enhancement and responsive design in any user-facing web design and development
- Develop an agile approach to digital development, supplemented by the use of a version control system (Git)

c. Course student outcomes:

At the conclusion of this course, successful students will be able to:

- Learn to locate, read, and act in accordance with Web API providers' terms of service
- Understand and employ lightweight data serialization formats (e.g., JSON) on both the client- and server-side
- Understand and implement web standards from the World Wide Web Consortium (W3C), Ecma, ISO, and other standards-issuing groups

Topics to be covered:

- a. Request/Response and RESTful Architecture
- b. API Structure and Endpoints
- c. API Documentation
- d. Data Serialization Formats I: CSV, XML
- e. Data Serialization Formats II: JSON, JSONP
- f. Asynchronous Requests
- g. CORS Restrictions
- h. Validating Data Integrity
- i. From Consuming to Publishing APIs
- j. API-First Design
- k. Evaluating End-User/Consumer Needs
- l. Serving APIs: CORS Configuration
- m. Versioning API URLs and Schema
- n. Restrictive Access Strategies
- o. Client- and Server-Side Storage and Caching

ITMD 447 SYLLABUS

ITMD 447 Web Systems Integration
(This course was previously ITMD 467)

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

Design It! From Programmer to Software Architect, Michael Keeling, 2017

Release It! Design and Deploy Production-Ready Software, Michael Keeling, 2015

Specific course information

- a. **Catalog description:** In this project-based course, student teams will build an enterprise-grade website and web infrastructure integrating server-side applications, databases, and client-side rich internet applications as a solution to a defined business problem.
- b. **Prerequisites:** ITMD 465 and ITMD 462

Specific goals for the course

- a. **Course Outcomes:**
 - Problem-solve and create innovative answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.
 - Perform requirements analyses, design, and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
 - Apply current technical and mathematical concepts and practices in the core information technologies and recognize the need to engage in continuing professional development.
- b. **Course student outcomes:**
 - Analyze complex computing problems and apply principles of computing and other relevant disciplines to identify solutions
 - Assist in the creation, execution, and ongoing revision of an effective project plan
 - Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements
 - Communicate effectively in a variety of professional contexts
 - Function effectively as a member or leader of a team engaged in activities appropriate to web-systems integration tasks
 - Identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems

- Apply security principles and practices to maintain operations in the presence of risks and threats

Topics to be covered

- a. Starting Lines; Shaping and Sketching
- b. System-Design Fundamentals and Strategies
- c. One-on-One Conferences; Project Two
- d. System Architecture and Patterns; Dev Teams as Design Studios
- e. Describing and Illustrating System Architectures
- f. Dependence on Libraries and Frameworks
- g. Isolating Code and Working with Asynchronous Data
- h. Working with Data Structures & Data Stores
- i. Q&A; Document-style Databases, Configuration
- j. Shifting from Alphas to Betas, Designing for Production
- k. Interconnection, Configuration, Security
- l. Advanced Front-End Techniques
- m. Code Quality
- n. Deployment and Monitoring; CI/CD

ITMD 449 SYLLABUS

ITMD 449 Topics in Web Development

Hours: Variable

Instructor: Karl Stolley

Specific course information:

- a. Catalog description: This course will cover a particular topic in web development, varying from semester to semester, in response to a specific student or faculty interest. This course may be taken more than once, but only 9 hours of ITMD 449 credit may be applied to a degree.

ITMD 453 SYLLABUS

ITMD 453 Enterprise Intelligent Device Applications

Hours: 3 credit hours / 45 contact hours

Instructor: Martin Schray

Textbook, title, author, and year: *Windows Store App Development: C# and XAML*, Pete Brown, 2014.

Specific course information

- a. Catalog description: Intelligent device application development is covered with proprietary enterprise and open-source technologies on media device, mobile and robotic platforms. Utilizing contemporary toolkits, the student considers design and development on simulated and real “smart” devices including smart phones, tablets, sensors, actuators, drones, and robots. Numerous exercises reinforce concepts gained throughout the course. A term project will integrate course topics into a comprehensive intelligent device application.
- b. **Prerequisites:** ITMD 311
- c. **Required.**

Specific goals for the course

- a. **Course Outcomes:** Engage students in learning about Mobile Application development. This course will focus on mobile applications for the Windows Store (e.g. Windows RT Apps) and Windows Phone stores via Universal Apps. Students will learn how to successfully build apps and have their apps accepted for the Windows/Windows Phone Store. Students complete this course with practical ideas and skills letting you build cool, edgy and useful apps that will be successful in the Windows/Windows Phone Stores.
- b. **Course Student Outcomes:** Upon successful completion of the course the student should be able to do the following:
 - Create robust Windows apps formatted for submission to the Windows App Store
 - create apps using external libraries, control libraries, and third-party controls
 - create multiple page apps with appropriate page navigation
 - employ touch friendly design idea and controls
 - manipulate and employ images in apps
 - Submit apps completed as assignments to the Windows App Store
 - have 75% or more of apps submitted accepted into the store
 - achieve 100 plus downloads of accepted apps

Topics to be covered

- a. XAML development, Windows 8 dev methods
- b. Event handlers, constructing single page apps
- c. Windows store accounts, stacking panels and control composition, third-party controls
- d. Build app UI through “control composition”, more third party controls, basic use/UI building with infragistics controls
- e. Using MVVM & a modelview, implementing a modelview
- f. Async, Azure mobile services, addressing slow app/system processes
- g. Applications reuse between Windows 8 and Windows Phone enhance visual appeal of apps via animations and transforms
- h. Modify standard templates to change look & feel
- i. Using multiple pages, page navigation, manipulate application state application suspend and resume in multi-page apps
- j. Employ touch friendly design idea and controls
- k. Manipulate and employ images in apps
- l. Using calls to native code to add functionality
- m. Final Project work

ITMD 454 SYLLABUS

ITMD 454 Mass-Market Intelligent Device Applications

Hours: 3 credit hours / 45 contact hours

Instructor: TBD

Textbook, title, author, and year:

Java EE 8 Recipes A Problem-Solution Approach, Josh Juneau, 2018

The Java EE 8 Tutorial. Oracle Corporation, 2017
<https://javaee.github.io/tutorial/toc.html>

Specific course information

- a. **Catalog description:** Intelligent device application development is covered with leading mass-market and open-source technologies on media device, mobile and robotic platforms. Utilizing contemporary toolkits, the student considers design and development on simulated and real “smart” devices including smart phones, tablets, sensors, actuators, drones, and robots. Numerous exercises reinforce concepts gained throughout the course. A term project will integrate course topics into a comprehensive intelligent device application.
- b. **Prerequisites:** ITMD 312

Specific goals for the course

- a. **Course Outcome:** Students will become familiar with iOS architectures and fundamentals; software engineering principles and best practices; and design and development of iOS based applications for iPhone, iPad and iPod devices. The student will create a number of small applications; each will highlight specific iOS features. These exercises reinforce concepts gained throughout the course. This course will help the student to lay a solid foundation down the path to creating your own user friendly, robust, efficient iOS applications in the future.
- b. **Course Student Outcomes:**
Upon successful completion of the course the student should be able to:
 - Describe and use the iOS Application development process including the use of iOS SDK/APIs
 - Use the development toolkit XCode, Emulator, debugger, logging mechanism
 - Create Swift Programming language constructs
 - Design and create iOS UI components, controls and views
 - Use multimedia processing including image, sound and video animation

- Describe and create common applications such as messaging, maps, databases and networking
- Successfully build and publish applications for the Apple Store

Topics to be covered

- a. Introduction to iOS app, XCode, Simple App, UI elements, Swift Values and Types
- b. Simple App and Handling Basic Interaction, Swift Control Structures
- c. More UI, Rotation and Layout, Swift Control Structures
- d. Multi-view Apps, Swift Classes and Objects
- e. Tab Bars and Pickers
- f. Table Views
- g. Navigation Controllers and Table Views
- h. Collection Views
- i. Using Split Views and Popovers
- j. Application Settings and User Defaults
- k. Basic Data persistence
- l. Documents and iCloud
- m. Grand Central Dispatch, Background and You
- n. Drawing with Core Graphics

ITMD 455 SYLLABUS

ITMD 455 Open-Source Intelligent Device Applications

Hours: 3 credit hours / 45 contact hours

Instructor: James Papademas

Textbook, title, author, and year: *Professional Android 4 Application Development*, Reto Meier, 2012

Specific course information

- a. **Catalog description:** Intelligent device application development is covered with various technologies on mobile and robotic platforms. Utilizing contemporary toolkits, the student considers design and development on emulated and real "smart" devices including smart phones, personal digital assistants, sensors, actuators, and robots. Numerous exercises reinforce concepts gained throughout the course. A term project will integrate course topics into a comprehensive intelligent device application.
- b. **Prerequisites:** ITM 311

Specific goals for the course

- a. **Course Outcomes:** Engage students in learning about Mobile Application development. This section will focus on mobile applications for Android platforms. Students will learn how to successfully build and publish apps for the Android Marketplace.
- b. **Course student outcomes:**
 - Describe technical challenges posed by current mobile devices, including competitive devices and wireless communications; evaluate and select appropriate solutions.
 - Describe the need to keep up with rapid changes and new developments.
 - Identify current trends in mobile communications technologies and systems.
 - Use mobile analytics.
 - Select and evaluate suitable software tools and Google APIs for the development of a particular mobile application and understand their strengths, scope and limitations.
 - Use an appropriate application development to design, write and test small interactive programs for mobile devices (cells or tablets).
 - Demonstrate a deployable working app to sites such to Google's Play store and similar type stores.
 - Work a fully documented prototype model of the app and presentation of the mobile app. Themes are selected based on current trends in the mobile world.

Topics to be covered

- a. Course overview: Android versioning, the SDK/Studio IDE overview; Gradle The AVD.
- b. Android features and capabilities.
- c. App Creation Essentials.
- d. UI layouts.
- e. Life Cycles.
- f. Android Architecture.
- g. Intents, Splash screens.
- h. Creating Views.
- i. Adapters.
- j. File Processing.
- k. XML parsing vs. JSON parsing.
- l. Fragments.
- m. Advanced interfaces-Navigation Drawer, Tabbed Layouts.
- n. SQLite database: Content Providers, Preferences, Cursors, LogCat, Master / Detail
- o. Concurrency, background services.
- p. Advanced UX.
- q. Intro to API's- Google, Facebook, Firebase
- r. Distribution of App on the Play Store.
- s. Marketing your App.
- t. Final project

ITMD 460 SYLLABUS

ITMD 460 Fundamentals of Multimedia

Hours: 3 credit hours / 45 contact hours

Instructor: Philip Matuszak

Textbook, title, author, and year:

Multimedia: Making It Work, Ninth Edition; Tay
Vaughan, 2014

The Non-Designer's Design Book 4th Edition; Robin
Williams, 2014

Specific course information

- a. **Catalog description:** Students are introduced to computer-based multimedia theory, concepts, and applications. Topics include desktop publishing, hypermedia, presentation graphics, graphic images, animation, sound, video, multimedia on the World Wide Web and integrated multimedia authoring techniques.
- b. **Prerequisites:** None

Specific goals for the course

- a. **Course Outcome:** Each successful student in this survey course will be brought "up to speed" on the latest multimedia concepts, terms, and software types. While designed to give an overview of today's multimedia world and production methods, students in the course will gain enough practical knowledge to begin creating productions for desktop and Internet use.
- b. **Course Student Outcomes:**
Upon successful completion of the course the student should be able to:
 - Describe and discuss current trends in multimedia by experiencing a variety of applications and development packages.
 - Explain the preproduction process including content acquisition and development, process flow, team management and integration, and legal issues surrounding multimedia.
 - Demonstrate technical knowledge and limited proficiency in designing production elements in each of the multimedia disciplines.
 - Create a multimedia project for the desktop or Internet.

Topics to be covered

- a. Introduction to Multimedia
- b. Design Concepts
- c. Presentation Graphics
- d. Typefaces
- e. What is Multimedia
- f. Graphics
- g. Hardware & Software
- h. Use of Color and Cultural Contexts of Color
- i. Desktop Publishing
- j. Pre-Production
- k. Production Planning and design
- l. Hypermedia Authoring Concepts
- m. User Interface Design
- n. Presentation Graphics Design
- o. Multimedia Sound
- p. The Multimedia Team
- q. File Compression
- r. Animation
- s. Video Production
- t. Digital Video
- u. Authoring Part 2: HTML & Web-Based Multimedia
- v. Designing Web-based Multimedia
- w. Producing Multimedia
- x. Content & Legal Considerations for Multimedia
- y. Final Presentations

ITMD 466 SYLLABUS

ITMD 466 Service-Oriented Architecture

Hours: 3 credit hours / 45 contact hours

Instructor: Hosea Lee

Textbook, title, author, and year: *SOA with REST*,
Thomas Erl & Benjamin Carlyle, 2012.
https://javabrainz.io/courses/javaee_jaxrs

Specific course information

- a. **Catalog description:** This course covers IT enterprise systems employing web services technologies in SOA and ESB architectural patterns. The student considers SOA which defines and provisions IT infrastructure and allows for a loosely-coupled data exchange over disparate applications participating in business processes. The simplification of integration and flexible reuse of business components within SOA is greatly furthered by ESB. Lab exercises using contemporary tool-kits are utilized to reinforce platform-agnostic course topics.
- b. **Prerequisites:** ITMD 411 and 461

Specific goals for the course

- a. **Course Outcomes:** Each student will learn the fundamentals of SOA and REST and SOA with REST. Also, we will implement an existing RESTful service on a custom web page. In this class we will build REST services using JAX-RS. After the class, each student will have a strong understanding of SOA, REST services and be able to build his/her own REST services.
- b. **Course student outcomes:**
At the conclusion of this course each student will be able to:
 - Explain and employ Service-Oriented Computing
 - Describe SOA Design Principles with REST
 - Explain analysis and service modeling of REST
 - Recall REST restraints and goals
 - Describe the role and use of XML in SOA
 - Employ service-oriented design with REST
 - Build a REST service using JAX-RS
 - Implement an existing REST service (twitter) on a custom HTML page

Topics to be covered

- a. Introduction to SOA/Services
- b. Goals and benefits of Service Oriented Computing Service
- c. Orientation Design Principles
- d. Introduction to REST
- e. JAX-RS Project Overview
- f. REST Constraints and Goals
- g. Service Contracts with REST
- h. Service-Oriented with REST – JAX-RS
- i. Analysis and Service Modeling with REST
- j. REST design patterns
- k. SOA Methodology and REST

ITMD 469 SYLLABUS

ITMD 469 Topics in Application Development

Hours: Variable, but normally 3 credit hours / 45 contact hours

Instructor: TBD

Textbook, title, author, and year: Will vary based on specific course content.

Specific course information

- a. **Catalog description:** This course will cover a particular topic in application development, varying from semester to semester, in which there is particular student or staff interest. This course may be taken more than once but only 9 hours of ITMD 469/569 credit may be applied to a degree.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will vary based on specific course content.
- b. **Course Student Outcomes:** Will vary based on specific course content.

Topics to be covered: Will vary based on specific course content.

ITMD 504 SYLLABUS

ITMD 504 Programing and Application Foundations

Hours: 3 credit hours / 45 contact hours

Instructor: Travis Smith

Textbook, title, author, and year:

- a. Online readings as assigned in Coursera

Specific course information:

- a. **Catalog description:** This course covers creation and deployment of modern, standards-compliant web pages written in HTML, CSS, and JavaScript in the context of the client-server architecture of the web. Fundamentals of programming in a modern programming language is covered. Basic data modeling concepts are also introduced, including hands-on database design, implementation, and administration of single-user and shared multi-user database applications using a contemporary relational database management system. These topics are taught in an integrated hands-on manner, where students will learn how frontend, backend, and database systems are written and deployed to work together in a full stack web application.
- b. **Prerequisite/Corequisite:** None

Specific goals for the course

- a. **Program Education Objective:**
 2. Manage and deploy secure cloud-based information resources in an enterprise setting.
- b. **Course Outcomes:**
At the conclusion of this course, each successful student will be able to:
 - Build foundational architecture of modern web application using appropriate technologies for frontend user interfaces, backend web servers, and backend databases.
 - Write programs in a modern programming language to solve defined problems, using pseudocode, basic and abstract data types, control structures, code modularization, arrays, and object-orientation including classes, inheritance, and polymorphism.
 - Create a responsive website with standards-compliant web applications using HTML, CSS, and JavaScript.
 - Pass data from a front-end client to a backend web server and database.
 - Using valid database design concepts, implement and administer single-user and shared multi-user database applications using a contemporary relational database management system.
 - Integrate frontend, backend, and database systems to work together in a full stack application.
- c. **Course student outcomes:**
At the conclusion of this course, each successful student will be able to:

- Recall and describe software application and development theory and concepts
- Write, compile, execute, troubleshoot, analyze, evaluate, and resolve simple problems through program coding using Python.
- Develop, synthesize, and identify important language standard libraries and utilities
- Apply data transfer techniques between modules using parameters and return values
- Construct applications to use simple files for input and output
- Implement arrays as structures to contain data
- Use a higher-level programming language to code, test, and debug software designs
- Implement concepts of object-oriented programming (oop), inheritance and polymorphism
- Recognize HTML, CSS, and JavaScript markup and code in a web page/application
- Select the proper mark-up tags or code to achieve a particular result
- Identify improperly used markup & code
- Produce standards compliant web pages
- Create interactive web pages using Javascript
- Describe the theoretical and physical concepts of a relational database.
- Explain design methodology for databases and verify structural correctness
- Use query language, primarily SQL, and their database related supported software.
- Implement the theory behind the various database models and query languages.
- Design and build a simple database management system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.
- Outline essential DBMS concepts: database security, high availability, backup and recovery and SQL database tuning
- Create a website that integrates content from a DBMS

Topics to be covered

- a. Introduction to databases & DBMS Concepts
- b. Introduction. IIT banner, class basics, intro to IDE for Python.
- c. Introducing Python Programming
- d. Data Types
- e. Functions
- f. Control flow statements
- g. Lists and Tuples
- h. Dictionaries and Sets
- i. Modules, Packages, File Operations
- j. Error Handling
- k. Introduction to Python web application frameworks
- l. HTML basics, basic tenets, hyperlinks, formatting elements, HTML5, content organization, semantic HTML
- m. Core CSS Concepts, basic structure, stylesheets, inheritance, CSS Float, CSS element positioning, CSS overflow property

- n. CSS and UI Content, User interfaces, CSS and content Flow.
- o. CSS and Graphics
- p. Javascript introduction, updating UI with Javascript.
- q. Animations with Javascript and Data
- r. Touch interfaces and APIs
- s. Device and operating system resources
- t. Understanding Core Database Concepts
- u. Basics of Web security
- v. Database tables
- w. Relational Database Core Concepts
- x. Data Manipulation language (DML)
- y. Data Definition Language (DDL)
- z. Creating Database Objects
- aa. Choosing Data Types
- bb. Database Object Management using DDL
- cc. Constructing queries to create, alter, and drop views
- dd. Create Stored Procedures and functions
- ee. Manipulating Data in Database tables
- ff. Select, Insert, Update, Delete Data
- gg. Transaction Controls
- hh. Database normalization
- ii. Primary Foreign, and Composite keys
- jj. Clustered vs non clustered data
- kk. Database administration
- ll. Database security concepts
- mm. Database Backup and Disaster recovery
- nn. HTTP 1.1, HTTP/2, and REST architecture; server-side web development
- oo. Introduction to NoSQL databases and Big Data
- pp. Basics of CRUD applications

ITMD 522 SYLLABUS

ITMD 522 Data Mining and Machine Learning

Hours: 3 credit hours / 45 contact hours

Instructor: Yong Zheng

Textbook, title, author, and year:

(all are optional but are strongly recommended)

Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Peter Flach. 2012

Python Data Science Handbook: Essential Tools for Working with Data 1st Edition, Jake VanderPlas. 2016

Specific course information

- d. **Catalog description:** Data mining is a useful tool to uncover patterns and underlying relationships in large data by using data analytics and knowledge discovery techniques. Machine learning algorithms additionally learn from the data and make predictions or decisions by different optimization methods. This course is a graduate level survey of concepts, principles and techniques related to data mining and machine learning. Students will be familiar with data preprocessing skills and the popular data mining and machine learning techniques, including the supervised learning (regressions and classification) and unsupervised learning (clustering and association rules analysis), as well as semi-supervised learning and ensemble learning. Students will also learn the related applications, including text mining/NLP, Web mining, information retrieval and recommender systems. Students will learn R and Python programming for data mining and machine learning and be able to handle real-world data or applications.
- e. **Prerequisites/Corequisites:** ITMD 514 or ITMS 514

Specific goals for the course

- a. **Program Educational Outcome:** Deliver optimal technical and policy technology solutions for the problems of business, industry, government, non-profit organizations, and individuals in each student's particular area of focus.
- b. **Course Outcomes:** At the completion of the course, each student will have the capability to deal with real-world data mining problems by using standard knowledge discovery in databases (KDD) process. More specifically, students are able to perform data selection, data preprocessing and data mining on real-world data sets. Students will have demonstrated their knowledge and skills in classical data mining techniques, including classification, clustering, association rule analysis, etc. Students are able to use related tools (R and Python) to apply these data mining techniques.

