

PureCode.ai

Table of Contents

- [PureCode.ai](#)
- [Table of Contents](#)
- [Landing Page Thoughts](#)
 - [`<head>` element](#)
 - [`<body>` element](#)
 - [Unauthenticated User](#)
 - [Desktop \(1440px viewport - MacBook Air MX\) - top to bottom page analysis](#)
 - [`<header>` element](#)
 - [Call-To-Action](#)
 - [Call-To-Action-Made-Up-Statistic \(😅 \)](#)
 - [Features Card Section \(General\)](#)
 - [Features Card Section \(Detail\)](#)
 - [Pre-Generated Components Example Grid](#)
 - [Testimonial Card "Carousel"](#)
 - [Pricing Comparison Chart](#)
 - [FAQ Section](#)
 - [Miscellaneous](#)
 - [Tablet \(768px viewport - iPadMini\)\[^1\]](#)
 - [Mobile \(430px viewport - iPhone 15 Pro Max\)\[^1\]](#)
 - [Authenticated User](#)
 - [Desktop \(1440px viewport - MacBook Air MX\)\[^2\]](#)
 - [Tablet \(768px viewport - iPadMini\)\[^1\]](#)
 - [Mobile \(430px viewport - iPhone 15 Pro Max\)\[^1\]](#)
 - [General Positive Aspects of Landing Page \(my bias - *strong opinions, weakly held*\)](#)
 - [Bugs/Possible Areas to Improve/General Critiques \(my bias - *strong opinions, weakly held*\)](#)
- [Product Thoughts \(web\) - \(very rough draft of my thoughts atm\)](#)
- [VSCode Extension Thoughts](#)

Landing Page Thoughts

`<head>` element

- [`<link rel="preload" as="image" href="/img/project_table_section_dark.png">`](#)
 - Suggest converting image to modern `.webp` format to align with other images on landing page and improve SEO score (included image conversion. see `images` directory; 20% size reduction vs `.png`, lossless conversion)

The screenshot shows a dark-themed application interface titled "Projects". A table lists several items with columns for Name, Duration, Size, Owner, Created, and Last Viewed. One row for "Add stock media (Jack's Demo)" is selected, highlighting the owner "Robert". A tooltip appears over Robert's name with the text "Make the text larger or change the colors of this element." and a blue "Update" button. Below the table, a "Pick & Edit" section with a cursor icon and the text "Select and edit any section within the component effortlessly" is visible.

- <meta name="description" content="A simple and powerful frontend AI copilot for the web.">
 - Currently when I google "frontend ui component ai generator copilot" - the friendliest keywords I could mash together - [purecode.ai](#) shows up as the 28th link

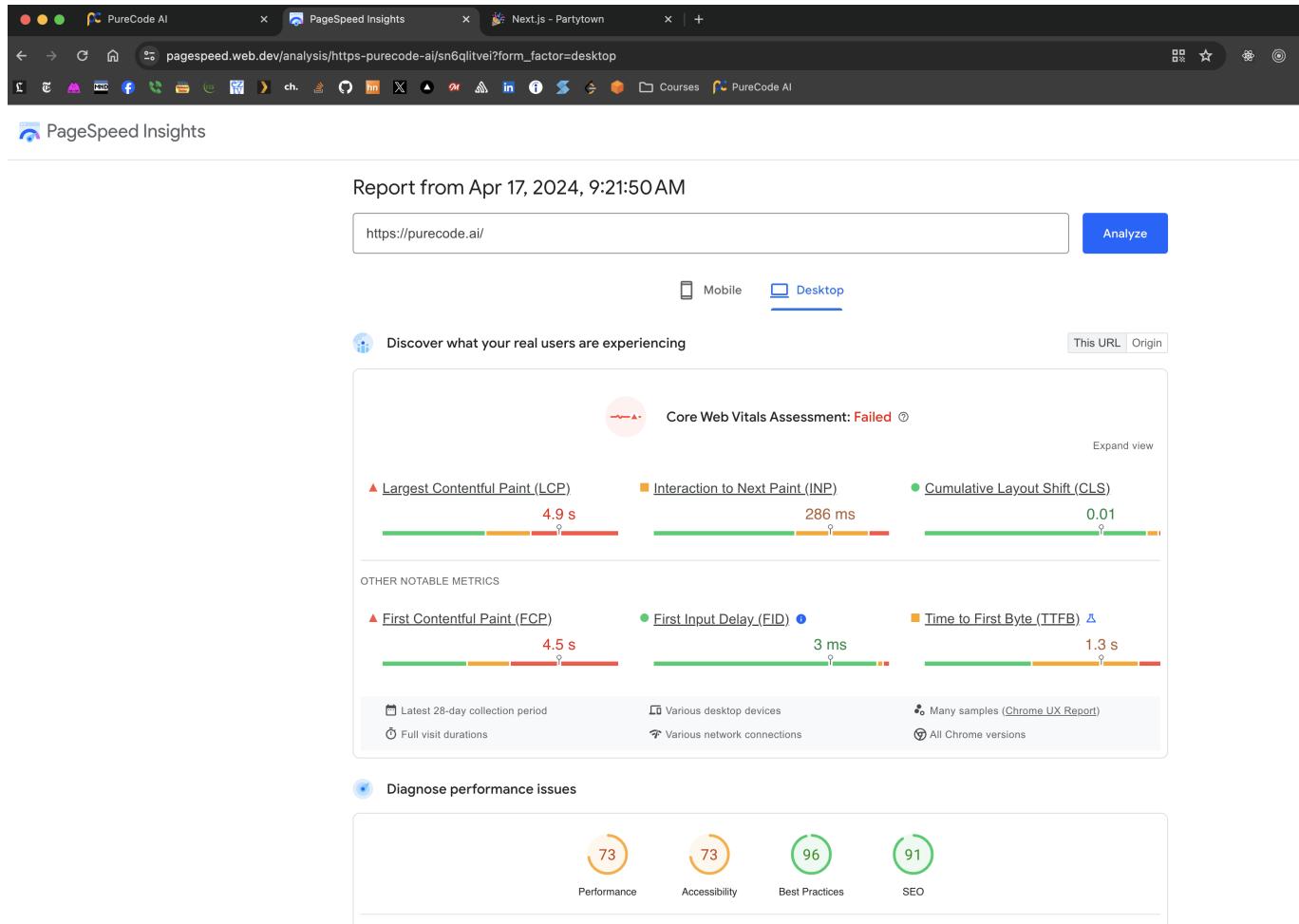
A screenshot of a Google search results page for the query "frontend ui component ai generator copilot". The results list includes links from Scalable Path, My Week With GitHub Copilot, and PureCode AI. A pink arrow points to the PureCode AI result, which is labeled as the "28th link".

