Prompt Engineering 2

0, Introductie

source

We hebben in prompt engineering 1 geleerd waaraan een goede prompt moest voldoen.

Dit zijn de basis kenmerken van een goede prompt. De eerste drie kenmerken moet je prompt **altijd** aan voldoen!

1. **Context** - een goede prompt heeft voldoende context.
2. **Details/Specifiek** - een goed prompt heeft voldoende details en is zo specifiek mogelijk.
3. **Duidelijkheid** - een goede prompt is duidelijk.
4. **Doelgericht** - een goede prompt is doelgericht.
5. **Vorm** - in een goede prompt kan je de output in een bepalade vorm vragen.
6. **Toon** -door in de prompt de toon op te nemen, bepaal je de vorm van het antwoord.

In deze module gaan we 6 **advanced prompt-technieken** leren. Deze technieken heb je niet altijd nodig maar het is handig om deze technieken te kennen.

Bovendien zijn de meeste technieken ook toepasbaar in als '**problem solving**' technieken.

1. **Isolate the problem**
   Focus alleen op het onderdeel dat opgelost moet worden.
   Laat overbodige context of code weg, zodat de AI zich op het juiste richt.
2. **Provide lists in bullet points**
   Structuur helpt de AI om overzichtelijke en duidelijke antwoorden te geven.
3. **Provide the order if you ask for multiple tasks**
   Geef een logische volgorde bij samengestelde opdrachten.
4. **Geef voorbeelden (few-shot prompting)**
   Laat zien wat je bedoelt met een input/output-voorbeeld.

5. **Stel voorwaarden of beperkingen**
   Geef grenzen aan zoals "gebruik max. 100 woorden" of "geen disclaimers".
6. **Werk in stappen (chain-of-thought prompting)**
   Vraag de AI om stap voor stap te redeneren of eerst een plan te maken.

??Opdracht

Situatie

Een student wil een AI laten helpen bij het schrijven van een korte uitleg over **variabelen in Python** voor een klas van 14-jarigen. Hieronder staan drie prompts.

Lees de drie prompts en bepaal bij **welke prompt** de **context**, de **details**, of de **duidelijkheid** het sterkst aanwezig is.

Opgave 1

> Leg in eenvoudige taal uit wat een variabele is in Python. Gebruik korte zinnen en vermijd moeilijke woorden.

□□ **Vraag:** Waar blinkt deze prompt vooral in uit?

- Context
- Details
- Duidelijkheid

Opgave 2

> Ik geef les aan 14-jarige leerlingen die net beginnen met programmeren in Python. Ze begrijpen wat print() doet, maar hebben nog nooit met variabelen gewerkt. Leg uit wat een variabele is, geef een eenvoudig voorbeeld en gebruik taal die past bij hun leeftijd.

□□ **Vraag:** Waar blinkt deze prompt vooral in uit?

- Context
- Details

- Duidelijkheid

Opgave 3

> Schrijf een uitleg van 100 woorden over variabelen in Python. Gebruik de termen “waarde”, “naam”, en “toewijzing”. Voeg twee codevoorbeelden toe: één met een getal en één met tekst. Sluit af met een korte quizvraag.

Vraag: Waar blinkt deze prompt vooral in uit?

- Context
- Details
- Duidelijkheid

? Inleveren

Geef in een txt bestand (of in het tkstveld) weer van elke van de **drie opgaven** welk onderdeel, **context**, **details**, of **duidelijkheid** het sterkst aanwezig is.

1, Isolate the problem

? Uitleg

Richt je prompt op het exacte probleem. In plaats van een hele codepagina te geven, geef alleen het stuk code of de situatie waar het om draait. Hoe minder ruis, hoe beter de AI kan helpen.

> Je ziet dus dat je voor een goede prompt de code goed moet kunnen lezen.

??Opdracht

Je hebt een PHP-pagina met een formulier dat soms geen gegevens doorstuurt.

Maak een prompt voor ChatGPT waarin je **alleen het relevante deel van de code opneemt** en waarin je de AI vraagt om te helpen bij het vinden van de fout.

Houd nog steeds rekening met de **context, details en duidelijkheid**.

