컴퓨터 네트워크 – 프로그램 설계서

채팅 프로그램 ProtoChat의 설계서

컴퓨터과학과 2019010884 배정훈

1. 개요

ProtoChat는 클라이언트-서버 구조를 갖는 실시간 채팅 애플리케이션입니다. 프로그램 내에서 클라이언트 혹은 서버를 지원하며, 프로그램 내에서 통신 프로토콜로 UDP와 TCP 중에서 설정할 수 있습니다.

2. 기능 및 요구사항

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ProtoChat의 기능 요구사항 명세서 | | | | |
| **분류1** | **분류2** | **ID** | **기능 명** | **기능 설명** |
| 공통 기능 | 화면 공통 | FNC\_COM\_001 | 프로그램 종료 | 프로그램을 종료합니다. |
| FNC\_COM\_002 | 창 최소화 | 현재 화면을 최소화합니다. |
| FNC\_COM\_003 | 창 최대화 | 현재 화면을 최대화합니다. |
| 초기 설정 | 초기 설정 화면 | FNC\_INIT\_001 | 역할 선택 | 호스트의 역할을 설정합니다. 라디오 버튼을 통해 Client와 Server 중 택1을 통해 진행합니다. |
| FNC\_INIT\_002 | 프로토콜 선택 | 사용할 프로토콜을 선택합니다. 라디오 버튼을 통해 TCP와 UDP 중 택1을 통해 진행합니다. |
| FNC\_INIT\_003 | 제출 버튼 | 사용자가 선택한 역할과 프로토콜을 확정하고 다음으로 진행합니다. |
| 서버 | 연결 설정 화면 | FNC\_SVR\_001 | 서버 초기 설정 | 서버에서 사용할 포트를 설정하는 화면입니다 |
| 로비 화면 | FNC\_SVR\_002 | 로비 화면 | 서버의 상태를 모니터링하기 위한 로비 화면입니다. |
| FNC\_SVR\_003 | 채팅 방 목록 | 서버에서 현재 열려 있는 채팅방의 목록을 확인할 수 있는 영역입니다. 채팅방의 최대 인원수와 현재 인원수를 확인할 수 있는 간략한 정보도 함께 표시됩니다. 채팅방을 선택함으로써 해당 채팅방에 입장할 수 있습니다. 채팅방에 입장하면 채팅 방 내에서 채팅 할 수 있는 팝업이 불러와집니다. |
| FNC\_SVR\_004 | 채팅 방 만들기 버튼 | 채팅 방 생성 요청 팝업을 불러올 수 있는 버튼입니다. |
| FNC\_SVR\_005 | 안내 영역 | 다양한 정보를 사용자에게 알리기 위해 사용됩니다. |
| FNC\_SVR\_006 | 새로 고침 버튼 | 서버로부터 채팅 방 목록을 새로 고침 합니다. |
| FNC\_SVR\_007 | 온라인 유저 목록 | 서버에 연결되어 있는 유저들을 보여주는 영역입니다. |
| FNC\_SVR\_008 | 서버 로그 버튼 | 서버 로그 팝업을 화면으로 불러오는 버튼입니다. |
| 로비화면 – 채팅 방 생성 요청 팝업 | FNC\_SVR\_009 | 방 이름 입력필드 | 만들어질 채팅방의 이름을 입력하는 필드입니다. |
| FNC\_SVR\_010 | 방 최대인원 입력필드 | 만들어질 채팅방의 최대 수용 가능 인원을 입력하는 필드입니다. |
| FNC\_SVR\_011 | 생성 요청 버튼 | 입력 필드를 통해 정한 채팅방을 서버에 요청하는 버튼입니다. 입력 값이 올바르다면, 새로운 채팅방이 생성됩니다. |
| 로비화면 - 서버 로그 팝업 | FNC\_SVR\_012 | 로그 출력 영역 | 서버 내부에서 발생하는 로그를 화면에 보이는 영역입니다. 서버가 수신하는 메시지의 정보, 서버 내부에서 동작하는 과정과 결과, 특이사항이 로그로 표현될 수 있습니다. |
| FNC\_SVR\_013 | 로그 지우기 버튼 | 화면에 쌓인 로그를 삭제하는 버튼입니다. |
| FNC\_SVR\_014 | 로그 모으기 버튼 | 같은 종류의 로그들을 묶고 대신 개수를 표현하는 버튼입니다. 토글 형식으로 동작합니다. |
