8 & 9 OCTOBER 2026

brightonSEO. United Kingdom

The Art Of Vibe Coding: Shipping Fast Without Breaking Anything



Ahmed Bhula

REM Digital Ltd.

@AhmedBhula

VIBE CODING?

You describe what you want in plain English.
AI writes the code. You review, refine, repeat.

"Fully giving in to the vibes"

TWO FLAVOURS OF VIBE CODING

ALL-IN-ONE PLATFORMS

bolt.new

Lovable

Replit

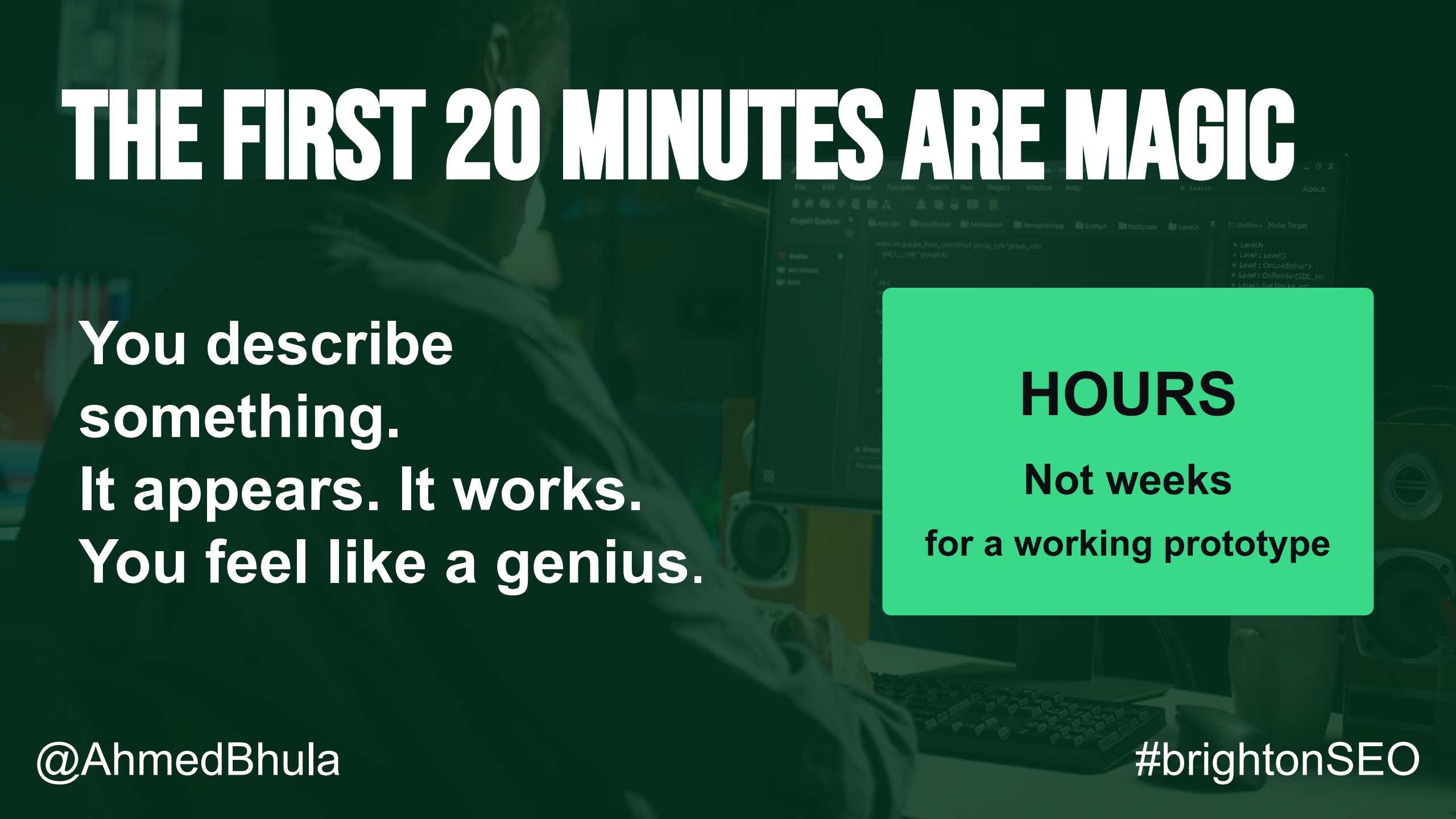
AI ASSISTANTS

Claude

GitHub Copilot



THE FIRST 20 MINUTES ARE MAGIC



You describe
something.
It appears. It works.
You feel like a genius.

HOURS

Not weeks

for a working prototype

@AhmedBhula

#brightonSEO

THEN YOU CHANGE ONE THING.

