Complete Turn

```
Actor    :Client    :FacadeController    :UserController    :Database    :UserProfile    :MatchController    :Match
```

Actor → :Client : View Current Turns

:Client → :FacadeController : getCurrentTurns(ID)

:FacadeController → :UserController : getCurrentTurns(ID)

:UserController → :Database : getProfile(ID)

:Database ⇢ :UserController : user_profile

:UserController → :UserProfile : getTurns()

:UserProfile ⇢ :UserController : turns

:UserController ⇢ :FacadeController : turns

:FacadeController ⇢ :Client : turns

:Client → Actor : display(turns)

Actor → :Client : Submit Turn

:Client → :FacadeController : processTurn(turn)

:FacadeController → :MatchController : processTurn(turn)

:MatchController → :Match : validMove(from, to, player)

Alt

Valid Move

:MatchController → :Match : update(from, to)

:MatchController → :Database : update(match)

:FacadeController → :MatchController : getTurn()

:MatchController ⇢ :Turn : nextTurn()

:MatchController ⇢ :FacadeController : turn

:FacadeController → :UserController : addTurn(ID, turn)

:UserController → :Database : getProfile(ID)

:Database ⇢ :UserController : user_profile

:UserController → :UserProfile : addTurn(turn)

:UserController → :Database : updateProfile(profile)

:Client → Actor : notify(msg)

Invalid Move

:Client → Actor : notify(msg)