They are expected to be familiar with popular applications, including Web mining, information retrieval and recommender systems.

c. Course Student Outcomes:

Upon successful completion of the course the student should be able to do the following:

- Describe what data mining is and why it is so useful
 - Discuss real-world data mining techniques and applications
- Recall and distinguish between related terms: artificial intelligence, data mining, machine learning, etc
- Explain the standard Knowledge Discovery in Databases (KDD) process
- Perform popular data preprocessing: data selection, data cleaning, data transformation, etc
- Explain the difference between classification and clustering
- Perform and evaluate different classification algorithms over real-world data sets
- Perform and evaluate different clustering algorithms over real-world data sets
- Explain how association rule analysis works and how it is used to assist business intelligence
- Perform and evaluate associate rule mining to discover useful patterns over real-world data sets
- Describe the principles of and build related applications, including Web Mining applications, information retrievals, recommender systems, etc
- Explain how regression analysis works and how useful it is
- Use data mining tools (e.g., R and Python) to perform data mining tasks
- Describe basic evaluation protocols, challenges (e.g., overfitting, imbalance issues) and corresponding solutions in different data mining tasks
- Explain the pros and cons of each commonly used technique for specific data mining tasks. For example, among different classification techniques, what are their pros and cons.
 - Determine which techniques should be used in various situations for specific problems.
- Identify problems and determine which of multiple appropriate techniques are optimal to solve the problems.

Topics to be covered

- a. Intro: Data Science & KDD: Preprocessing
- b. Classification by KNN & Naïve Bayes
- c. Application: Information Retrieval and Text Classifications
- d. Classification by Decision Trees and SVM
- e. Ensemble Methods a Multi-Label Classifications

- f. Clustering Techniques
- g. Association Rules and Web Mining
- h. Outlier Detection, Feature Selection and Reductions
- i. R for Data Science
- j. R for Data Science with Practice
- k. Recommender Systems and Collaborative Filtering
- l. Traditional Recommender Systems
- m. Python for Data Science
- n. Python for Data Science with Practice
- o. Advanced Topics in Machine Learning
- p. Exams or Project Presentations
- q. Applications: Recommender Systems
- r. Exams or Project Presentations

ITMD 524 SYLLABUS

ITMD 524 Applied Artificial Intelligence and Deep Learning

Hours: 3 credit hours / 45 contact hours

Instructor: Yong Zheng

Textbook, title, author, and year:

(all are optional but are strongly recommended)

Artificial Intelligence: A Modern Approach (4th Edition). Stuart Russell, Peter Norvig. 2020

Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Peter Flach. 2012

Deep Learning. Ian Goodfellow, Yoshua Bengio, Aaron Courville. 2015

Specific course information

- a. **Catalog description:** Artificial Intelligence (AI) is being used extensively to solve real-world complex problems. This course will deliver concepts and skills in both classical AI and modern AI. The classical AI refers to the fundamental knowledge in AI, such as search, logic, planning, uncertainty, game theory, Markov models, etc. Modern AI, by contrast, will be concentrated on machine learning and deep learning techniques, especially their applications in NLP, object recognition, recommender systems, etc. Students will learn how to use Python to solve specific AI problems.
- b. **Prerequisites:** ITMD 514 or ITMS 514

Specific goals for the course

- a. **Program Educational Outcome:** Deliver optimal technical and policy technology solutions for the problems of business, industry, government, non-profit organizations, and individuals in each student's particular area of focus.
- b. **Course Outcomes:** At the completion of the course, each student will have the capability to deal with real-world AI problems, such as search, logics, uncertainty, optimization, etc. Particularly, students should be familiar with the knowledge and concepts in machine learning and deep learning, and they are able to utilize the libraries in Python to deal with deep learning applications, such as NLP, object recognition, recommender systems, etc.
- c. **Course Student Outcomes:** Upon successful completion of the course the student should be able to do the following:
 - Describe what AI is and why it is so useful
 - Describe real-world AI problems and applications
 - Recall and distinguish between related terms: artificial intelligence, data mining, machine learning, etc

- Explain and employ traditional AI problems and techniques, including search, logics, uncertainty, probability, markov model, game theory, etc
- Perform popular data preprocessing tasks: data selection, data cleaning, data transformation, etc
- Describe the difference between different machine learning categories, including supervised learning, unsupervised learning, semi-supervised learning, solutions
- Perform and evaluate different AI solutions using Python
- Use the deep learning framework TensorFlow
 - Use TensorFlow to solve AI problems and build AI applications

Topics to be covered

- a. Intro: Data Mining, Machine Learning & AI
- b. Traditional AI: Search 1
- c. Traditional AI: Search 2
- d. Traditional AI: First-Order Logics
- e. Traditional AI: Bayes Nets and Uncertainty
- f. Traditional AI: Uncertainty and Probability
- g. Traditional AI: Markov Model & Game Theory
- h. Midterm Exam and Intro to Machine Learning
- i. Modern AI: Machine Learning & Optimization
- j. Modern AI: Neural Networks & Deep Learning
- k. Modern AI: ANN, CNN, RNN
- l. Modern AI: ANN, CNN, RNN with TensorFlow
- m. Applications: NLP
- n. Applications: Object Recognition
- o. Applications: Recommender Systems
- p. Exams or Project Presentations

ITMD 541 SYLLABUS

ITMD 541 Web Application Foundations

Hours: 3 credit hours / 45 contact hours

Instructor: Brian Bailey

Textbook, title, author, and year:

- a. None; online material is used

Specific course information:

- a. **Catalog description:** In this course students examine core web technologies that are integral in the creation of web-based applications typically delivered in a browser. The course will cover fundamental web protocols, web application architectures, markup, and scripting languages. A focus will be placed on writing modern, standards-compliant JavaScript and how it is used to interact with HTML and CSS to enable rich user interfaces and communication with other services. Current frameworks, libraries, and tools will also be explored.
- b. **Prerequisites:** None.
- c. **Required.**

Specific goals for the course

- a. **Course Outcomes:** Students completing this course learn about the core technologies used to develop contemporary web applications. This includes the primary protocols, tools, software, markup, and scripting languages used in modern web development and how they have progressed over time. Each student successfully completing this course will demonstrate a strong foundational knowledge in the design and development of web-based internet applications. Students will use standards compliant HTML and CSS to create responsive user interfaces targeting modern browsers. Additionally, students learn about the JavaScript language, debugging techniques, and JavaScript APIs and how to effectively utilize them.
- b. **Course student outcomes:**
At the conclusion of this course, successful students will be able to:
 - Recognize HTML, CSS, and JavaScript markup and code in a web page/application
 - Describe the differences between server and client-side technology as it relates to applications delivered through internet browsers
 - Explain the history, role, and use of enabling technologies in modern web-based applications, including JavaScript, Cascading Style Sheets (CSS), and HTML including their impact
 - Outline strategies for turning interface mockups into working code prototypes
 - Explain the role of the HTTP and HTTPS protocol and the request/response cycle

- Master valid standards compliant HTML, CSS, and JavaScript to build web pages and applications
- Use modern browser developer tools to inspect and debug web applications
- Demonstrate basic responsive design principles and techniques and explain the advantages of responsive design
- Discuss differences in ES5 and ES6+ JavaScript language and syntax
- Understand native ES6+ JavaScript for use in the browser
- Describe terminology and functionality afforded by advanced JavaScript programming paradigms including: Prototype, Object Oriented Programming (OOP), JavaScript Object Notation (JSON), Namespacing, Modules
- Demonstrate AJAX techniques for asynchronously loading content
- Master Document Object Model (DOM) manipulation in native JavaScript
- Explain and utilize HTML5 APIs
- Discuss the potential security threats posed to internet applications and their users and strategies for mitigating these risks
- Understand the use of tools to manage project dependencies and source code management tools for version control in a project

Topics to be covered:

- a. JavaScript, Cascading Style Sheets (CSS), HTML
- b. Standard Compliant markup and code
- c. HTML5 features, markup, and APIs
- d. JavaScript language syntax
- e. JavaScript version differences
- f. History of Internet applications and web technologies
- g. HTTP / HTTPS Protocol
- h. Client/Server architecture
- i. AJAX techniques for asynchronously loading content
- j. Creating working code prototypes from interface mockups
- k. Object Oriented Programming (OOP)
- l. JavaScript Object Notation (JSON)
- m. Namespacing
- n. ES6 Modules
- o. Security threats posed to internet applications
- p. DOM manipulation
- q. Responsive design principles
- r. Introduction to command line tools and version control (Git)
- s. Frameworks
- t. Online API documentation

ITMD 542 SYLLABUS**ITMD 542 Full-Stack Development**

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *Agile Development with Rails 7*. Ruby, S. with D. Thomas, 2022.

Specific course information:

- a. Catalog description: This course covers the fundamental concepts and techniques of full-stack web development, focusing on server-delivered front-end content such as server-rendered HTML or JSON and its integration with back-end architectures and data stores.
- b. **Prerequisites:** ITMD 441 or ITMD 541.
- c. **Required.**

Specific goals for the course**a. Course Outcomes:**

Students completing this course will learn to:

- Understand the connections between MVC architecture and full-stack web development
- Leverage an object-relational mapper (ORM) for data-driven application design and database-neutral implementation
- Become conversant with documentation and release notes in order to properly implement and stay current with ongoing development to a full-stack web framework
- Evaluate the design and architecture of a web or mobile system, including issues such as design patterns (including MVC), layers, tradeoffs between redundancy and scalability, state management, and search engine optimization.
- Identify, analyze, and assess user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems
- Assess, apply, and continually monitor security principles and practices used to maintain operations in the presence of risks and threats

c. Course student outcomes:

At the conclusion of this course, successful students will be able to:

- Analyze specific computing problems of information storage and dissemination, and articulate their requirements and appropriate solution in object-oriented languages
- Design, implement, and evaluate a Ruby on Rails web application that meets specific, desired user needs
- Understand and articulate how full-stack frameworks adhere, or fail to adhere, to open standards for networking and the web
- Read and interpret documentation and release notes for languages, libraries, and frameworks, and adjust professional practice based on the contents of that material
- Describe how authentication, secure certificates, and secure communication can be used in web sessions.
- Investigate and apply current technical and mathematical concepts and practices in core information technologies and recognize the

need to engage in continuing professional development.

- Develop familiarity with the academic research and scholarship surrounding web design and development.

Topics to be covered:

- a. Ruby Fundamentals
- b. Ruby on Rails Fundamentals
- c. The Structure of Full-Stack Web Applications
- d. The Structure of Application Data
- e. Models
- f. Unit Testing
- g. Controllers
- h. Functional Testing
- i. Views
- j. Integration and Regression Testing
- k. Session Data and Users
- l. Third-Party Integrations (OAuth)
- m. Continuous Integration/Continuous Deployment
- n. Refactoring
- o. Tuning for Performance: Compression, Minifications

ITMD 543 SYLLABUS

ITMD 543 Front-End Development

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *Practical SVG*, Coyier, C. 2016
- b. *Going Responsive*, McGrane, K. 2015
- c. *On Web Typography*, Santa Maria, J., 2014
- d. *Cross-Cultural Design*, Akpem, S., 2020.

Specific course information:

- a. Catalog description: This course emphasizes front-end, browser-based components of web application development. It includes a robust survey of Web APIs in addition to advanced coverage of visual design executed in leading-edge CSS.
- b. **Prerequisites:** ITMD 442 or ITMD 542.

Specific goals for the course

- a. **Course Outcomes:**
Students completing this course will learn to:
 - Design a responsive web application utilizing a web framework and presentation technologies in support of a diverse online community.
 - Apply principles of UXD to enhance the user experience of a web site or mobile application.
 - Apply leading-edge web standards specifications in a backward-compatible way
 - Extract front-end components into a reusable system or framework
 - Track and perhaps contribute to the development of open web standards
- c. **Course student outcomes:**
At the conclusion of this course, successful students will be able to:
 - Compare characteristics such as color depth, compression, codec, and server requirements for media file formats.
 - Propose a graphic file type for a given set of image characteristics.
 - Engage in robust test coverage for all facets of front-end development.
 - Explore problems and develop actionable answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.

Topics to be covered:

- a. ARIA Roles and ARIA in HTML
- b. Function-Syntax CSS
- c. Intrinsic Design
- d. Visual Regression Testing
- e. End-to-End Testing
- f. Variable Fonts
- g. SVG for Illustration
- h. SVG for User-Interface Components
- i. Cross-Cultural Design
- j. Internationalization (i18n)
- k. Service Workers
- l. Web Notifications
- m. Progressive Web Apps
- n. Front-End Preprocessors
- o. Auditing Front-End Assets

ITMD 544 SYLLABUS

ITMD 544 Back-End Development

Hours: 3 credit hours / 45 contact hours

Instructor: Brian Bailly

Textbook, title, author, and year: [*Getting MEAN with Mongo, Express, Angular, and Node*](#) 2nd Edition. Holmes, Simon and Harber, Clive. 2019

Specific course information

- a. **Catalog description:** This course emphasizes back-end, server-side components of web application development. It provides broad coverage of server-side data stores and languages, and surveys multiple leading server-side web development frameworks.
- b. **Prerequisites:** ITMD 542

Specific goals for the course

- a. **Course Outcomes:**
Students learn to interact with back-end portion of web application development. This includes deploying standard back-end frameworks and deploying opensource datastores. The class will cover a single back-end development framework and focus on deploying a complete working web application
- b. **Course student outcomes:**
 - Explain the client and server architecture of the Internet and related web technologies.
 - Compare standard open source application back-end frameworks
 - Deploy a standard open source application back-end framework
 - Deploy back-end systems to communicate with frontends via APIs
 - Explain the benefits of using API layers for communication with datastores
 - Explain the benefits of full-stack development
 - Compare relational and non-relational (NoSQL) datastores fitness for deployment in applications
 - Implement and describe security concerns for back-end frameworks
 - Explain back-end security concerns and common security mitigation strategies

- Describe the role of HTTP and HTTPS in the context of web applications
- Evaluate back-end authentication standards
- Develop a web application that validates data inputs on the client- and server-side as appropriate
- Describe various frameworks, open source applications and their use cases
- Evaluate the design and architecture of a web system, including issues such as design patterns, layers, tradeoffs between redundancy and scalability, state and management
- Use version control tools to work with a team on a web development
- Use industry-standard tools and technologies for web development

Topics to be covered

- a. Introductions / Syllabus Review / Overview
- b. Introduction to commandline tooling and version control (Git)
- c. Introducing Node.js and Express
- d. Introducing MongoDB
- e. Discussing common MEAN Architectures
- f. Designing a flexible MEAN Architectures
- g. Setting up MEAN projects
- h. Defining Express routes
- i. Using Mongoose with MongoDB
- j. Exposing MongoDB via the application API
- k. Managing User Authentication
- l. Advanced Topics

ITMD 546 SYLLABUS**ITMD 546 Web Microservices and APIs**

Hours: 3 credit hours / 45 contact hours

Instructor: Karl Stolley

Textbook, title, author, and year:

- a. *Design and Build Great Web APIs*, Admundsen, M, 2020.

Specific course information:

- a. Catalog description: This course covers fundamental principles and methods for programmatically accessing and parsing data returned by internet-available data APIs. The course guides students in carefully examining the structure of API endpoints expressed as URLs and the conventions of RESTful architecture.
- b. **Prerequisites:** ITMD 541.

Specific goals for the course**a. Course Outcomes:**

Students completing this course will learn to:

- Develop familiarity with the key principles of Application Programming Interfaces, for both programming languages and data/software services, and as both a consumer and producer of APIs
- Develop critical reading skills for navigating and comprehending the documentation for APIs
- Understand and apply progressive enhancement and responsive design in any user-facing web design and development
- Develop an agile approach to digital development, supplemented by the use of a version control system (Git)
- Identify, analyze, and assess user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems
- Assess, apply, and continually monitor security principles and practices used to maintain operations in the presence of risks and threats

c. Course student outcomes:

At the conclusion of this course, successful students will be able to:

- Learn to locate, read, and act in accordance with Web API providers' terms of service
- Understand and employ lightweight data serialization formats (e.g., JSON) on both the client- and server-side
- Understand and implement web standards from the World Wide Web Consortium (W3C), Ecma, ISO, and other standards-issuing groups
- Investigate and apply current technical and mathematical concepts and practices in core information technologies and recognize the need to engage in continuing professional development.
- Develop familiarity with the academic research and scholarship surrounding web design and development.

Topics to be covered:

- a. Request/Response and RESTful Architecture
- b. API Structure and Endpoints
- c. API Documentation
- d. Data Serialization Formats I: CSV, XML

- e. Data Serialization Formats II: JSON, JSONP
- f. Asynchronous Requests
- g. CORS Restrictions
- h. Validating Data Integrity
- i. From Consuming to Publishing APIs
- j. API-First Design
- k. Evaluating End-User/Consumer Needs
- l. Serving APIs: CORS Configuration
- m. Versioning API URLs and Schema
- n. Restrictive Access Strategies
- o. Client- and Server-Side Storage and Caching

ITMD 549 SYLLABUS

ITMD 549 Topics in Web Development

Hours: Variable

Instructor: Karl Stolley

Specific course information:

- a. Catalog description: This course will cover a particular topic in web development, varying from semester to semester, in response to a specific student or faculty interest. This course may be taken more than once, but only 9 hours of ITMD 549 credit may be applied to a degree.

ITMM 464 SYLLABUS

ITMM 464 Social Media Marketing

Hours: 3 credit hours / 45 contact hours

Instructor: TBD

Textbook, title, author, and year:

The Social Media Bible, Second Edition, Safko, Lon, 201

Social Media Marketing: The Next Generation of Business Engagement, Evans, Dave, 2010

- b. Media Tools: Review of current products and services
- c. Social Media Strategies
- d. Running a Social Business
- e. Social CRM Basic Data persistence
- f. Documents and iCloud
- g. Grand Central Dispatch, Background and You
- h. Drawing with Core Graphics

Specific course information

- a. **Catalog description:** Class participants will explore the tactics, tools and strategies of incorporating new media channels to successfully grow a business, and/or to maximize the goals of other types of organizations.
- b. **Prerequisites:** None.

