개발 완료 보고서

제출일 : 2023년 9월 08일

| 참여인원 | | 박영진 | | |
| --- | --- | --- | --- | --- |
| **개발프로젝트 소개** | | | | |
| **프로젝트 명** | 사내 메신저 프로그램 | | | |
| **활동일시** | 8 / 10 ~ 9 / 8 | | **장소** | 광주인력개발원 공학1관 융합1실 |
| **주요 주제** | C#, TCP 소켓 통신으로 사내 메신저 프로그램 구현 | | | |
| **개발목적** | 소켓통신과 윈폼 UI작업 숙달 | | | |
|
| **개발환경** | Windows OS / Visual Studio 2019 WinForm | | | |
|
| **요구분석서** |  | | | |
| **코드**  **설명** |  | 박영진  [서버]  **Class** TcpSocket   * 소켓 연결을 비롯한 클라이언트와   송수신 작업이 이루어지는 클래스   * **public async Task StartServerAsync()**   + 최대 대기 큐 크기를 설정하고   반복문에서 비동기로 클라이언트와  연결을 대기 및 수락   * **private async Task HandleClientAsync(Socket clientSocket)**   + 반복문으로 데이터를 수신받아 클라이언트가 요청한 데이터에 맞는   메서드를 실행시키는 메서드   * **private async Task LoginData(Socket clientSocket)**   + 수신받은 ID에 해당하는 PW를 찾고   결과를 체크한다.   * **private async Task LoadEmpList(Socket clientSocket)**   + DB에서 가져온 사원목록을 클라이언트에게 보내준다. * **private async Task LoadRoomList(Socket clientSocket)**   + DB에서 가져온 채팅방 목록을 클라이언트에게 보내준다. * **private async Task ChatRoutine(Socket clientSocket)**   + 채팅에 필요한 데이터를 수신받아 가공, 기록한 뒤   채팅방에 속한 인원에게 전달한다.  **private void WriteChat(string index, string id, string msg)**   * **채팅방의 대화를 파일에 기록한다.**   **private async Task SendChat(string index, string id, string msg)**   * **ChatRoutine에서 호출되어 서버에 접속중이면서**   **채팅이 보내진 채팅방에 속한 인원에게 채팅을 전송한다.**  **private async Task JoinRoom(Socket clientSocket)**   * **채팅방에 들어가면 접속중인 유저 목록에 정보를 추가하고**   **대화기록을 보내준다.**  **private async Task CreateRoom(Socket clientSocket)**   * **중복을 체크하고 채팅방을 생성한다.**   **private async Task Register(Socket clientSocket)**   * **회원가입 정보를 수신받아 DB에 INSERT한다.**   **private async Task ListCheck(Socket clientSocket)**   * **채팅방 목록을 확인하고 갱신해준다.**   **private async Task SendData<T>(Socket clientSocket, T data)**   * **인자로 전달받은 data에 따라 바이트 변환 및 헤더를 생성하고**   **헤더, 데이터 순서로 클라이언트에게 전송한다.**  **private async Task<(string type ,byte[] data)> RecvData(Socket clientSocket)**   * **고정된 헤더 크기로 헤더를 받고 헤더를 가공해 얻은**   **데이터의 크기만큼 데이터를 수신받는다.**  **private void RemoveClient(Socket clientSocket)**   * **소켓을 관리하는 딕셔너리에서 연결이 끊긴 소켓을 제거한다.**   **private async Task Leave(Socket clientSocket)**   * **채팅방을 나갈때 접속목록을 관리하는 리스트에서 제거해준다.**   **class DBClass**   * **DB에 관련된 동작을 하는 클래스**   **class Converter**   * **데이터를 바이트나 json 등으로 변환하거나 바이트를 데이터로 변환하기 위한 클래스**   **[클라이언트]**  **class TcpClass**   * 소켓 연결을 비롯한 서버에 데이터를 요청하는 등   송수신 작업이 이루어지는 클래스  public partial class MdiParents : Form   * **각 페이지들을 담기 위한 Mdi 부모 폼** | | |
| **후기** | 박영진 | C에비해 C#은 알아서 지원해주는게 많아서  네트워크 통신이 비교적 쉬운편이었다.  자잘한 에러 등은 생겼지만 생각한대로 된 코드가 많아서 좋았었고  막힌 문제들을 해결하면서 실력이 많이 늘어난 것 같다.  처음부터 설계를 잘 하지 못해서 나중에 잘 수정한 부분도 많았는데  서버에서 클래스 분할이나 SaveFileDialog문제를 잘 하지 못해 아쉽다.  문제가 발생한 것들 중에 하루 이상 걸린 어려운 문제도 있었는데  기간에 맞추기 위해 빠르게 넘기지 않고 붙잡고 있던 단점이  이번에는 오히려 도움이 되어 잘 해결한 것 같다. | | |
| **비고** |  |  | | |