

대학/학과	동국대학교 컴퓨터공학과
강좌명	동시성프로그래밍
담당교수	문 봉 교

<Project#2>

1. 과제 명: 얼랭 기반의 IRC Lite 채팅 어플리케이션 테스트

2. 과제 목표: 주어진 채팅 소프트웨어(클라이언트 및 서버)의 실행 및 테스트를 통한 소스 분석

3. 과제 내용 검토: 교재 11장의 프로그램은 순수 얼랭 프로그램으로 OTP라이브러리를 사용하지 않고 표준 라이브러리도 최소한으로 사용한다. 따라서 이 프로그램은 그 자체로 완전한 클라이언트-서버 아키텍처와 명시적인 연결 조작을 기반으로 한 오류 복구 형태를 가진다. 라이브러리를 사용하지 않는 이유는 한 번에 한 가지씩 개념만 소개하고, 얼랭 자체와 최소한의 라이브러리만으로도 구현할 수 있음을 보여주려는 의도이다. 코드의 일련의 구성요소는 복잡한 방법으로 서로 얹혀서 돌아가는데, 나중에 OTP라이브러리를 사용한다면 이런 복잡함은 대부분 제거할 수 있다. 실제 대부분의 얼랭 서버는 OTP 기반의 generic server 라이브러리를 이용하여 구현된다.

주어진 교재의 참고사항은 다음과 같다.

- 11.1절은 메시지 시퀀스 다이어그램으로 메시지의 흐름을 설명한다.
- 11.2절은 사용자 인터페이스를 위한 io위젯에 대해서 설명한다.
- 11.3절은 세 개의 프로세스(io위젯, 채팅 클라이언트, 미들맨)로 구성된 클라이언트 측 소프트웨어를 설명한다.
- 11.4절은 채팅 컨트롤러(채팅 클라이언트 각각에 대하여 채팅 서버와의 연결을 담당), 채팅서버(진행 중인 모든 채팅 세션에 대해 알고 있다), 그룹관리자(채팅 그룹당 하나씩 개별 채팅 그룹을 관리)로 구성된 서버 측 소프트웨어를 설명한다.
- 11.5절은 전체 어플리케이션을 실행하는 방법을 설명한다.
- 11.6절은 전체 채팅 프로그램 소스코드를 포함한다.

4. 채팅서버 확장시 고려사항:

여유가 있다면 11.7의 연습문제들을 하나씩 해결해 보아라.

5. 평가 기준:

채팅 소프트웨어를 실제로 실행하여 테스트를 하고(교재의 내용과 동일하게 Linux상에서

테스트하는 것이 가장 좋음), 여러 가지 테스트 시나리오를 제시하고 이 시나리오에 따라 테스트를 수행한 내용을 체계적으로 포함하고 있으면 보너스 점수가 부여된다.

1. chat_client 모듈과 chat_server에 대한 구체적인 분석(30%)
2. 그 외의 주요 코드에 대한 코멘트 및 설명(15%)
3. 프로그램의 기능별 데모를 보여주는 화면 캡처(15%)
4. 연습문제의 개별 시나리오에 따라 채팅 프로그램 테스트 여부(25%)
5. 주어진 채팅 프로그램이 갑자기 클라이언트의 수가 증가하는 모바일 환경에 대해서 어떻게 동작할지 개인적인 예측을 기술해야 한다.(15%)