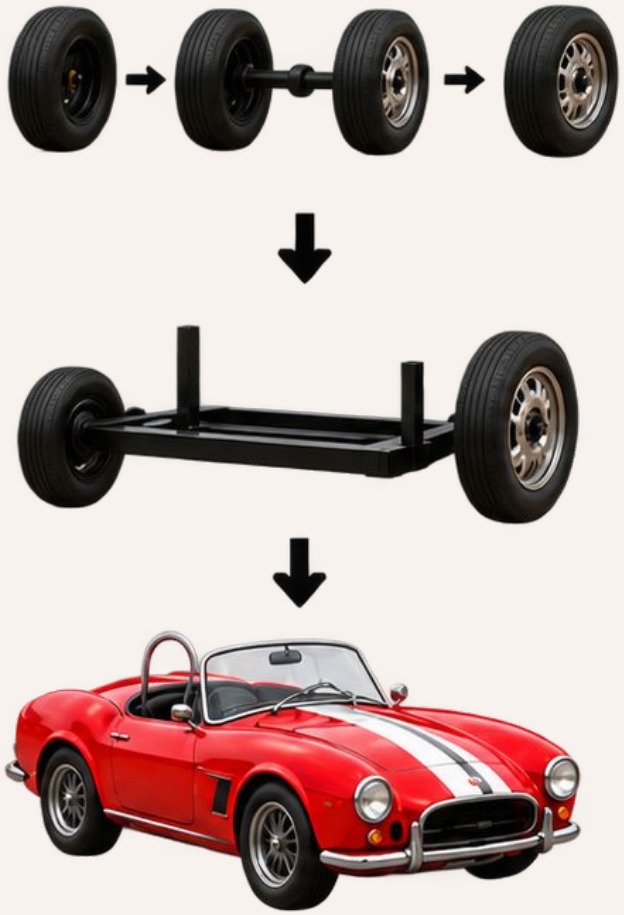


@AhmedBhula

#brightonSEO

WATERFALL

SEQUENTIAL ENGINEERING



AGILE

INCREMENTAL DELIVERY



VIBE CODING

NO PROCESS CHAOS



THIS IS HOW THE CHAOS STARTS

“Add a contact form”

Works perfectly 🤩

“Move it below the reviews section”

Form moves 🤩

But footer styling breaks 🤦‍♂️

“Fix the footer”

Footer fixed 🤩

But form no longer submits 🤦‍♂️

DEVELOPERS ALREADY SOLVED THIS.

These problems aren't new. Traditional developers have dealt with them for years.

8 RULES FOR VIBE CODING WITHOUT THE CHAOS



@AhmedBhula

#brightonSEO

ONE PROMPT, ONE JOB

Don't ask the AI to do three things at once.



✗ TOO BROAD

“Add a contact form, fix the footer alignment, and make the fonts bigger.”

✓ FOCUSED

“Add a contact form below the hero section with fields for name, email, and message.”

KEEP A PROMPT LOG

Every time you prompt the AI, log it in a simple spreadsheet



WHY IT MATTERS

Rebuild from scratch

If the project breaks beyond repair, replay your prompts in order. Skip ones that caused problems.

Find the exact cause

When something breaks, look back at the log and find which prompt caused the issue.

