

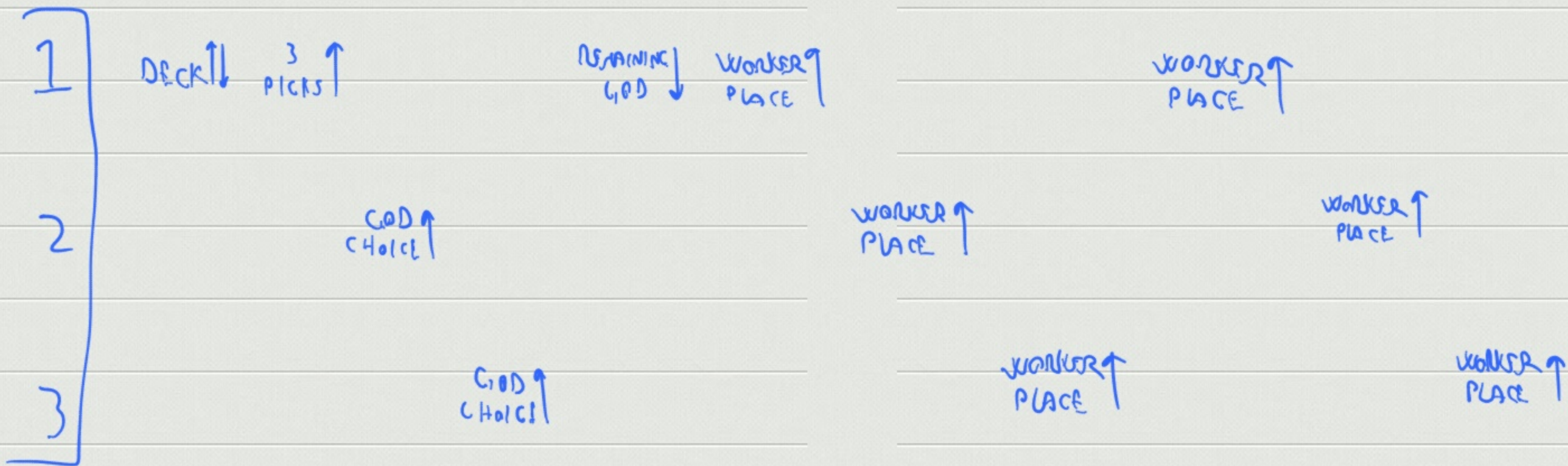
CONNECTION

1 ID↑↓ ^{NUM}PLAYERS↑ NAME↑

2 ID↑↓ ^{NUM}PLAYERS↑↓ NAME↑

3 ID↑↓ ^{NUM}PLAYERS↑↓ NAME↑

SETUP



1 } ACTIONRES \updownarrow SELECT \uparrow ACTION \uparrow ACTION \uparrow ACTION \uparrow

2 }

3 }

NEWSFEEDS

PLAYERS	ACTIONS	GAME	REQUESTS
NAME GOD QUIT CURRENTLY PLAYING ID FAIL	PLACE SELECT MOVE BUILD MOVE WITH FORCE • ACTIONT RSS	WIN LOSE NUM PLAYERS PICKED GODS • DECK	ID NUM PLAYERS DECK ACTIONT RSS

PUBLIC

MAKE
PRIVATE

{
ACTIONS FEED
GAME FEED
PLAYERS FEED
{ REQUESTS CHANNEL

ACTIONS ARCHIVE

GAME ARCHIVE

PLAYERS ARCHIVE

PERMISSIONS DI
RECONSTRUCTURE
PARTITA

VIEW implements PLAYER'S OBSERVER, GAME OBSERVER, ACTIONS OBSERVER

PUBLIC INTERFACE GAME OBSERVER {

PUBLIC VOID UPDATE WIN (...);

PUBLIC VOID UPDATE LOSS (...);

⋮

}

ObservableActions extends Observable {

@Override

public void addObserver(...)

AGGIUNGE UN ACTIONSOBSERVER ALLA LISTA

}

@Override

public void notifyObservers(...)

