**게임개발자전문가 과정**

**1,과목명: 게임서버프로그래밍**

**2,능력단위: 게임네트워크프로그래밍**

**3,제출일자: 2022 년 1월 25일**

**4,포트폴리오: TCP프로토콜의 라이브러리 및 채팅서버 구현**

**5,작성자: 구교진**

**<제출내역>**

1. **라이브러리 및 채팅서버 구현 프로젝트**
2. **라이브러리 및 채팅서버 구현 분석 및 세부 문서**

**<증거자료>**

|  |
| --- |
| * **반드시 폰트 크기= 10** * **줄 2칸 이상 엔터 금지** * **페이지 여백 조정 불가**   **< 현페이지 수정 불가 및**  **증거자료 공란 유지 >** |

**1, 라이브러리 및 채팅서버 구현 프로젝트 분석**

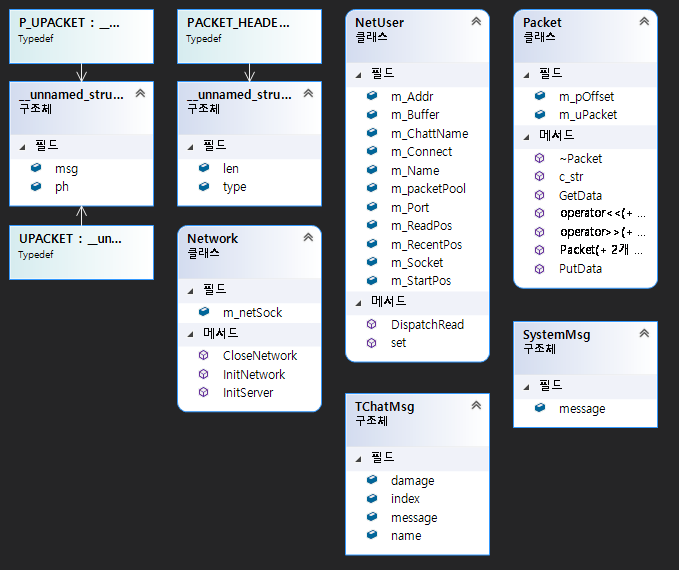
**클라이언트 및 서버 클래스 다이어그램**

**텍스트이(가) 표시된 사진

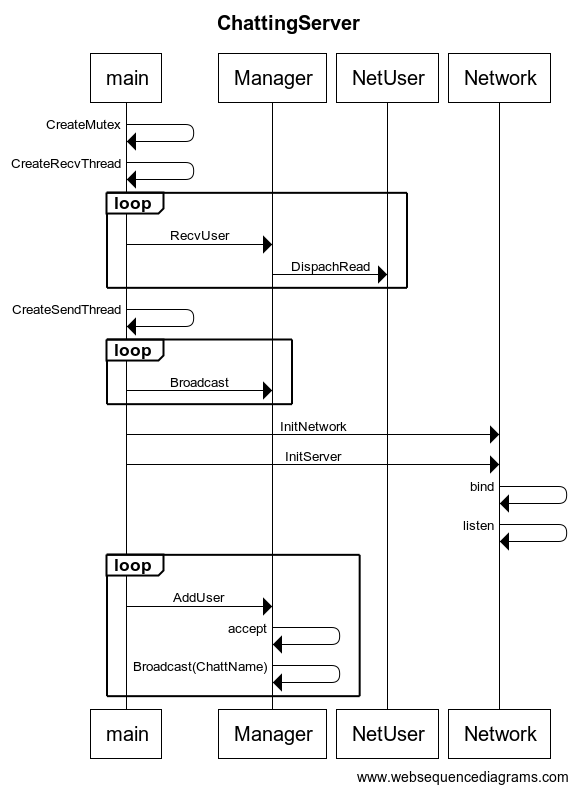
자동 생성된 설명** **텍스트이(가) 표시된 사진

자동 생성된 설명**

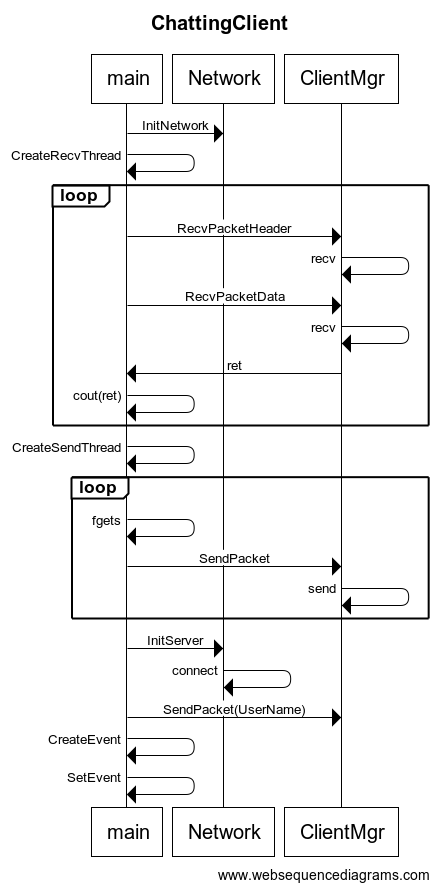
라이브러리 다이어그램

****

**서버 시퀀스 다이어그램**

****

**클라이언트 시퀀스 다이어그램**

****

**TCP와UDP**

|  |  |
| --- | --- |
| **TCP** | **UDP** |
| **신뢰성이 있다**  **신뢰성을 위해 Ack,Checksum등을 사용한다** | **신뢰성이 없다** |
| **연결 지향성이다**  **Connection을 맺고 통신한다** | **비 연결적이다** |
| **오류 및 패킷 손실 검출시 재전송을 요청한다** | **오류 및 패킷 손실 검출시 재전송을 요청하지않는다** |
| **Flow control을 위해 windowing을 사용한다 속도가 다소 느리지만 신뢰성이 있다** | **신뢰성을 보장하지 않지만 고속데이터 전송으로 실시간 전송에 적합하다** |
| **신뢰성이 필요한 통신에 사용된다** | **총 패킷수가 적은 통신 또는 동영상 및 음성 등 멀티미디어 통신에 사용된다** |