Specific goals for the course

- a. **Course Outcome:** Students completing this course will have produced several case study reports demonstrating knowledge of the major tools, tactics, and strategies presented in the course, and will have developed a Social Media Marketing program for a real or fictional organization.
- b. **Course Student Outcomes:** Upon successful completion of the course the student should be able to:
 - Recall and discuss current tools, tactics, and strategies of Social Media Marketing appropriate to business requirements at the professional, managerial and executive levels.
 - Apply techniques learned in class in the development of a Social Media Marketing plan appropriate to each of these levels.
 - Describe and discuss how Social Media Marketing can be integrated in an organization's overall marketing plan.
 - Develop a Social Media Marketing program.

Topics to be covered

- a. Introduction Social Media Components and Tactics:
 - i. Social Networks
 - ii. Blogs & Microblogs
 - iii. Photo Sharing
 - iv. Audiocasting/Videocasting
 - v. Search Engine Optimization
 - vi. Search Engine Marketing
 - vii. Mobile Apps

ITMM 470 SYLLABUS**ITMM 470 Fundamentals of Management for
Technology Professionals**

Hours: 3 credit hours / 45 contact hours

Instructor: Chuck Beck

Textbook, title, author, and year:

Essentials of Management, 9th Edition, Andrew J.
DuBrin, 2011

*Schaum's Outline of Managerial Accounting, 2nd
Edition (Schaum's Outlines)*, Joel G. Siegel and
Jae K. Shim, 2011

Budgeting for Managers, 1st Edition, Eric Dunbar and
Sid Kemp, 2003.

Specific course information:

- a. **Catalog description:** This course explores fundamentals of management for professionals in high-technology fields. It addresses the challenges of the following: managing technical professionals and technology assets; human resource management; budgeting and managerial accounting; management of services, infrastructure, outsourcing, and vendor relationships; technology governance and strategy; and resource planning.
- b. **Prerequisites:** None.

Specific goals for the course

- a. **Course Outcomes:** The objective of the course is to expose students to the many facets of management and to explore how each of these integrate with day-to-day management, strategic planning and leadership. Students will understand challenges associated with moving from a technical position to a management position. Each successful student will demonstrate foundation knowledge and application of the core concepts of management, of managerial accounting, and resource and asset management
- b. **Course Student Outcomes:**
At the conclusion of this course, students will be able to:
 - Recall and describe core concepts of management
 - Recall and explain key elements of managerial accounting
 - Calculate Return on Investment (ROI)
 - Read and interpret a balance sheet/financial statement
 - Describe resource and asset management
 - Prepare and manage a budget

- Describe and discuss methods and procedures for human resource management
 - Hiring employees
 - Evaluating employees
 - Motivating employees
 - Terminating employees
- Identify and discuss specific challenges related to the management of technology and technology professionals

Topics to be covered

- a. Introduction to Management
- b. Ethics Planning and Tools
- c. Managerial Accounting: Introduction
- d. Managerial Accounting: ROI, Capital Budgeting & Financial Statements
- e. Budgeting: Creating a Budget
- f. Budgeting: Spending & Tracking
- g. Human Resources
- h. Human Resources & Recruiting
- i. Performance Management
- j. Leadership & Motivation
- k. Communication & Collaboration
- l. Controlling
- m. Online Assets, Offshore Staffing, Vendor Management, and Outsourcing

ITMM 471 SYLLABUS**ITMM 471 Project Management for IT**

Hours: 3 credit hours / 45 contact hours

Instructor: Brian Vanderjack, MBA, PMP, CSM, PMI-ACP

Textbook, title, author, and year*The 7 Habits of Highly Effective People*, Stephen R. Covey, 1989*A Guide to the Project Management Body of Knowledge 6th ed.* (A.K.A. the PMBOK), Project Management Institute, 2017**Specific course information**

- a. **Catalog description:** Basic principles of project management are taught with a particular focus on project planning for information technology hardware, software and networking project implementation. Management of application development and major Web development projects will also be addressed.
- b. **Prerequisites:** ITM 100
- c. **Required.**

Specific goals for the course

- a. **Program Educational Objective:**
 3. Apply current technical and mathematical concepts and practices in the core information technologies and recognize the need to engage in continuing professional development
- b. **Course Outcomes:** Each successful student will be able to use and understand the concepts of project management as it applies to an information technology project.
- c. **Course student outcomes:**
 - Recall and describe concepts of project management
 - Recall and describe the Approaches for Agile
 - Describe Iterative and Adaptive Environments
 - Describe and successfully use Scrum/Agile
 - Describe how these practices integrate in project settings
 - Discuss strategic and business knowledge--including discussion of project management business documents--and information on the PMI Talent Triangle(tm) and the essential skills for success in today's market.
 - Explain the role of information technology as the facilitating function in a modern business enterprise
 - Describe considerations for deployment of technology in a business setting

- Explain the standards of professional communication used within the information technology profession
 - Communicate (understand, and respond) in a manner appropriate to the profession
- Successfully resolve conflicts related to a project
- Analyze a business problem and identify and define computing requirements and manage a project appropriate to its solution
- Communicate effectively in a variety of professional contexts
(ABET Computing Criterion 3.3)
- Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles
(ABET Computing Criterion 3.4)
- Function effectively as a member or leader of a team engaged in the design and development of information technology solutions to business problems
(ABET Computing Criterion 3.5)
- Identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems
(ABET Computing Criterion 3.6)
- Assist in the creation of an effective project plan
(Illinois Tech ITM Student Outcome (g))

Topics to be covered

- a. What is a project?
- b. Design Thinking
- c. Scrum/Agile
- d. Leadership
- e. Conflict Management
- f. PMI on Integration Management*
- g. PMI on Scope Management*
- h. PMI on Time Management*
- i. PMI on Cost Management*
- j. PMI on Quality Management*
- k. PMI on Human Resources Management*
- l. PMI on Communications Management*
- m. PMI on Risk Management*
- n. PMI on Procurement Management*
- o. PMI on Stakeholder Management*
- p. PMI On Ethics

ITMM 481 SYLLABUS**ITMM 481 IT Entrepreneurship**

Hours: 3 credit hours / 45 contact hours

Instructor: Adarsh Arora

Textbook, title, author, and year: *Disciplined Entrepreneurship*, Bill Aulet, Wiley publishing, 2013**Specific course information:**

- a. **Catalog description:** This course offers a comprehensive, integrated, and proven step-by-step approach to creating innovative, highly successful IT enterprises. Students follow a multi-step process resulting in their presenting a product/service plan for building a start-up IT enterprise. Students are expected to be prepared to discuss their ideas in each class as venture experience faculty mentor them in the formulation of their plan
- b. **Prerequisites:** None.

Specific goals for the course

- a. **Course outcomes:** Each student group will develop and present a product/service plan for a possible IT start-up gain knowledge of a proven approach to creating innovative IT start-ups exhibit in-class leadership, teamwork, and communication skills through regular classroom student presentations
- b. **Course student outcomes:**
At the conclusion of this course each student should be able to:
 - Evaluate Market Segmentation & select a beached market
 - Build an end user profile
 - Conduct a Tam Analysis
 - Create a Customer profile
 - Complete a use case needs analysis
 - Create High level product specifications and prototype specifications
 - Quantify a Value Proposition
 - Define your enterprise's core values
 - Chart a Competitive position
 - Determine Customer Decision making unit, roles, and influencers.
 - Define a process to acquire paying customer
 - Describe a Sales Process
 - Design a business model
 - Set a pricing structure
 - Calculate lifetime value and cost of customer acquisition
 - Identify key customer and product assumptions
 - Test key assumptions
 - Determine Market Size & a Product plan
 - Explain Funding Strategies
 - Describe types of companies, equity distribution and tax consequences

Topics to be covered

- a. What is your idea?
 - i. **Course overview and Possible Projects**
- b. Who is your Customer?
 - i. Step 1, 2 - Market Segmentation & select a beached market
 - ii. Step 3 - Build end user profile
 - iii. Step 4,5,9 - Tam Analysis, Customer profile and next 10 customers
- c. What can you do for your customer?
 - i. Step 6 - Use case needs analysis
 - ii. Step 7 - High level product specification, Prototype specification.
 - iii. Step 8 - Quantify Value Proposition
 - iv. Step 10 - Define your core values
 - v. Step 11 - Chart Competitive position
- d. How does your customer acquire your product?
 - i. Step 12 - Determine Customer Decision making unit, roles, influencers.
 - ii. Step 13 - Define process to acquire paying customer
 - iii. Step 18 - Sales Process
- e. How do you make money off your product?
 - i. Step 15 - Design a business model
 - ii. Step 16 - Set your pricing structure
 - iii. Step 17,19 - Calculate lifetime value and cost of customer acquisition.
- f. How do you design and build your product?
 - i. Step 20 - Identify key Assumptions
 - ii. Step 21 - Test key assumptions
 - iii. Step 22,23 - Show the dogs will eat the dog food.
- g. How do you scale your business?
 - i. Step 14 - Determine Market Size & Product plan
- h. Funding Strategies
- i. Types of companies, Equity distribution and tax consequences

ITMM 482 SYLLABUS

ITMM 482 Business Innovation

Hours: 3 credit hours / 45 contact hours

Instructor: Adarsh Arora

Textbook, title, author, and year: *Ten Types of Innovation*, Larry Keeley, 2013.

Specific course information:

- a. **Catalog Information:** This course offers comprehensive, integrated, and proven approaches for creating innovations in a business setting. Students follow a multi-dimensional framework to create and identify specific areas of innovations in their projects. Students are expected to be prepared to discuss their ideas in each class as industry experienced faculty mentor them in the formulation and presentation of their projects.
- b. **Prerequisites:** None.

Specific goals for the course:

- a. **Course outcomes:** Each student group will
 - Develop and present a product/service plan for a possible IT start-up
 - Gain knowledge of a proven approach to creating innovative IT start-ups
 - Exhibit in-class leadership, teamwork, and communication skills through regular classroom student presentations
- b. **Course student outcomes:**
At the conclusion of this course each student should be able to:
 - Describe the innovation gap
 - Explain the concept of balanced innovation
 - Recall and describe Porter's five forces for new offerings
 - Recall, describe, and apply Ten Types of Innovation
 - Discuss the propeller model of organizational dynamics
 - Develop and present a product/service plan for a possible IT start-up

Topics to be covered

- a. Innovation Gap – Why most innovations fail
- b. Why teams are lazy – Social loafing
- c. Balanced Innovations
- d. Porter's five forces for new offerings

- e. Ten Types of Innovations
 - i. Innovations in Business models
 - ii. Innovations in network and partnerships
 - iii. Innovations in enabling process/structure
 - iv. Innovations in core process
 - v. Innovations in product performance
 - vi. Innovations in product systems
 - vii. Innovations in services
 - viii. Innovations in brands
 - ix. Innovations in channels
 - x. Innovations in customer experience
- f. Propeller model of organizational dynamics – a case study
- g. Invited presentations from industry

ITMM 485 SYLLABUS**ITMM 485 Legal and Ethical Issues in Information Technology**

Hours: 3 credit hours / 45 contact hours

Instructor: Ray Trygstad

Textbook, title, author, and year:*Ethics and Technology: Controversies, Questions, and Strategies for Ethical Computing, 5th Edition,* Herman Tavani, 2016*The Future Computed: Artificial Intelligence and its role in society,* Microsoft Corporation, 2018. eBook provided on Blackboard.**Specific course information:**

- a. **Catalog description:** Current legal issues in information technology are addressed including elements of contracting, payment systems and digital signatures, privacy concerns, intellectual property, business torts, and criminal liability including hacking, computer trespass and fraud. Examination of ethical issues including privacy, system abuse, and ethical practices in information technology equip students to make sound ethical choices and resolve legal and moral issues that arise in information technology
- b. **Prerequisites:** None.
- c. **Required for Applied Cybersecurity and Information Technology.**

Specific goals for the course

- a. **Course Outcomes:** Each successful student will demonstrate an understanding of professional, ethical, legal, security and social issues and responsibilities in information technology; and will demonstrate an ability to make sound ethical choices and resolve legal and moral issues that arise in professional practice.
- b. **Course student outcomes:**
At the conclusion of this course, each successful student will be able to:
 - Discuss basic concepts of ethics, morality and the law.
 - Describe the differences between ethics, morality and law.
 - Identify ethical procedures and behaviors in the organization related to the employment and use of information technology.
 - Identify key ethical concerns of information technology professionals.
 - Recognize and describe Codes of Ethics and Professional Conduct and identify the industry organizations they are associated with.

- Describe issues related to privacy and confidentiality as they relate to information technology.
 - Identify key laws and regulations related to privacy and confidentiality.
- Discuss issues in cybercrime and technology-facilitated crime.
 - Identify key laws and regulations related to cybercrime.
- Describe issues related to intellectual property, intellectual property law, freedom of expression, and intellectual freedom as they relate to information technology.
 - Describe the differences between copyrights, patents, trademarks and trade secrets.
- Explain specific issues of contract law common in information technology
 - Describe and discuss issues related to outsourcing and Service Level Agreements.
- Discuss the social impact of information technology on society.
 - Explain the concept of “digital divide”, identify some causes and discuss possible solutions.
 - Identify underlying gender, cultural and diversity related issues in information technology.
- Create policies and procedures for an organization that are ethically, morally and legally sound.
 - Produce an Acceptable Use Policy with appropriate mechanisms for enforcement.

Topics to be covered

- a. Introduction to Ethics and the Law
- b. Ethical Concepts and Theories
- c. Ethics, Law and Conflict
- d. Professional Ethics in Information Technology
Privacy
- e. Security and Compliance
- f. Cybercrime and Technology-Facilitated Crime
- g. Intellectual Property Issues
- h. Regulating Commerce and Speech
- i. Information Technology Contract Law: SLAs and Outsourcing Technology, Work and the Digital Divide
- j. Online Identity, Community, and Artificial Intelligence Ethics, Law and Emerging Technologies
- k. IT Law and Ethics in Context

ITMM 487 SYLLABUS

ITMM 487 Product Management

Hours: 3 credit hours / 45 contact hours

Instructor: Chuck Beck

Textbook, title, author, and year:

None. All assigned reading will come from online sources or materials shared by the instructor.

Specific course information:

- a. **Catalog description:** This course explores the Product Management role in information technology and common processes, tools, and methods employed. Students will learn to identify business opportunities and market strategies as well as management of development teams in product developments that cross business and technology boundaries.
- b. **Prerequisites:** None.

Specific goals for the course

- a. **Course Outcomes:** The objective of the course is to expose students to the many facets of product management and to explore how a product backlog can be defined, built, and delivered. We will explore common tasks, challenges, and interactions that product managers face on a recurring basis. Each successful student will be able to demonstrate foundational knowledge and application of the core concepts of product management.
- b. **Course Student Outcomes:** At the conclusion of this course, students will be able to:
 - Recall and explain the role of a product manager
 - Engage stakeholders to identify business opportunities that product management can help solve
 - Engage users to identify value adds and efficiency gains that product management can help solve
 - Perform competitive analysis and research market conditions related to a software product
 - Transform ideas into product designs and describe how to work with UI/UX resources
 - Describe common methodologies used in product management
 - Refine a product backlog and prioritize requests including how to obtain level of effort estimates
 - Describe what a minimum viable product (MVP) is and negotiate scope
 - Recall and explain the differences between epics, features, and user stories
 - Create clearly written, non-ambiguous user stories that deliver business value

- Formulate acceptance criteria and the definition of done for a user story / feature
- Define KPI's and use metrics to analyze product performance
- Describe methods to determine if roadmap progress is on track and how to estimate delivery dates
- Lead cross-functional teams to deliver value through product development and enhancements
- Recall and describe the role of product management after a product is released
- Recall and describe key differences between physical vs. software product management challenges
- Identify and discuss common tools used in product management
- Recall certifications and training for product management and describe how product managers continue to advance their knowledge and skill sets

Topics to be covered

- a. Discovery
- b. Market Research & Analysis
- c. Design
- d. Breaking Down the Work
- e. Testing & Validation
- f. Methodologies
- g. Backlogs & Prioritization
- h. Roadmap & MVP
- i. KPI's and Metrics
- j. Leading Cross Functional Teams
- k. Support / Training
- l. Warranty / Documentation
- m. Digital vs. Physical Products & Tools of the Trade
- n. Certification, Training, and Beyond

ITMM 587 SYLLABUS

ITMM 587 Product Management

Hours: 3 credit hours / 45 contact hours

Instructor: Chuck Beck

Textbook, title, author, and year:

None. All assigned reading will come from online sources or materials shared by the instructor.

Specific course information:

- a. **Catalog description:** This course explores the Product Management role in information technology and common processes, tools, and methods employed. Students will learn to identify business opportunities and market strategies as well as management of development teams in product developments that cross business and technology boundaries.
- b. **Prerequisites:** None.

Specific goals for the course

- a. **Course Outcomes:** The objective of the course is to expose students to the many facets of product management and to explore how a product backlog can be defined, built, and delivered. We will explore common tasks, challenges, and inter-actions that product managers face on a recurring basis. Each successful student will be able to demonstrate foundational knowledge and application of the core concepts of product management.
- b. **Course Student Outcomes:** At the conclusion of this course, students will be able to:
 - Recall and explain the role of a product manager
 - Engage stakeholders to identify business opportunities that product management can help solve
 - Engage users to identify value adds and efficiency gains that product management can help solve
 - Perform competitive analysis and research market conditions related to a software product
 - Transform ideas into product designs and describe how to work with UI/UX resources
 - Recall and describe common methodologies used in product management
 - Refine a product backlog and prioritize requests including how to obtain level of effort estimates
 - Explain what a minimum viable product (MVP) is and negotiate scope
 - Recall and explain the differences between epics, features, and user stories
 - Create clearly written, non-ambiguous user stories that deliver business value

- Formulate acceptance criteria and the definition of done for a user story / feature
- Define KPI's and use metrics to analyze product performance
- Explain how to determine if roadmap progress is on track and how to estimate delivery dates
- Lead cross-functional teams to deliver value through product development and enhancements
- Recall and describe the role of product management after a product is released
- Recall and explain key differences between physical vs. software product management challenges
- Identify and discuss common tools used in product management
- Recall certifications and training for product management and describe how product managers continue to advance their knowledge and skill sets

Topics to be covered

- a. Discovery
- b. Market Research & Analysis
- c. Design
- d. Breaking Down the Work
- e. Testing & Validation
- f. Methodologies
- g. Backlogs & Prioritization
- h. Roadmap & MVP
- i. KPI's and Metrics
- j. Leading Cross Functional Teams
- k. Support / Training
- l. Warranty / Documentation
- m. Digital vs. Physical Products & Tools of the Trade
- n. Certification, Training, and Beyond

ITMO 340 SYLLABUS

ITMO 340 Introduction to Data Networking and the Internet

Hours: 3 credit hours / 45 contact hours

Instructor: Louis F. McHugh IV

Textbook, title, author, and year: *The Official CompTIA Network+ Student Guide* James Pengelly, 2019

Specific course information

- a. **Catalog description:** This course covers current and evolving data network technologies, protocols, network components, and the networks that use them, focusing on the Internet and related LANs. The state of worldwide networking and its evolution will be discussed. This course covers the Internet architecture, organization, and protocols including Ethernet, 802.11, routing, the TCP/UDP/IP suite, DNS, SNMP, DHCP, and more. Students will be presented with Internet-specific networking tools for searching, testing, debugging, and configuring networks and network-connected host computers. There will be opportunities for network configuration and hands-on use of tools.
- b. **Prerequisites:** None.
- c. **Required**

