Real-Time Networking

Spring 2018

CSC 486

instructor:	Will Meyers
email:	ekeenan2@cdm.depaul.edu
office hours:	Tuesday, 12:30 – 5:45pm in classroom or by email appointment
office:	CDM 849
website:	Piazza (Preferred communication)
lecture:	Lewis Center 1108, Wednesday, 5:45-9:00pm
Desired to Learn (D2L):	d2l.depaul.edu (Grades, Viewing lectures, Announcements)
Version Control:	perforce: ***140.192.39.61:1666***

Description:

Applied real-time networking fundamentals. Topics include low-level TCP/UDP socket programming, serialization of data for network transportation, deterministic data flow and queuing in applications, dynamic configuration and session communications. Students will explore issues related to performance-based networking in different topologies: client/server and peer to peer. Students will create and build real-time applications using different protocols (TCP, UDP, RUDP) with active network simulation of loss, delayed, corrupted and out of order data packets. Architecture of different bandwidth compression and estimation techniques will be investigated. Research of wireless ad hoc networks such as Bluetooth, LTE, WIFI, Zigbee will be explored.

Prerequisites:
- CSC 461 and (SE 456 or SE 450)
- or instructor consent

Learning Goals:
- Students should be able to implement TCP/UDP socket programming.
- Students will be able to serialized data structures for network transmission.
- Students will be able to create a deterministic data driven flow in their applications.
- Students should be able to understand the components and communicate with lobby services.
- Students will be able to implement bandwidth compensation techniques for real-time applications such as dead-reckoning and estimation.
- Students will be able to research and comprehend the impact of emerging networking technology trends.

Grading:

10% - Research Paper on a Wireless Protocol (Bluetooth, LTE, WIFI, ZIGBEE) (5-page minimum)

40% - Final Network Game project with bandwidth compensation

35% - Programming Assignments (5 assignments)

- PA1 - Environment (0% - required)
- PA2 - Serialization (10%)
- PA3 - Client / Server (10%)
- PA4 - Data Driven (5%)
- PA6 – Bandwidth Compensation (10%)

15% - Final Exam

NOTE: You must pass the final exam (60% or higher) to receive a passing grade in the class. *(It's not an easy exam)*

Textbooks and printed resources

- ***Books***
  - ***Computer Networking: A Top-Down Approach (6th Edition)***, 2012, Kurose J., Ross K., Addison Wesley, ISBN-13: 978-0132856201
    - 7th edition is OK as well
- ***Material*** will be provided by the instructor
  - Lectures, links, SDKs and other corresponding material

Software

- ***Microsoft Visual Studio 2017 Enterprise Edition***
  - [Visual Studio Enterprise 2017](#)
    - C+ and C# install
- ***Perforce - Visual Client (p4v)***
  - www.perforce.com
- Download and configuration instructions will be provided in class
  - Server address: ***140.192.39.61:1666***

Programming assignments (PA):

- ***5 programming assignments (30%)***
  - PA1 - Environment (0% - lose 5% if now working)
    - C#, C++, Perforce working, Fundamentals
  - PA2 - Serialization (10%)
    - Packing data into continuous small packets for inflation and deflation
  - PA3 - Client / Server (10%)
    - TCP/UDP protocols
  - PA4 - Data Driven (5%)

- Object communication serially driven from a queue
    - PA5 – Network Prototype (0% - Undergrad assignment)
        - Data Driven networking prototype – proof of concept
    - PA6 – Bandwidth Compensation (5%)
        - Dead reckoning – prediction and smoothing
- ***Submission of all development projects***
    - Submitted to Perforce

Research Paper – 10%

- Research a wireless ad hoc network (Bluetooth, LTE, WIFI, or Zigbee)
    - Research the mechanics and behavior of Bluetooth, LTE, WIFI, or Zigbee networks
    - Identifying the way the networks are created, limitations, how to discover, connect, ownership, topology, etc….
- Read the material from the class reference books and external materials
    - journals, white papers, technology specifications
- Write a 5 page paper
    - Summarizing the technology and the impact on future networking systems / applications
- Papers will be check for plagiarism using ***Turn It In*** program

Final Project (40%)

- Modify an existing game to use networking
- Game must have a Lobby / Chat room
    - Ability to join
    - Match making
    - Friends list
    - Invite list
- Synchronization
    - Games must be synchronized before start
    - Code must monitor its synchronization during the game
- Networking choice either:
    - Client / Server
    - Peer to Peer
- Demo must be
    - Warning free
    - Stable
    - Working!!!!
- Art and models may be lifted from demos and other sources
    - Programmer art and models are allowed