| FNC\_SVR\_015 | 로그 검색필드 | 필드에 입력한 문자열을 포함하는 로그만 선택적으로 나타나게 하는 필드입니다. |
| FNC\_SVR\_016 | 로그 타입 분류 버튼 | 발생한 로그의 종류에 따라 선택적으로 화면에 나타나게끔 하는 버튼입니다. |
| FNC\_SVR\_017 | 로그 팝업 닫기 | 팝업을 닫는 버튼입니다. |
| FNC\_SVR\_018 | 로그 팝업 스크롤바 | 로그의 개수가 많아 한 화면에 다 나타내지 못할 때 활성화되는 스크롤 바입니다. |
| FNC\_SVR\_019 | 로그 팝업 크기 조절 버튼 | 로그 팝업의 크기를 조절하는 버튼입니다. 버튼을 클릭한 채 드래그 함으로써 크기를 조절합니다. |
| 로비 화면 - 채팅 팝업 | FNC\_SVR\_020 | 채팅 화면 | 채팅방에서 오가는 메세지들을 화면에 나타내는 영역입니다 |
| FNC\_SVR\_021 | 채팅 종료 | 현재 채팅을 그만두고 채팅방에서 나가는 버튼입니다.. |
| FNC\_SVR\_022 | 채팅 메시지 전송 | 채팅창에 입력한 메시지를 채팅방에 전송. |
| FNC\_SVR\_023 | 이미지 버튼 | JPG, PNG 등 호스트의 저장장치에 저장되어 있는 이미지를 메시지에 첨부할 수 있는 버튼. OS의 파일 탐색기를 통해 원하는 이미지 파일을 첨부할 수 있습니다. |
| 클라이언트 | 연결 설정 화면 | FNC\_CLT\_001 | 서버 연결 설정 | 클라이언트가 서버에 접속하기 위해 연결을 설정하는 화면입니다. 서버의 IP주소, 포트 그리고 클라이언트가 사용할 이름을 설정할 수 있습니다. 공란으로 남겨놓으면 임의의 숫자와 함께 게스트 닉네임으로 설정됩니다 |
| 로비 화면 | FNC\_CLT\_002 | 로비 화면 | 기본 로비 화면입니다. 이 화면에서 새로운 채팅방을 만들거나, 채팅방에 참가할 수 있습니다. 서버의 채팅 방 목록과 연결된 유저 목록을 볼 수 있습니다. |
| FNC\_CLT\_003 | 채팅 방 목록 | 서버에서 현재 열려 있는 채팅방의 목록을 확인할 수 있는 영역입니다. 채팅방의 최대 인원수와 현재 인원수를 확인할 수 있는 간략한 정보도 함께 표시됩니다. 채팅방을 선택함으로써 해당 채팅방에 입장할 수 있습니다. 채팅방에 입장하면 채팅 방 내에서 채팅 할 수 있는 팝업이 불러와집니다. |
| FNC\_CLT\_004 | 채팅 방 만들기 버튼 | 사용자가 새로운 채팅방을 만들기 위한 화면입니다. 사용자는 채팅방의 제목, 참가 가능 인원 수를 설정할 수 있습니다. |
| FNC\_CLT\_005 | 새로 고침 버튼 | 서버로부터 채팅 방 목록을 새로 고침 합니다. |
| FNC\_CLT\_006 | 온라인 유저 목록 | 서버에 연결되어 있는 유저들을 보여주는 영역입니다. |
| 로비화면 – 채팅 방 생성 요청 팝업 | FNC\_CLT\_007 | 방 이름 입력필드 | 만들어질 채팅방의 이름을 입력하는 필드입니다. |
| FNC\_CLT\_008 | 방 최대인원 입력필드 | 만들어질 채팅방의 최대 수용 가능 인원을 입력하는 필드입니다. |
| FNC\_CLT\_009 | 생성 요청 버튼 | 입력 필드를 통해 정한 채팅방을 서버에 요청하는 버튼입니다. 입력 값이 올바르다면, 새로운 채팅방이 생성됩니다. |
| 로비 화면 - 채팅 팝업 | FNC\_CLT\_010 | 채팅 화면 | 채팅방에서 오가는 메세지들을 화면에 나타내는 영역입니다 |
| FNC\_CLT\_011 | 채팅 종료 | 현재 채팅을 그만두고 채팅방에서 나가는 기능. |
| FNC\_CLT\_012 | 채팅 메시지 전송 | 채팅창에 입력한 메시지를 채팅방에 전송. |
| FNC\_CLT\_013 | 이미지 버튼 | JPG, PNG 등 호스트의 저장장치에 저장되어 있는 이미지를 메시지에 첨부할 수 있는 버튼. OS의 파일 탐색기를 통해 원하는 이미지 파일을 첨부할 수 있습니다. |