**비동기 및 동기 입출력**

**동기 입출력**

**동기식 입출력**은 입출력 요청 후 입출력 작업이 완료된 후에야 CPU의 제어권이 그 프로그램에게 다시 넘어갈 수 있는 방식

동기식 입출력은 입출력이 진행되는 동안 그 프로그램의 다음 명령을 수행하지 않고 기다리게된다 그러다가 입출력이 완료되어 인터럽트를 통해 그 사실이 전달된 후에야 CPU의 제어권이 그 프로그램에게 넘어가서 다음 명령을 수행할 수 있게된다

단점으로는 CPU의 명령 수행 속도는 빠른 반면 외부 장치에서 데이터를 읽어오는 등의 입출력 연산은 상대적으로 속도가 느리다. 그럼에도 불구하고 입출력이 완료될 때까지 기다렸다가 사용자 프로그램에게 CPU 제어권을 방식은 CPU가 입출력이 완료될 때까지 아무 일도 하지 못하기 때문에 자원의 낭비가 발생한다.

**비동기 입출력**

비동기식 입출력은 입출력 연산을 요청한 후에 연산이 끝나기를 기다리는 것이 아니라 CPU의 제어권을 입출력 연산을 호출한 그 프로그램에게 곧바로 다시 부여하는 방식이다.

비동기식 입출력에서는 입출력이 필요없는 작업을 먼저 수행하고, 읽어오는 데이터가 반드시 있어야 수행할 수 있는 일들은 입출력이 완료된 후에야 수행하게 된다.

또한, 입출력 요청이 디스크에서 읽어오는 요청이 아니라 디스크에 쓰는 요청이라면 쓰기 작업이 완료되기 전에도 다음 명령을 수행할 수 있으므로 비동기식 입출력이 사용될 수 있다.

**입출력 모델의 종류 및 설명**

1.Select

다른 모든 함수는 넌블로킹 상태로 작동하고 Select 함수만 블로킹 상태로 작동한다 flag를 통해 select함수에 신호를 주면 신호에 맞는 작업을 하게된다 그로인해 단 하나의 쓰레드를 가지고 다중 클라이언트 통신이 가능하다 다만 select함수로 처리 가능한 소캣의 수량은 쓰레드당 64개 이다

2.AsyncSelect

윈속의 소켓 이벤트를 윈도우 메시지를 통하여 비동기적으로 통보 받는다 함수를 호출하면 강제적으로 넌블로킹으로 바뀐다 따라서 함수의 리턴값이 아닌 이벤트에 의존하여 성공여부를 판단해야한다 윈도우 메시지 큐에 의존하여 처리가 이루어지기 때문에 많은 수의 소켓 이벤트를 처리해야 하는 경우 윈도우 메시지 큐가 고갈될 수 있다.