CHIAMATA ACTIONSUPDATE(...) SU

OGNI ACTIONSOBSERVER

}


```
PUBLIC VOID NOTIFY (VIEW V, ...) {
```

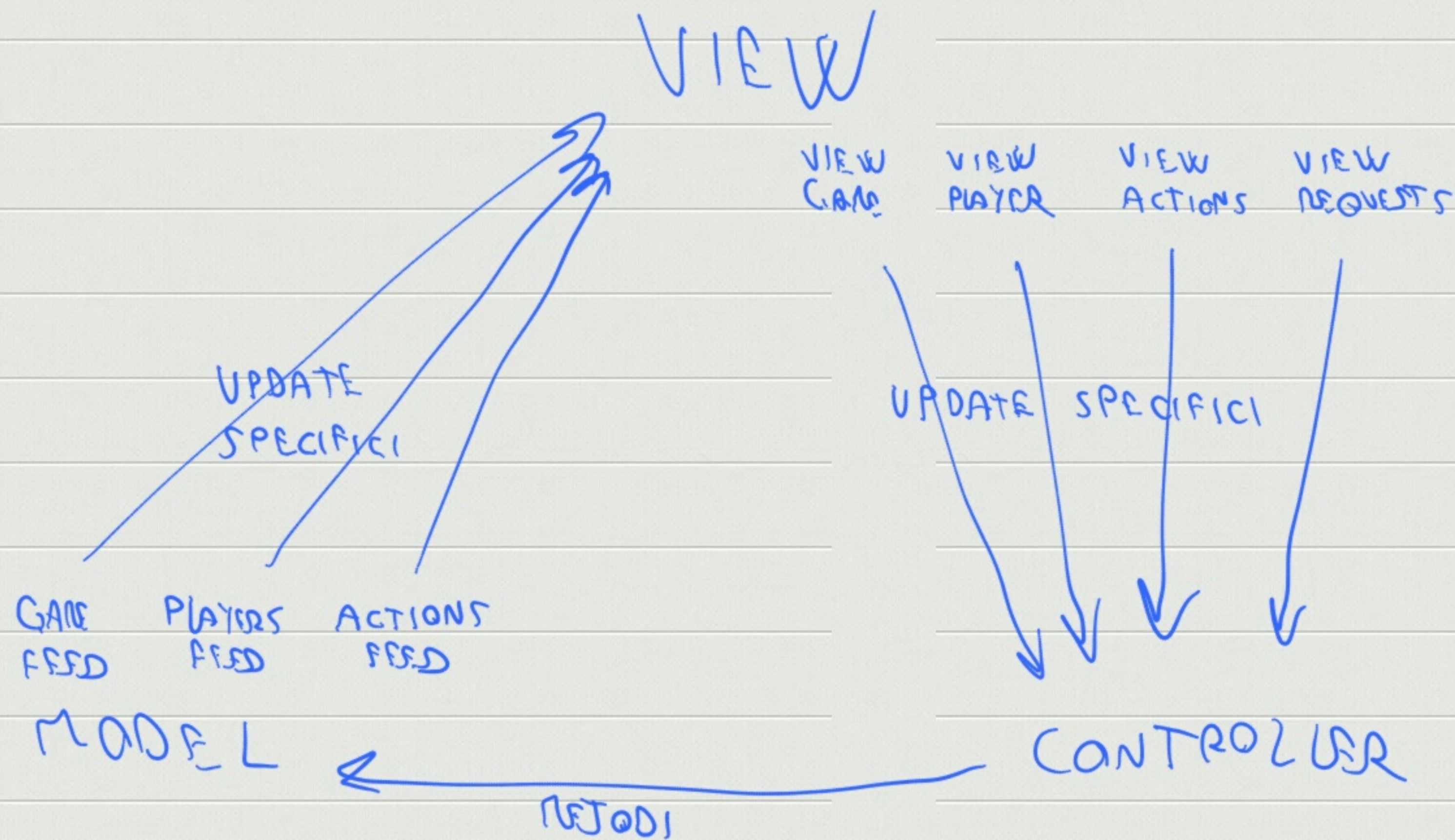
CHIAMA ACTIONS UPDATE SOLO SU V.

AVENDO REMPLIMENTATO ADDOBSERVER

ABBIAMO CONTROLLO SULLA LISTA DI OBSERVERS

```
}
```

```
}
```

ANCHE INSERIRE
OBSERVE SPECIFIC IMPLEMENTABILI TRAMITE INTERFACCIA
AD HOC CHE PRESENTA LE UPDATE SEPARATE

PRESERVARE MULTIPLI DI VIEW IMPLEMENTABILI ANCHE TRAMITE
ESTENSIONE DI CLASSE AD HOC CHE PRESENTA LE NOTIFY SEPARATE