

PRD GUIDE

Sådan skriver du et godt Product Requirements Document
til vibe coding med AI

Anne Lindberg / Syndicate

TIPS OG TRICKS INDEN DU STARTER:

1. Skriv til en chatbot (Claude, ChatGPT, Copilot osv) at du skal skrive en PRD til din løsning/feature, og bed den stille dig opklarende spørgsmål til I kommer frem til en PRD sammen.
2. Hvis outputtet ikke er godt, ret i din PRD og start forfra. Dette kan i nogen tilfælde være nemmere end at forsøge at prompte sig til et andet output

Del 1: PRD-skabelon

Brug denne skabelon, når du starter et nyt vibe coding-projekt. Udfyld alle obligatoriske felter for det bedste resultat. Sektionerne er ordnet i den rækkefølge, som AI-værktøjerne (Lovable, Cursor, Claude Code) bearbejder dem bedst.

1 Header og metadata

Obligatorisk

Projektets navn, versionsnummer, forfatter og dato. Holder dokumentet forankret i tid — særligt vigtigt, når du sender det til et AI-værktøj. Giv det også et klart produktnavn, som AI'en kan bruge i UI og kode.

Projektnavn	[Navn på produktet eller featuren]
Version	1.0
Forfatter	[Dit navn / virksomhed]
Kunde	[Kundens navn]
Dato	[DD.MM.AAAA]

2 Baggrund og problemstilling

Obligatorisk

2-4 sætninger: hvem er brugerne, hvilken smerte har de i dag, og hvorfor er det vigtigt at løse. Dette er 'hvorfor bygger vi det her' -sektionen. Lovable og Claude bruger begge dette som kontekst, når de træffer design- og UX-beslutninger.

[Beskriv den nuværende situation og det problem, brugerne oplever. Forklar, hvad der mangler i dag, og hvad det koster ikke at have det. En god tommelfingerregel: skriv det som en brief til en ny udvikler.]

3 Tech stack

Obligatorisk

Frontend, backend, database, auth — en linje til hvert lag. Fortæller AI'en, hvilke værktøjer og konventioner der skal bruges. Uden denne får du generiske valg, der kan kollidere med dit setup.

Frontend	[f.eks. Lovable / Next.js / React + Tailwind]
Database	[f.eks. Supabase / PostgreSQL / Firebase]
Auth	[f.eks. Supabase Auth / Clerk / NextAuth]
Hosting	[f.eks. Vercel / Netlify]

4 Branding og designtokens

Obligatorisk

Eksakte hex-værdier for primære/accent-farver, fontvalg og et kort tone-ord. Vibe coding-værktøjer defaultter til generisk lilla/blå/hvid — vil du have et specifikt udtryk, SKAL du angive det her.

Primaer baggrund	#[hex]
Primaer tekst	#[hex]
Accent	#[hex] ([farvenavn])
Sekundar accent	#[hex] ([farvenavn])
Headline-font	[font og vægt, f.eks. Inter Black]
Body-font	[font og vægt, f.eks. Inter Regular]
Tone	[f.eks. Minimal, professionel, legende]

5 Auth og rettigheder

Obligatorisk

Hvem kan gøre hvad — og hvad sker der, hvis de ikke er logget ind. Definer som minimum: sign up, login, logout og den vigtigste permission-split. Dette styr er direkte input til Supabase Row Level Security og komponent-guards.

Eksempelvis:

- Sign up og login med [metode, f.eks. email + password] via [auth-udbyder]
- Log ud
- Kun brugere som er logget ind kan [skriv de handlinger, der kræver login]

6 Databaseskema

Obligatorisk

Hver tabel: kolonnenavne, typer og formål. Inkluder PKs, FKs og eventuelle unique constraints. Seed-data hvis nødvendigt. Dette er den enkeltsektion, der bedst forebygger hallucinerede skemaer og kæde migrationer.

Tabel: [tabelnavn]

Kolonne	Type	Beskrivelse
id	uuid (PK)	Auto-genereret primærnøgle
[felt]	text / integer / boolean	[Hvad gemmer det?]
created_at	timestamp	Auto-genereret
user_id	uuid (FK -> auth.users)	Hvem ejer raden

Inkluder UNIQUE constraints, f.eks. (request_id, user_id) hvis en bruger kun må gøre noget en gang. Tilføj seed-data under tabeldefinitionen, hvis boardet skal være forudfyldt ved demo.

7 Funktioner og flows

Obligatorisk

Grupper efter skærm eller brugerhandling. Hvert punkt beskriver en enkelt adfærd: hvad brugeren gør, hvad appen gør, hvad udfaldet er. Vær eksplicit om edge cases som 'allerede stemt' eller tomme tilstande.

[Skærmens navn]

- Vis [hvad der vises] som [korttype] med: [felt 1], [felt 2], [felt 3]
- Sorteret efter [kriterie] som standard

[Brugerhandling]

- [Hvad sker der når brugeren klikker/indsender/vælger]
- [Hvad opdateres i databasen]
- [Hvad ser brugeren efterfølgende]

8 Uden for scope

Valgfri

Selv en kort liste over hvad der IKKE bygges i denne version er guld værd. Det sætter forventninger hurtigt og forebygger scope creep, når du arbejder med kunder eller giver stafetten videre.

