# Algemeen

* Denken hoe de scorebord weet hoe een game start
* Mogelijkheid om team naam te veranderen na inschrijven - requirement.
  + Bij team deny, client terug naar homescreen.
* Start round en select question buttons
* 3 component bij wachten op quiz master: undefined, goed, fout

# Interactions

* Fetch op laden van nieuwe pagina, of van scherm wisselen wanneer er nieuwe data binnen is. Of data doorgeven als props. Op redux is dit allemaal in een store. Component met een switch voor elke state
* websocket naar quiz\_master met team\_ready geen goed idee
  + post request naar de server
  + server heeft team ontvangen
  + Quizmaster moet weten dat er een team komt
    - express route handler stuurt websocket berichten naar de websocket
    - server stuurt websocket bericht naar quizzer
    - quizmaster krijgt een signaal dat een nieuwe team krijgt. Quiz master doet dan een get request
    - Dus: Alleen vanuit server route handler een websocket message met instructie sturen. Quiz master doet http request om teams op te halen
    - Teams opslaan in database op moment dat quiz master de team accepteert.

# Components

* Scorebord in een container component
  + Container component = Losse component bij elkaar voegen. Voordeel is herbruikbaarheid bij herhaling. Een router moet je met een component hangen.
* Eindscherm scorebord alle plekken andere componenten van maken, ook voor herbruikbaarheid.

# Database

* Categorieën en vragen
  + Categorieën in een string gooien in plaats van een aparte collection.
* Quiz
  + Bijhouden welk antwoord de deelnemer heeft gegeven.
  + Punten bijhouden voor elk team.
  + Huidige vraag kopiëren in plaats van linken naar een collection.