In dit voorbeeld zou je makkelijk de hele code kunnen verturen, maar in de echte wereld heb je veel meer code en daarom is dit een goede oefening in het isoleren van code die relevant is.

> Tip: je kan ook andersom redeneren en alle code waarvan je zeker weet die goed is of niets met het formulier te maken heeft weghaald.

```
<?php
$bericht = "";
$success = false;

if ($_SERVER["REQUEST_METHOD"] === "POST") {
    if (!empty($_POST["naam"])) {
        $naam = $_POST["naam"];
        $bericht = "Hallo, " . htmlspecialchars($naam) . "!";
        $success = true;
    } else {
        $bericht = "Je moet je naam invullen.";
    }
}
?>

<!DOCTYPE html>
<html lang="nl">
<head>
    <meta charset="UTF-8">
    <title>Contactpagina</title>
    <style>
        body { font-family: Arial; background-color: #f0f0f0; margin: 2rem; }
        .container { background: white; padding: 2rem; border-radius: 8px; max-width: 600px;
margin: auto; }
        .error { color: red; }
        .success { color: green; }
        footer { margin-top: 4rem; font-size: 0.8em; text-align: center; color: #666; }
    </style>
    <script>
        function checkForm() {
            const naam = document.getElementById("naam").value;
            if (naam.trim() === "") {
                alert("Naam is verplicht!");
```

```
                return false;
            }
            return true;
        }
    </script>
</head>
<body>

<div class="container">
    <h1>Neem contact met ons op</h1>

    <form method="get" action="" onsubmit="return checkForm();">
        <label for="naam">Naam:</label><br>
        <input type="text" id="naam" name="naam" placeholder="Vul je naam in"><br><br>
        <label for="email">E-mail:</label><br>
        <input type="email" id="email" name="email" placeholder="voorbeeld@domein.nl"><br><br>
        <label for="vraag">Je vraag:</label><br>
        <textarea id="vraag" name="vraag"></textarea><br><br>
        <button type="submit">Verzenden</button>
    </form>

    <?php if ($bericht): ?>
        <p class="<?= $success ? 'success' : 'error' ?>"><?= $bericht ?></p>
    <?php endif; ?>
</div>

<footer>
    &copy; 2025 Webdev Company | <a href="#">Privacybeleid</a>
</footer>

</body>
</html>
```

? Inleveren

De prompt die het probleem uiteindelijk vond en waarin ***alleen*** de relevante code staat.

De prompt voldoet tevens minimaal aan de basis kenmerken van een prompt: **context, details en duidelijkheid**.

2, Provide lists in bullet points

? Uitleg

Als je meerdere dingen van een AI vraagt, is het belangrijk om structuur aan te brengen in je prompt. Door **bullet points of genummerde lijsten** te gebruiken:

- Maak je je prompt overzichtelijk.
- Zorg je ervoor dat de AI geen onderdelen vergeet.
- Help je jezelf om duidelijk te verwoorden wat je wilt.

Bullet points zijn dus niet alleen netjes — ze zijn slim.

??Opdracht

Hieronder zie je een afbeelding van een eenvoudige webpagina met meerdere onderdelen.

Het plaatje kan je hier downloaden: Berglandschap.png.

image.png

1. **Bestudeer de web pagina goed.** Wat zie je allemaal op de pagina?
2. **Bedenk wat je aan de AI zou vragen om precies deze pagina te laten maken.**
3. **Noteer minstens 4 onderdelen van de pagina als bullet points** in je prompt.

? Inleveren

1. Prompt (txt)
2. Resultaat in code (html)
3. Schermafdruk van resultaat (jpg of png)

3, Provide the order if you ask for multiple tasks

? Uitleg

Als je aan de AI vraagt om **meerdere dingen tegelijk te doen**, dan is het belangrijk dat je duidelijk maakt in **welke volgorde** dat moet gebeuren.

AI volgt jouw instructies letterlijk — dus als de volgorde onduidelijk is, krijg je soms een rommelig of onvolledig resultaat.

Door een logische **nummering** of expliciete volgorde te geven, help je de AI om stapsgewijs te werken.

Voorbeeld (vaag):
Maak een invoerpagina met HTML en verwerk de gegevens met PHP.