- Suggest optimizing metadescription tag to include more keywords to move up SEO (e.g. *A simple and powerful frontend UI component generator for the web powered by AI copilot.*)
- Related: Suggest including meta_keywords tag to include additional keywords. (e.g. <meta name="keywords" content="UI, component generator, AI copilot, frontend copilot">)
- Suggest running site through [SEO optimizer](#)

The screenshot shows the SEOptimer SEO Audit interface. At the top, there are tabs for 'PureCode AI', 'project_table_section_dark.p', 'SEO Audit for purecode.ai - S', 'Save Image As PNG', 'Save Image As PNG', and 'SEOquake linkinfo'. Below the tabs, the URL 'seoptimer.com/purecode.ai#recommendation' is visible. The main header features the SEOptimer logo, a 'Website URL' input field, a 'Quick Audit' button, and navigation links for 'Features', 'Pricing', 'Resources', and 'Login'. A 'Premium - Free Trial' button is also present. The main content area is titled 'Recommendations' and includes a 'Toggle:' dropdown with options: Links, On-Page SEO, Social, Other, Performance, and Usability. Below this, a list of 10 recommendations is displayed, each with a category label (Links, Usability, On-Page SEO, Social, or Performance) and a priority color-coded box (High Priority, Medium Priority, or Low Priority). The recommendations are:

- Execute a Link Building Strategy (Links, High Priority)
- Optimize for Core Web Vitals (Usability, Medium Priority)
- Lengthen meta description (to between 70 and 160 characters) (On-Page SEO, Medium Priority)
- Add Canonical Tag (On-Page SEO, Medium Priority)
- Remove Duplicate H1 Tags (On-Page SEO, Medium Priority)
- Create and link your Facebook Page (Social, Low Priority)
- Create and link your X profile (Social, Low Priority)
- Use your main keywords across the important HTML tags (On-Page SEO, Low Priority)
- Make use of HTTP/2+ protocol (Performance, Low Priority)

- <script id="hotjar-init-script" crossorigin="anonymous">
(function(h,o,t,j,a,r){h.hj=h.hj||function(){(h.hj.q=h.hj.q||[]).push(arguments)};h._hjSettings={
hjid:3719365,hjsv:6,hjdebug:false};a=o.getElementsByTagName('head')[0];r=o.createElement('script');r.async=1;r.src=t+h._hjSettings.hjid+j+h._hjSettings.hjsv;a.appendChild(r);})
(window,document,'https://static.hotjar.com/c/hotjar-', '.js?sv=');</script>
 - Suggest offloading 3rd party analytics scripts off the main thread and on to service workers to increase [PageSpeed Score](#) (e.g. Hotjar) by implementing [Partytown package](#)



- Run site through a [semantic HTML inspector](#) (might be a lot of noise and not worth the effort - just including it to be thorough)

