


```
+ ChangeAndNotify( userInput ) :void
+ closeScanner() :void
+ getCurrentPlayer() :Player
+ getMenu() :ClientMenu
+ getModel() :Model
+ getViewID() :InetAddress
+ isRunning() :boolean
+ run() :void
+ setCurrentPlayer( Player ) :void
+ setMatches( List<OnlineGameInfo> ) :void
+ setViewID( InetAddress ) :void
# showPlayerInfo( Player ) :String
+ update( Observable, Object ) :void
+ View()
+ visitEvent( EndOfTurnEvent ) :void
+ visitEvent( EndGameEvent ) :void
+ visitEvent( IllegalActionEvent ) :void
+ visitEvent( IllegalPlayerEvent ) :void
+ visitEvent( IOErrorEvent ) :void
+ visitEvent( ItemRequestEvent ) :void
+ visitEvent( StartPlayerTurnEvent ) :void
+ visitEvent( SuccessfulAttackEvent ) :void
+ visitEvent( SuccessfulItemAdditionEvent ) :void
+ visitEvent( SuccessfulMoveOnDangerousSectorEvent ) :void
+ visitEvent( SuccessfulMoveOnEscapePodSectorEvent ) :void
+ visitEvent( SuccessfulMoveOnSafeSectorEvent ) :void
+ visitEvent( TooMuchItemsEvent ) :void
+ visitEvent( UnableToAttackEvent ) :void
+ visitEvent( GreenEscapePodEvent ) :void
+ visitEvent( NoiseInAnySectorEvent ) :void
+ visitEvent( NoiseInYourSectorEvent ) :void
+ visitEvent( RedEscapePodEvent ) :void
+ visitEvent( SilenceEvent ) :void
+ visitEvent( SuccessfulGivenItemListEvent ) :void
+ visitEvent( SuccessfulMoveOnStartingSectorEvent ) :void
+ visitEvent( SuccessfulMoveOnWallSectorEvent ) :void
+ visitEvent( SuccessfulUseOfItemEvent ) :void
+ visitEvent( SuccessfulUseOfSpotLightItemEvent ) :void
+ visitEvent( GameStartedEvent ) :void
+ visitEvent( ConnectedToGameEvent ) :void
+ visitEvent( LogPrintEvent ) :void
+ writeMessage( String, boolean ) :void
```