Specific goals for the course

- a. **Program Educational Objective**
 2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
 3. Apply current technical and mathematical concepts and practices in the core information technologies and recognize the need to engage in continuing professional development.
- b. **Course Outcomes:**
The course is a foundation course in the basics of Data Communications and Computer Networks. It is intent is to serve as a basis for practical studies in field of Computer Networking and Network Administration. Upon completion, a student should be able to understand how a Computer Network works from both a practical and theoretical perspective. They should understand OSI & TCP/IP Models, Various Networking Protocols, Data Circuits, Switches, and Routers. They will also have an understanding of troubleshooting and management of networks by usage of various tools.

c. Course Student Outcomes:

Each successful student will demonstrate foundation knowledge and application of the following skills:

- Outline the basics components of a computer network using both the TCP/IP protocol suite and the OSI model.
- Identify the various types of network systems, including local area networks, metropolitan area networks, wide area networks, and voice/data delivery networks.
- Enumerate the various transmission media commonly used in carrier systems, i.e. twisted pair, coaxial cable, fiber optic cable, terrestrial microwave, satellite, as well as other wireless technologies.
- Recognize the basics of data communications, including data, signals, conversions between data and signals, encoding techniques, multiplexing, and modulation.
- Identify the various types of error detection and error corrections schemes.
- Identify the basics of T-carrier systems, frame relay, asynchronous transfer mode, DSL, and cable modems, and be able to compare and contrast their characteristics.
- Describe the basic operating procedures of the Internet and how it relates to data and voice communications.
- Enumerate the differences between the wireless telephone systems D-AMPS, TDMA, CDMA, GSM, and others.
- Document the characteristics of local area networks, including hub and switch technologies.
- Complete a case study in which, given a minimum set of requirements, you will recommend wide area network solutions.
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions (**ABET Computing Criterion 3.1**)
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline (**ABET Computing Criterion 3.2**)
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems (**ABET IT Criterion 3.6**)

ITMO 340 Topics to be covered

- a. Explaining the OSI and TCP/IP Models 1
- b. Explaining Properties of Network Traffic
- c. Installing and Configuring Switched Networks
- d. Configuring IP Networks
- e. Installing and Configuring Routed Networks
- f. Configuring and Monitoring Ports and Protocols
- g. Explaining Network Applications and Storage Services
- h. Monitoring and Troubleshooting Networks
- i. Explaining Networking Attacks and Mitigations
- j. Installing and Configuring Security Devices
- k. Explaining Authentication and Access Control
- l. Deploying and Troubleshooting Cabling Solutions
- m. Implementing and Troubleshooting Wireless Technology
- n. Comparing and Contrasting WAN Technologies
- o. Using Remote Access Methods
- p. Identifying Site Policies and Best Practices

ITMO 356 SYLLABUS

ITMO 356 Introduction to Open Source Operating Systems

Hours: 3 credit hours / 60 contact hours; 30 hours lecture, 30 hours lab

Instructor: Sean Hughes-Durkin

Textbook, title, author, and year: *Linux+ Guide to Linux Certification, 4th/ED* Cengage, Jason W. Eckert, 2016

Specific course information:

- a. **Catalog description:** Students learn to set up and configure an industry-standard open source operating system including system installation and basic system administration; system architecture; package management; command-line commands; devices, filesystems, and the filesystem hierarchy standard. Also addressed are applications, shells, scripting and data management; user interfaces and desktops; administrative tasks; essential system services; networking fundamentals; and security, as well as support issues for open source software. Multiple distributions are covered with emphasis on the two leading major distribution forks.
- b. **Prerequisites:** None
- c. **Required.**

Specific goals for the course

- a. **Program Educational Objective:**
 2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
- b. **Course Outcomes:**
 - Describe the origins of and explain the philosophy of Open Source Software
 - Install, configure and administer an industry-standard distribution of the Linux operating system.
 - Troubleshoot and resolve Linux installation problems and common system problems
- c. **Course student outcomes:** Students completing this course will be able to:
 - Use and administer Linux as both a server and desktop operating system
 - Describe the GPL, GNU, and history of the Linux operating system
 - Install different Linux distributions with custom partitioning

- Navigate the graphical interface of the Linux operating system
- Navigate the filesystem using the command line
- Interact with the Linux shell
- Recall and use key Linux utilities
- Install software for use with the Linux operating system
- Use networking services and describe how to troubleshoot issues
- Use SSH for remote administration and create customer host firewall rules
- Configure an Apache web server
- Create shell scripts for use with automation
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline
(ABET Computing Criteria 3.2)
- Identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems discipline
(ABET Computing Criteria 3.6)

Topics to be covered

- a. Introduction to Open Source Software
- b. Installing Linux
- c. Exploring Linux Filesystems
- d. Linux Filesystem Management
- e. Filesystem Administration
- f. Linux Server Deployment
- g. Working with the BASH Shell
- h. System Initialization and X Windows
- i. Process Management
- j. Administrative Tasks Compression, Backups, and Software Install
- k. Network Configuration
- l. Network Services
- m. Troubleshooting
- n. Performance
- o. Security

ITMO 417 SYLLABUS

ITMO 417 Shell Scripting for System Administration

Hours: 3 credit hours / 45 contact hours

Instructor: Sheikh "Sam" Shamsuddin

Textbook, title, author, and year: *Practical Guide to Linux® Commands, Editors, and Shell Programming, 3rd Edition*, Mark G. Sobell, 2012.

Specific course information

- a. **Catalog description:** This course focuses on preparation of shell scripts to enhance and streamline system administration tasks in all contemporary server operating systems. Scripting will be taught in both native and portable environments. The course will address shell programming, regular expressions, common and system-specific shell utilities and built-in commands, user defined and shell variables, flow control structures, shell functions, and the creation and execution of shell scripts. Homework and hands-on exercises will provide practical experience in contemporary server environments.
- b. **Prerequisites:** ITMO 556

Specific goals for the course

- a. **Course Outcomes:**
 - Gain experience skills to write, compile, execute, troubleshoot, analyze, evaluate, and resolve problems using major Shell Scripting Languages - BASH, AWK, Perl, and PowerShell
 - Acquire the ability to develop, synthesize, and identify important language standard libraries and utilities
 - Build experience in software application and development theory and concepts
 - Gain the ability to locate and use of Help Resources
- b. **Course Student Outcomes:**
Upon successful completion of the course the student should be able to do the following:
 - Write and resolve programming problems using BASH, Perl, and PowerShell
 - Apply and use script utilities such as grep, sed, and system commands
 - Complete programming assignments using various scripting languages
 - Describe the integration and implementation of scripting and operating system commands
 - Recall scripting syntax and the language features
 - Create effective and useful Shell Script Programs

Topics to be covered

- a. Data Class and course introduction
- b. BASH Scripting, AWK, grep and sed
 - i. Shell commands. Shell Variables. Linux utilities.
 - ii. Selection Control, I/O and Redirection.
 - iii. Loops, Functions
 - iv. Regular Expressions – Grep and Sed
 - v. Arrays and File I/O
 - vi. Scripting using AWK
- c. Perl Scripting
 - i. Scalar Variables, Lists and Arrays.
 - ii. Hashes
 - iii. I/O, Selection Control, Loops
 - iv. Subroutines, File I/O
- d. PowerShell
 - i. Simple programming. Selection Control and Loop.
 - ii. Arrays, Modules, and Files

ITMO 433 SYLLABUS**ITMO 443 Windows Enterprise Server Administration**

Hours: 3 credit hours / 60 contact hours

Instructor: Louis F. McHugh IV

Textbook, title, author, and year: *Microsoft Official Academic Course: Installing and Configuring Windows Server 2012*, Craig Zacker, 2014

Specific course information

- a. **Catalog Description:** Students learn to set up, maintain, and administer x86-based servers and associated networks using a contemporary industry-standard proprietary operating system. Topics include hardware requirements; software compatibility; system installation, configuration and options, and post-installation topics; administrative and technical practices required for system security; process management; performance monitoring and tuning; storage management; back-up and restoration of data; and disaster recovery and prevention. Also addressed is configuration and administration of common network and server services such as DNS, DHCP, remote access, email, basic virtualization, web and web services, and more.
- b. **Prerequisites:** ITM 301 and ITMO 340/540

Specific goals for the course

- a. **Course Outcomes:** Upon completion, a student should be able to:
 - Setup Active Domain within a Client Server Environment
 - Administer x86-based servers such as Windows Enterprise Server 2012
 - Configure server, network, and software applications setup environment
 - Explain a server administrator's responsibilities and support of enterprise applications
 - Analyze server recurring issues and work on sustainable solutions
 - Provide support planning, development, integration, testing and management of IT services
- b. **Course student outcomes:**
Each successful student will demonstrate foundation knowledge and application of the following skills:
 - Install and manage Windows Enterprise Server and Active Directory Domain
 - Manage server, active directory, client computers and users
 - Apply server security, software distribution and updates

- Perform schedules vulnerability-assessment scans and monitoring server performance
- Design Infrastructure requirements
- Setup and manage server system virtualization
- Demonstrate day-to-day management of the server operating system, file structure, and directory services
- Monitor logs for firewalls and intrusion detection systems

Topics to be covered

- a. Introduction, Setting up MLO
- b. Installing Servers,
- c. Configuring Servers
- d. Configuring Local Storage
- e. Configuring File and Share Access
- f. Configuring Print and Document Services
- g. Configuring Servers for Remote Access
- h. Creating and Configuring Virtual Machine Settings
- i. Creating and Configuring Virtual Machine Storage
- j. Creating and Configuring Virtual Machine Networks
- k. Configuring IPv4 and IPv6 Addressing
- l. Deploying and Configuring DHCP
- m. Deploying and Configuring DNS
- n. Installing Domain Controllers
- o. Creating and Managing Active Directory (AD) Users and Computers
- p. Creating and Managing Active Directory (AD) Groups and Organizational Units
- q. Creating Group Policy Objects (GPO)
- r. Configuring Security Policies
- s. Configuring Application Restriction Policies
- t. Configuring Windows Firewall

ITMO 441 SYLLABUS

ITMO 441 Network Application and Operations

Hours: 3 credit hours / 45 contact hours or 3 credit hours / 60 contact hours

Instructor: Kevin Vaccaro

Textbook, title, author, and year: *Linux Administration a Beginner's Guide. 7th edition*, Wale Soyinka, 2016.

Specific course information

- a. **Catalog description:** Students learn the details, use, and configuration of network applications. Currently protocols and application technologies considered include SNMP, SMTP, IMAP, POP, MIME, BOOTP, DHCP, SAMBA, NFS, AFS, X, HTTP, DNS, NetBIOS, and CIFS/SMB. Windows workgroups and domains: file and printer sharing, remote access, and Windows networking are addressed. A research paper in the above topic areas is required.
- b. **Prerequisites:** ITMO 340

Specific goals for the course

- a. **Course Outcomes:**
 - Implement various protocol services using the Linux operating system
 - Analyze Internet protocols
 - Understand how data flows and moves thru the internet
 - Configuration and operation of Linux as a server
 - Write and submit an original research paper
- b. **Course student outcomes:**

At the conclusion of this course, each successful student will be able to:

 - Examine the different services available at the Application Level of the TCP/IP model using the Linux operating System
 - Examine each application protocol in detail
 - Use the Linux OS to implement different application services as hands on labs
 - Perform and share original research on a chosen Network related topic

Topics to be covered

- a. Review TCP/IP
- b. DNS
- c. FTP
- d. SMTP/POP/IMAP
- e. HTTP
- f. SSL
- g. SSH/SCP
- h. NFS
- i. Linux Firewall
- j. SAMBA
- k. NIS
- l. LDAP
- m. DHCP

ITMO 444 SYLLABUS

ITMO 444 Cloud Computing Technologies

Hours: 3 credit hours / 45 contact hours

Instructor: Jeremy Hajek

Textbook, title, author, and year: *The Practice of Cloud System Administration: Designing and Operating Large Distributed Systems, Volume 2*, Thomas A. Limoncelli, Strata R. Chalup and Christina J. Hogan, 2014

Specific course information

- a. **Catalog description:** Computing applications hosted on dynamically-scaled virtual resources available as services are considered. Collaborative and non-collaborative “cloud-resident” applications are analyzed with respect to cost, device/location independence, scalability, reliability, security, and sustainability. Commercial and local cloud architectures are examined. A group-based integration of course topics will result in a project employing various cloud computing technologies.
- b. **Prerequisite:** ITMD 321

Specific goals for the course

- a. **Course Outcomes:** Each successful student will be able to demonstrate building and running cloud-based services on a large scale. They will gain the knowledge of deploying and managing elastic and cloud-based applications on industry standard platforms as well as open-source platforms. Students will be prepared with knowledge of Cloud Based Operations and Application Development.
- b. **Course student outcomes:**
 - Explain, document, and implement the fundamental aspects of IaaS, PaaS, SaaS
 - Use and administer industry standard cloud resources
 - Correctly identify cloud native operations and development methodologies
 - Build and deploy elastic scaling applications on a cloud platform
 - Design applications using a cloud native architecture
 - Describe and employ web technologies for software deployment

Topics to be covered

- a. Thinking Cloud
- b. Designing in a Distributed World
- c. Designing for Operations/ Service Platforms
- d. AWS Introduction
- e. History of Cloud Computing
- f. Application Architectures
- g. Design Patterns for Scaling
- h. Design Patterns for Resiliency
- i. Operations in a Distributed World
- j. Cloud Concepts w/AWS
- k. Design Documents & Monitoring

ITMO 446 SYLLABUS

ITMO 446 Telecommunications over Data Networks

Hours: 3 credit hours / 45 contact hours

Instructor: Carol Davids

Textbook, title, author, and year: *SIP: Understanding the Session Initiation Protocol 3rd Edition*, Alan B. Johnston, 2009

Specific course information

- a. **Catalog description:** This course covers a suite of application protocols known as Voice over IP (VoIP). It covers key protocols within that suite, including Session Initiation Protocol (SIP), Real-time Transport Protocol (RTP) and Session Description Protocol (SDP) as well as the architectures of various VoIP installations including on-net to on-net; on-net to PSTN; and inter-domain scenarios. The functions of the Network Elements in these architectures are defined and examples of products that include these network elements are examined. Contrast with circuit-switched and web-based communications systems is provided.
- b. **Prerequisites:** ITMO 340.

Specific goals for the course

- a. **Course Outcome:** The goal of the course is to provide an understanding of how audio and video communications in real-time can be provided over Internet Protocol networks using protocols, including Session Initiation Protocol (SIP), Real-time Transport Protocol (RTP) and Session Description Protocol (SDP) defined by the Internet Engineering Task Force (IETF.) A further goal of this work is to provide an evolutionary perspective on the SIP-based approach contrasting circuit-switched communications and web-based approaches. This organization of the material is designed to foster innovative thinking and development in the field of real-time communications, based on hands-on work and an understanding of past innovation and development. The successful student will have the necessary knowledge and skills to work in the field of IP-based telecommunications at an entry level.
- b. **Course Student Outcomes:** Upon successful completion of the course the student should be able to:
 - Use protocol analysis tools to analyze the message flows between SIP functional elements.
 - Draw message sequence charts to aid in message flow analysis.

- Identify the header fields and parameters that may change as the messages traverse the network.
- Use this message flow analysis to verify correct behavior and to isolate trouble.
- Identify the media streams and signaling messages associated with a SIP call.
- Analyze contents of media streams and signaling messages associated with a SIP call.
- Perform tasks and demonstrate skills necessary to work in the field of IP-based telecommunications at an entry level.

Topics to be covered

- a. Protocol (SDP) and Real-time Transport Protocol (RTP)
- b. SIP functional elements and architecture
- c. SIP message syntax and call flows
- d. SIP relationships – transactions, dialogs and sessions
- e. Voice payload digitization
- f. Codecs
- g. Real-time Transport Protocol - RTP and RTCP
- h. Session Description protocol (SDP)
- i. SIP Methods
- j. SIP Feature Creation
- k. SIP Architectures
- l. Project presentations and demonstrations

ITMO 450 SYLLABUS

ITMO 450 Enterprise End-User System Administration

Hours: 3 credit hours / 45 contact hours

Instructor: Vasilios "Billy" Pappademetriou

Text book, title, author, and year: *Exam Ref 70-698
Installing and Configuring Windows 10 (2nd Edition)*
by Andrew Bettany and Andrew Warren, 2018

Specific course information

- a. **Catalog description:** Students learn to set up, configure, and maintain end-user desktop and portable computers and devices in an enterprise environment using a contemporary proprietary operating system, including the actual installation of the operating system in a networked client-server environment. User account management, security, printing, disk configuration, and backup procedures are addressed with particular attention to coverage of networked applications. System installation, configuration, and administration issues as well as network file systems, network access, and compatibility with other operating systems are also addressed. Administration of central server resources associated with management and provisioning of end-user systems in workgroups, domains, or forests is also addressed.
- b. **Prerequisites:** ITM 301
- c. **Selected Elective.**

Specific goals for the course

- a. **Course Outcomes:**
Upon successful completion of the course the student will gain basic abilities to:
 - Administer End-User Enterprise Servers as a System Administrator
 - Configure server, network, and software applications setup environment
 - Set up a Client Server Environment
 - Demonstrate skills and knowledge of server administrator's responsibilities and support of enterprise applications
 - Provide support planning, development, integration, testing and management of IT services
 - Provision of end-user systems in workgroups, domains
 - Analyze server recurring issues and work on sustainable solutions.
- b. **Course student outcomes:**
At the conclusion of this course, each student should be able to:
 - Install and manage End-User Enterprise Servers
 - Implement Windows in an Enterprise Environment

-
- Manage server, client computers, users, and active directory
- Apply server security, software distribution and updates
- Perform scheduled vulnerability-assessment scans and monitoring server performance
- Design Infrastructure requirements and manage compatibility issues
- Set up and manage server system networking
- Manage storage, file structure, and directory services
- Monitor logs for firewalls and intrusion detection systems

Topics to be covered

- a. Introduction. IIT Banner, Class basics
- b. Preparing for Installation Requirements
- c. Configuring Devices and Device Drivers
- d. Performing Post-Installation Configuration
- e. Implementing Windows in an Enterprise Environment
- f. Configuring Networking
- g. Midterm Presentations / Midterm Exam
- h. Configuring Data Access and Usage
- i. Implementing Apps
- j. Configuring Remote Management
- k. Monitoring Windows
- l. Configuring System and Data Recovery
- m. Configuration Advanced Management Tools

ITMO 453 SYLLABUS

ITMO 453 OSS System Administration

Hours: 3 credit hours / 45 contact hours

Instructor: Jeremy Hajek

Textbook, title, author, and year:

The Art of Monitoring, James Turnbull, 2016

Monitoring with Prometheus, James Turnbull, 2018

Specific course information

- a. **Catalog description:** Students learn the administration topics and concepts of IT Orchestration, Automation, Monitoring, and Metric Collection. Topics include configuring industry standard automation tooling and using scripting to achieve immutable infrastructure. Students will learn how to monitor and collect and present metrics in regards to the infrastructure they deploy.
- b. **Prerequisites:** ITMO 340 and ITMO 356

Specific goals for the course

- a. **Course Outcomes:** This course will enable students to be ready to design, build, and implement logging and metrics in monitored applications. Implementing these foundations will allow any system administrator to integrate logging and metric collection to correlate with business objectives.
- b. **Course student outcomes:**
At the conclusion of this course, each student should be able to:
 - Explain the difference between push and pull metrics
 - Explain the difference between logging and metrics
 - Describe event streams are and how they are used in monitoring and metric collection
 - Explain the use of logging and metrics in regards to Operating System containers
 - Design, build, and implement logging and metrics in monitored applications

Topics to be covered

- n. Intro - Monitoring & Measurement Framework
- o. Managing events and Metrics & Graphing
- p. Event Routing and Collection
- q. Containers and Logs
- r. Building an app & Notifications
- s. Getting Started & Monitoring Nodes
- t. Service Discovery
- u. Alerting & Scaling and Reliability
- v. Instrumenting Applications & Logging
- w. Building Monitored Applications & Notifications

ITMO 454 SYLLABUS

ITMO 454 Operating System Virtualization

Hours: 3 credit hours / 45 contact hours

Instructor: Philip Matuszak

Textbook, title, author, and year: *Virtualization Essentials, Second Edition*, Matthew Portnoy, 2016

Specific course information

- a. **Catalog description:** Each successful student in this course will become familiar with hypervisors, virtualization terms, infrastructure considerations, and appropriate use cases. While designed to give an overview of today's Virtualization technologies and methods, students in the course will gain enough practical knowledge to begin deploying various hypervisors and virtual machine environments using current industry standard platforms.
- b. **Prerequisites:** None.

