

### 3f. Verifica di Qualità dello Schema:

SQUADRA(nome, codSfida<sup>SFIDA</sup>, numDadi, punteggioAttuale, nomeIcona<sup>ICONA</sup>)

nome, codSfida -> numDadi, punteggioAttuale, nomeIcona

ICONA(nomeIcona, nomeSet<sup>SET\_ICONE</sup>)

nomeIcona -> nomeSet

PARTECIPA(nome<sup>SQUADRA</sup>, codSfida<sup>SFIDA</sup>, nickname<sup>UTENTE</sup>)

CASELLA\_PODIO(x, y, codGioco<sup>GIOCO</sup>, nome<sup>SQUADRA</sup>, codSfida<sup>SFIDA</sup>, posto)

x, y, codGioco -> nome, codSfida, posto

TURNO\_DI\_GIOCO(codTurno, nome<sup>SQUADRA</sup>, codSfida<sup>SQUADRA</sup>, codCasella<sup>CASELLA\_GIOCO</sup>)

codTurno, nome, codSfida -> codCasella

DADO(codDado, nome<sup>SQUADRA</sup>, codSfida<sup>SQUADRA</sup>, valore, codGioco<sup>GIOCO</sup>)

codDado -> nome, codSfida, valore, codGioco

SFIDA(codSfida, dataOra, durataMax, codGioco<sup>GIOCO</sup>, moderata)

codSfida -> dataOra, durataMax, codGioco, moderata

CASELLA\_GIOCO(codCasella, x, y, numero, codGioco<sup>GIOCO</sup>, video<sub>o</sub>, tipologia, codCasellaArrivo<sup>CASELLA\_GIOCO</sup>)

codCasella -> x, y, numero, codGioco, video, tipologia, codCasellaArrivo

x, y, numero, codGioco -> codCasella, video, tipologia, codCasellaArrivo

GIOCO(codGioco, dadiRichiesti, maxSquadre, planciaDiGioco, nomeSet<sup>SET\_ICONE</sup>)

codGioco -> dadiRichiesti, maxSquadre, planciaDiGioco, nomeSet

SET\_ICONE(nomeSet, tema)

nomeSet -> tema

QUIZ(codQuiz, testoHtml, punteggio, tempoMassimo, immagine<sub>o</sub>, codCasella<sup>CASELLA\_GIOCO</sup>)

codQuiz -> testoHtml, punteggio, tempoMassimo, immagine, codCasella

testoHtml -> codQuiz, punteggio, tempoMassimo, immagine, codCasella

TASK(codTask, *testoHtml*, punteggio, tempoMassimo, codCasella<sup>CASELLA\_GIOCO</sup>)

codTask -> testoHtml, punteggio, tempoMassimo, codCasella

testoHtml -> codTask, punteggio, tempoMassimo, codCasella

RISPOSTA\_QUIZ(codRisp, *testoHtml*, immagine<sub>o</sub>, *codQuiz*<sup>QUIZ</sup>, nicknameModeratore<sup>MODERATORE</sup><sub>o</sub>, corretta, scelta<sub>o</sub>)

codRisp -> testoHtml, immagine, codQuiz, nicknameModeratore, corretta, scelta

testoHtml, codQuiz -> codRisp, immagine, nicknameModeratore, corretta, scelta

RISPOSTA\_TASK(idFile, codTask<sup>TASK</sup>, nicknameModeratore<sup>MODERATORE</sup><sub>o</sub>, codAdmin<sup>ADMIN</sup>, corretta, scelta<sub>o</sub>)

idFile -> codTask, nicknameModeratore, codAdmin, corretta, scelta

ADMIN(codAdmin)

UTENTE(nickname, eMail, nome<sub>o</sub>, cognome<sub>o</sub>, dataN<sub>o</sub>)

nickname -> eMail, nome, cognome, dataN

MODERATORE(nickname<sup>UTENTE</sup>)

GIOCATORE(nickname<sup>UTENTE</sup>, codRisp<sup>RISPOSTA\_QUIZ</sup><sub>o</sub>, idFile<sup>RISPOSTA\_TASK</sup><sub>o</sub>)

nickname -> codRisp, idFile

Tutte le parti sinistre delle dipendenze funzionali sono chiavi candidate per le relazioni.

Lo schema è quindi già normalizzato secondo la forma normale di Boyce-Codd.

### 3f (bis). Eventuali ottimizzazioni applicate tenendo in considerazione il carico di lavoro:

1. Abbiamo inserito l'attributo "punteggioAttuale" in Squadra, inserendo ridondanza, ma semplificando le operazioni di calcolo dei punteggi nei vari turni di gioco; questo ha semplificato di molto il nostro lavoro nel trigger 4b.

2. Abbiamo inserito l'attributo "dadiRichiesti" in Gioco per semplificare un'interrogazione nel carico di lavoro.
3. Abbiamo posto codTurno come SERIAL per gestire il secondo trigger con più facilità; questo ci dà infatti la possibilità di riferirci al turno precedente.