Voorbeeld (duidelijker):

1. Maak eerst een HTML-pagina met een formulier waarin wordt gevraagd naar de tijd en de klantnaam
2. Voeg daarna de PHP-code toe die de ingevulde gegevens toont.

??Opdracht

Je wilt de AI vragen om je te helpen met het maken van een eenvoudige contactpagina.

Die moet bestaan uit:

- Een HTML-formulier waarin je je naam en e-mailadres kunt invullen
- Een PHP-script dat de gegevens verwerkt en netjes op het scherm toont

1. Schrijf een prompt waarin je deze twee onderdelen vraagt in de juiste **volgorde**.
2. Gebruik een **genummerde lijst** of maak duidelijk met woorden wat “eerst” en “daarna” moet gebeuren.
3. Houd rekening met de **context, details en duidelijkheid**.

? Inleveren

1. Je volledige prompt (txt)
2. De gegenereerde HTML (.html)

3. De gegenereerde PHP (.php)
4. Een screenshot van het resultaat in de browser

4, Geef voorbeelden (few-shot prompting)

? Uitleg

Als je wilt dat de AI **op een specifieke manier code genereert of uitlegt**, dan helpt het enorm als je eerst **een paar voorbeelden** geeft. Dit heet **few-shot prompting**.

Je laat zien wat jij bedoelt — de AI herkent het patroon en volgt het. Dit is heel handig bij het ontwerpen van knoppen, formulieren of herhalende structuren.

Voorbeeld prompt

```
<button style="background-color: red;">Verwijderen</button><button
    style="
        background-color: #dc3545;
        color: white;
        border: none;
        padding: 10px 16px;
        font-size: 16px;
        border-radius: 4px;
        cursor: pointer;
        transition: background-color 0.3s ease;
    "
    onmouseover="this.style.backgroundColor='#c82333'"
    onmouseout="this.style.backgroundColor='#dc3545'"
    onclick="handleDelete()"
>
    Verwijderen
</button>

Deze knop ziet er professioneel uit: rood met witte tekst, afgeronde hoeken en een vloeiend
hover-effect dat de kleur donkerder maakt wanneer je erover beweegt.
Klikt de gebruiker op de knop, dan wordt de JS code handleDelete() aangeroepen.
```

```
Maak op dezelfde manier een blauwe knop 'toevoegen'.
```

??Opdracht

Je wilt een aantal drop down menu's maken. Hier is een voorbeeld.

```
<style>
  .form-control {
    width: 250px;
    padding: 8px 12px;
    font-size: 16px;
    border: 1px solid #ccc;
    border-radius: 6px;
    background-color: #f9f9f9;
    appearance: none; /* verwijder standaardstijl in sommige browsers */
  }

  .form-control:focus {
    border-color: #007BFF;
    outline: none;
    background-color: #fff;
    box-shadow: 0 0 5px rgba(0, 123, 255, 0.3);
  }
</style>

<select id="taalSelect" name="taal" class="form-control">
  <option value="en">Engels</option>
  <option value="nl" selected>Nederlands</option>
  <option value="fr">Frans</option>
</select>
```

Maak een multi-shot prompt waarbij je het menu uit de code hierboven als voorbeeld gebruikt en waarbij je het volgende menu maakt:

image.png

> Tip: denk goed na over wat je de AI precies als voorbeeld geeft, ***alleen*** de HTML of HTML **met** CSS? Denk hierbij ook aan *Isolate the problem*.

? Inleveren

1. Je volledige prompt inclusief voorbeeld(en) en de responce van AI (.txt)
2. De gegenereerde knop (.html)
3. Een screenshot van de knop in de browser (.png, .jpg)

5, Stel voorwaarden of beperkingen

? Uitleg

Soms wil je meer controle over **hoe** de AI iets oplevert. Je kunt dan in je prompt **voorwaarden of beperkingen** opnemen.

Bijvoorbeeld:

- Beperk het aantal woorden of regels code
- Vermijd bepaalde technieken (zoals frameworks)
- Vraag om iets op een specifieke manier te doen (bijv. alleen in-line CSS)

Door deze randvoorwaarden op te nemen, stuur je de AI veel gerichter aan.