Specific goals for the course

- a. **Course Outcomes:** This course exposes students to virtualization in an enterprise setting as a tool for the deployment, configuration, and management of server and desktop resources. Students will experience a variety of virtualization environment and products. Students will work with technical implementations of virtualization and learn to design and manage physical to virtual migration.
- b. **Course Student Outcomes:** Students completing this course will be able to:
 - Describe and discuss current trends in Operating System Virtualization by experiencing a variety of applications and software packages.
 - Explain what a hypervisor is, what it does, and the various types involved and when each is used.
 - Demonstrate technical knowledge and limited proficiency in designing and deploying virtualized environments
 - Identify and describe various Virtualization platforms and software such as VMware, XenServer, Hyper-V, Virtual box, and VMware workstation, and open source hypervisors.
 - Create a proposal and design for migrating an existing physical environment to a virtual environment.

Topics to be covered

- a. Introduction to Virtualization & Hypervisors
- b. Type 2 Hypervisors and VMs
- c. Hardware, Infrastructure, & Type 1 HV
- d. VM Creation and Management
- e. CPU, Memory, and Consumables
- f. Storage
- g. Networking
- h. Management
- i. Availability
- j. Virtual Applications
- k. VDI
- l. Security
- m. Backup and Recovery
- n. Open Source / Apple Virtualization

ITMO 463 SYLLABUS

ITMO 463 Cloud: Software as a Service

Hours: 3 credit hours / 45 contact hours

Instructor: TBA

Textbook, title, author, and year

- a. Online readings as assigned in Coursera

Specific course information

- a. **Catalog description:** Software as a Service (SaaS) allows consumers to use a provider's applications running on a cloud infrastructure, accessible from client devices over a network through either a thin client interface, such as a web browser, or a program interface. Students will explore different approaches, techniques, tools and technologies to build, deploy, and manage cloud native applications.
- b. **Prerequisites:** ITMO 444
- c. **Required.**

Specific goals for the course

- a. **Program Educational Outcome:**
 1. Problem solve, create, and effectively communicate innovative answers to provide cloud-based technology solutions for the problems of business, industry, government, non-profit organizations, and individuals
 2. Perform requirements analysis, design and administration of secure cloud-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
 - b. **Course Outcomes:** Each successful student will be able to use concepts of cloud computing and cloud-native applications to design and provision software as a service using standard planning methodologies to meet identified business and/or consumer needs.
 - c. **Course student outcomes:** Upon completion of this course the student should be able to do the following:
 - Define SaaS and compare it to other cloud service models
 - Identify examples of SaaS applications
 - Describe the benefits and challenges of using SaaS
 - Describe the different deployment models for SaaS (public, private, hybrid)
 - Compare the advantages and disadvantages of each deployment model
 - Choose a suitable deployment model for a given scenario
 - Describe the architectural components of a SaaS application
 - Design a basic SaaS application using multi-tenancy principles
 - Implement a simple SaaS application using a cloud platform
 - Identify the key security and compliance considerations for SaaS solutions.
- Describe the measures that SaaS providers take to ensure security
 - Implement security measures to mitigate risks/vulnerabilities for a SaaS application
 - Explain the performance and scalability challenges of SaaS
 - Identify strategies for improving performance and scalability in a SaaS application
 - Analyze the costs and benefits of different SaaS performance and scalability optimization strategies.
 - Implement performance and scalability improvements for a SaaS application
 - Describe the challenges of integrating and interoperating with SaaS applications
 - Identify strategies for integrating and interoperating with SaaS applications
 - Implement integration and interoperability for a SaaS application
 - Describe the challenges of managing and operating SaaS applications
 - Identify strategies for managing and operating SaaS applications
 - Implement management and operations for a SaaS application
 - Explain the different pricing and licensing models for SaaS
 - Compare the advantages and disadvantages of different pricing and licensing models
 - Choose a suitable pricing and licensing model for a given scenario
 - Describe the major SaaS providers and markets
 - Compare the features and services offered by different SaaS providers
 - Evaluate the suitability of different SaaS providers for a given scenario
 - Analyze real-world case studies of SaaS implementations
 - Identify best practices and lessons learned from the case studies
 - Apply the concepts learned to a case study analysis
 - Define DevOps and its principles
 - Describe the benefits of using DevOps in SaaS development and operations
 - Implement DevOps practices for a SaaS application
 - Describe key considerations for testing and quality assurance for SaaS solutions.
 - Compare costs and benefits of different SaaS testing and quality assurance strategies.
 - Implement testing and quality assurance for SaaS solutions.
 - Identify key metrics and indicators for measuring the performance and effectiveness of SaaS solutions.
 - Analyze the costs and benefits of different SaaS metrics and monitoring strategies.
 - Develop a plan to monitor and measure SaaS solutions.

- Describe the current state and future trends of SaaS and cloud computing
 - Analyze the potential impact of these trends on the industry
 - Evaluate the opportunities and challenges that these trends present for SaaS developers and users

Topics to be covered

- a. Introduction to Software as a Service (SaaS)
- b. Deployment Models
- c. Architecture and Design
- d. Security
- e. Performance and Scalability
- f. Integration and Interoperability
- g. Management and Operations
- h. Pricing and Licensing
- i. Providers and Markets
- j. Case Studies
- k. DevOps for Software as a Service
- l. Testing and Quality Assurance
- m. Metrics and Monitoring
- n. Software as a Service and the Future of Cloud Computing
- o. Project Presentations and Final Exam Review

ITMO 464 SYLLABUS

ITMO 464 Cloud: Platform as a Service

Hours: 3 credit hours / 45 contact hours

Instructor: TBA

Textbook, title, author, and year

a. Online readings as assigned in Coursera

Specific course information

a. **Catalog description:** Platform as a Service (PaaS) allows developers to deploy onto the cloud infrastructure developer-created or acquired applications created using programming languages, libraries, services, and tools supported by the cloud provider. Students learn to develop applications and services using popular platforms and service tools, and to manage deployed applications as well as configuration settings for the application-hosting environment.

b. **Prerequisites:** ITMO 444

c. **Required.**

Specific goals for the course

a. **Program Educational Outcome:**

1. Problem solve, create, and effectively communicate innovative answers to provide cloud-based technology solutions for the problems of business, industry, government, non-profit organizations, and individuals
2. Perform requirements analysis, design and administration of secure cloud-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.

b. **Course Outcomes:** Each successful student will be able to use concepts of cloud computing and cloud-native applications to design and provision platform as a service using standard planning methodologies to meet identified business and/or consumer needs.

c. **Course student outcomes:**

Upon completion of this course the student should be able to do the following:

- Define Platform as a Service (PaaS)
 - Compare and contrast PaaS with Infrastructure as a Service (IaaS) and Software as a Service (SaaS)
 - Describe the benefits and drawbacks of using PaaS
 - List common PaaS providers and their offerings
- Explain the concept of containers and containerization
 - Describe the benefits of using containers in cloud computing
 - Compare and contrast containers with virtual machines
 - Create and run a container using Docker
- Describe the challenges of processing big data and implementing machine learning in the cloud
 - Describe the PaaS solutions for big data processing and machine learning

- Use a PaaS provider to process big data or to implement a machine learning model
- Describe the challenges of developing and deploying mobile applications in the cloud
 - Describe the PaaS solutions for mobile application development and deployment
 - Use a PaaS provider to develop and deploy a mobile application
- Define serverless computing and its benefits
 - Compare and contrast serverless computing with traditional computing models
 - Design and implement a serverless application using a PaaS provider
- Define DevOps and its principles and describe how they apply to PaaS
 - Describe the benefits of adopting a DevOps culture
 - Describe the benefits of using PaaS for DevOps
 - Implement a continuous integration and delivery pipeline using a PaaS provider
- Explain the shared responsibility model for cloud security
 - Identify common threats and vulnerabilities in cloud environments
 - Describe the security features and controls provided by PaaS providers
 - Implement security measures, such as access controls and encryption, for a PaaS environment
- Compare and contrast the offerings of different PaaS providers, including Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform
 - Choose a suitable PaaS provider for a given scenario based on requirements and constraints
 - Migrate an application from one PaaS provider to another
- Describe the architecture and components of Cloud Foundry
 - Deploy and manage applications on Cloud Foundry
 - Integrate Cloud Foundry with external services and tools
- Describe the features and capabilities of Heroku as a PaaS provider
 - Deploy and manage applications on Heroku
 - Use Heroku add-ons and buildpacks to extend the functionality of applications
- Describe the features and capabilities of Azure App Service as a PaaS provider
 - Deploy and manage applications on Azure App Service
 - Use Azure App Service features, such as deployment slots and auto-scaling, to improve the reliability and scalability of applications
- Describe the features and capabilities of Google App Engine as a PaaS provider
 - Deploy and manage applications on Google App Engine

- Use Google App Engine features, such as traffic splitting and versioning, to improve the reliability and scalability of applications
- Identify best practices for designing and deploying applications on a PaaS platform
 - Implement best practices, such as design patterns and architecture principles, in the development of a PaaS application
 - Evaluate the effectiveness of best practices in improving the reliability, scalability, and maintainability of a PaaS application
- Implement monitoring and logging for a PaaS application
 - Use monitoring and logging data to identify and troubleshoot issues in a PaaS application
 - Implement debugging techniques, such as remote debugging and error handling, for a PaaS applications

Topics to be covered

- a. Introduction to Platform as a Service
- b. Containers and Containerization
- c. Platform as a Service for Big Data and Machine Learning
- d. Platform as a Service for Mobile Applications
- e. Serverless Computing
- f. DevOps and Platform as a Service
- g. Security
- h. Platform as a Service Providers
- i. Use PaaS: Cloud Foundry
- j. Use PaaS: Heroku
- k. Use PaaS: Azure App Service
- l. Use PaaS: Google App Engine
- m. Best Practices
- n. Monitoring and Debugging
- o. Course Review and Final Project

ITMO 465 SYLLABUS

ITMO 465 Cloud: Infrastructure as a Service

Hours: 3 credit hours / 45 contact hours

Instructor: TBA

Textbook, title, author, and year

a. Online readings as assigned in Coursera

Specific course information

a. **Catalog description:** Infrastructure as a Service (SaaS) allows users to provision processing, storage, networks, and other fundamental computing resources which then allows them to deploy and run arbitrary software, which can include operating systems and applications. Students will learn how to provision, deploy and manage operating systems, storage, and deployed applications as well as virtual networking components such as switches, routers, and firewalls in a cloud environment accessible remotely through a network.

b. **Prerequisites:** ITMO 444

c. **Required.**

Specific goals for the course

a. **Program Educational Outcome:**

1. Problem solve, create, and effectively communicate innovative answers to provide cloud-based technology solutions for the problems of business, industry, government, non-profit organizations, and individuals
2. Perform requirements analysis, design and administration of secure cloud-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.

b. **Course Outcomes:** Each successful student will be able to use concepts of cloud computing and cloud-native applications to provision, deploy and manage operating systems, storage, and deployed applications as well as virtual networking components such as switches, routers, and firewalls in a cloud environment accessible remotely through a network, using standard planning methodologies to meet identified business and/or consumer needs.

c. **Course student outcomes:**

Upon completion of this course the student should be able to do the following:

- Define the term "Infrastructure as a Service" and distinguish it from other forms of cloud computing.
 - Describe the main features and benefits of IaaS.
 - Identify the key components of an IaaS architecture and their functions.
 - List the main providers of IaaS and compare their offerings.
 - Compare and contrast the various IaaS providers (e.g. AWS, Azure, Google Cloud)
- Explain the concept of virtualization and how it relates to IaaS

- Explain what containers are and how they differ from virtual machines.
- Compare and contrast different containerization technologies, such as Docker and Kubernetes.
- Deploy and manage a simple containerized application using a container orchestration platform (e.g. Docker, Kubernetes)
- Explain the networking concepts and abstractions that are relevant to IaaS. (e.g. VPCs, subnets, security groups)
 - Describe the main networking options and configurations that are available in IaaS.
 - Configure network resources and connectivity in a cloud environment
 - Use load balancers and DNS to scale and secure applications in the cloud
- Explain the storage options and abstractions that are available in IaaS. (e.g. object storage, block storage, file storage)
 - Compare and contrast the different storage options in terms of cost and performance
 - Use cloud storage APIs to store and retrieve data in a cloud environment
 - Configure storage for a cloud-based application.
- Describe the principles of DevOps and the benefits of a DevOps culture
 - Use version control systems (e.g. Git) to manage code changes
 - Implement continuous integration/delivery pipelines using tools such as Jenkins
- Describe factors to consider when planning a cloud migration
 - Use tools and techniques to migrate data and applications to the cloud
 - Test and validate the migration to ensure it is successful
- Describe the various pricing models for cloud services
 - Use cost optimization techniques to reduce cloud costs
 - Use cost management tools to track and forecast cloud expenses
- Describe the importance of monitoring and logging in the context of IaaS.
 - Describe the main types of metrics and logs that are relevant to IaaS.
 - Set up and configure monitoring and logging for a cloud-based system.
 - Use log analysis tools to troubleshoot issues in the cloud
- Describe the shared responsibility model for security in the cloud
 - Use security best practices to secure cloud resources and applications
 - Describe the main security controls that are available in IaaS, such as security groups and encryption.
 - Implement basic security measures for a cloud-based system.

- Implement identity and access management (IAM) in the cloud
- Describe the importance of disaster recovery and business continuity in the cloud
 - Implement disaster recovery strategies for cloud applications and data
 - Test disaster recovery plans to ensure they are effective
- Explain what serverless computing is and how it differs from traditional IaaS.
 - Describe the main features and benefits of serverless computing.
 - Use serverless platforms (e.g. AWS Lambda, Azure Functions) to build and deploy applications
 - Implement a simple serverless application using a cloud platform.
- Describe the principles of cloud-native design and how it differs from traditional application design
 - Explain what microservices are and how they differ from monolithic architectures.
 - Explain what event-driven architectures are and how they differ from traditional architectures.
 - Use cloud-native technologies (e.g. microservices, containers) to build cloud-native applications.
- Describe the various tools available for managing cloud environments
 - Use cloud management platforms (e.g. AWS CloudFormation, Azure Resource Manager) to automate cloud resource provisioning and management
 - Use tools for tracking and optimizing cloud usage and costs
- Analyze real-world examples of companies using IaaS to solve business problems
 - Evaluate the benefits and challenges of the IaaS solutions used in the case studies
 - Reflect on the key takeaways from the case studies and how they can be applied in a professional setting

Topics to be covered

- a. Introduction to Infrastructure as a Service
- b. Virtualization and Containers
- c. Networking in the Cloud
- d. Storage in the Cloud
- e. DevOps and Continuous Integration/Delivery
- f. Cloud Migration
- g. Cloud Cost Management
- h. Monitoring and Logging
- i. Security in the Cloud
- j. Disaster Recovery and Business Continuity
- k. Serverless Computing
- l. Cloud-Native Applications
- m. Cloud Management Tools
- n. Industry Case Studies
- o. Cloud Computing Capstone Project

ITMO 503 SYLLABUS

ITMO 503 Hardware and Operating System Foundations

Hours: 3 credit hours / 45 contact hours

Instructor: Vasilios "Billy" Pappademetriou

Textbook, title, author, and year:

a. Online readings as assigned in Coursera

Specific course information

- a. **Catalog description:** Explores the basics of computer architecture and use of contemporary operating systems and networking. Covers hardware requirements, components, software compatibility, and system installation topics as well as other key operating systems functions. Networking, virtualization, cloud computing, and security concepts are introduced. Introduces features of an advanced operating system, including basic commands, file and directory manipulation, security, and suitability for server applications. Popular and business-focused desktop and mobile device operating systems will be examined, as well as enterprise and open-source server implementations.
- b. **Prerequisites:** None

Specific goals for the course

- a. **Program Educational Objectives**
2. Manage and deploy secure cloud-based information resources in an enterprise setting..
- b. **Course Outcomes:**
This is a foundation course in the basics of PC functioning from a hardware level to the upper level operating system. Upon completion, a student should be able to understand how a PC functions, be able to troubleshoot and repair a PC, and understand its workings in a networked environment from a hardware level to OS level. Students recall and employ concepts of operating systems including Windows, MacOS, Linux; networking; virtualization; servers; security concepts; and tools for managing IT. Students will be able to understand, use, and manage industry-standard operating systems.
- c. **Course student outcomes:**
Upon completion of this course the student should be able to do the following
- Recall and describe the components of a computer
 - Explain the functioning of processors
 - Describe the purpose and operation of motherboards, buses, architecture, and memory
 - Explain the role and operation of storage, monitors, and other peripherals
 - Assemble a computer from a set of components
 - Demonstrate methods for troubleshooting hardware

- Describe fundamental concepts of networking including physical media, devices, protocols, and standards
 - Explain and demonstrate installation and configuration of a small office/ home office network
- Explain & demonstrate installation & basic configuration of an operating system (OS)
 - Administer operating systems at a basic level
 - Use operating system utilities
- Describe cloud computing concepts
- Describe and employ operating system virtualization
- Operating Systems
 - Compare and contrast common operating systems and their purposes.
 - Compare and contrast features of Microsoft Windows versions.
 - Summarize general OS installation considerations and upgrade methods.
 - Use appropriate Microsoft command line & GUI tools, features and Control Panel utilities.
 - Install and configure applications.
 - Configure Microsoft Windows networking on a client/desktop system.
 - Use features and tools of Mac OS and Linux client/desktop operating systems.
- Security
 - Summarize the importance of physical security measures.
 - Explain logical security concepts.
 - Compare and contrast wireless security protocols and authentication methods.
 - Detect, remove, and prevent malware using appropriate tools and methods.
 - Compare and contrast social engineering, threats, and vulnerabilities.
 - Compare and contrast the differences in basic Microsoft Windows OS security settings
 - Implement security best practices to secure a workstation, mobile device, SOHO wired or wireless devices.
 - Implement appropriate data destruction and disposal methods.
- Software Troubleshooting
 - Troubleshoot:
 - Microsoft Windows OS and resolve PC Security issues.
 - Mobile OS and Application issues.
 - Mobile OS and Application Security Issues.
 - Use best practices for malware removal.
- Operational Procedures
 - Compare and contrast best practices associated with types of documentation.
 - Implement:
 - Change management best practices.
 - Basic disaster prevention and recovery methods.

