Notities sprint review:

## Float out ontwerp:

* Lijkt op een tabel
* Is een feite ook een tree
* Voordeel is dat het het scherm niet vult
* Nadeel is dat zonder uitgebreide breadcrumb het overzicht wordt verloren
* Stel een document met veel arrays en objecten dan verlies je het overzicht
* Breid de breadcrumb uit met document properties voor meer overzicht
* Ontwerp is wel leuk en mooi
* Leuk is dat de vergelijking met sql er niet is maar wel vergelijking met verkenner
* Nadeel is dat het totaalplaatje is verloren, vervelend als je iets snel wilt opzoeken

## Reddit ontwerp

* Lijkt onoverzichtelijk als je snel wilt wisselen tussen collecties/andere dingen
* Je moet steeds door de hele tree om ergens anders te komen
* Heel simpel
* Alles plakt aan elkaar
* Eerste was leuker

## Dropdown

* Prima en mooi
* Heeft ook nog iets van breadcrumb/header nodig
* Voordeel is dat het hele scherm beschikbaar is
* Het overzicht van databases en collecties is niet continu in zicht en bespaart ruimte
* Mooiste zou zijn als je binnen het scherm kan wisselen tussen treeview en finderview
* Finder gaat minder om de inhoud
* Denk na over een preview
* Finder kan (waarschijnlijk) comfortabeler en makkelijker updaten
* Finder mist overzicht

## Andere puntjes:

Verschillende gebruikers willen verschillende soort overzicht

In een later stadium pas verder nadenken over soorten overzicht

Hoe worden lange teksten weergegeven?

Een treeview houdt beter overzicht en kan meer open hebben dan een finderview

Een treeview kan beter met lange teksten omgaan dan een finderview

Het nadeel dat scherm het vol is met treeview is voor sommige een voordeel

De finderview uitbreiden tot aan de hosts wordt onoverzichtelijk

Het is mooier als het content gedeelte collecties bevatten

Mooiste is al er meerdere soorten overzicht zijn en een keuze geeft met bijvoorbeeld een toggle

Overzicht button die aangeeft waar in het overzicht je zit zou mooi zijn

JSONview bij couchdb geeft al een stuk overzicht

Advies: zoek naar voorbeelden of breid componenten uit.

Serieus naar elk alternatief kijken

Maandag beslissen welke optie

## Mongo verbinding

nodeJs verbind met mongo server via mongo client

clientApp -> <- server -> <-database

connectie met mongo heeft 3 dingen nodig: protocol, server (ip/domeinnaam), port

Er hoeft niks speciaals geprogrammeerd te worden voor verbinding met mongoserver

Connectie heeft misschien username en password nodig

Misschien een invoer scherm maken met default waardes localhost 27017

## User stories:

Het doel van data aanpassen is ‘zodat ik document kan aanpassen’

Ziet er prima uit

Acceptatie criteria:

1. Goed
2. Voorlopig alleen mongodb databases
3. Goed

Maandag 11:00 sprint planning c3xx

na de 2e sprint een basis hebben