??Opdracht

Je wilt een eenvoudige HTML-formulierpagina laten genereren, maar je stelt een aantal duidelijke voorwaarden.

1. **Bedenk minstens 3 voorwaarden** waaraan de code moet voldoen. Bijvoorbeeld:
   - Geen externe CSS-bestanden of frameworks zoals Bootstrap
   - Maximaal 30 regels HTML
   - Formulier moet naam en e-mailadres bevatten, met eenvoudige styling
2. **Schrijf een prompt** waarin je deze voorwaarden duidelijk formuleert.
   - Leg de context uit: *“Ik wil een eenvoudig formulier...”*
   - Som je voorwaarden op in een bulletlijst of genummerde lijst
   - Sluit duidelijk af met je verzoek: *“Geef alleen de HTML-code”*

3. Zorg dat je prompt voldoet aan de **context, details en duidelijkheid**.

? Inleveren

1. Je prompt (.txt)
2. De gegenereerde HTML-code (.html)
3. Een screenshot van het formulier (.jpg, .png)

6, Chain-of-thought prompting 1

? Uitleg

Bij opdrachten met meerdere onderdelen is het slim om de AI **eerst te laten nadenken en plannen**. Dat noem je *chain-of-thought prompting*.

Je vraagt de AI dus om niet meteen te bouwen, maar eerst te analyseren:

- Wat moet er gebeuren, welke stappen moet ik volgen?
- In welke volgorde?
- Wat zijn de componenten en technieken die ik moet gebruiken?

Daarna kan de AI **gestructureerd** de code aanleveren. Zo voorkom je onvolledige of chaotische antwoorden.

?Voorbeeld

Prompt

> *Ik wil een formulier dat een naam opslaat.*
> *Leg eerst uit welke stappen ik daarvoor nodig heb.*
> *Geef dan de HTML-code.*
> *Geef daarna de PHP-code.*
> *Leg elke stap kort uit.*

Verwachte AI-reactie

1. Je hebt een HTML-formulier nodig om de naam in te voeren.

2. Je hebt een PHP-script nodig dat de naam opvangt en verwerkt.
3. Je moet het formulier met method="post" versturen.
4. Daarna geef ik de HTML en PHP code apart, met uitleg.

Als je een stap niet snapt, of het niet eens bent, vraag dan verheldering. Vraag uitleg, bijvoorbeeld:

Prompt

> *Wat gebeuert er in stap 3 en waarom moet ik een 'method=post' gebruiken, zijn er alternatieven?*
>
> Op deze manier leer je een probleem in stappen opdelen en daarna de stappen één voor één op te lossen. Nu lijkt dat wellicht overbodig, maar bij grotere projecten is deze methode noodzakelijk.

??Opdracht

Je wil een AI vragen om een simpele “Contact opnemen”-pagina te maken met de volgende onderdelen:

- Een **HTML-formulier** met velden voor naam, e-mail en bericht
- Een **mooie layout met CSS** (geen framework)
- Een **PHP-bestand** dat het formulier verwerkt en de data netjes toont
- **De gegevens mogen niet verstuurd worden zonder geldige invoer**

Maar: je wil niet meteen de code, je wil dat de AI eerst in stappen uitlegt *hoe je dit moet aanpakken*.

1. **Schrijf een prompt** waarin je aan de AI vraagt om:
   - Eerst te analyseren wat deze opdracht inhoudt
   - Daarna een overzicht in stappen te geven van wat er allemaal moet gebeuren (HTML-structuur, input validatie, PHP-verwerking, opmaak)
   - Pas daarna per stap de juiste code te geven, met uitleg
2. Gebruik termen als:
   - “Geef eerst een analyse van de opdracht”
   - “Schrijf daarna de stappen uit in logische volgorde”
   - “Geef dan pas de bijbehorende code, met uitleg per stap”

3. Voeg in je prompt beperkingen of voorkeuren toe, zoals:
    - “Gebruik geen externe CSS-frameworks”
    - “Gebruik eenvoudige PHP, geen database”
    - “Geef geen code voordat alle stappen duidelijk zijn”
4. Zorg dat je prompt voldoet aan de kenmerken: **context, details, duidelijkheid, volgorde**

? Inleveren

1. Je volledige prompt en antwoord (.txt)
2. De uiteindelijke werkende HTML (.html)
3. De PHP (.php)
4. Screenshot van de werkende pagina (.jpg, .png)

7, Combineer technieken (integratieopdracht)

? Uitleg

Je kent nu zes *advanced prompt-technieken*. In de praktijk gebruik je ze vaak **samen**: je begint met een duidelijke context, kiest de juiste details, beperkt de output, vraagt een logische volgorde en laat de AI stap voor stap redeneren. In deze opdracht ga je laten zien dat je bewust kunt kiezen welke technieken je inzet — en waarom.

