

Hieronder geef ik een gedetailleerde beschrijving van de tabelstructuren, inhoud, onderlinge verbindingen en instructies die je in een PDF-bestand kunt opnemen voor gebruik met Cursor AI en Claude 3.7 Sonnet. Omdat ik als Grok geen PDF kan genereren, zal ik de inhoud structureren als platte tekst die je zelf in een PDF kunt kopiëren (bijv. via een tekstverwerker zoals Word of Google Docs). Vervolgens kun je dit document uploaden naar `@docs` in Cursor AI.

---

### ### Inhoud voor het PDF-Document: Overzicht Database Structuur en Instructies

#### #### Titel: Database Structuur en Instructies voor Cursor AI (Claude 3.7 Sonnet)

#### #### 1. Introductie

Dit document beschrijft de database-structuur van een verzamelingssysteem met drie hoofdcategorieën: **Sports**, **TCG (Trading Card Games)**, en **Entertainment**. Elke categorie heeft een set tabellen die informatie opslaan over kaarten, hun attributen, en listings (aanbod/verkoop). Het doel is om Cursor AI (met Claude 3.7 Sonnet) te instrueren waar gegevens te vinden zijn, hoe tabellen verbonden zijn, en hoe subcategorieën werken. Dit document is bedoeld als een duidelijke handleiding voor het opvragen en manipuleren van gegevens.

---

#### #### 2. De Drie Hoofdcategorieën

De database is verdeeld in drie schema's, elk met een eigen set tabellen:

1. **Sports**: Bevat gegevens over sportkaarten (bijv. voetbal, basketbal).
2. **TCG**: Bevat gegevens over trading card games (bijv. Pokémon, Magic: The Gathering).
3. **Entertainment**: Bevat gegevens over entertainmentkaarten (bijv. Star Wars, Disney).

Elk schema heeft een vergelijkbare structuur maar met specifieke tabellen en velden die passen bij de categorie. Subcategorieën spelen een centrale rol in het organiseren van kaarten binnen deze hoofdcategorieën (zie sectie 3).

---

### #### 3. Werking van Subcategorieën

Subcategorieën definiëren de specifieke typen kaarten binnen een hoofdcategorie. Ze fungeren als een basislaag waarop andere tabellen bouwen:

- **Sports**: Subcategorieën zijn sporttypes (bijv. "Soccer", "Basketball"), opgeslagen in `sports.sports\_types`.
- **TCG**: Subcategorieën zijn spellen (bijv. "Pokémon", "Magic: The Gathering"), opgeslagen in `tcg.games`.
- **Entertainment**: Subcategorieën zijn thema's (bijv. "Star Wars", "Walt Disney"), opgeslagen in `entertainment.subcategories`.

Deze subcategorieën hebben een `id`-veld dat via foreign keys wordt gebruikt in andere tabellen (bijv. `sports.leagues.sports\_type\_id`, `tcg.series.game\_id`, `entertainment.cards.subcategory\_id`) om kaarten en gerelateerde gegevens te koppelen aan hun specifieke context.

---

### #### 4. Tabelstructuur en Inhoud

#### ##### 4.1 Sports Schema

- **sports.sports\_types** (Subcategorieën)
  - **Structuur**: `id` SERIAL PRIMARY KEY, name VARCHAR, description VARCHAR
  - **Inhoud**: Sporttypes (bijv. "Soccer", "Basketball").
  - **Voorbeeld**: `(1, 'Soccer', 'The world's most popular sport.')
  - **Verbinding**: `id` wordt gebruikt in `sports.leagues.sports\_type\_id` en `sports.teams.sports\_type\_id`.
- **sports.leagues**
  - **Structuur**: `id` SERIAL PRIMARY KEY, sports\_type\_id INT REFERENCES sports.sports\_types(id), name VARCHAR, country VARCHAR
  - **Inhoud**: Competities (bijv. "Premier League", "NBA").
  - **Voorbeeld**: `(1, 1, 'Premier League', 'England')`
  - **Verbinding**: `sports\_type\_id` verwijst naar `sports\_types.id`; `id` wordt gebruikt in `sports.seasons.league\_id`.

- **sports.seasons**

- **Structuur**: `id` SERIAL PRIMARY KEY, league\_id INT REFERENCES sports.leagues(id), name VARCHAR, start\_date DATE, end\_date DATE`

- **Inhoud**: Seizoenen van competities (bijv. "2023-2024").