<body> element

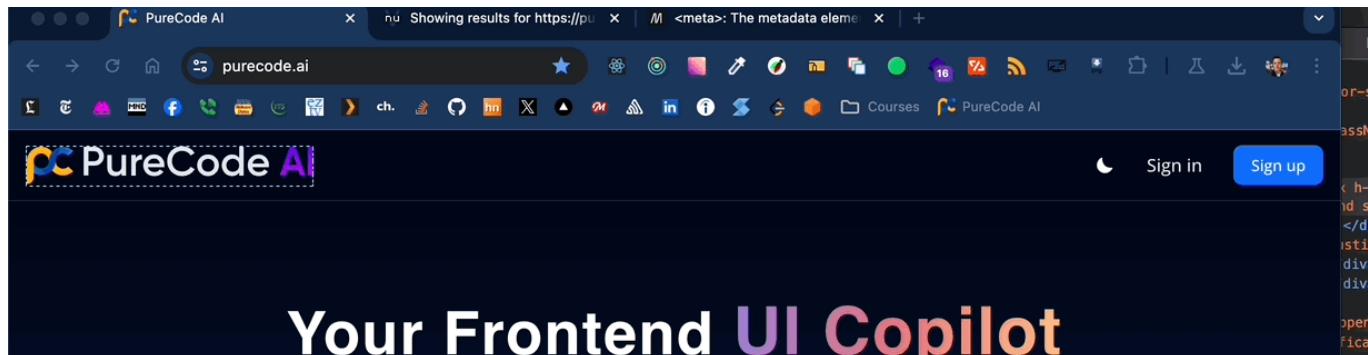
Unauthenticated User

Desktop (1440px viewport - MacBook Air MX) - top to bottom page analysis

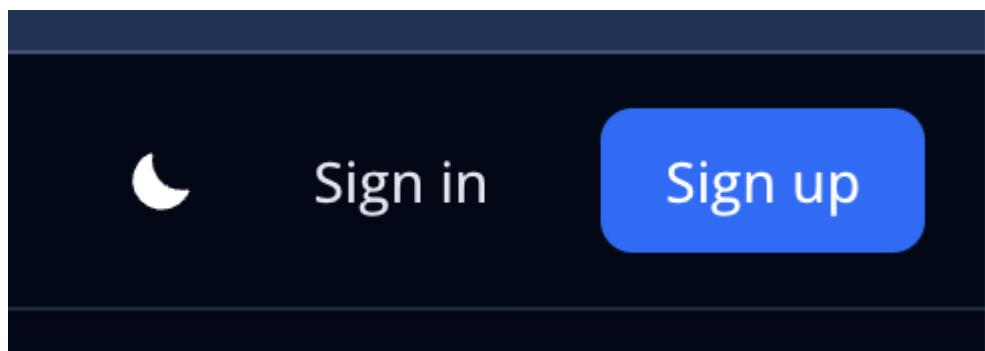
<header> element



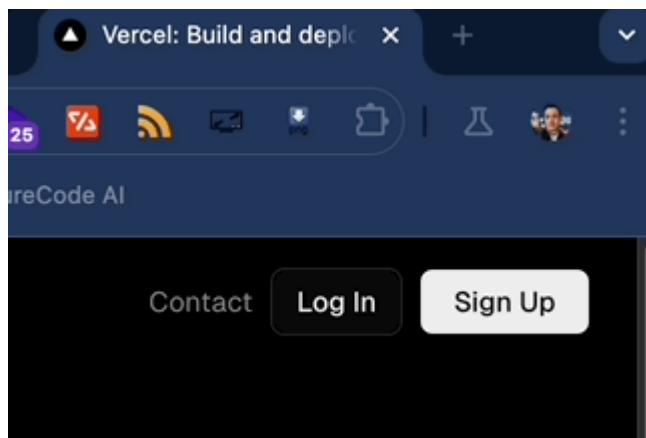
- NitPicky: [Sign In](#) button is slow to render wrt to [Sign Up](#) button. I've only used Clerk once in a side-project, but I don't recall this being an inherent issue with the library.



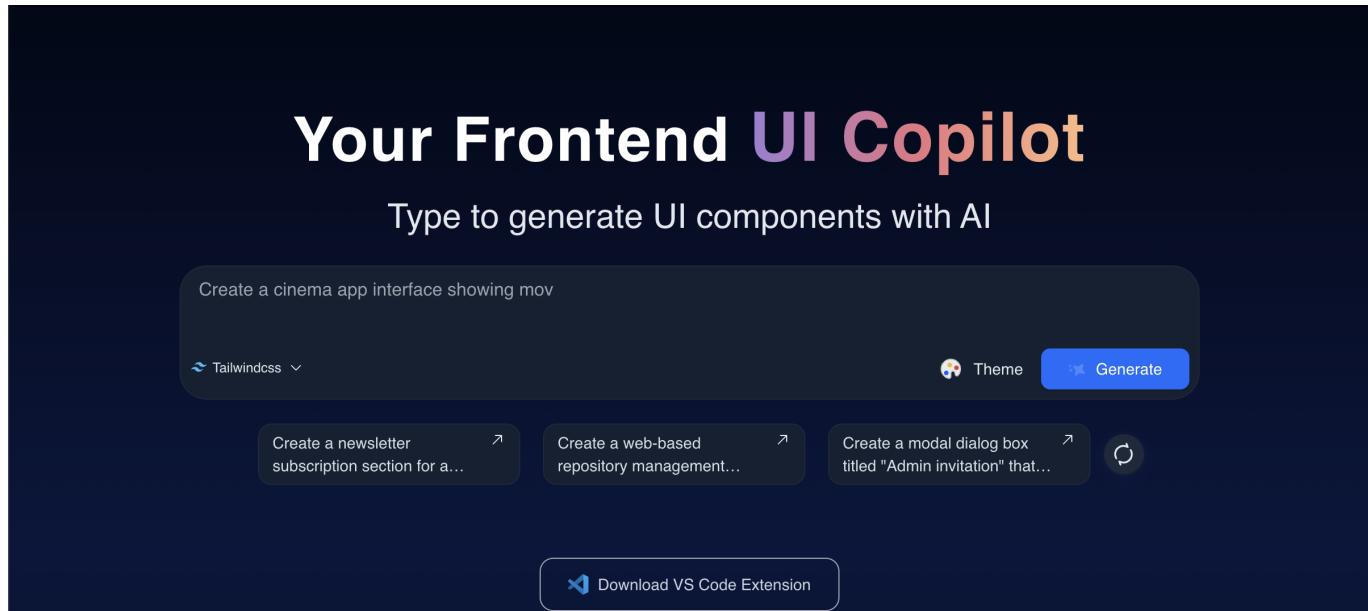
- NitPicky: **Sign In** font-size is noticeably larger (1rem vs 0.875rem) than **Sign Up** font - minor point but I can't seem to justify the inconsistency as a prudent style choice (most likely overlooked - again ... nit-picky)
 - I fixed the inconsistency below by changing **Sign In** font-size to 0.875rem