- [Funktion der udskydes til v2]
- [Integration der ikke er inkluderet]
- [Brugertype der ikke håndteres endnu]

9 Vibe coding-prompt

Valgfri

Den sammenfattede prompt, du indsætter i Lovable, Claude eller Cursor. Den destillerer alt ovenstående til en sammenhængende instruktion. Valgfri i selve PRD'en — men altid nødvendig i praksis. At skrive den som del af PRD'en tvinger dig til at kontrollere, om der er huller.

Del 2: Eksempel-PRD

Herunder er den komplette PRD fra webinar-demoen. Den kan bruges som reference og inspiration, når du udfylder skabelonen ovenfor.

Feature Request Tracker

Kunde	Flowboard
Version	1.0
Forfatter	Anne Lindberg / LindbergAI

Background

Flowboard is a Danish SaaS product team building project and workflow software. Their customers submit feature requests via Slack, email and support tickets, but there is no structure — the product team does not know what is actually most requested. Flowboard wants a portal where their customers can submit and vote on feature requests, so the product team can prioritize based on real demand.

Tech Stack

Frontend	Lovable (React/Tailwind)
Database	Supabase (PostgreSQL)
Auth	Supabase Auth (email/password)

Branding

Element	Vaerdi
Primary background	#000000 (black)
Primary text	#FFFFFF (white)
Accent	#D1FFD6 (mint green)
Secondary accent	#FFC2BF (pink)
Headline font	Inter Black or similar bold sans-serif
Body font	Inter Regular
Tone	Clean, minimal, modern

Auth

- Sign up and login with email + password via Supabase Auth

- Log out
- Only logged-in users can vote and create requests
- Non-logged-in users can view the board but not interact

Database

Table: feature_requests

Column	Type	Description
id	uuid (PK)	Auto-generated
title	text	Title of the request
description	text	Short description
tag	text	One of: Workflows, Integrations, Reporting, UI/UX, Mobile
vote_count	integer	Total number of votes
created_at	timestamp	Auto-generated
created_by	uuid (FK -> auth.users)	Who created it

Table: votes

Column	Type	Description
id	uuid (PK)	Auto-generated
request_id	uuid (FK -> feature_requests)	Which request
user_id	uuid (FK -> auth.users)	Who voted
created_at	timestamp	Auto-generated

Unique constraint on (request_id, user_id) — one vote per user per request.

Seed Data

Automatically insert the following rows if feature_requests is empty on first load:

Title	Description	Tag	Votes
Recurring tasks	We need tasks that automatically repeat on a schedule	Workflows	41
Slack integration	Send task updates and mentions directly to Slack channels	Integrations	36

Time tracking per task	Log hours directly on tasks so we can report on actual vs. estimated time	Reporting	29
Bulk edit tasks	Select multiple tasks and change assignee, due date or status in one action	UI/UX	24
Google Calendar sync	Two-way sync between Flowboard deadlines and Google Calendar	Integrations	21
Mobile app for iOS	A native iOS app so we can check and update tasks on the go	Mobile	17

Features

Feature Request Board

- Show all requests as cards with: title, description, tag, vote count and upvote button
- Sorted by votes by default

Voting

- One-click upvote updates `vote_count` and inserts a row in votes
- Upvote button shows as active if the current user has already voted

Filter & Sort

- Filter by tag: Workflows, Integrations, Reporting, UI/UX, Mobile
- Sort by: Most Voted / Newest

Add Request

- Form with title, description and tag selector
- Added directly to the board on submit

Lovable Prompt

Paste the following prompt into Lovable to scaffold the full app:

```
Build a Feature Request Tracker app for Flowboard – a Danish SaaS company building project and workflow software. Their customers submit feature requests through Slack and email with no structure. This portal lets customers submit and vote on features so the product team can prioritize based on real demand.
Design: dark background (#000000), white text, mint green (#D1FFD6) as accent color, pink (#FFC2BF) as secondary accent. Clean, minimal, professional – like a real product tool. Use Inter or a similar modern sans-serif font. Show 'Flowboard' as the product name in the header.
```

Auth: Add email/password sign up and login using Supabase Auth. Only logged-in users can vote or submit requests. Non-logged-in users can view the board but not interact.

Database: Create two tables in Supabase:

1. feature_requests: id (uuid PK), title (text), description (text), tag (text), vote_count (integer), created_at (timestamp), created_by (uuid FK to auth.users)

2. votes: id (uuid PK), request_id (uuid FK), user_id (uuid FK), created_at (timestamp) – with a unique constraint on (request_id, user_id) so each user can only vote once per request.

Seed data: On first load, if the feature_requests table is empty, automatically insert these 6 rows:

- 'Recurring tasks' / Workflows / 41 votes
- 'Slack integration' / Integrations / 36 votes
- 'Time tracking per task' / Reporting / 29 votes
- 'Bulk edit tasks' / UI/UX / 24 votes
- 'Google Calendar sync' / Integrations / 21 votes
- 'Mobile app for iOS' / Mobile / 17 votes

Features: board showing all requests as cards sorted by votes; one-click upvoting; filter by tag; sort by Most Voted / Newest; form to submit new requests.

Make it look and feel like a real product. No placeholder UI.