

2017학년도 2학기 네트워크 프로그래밍

- 개인 프로젝트 제안서 -



과목	네트워크 프로그래밍
담당교수	김기천/진정하 교수님
학과	컴퓨터 공학부
학번	201311269
이름	김제헌
제출일	2017-10-14-토

1. 프로젝트의 목적

- 네트워크 프로그램 프로젝트 완성
 - 지금까지 로컬환경에서 개별적으로 돌아가는 프로그램만 작성하였지만, 아직 도전해보지 않은 분야인 네트워크 프로그래밍을 완성시키는데 의의가 있습니다.

- 처음으로 개인 프로젝트 완성
 - 지금까지 해오던 여럿이서 진행하는 팀 프로젝트가 아닌 혼자서 진행하는 개인 프로젝트를 [도중에 낙오하지 않고] 끝까지 성공적으로 완성시키는데 의의가 있습니다.

- 여러 가지 도전적인 문제들을 해결하고 프로젝트 완성
 - 앞의 2가지 외에도, C++ 및 GUI 등등 해결해야 할 문제들을 해결해야 해서 아직까지는 불투명하지만 이러한 것들을 이겨내고 프로젝트를 완성시키는데 의의가 있습니다.

2. 프로젝트의 목표

○ 프로젝트의 주제

- 기본적으로 서버-클라이언트 1:N기반의 테트리스 게임을 만드는 것이 목적입니다.
- 왜 하필 테트리스인가?
 - 아직 이 네트워크 프로그래밍이라는 분야를 전혀 모르는 제게 있어서, '할 수 있는 범위 내에서' 적절하게 타협도 하면서, 어떤 주제를 정해야 할지 마땅히 생각나는 게 없어 고민하던 도중 그냥 무난하게 게임을 해보면 어떨까? 라는 생각을 하게 되었습니다.
 - 앞의 요인도 요인이지만 무엇보다 결정적으로 제가 군대 가기 전인 3년 전쯤에, 2학년 2학기 수업인 '컴응2 C++' 실습수업의 마지막 과제가 '테트리스 만들기'였기 때문에, 그 당시 만들었던 C++ 테트리스 소스코드가 있습니다.
 - 그 당시 완성도를 떠나서 주먹구구식으로 코딩을 했고, 객체지향 따윈 집어 치우고 어떻게든 만들어내긴 했습니다. 1인용으로 콘솔 환경에서 돌아갔고, 사실 예러도 많기도 한 그런 코드입니다.
 - 이를 코드도 다시 짚 보며 예러도 잡고 새롭게 객체 지향적인 설계 + GUI 적용 + 소켓 프로그램을 이용한 1:N 대전으로 만들어볼까 합니다.
- 그래도 뼈대 알고리즘인 테트리스 코드가 있다보니 아무래도 테트리스 자체보다는, 이에 소켓 프로그래밍을 적용해서 어떻게 하면 실시간 문제 등을 처리할 것인가 등을 중점으로 다뤄볼 생각입니다.