3.EventSelect

이벤트 오브젝트를 이용해 소켓의 통지를 받는다 비신호 상태로 있다가 소켓 이벤트가 발생하면 신호상태로 전환된다 입출력을 처리하고 난 뒤에는 수동으로 리셋시켜야 한다 한번에 하나의 이벤트만을 처리하기 때문에 소켓배열의 뒤로갈수록 이벤트를 처리할 기회가 적어진다 따라서 이벤트가 신호상태가 되었을 때 다른 이벤트의 상태도 검사해야한다

asyncselect와는 다르게 윈도우 메시지가 아닌 이벤트 오브젝트에 시그널을 변경시키는 방식으로 통지하기 때문에 메시지큐가 고갈될 위험이 없다

4.IOCP

모든 소켓 입출력 모델중 가장 뛰어난 성능을 제공한다 비동기 입출력 결과와 이 결과를 처리할 스레드에 대한 정보를 담고있는 구조로 APC큐와 비슷한 개념이다 차이점은 apc큐는 스레드마다 자동으로 생성되고 파괴되지만 iocp는 함수를 호출하여 생성과 파괴를 한다 apc큐는 저장된 결과를 apc큐를 소유한 스레드에서만 확인가능 하지만 iocp는 어느곳에서든 확인이 가능하다 apc큐는 저장된 결과를 처리하면 스레드가 alertable wait상태에 진입해야 하지만 iocp는 getqueuedcompletionstatu함수를 호출해야한다.

**라이브러리 및 채팅서버 구현 프로젝트 세부문서**

**라이브러리**

|  |  |
| --- | --- |
| **Protocol** | |
| Len | 패킷의 길이 |
| Type | 패킷의 타입 |
| Ph | 패킷헤더의 정보 |
| Msg | 패킷의 내용 |
| Name | 패킷의 내용중 이름 |
| Message | 패킷의 내용중 메세지 |

|  |  |
| --- | --- |
| **Packet** | |
| M\_offset | 패킷의 시작점 |
| M\_upacket | 패킷 구조체 |
| Putdata | 패킷에 데이터를 저장하는 함수 |
| Getdata | 패킷에 저장된 데이터를 가져오는 함수 |
| Operator << | 데이터 타입마다 패킷에 저장하기 위한 연산자 오버로딩 |
| Operator >> | 데이터 타입마다 패킷에서 자료를 가져올 연산자 오버로딩 |

|  |  |
| --- | --- |
| **Network** | |
| m\_netsock | 소켓 |
| Initnetwork | 윈속 초기화 |
| Closenetwork | 윈속 종료 |
| initserver | 서버 구축 및 접속 |

|  |  |
| --- | --- |
| **NetUser** | |
| M\_socket | 소켓 |
| M\_addr | 소켓주소 |
| M\_name | 유저소켓주소 |
| M\_chattname | 유저명 |
| M\_port | 포트번호 |
| M\_connect | 연결여부 |
| M\_buffer | 패킷버퍼 |
| M\_startpos | 읽어들일 패킷의 시작위치 |
| M\_recentpos | 마지막으로 읽어들인 패킷의 위치 |
| M\_readpos | 현재 읽어들인 패킷의 위치 |
| M\_packetpool | 패킷 리스트 |
| Set | 맴버변수를 초기화하는 함수 |
| Dispatchread | 패킷을 해석하는 함수 |

**서버**

|  |  |
| --- | --- |
| **Manager** | |
| Userlist | 유저 리스트 |
| M\_mutex | 뮤텍스 핸들 |
| sendmsg | 패킷을 전송하는 함수 |
| Logoutmsg | 유저의 퇴장을 전송하는 함수 |
| Adduser | 유저리스트에 유저를 추가하는 함수 |
| recvuser | 패킷을 받아내는 함수 |
| broadcast | 리스트에 있는 유저수만큼 sendmsg함수를 실행하는 함수 |

**클라이언트**

|  |  |
| --- | --- |
| **ClientMgr** | |
| Username | 유저명 |
| Sendpacket | 패킷을 전송하는 함수 |
| Recvpacketheader | 패킷의 헤더를 받는 함수 |
| Recvpacketdata | 패킷헤더를 이용해 데이터를 받는 함수 |

**기능 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

유저가 서버에 접속을 요청할 경우 주소와 포트번호를 출력하여 접속을 수락할지 결정

텍스트이(가) 표시된 사진

자동 생성된 설명

거절할 경우 접속요청 거부 패킷을 보내고 소켓을 닫음

텍스트이(가) 표시된 사진

자동 생성된 설명

수락할경우 유저가 접속함과 동시에 보내는 이름 패킷을 접속한 모든 유저에게 전송

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

이후 유저가 접속을 종료했을시 저장해둔 유저명을 이용하여 종료메세지를 남은 유저들에게 전송

텍스트이(가) 표시된 사진

자동 생성된 설명