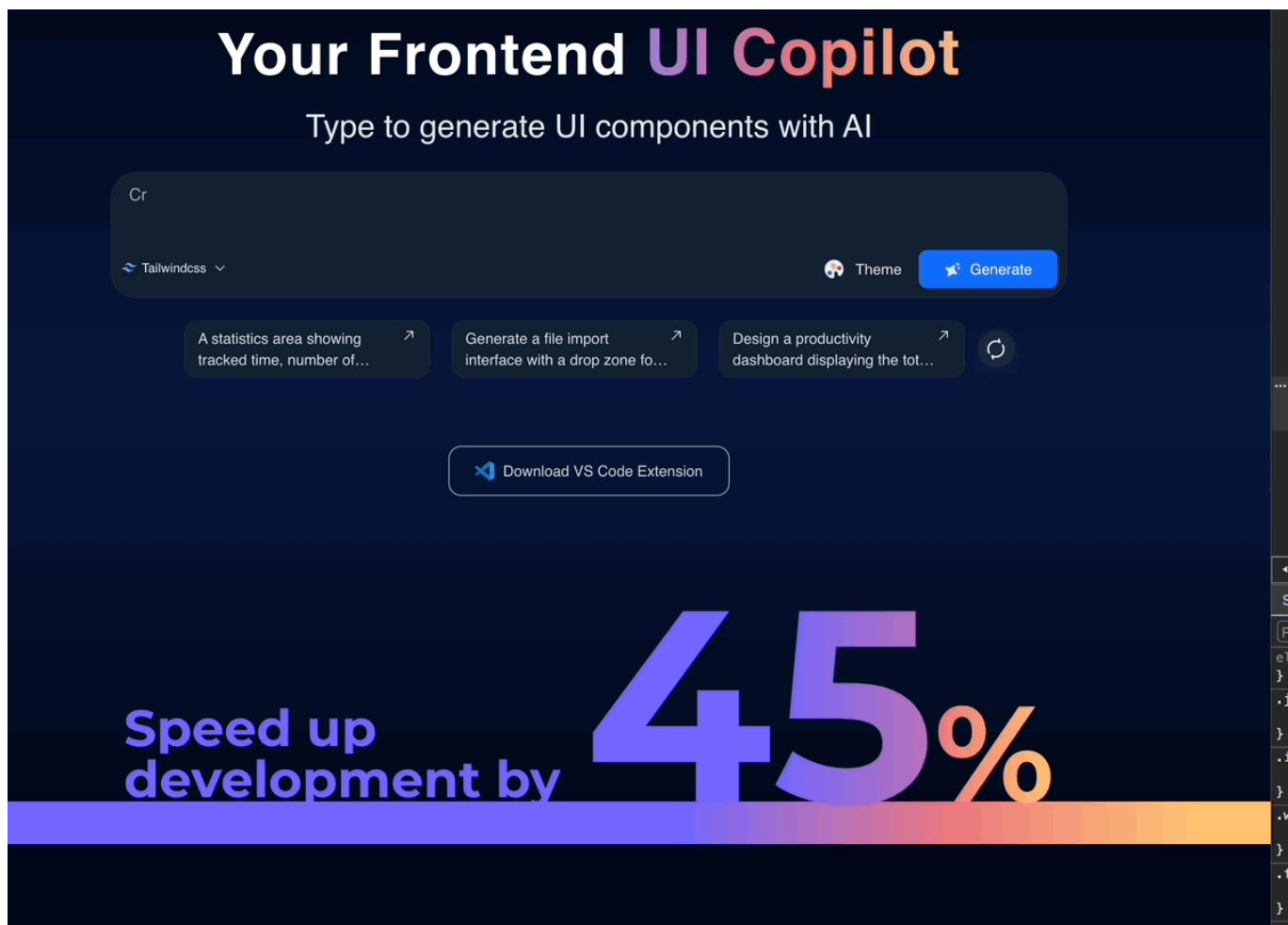
- NitPicky and EasyFix: The **Sign In** and **Sign Up** buttons would benefit from hover styles to enhance UX (see [vercel.com](#) gif below)



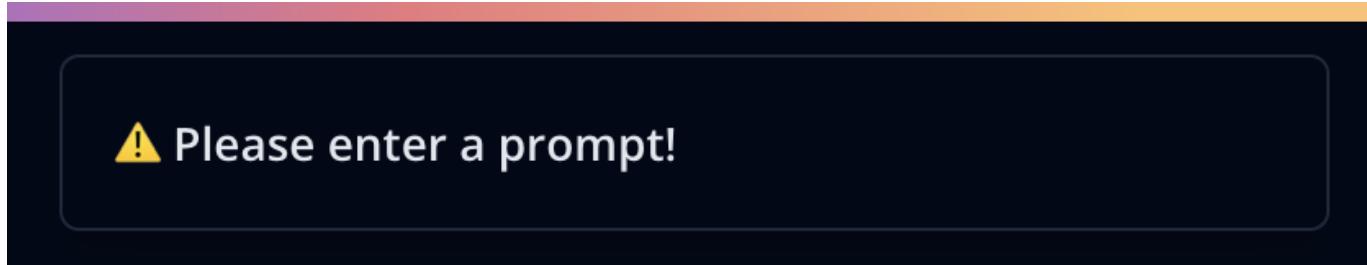
Call-To-Action



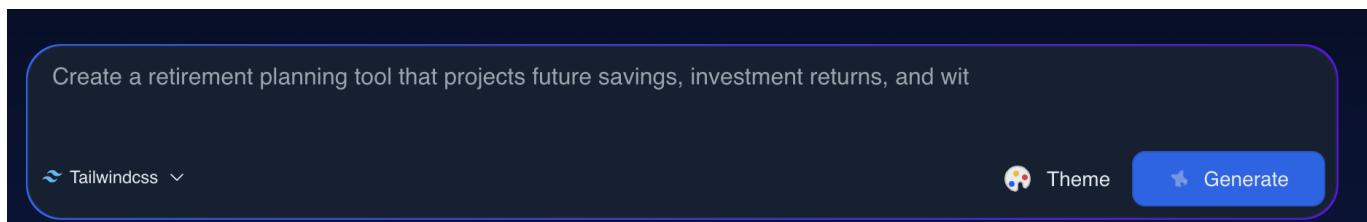
- StylisticBias 🎨 : If the user clicks **Generate** button without providing a prompt a feedback toast is presented to the user. Great! The toast is, however, some times difficult to notice because its background-color and the background-color of page is the same (toast border styling is the sole source of contrast here). There are multiple valid approaches to improve the UI here. One possible solution would be to add an emoji to the toast text in order to make it pop more and draw the user's eye towards it.



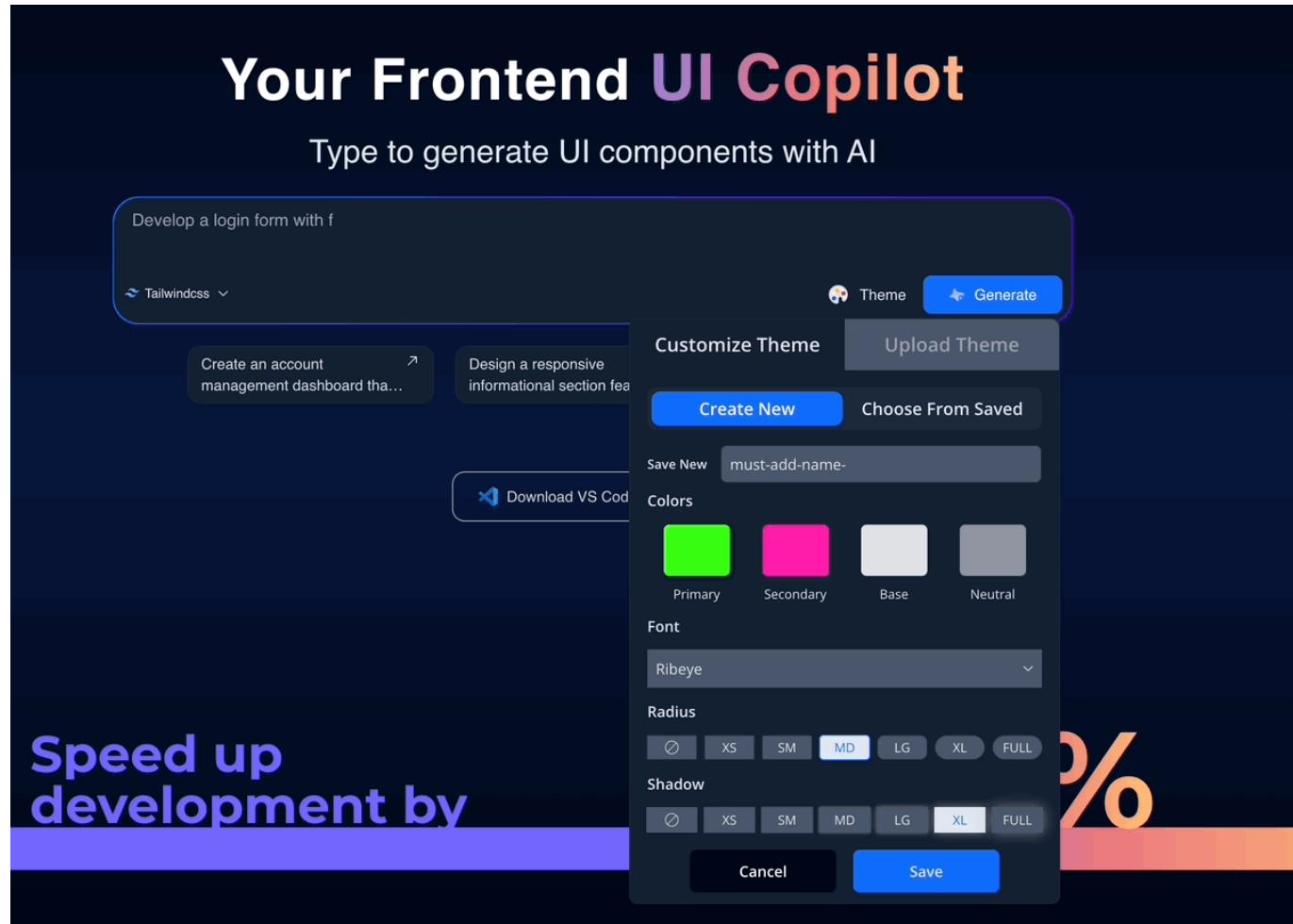
- Example of adding toast icon to enhance UX



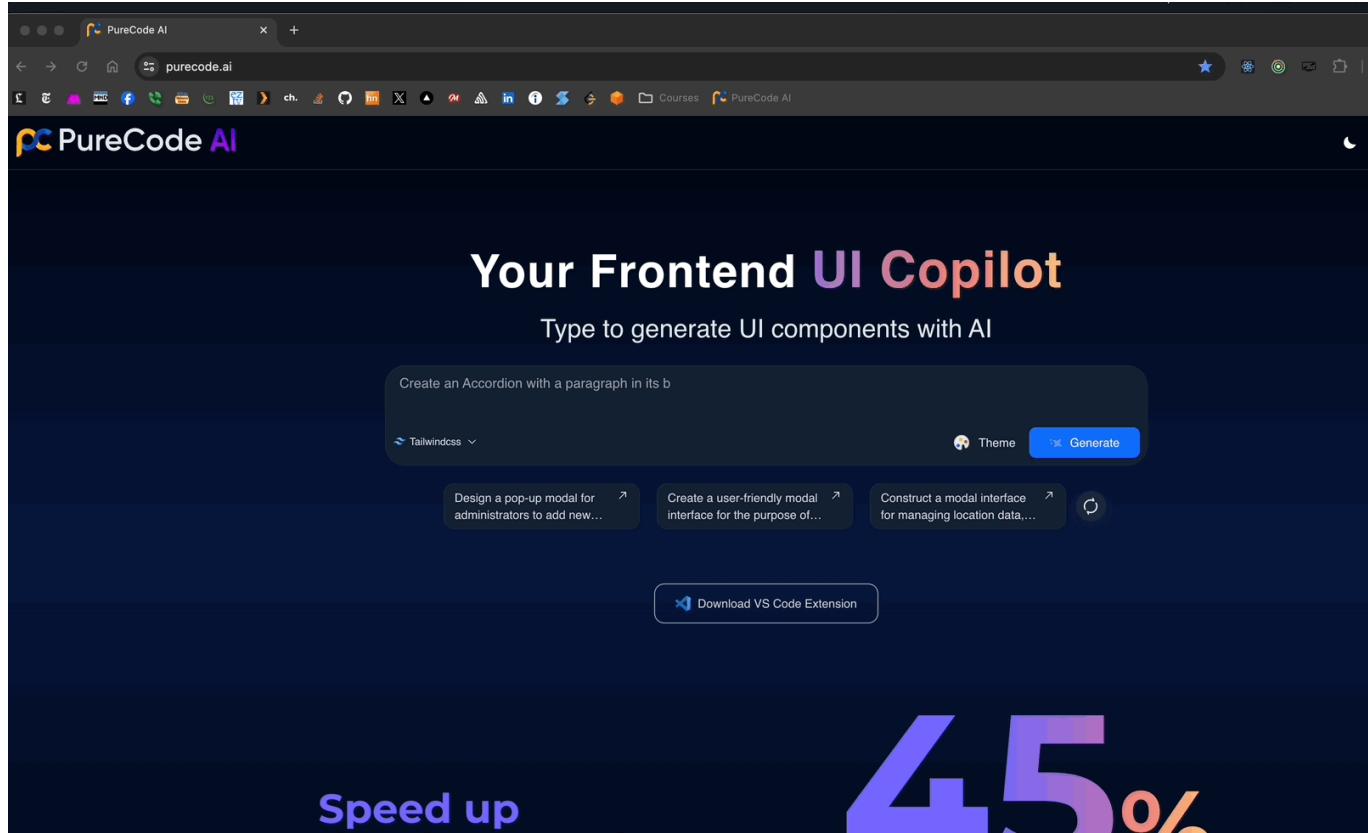
- UX-Kudos : I like the *prompt field focus styling* that is applied when the user clicks on the **Generate** button without providing a prompt. Nice touch!