- Explain:
 - Common safety procedures/practices.
 - Environmental impacts and appropriate controls.
- Describe processes for addressing prohibited content/activity, and privacy, licensing, and policy concepts.
- Use proper communication techniques and professionalism.
- Identify the basics of scripting.
- Use remote access technologies

Topics to be covered

- a. Introduction. IIT Banner, Class basics, intro to the Python IDE
- b. Introduction to Virtualization
- c. Operating System Basics
- d. Intro to Hardware
 - i. Motherboards, Processors, and Memory
 - ii. Internal Expansion, Storage Devices, and Power Supplies
 - iii. Peripherals, Connectors, and Printers
- e. Custom System Configurations
- f. Operating System Administration
 - i. Working with Windows Versions
 - ii. Maintain and Troubleshoot MS Windows
 - iii. Working with macOS and Linux
- g. Support Operating Systems
- h. Install, Configure, and Maintain an OS
- i. Manage Users, Workstations & Shared Resources
- j. Networking Fundamentals and TCP/IP
 - i. Installing Wireless and SOHO Networks
 - ii. Configure and Troubleshoot Networks
- k. Network Services, enterprise virtualization, and Cloud Computing
- l. Security Concepts
 - i. Securing Operating Systems
- m. Scripting and Remote Access
- n. Working with Laptops and Mobile Devices
- o. Troubleshooting Methodologies
 - i. Hardware and Network Troubleshooting
 - ii. Troubleshooting Operating Systems and Security
- p. Support Operating Systems
- q. Security Concepts
- r. Secure Workstation and Data
- s. Troubleshoot Workstation Security Issues
- t. Support and Troubleshoot Mobile Devices
- u. Implement Operational Procedures
- v. Documentation and Professionalism

ITMO 563 SYLLABUS

ITMO 563 Cloud: Software as a Service

Hours: 3 credit hours / 45 contact hours

Instructor: TBA

Textbook, title, author, and year

a. Online readings as assigned in Coursera

Specific course information

a. **Catalog description:** Software as a Service (SaaS) allows consumers to use a provider's applications running on a cloud infrastructure, accessible from client devices over a network through either a thin client interface, such as a web browser, or a program interface. Students will explore different approaches, techniques, tools and technologies to build, deploy, and manage cloud native applications.

b. **Prerequisites:** ITMO 544

c. **Required.**

Specific goals for the course

a. Program Educational Outcome:

1. Deliver optimal technical and policy cloud computing solutions for the problems of business, industry, government, non-profit organizations, and individuals
2. Manage and deploy cloud-based information resources in an enterprise setting.

b. **Course Outcomes:** Each successful student will be able to effectively use concepts of cloud computing and cloud-native applications to design and provision software as a service using standard planning methodologies to meet identified business and/or consumer needs.

c. Course student outcomes:

Upon completion of this course the student should be able to do the following:

- Define SaaS and compare it to other cloud service models
 - Identify examples of SaaS applications
 - Explain the benefits and challenges of using SaaS
- Describe the different deployment models for SaaS (public, private, hybrid)
 - Compare the advantages and disadvantages of each deployment model
 - Choose a suitable deployment model for a given scenario
- Describe the architectural components of a SaaS application
 - Design a basic SaaS application using multi-tenancy principles
 - Implement a simple SaaS application using a cloud platform
- Identify key security and compliance considerations for SaaS solutions.
 - Describe the measures that SaaS providers take to ensure security
 - Implement security measures to mitigate risks/vulnerabilities for a SaaS application

- Explain the performance and scalability challenges of SaaS
 - Identify strategies for improving performance and scalability in a SaaS application
 - Analyze the costs and benefits of different SaaS performance and scalability optimization strategies.
 - Implement performance and scalability improvements for a SaaS application
- Explain the challenges of integrating and interoperating with SaaS applications
 - Identify strategies for integrating and interoperating with SaaS applications
 - Implement integration and interoperability for a SaaS application
- Describe the challenges of managing and operating SaaS applications
 - Identify strategies for managing and operating SaaS applications
 - Implement management and operations for a SaaS application
- Explain the different pricing and licensing models for SaaS
 - Compare the advantages and disadvantages of different pricing and licensing models
 - Choose a suitable pricing and licensing model for a given scenario
- Recall and describe the major SaaS providers and markets
 - Compare the features and services offered by different SaaS providers
 - Evaluate the suitability of different SaaS providers for a given scenario
 - Select an appropriate provider based on the given scenario
- Analyze real-world case studies of SaaS implementations
 - Identify best practices and lessons learned from the case studies
 - Apply the concepts learned to a case study analysis
- Define DevOps and its principles
 - Describe the benefits of using DevOps in SaaS development and operations
 - Implement DevOps practices for a SaaS application
- Describe key considerations for testing and quality assurance for SaaS solutions.
 - Compare costs and benefits of different SaaS testing and quality assurance strategies.
 - Implement testing and quality assurance for SaaS solutions.
- Identify key metrics and indicators for measuring the performance and effectiveness of SaaS solutions.
 - Analyze the costs and benefits of different SaaS metrics and monitoring strategies.
 - Develop a plan to monitor and measure SaaS solutions.

- Describe the current state and future trends of SaaS and cloud computing
 - Analyze the potential impact of these trends on the industry
 - Evaluate the opportunities and challenges that these trends present for SaaS developers and users

Topics to be covered

- a. Introduction to Software as a Service (SaaS)
- b. Deployment Models
- c. Architecture and Design
- d. Security
- e. Performance and Scalability
- f. Integration and Interoperability
- g. Management and Operations
- h. Pricing and Licensing
- i. Providers and Markets
- j. Case Studies
- k. DevOps for Software as a Service
- l. Testing and Quality Assurance
- m. Metrics and Monitoring
- n. Software as a Service and the Future of Cloud Computing
- o. Project Presentations and Final Exam Review

ITMO 564 SYLLABUS

ITMO 564 Cloud: Platform as a Service

Hours: 3 credit hours / 45 contact hours

Instructor: TBA

Textbook, title, author, and year

a. Online readings as assigned in Coursera

Specific course information

- a. **Catalog description:** Platform as a Service (PaaS) allows developers to deploy onto the cloud infrastructure developer-created or acquired applications created using programming languages, libraries, services, and tools supported by the cloud provider. Students learn to develop applications and services using popular platforms and service tools, and to manage deployed applications as well as configuration settings for the application-hosting environment.
- b. **Prerequisites:** ITMO 544
- c. **Required.**

Specific goals for the course

- a. **Program Educational Outcome:**
 1. Deliver optimal technical and policy cloud computing solutions for the problems of business, industry, government, non-profit organizations, and individuals
 2. Manage and deploy cloud-based information resources in an enterprise setting.
- b. **Course Outcomes:** Each successful student will be able to effectively use concepts of cloud computing and cloud-native applications to design and provision platform as a service using standard planning methodologies to meet identified business and/or consumer needs.
- c. **Course student outcomes:** Upon completion of this course the student should be able to do the following:
 - Define Platform as a Service (PaaS)
 - Compare and contrast PaaS with Infrastructure as a Service (IaaS) and Software as a Service (SaaS)
 - Explain the benefits and drawbacks of using PaaS
 - List common PaaS providers and their offerings
 - Explain the concept of containers and containerization
 - Explain the benefits of using containers in cloud computing
 - Compare and contrast containers with virtual machines
 - Create and run a container using Docker
 - Describe the challenges of processing big data in the cloud
 - Describe the challenges of implementing machine learning in the cloud
 - Describe the PaaS solutions for big data processing and machine learning

- Use a PaaS provider to process big data or to implement a machine learning model
- Describe the challenges of developing and deploying mobile applications in the cloud
 - Describe the PaaS solutions for mobile application development and deployment
 - Use a PaaS provider to develop and deploy a mobile application
- Define serverless computing and its benefits
 - Compare and contrast serverless computing with traditional computing models
 - Design and implement a serverless application using a PaaS provider
- Define DevOps and its principles and describe how they apply to PaaS
 - Explain the benefits of adopting a DevOps culture
 - Describe the benefits of using PaaS for DevOps
 - Implement a continuous integration and delivery pipeline using a PaaS provider
- Explain the shared responsibility model for cloud security
 - Identify common threats and vulnerabilities in cloud environments
 - Describe and critique the security features and controls provided by PaaS providers
 - Implement security measures, such as access controls and encryption, for a PaaS environment
- Compare and contrast the offerings of different PaaS providers, including Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform
 - Choose a suitable PaaS provider for a given scenario based on requirements and constraints
 - Migrate an application from one PaaS provider to another
- Describe the architecture and components of Cloud Foundry
 - Deploy and manage applications on Cloud Foundry
 - Integrate Cloud Foundry with external services and tools
- Describe the features and capabilities of Heroku as a PaaS provider
 - Deploy and manage applications on Heroku
 - Use Heroku add-ons and buildpacks to extend the functionality of applications
- Describe the features and capabilities of Azure App Service as a PaaS provider
 - Deploy and manage applications on Azure App Service
 - Use Azure App Service features, such as deployment slots and auto-scaling, to improve the reliability and scalability of applications

- Describe the features and capabilities of Google App Engine as a PaaS provider
 - Deploy and manage applications on Google App Engine
 - Use Google App Engine features, such as traffic splitting and versioning, to improve the reliability and scalability of applications
- Identify best practices for designing and deploying applications on a PaaS platform
 - Implement best practices, such as design patterns and architecture principles, in the development of a PaaS application
 - Evaluate the effectiveness of best practices in improving the reliability, scalability, and maintainability of a PaaS application
- Implement monitoring and logging for a PaaS application
 - Use monitoring and logging data to identify and troubleshoot issues in a PaaS application
 - Implement debugging techniques, such as remote debugging and error handling, for a PaaS applications

Topics to be covered

- a. Introduction to Platform as a Service
- b. Containers and Containerization
- c. Platform as a Service for Big Data and Machine Learning
- d. Platform as a Service for Mobile Applications
- e. Serverless Computing
- f. DevOps and Platform as a Service
- g. Security
- h. Platform as a Service Providers
- i. Use PaaS: Cloud Foundry
- j. Use PaaS: Heroku
- k. Use PaaS: Azure App Service
- l. Use PaaS: Google App Engine
- m. Best Practices
- n. Monitoring and Debugging
- o. Course Review and Final Project

ITMO 565 SYLLABUS

ITMO 565 Cloud: Infrastructure as a Service

Hours: 3 credit hours / 45 contact hours

Instructor: TBA

Textbook, title, author, and year

a. Online readings as assigned in Coursera

Specific course information

a. **Catalog description:** Infrastructure as a Service (SaaS) allows users to provision processing, storage, networks, and other fundamental computing resources which then allows them to deploy and run arbitrary software, which can include operating systems and applications. Students will learn how to provision, deploy and manage operating systems, storage, and deployed applications as well as virtual networking components such as switches, routers, and firewalls in a cloud environment accessible remotely through a network.

b. **Prerequisites:** ITMO 544

c. **Required.**

Specific goals for the course

a. Program Educational Outcome:

1. Deliver optimal technical and policy cloud computing solutions for the problems of business, industry, government, non-profit organizations, and individuals
2. Manage and deploy cloud-based information resources in an enterprise setting.

b. **Course Outcomes:** Each successful student will be able to use concepts of cloud computing and cloud-native applications to provision, deploy and manage operating systems, storage, and deployed applications as well as virtual networking components such as switches, routers, and firewalls in a cloud environment accessible remotely through a network, using standard planning methodologies to meet identified business and/or consumer needs.

c. Course student outcomes:

Upon completion of this course the student should be able to do the following:

- Define the term "Infrastructure as a Service" and distinguish it from other forms of cloud computing.
 - Describe the main features and benefits of IaaS.
 - Identify key components of an IaaS architecture and explain their functions.
 - List the main providers of IaaS and compare their offerings.
 - Compare and contrast the various IaaS providers (e.g. AWS, Azure, Google Cloud)
- Explain the concept of virtualization and how it relates to IaaS
 - Explain what containers are and how they differ from virtual machines.
 - Compare and contrast different containerization technologies, such as Docker and Kubernetes.

- Deploy and manage a simple containerized application using a container orchestration platform (e.g. Docker, Kubernetes)
- Explain the networking concepts and abstractions that are relevant to IaaS. (e.g. VPCs, subnets, security groups)
 - Describe the main networking options and configurations that are available in IaaS.
 - Configure network resources and connectivity in a cloud environment
 - Use load balancers and DNS to scale and secure applications in the cloud
- Explain storage options and abstractions that are available in IaaS. (e.g. object storage, block storage, file storage)
 - Compare and contrast the different storage options in terms of cost and performance
 - Use cloud storage APIs to store and retrieve data in a cloud environment
 - Configure storage for a cloud-based application.
- Explain the principles of DevOps and the benefits of a DevOps culture
 - Use version control systems (e.g. Git) to manage code changes
 - Implement continuous integration/delivery pipelines using tools such as Jenkins
- Describe factors to consider when planning a cloud migration
 - Use tools and techniques to migrate data and applications to the cloud
 - Test and validate the migration to ensure it is successful
- Explain the various pricing models for cloud services
 - Use cost optimization techniques to reduce cloud costs
 - Use cost management tools to track and forecast cloud expenses
- Describe the importance of monitoring and logging in the context of IaaS.
 - Explain what main types of metrics and logs are relevant to IaaS.
 - Set up and configure monitoring and logging for a cloud-based system.
 - Use log analysis tools to troubleshoot issues in the cloud
- Explain the shared responsibility model for security in the cloud
 - Use security best practices to secure cloud resources and applications
 - Describe the main security controls that are available in IaaS, such as security groups and encryption.
 - Implement basic security measures for a cloud-based system.
 - Implement identity and access management (IAM) in the cloud
- Explain the importance of disaster recovery and business continuity in the cloud
 - Implement disaster recovery strategies for cloud applications and data

- Test disaster recovery plans to ensure they are effective
- Explain what serverless computing is and how it differs from traditional IaaS.
 - Describe the main features and benefits of serverless computing.
 - Use serverless platforms (e.g. AWS Lambda, Azure Functions) to build and deploy applications
 - Implement a simple serverless application using a cloud platform.
- Explain principles of cloud-native design and how it differs from traditional application design
 - Explain what microservices are and how they differ from monolithic architectures.
 - Explain what event-driven architectures are and how they differ from traditional architectures.
 - Use cloud-native technologies (e.g. micro-services, containers) to build cloud-native applications.
- Recall and describe the various tools available for managing cloud environments
 - Use cloud management platforms (e.g. AWS CloudFormation, Azure Resource Manager) to automate cloud resource provisioning and management
 - Use tools for tracking and optimizing cloud usage and costs
- Analyze real-world examples of companies using IaaS to solve business problems
 - Evaluate the benefits and challenges of the IaaS solutions used in the case studies
 - Reflect on the key takeaways from the case studies and how they can be applied in a professional setting

Topics to be covered

- a. Introduction to Infrastructure as a Service
- b. Virtualization and Containers
- c. Networking in the Cloud
- d. Storage in the Cloud
- e. DevOps and Continuous Integration/Delivery
- f. Cloud Migration
- g. Cloud Cost Management
- h. Monitoring and Logging
- i. Security in the Cloud
- j. Disaster Recovery and Business Continuity
- k. Serverless Computing
- l. Cloud-Native Applications
- m. Cloud Management Tools
- n. Industry Case Studies
- o. Cloud Computing Capstone Project

ITMS 400 SYLLABUS

ITMS 400 Technology Fundamentals for Cybersecurity

Hours: 3 credit hours / 45 contact hours

Instructor: TBD

Textbook, title, author, and year:

Information Technology: An Introduction for Today's Digital World, Richard Fox 2021

A Byte of Python, C.H. Swaroop,, N.D.

Database Design – 2nd Edition, Adrienne Watt 2018

Additional online resources

Specific course information

- a. **Catalog description:** Hardware and operating system fundamentals; programming basics in Python; introduction to databases, data management, and SQL; networking and data communications fundamentals. Basic cybersecurity concepts are introduced as each technology area is addressed. This course is not available to undergraduates in the Department of Information Technology and Management.
- b. **Prerequisites:** None
- c. **Optional.**

Specific goals for the course

- f. **Program Educational Objectives**
 2. Perform requirements analysis, design, and administration of secure computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
- g. **Course Outcomes:**
This course is a complete examination of fundamental technologies required to construct and operate information systems to facilitate the operation of a modern business enterprise, with an introduction to basic cybersecurity concepts used in each area. The course design will allow students an opportunity to use each of the technologies in hands-on exercises. This course will impart the necessary foundational knowledge for later courses in cybersecurity.
- e. **Course student outcomes:**
Upon completion of this course the student should be able to do the following
 - Recall and describe the components of a computer
 - Explain the functioning of processors
 - Describe the purpose and operation of motherboards, buses, architecture, and memory

- Describe the purpose and basic functions of an operating system
 - Explain operating system virtualization
 - Install and configure a contemporary operating system in a virtual machine
- Explain fundamental concepts of programming
 - Describe object orientation and explain the value and uses of object orientation
 - Create and run programs in an object-oriented programming language
- Describe fundamental concepts of networking and data communications including physical media, devices, protocols, and standards
 - Configure a computing device to connect to a network
- Describe basic functionality of the World Wide Web
- Describe cloud computing concepts
- Recall and describe fundamental concepts of relational database systems, including data modeling, the entity relationship data model, normalization, and query languages
 - Create and manipulate a relational database in MySQL

2. Topics to be covered
 - a. Introduction to Data, Computer, and Network Technologies
 - b. CPU and Memory
 - c. Operating systems
 - d. OS Installation and configuration
 - e. Software
 - f. Programming basics
 - g. Operators, expressions, functions
 - h. Data structures and I/O
 - i. Object orientation
 - j. Network Infrastructure Concepts
 - k. The Internet and internetworking
 - l. Network configuration
 - m. The Web and the Cloud
 - n. Introduction to Databases, data modeling and the relational database model
 - o. Entity relationship models and normalization
 - p. Structured Query Language and using MySQL

ITMS 418 SYLLABUS

ITMS 418 Coding Security

Hours: 3 credit hours / 45 contact hours

Instructor: Bonnie A. Goins

Textbook, title, author, and year: Online publications from The Open Web Application Security Project (OWASP) at <https://owasp.org/>

- d. Software Assurance Maturity Model (SAMM) – Intro
- e. SAMM – Business Functions
- f. SAMM – Security Practices
- g. Security Practices
- h. SAMM – Security Practices
- i. Application Security Assessment
- j. Application Vulnerability Management

Specific course information

- a. **Catalog description:** In-depth examination of topics in application security program development, stakeholder engagement, software assurance maturity measurement, identification and treatment of software vulnerabilities and implementation of a formal vulnerability management program. Homework is required for this course.
- b. **Prerequisites:** ITMD 411.
- c. **Required for Applied Cybersecurity and Information Technology.**

Specific goals for the course

- a. **Course outcomes:** Each successful student will demonstrate foundation knowledge of application security concepts and best practices. Students will describe and identify application security vulnerabilities and weaknesses, how to assess for them in an environment, how to treat these vulnerabilities and how to respond to incidents involving coding issues.
- b. **Course student outcomes:**
At the conclusion of this course each student should be able to:
 - Explain the concept of application security
 - Identify and describe the OWASP Top Ten application vulnerabilities
 - Recall and describe the secure software/system development lifecycle
 - Create, implement, and maintain a formal Application Security Program
 - Describe how a formal program assists the CISO and other business stakeholders in maintaining a robust security program
 - Describe the application security assessment process
 - Write an appropriate application security report

Topics to be covered

- a. Introduction to Application Security
- b. Building an Application Security Program – Day 1
- c. Building an Application Security Program – Day 2

ITMS 428 SYLLABUS

ITMS 428 Database Security

Hours: 3 credit hours / 45 contact hours

Instructor: Kevin Vaccaro

Textbook, title, author, and year: *Database Security*,
Alfred Basta & Melissa Zgola, 2011.

Specific course information

- a. **Catalog description:** Students will engage in an in-depth examination of topics in data security, including security considerations in applications and systems development, encryption methods, Cryptography and security architecture models, policy, testing, and auditing.
- b. **Prerequisites:** ITMD 321.