3. 인터페이스 및 사용자 경험

|  |  |  |
| --- | --- | --- |
| 화면 이름 | 초기 설정 화면 | 서버의 연결 설정 화면 |
| 이미지 |  |  |
| 설명 | FNC\_INIT\_001 ~ FNC\_INIT\_003의 기능이 나타나는 화면 | FNC\_SVR\_001의 기능이 나타나는 화면 |
| 화면 이름 | 서버의 로비화면 | 서버의 로비화면 |
| 이미지 |  |  |
| 설명 | FNC\_SVR\_002~ FNC\_SVR\_009까지의 기능이 나타나는 화면.  위는 포트를 7777로 설정하고 서버를 첫 동작시킨 화면이다. | 클라이언트 ‘Guest43’을 서버에 연결시키고 난 후의 화면.  FNC\_SVR\_007은 5초에 한 번씩 서버로부터 온라인 클라이언트의 목록을 받아온다.  ‘@Server’은 서버를 나타낸다. |
| 화면 이름 | 서버의 로비화면 – 채팅 방 생성 요청 팝업 | 서버의 로비화면 |
| 이미지 |  |  |
| 설명 | FNC\_SVR\_009~ FNC\_SVR\_011의 기능을 나타내는 화면.  해당 팝업은 FNC\_SVR\_004 기능을 통해 불러왔다. | 채팅방의 이름을 ‘Room1’, 최대 인원을 5명으로 설정하고 서버에 요청을 보내 화면과 같이 생성된 모습이다.  채팅방을 생성한 후, 반드시 FNC\_SVR\_006을 통해 새롭게 목록을 불러와야 화면과 같이 나타낼 수 있다.  현재는 아무도 채팅방에 입장하지 않아, 채팅 방 인원이 0명인 것으로 나타난다. |

|  |  |
| --- | --- |
| 화면 이름 | 서버의 로비화면 – 채팅 팝업 |
| 이미지 |  |
| 설명 | Room1에서 서버와 클라이언트가 채팅 하는 화면. 왼쪽이 서버 프로그램이고, 오른쪽이 클라이언트 프로그램이다.  서버가 채팅방에 직접 입장해서 일반 클라이언트의 기능을 동일하게 수행할 수 있다.  FNC\_SVR\_020~ FNC\_SVR\_023기능과, FNC\_CLT\_010~ FNC\_CLT\_013의 기능을 수행한다. |
| 화면 이름 | \*의 로비화면 – 채팅 팝업 |
| 이미지 |  |
| 설명 | FNC\_SVR\_023기능을 사용하여 채팅방에 사진을 보내고자 하는 화면. 이미지 버튼을 통해 왼쪽과 같이 OS에서 지원하는 파일 탐색기를 열어 이미지 파일을 선택할 수 있다. 오른편의 화면은 현재 이미지가 적재되어 전송할 준비가 완료된 상태. ‘전송’버튼을 통해 이 이미지를 채팅방에 전송할 수 있다. |
| 화면 이름 | 서버의 로비화면 - 서버 로그 팝업 |
| 이미지 |  |
| 설명 | FNC\_SVR\_008기능을 통해 서버 로그 팝업을 화면으로 불러온 상태. FNC\_SVR\_012 ~ FNC\_SVR\_019의 기능을 오른편 화면처럼 사용할 수 있다. |

|  |  |  |
| --- | --- | --- |
| 화면 이름 | 서버의 로비화면 – 채팅 팝업 | 서버의 로비화면 – 채팅 팝업 |
| 이미지 |  |  |
| 설명 | 위와 같이 채팅방에 속해 있던 클라이언트가 채팅방에서 나갔을 때 자동으로 채팅방에 이를 알린다. | 위와 같이 채팅방에 속해 있던 모든 클라이언트가 나가 채팅방에 더 이상 아무도 존재하지 않게 되면, 자동으로 채팅방을 삭제한다. |
| 화면 이름 | 동시에 1개의 서버와 다수의 클라이언트가 동작하는 화면 | |
| 이미지 |  | |
| 설명 | 둘 이상의 클라이언트가 채팅방에서 채팅을 이어 나가는 모습 | |

4. 메시지 포맷

서버와 클라이언트 간에서 송수신하는 주요 메시지의 종류와 기능.