Final Project (40% Grade)

Create a 2 player networked game in C# Language

Sprints are suggested mini-milestones, grading at the end of quarter

Sprint 1:   Game Data Driven (15%)

- Convert an existing a 2 player space war game
  - Game works, great graphics, but never was intended to be networked
  - Refactor the game to become data driven with a queuing mechanics

Sprint 2: Lobby (10%)

- Create a game Lobby
  - Join and enter matches
  - Find friends
  - Return to lobby after game

Sprint 3: Moving Objects and Ships (10%)

- Network the game using reliable UDP
  - Moving ships, tracking bombs, missiles

Write up (5%)

- 5 pages
- Video capture with commentary
- Design discussion and lessons learned

Final Exam - 15%

- A comprehensive final exam, covering the concepts of this class.
- Closed book written exam.
  - Final week of class
- You must pass the final exam (60% or higher) to receive a passing grade in the class.
  - *It's not an easy exam*

Perforce Submissions

- Everyone is expected to submit several <u>submissions</u> to perforce a week.
  - Minimum 5 significant (real) submissions on 3 separate days.
  - To promote incremental development and prevent last day rush.
  - Grade deduction will occur if not followed
- The biggest reason students get into trouble with software design:
  - Not starting the project early
  - Not working on the material frequently enough
  - Taking too large of a bite(byte) of the design
- Both are minimized with this *Perforce RULE*
- Even my simplest programs take 10-20 submissions.
  - For these project assignments my average is 40-400 submissions, so 5 will be no problem.
- Detailed perforce changelist comments are expected

Piazza Discussion forum

- Statistics show: students who participate more and help other students <u>do better</u>!
  - The correlation is ridiculous!
    - Poor understanding / poor participation.
    - Great understanding / Great participation
  - As you master the material, help others learn!
    - Want to be a Master programmer so master it!
- Everyone is ***expected*** and encouraged to participate on the Piazza discussion forum. All class-related discussion here this term.
  - At least one real question or response per week from every student.
- Everyone is expected to keep up with the material on Piazza and are responsible for its content. Critical class updates and directions will be presented there.
  - Not participating or reading the material on Piazza is not an excuse.
- All correspondence that is not personal in nature should be vectored through Piazza
  - Sensitive material, use Piazza private note, not email.
- The quicker you begin asking questions on Piazza (rather than via emails), the quicker you'll benefit from the collective knowledge of your classmates and instructors. I encourage you to ask questions when you are struggling to understand a concept.
- Keep the forum professional and positive, help each other out.
  - Karma really pays off here.
  - Help each other whenever you can.
    - There will be a section where you'll need help (trust me).

NOTE: Do ***NOT*** post until you have watched the entire lecture ***FIRST*** (in class or online)

　　　This will prevent frustration on all sides (members asking or answering questions)

Collaborating together on programming assignments

- You are encouraged to work together
  - o Use the Piazza forums heavy
  - o Even share your material with others in the common directory
    - ▪ Obviously not the answers
- Everyone is 100% responsible for the work they do.
  - o If you get help with a section of code,
  - o Please refactor the code the ***snot out of it***
    - ▪ Comment and understand that material
    - ▪ Transform the code to ***make it yours***
  - o Be able to answer ***any*** question regarding the code you commit
- System for Detecting Software Plagiarism
  - o We will be using MOSS - Measure of Software Similarity (Stanford University)
    - ▪ Indicates possible code infringements (plagiarism)
    - ▪ MOSS - will detect the similarity independent of naming convention, indentation style or formatting, it compares abstract syntax tree of your code.
  - o I will pursue any plagiarism/integrity violations aggressively, arguing for full expulsion from the university for the offenders.
    - ▪ Don't put me or you in this scenario
- If you gain significant support / help from another student
  - o Fully disclose the support / help you had in a Readme.txt file submitted with your assignments.
    - ▪ Disclosing the help, is ***not permission*** for copying the code.
    - ▪ Only there to clarify and acknowledge help you were given from a fellow student.
- Modifying any Unit Test to alter the outcome results is also an ***Academic Integrity Violation***
- If you are stuck and find yourself even tempted to plagiarize
  - o Ask for help !!!!
    - ▪ Use on Piazza -> Visit during offices hours, make an appointment
    - ▪ ***Don't ever compromise your integrity!***
- Material was uniquely created for this Class.
  - o You indirectly by the process of tuition, "paid" for the contents and material of this class.
    - ▪ Do not share this ***copyrighted*** material in any form
    - ▪ It is design for your personal use, while enrolled in the Class.
  - o Do ***NOT*** post any content or revealing material to any external website or forum outside of this class.
    - ▪ The Class Piazza forum is provided for this service, ask questions there, not on the internet (i.e. StackOverflow and other software forums)
- After you leave this class
  - o You are expressly ***FORBIDDEN*** to provide or share the content with others.
  - o Academic Integrity Violations can still be applied to students who provide material support to other students even after completion of the class.
- Just follow the golden rule:
  - o ***"I have neither given, nor received, nor have I tolerated others' use of unauthorized aid."***

Tentative Schedule:

Week	Lecture	Assign	Due
1	Syllabus, Class Overview Network Introduction History of Game Networking	PA1 - Environment	
2	Data Alignment and Padding Serialization Networking IP DNS	PA2 - Serialization	PA1
3	Sockets Socket Programming Networking NAT	PA3 - TCP/UDP	PA2
4	C# Basics Protocol Internals	PA4 - Data Driven Queue	PA3
5	Data Driven Networking Debugging	Research Paper (PA5 – Prototype)	PA4
6	Game State Architecture	Sprint 1 - Game Data Driven	(PA5 – no credit)
7	Game Lobby Latency	Sprint 2 - Lobby	Sprint 1
8	Prediction Concurrent Networking	PA6 - Prediction	Sprint 2
9	Consistency Replication	Sprint 3 – Moving Ships	PA6 Research Paper
10	Cheating / Security Review		Sprint 3
11	Final Exam		Final Project

April 1, 2018	Last day to add classes to SQ2018 schedule
April 6, 2018	Last day to drop classes with no penalty, Last day to select pass/fail option
April 7, 2018	Grades of "W" assigned for SQ2018 classes dropped on or after this day
April 13, 2018	Last day to select auditor status
May 11, 2018	Last day to withdraw from SQ2018 classes

Course Policies

Changes to Syllabus

This syllabus is subject to change as necessary during the quarter. If a change occurs, it will be thoroughly addressed during class, posted under Announcements in D2L and sent via email.

Online Course Evaluations

Evaluations are a way for students to provide valuable feedback regarding their instructor and the course. Detailed feedback will enable the instructor to continuously tailor teaching methods and course content to meet the learning goals of the course and the academic needs of the students. They are a requirement of the course and are key to continue to provide you with the highest quality of teaching. The evaluations are anonymous; the instructor and administration do not track who entered what responses. A program is used to check if the student completed the evaluations, but the evaluation is completely separate from the student's identity. Since 100% participation is our goal, students are sent periodic reminders over three weeks. Students do not receive reminders once they complete the evaluation. Students complete the evaluation online in CampusConnect.

Academic Integrity and Plagiarism

This course will be subject to the university's academic integrity policy. More information can be found at http://academicintegrity.depaul.edu/. If you have any questions be sure to consult with your professor.

Academic Policies

All students are required to manage their class schedules each term in accordance with the deadlines for enrolling and withdrawing as indicated in the University Academic Calendar. Information on enrollment, withdrawal, grading and incompletes can be found at: cdm.depaul.edu/enrollment.

Students with Disabilities

Students who feel they may need an accommodation based on the impact of a disability should contact the instructor privately to discuss their specific needs. All discussions will remain confidential.
To ensure that you receive the most appropriate accommodation based on your needs, contact the instructor as early as possible in the quarter (preferably within the first week of class), and make sure that you have contacted the Center for Students with Disabilities (CSD) at: csd@depaul.edu.

Lewis Center 1420, 25 East Jackson Blvd.
Phone number: (312)362-8002
Fax: (312)362-6544
TTY: (773)325.7296

Retroactive withdrawal

This policy exists to assist students for whom extenuating circumstances prevented them from meeting the withdrawal deadline. During their college career students may be allowed one medical/personal administrative withdrawal and one college office administrative withdrawal, each for one or more courses in a single term. Repeated requests will not be considered. Submitting an appeal for retroactive withdrawal does not guarantee approval. Information on enrollment, withdrawal, grading and incompletes can be found at:
http://www.cdm.depaul.edu/Enrollment-Policies.aspx