Specific goals for the course

- a. **Course Outcomes:** Each student will learn the fundamentals of database security as well as concepts and technologies such as encapsulation (information hiding) and using relational database security management techniques. They will be conversant with database hardening on a variety of platforms, defense against the most common threats and attacks, and the legal and regulatory environment affecting database security.
- b. **Course student outcomes:**
At the conclusion of this course, each successful student will be able to:
 - Recall and describe concepts of information security
 - Describe and explain security architectures for protection of database resources
 - Secure and harden database deployments using leading industry-standard database management systems
 - Recall and describe access control approaches, including authentication, authorization, privileges and roles
 - Discuss cryptography and encryption
 - Identify elements of a cryptographic system
 - Describe how crypto can be used, strengths and weaknesses, modes, and issues that must be addressed in an implementation
 - Describe the technical details of SQL injection attacks
 - Explain how to protect against SQL injection attacks
 - Discuss issues and recall techniques and best practices in the protection of Big Data and data in the cloud
 - Describe and discuss the processes of auditing and testing database security
 - Describe and understand NoSQL and different types of NoSQL

Topics to be covered

- a. Security and Information Technology Security and Information Technology Operating System Best Practices and Review / Virtual Machines Database Review
- b. Database Hardening: MySQL Database Hardening: SQL Server
- c. Database Hardening: PostgreSQL
- d. Cloud Databases / NoSQL / Other DB Types
- e. SQL Injection – Identification and Procedure
- f. Passwords, Profiles, Privileges, and Roles Encryption Policy, Documents, and Education Database Security Auditing
- g. Security and System Testing

ITMS 438 SYLLABUS

ITMS 438 Cyber Forensics

Hours: 3 credit hours / 45 contact hours

Instructor: William Lidinsky

Textbook, title, author, and year:

Guide to Computer Forensics and Investigations, B.

Nelson, A. Phillips, C. Steuart, 2019

File System Forensic Analysis, B. Carrier, 2005

Specific course information

- a. **Catalog description:** This course will address methods to properly conduct a computer and/or network forensics investigation including digital evidence collection and evaluation and legal issues involved in network forensics. Technical issues in acquiring court admissible chains-of-evidence using various forensic tools that reconstruct criminally liable actions at the physical and logical levels are also addressed. Technical topics covered include detailed analysis of hard disks, files systems (including FAT, NTFS and EXT) and removable storage media; mechanisms for hiding and detecting hidden information; and the hands-on use of powerful forensic analysis tools.
- b. **Prerequisites:** ITMS 448 and ITMO 356
- c. **Required for Applied Cybersecurity and Information Technology.**

Specific goals for the course

- a. **Course Outcomes:**
 - Demonstrate knowledge of cyber forensic analysis at levels ranging from professional to executive levels including applicable legal issues.
 - Apply this knowledge to planning and executing specific cyber forensic analyses. This includes the use of cyber forensic tools.
 - Demonstrate knowledge of steganography and steganalysis and apply it to determination of existence of covert information.
- c. **Course student outcomes:**
At the conclusion of this course, each student should be able to:
 - Demonstrate knowledge of cyber forensic procedures, planning of analyses and the use of common tools for analysis
 - Describe several file systems including FAT, EXT, YAFFS and NTFS.
 - Describe several common booting procedures.
 - Describe how to find file system objects that have been deleted or obfuscated.

- Describe how to track past computer and Internet activity and to establish time lines for this activity.
- Describe techniques for inserting covert information in various text, document and image carrier files.
- Demonstrate the ability to use tools such as WinHex, EnCase, SleuthKit and Autopsy. Also, several forensic imaging, carving and discovery tools.

Topics to be covered

- a. Course Introduction. ForSec Lab Discussion. Introduction to Network & Computer Forensics.
- b. Computer Investigations. Forensic Tools and Tool systems
- c. Certification: Investigators and Laboratories.
- d. Data Acquisition & Image Creation. Proc. Crimes & Incidents.
- e. Mass storage. Solid state (flash) and rotating magnetic drives.
- f. Volumes & Partitions.
- g. MBR Partitions. GPT Partitions.
- h. FAT File system. NTFS File system.
- i. Linux Boot & Disk & Partition. EXT File Systems. Sleuthkit
- j. File Carving. File carving analysis & lab
- k. ADS. ADS lab.
- l. Memory (RAM) forensics.
- m. Virtual machine forensics.

ITMS 443 SYLLABUS

ITMS 443 Vulnerability Analysis and Control

Hours: 3 credit hours / 45 contact hours

Instructor: Kevin Vaccaro

Textbook, title, author, and year: *Mastering Kali Linux for Advanced Penetration Testing*, Vijay Kumar Velu, 2017

Specific course information

- a. **Catalog description:** This course addresses hands-on ethical hacking, penetration testing, and detection of malicious probes and their prevention. It provides students with in-depth theoretical and practical knowledge of the vulnerabilities of networks of computers including the networks themselves, operating systems, and important applications. Integrated with the lectures are laboratories focusing on the use of open source and freeware tools; students will learn in a closed environment to probe, penetrate, and hack other networks. It is recommended, but not required, that students also take ITMS 448 prior to or in parallel with this course.
- b. **Prerequisites:** None
- c. **Required for Applied Cybersecurity and Information Technology.**

Specific goals for the course

- a. **Course Outcomes:** Each student will be able to explain the professional hacker's methodology for attacking a network and differentiate between different methods of attacks and countermeasures.
- b. **Course student outcomes:**
At the conclusion of this course, each student should be able to:
 - Explain the professional hacker's methodology for attacking a network.
 - Explain the script kiddie's methodology for attacking network.
 - Explain Network Security vulnerabilities.
 - Explain Hackers, hacker techniques, tools and methodologies
 - Describe hacker motivation, perform network reconnaissance and network scanning methods
 - Describe and perform covering tracks after gaining access to a network.
 - Describe the general symptoms of a virus attack
 - Define and describe the two basic approaches to antivirus software.
 - Describe how to defend against a worm and virus attack.

- Describe the steps in planning for a computer incident.
- Identify the difficulty is establishing who has jurisdiction over a computer crime.
- Understand the legal issues with regard to preserving digital evidence.
- Identify and describe the incident response goals and priorities.
- Describe the factors involved in identifying a computer incident.
- Describe and use the various tools associated with identifying an intruder.
- Describe how to handle and evaluate a computer incident.
- Recognize the role of law enforcement and rule of particularity in executing a search warrant.
- Describe the role the network security specialist would play in assisting the law enforcement and prosecution effort.
- Describe the difficulties in prosecuting a computer crime incident.
- Differentiate between competitive intelligence, economic intelligence, and industrial espionage

Topics to be covered

- a. Goal Based Penetration Testing
- b. Kali / Using Linux /Basic Scripting
- c. Open Source Intelligence and Passive Reconnaissance
- d. Active Reconnaissance of External and Internal Networks
- e. Vulnerability Assessment
- f. Physical Security and Social Engineering
- g. Reconnaissance and Exploitation of Web-Based Applications
- h. Attacking Remote Access
- i. Client-Side-Exploitation
- j. Bypassing Security Controls
- k. Exploitation
- l. Action on Objective
- m. Privilege Escalation
- n. Command and Control

ITMS 446 SYLLABUS

ITMS 446 Active Cyber Defense

Hours: 3 credit hours / 60 contact hours

Instructor: Louis McHugh

Textbook, title, author, and year: *TestOut CyberDefense Pro*, TestOut ISBN: 978-1-935080-73-2, 2021

Specific course information

- a. **Catalog description:** This course covers the duties of cybersecurity analysts who are responsible for monitoring and detecting security incidents in information systems and networks, and for executing a proper response to such incidents. The course introduces tools and tactics to manage cybersecurity risks, identify various types of common threats, evaluate the organization's security, collect and analyze cybersecurity intelligence, and handle incidents as they occur.
- b. **Prerequisites:** ITMS 448 or ITMS 443.

Specific goals for the course

- a. **Course Outcomes:** This course will address aspects of threat and vulnerability management; software and systems security; security operations and monitoring; incident response; and compliance and assessment necessary to prepare students to properly defend an enterprise against cyber attacks. This course and the concepts described in the class cover topics included in the CompTIA Cybersecurity Analyst (CySA+) professional certification.
- b. **Course student outcomes:**
Students completing this course will be able to:
 - Collect and use cybersecurity intelligence and threat data.
 - Recall and describe modern cybersecurity threat actors' types and tactics, techniques, and procedures.
 - Analyze data collected from security and event logs, and network packet captures.
 - Respond to and investigate cybersecurity incidents using appropriate forensic analysis techniques.
 - Assess information security risk in computing and network environments.
 - Implement a vulnerability management program.
 - Analyze and address security issues with an organization's network architecture.
 - Describe data governance controls and discuss their importance.
 - Describe and address security issues with an organization's software development life cycle.
 - Describe and address security issues with an organization's use of cloud and service-oriented architecture.

Topics to be covered

- a. Threat Intelligence
- b. Risk Mitigation
- c. Social and Physical Security
- d. Reconnaissance
- e. Enumeration
- f. Vulnerability Management
- g. Identity and Access Management Security (IAM)
- h. Cybersecurity Threats
- i. Infrastructure Security
- j. Wireless and IOT Security
- k. Infrastructure Analysis
- l. Software Assurance
- m. Data Analysis

INCIDENT RESPONSE

ITMS 448 SYLLABUS

ITMS 448 Cyber Security Technologies

Hours: 3 credit hours / 45 contact hours

Instructor: Maurice Dawson

Textbook, title, author, and year: *Official (ISC) 2 Guide to the CISSP CBK*. CRC Press. Gordon, A. (Ed.). 2015

Specific course information

- a. **Catalog description:** Prepares students for a role as a network security analyst and administrator. Topics include viruses, worms, and other attack mechanisms, vulnerabilities, and countermeasures; network security protocols, encryption, identity and authentication, scanning, firewalls, security tools, and organizations addressing security. A component of this course is a self-contained team project that, if the student wishes, can be extended into a fully operational security system in a subsequent course.
- b. **Prerequisites:** ITMO 440
- c. **Required.**

Specific goals for the course

- a. **Program Educational Objective**
2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
- b. **Course Outcomes:**
Each successful student will gain an in-depth understanding of various important network and computer security concepts and practices. Students, through their course exams, labs, and homework will demonstrate the ability to apply information assurance and security concepts, specifically on the topics of malware analysis, attack vectors, mitigation/deterrents, cryptography, steganography, computer forensics, firewalls, IDS/IPS, internet security protocols, authentication, and wireless network security.
- c. **Course student outcomes:**
 - Recall and describe various careers in cybersecurity
 - Describe Access Control and Bash Scripting
 - Describe use of the NIST SP 800 Series Publications
 - Describe Security Architecture and Design, DIACAP IA Controls, Virtualization
 - Recall and describe key concepts of Physical and Environmental Security
 - Recall and describe key concepts of Telecommunications and Network Security

- Recall and describe key concepts of Cryptography and Cryptographic Applications
- Describe the need for and function of Business Continuity and Disaster Recovery
- Recall applicable Laws, Regulations, Compliance, and Investigations and describe their application
- Demonstrate knowledge of Application Security
- Demonstrate knowledge of Operations Security
- Recall and describe special topics in cybersecurity
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions (**ABET Computing Criterion 3.1**)
- Communicate effectively in a variety of professional contexts (**ABET Computing Criterion 3.3**)
- Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline (**ABET Computing Criterion 3.5**)
- Assist in the creation of an effective project plan

Topics to be covered

- a. Careers in Cyber Security, Security Trends, Information Security and Risk Management, Introduction to Linux
- b. Access Control, Bash Scripting, Introduction to NIST SP 800 Series
- c. Security Architecture and Design, DIACAP IA Controls, Virtualization
- d. Physical and Environmental Security
- e. Telecommunications and Network Security
- f. Cryptography and Cryptographic Applications
- g. Business Continuity and Disaster Recovery
- h. Legal, Regulations, Compliance, and Investigations
- i. Application Security
- j. Operations Security
- k. Special Topics: Cyber Terrorism, Destructive Coding Practices, Human Intelligence (HUMINT), Signals Intelligence (SIGINT), Imagery Intelligence (IMINT), Measurement and Signatures Intelligence (MASINT), Open Source Intelligence (OSINT), Offensive Security, Defensive Security, Certification and Accreditation (DIACAP, NIST SP 800 Series, Common Criteria, NSA Rainbow Series, Directive of Central Intelligence Directives), Reverse Engineering

ITMS 458 SYLLABUS

ITMS 458 Operating System Security

Hours: 3 credit hours / 60 contact hours

Instructor: Sean Hughes-Durkin

Textbook, title, author, and year: Online materials will be assigned for reading

Specific course information

- c. **Catalog description:** This course will address theoretical concepts of operating system security, security architectures of current operating systems, and details of security implementation using best practices to configure operating systems to industry security standards. Server configuration, system-level firewalls, file system security, logging, anti-virus and anti-spyware measures and other operating system security strategies will be examined.
- d. **Prerequisites:** ITMO 356.
- e. **Required for Applied Cybersecurity and Information Technology.**

Specific goals for the course

- c. **Course Outcomes:** Each successful student will be able to describe the different types of malicious threats targeted to an operating system. The student will be able to explain ways to mitigate these threats, correct vulnerable configurations, and use best practices to harden systems. This course and the concepts described in the class cover topics included on the Certified Information Systems Security Professional (CISSP). The GIAC Security Essentials (GSEC) certification is another recognized security certification that covers the concepts the student will learn throughout this course.
- d. **Course student outcomes:**
Students completing this course will be able to:
 - Describe potential system attacks and the actors that might perform them
 - Describe appropriate measures to be taken should a system compromise occur
 - Describe characteristics of malware and identify different malware
 - Apply tools and techniques for identifying vulnerabilities
 - Describe, for a given OS, the steps necessary for hardening the OS with respect to various applications
 - Securely install a given OS, remove or shut down unnecessary components and services, close unnecessary ports, ensure that all patches and updates are applied
 - Identify the major concepts in modern operating systems and the basic security issues in OS design and implementation (how the first principles of security apply to operating systems)

Topics to be covered

- n. Malicious Software/Attacks (2 parts)
- o. Incident Handling
- p. User Authentication & Access Control Cryptographic Tool
- q. Host Firewalls
- r. Host Based Intrusion Detection (2 parts)
- s. General OS Hardening
- t. Linux Hardening
- u. Windows Hardening
- v. Post OS Hardening Testing

ITMS 478 SYLLABUS

ITMS 478 Cyber Security Management

Hours: 3 credit hours / 45 contact hours

Instructor: Ray Trygstad

Textbook, title, author, and year: *Management of Information Security, Sixth Edition*, Michael E. Whitman & Herbert J. Mattord, 2018

Specific course information

- a. **Catalog description:** In-depth examination of topics in the management of information technology security including access control systems & methodology, business continuity & disaster recovery planning, legal issues in information system security, ethics, computer operations security, physical security and security architecture & models using current standards and models.
- b. **Prerequisites:** None.
- c. **Required for Applied Cybersecurity and Information Technology.**

Specific goals for the course

- a. **Course Outcomes:** Each successful student will demonstrate foundation knowledge and application of cybersecurity concepts as they to apply the management of information system security in a large organizational environment. Students will describe and identify policy frameworks, legal and moral implications, and best practices in information security management. Students will be able assist in the conduct of a security audit of an organization and report on the results with appropriate suggestions for amelioration of problem areas identified.
- b. **Course student outcomes:** Upon completion of this course, each student should be able to:
 - Discuss the history of computer security and how it evolved into information security
 - Identify and define key terms and critical concepts of information security
 - Describe the business need for information security
 - Differentiate between laws and ethics, describe the role of ethics in professional practice in information security, and identify major national laws that relate to the practice of information security
 - Define risk management and its role in the Security Systems Development Life Cycle
 - Assist in the preparation and conduct of a cybersecurity audit of an existing business, government agency or organization and prepare a complete audit report with

- appropriate suggestions for amelioration of problem areas identified
- Describe management's role in the development, maintenance, and enforcement of information security policy, standards, practices, procedures, and guidelines
- Assist in the design and implementation of a comprehensive enterprise security program using policy and standards to implement technical, operational, and managerial controls
- Recall and describe recommended security management models
- Explain what contingency planning is and how incident response planning, disaster recovery planning, and business continuity plans are related to contingency planning.
- Describe common technical security controls, implementations in an enterprise setting, and how they are driven by policy and standards

Topics to be covered

- a. Introduction to Information Security
- b. Compliance, Legal, Ethical/Professional Issues Governance and Planning for Security
- c. Security Policy
- d. Developing Security Programs
- e. Risk Management I
- f. Risk Management II
- g. The Information Security Audit
- h. Security Management Models
- i. Security Management Practices
- j. Contingency Planning: Disasters/Business Continuity
- k. Security Maintenance and Digital Forensics Protection Mechanisms
- l. HIPAA

ITMS 479 SYLLABUS

ITMS 479 Topics in Information Security

Hours: Variable, but normally 3 credit hours /
45 contact hours

Instructor: TBD

Textbook, title, author, and year: Will vary based on
specific course content.

Specific course information

- a. **Catalog description:** This course will cover a particular topic in Information Security, varying from semester to semester, in which there is particular student or staff interest. This course may be taken more than once but only 9 hours of ITMS 479/579 credit may be applied to a degree.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will vary based on specific course content.
- b. **Course Student Outcomes:** Will vary based on specific course content.

Topics to be covered: Will vary based on specific course content.

ITMS 483 SYLLABUS

ITMS 483 Digital Evidence

Hours: 3 credit hours / 45 contact hours

Instructor: Shawn Davis

Textbook, title, author, and year: *E-discovery: An Introduction to Digital Evidence*, Phillips, Amelia; Godfrey, Ronald; Steuart, Christopher; Brown, Christine, 2014

Specific course information

- a. **Catalog description:** In this course, students learn the fundamental principles and concepts in the conduct of investigations in the digital realm. Students will learn the process and methods of obtaining, preserving and presenting digital information for use as evidence in civil, criminal, or administrative cases. Topics include legal concepts and terminology, ethics, computer crime, investigative procedures, chain of custody, digital evidence controls, processing crime and incident scenes, data acquisition, email investigations, applicable case law, and appearance as an expert witness in a judicial or administrative proceeding.
- b. **Prerequisites:** ITMS 438
- c. **Required for Applied Cybersecurity and Information Technology.**

Specific goals for the course

- a. **Course Outcomes:** Each successful student will demonstrate foundation knowledge and application of digital evidence and e discovery concepts as they apply to the investigation of computer crimes and cyber security incidents in a large organizational environment. Students will describe and identify policy frameworks, legal and moral implications, and best practices in the collection, processing and presentation of digital evidence. Students will be able to conduct digital investigations in full compliance with applicable law, policy, and regulations, and present the investigative results as an expert witness.
- b. **Course student outcomes:**
 - Acquire, process, preserve, evaluate, and present digital evidence in a forensically and legally sound manner.
 - Recall and describe law, theories, techniques, and practices that apply to digital forensic investigations.
 - Identify and describe types of computer and Internet crimes.
 - Preserve and process a crime scene involving digital evidence.

- Explain the legal procedures and standards in the collection and analysis of digital evidence.
- Prepare a report of a digital investigation for appropriate stakeholders and defend your findings.
- Present an analysis of digital evidence in a legal or administrative proceeding as an expert witness.

Topics to be covered

- a. Introduction to Legal Concepts and Terminology
- b. Introduction to Digital Evidence
- c. History and Ethics of E-discovery and Digital Evidence
- d. Planning and Tools
- e. Experts in Digital Evidence and E Discovery
- f. Digital Evidence Case Flow
- g. Case Study: From Beginning to Trial
- h. Information Governance and Litigation Preparedness
- i. Presenting Digital Evidence in Court
- j. Digital Evidence Case Law
- k. The Future of Digital Evidence

ITMS 484 SYLLABUS

ITMS 484 Governance, Risk and Compliance

Hours: 3 credit hours / 45 contact hours

Instructor: Bonnie A. Goins

Textbook, title, author, and year: Online resources including COBIT, NIST, and other sources

Specific course information

- a. **Catalog description:** In-depth examination of topics in governance, risk and compliance, including security program development; development and implementation of policies, standards and procedures; risk management and assessment methodologies, practices and outcomes; compliance standards, methods, processes and practices.
- b. **Prerequisites:** None.