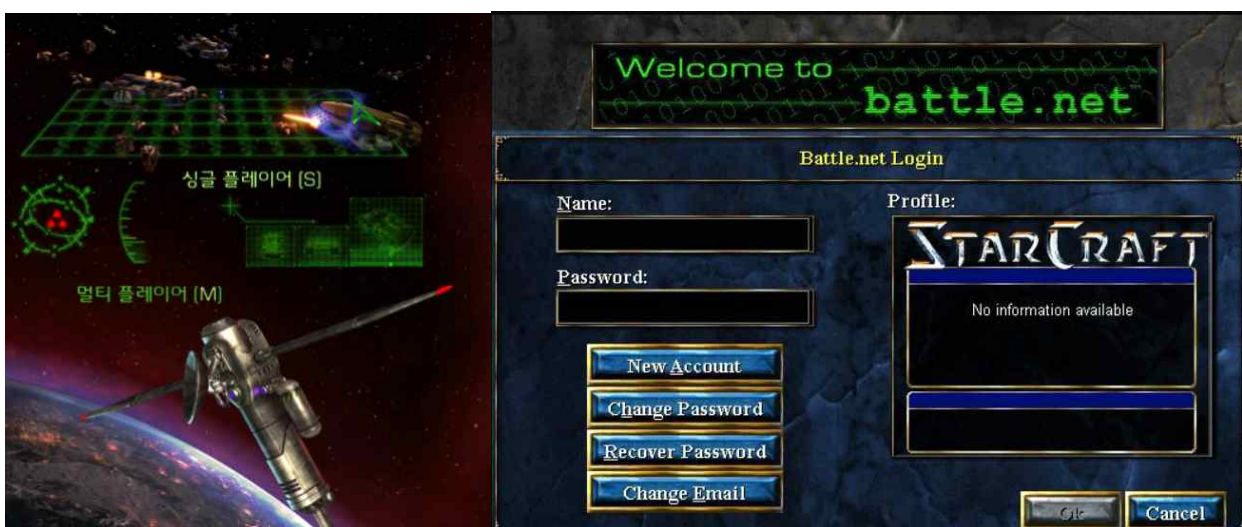
○ 프로그램의 기능 요구사항

- 싱글 모드와 멀티 모드
 - 간단하게 혼자서 테트리스를 즐길 수 있는 싱글 모드와 여럿이서 테트리스를 즐길 수 있는 멀티 모드로 나눌 예정입니다.
- 서버로의 로그인 및 회원가입 기능
 - 아이디와 비밀번호를 입력해서 멀티 모드인 서버로 접속하는 로그인 기능을 간단하게 추가할 예정입니다.
 - 회원가입 기능도 아주 간단하게 구현할 예정입니다.
- 전체 채팅 및 방 생성 및 방 목록 보기 기능
 - 멀티모드인 서버에 접속하면, 전체적으로 채팅을 할 수도 있고, 새로운 방을 생성하던지, 생성되어 있는 방 목록을 볼 수 있는 기능을 구현할 예정입니다.

- 방 목록 중에서의 방 입장 기능
 - 방 목록을 볼 수 있는 화면으로 넘어오면, 현재 생성되어 있는 방 목록이 보이며, 이에 입장할 수 있는 기능을 구현할 예정입니다.
- 각 방에서의 채팅 및 방장의 강퇴 기능, 레디 및 게임 시작 기능
 - 각 방에서도 채팅이 가능하고, 방장에게 있어 입장 가능한 자리를 열고 닫는 다던지 등의 강퇴 기능, 각 참가자들의 레디 기능, 모두 레디가 되었을 때 테트리스 게임을 시작하는 기능을 구현할 예정입니다.
- 어떤 방이 시작된 이후 테트리스 게임 시작
 - 해당 방의 참여자들끼리 개인전 테트리스를 진행하며, 게임을 하는 도중에 채팅도 할 수 있도록 구현할 예정입니다.
 - 게임이 끝난 후에는 다시 전체 방 목록을 볼 수 있는 화면으로 돌아가게 할 예정입니다.

○ 모티브로 삼은 것들

- 기본적으로 '스타크래프트1' + '한게임 테트리스'를 모티브로 삼았습니다.
- 스타크래프트1
 - 스타크래프트1에서도 싱글모드와 배틀넷 접속 등의 멀티모드가 있고, 배틀넷 접속 [로그인 및 회원가입] 이후에도, 전체 채팅방이 있으며[여기서는 채널개념이지만], 방을 생성하고, 입장하고, 채팅하고, 강퇴하는 등으로 게임 흐름에 있어 전체적인 뼈대는 스타크래프트1을 따라갈 생각입니다.



[싱글 모드 및 멀티모드 | 로그인 및 회원가입 기능]



[접속 후 전체채팅 가능, 방 생성 및 방 목록보기 | 방 목록 중 방 입장하기 | 해당 방에서 채팅, 강퇴, 준비 및 게임 시작]

- 한게임 테트리스

- 이건 게임 자체적인 GUI로써, 게임의 배치나 색상 방식 등을 이런 모습으로 따라 해볼 예정입니다.
- Hold 명령이나, Next 블록이 어떤 건지에 대해서도 보여주게 할까는 생각중입니다.
- 음악 변경이나 아이템 사용 등은 일이 커지는 것 같아서 아직까지는 보류중입니다.



