

Project

About

There is one mandatory final project for software technologies (CS3002) course. The project must be done along. Team projects are not allowed given that there are very few students who have opted for the course. The project should be done in C only.

You can choose any topic of your choice for project. Sample project choices are mentioned below to give you some idea about average size of projects. It is recommended that you choose something that you yourself want to do and avoid just selecting project from below list of project ideas.

Please note that topics that were chosen last year cannot be chosen again to avoid duplication of work. You can refer to http://www.sbarjatiya.in/website/course_2010_software_technologies_projects.php to see which topics are not acceptable.

Areas for project

The project should combine at least three of following topics. The idea is not to just include topics to make total count three but to ensure that project really requires the diverse knowledge of three different areas for the problem to get solved.

Areas for project are: (Topic should cover at least three of these)

1. Simple file handling (Buffered or Unbuffered)
2. File based databases (SQLite)
3. Server based databases (MySQL or PostgreSQL)
4. GUI programming (GTK)
5. TCP or UDP Socket programming
6. Capturing packets (libpcap)
7. Injecting packets (libnet)
8. Multi-threading (libpthread)
9. Multi-processing (fork(), exec(), kill())
10. Inter-process communication (Shared memory or Pipes)

11. Synchronization (Semaphores)

12. Web Interface (ccgi) (Wont be covered as part of course)

It is important to note that it may not be possible to cover all the above topics in regular lectures. Hence you may have to study some of these topics by yourself. Also even if the topic gets covered it may get done at very end of semester and you are expected to start working on project much earlier.

Hence choose areas such that either you are already aware of them or are covered or will get covered in regular lectures soon. It may not be possible for instructor or TAs to spend lot of time in solving your project related queries, so you should be able to learn things on your own.

Sample project ideas

Simple LAN messenger : The LAN messenger would consist of two components server and client. Both server and client would have GUI interface. When we start server it would ask which interface and which port to listen on. Then it would list the messages being sent by client and nicks of all connected clients.

The client on starting will ask server domain name or IP address and port number to connect to. After connecting the client will allow user to send messages to all other connected clients and see messages sent by other clients. The client can also see list of connected users.

The client and server should see client getting disconnected if some client closes his messenger. There should be some keepalive mechanism / time out mechanism to detect clients who are not reachable anymore. The messages sent should be logged with ID, timestamp, message, client IP address, client nick in some database so that we have accountability of messages sent over this simple messenger.

Basic check like two users which same nick should not be able to connect should also be done. This project covers three areas:

- GUI programming for interface design
- TCP or UDP socket programming for communication between server and clients
- Database connection for storing chat logs.

Fast and reliable file transfer using UDP : There are two very common transport layer protocols used in networks TCP and UDP. TCP

is reliable, connection oriented and has rate control. Due to rate control features of TCP it slows down when network is congested or when receiver is not ready to receive messages. This allows fair distribution of bandwidth among parallel TCP sessions.

Through this project you have to create reliable file transfer mechanism using UDP. So you would have to take care of sequence numbers, acknowledgment numbers, buffers, re-transmissions, timeouts etc. in your code. The advantage would be no rate control so that the clients can consume entire bandwidth without sharing it fairly with others.

The client should have command line options to run in server mode (accept file) or client mode (send file). The server would listen on some UDP port for file data and client would send file data to that port. The server would accept interface and port to listen on on command line as parameters. Client would accept server IP/name, port number and file to send on command line.

The areas getting covered in this project are:

- UDP socket programming
- File handling

The project would require very good understanding of networking and good algorithmic skills to take care of all situations like heavy packet loss, packets arriving out of order, The program should be able to handle large files (size ≥ 2 GB) during transfers. The program may also require multi-threading to receive acknowledgment and send data simultaneously.

Dead-lines

You are expected to send and get project ideas approved before 7th March, 2011 evening 06:00pm. All communication would be done over email and you have to send all emails to instructor. For common project ideas the person who approaches first would be given preference. The emails must be sent from your official IIIT email ID. Emails from gmail / yahoo etc. accounts wont be accepted.

The projects are expected to be completed by 7th April, 2011 evening 06:00pm, that is within one month after registering project topic. The final submission is supposed to be done on courses website.

Deliverables

You have to submit C code with doxygen style documentation along with Makefile and readme file in .tar.bz2 format. The readme file should contain information about special requirements for using the project (example postgresql should be installed, some database, username, password, tables must exist, etc.)

In case database must be configured before project can be used than give detailed command required to setup database and tables in readme file.

Caution

Please note that

- The code is documented and good variable names / indentation are used.
- The entire project should not be available on Internet. You are allowed to download libraries and sample code to use as starting point. In case you do so you must mention the links and sources. All the original files downloaded and their links must be part of project submission. You can put all this in folder named 'reference'.

You can use all course examples or assignment solutions to do project.

- The weightage of project is going to be very high as assignments are same as last time. Hence do not procrastinate working on project till the dead-line is very near. Ensure that you start working on project right away