CS 494P/594 Programming Project Specification

- You can do your project either individually or with up to 2 other partners.
- You can use any high level programming language for your project. Use of open-source libraries is permitted but your network protocol design must be completely your own.
- If you are planning to build your project on top of existing open-source libraries (e.g. Python's Twisted API) please check with me first.
- You are allowed to combine software development for your inter-networking project with another course project. However, you have to clearly state what the programming contribution for the internet working class, which has to be related to networks.

Project Option 1: Internet Relay Chat

IRC or Internet Relay Chat is an application that lets multiple users communicate via text messages with each other in common "virtual" rooms. You will be implementing an IRC client and server from scratch in this project using whatever programming language you are comfortable with. As the programmer, you are in charge of all of the protocol specifications and functionality of your IRC application. However, at a minimum, the basic functionality of being able to create a room, join a room, leave a room, and list rooms available should be implemented. Other features such as private chat, file transfer, buddy lists, etc. can be added, but must be specified and documented. Please refer to the IRC project grading criteria. Graphical user interfaces (GUI) development is not required for this project.

Along with the source code of your IRC client and server, you will turn in an RFC-style document that describes your IRC protocol. That is, describe the format of the messages that the client and server will exchange in order to properly implement the IRC application. An example RFC that you may base your protocol specification on is the IRC RFC 1459.

Project Option 2: Networked Games

You are to choose a multiplayer game, specify a network protocol for supporting the game on-line, and implement the protocol faithfully in a programming language of your choice. The protocol should be robust and concise. In particular, the client should perform input validation and only send the server valid messages. In addition, formatting strings and user interface messages should be mainly generated at the client. While the game should be fully functional, you will be mainly graded on the quality of your protocol specification and implementation.

Along with the source code of your game client and server, you will include an RFC-style document that specifies your game protocol messages. An example RFC that you may base your protocol specification on is the IRC RFC 1459. The RFC will specify the format of the messages that the client and server will exchange in order to properly implement the on-line game.

You are encouraged to discuss this project option with the instructor beforehand. Please finalize and submit grading criteria for this project to the instructor by **4/26/2023.**

Project Option 3: Propose your Own Project

Students can substitute a project proposal of their own instead of the IRC application or the networked game application. If you wish to do a project outside of either of these two, please discuss your project and seek approval from the instructor beforehand.

You are permitted to combine this project with a course project for another course. However, you must clearly identify which part of the project pertains to each course.

594 students are welcome to propose a project that aligns with their research area. Please finalize and submit grading criteria for this project with the instructor by **04/26/2023.**

Project Milestones

Milestone	Date	Instructions
Project Selection	**4/26/2023**	Please finalize project, team members and discuss grading criteria for your project with the instructor by this date. Upload to **Canvas.** If you are doing the IRC project, simply upload a statement to that effect with your team member names.
Draft Network Protocol Specification (RFC Document)	**5/21/2023**	Upload to **Canvas.**
Project source code RFC document	**6/9/2023**	Soft-copy of project source folder shared with the instructor. Upload to **Canvas** Link to shared repository (ėg. GitHub) is also accepted.
Project demo and code review	**Week of 6/2/2023 and 6/9/2023.**	Demos will be conducted via Zoom. Demo slots will be assigned via an online signup mechanism (eg. Doodle or Calendly poll.) Students who cannot make it to any of the proposed demo slots, can request a demo slot on an EARLIER date. No demos will be scheduled after **6/14/2023.** To discourage students from obtaining code from other sources, you will be walking through your code with us as well as giving us a demo. We will also ask specific questions about the code that you are expected to answer. You are advised to have complete knowledge of all code that you turn in.

: