



Vercel-Friendly Optimization Prompt for Markdown-to-PDF Website

Prompt:

"Optimize my Markdown-to-PDF web app to be fully Vercel-friendly.

My project currently uses Vite or Next.js and may include Puppeteer for PDF generation.

Please:

1. **Update** `vercel.json`:
2. Remove any deprecated `builds` property.
3. Keep only the new style config:

```
{
  "framework": "vite",
  "buildCommand": "npm run build",
  "outputDirectory": "dist",
  "functions": {
    "api/**/*.js": {
      "maxDuration": 10,
      "memory": 1024
    }
  }
}
```

4. Make sure this file works with Vercel's current runtime.

5. Adjust PDF generation for serverless:

6. Replace `puppeteer` with `puppeteer-core` and import `chrome-aws-lambda` for headless Chrome in AWS Lambda.
7. In your API route (e.g., `/api/generate-pdf`), dynamically import:

```
const chromium = await import('chrome-aws-lambda');
const puppeteer = await import('puppeteer-core');
```

and launch with:

```
const browser = await puppeteer.launch({
  args: chromium.args,
  executablePath: await chromium.executablePath,
  headless: chromium.headless,
});
```

8. If Chromium isn't supported, use the lightweight `md-to-pdf` library instead.

9. Update package.json scripts:

```
{
  "scripts": {
    "dev": "vite",
    "build": "vite build",
    "preview": "vite preview"
  }
}
```

10. Check environment compatibility:

- 11. Use ES Modules syntax or CommonJS consistently.
- 12. Avoid local filesystem writes (use in-memory buffers).
- 13. Ensure the `/api` folder contains only `.js` or `.ts` serverless routes.
- 14. **Output final instructions** for deploying to Vercel, including how to set:
- 15. `NODE_ENV=production`
- 16. `CHROME_PATH` (if needed)
- 17. Any required environment variables (for secrets or config).

Produce the final working `vercel.json`, updated code snippets for serverless PDF generation, and explain why each change ensures Vercel compatibility."

Bonus Tip:

If your app is static (no backend), host the UI on Vercel and use **Edge Functions** or a small **API route** just for PDF generation.