**Omgaan met ontbrekende properties**

1. **Het veld weglaten**

* **Beschrijving**
  + Bij deze aanpak wordt de property helemaal niet opgeslagen in het document als er geen waarde voor beschikbaar is.
* **Voordelen**
  + Database blijft schoon: er staan geen velden in zonder betekenis
  + Minder opslagruimte nodig, omdat er geen placeholders of nulls worden opgeslagen
  + Query’s kunnen makkelijk gebruik maken van **$exists** om filteren op het bestaan van velden.
  + Past bij het schemaless karakter van MongoDB: documenten bevatten alleen de relevante velden
* **Nadelen**
  + Frontend of API moet altijd controleren of het veld bestaat of niet.
  + Externe systemen of clients weten niet automatisch dat een veld optioneel is.
  + Aggregaties of complexere queries kunnen extra chekcs vereisen, bijvoorbeeld **$ifNull** of **$exists**.
* **Tip**
  + Deze aanpak werkt het beste wanneer de frontend of API defensief omgaan met ontbrekende velden of wanneer je echt alleen velden wilt opslaan die een waarde hebben.

1. **Het veld op null zetten**

* **Beschrijving**
  + Het veld word hierbij wel opgenomen in het document, maar krijgt de waarde **null** als er geen echte waarde voor beschikbaar is.
* **Voordelen**
  + Consistentie documentstructuur: frontend kan altijd rekenen op hetzelfde veld.
  + Typehandling in frontend-frameworks zoals React of Vue wordt makkelijker, omdat het veld altijd aanwezig is.
  + Query’s zijn eenvoudiger: filters zoals **{ property: null}** werken direct.
* **Nadelen**
  + Frontend moet nog steeds een fallback tonen, **null** wordt niet automatisch weergegeven.
  + Sematisch moet duidelijk zij dat null betekend “geen waarden”.
* **Tip**
  + Gebruik **null** intern in de database, en laat de frontend of API bij weergaven een placeholder tonen zodat gebruikers de null niet krijgen te zien

1. **Placeholder waarde gebruiken (bijvoorbeeld “NVT”)**

* **Beschrijving**
  + In dit geval wordt er een string zoals **“NVT”** opgeslagen als er geen echte waarde beschikbaar is.
* **Voordelen**
  + Frontend kan direct tonen zonder extra checks te moeten uitvoeren.
  + Eenvoudig wanneer je nooit nulls wilt verwerken in je code.
* **Nadelen**
  + Sematisch incorrect: **“NVT”** is wél een waarde, terwijl er eigenlijk geen echte data is.
  + Query’s worden lastiger: je moet filteren op **“NVT”** en op **null/afwezigheid**
  + Kan verwarrend worden als later een echte waarde **“NVT”** kan voorkomen.
  + Opslag wordt iets groter, en filters/aggregaties worden ingewikkelder.
* **Tip**
  + Gebruik een placeholder enkel voor presentatie of rapportage, niet als primaire opslag in een database. Intern is **null** beter, en de placeholder kan dynamisch bij het ophalen worden toegevoegd.

**Conclusie**

* Voor een **schone, toekomstbestendige database**: gebruik null voor ontbrekende waarde.
* Voor **webapplicatie-weergave:** laat de frontend een placeholder zoals “NVT” tonen.
* Het veld **helemaal weglaten** kan ook, maar vereist meer checks in latere stadia.