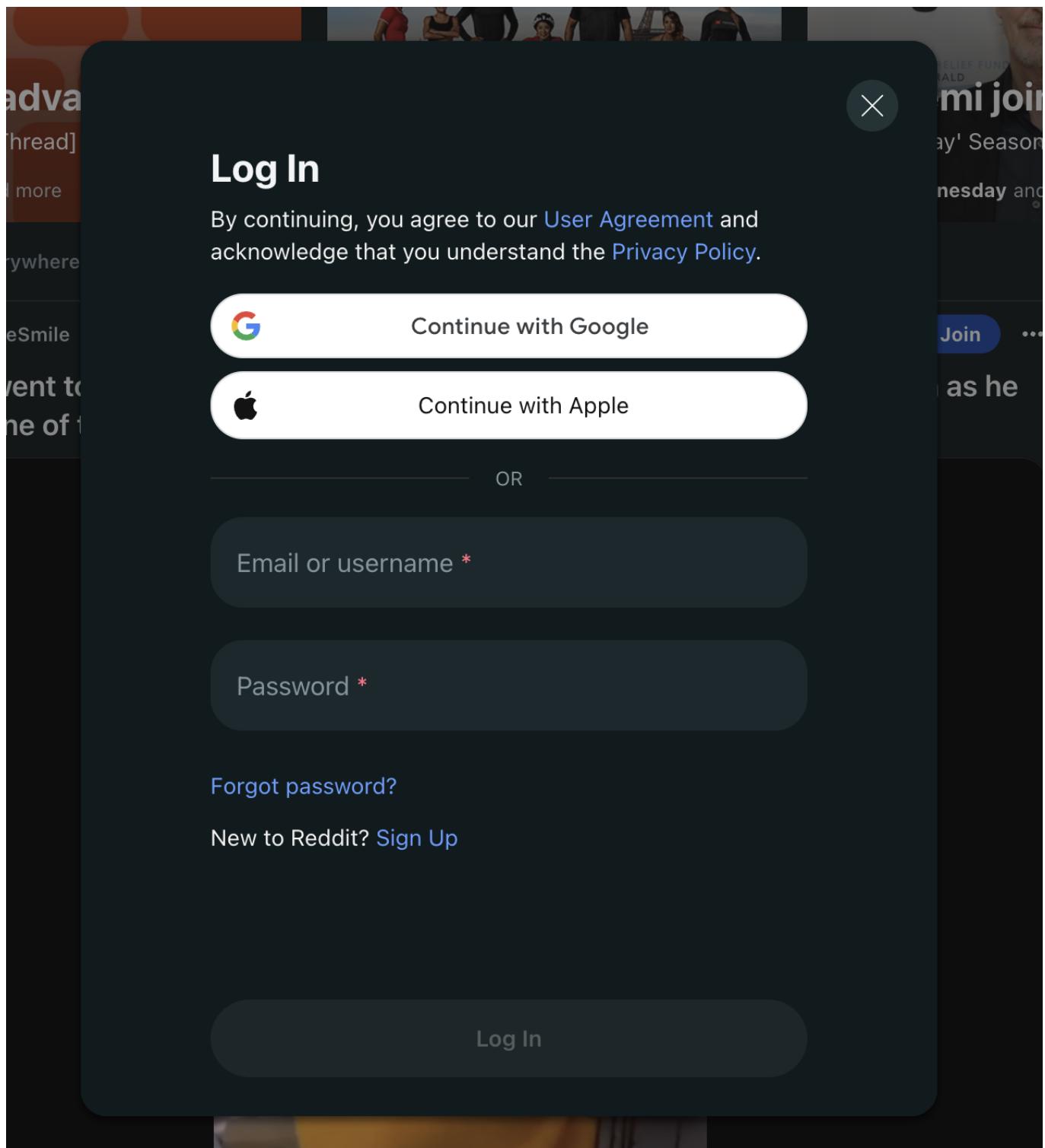
- Bug : It's not possible to **Save** a custom theme if the user is not authenticated.
 - Solution: hide the **Theme** button until the user is authenticated or prompt the user to authenticate via ClerkAuth when user clicks **Save** theme button - similar to the UX experienced when a user clicks on an example prompt.
- Confusing UX : It's not immediately apparent that the user needs to supply a value in the **name** field for the custom theme **Save** button to activate.
 - Suggest adding a tooltip that indicated the user must provide a name field when the custom theme **Save** button is hovered or on attempt to click disable button we could trigger another feedback toast.
 - More important is that this feature doesn't work for unauthenticated user's (which makes sense, but feels broken nonetheless - see Solution above)



- Bug 🐛 (or stylist-bias - very weakly held opinion here): When user clicks on a template prompt, they are prompted to authenticate via the Clerk modal. Great! However, if the user decides to navigate click the browser's back button the modal persist across page views which leads to a confusing UX (aside from reddit - which isn't known for great ux - I can't find another site on the web that has a similar auth modal behavior on browser navigation). Most sites seem to prefer a full-page login/sign-page over a floating modal. I will admit this might not be a "bug", just a bit odd, but certainly low priority.
 - Expectation: I would expect the authentication modal to vanish upon Back browser navigation.



- Minor UX Quirk 😊 : The only way to dismiss the authentication modal is for the user to hit the escape key. It would be nice if there was an explicit close (X) button on the modal or if the modal vanished if the user clicked outside of the modal frame. See reddit authentication modal.