1. 서버에서 수신(=클라이언트에서 송신)

|  |  |  |
| --- | --- | --- |
| **포맷** | **설명** | **예시** |
| $"&NAME|{ClientName}" | 클라이언트가 서버에 연결할 때 보내는 메시지.  클라이언트의 이름을 담아서 보낸다 | $"&NAME|{Client1 }" |
| $"&CREATEROOM|{roomName}|{maxPlayers}" | 클라이언트가 채팅방의 생성을 요청하는 메시지.  채팅방의 이름과 최대 인원을 담아서 보낸다 | $"&CREATEROOM|Room1|5" |
| &CHAT|{roomName}|{message} | 클라이언트가 채팅방 내에서 보내는 텍스트 메시지.  채팅방의 이름과 메시지 내용을 담아서 보낸다. | &CHAT|Room1|hello |
| $"&IMAGE|{roomName}|{imageString}" | 클라이언트가 채팅방 내에서 보내는 이미지 메시지.  채팅방의 이름과 메시지 내용을 담아서 보낸다. | $"&IMAGE|Room1|AAQSkZjRg... " |
| $"&ENTER|{roomName}" | 클라이언트가 채팅방에 입장할 때 보내는 메시지  채팅방의 이름을 담아서 보낸다. | $"&ENTER|Room1" |
| $"&LEAVE|{roomName}" | 클라이언트가 채팅방에서 나갈 때 보내는 메시지  채팅방의 이름을 담아서 보낸다. | $"&LEAVE|Room1" |
| “&ROOMLIST” | 클라이언트가 서버에 존재하는 채팅방의 목록을 요청할 때 보내는 메시지 | “&ROOMLIST” |
| "&USERLIST" | 클라이언트가 서버에 연결된 유저의 목록을 요청할 때 보내는 메시지 | "&USERLIST" |

2. 클라이언트에서 수신(=서버에서 송신)

|  |  |  |
| --- | --- | --- |
| **포맷** | **설명** | **예시** |
| "%NAME" | 서버가 클라이언트와 연결할 때 보내는 메시지. 클라이언트는 이 메시지를 받으면 자신의 이름을 답장. | "%NAME" |
| “%ROOMLIST|{room.roomName}| {room.currentMemNum}|{room.MaxMemNum}|...” | 서버에 있는 채팅방의 정보를 목록화해서 보내는 메시지. 클라이언트는 이 메시지를 기반으로 자신의 로컬 채팅 방 목록을 새로 고침 한다. | “%ROOMLIST|Room1|2|  5|Room2|1|2|...” |
| "%USERLIST|{\_client.clientName}  |{\_client.clientName}|...” | 서버에 연결된 클라이언트를 목록화해서 보내는 메시지. 클라이언트는 이 메시지를 기반으로 자신의 로컬 유저 목록을 새로 고침 한다. | "%USERLIST|User1|User2|…” |
| $"%CHAT|{c.clientName}|{message}" | 동일한 채팅 방 소속의 다른 클라이언트들에게 텍스트 메시지를 브로드 캐스팅하는 메시지. | $"%CHAT|Client1|hello” |
| $"%IMAGE|{c.clientName}|{imageString}" | 동일한 채팅 방 소속의 다른 클라이언트들에게 이미지 메시지를 멀티 캐스팅하는 메시지. | $"%IMAGE|Client1|  AAQSkZjRg..." |
| $"%NOTI|{context}" | 클라이언트에서 화면에 나타내야 하는 텍스트를 전송하는 메시지. 채팅 방 생성됨, 삭제됨 과 같은 메시지를 담아 보낸다. | $"$"%NOTI|  Room1 방이 삭제되었습니다." |