Specific goals for the course

- a. **Course student outcomes:**
Each successful student will demonstrate foundation knowledge of governance, risk and (GRC) concepts, practice and outcomes as they apply to an organization. Students will be able to describe and identify policy frameworks and best practices for GRC.
- b. **Course student outcomes:**
At the conclusion of this course, each student should be able to:
 - Describe governance, risk and compliance frameworks.
 - Describe governance, risk and compliance methodologies.
 - Identify and detail program components for governance, risk management and compliance programs
 - Apply GRC concepts in the build of a Data Governance model and corresponding deliverables.
 - Evaluate and categorize risk 1) with respect to technology; 2) with respect to individuals, and 3) in the enterprise, and recommend appropriate responses.
 - Compare advantages and disadvantages of various risk assessment methodologies.
 - Select the optimal methodology based on needs, advantages and disadvantages.
 - Describe the impact of legal/regulatory standards on a given system.
 - Describe how audits are conducted using the SOC 2.
 - Describe the difference between auditing and assessment.

Topics to be covered

- a. Introduction to GRC
- b. Governance Introduction and Methodology
- c. Governance Implementation-Data Governance
- d. Governance Design
- e. Risk Management Introduction and Framework
- f. Risk Assessment Method
- g. Risk Assessment Threat and Vulnerability Considerations
- h. Risk Assessment Outcomes
- i. Compliance Introduction
- j. Compliance Audit and Assurance

ITMS 534 SYLLABUS

ITMS 546 Active Cyber Defense

Hours: 3 credit hours / 60 contact hours

Instructor: Calvin Nobles, Ph.D.

Textbook, title, author, and year: *Human Factors in Simple and Complex Systems 3rd Edition*, Proctor, Robert W. and Trisha Van Zandt December 26, 2017, CRC Press; ISBN 9781482229561

Topics to be covered

- a. User interface (UI) design (web, mobile, tablet)
- b. Interaction Design & User Experience
- c. Usability testing
- d. Wireframing & Prototyping (Rapid, Paper, Interactive)
- e. Storyboarding, Ideation
- f. User-centered Design (UCD)
- g. Cybersecurity Use Cases

Specific course information

- f. **Catalog description:** This course introduces the applied theories relevant to human factors in information security, digitalization, and sociotechnical environments. Examines the human element through a comprehensive approach that explores human errors, new technologies, and cybersecurity incidents. Investigates human-related aspects that have an impact on the practices, policies, and procedures that are in place in an organization to secure the firm's information. Topic areas include human behavior, ethics, psychology, social engineering, the culture of hacking, cybercrimes, security fatigue, and burnout. The analysis covers techniques to prevent intrusions and attacks that threaten organizational data and methods to identify potential insider threats.
- g. **Prerequisites:** None.

Specific goals for the course

- e. **Course Outcomes:** Upon successful completion of this course, students should be able to recall and employ principles of human factors, human computer interaction, user interface design, and user experience to enhance cybersecurity in an enterprise setting through system error reduction, countering of social engineering techniques, and employment of human factors as an element of active cyber defense.
- f. **Course student outcomes:** Students completing this course will be able to:
 - Identify and articulate key human factors concepts
 - Identify human factors shortfalls in major data breaches and cybersecurity incidents.
 - Effectively evaluate human errors and mistakes using the Human Factors and Analysis Classification System to prevent/reduce such mistakes from occurring in the future.
 - Apply human factors principles to reduce high friction point in system designs and cybersecurity operations.
 - Describe and apply core theories, models and methodologies from the field of human factors.
 - Describe and discuss current research in the field of human factors.

ITMS 546 SYLLABUS

ITMS 546 Active Cyber Defense

Hours: 3 credit hours / 60 contact hours

Instructor: Louis McHugh

Textbook, title, author, and year: TestOut; *TestOut CyberDefense Pro* ISBN: 978-1-935080-73-2, 2021

Specific course information

- h. **Catalog description:** This course covers the duties of cybersecurity analysts who are responsible for monitoring and detecting security incidents in information systems and networks, and for executing a proper response to such incidents. The course introduces tools and tactics to manage cybersecurity risks, identify various types of common threats, evaluate the organization's security, collect and analyze cybersecurity intelligence, and handle incidents as they occur.
- i. **Prerequisites:** ITMS 548 or ITMS 543.

Specific goals for the course

- g. **Course Outcomes:** This course will address aspects of threat and vulnerability management; software and systems security; security operations and monitoring; incident response; and compliance and assessment necessary to prepare students to properly defend an enterprise against cyber attacks. This course and the concepts described in the class cover topics included in the CompTIA Cybersecurity Analyst (CySA+) professional certification.
- h. **Course student outcomes:** Students completing this course will be able to:
 - Collect and use cybersecurity intelligence and threat data.
 - Recall and describe modern cybersecurity threat actors' types and tactics, techniques, and procedures.
 - Analyze data collected from security and event logs, and network packet captures.
 - Respond to and investigate cybersecurity incidents using appropriate forensic analysis techniques.
 - Assess and document information security risk in computing and network environments.
 - Create and implement a vulnerability management program.
 - Analyze and address security issues with an organization's network architecture.
 - Describe data governance controls and explain their importance.
 - Explain and address security issues with an organization's software development life cycle.
 - Explain and address security issues with an organization's use of cloud and service-oriented architecture.

Topics to be covered

- w. Threat Intelligence
- x. Risk Mitigation
- y. Social and Physical Security
- z. Reconnaissance
- aa. Enumeration
- bb. Vulnerability Management
- cc. Identity and Access Management Security (IAM)
- dd. Cybersecurity Threats
- ee. Infrastructure Security
- ff. Wireless and IOT Security
- gg. Infrastructure Analysis
- hh. Software Assurance
- ii. Data Analysis
- jj. Incident Response

ITMT 330 SYLLABUS**ITMT 330 Introduction to Information Systems and the IT Profession** (Formerly ITM 100)

Hours: 3 credit hours / 45 contact hours

Instructor: Ray Trygstad

Textbook, title, author, and year

- a. *Information Systems for Business and Beyond* Bourgeois, David T. 2014
- b. *Business Processes & Information Technology* Gelinas, Ulric J. Jr., Sutton, Steve G., Federowicz, Jane 2008
- c. *Information Technology for Management* Lucas, Henry C. Jr. 2009
- d. *Introduction to information systems—16th ed.* Marakas, George M. & O'Brien, James 2013
- e. *Information Systems* Watson, Richard T. (Ed.) 2007
- f. Supplemental Materials: Online readings as assigned in Blackboard

Specific course information

- a. **Catalog description:** Introduces students to concepts of systems, systems theory and modeling, information systems, and system integration. Examines the steps necessary to analyze a business problem and identify and define the computing and information requirements appropriate to its solution, with a focus on how to design, implement, and evaluate a technology-based system to meet desired needs. Students learn to analyze the local and global impact of computing on individuals, organizations, and society. Leads students to recognize of the need for continuing professional development, and imparts an understanding of professional, ethical, legal, security and social issues and responsibilities in information technology. Students write and present, building their ability to communicate effectively with a range of audiences, and using standard planning methodologies design an information system to meet the information needs of a small business. This course meets the university Introduction to the Profession Core Course requirement.
- b. **Prerequisites:** ITM 301 and (ITM 311 or ITM 312 or ITM313 or CS 116 or CS 201)
- c. **Required.**

Specific goals for the course

- a. **Program Educational Outcome:**
 3. Apply current technical and mathematical concepts and practices in the core information technologies and recognize the need to engage in continuing professional development.
- b. **Course Outcomes:** Each successful student will be able to use concepts of systems theory, systems modeling, and system integration to design an information system using standard planning methodologies to meet identified business needs. They will be able to analyze the local and global impact of computing on individuals, organizations, and society, engage in continuing professional development, and discuss professional, ethical, legal, security & social issues and responsibilities.

c. Course student outcomes:

Upon completion of this course the student should be able to do the following:

- Recall and describe concepts of system theory and system modeling
- Outline and describe the components of an information system
- Explain the role of information technology as the facilitating function in a modern business enterprise
- Discuss considerations for deployment of technology in a business setting
- Explain the standards of professional communication used within the information technology profession
 - Communicate (understand, and respond) in a manner appropriate to the profession
- Analyze a business problem and identify and define computing requirements appropriate to its solution
- Design a computer-based information system to meet desired business needs
- Describe concepts of system integration and its application to information system design
- Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles (**ABET Computing Criterion 3.4**)
 - Explain the importance of ethics to the information technology profession
 - Be able to recognize ethical issues and propose appropriate responses to ethical problems
 - Describe the norms of professional behavior as an information technology professional
 - Discuss how professional conduct reflects on and supports the information technology profession
 - Recognize the need for and engage in continuing professional development
- Function effectively as a member or leader of a team engaged in the design and development of information technology solutions to business problems (**ABET Computing Criterion 3.5**)

Topics to be covered

- a. Introduction to Systems & Systems Theory
- b. System Modeling Concepts and Methods
- c. Introduction to Information Systems
- d. Data, Information, and Data Management
- e. Communicating Technology & Project Management
- f. Professionalism as an Information Technology and/or Cybersecurity Professional
- g. Information in the Enterprise
- h. Systems Analysis & Requirements Definition
- i. Systems Design and Implementation
- j. Integrating Systems
- k. Information in Society and the World
- l. Ethics and Professional Responsibility
- m. Legal and Security Responsibility

ITMT 430 Syllabus

ITMT 430 Systems Integration (Senior Capstone Course)

Hours: 3 credit hours / 60 contact hours; 30 hours lecture, 30 hours lab

Instructor: Jeremy Hajek

3. Textbook, title, author, and year:

- a. *DevOps Handbook - How to Create World-Class Agility, Reliability, & Security in Technology Organizations*, Gene K., Patrick D., John W., Jez H., 2016.
- b. *Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations*, Nicole F., Jez H., Gene Kim, 2018.

Specific course information

- a. **Catalog description:** In this capstone course, students will identify, gather, analyze, and write requirements based on user needs and will then design, construct, integrate, and implement an information system as a solution to a business problem. Students will document integration architecture, methodologies, and technologies using industry best practices. User needs and user centered design will be applied in the selection, creation, evaluation, and administration of the resulting system. The system design process will take into account professional, ethical, legal, security, and social issues and responsibilities and stress the local and global impact of computing on individuals, organizations, and society. Discussion will also cover the need to engage in continuing professional development.
- b. **Prerequisites:** ITMD 321, ITMD 411, ITMD 362, ITM 100, ITMM 471, ITMO 340, and ITMO 356
- c. **Required**

Specific goals for the course

- a. **Program Educational Objectives:**
 1. Problem solve and create innovative answers to provide technology solutions for the problems of business, industry, government, non-profit organizations, and individuals.
 2. Perform requirements analysis, design and administration of computer and network-based systems conforming to policy and best practices, and monitor and support continuing development of relevant policy and best practices as appropriate.
 3. Apply current technical and mathematical concepts and practices in the core information technologies and recognize the need to engage in continuing professional development.
- b. **Course Outcomes:**
At the completion of this course you will have experienced software application development in a team setting. You will understand the roles of the project manager, a software developer, security analyst, IT operations, and UI/UX developer. You will have produced artifacts consistent with the

nature of each job and applied the techniques and concepts learned in all of your pre-requisite courses. The final measurable outcome will be a full deployment of a working application from scratch. You will be familiar with DevOps terminology and development practices. You will have integrated hardware and software into a complete information system to meet identified user needs as a solution to a defined business problem and demonstrated ethics, and an understanding of legal, security, and social issues and responsibilities of information systems. You will have integrated hardware and software into a complete information system to meet identified user needs as a solution to a defined business problem. You will have demonstrated building world class reliable, agile, and secure cloud native applications.

c. Course student outcomes:

At the conclusion of this course, each successful student will be able to:

- Identify, gather, analyze, and write information system requirements based on user needs.
- Document integration requirements using business process models.
- Design, construct, integrate, and implement an information system as a solution to a business problem.
- Apply key systems integration architecture, methodologies, and technologies in the construction of an information system using industry best practices.
- Based on identified user needs, demonstrate the use of user centered design in the selection, creation, evaluation, and administration of an information system.
- Recall and explain professional, ethical, legal, security, and social issues and responsibilities in information systems.
- Describe the local and global impact of computing on individuals, organizations, and society
- Describe the need to engage in continuing professional development and explain how this may be achieved.
- Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions **(ABET Computing Criterion 3.1)**
- Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline **(ABET Computing Criterion 3.2)**
- Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline **(ABET Computing Criteria 3.5)**
- Identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems **(ABET IT Criterion 3.6)**
- Assist in the creation of an effective project plan

Topics to be covered

- a. Course Introduction
- b. Introduction to Tooling, Project Management, & Communication
- c. The Three Ways and Where to Start
- d. Technical Practices of Flow
- e. Technical Practices of Feedback
- f. Technical Practices of Learning
- g. Technical Practices of Information Security
- h. Measuring Performance; Measuring & Changing Culture
- i. Technical Practices of Architecture
- j. Information Security and Management Practices
- k. Product Development; Making Work Sustainable
- l. Leading & Managing; Data

ITMT 491 SYLLABUS

ITMT 491 Undergraduate Research

Hours: Variable, but normally 1-3 credit hours / 15-45 contact hours

Instructor: TBD

Textbook, title, author, and year: Will vary based on specific course content.

Specific course information

- a. **Undergraduate research.** Written consent of instructor is required.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will be defined by student research proposal or prospectus.
- b. **Course Student Outcomes:** Will be defined by student research proposal or prospectus.

Topics to be covered: Will be defined by student research proposal or prospectus.

ITMT 492 SYLLABUS

ITMT 492 Introduction to Smart Technologies

Hours: 3 credit hours / 45 contact hours

Instructor: Jeremy Hajek

Textbook, title, author, and year: Online resources will be assigned in Blackboard.

Specific course information

- a. **Catalog description:** This course covers reconfigurable intelligent devices programmed with modern high-level languages focusing on design and integration to modern environments. This course also covers the topic and deployment of wireless sensor networks and the use of rapid prototyping for commercial application. Students will discover hardware, software and firmware design trade-offs as well as best practices in current embedded systems development. A final project will integrate course topics into a system using an embeddable single-board microcontroller.
- b. **Prerequisites:** ITM 311 or ITM 312 or ITM 313.

Specific goals for the course

- a. **Course Outcomes:** The student will be exposed to a wide array of tools and services that support smart technology. They will be able to solve problems that involve the concepts of data collection, data transmission, and data presentation by using the technologies learned in the course. This survey of Smart Technologies covering wireless protocols, AR devices, voice assistants, solar powered and battery back sensor networks and cloud storage will give students a sufficient ability to create innovative solutions to problems they encounter.
- b. **Student course outcomes:**
At the conclusion of the course, the student should be able to:
 - Describe and apply principles of electricity and electronics that support smart tech.
 - Read and use schematics, diagrams, and electronic symbols.
 - Explain concepts of Data Collection, Data Transmission, and Data presentation using small computers and sensor networks
 - Recall the fundamentals and use of wireless communication standards: Bluetooth, NFC, xBee (802.15), Wi-Fi
 - Describe concepts of solar panels and LiPo batteries
 - Deploy solar panels and LiPo batteries
 - Describe the use of cloud data storage for smart technology
 - Recall the basics of Augmented Reality devices

- Describe the principles and use of Voice Assistants
 - Demonstrate a basic working knowledge of Voice Assistants

Topics to be covered

- a. Data Transmission: wireless tech Wi-Fi
- b. Adafruit IoT Portal
- c. AWS/Azure IoT Portals
- d. Intro to Voice Assistants
- e. Final Project

ITMT 495 SYLLABUS

ITMT 495 Topics in Information Technology

Hours: Variable, but normally 3 credit hours /
45 contact hours

Instructor: TBD

Textbook, title, author, and year: Will vary based on
specific course content.

Specific course information

- a. **Catalog description:** This course will cover a particular topic varying from semester to semester in which there is particular student or staff interest.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will vary based on specific course content.
- b. **Course Student Outcomes:** Will vary based on specific course content.

Topics to be covered: Will vary based on specific course content.

TECH 465 SYLLABUS**TECH 465 Introduction to Social Commerce**

Hours: 3 credit hours / 45 contact hours

Instructor: TBD

Textbook, title, author, and year: None assigned.

Excerpts from books, web sites and articles will be used.

Specific course information

- a. **Catalog description:** Provides an introduction and basic knowledge of social commerce to help students develop a practical understanding of the design, construction, market readiness and synergistic integration of a business mobile application. The course will provide a practitioner focus that will benefit students in a start-up or company/corporate setting.
- b. **Prerequisites:** None

Specific goals for the course

- a. **Course Outcomes:** Understand and apply the basic concepts of social commerce: definition and structure; design which meets customer needs; technology factors to make a winning mobile application; understanding and application of basic algorithms and data analytics; integration with partner and customer applications, systems and rewards; working as part of a team-design, develop and prepare proposal for an actual social commerce mobile application and how to win in the market place with the application.
- b. **Course Student Outcomes:**
Upon completion of this course, students should be able to:
 - Recall basic concepts and components of social commerce
 - Explain how to meet customer needs and the role of research and analysis
 - Describe market research for social commerce
 - Identify and employ algorithms for customer experiences, partner experiences, and rewards
 - Describe the use of data analytics and data mining in social commerce
 - Explain integration with partner and customer applications, systems and rewards
 - Recall and describe the principles and concepts of mobile application development
 - Develop and prepare a proposal for a social commerce mobile application
 - Explain how to win in the marketplace with the proposed application

Topics to be covered:

- a. Syllabus; Intro: Definition, Structure, Scope, Business future
- b. Components of Social Commerce: Needs, Configuration, Design
- c. Customer needs research and analysis
- d. Customer factors and Ideation
- e. Competitive Advantage-Competition research
- f. Market Research-Proof of Idea
- g. Risks and Options-Market for Idea
- h. Final design of Idea-Lean Canvas
- i. Algorithms for Customer experiences
- j. Algorithms for Partner experiences
- k. Algorithms for Rewards
- l. Data Analytics
- m. Data Mining
- n. Big data mining integration, synergies and rewards
- o. System flow of Idea include: DBs, files, structure
- p. Mobile application basics
- q. Mobile application designs, data handling/transmission
- r. Mobile application final design for Idea
- s. Mobile application security
- t. Mobile application process-order
- u. Mobile application process-payment
- v. Mobile application data base design and uses
- w. Mobile application final design bench test
- x. Mobile application integration with other apps, systems, etc.
- y. Idea application programming requirements
- z. Idea application testing requirements
- aa. Go to market-pricing and channels of distribution
- bb. Roll-out and Feedback loop
- cc. Business Model and Plan Bus Group Paper
- dd. Group Presentation

TECH 497 SYLLABUS

TECH 497 Independent Study

Hours: Variable, but normally 1-3 credit hours / 15-45 contact hours

Instructor: TBD

Textbook, title, author, and year: Will vary based on specific course content.

Specific course information

- a. **Catalog description:** Independent study and projects in applied technology that are multi/cross-disciplinary not tied to a specific department.
- b. **Prerequisites:** Will vary based on course content.

Specific goals for the course

- a. **Course Outcomes:** Will vary based on course content or will be defined by student course proposal or prospectus.
- b. **Course Student Outcomes:** Will vary based on course content or will be defined by student course proposal or prospectus.

Topics to be covered: Will vary based on course content or will be defined by student course proposal or prospectus.