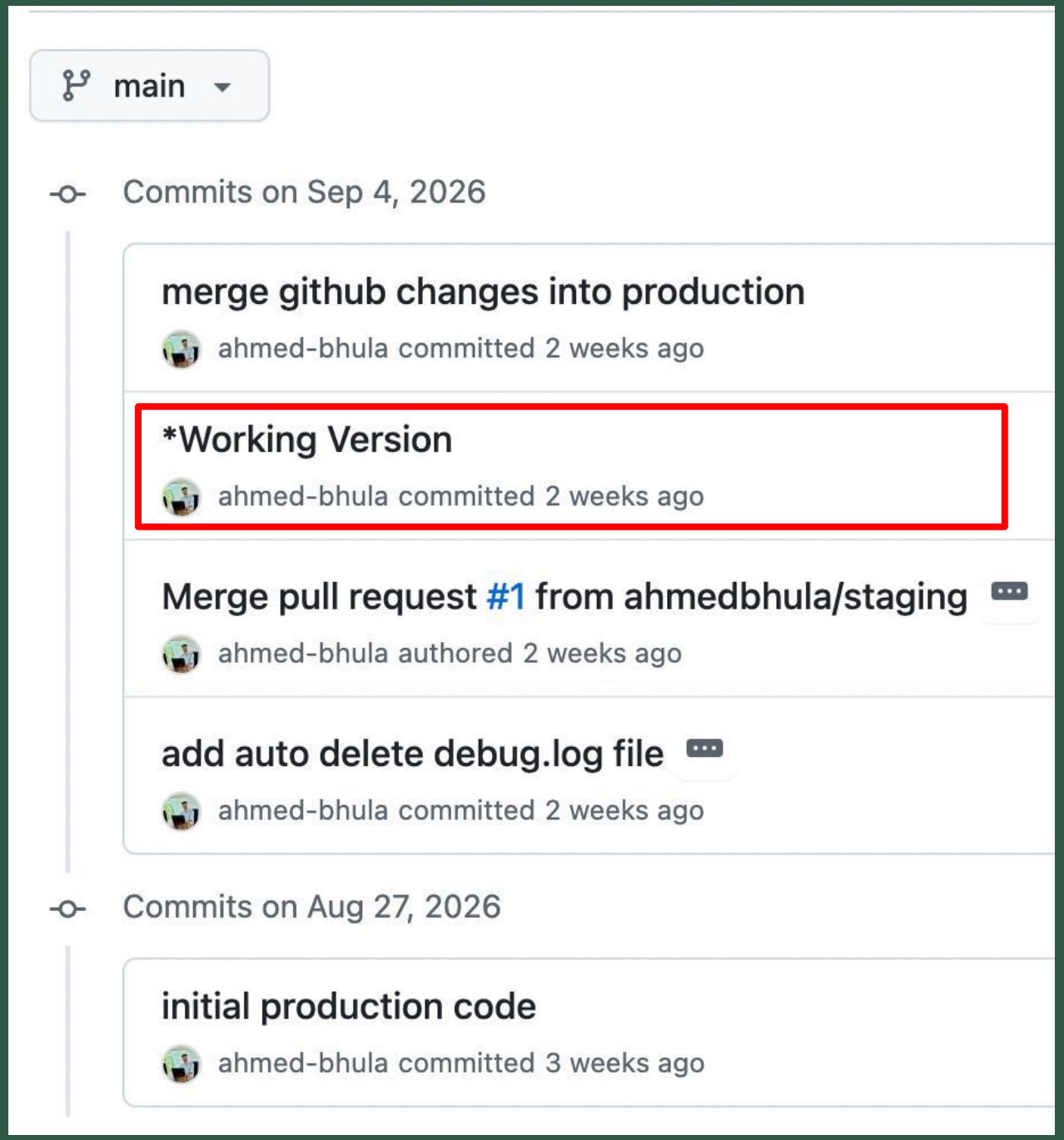
THE PROMPT LOG

#	What you asked	What changed	OK?
1	Add signup form	Form appeared on homepage	Yes
2	Change header colour	Header updated, no issues	Yes
3	Add date picker to booking	Date picker added, footer broke	No
4	Fix the footer	Footer fixed, form stopped submitting	No

SAVE BEFORE YOU EXPERIMENT



Before any prompt that makes a big change, save a working copy.



ALWAYS KEEP A WORKING VERSION YOU CAN COME BACK TO.

TEST WHAT YOU DON'T TOUCH



- ✓ Booking form submission works?
- ✓ Login still works?
- ✓ Navigation menu works?
- ✓ Checkout (if applicable) works?
- ✓ Contact form works?
- ✓ Mobile layout looks correct?
- ✓ No console errors?

FENCE YOUR PROMPTS



Tell the AI what to change and what *NOT* to change

✗ UNFENCED

“Add a date picker to the booking form.”

✓ FENCED

"Add a date picker to the booking form. **Do not change header, footer, navigation, or existing form fields.**”

KNOW WHEN TO REVERT



If you've prompted AI 2-3 times to fix the same bug and each fix creates another problem...

STOP

Revert to the last working version and try a different approach.

CHECK SECURITY BEFORE DEPLOYING



- ✓ No passwords or API keys exposed
- ✓ Authentication and permissions checked
- ✓ User inputs are validated and sanitised
- ✓ Dependencies and plugins are up to date
- ✓ No sensitive data in the code or database
- ✓ File permissions are correct
- ✓ Test on a staging site first (if possible)

ASK AI TO EXPLAIN THE CODE



Your Prompt

Explain what this code changed, which files were modified, and what could break as a result. Keep it in simple terms.

AI Response

- What the code does
- Which files were modified
- Potential risks or side effects
- Any follow-up actions

CHECKLIST

1. One prompt, one job.
2. Keep a prompt log
3. Save before experimenting
4. Test what you don’t touch
5. Fence your prompts
6. Know when to revert
7. Check security
8. Ask AI to explain the code

View my presentation and detailed notes at

REMDigital.co.uk/BSEO



@AhmedBhula

#brightonSEO