5. 주요 데이터 항목

|  |  |  |
| --- | --- | --- |
| **항목명** | **Type** | **용도 및 설명** |
| tcpClients | List<TcpClient> | 클라이언트들에 대한 정보를 저장, 관리하기 위한 List.  TcpClient는 System.Net.Sockets에 의해 제공. |
| tcpDisconnectClients | List<TcpClient> | 연결이 끊긴 클라이언트들을 일시적으로 관리하기 위한 List. |
| tcpServer | TcpListener | 서버측에서 클라이언트의 연결 요청을 받아들이고, 클라이언트와 통신할 수 있는 소켓 객체를 반환한다.  System.Net.Sockets에 의해 제공. |
| tcpSocket | TcpClient | 클라이언트가 서버로부터 데이터를 송수신할 수 있도록 하는 객체. |
| udpClient | UdpClient | 원격 UDP 엔드포인트로 데이터를 전송, 혹은 수신할 수 있도록 하는 객체. |
| IPInput | String | 클라이언트가 연결하고자 하는 서버의 IP주소를 저장하는 String. |
| PortInput | String | 클라이언트가 연결하고자 하는 서버의 포트 번호를 저장하는 String. |

|  |  |  |  |
| --- | --- | --- | --- |
| **항목명** | **타입** | **용도 및 설명** | **예시** |
| tcpClients | List<ServerClient> | 연결된 모든 클라이언트를 저장하는 리스트 | tcpClients = new List<ServerClient>(); |
| tcpDisconnectClients | List<ServerClient> | 연결이 끊긴 클라이언트를 저장하는 리스트 | tcpDisconnectClients = new List<ServerClient>(); |
| Rooms | List<Room> | 모든 방(Room)들을 저장하는 리스트 | Rooms = new List<Room>(); |
| tcpServer | TcpListener | 실제 서버를 나타내는 TcpListener 객체 | tcpServer = new TcpListener(IPAddress.Any,port); |
| serverStarted | bool | 서버의 시작 여부를 나타내는 변수 | serverStarted = true; |
| imageSavePath | string | 이미지를 저장할 경로를 나타내는 문자열 | imageSavePath = @"C:\ProtoChatData\Server"; |
| ServerInternalClient | ServerClient | 서버 내부 클라이언트를 나타내는 ServerClient 객체 | ServerInternalClient = new ServerClient(TCPClient.instance.tcpSocket); |
| udpServer | UdpClient | UDP 서버를 나타내는 UdpClient 객체 | udpServer = new UdpClient(port); |
| udpClients | List<IPEndPoint> | 연결된 모든 클라이언트의 IP 주소와 포트 번호를 저장하는 리스트 | udpClients = new List<IPEndPoint>(); |
| udpDisconnectClients | List<IPEndPoint> | 연결이 끊긴 클라이언트의 IP 주소와 포트 번호를 저장하는 리스트 | udpDisconnectClients = new List<IPEndPoint>(); |
| tcpSocket | TcpClient | 서버와 통신할 TcpClient를 저장합니다. |  |
| LocalRooms | List<Room> | 로컬 채팅방 목록을 저장합니다. |  |
| imageSavePath | string | 이미지를 저장할 경로를 저장합니다. | "C:\ProtoChatData\Client" |

6. 동작 환경

Windows OS

6. 개발 환경

|  |  |  |
| --- | --- | --- |
| 개발 환경 | OS | Windows 11 64-bit |
| 개발 언어 | C# |
| IDE | Microsoft Visual Studio 2022  Unity |
| 주요 클래스(모듈)  기능 설명 | Room | 채팅 방을 구현하기 위해 직접 구현한 클래스.  방 이름, 현재 인원수, 최대 인원수, 소속 클라이언트를 멤버 변수로 갖고,  소속 멤버 관련 기능을 제공하기 위한 여러 메서드들을 갖고 있다.  e.g.) getMembers(),removeAllMember(),removeMember(), ... |
| System.Net | 통신과 관련된 다양한 작업을 수행하기 위한 클래스들을 제공.  e.g.) WebClient Class, SmtpClient Class, Dns Class, IP Address, … |
| System.Net.Sockets | 소켓 프로그래밍을 위한 클래스들을 제공.  e.g.) Socket Class, TcpListener/Client Class, UdpClient Class, … |
| System.IO | 파일과 데이터 스트림을 읽고 쓰는 데 필요한  클래스와 인터페이스들을 제공.  e.g.) FileStream Class, StreamReader/Writer Class, … |
| System.Collections.Generic | 기본 컬렉션 클래스들의 제네릭(Generic) 버전을 제공.  e.g.) List<T>, Queue<T>, Stack<T>, … |
| System.Threading | 다중 스레드 프로그래밍을 구현하기 위한 클래스와 기능을 제공.  e.g.) Task, Mutex, Monitor, Semaphore, ... |
| UnityEngine.UI | Unity 게임 엔진에서 사용되는 사용자 인터페이스(UI) 요소와 관련된 클래스와 함수들을 포함하는 네임스페이스  e.g.) Text, Image, Button, InputField, ... |