~3월 26일전까지

1. 서버 정리
   1. 네트워크
      1. TCP, UDP
      2. 동기, 비동기
      3. UTF-8, UTF-16
   2. 클라 to 클라, only 서버
   3. 데이터베이스
   4. 더미 클라이언트
   5. 하드웨어
   6. 보안
   7. GUI
      1. 로그
      2. WPF, WinForm
   8. 안정성
      1. 동기화
      2. 동시성
   9. 서버 분할
2. 서버 개발 순서 정의
   1. 네트워크 정의
      1. 비동기 TCP
   2. 클라이언트와 네트워크 연결(echo)
   3. 클라이언트와 메시지를 전송하기 위한 패킷과 프로토콜 설정
   4. 많은 유저들을 관리하기위한 환경 구축
   5. 더미 클라이언트를 만들어 테스트환경 구축

~ 03. 26

1. 개발과정
   1. 네트워크 연결 클래스 생성
      1. TcpListener를 테스트위해 에코서버와 에코클라이언트 생성
      2. 비동기 문제로(?) 직접 만들기로 결정
      3. 네트워크 소켓 클래스 생성(NetworkService)
      4. 각 서버에서 유저를 받기 위한 리스너 생성(Listener)
      5. 소켓의 네트워크 통신 버퍼를 관리하기 위한 클래스 생성 (BufferManager)
   2. 서버에 대량 접속으로 인한 대량의 소켓의 생성과 소멸로 병목발생 예상
      1. 초기 구동시에 소켓을 생성 및 재활용하여 직접 관리(SocketAsyncEventArgsPool)
   3. 소켓 관리 클래스 생성서버와 클라이언트가 연락하기 위한 패킷 구축(Packet)
      1. 패킷 인코딩과 디코딩을 위한 번역 클래스 생성(MessageTranslator)
      2. 패킷의 대량 생성과 소멸로 병목발생 예상
         1. 패킷 관리자 생성(PacketBufferManager)
   4. 접속한 유저마다 각각 통신하는 클래스 생성(UserToken)
   5. 접속한 유저의 대응하는 역할을 규정하는 인터페이스 생성(IPeer)
   6. 서버에서 해당 유저의 통신한 패킷을 처리하는 클래스 생성(GameUser : IPeer)
   7. 클라에서 서버에게 받은 패킷을 처리하는 클래스 생성(RemoteServerPeer : IPeer) 03. 27 ~ 03. 31
2. 더미 클라이언트 생성
3. 1:1 접속에서 다중 접속으로 변경
4. 로그인 프로토콜 생성
5. 리펙터링

04. 01 ~ 04. 07

1. 클라이언트와 서버개발 독립적으로 가능하도록 설정
2. 유니티에서 씬(Scene) 전환시 네트워크가 유지되게 설정
3. 데이터베이스 관련 클래스 생성
4. 통신 종류 프로토콜에 정의(Single, Multi)
5. 각 캐릭터마다 식별방법 필요
6. 리펙터링

04. 08 ~ 04. 14

1. 접속한 유저목록을 관리하는 싱글클래스 생성(UserList)
2. 리펙터링
   1. MessageTranslator에서 콜백 제거
3. 서버 개발 다음 일정 추가
4. 서버 분산시 방법 정의
5. 캐릭터 이동 프로토콜 추가

04. 15 ~ 04. 21

1. 리펙터링
   1. 디렉터리 정리
   2. 중복 코드 제거
   3. 불필요한 주석 제거

04. 22 ~ 04. 28

1. 리펙터링
   1. 데이터베이스 연동 클래스와 데이터에베이스 클래스 분리

04. 29 ~ 05. 05

1. 버그 수정
   1. 전송 패킷 생성 오류 수정
   2. 호환성 문제 해결
   3. 키를 입력할 때 여러 캐릭터가 동시에 움직이는 문제 수정
2. 캐릭터연동을 위해 고유의Id를 생성하는 클래스 생성(NumberingWaitObj)
3. 콘솔이 아닌 GUI로 모니터링을 하기로 결정(WPF)
4. 여러 캐릭터 생성

05. 06 ~ 05. 12

1. 버그 수정
   1. 카메라 고정문제 수정
2. 모니터링에 사용할 CPU, RAM 이용률 추가
3. 콘솔과 GUI코드의 혼재로 코드 분리

05. 13 ~ 05. 19

1. 버그 수정
   1. 먼저 접속한 유저가 안보이던 현상 수정
2. UI 레이아웃 설정
3. CPU, RAM 이용률 GUI에 추가
4. 서버 접속 시 서로 캐릭터 생성
5. 깃헙 다이어트