- **Voorbeeld**: `(1, 1, '2023-2024', '2023-08-01', '2024-05-31')`

- **Verbinding**: `league\_id` verwijst naar `sports.leagues.id`; `id` wordt gebruikt in `sports.teams.season\_id`.

- **sports.teams**

- **Structuur**: `id` SERIAL PRIMARY KEY, season\_id INT REFERENCES sports.seasons(id), name VARCHAR, city VARCHAR, sports\_type\_id INT REFERENCES sports.sports\_types(id)`

- **Inhoud**: Teams (bijv. "Manchester City", "Los Angeles Lakers").

- **Voorbeeld**: `(1, 1, 'Manchester City', 'Manchester', 1)`

- **Verbinding**: `season\_id` naar `sports.seasons.id`; `sports\_type\_id` naar `sports.sports\_types.id`; `id` gebruikt in `sports.players.team\_id`.

- **sports.players**

- **Structuur**: `id` SERIAL PRIMARY KEY, team\_id INT REFERENCES sports.teams(id), name VARCHAR, position VARCHAR, sports\_type\_id INT REFERENCES sports.sports\_types(id)`

- **Inhoud**: Spelers (bijv. "Erling Haaland", "LeBron James").

- **Voorbeeld**: `(1, 1, 'Erling Haaland', 'Striker', 1)`

- **Verbinding**: `team\_id` naar `sports.teams.id`; `sports\_type\_id` naar `sports.sports\_types.id`; `id` gebruikt in `sports.cards.player\_id`.

- **sports.cards**

- **Structuur**: `id` SERIAL PRIMARY KEY, player\_id INT REFERENCES sports.players(id), name VARCHAR, card\_number VARCHAR, rarity VARCHAR, image\_url VARCHAR, description VARCHAR, year INT, card\_type VARCHAR`

- **Inhoud**: Kaarten (bijv. "Erling Haaland Rookie").

- **Voorbeeld**: `(1, 1, 'Erling Haaland Rookie', '001/500', 'Rare', 'https://example.com/haaland.jpg', 'Rookie card of a top striker.', 2023, 'Rookie')`

- **Verbinding**: `player\_id` naar `sports.players.id`; `id` gebruikt in `sports.card\_attributes.card\_id` en `sports.listings.card\_id`.

- **sports.card\_attributes**

- **Structuur**: `id SERIAL PRIMARY KEY, card\_id INT REFERENCES sports.cards(id), attribute\_key VARCHAR, attribute\_value VARCHAR`
  - **Inhoud**: Attributen van kaarten (bijv. "numbered", "patch").
  - **Voorbeeld**: `(1, 1, 'numbered', '001/500'), (2, 1, 'patch', 'No')`
  - **Verbinding**: `card\_id` naar `sports.cards.id`.
- 
- **sports.listings**
    - **Structuur**: `id SERIAL PRIMARY KEY, user\_id UUID, card\_id INT REFERENCES sports.cards(id), type VARCHAR, price DECIMAL, condition VARCHAR, expires\_at TIMESTAMP`
    - **Inhoud**: Aanbod/verkoop (bijv. "trade", "sale").
    - **Voorbeeld**: `(1, '1fe125ce-6193-4323-99d0-5653415d32f6', 1, 'trade', NULL, 'Mint', NULL)`
    - **Verbinding**: `card\_id` naar `sports.cards.id`.

#### #### 4.2 TCG Schema

- **tcg.games** (Subcategorieën)
  - **Structuur**: `id SERIAL PRIMARY KEY, name VARCHAR, description VARCHAR`
  - **Inhoud**: Spellen (bijv. "Pokémon", "Magic: The Gathering").
  - **Voorbeeld**: `(1, 'Pokémon', 'A popular trading card game with pocket monsters.')
  - **Verbinding**: `id` gebruikt in `tcg.series.game\_id`.
  
- **tcg.series**
  - **Structuur**: `id SERIAL PRIMARY KEY, game\_id INT REFERENCES tcg.games(id), name VARCHAR, release\_date DATE`
  - **Inhoud**: Series (bijv. "Sword & Shield").
  - **Voorbeeld**: `(1, 1, 'Sword & Shield', '2020-02-07')`
  - **Verbinding**: `game\_id` naar `tcg.games.id`; `id` gebruikt in `tcg.sets.series\_id`.
  
