Maly opis tego co mi się udalo wymyslic, wszystkie klasy znajduja się w paczce Temps.Audio:

1. ChatCapture.java

Nic ciekawego tu się nie dzieje- sluzy do pobrania informacji o urzadzeniach na kompie, będzie dzialac bez problemow jeżeli zainstaluje się jmf i do projektu podebnie się jarki które on stworzy- sa w folderze instalacji jmf w folderze lib

1. ChatFrame.java

Wiadomo- GUI tego programu, był w jakims dziwnym jezyku i formatowanie sobie nie dalo z tym rady- stad te kwadraciki i tym podobne sprawy.

1. ChatLoginFrame.java

Ekran logowania dla aplikacji, ponownie zobaczymy kwadraciki…

1. ChatServer.java

Serwer aplikacji, czyli caly wywolywacz odpowiednich metod, rozdziela akcje w switchu który nie ma komentarzy wiec ciezko go zajarzyc. Zmienna num jest zmienna sterujaca wysylana w tzw. ctrlStreamie czyli pasmie kontroli dla aplikacji. Wiecej o tym w nastepnych klasach.

1. RTPSocketAdapter.java

Klasa implementujaca wymagany interfejs o którym możecie poczytac w API javy. Tutaj mamy implementacji wpisywania i wczytywania bajtow z socketa (na koncu sa dwie klasy wewnetrzne). Tutaj tez będzie trzeba podpiac szyfrowanie do tablicy bajtow. Jedna z niewielu klas gdzie sa czytelne komentarze wiec to warto poczytac. Klasa PushSourceStream będzie wpisywac (wysylac) bajty po sokecie. Wywolanie tej metody jest sterowane eventami które sa gdzies gleboko zaszyte…

1. TraAndRec.java

Najbardziej pogieta klasa ale chyba najbardziej potrzebna do rozkminienia, tutaj dzieje się cala inicjalizcja i gdzies tutaj musza być tworzone eventy które startuja wysylanie i otrzymywanie bajtow. Implementuje listenera który odtwarza sesje otrzymana, (ReceiveStreamListener), SessionListenera (nie wiem do czego…) i ControllerListener do obslugi zdarzen podczas zmiany stanu playera i procesorow (metoda controllerUpdate)

1. UDPReceive.java i UDPTransmit.java

Klasy implementujace sokety UDP odpowiednio odbierajace i wysylajace bajty, uzywane sa w RTPSocketAdapter i w TraAndRec. Mysle ze można je przyjac w naszej aplikacji w niezmienionej (lub nieznacznie zmienionej) formie.