3. 개발 환경

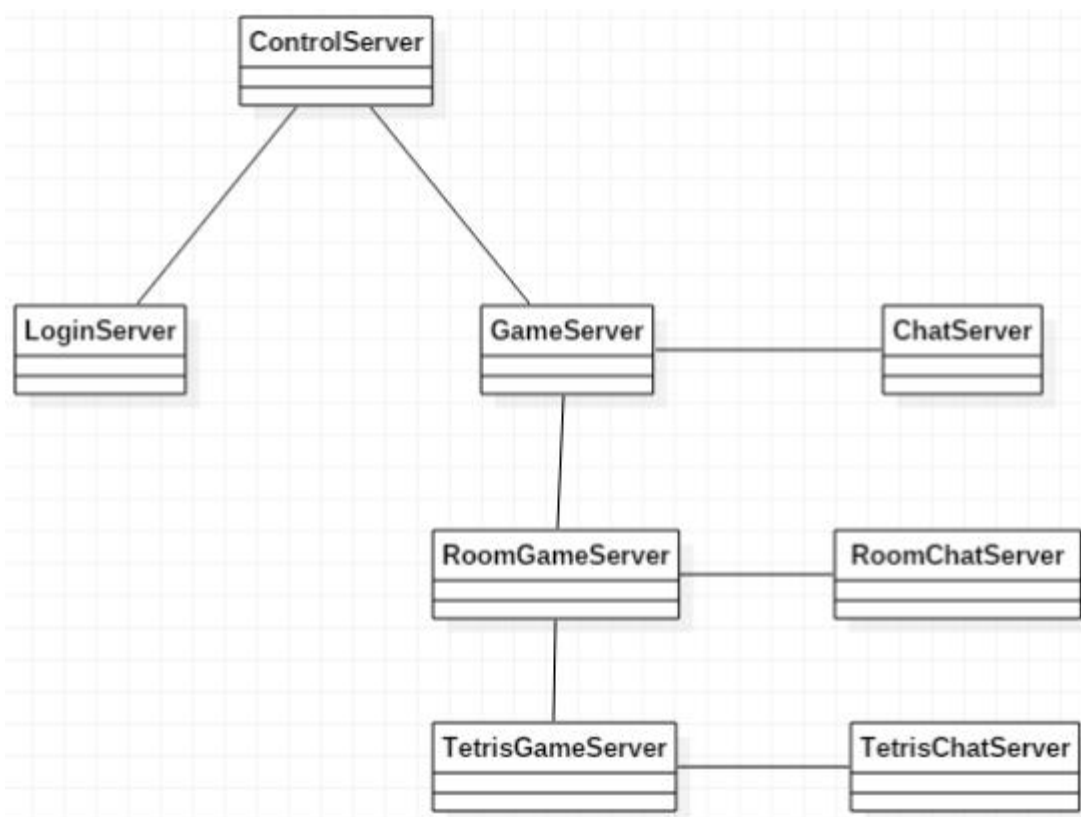
- Development Environment
 - A. 운영체제 : Windows 10 [64비트]
 - 지금 사용하고 있는 노트북 사양입니다.
 - B. CPU : Intel(R) Core(TM) i7-6700HQ
 - 지금 사용하고 있는 노트북 사양입니다.
 - C. 개발 언어 : C++
 - C++인 이유 :
 1. 기존 테트리스 코드가 C++인 이유도 있고,
 2. 이번학기에 교양으로 C++을 배우고 있는 것도 있으며[물론 기초과정],
 3. 리눅스 환경이긴 하지만 네프 본 수업에서 진행하고 있는 C와 함수적으로 크게 차이가 없을것으로 예상되서 추후 네트워크 프로그래밍을 진행하는데 있어 익숙할 것 같아서도 있고,
 4. 아무리 그래도 기존 C++ 테트리스 코드가 그냥 바꾸면 되니까 결국 자바가 GUI 등 여러 가지 측면에서 편하지만, 결정적으로 졸업하신 어떤 동아리 선배가 '학교 프로젝트에서나 자바쓰지 밖에서 요즘 누가 자바쓰냐고, 제대로 공부 할꺼면 C++이나 파이썬으로 프로젝트 해라'는 조언을 받아서, 공부차원에서 C++을 선택하게 되었습니다.
 - D. IDE : Visual Studio 및 QT
 - 해당 툴 선정 이유 : Visual Studio야 기존부터 C++을 코딩하던 툴이었으며 이런저런 시행착오는 여기서 겪을 예정입니다. 또한 C++ GUI를 만드는데 있어서 MFC로는 작업할 엄두가 안 나기 때문에 이것저것 알아보던 도중 QT라는 툴이 C++ GUI를 쉽게 만드는 것을 지원해준다고해서, 아직 모르지만 배워서 적용 해볼 생각입니다.
 - E. UML 툴 : StarUML
 - 추후에 Class Diagram, Sequence Diagram 등의 설계과정을 표현하는데 있어 이용할 툴입니다.