- **tcg.sets**
  - **Structuur**: `id SERIAL PRIMARY KEY, series\_id INT REFERENCES tcg.series(id), name VARCHAR, release\_date DATE`
  - **Inhoud**: Sets (bijv. "Evolving Skies").
  - **Voorbeeld**: `(1, 1, 'Evolving Skies', '2021-08-27')`

- **Verbinding**: `series\_id` naar `tcg.series.id`; `id` gebruikt in `tcg.cards.set\_id`.

- **tcg.cards**

- **Structuur**: `id` SERIAL PRIMARY KEY, set\_id INT REFERENCES tcg.sets(id), name VARCHAR, card\_number VARCHAR, rarity VARCHAR, image\_url VARCHAR, description VARCHAR`

- **Inhoud**: Kaarten (bijv. "Charizard VMAX").

- **Voorbeeld**: `(1, 1, 'Charizard VMAX', '001/100', 'Ultra Rare', 'https://example.com/charizard.jpg', 'A fiery dragon Pokémon.')

- **Verbinding**: `set\_id` naar `tcg.sets.id`; `id` gebruikt in `tcg.pokemon\_attributes.card\_id`, `tcg.magic\_attributes.card\_id`, en `tcg.listings.card\_id`.

- **tcg.pokemon\_attributes**

- **Structuur**: `id` SERIAL PRIMARY KEY, card\_id INT REFERENCES tcg.cards(id), hp INT, type VARCHAR, stage VARCHAR, weakness VARCHAR`

- **Inhoud**: Pokémon-specifieke attributen.

- **Voorbeeld**: `(1, 1, 330, 'Fire', 'VMAX', 'Water')`

- **Verbinding**: `card\_id` naar `tcg.cards.id`.

- **tcg.magic\_attributes**

- **Structuur**: `id` SERIAL PRIMARY KEY, card\_id INT REFERENCES tcg.cards(id), mana\_cost VARCHAR, card\_type VARCHAR, power VARCHAR, toughness VARCHAR`

- **Inhoud**: Magic-specifieke attributen.

- **Voorbeeld**: `(1, 2, '0', 'Artifact', NULL, NULL)`

- **Verbinding**: `card\_id` naar `tcg.cards.id`.

- **tcg.listings**

- **Structuur**: `id` SERIAL PRIMARY KEY, user\_id UUID, card\_id INT REFERENCES tcg.cards(id), type VARCHAR, price DECIMAL, condition VARCHAR, expires\_at TIMESTAMP`

- **Inhoud**: Aanbod/verkoop.

- **Voorbeeld**: `(1, '1fe125ce-6193-4323-99d0-5653415d32f6', 1, 'sale', 150.00, 'Near Mint', '2025-04-01 12:00:00+00')`

- **Verbinding**: `card\_id` naar `tcg.cards.id`.

#### #### 4.3 Entertainment Schema

- **entertainment.subcategories** (Subcategorieën)

- **Structuur**: `id SERIAL PRIMARY KEY, name VARCHAR, description VARCHAR`

- **Inhoud**: Thema's (bijv. "Star Wars", "Walt Disney").

- **Voorbeeld**: `(1, 'Star Wars', 'Cards based on the Star Wars universe.')

- **Verbinding**: `id` gebruikt in `entertainment.cards.subcategory\_id`.

- **entertainment.cards**

- **Structuur**: `id SERIAL PRIMARY KEY, subcategory\_id INT REFERENCES entertainment.subcategories(id), name VARCHAR, card\_number VARCHAR, rarity VARCHAR, image\_url VARCHAR, description VARCHAR, character VARCHAR`

- **Inhoud**: Kaarten (bijv. "Darth Vader").

- **Voorbeeld**: `(1, 1, 'Darth Vader', '001/200', 'Rare', 'https://example.com/vader.jpg', 'The Sith Lord in action.', 'Darth Vader')`

- **Verbinding**: `subcategory\_id` naar `entertainment.subcategories.id`; `id` gebruikt in `entertainment.attributes.card\_id` en `entertainment.listings.card\_id`.

- **entertainment.attributes**

- **Structuur**: `id SERIAL PRIMARY KEY, card\_id INT REFERENCES entertainment.cards(id), attribute\_key VARCHAR, attribute\_value VARCHAR`

- **Inhoud**: Attributen (bijv. "movie", "edition").

- **Voorbeeld**: `(1, 1, 'movie', 'The Empire Strikes Back'), (2, 1, 'edition', 'Limited')`

- **Verbinding**: `card\_id` naar `entertainment.cards.id`.