?? Opdracht

Je wil een AI vragen om een **kleine mini-webapplicatie** te maken waarin een gebruiker zijn **favoriete programmeertaal** kan kiezen en een korte uitleg krijgt over die taal.

Maak een prompt waarin je minstens **vier** van de volgende technieken toepast:

1. **Isolate the problem** – focus op de essentie van de taak.
2. **Provide lists in bullet points** – geef structuur aan je vraag.
3. **Provide the order** – geef aan in welke volgorde de AI moet werken.

4. **Few-shot prompting** – geef één of meer voorbeelden van input en gewenste output.
5. **Stel voorwaarden of beperkingen** – leg grenzen vast (zoals aantal regels, geen frameworks, enz.).
6. **Chain-of-thought prompting** – laat de AI eerst een plan maken en daarna pas uitvoeren.

Je prompt moet daarnaast voldoen aan de drie **basiskenmerken** van een goede prompt: **context**, **details** en **duidelijkheid**.

? Inleveren

1. Je volledige prompt met AI antwoord (.txt)
2. Een korte uitleg (minimaal 100 woorden) waarin je beantwoordt:
   - Aan welke **vier technieken** jouw prompt voldoet (.txt)
   - **Hoe** die technieken terugkomen in je prompt.
   - **Waarom** je juist deze technieken hebt gekozen.
   - Aan welke **basiskenmerken** (context, details, duidelijkheid) jouw prompt voldoet, en hoe dat in je tekst zichtbaar is.

8, Afsluitende Reflectie

? Uitleg

Je hebt nu geleerd hoe je met verschillende technieken betere prompts kunt schrijven en hoe je de AI kunt sturen met **context**, **details**, **duidelijkheid** en **doelgerichtheid**. In deze laatste opdracht denk je terug op wat je hebt geleerd — *niet over de inhoud van de code*, maar over je eigen denkproces.

?? Opdracht

Beantwoord de onderstaande vragen in je eigen woorden. Gebruik **geen AI-hulp** voor het schrijven van de tekst. De opdracht is bedoeld om te laten zien wat jij zélf hebt geleerd en ervaren.

> **Belangrijk:** De tekst wordt beoordeeld op echtheid en reflectievermogen. AI-gegenereerde teksten klinken vaak te algemeen of te perfect. Gebruik daarom concrete voorbeelden uit jouw eigen opdrachten en benoem specifieke situaties of keuzes.

Vragen

1. Welke **twee prompt-technieken** vond jij het meest nuttig, en waarom?
2. Beschrijf een moment waarop je eerst een slechte prompt maakte, maar daarna verbeterde. Wat heb je precies aangepast?
3. Welke **basiskenmerken** (context, details, duidelijkheid, doelgerichtheid, vorm, toon) vind jij het lastigst toe te passen? Leg uit waarom.
4. Hoe heeft het leren van deze technieken jouw manier van denken of werken met AI veranderd?
5. Wat zou je in een volgende opdracht anders aanpakken als je opnieuw met een AI werkt?

? Inleveren

1. Beantwoord de vragen één voor één en nummer de antwoorden. Lever PDF in.
2. De tekst bevat minimaal **200 woorden** en maximaal **400 woorden**.

> **Tip:** Vermijd vage zinnen als “Ik heb veel geleerd over prompts.” Schrijf concreet, bijvoorbeeld: “Bij opdracht 4 gebruikte ik eerst te veel code in mijn prompt. Toen ik het beperkte tot alleen het formulier, werkte het beter.”

Revision #26
Created 2025-05-13 17:32:12 UTC by Max
Updated 2026-07-10 08:35:14 UTC by Max