4. 기본 알고리즘

- 프로젝트를 진행하는데 있어 해결해야 할 문제 및 과정들
 - 1단계. 기존 C++ 테트리스 코드 수정
 - 기존 코드를 재분석하고, 에러를 잡아내고, 객체 지향적인 설계를 적용해서 코드를 수정해야 합니다.
 - 2단계. 싱글 모드 기반으로 테트리스 GUI 프로그래밍
 - 기존의 콘솔 모드를 떠나서 QT툴도 익히고, C++ GUI프로그래밍을 익힌 이후에 싱글 모드로 할 수 있는 테트리스 GUI 프로그래밍을 해야 합니다.
 - 3단계. 1:N기반의 로그인, 회원가입, 채팅 등등의 채팅서버 구현
 - 우선은 순수하게 '채팅'만 할 수 있는 1:N 채팅서버를 만들어볼 생각입니다.
 - 방을 생성하고, 강퇴하고, 방에서 들어가고, 나오고, 게임을 시작하고 등의 과정이 아직 감이 잘 안오기 때문에 이와 관련한 큰 틀을 만들어보려고 합니다.
 - 4단계. 특정 방 기준으로 1:N 테트리스 게임 구현
 - 2단계까지 만든 [완성된?] 테트리스 게임을 1:N 방식으로 바꿔서 구현해볼 생각입니다.
 - 어떻게 하면 실시간적으로 동시성을 처리할 것인가가 핵심인 것 같습니다.
 - 5단계. 3단계와 4단계를 합친다.
 - 마치 네트 FT 기능 구현 과제에서 에코 채팅 서버를 구현하고, FT 기능 서버를 구현한 후에 이 둘을 합친 것처럼 그런 느낌으로 합쳐볼 생각입니다.
 - 프로젝트 진행 순서
 - 1단계 및 2단계를 먼저 진행해야 할 것 같고, 그 이후에 3~5단계를 진행하는데, 3단계 및 4단계는 크게 관련은 없지만, 네트워크 프로그래밍이 아직 익숙하지 않기 때문에 3단계를 완성시켜서 감을 잡은 후에 4단계를 진행하는 것이 좋지 않을까 싶습니다. 4단계를 진행할 때는 5단계의 합치는 것까지 염두에 뒀서 진행을 해야 할 것 같습니다.

- 기본적인 시스템 기능 및 흐름도와 관련해서는 앞서 설명했으므로 생략하겠습니다.
- 아직 완벽하진 않지만 대충 생각하고 있는 전체적인 클래스 구조는 다음과 같습니다.

- 서버
- 1. ControlServer [LoginServer 및 GameServer를 제어]
 - 2. LoginServer [회원가입 및 로그인, 이후 GameServer로 넘어감]
 - 3. GameServer [방 생성 및 입장 등 각각의 RoomGameServer들을 관리]
 - 3.1. ChatServer [GameServer에서 전체 채팅 기능을 보조]
 - 4. RoomGameServer [해당 방에 속한 플레이어들끼리의 강퇴, 퇴장 등 처리]
 - 4.1. RoomChatServer [해당 방에 속한 플레이어들끼리의 채팅 기능을 보조]
 - 5. TetrisGameServer [해당 방에 속한 플레이어들끼리의 테트리스 진행을 처리]
 - 5.1. TetrisChatServer [해당 방에 속한 플레이어들끼리의 채팅 기능을 보조]



- 아직까지 막연하게 필요할거라 생각되는 기능들만 클래스화한 것이고, 추후 구현을 진행하면서 Chat와 관련된 서버들을 그냥 메소드로 통합한다던지 할 수도 있습니다.
- 클라이언트쪽은 기본구조는 서버와 비슷한 매칭으로 가되, 최상위 클래스인 MainSystem에서 싱글모드와 멀티모드로 나누는 등의 세부적인 추가기능이 더 있을 것 같습니다.

- TetrisGameServer에서, 게임의 실시간 동시성을 어떻게 잡을 것인가? [구상]
 - A~E까지의 5명이 입장한 어느 한 테트리스 게임이 실행되었다고 가정
 - A입장에서,
 - A는 자기가 하는걸 자기화면에 바로바로 적용시키면서, 그때마다 TetrisGame Server로 Send를 보낸다.
 - TetrisGameServer는 해당 Send를 받고, 보낸 클라이언트[Ex. A]를 제외한 나머지 클라이언트[Ex. B~E]들에게 받은 메시지를 Send를 보낸다.
 - A의 쓰레드 중에서 TetrisGameServer로부터 계속 Recv를 받아서 처리하는 메소드가 존재하며 이 쓰레드가 나머지 B~E까지의 진행과정을 실시간으로 동기화시켜준다.
 - B~E입장에서,
 - A와 동일하게 동작한다.



5. 프로젝트 진행 일정

(개발기간 : 2017. 8. 28. ~ 2017. 11. 30.)														
주차 세부내용	1	2	3	4	5	6	7	8	9	10	11	12	13	14
프로젝트 주제 정하기 및 제안서 쓰기														
중간고사														
1단계 및 2단계 진행하기														
3단계 및 4단계 진행하기														
5단계 진행하기 및 테스팅 등 마무리하기														
최종 발표준비														