- **entertainment.listings**

- **Structuur**: `id SERIAL PRIMARY KEY, user\_id UUID, card\_id INT REFERENCES entertainment.cards(id), type VARCHAR, price DECIMAL, condition VARCHAR, expires\_at TIMESTAMP`

- **Inhoud**: Aanbod/verkoop.

- **Voorbeeld**: `(1, '1fe125ce-6193-4323-99d0-5653415d32f6', 1, 'auction', 50.00, 'Mint', '2025-03-25 10:00:00+00')`

- **Verbinding**: `card\_id` naar `entertainment.cards.id`.

---

#### #### 5. Onderlinge Verbindingen

##### - **Sports**:

- `sports\_types.id` → `leagues.sports\_type\_id`, `teams.sports\_type\_id`, `players.sports\_type\_id`
- `leagues.id` → `seasons.league\_id`
- `seasons.id` → `teams.season\_id`
- `teams.id` → `players.team\_id`
- `players.id` → `cards.player\_id`
- `cards.id` → `card\_attributes.card\_id`, `listings.card\_id`

##### - **TCG**:

- `games.id` → `series.game\_id`
- `series.id` → `sets.series\_id`
- `sets.id` → `cards.set\_id`
- `cards.id` → `pokemon\_attributes.card\_id`, `magic\_attributes.card\_id`, `listings.card\_id`

##### - **Entertainment**:

- `subcategories.id` → `cards.subcategory\_id`
- `cards.id` → `attributes.card\_id`, `listings.card\_id`

---

#### #### 6. Instructies voor Cursor AI (Claude 3.7 Sonnet)

**Beste Claude 3.7 Sonnet,**

Dit document beschrijft een database met drie schema's: `sports`, `tcg`, en `entertainment`. Volg deze instructies om gegevens op te vragen of te manipuleren:

##### 1. **Begrijp de Hoofdcategorieën**:

- `sports`: Sportkaarten (subcategorieën in `sports.sports\_types`).
- `tcg`: Trading card games (subcategorieën in `tcg.games`).
- `entertainment`: Entertainmentkaarten (subcategorieën in `entertainment.subcategories`).

## 2. **\*\*Zoek Subcategorieën\*\***:

- Voor sporttypes: ``SELECT * FROM sports.sports_types;``
- Voor spellen: ``SELECT * FROM tcg.games;``
- Voor thema's: ``SELECT * FROM entertainment.subcategories;``

## 3. **\*\*Volg de Verbindingen\*\***:

- Gebruik foreign keys om gerelateerde data te vinden (zie sectie 5).
- Bijvoorbeeld: om kaarten van een speler te vinden in ``sports``:

```
```sql
SELECT c.*
FROM sports.cards c
JOIN sports.players p ON c.player_id = p.id
WHERE p.name = 'Erling Haaland';
```
```

## 4. **\*\*Attributen Ophalen\*\***:

- Attributen zijn key-value-paren in ``sports.card_attributes``, ``tcg.pokemon_attributes`` / ``tcg.magic_attributes``, en ``entertainment.attributes``.
- Voorbeeld: attributen van een kaart in ``entertainment``:

```
```sql
SELECT a.attribute_key, a.attribute_value
FROM entertainment.attributes a
WHERE a.card_id = 1;
```
```

## 5. **\*\*Listings Opvragen\*\***:

- Gebruik ``listings``-tabellen om aanbod te vinden (bijv. ``sports.listings``, ``tcg.listings``).
- Voorbeeld: verkoopitems in ``tcg``:

```
```sql
SELECT l.*, c.name
FROM tcg.listings l
```



```
JOIN tcg.cards c ON l.card_id = c.id  
WHERE l.type = 'sale';  
` ` `
```

6. **\*\*Wees Specifiek\*\***:

- Geef altijd het schema aan (bijv. `sports.cards` in plaats van alleen `cards`) om verwarring te vermijden.
- Gebruik `JOIN`-queries om gegevens over tabellen heen te combineren.

**\*\*Voorbeeldquery\*\***: Vind alle kaarten van "Pokémon" in `tcg` met hun sets:

```
` ` `sql  
  
SELECT c.name, s.name AS set_name  
FROM tcg.cards c  
JOIN tcg.sets s ON c.set_id = s.id  
JOIN tcg.series sr ON s.series_id = sr.id  
JOIN tcg.games g ON sr.game_id = g.id  
WHERE g.name = 'Pokémon';  
` ` `
```

Met deze structuur en instructies kun je de database effectief gebruiken. Als je vragen hebt, analyseer dan de tabelstructuren in sectie 4 en de verbindingen in sectie 5.