class AuthorizedUser {

- username : string

- password : string

+Authorize(username : string, password : string);

+ LogOut()

+ GetUsername()

- SetUsername(username : string)

- GetPassword()

- SetPassword (password : string)

}

class NonAuthorizedUser {

+ Register(username : string, password : string)

}

class Comment {

+ author : AuthorizedUser

+ text : string

+ date : Date

}

class Score {

+ name : string

+ author : string

+ scoreFile : File

+ complexity : int

- comments : Comment(n)

+ GetComment(c : Comment)

}

class ScoreController {

# score : Score

+ currentInstrument : int

+ SelectInstrument(instrumentId : int);

+ SelectScore(score : Score);

}

class ScoreViewer {

+ viewMode : ViewMode;

+ ActionId : int

+ AddedComment()

+ ThrowError()

+ ThrowedError()

+ SelectedViewMode()

+ SelectVizualPartiturType()

+ SelectedVizualPartiturType()

+ SelectStandartView()

+ SelectedStandartView()

+ ShowScore()

+ GetInput()

+ AddComment(user : AuthorizedUser, text : string);

+ SelectViewMode(viewMode : ViewMode);

+ Exit()

}

class ScoreRedactor {

+ ImportScoreFromFile(scoreFile : File)

+ Imported()

+ ExportScore();

- Exported()

+ ChangeNoteDuration(notePosition : int, instrument : int, duration : int);

+ ChangedNoteDuration()

+ AddEffect(notePosition : int, effectIndex : int);

+ AddedEffect()

+ AddEpisodeTitle(tactPosition : int, title : string);

+ AddedEpisode()

+ ChangeTempo(tactPosition : int, tempoBPM : float);

+ ChangedTempo()

+ ChangeTimeSignature(tactPosition : int, numerator : int, denominator : int);

+ ChangedTime()

+ReturnToRedactor()

+ Exit()

}

class ScorePlayer {

+ Play();

+ Played()

+ Pause();

+ Paused()

+ MuteInstrument(instrumentId : int);

+ InstrumentMuted()

+ SoloInstrument(instrumentId : int);

+ InstrumentIsolated()

+ ReturnToPLayer()

+ Exit()

}