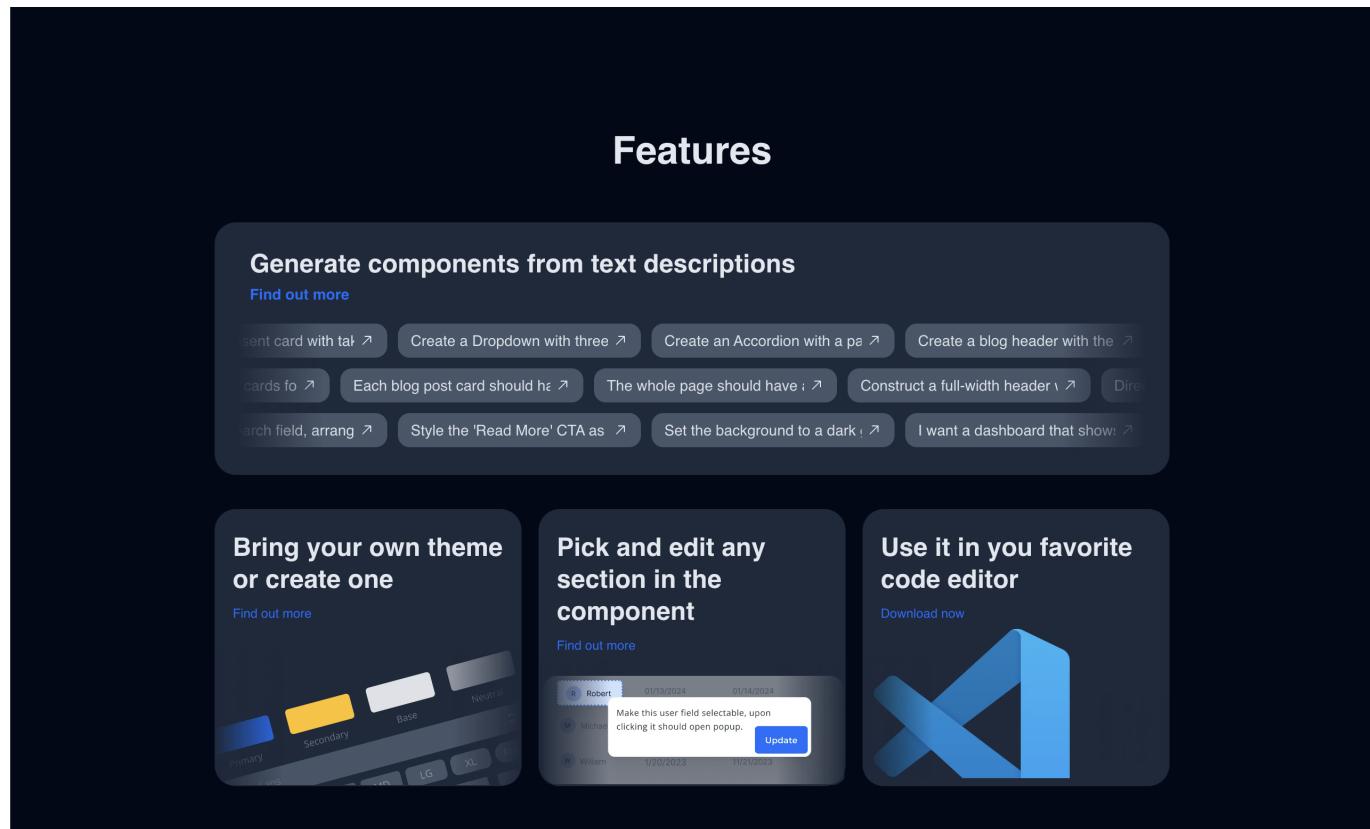


Call-To-Action-Made-Up-Statistic (😅)



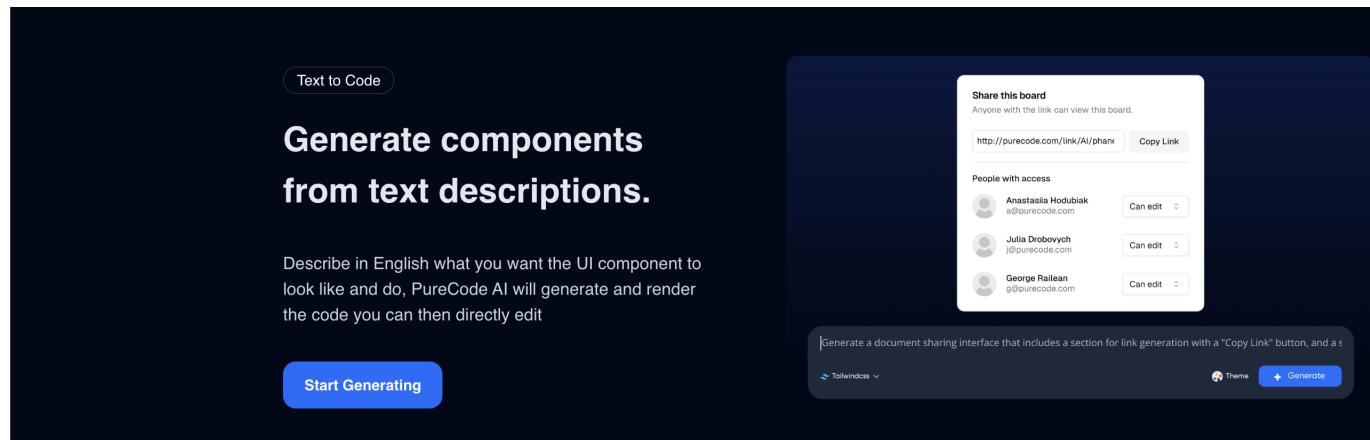
- No constructive comment. Ship it! 🚛

Features Card Section (General)



- Only one comment (nit-picky). Why doesn't the wide-card ([Generate components from text descriptions](#)) animate and grow on hover and provide a `pointer: cursor` to indicate that the entire card is clickable? Seems inconsistent. In function, it's no different than the cards that link to detail feature card 02 through 04. I do notice that the `cursor: pointer` shows up when a user hover over the example prompt rows (nice little left-right animation of the prompts - fancy)

Features Card Section (Detail)



The screenshot shows the Tailwind AI interface. On the left, there's a "Customize Theme" panel with sections for "Colors" (Primary, Secondary, Base), "Font" (Open Sans), "Radius" (XS, SM, MD, LG, XL), and "Shadow" (XS, SM, MD, LG, XL). It also has "Create New" and "Choose from Saved" buttons. In the center, there's a "Save new" button and a code editor window titled "Theme 3" containing the following configuration:

```

    // * @type {import('tailwindcss').Config}
    module.exports = {
      colors: {
        primary: "#1f78a6",
        secondary: "#7a58e7",
        orange: "#ff78a9",
        green: "#13c656",
        blue: "#1f78a6",
        gray: "#8492a6",
        dark: "#33334d46"
      },
      fontFamily: {
        sans: ["Graphik", "sans-serif"],
        serif: ["Merriweather", "serif"]
      }
    }
  
```

On the right, there's a "Theming" section with a "Create or Upload" heading, a sub-section "Generate components that match your theme by providing config.js files or creating from scratch.", and two buttons: "Create theme" and "Upload theme".

The screenshot shows the Tailwind AI interface. On the left, there's a "Fast and Easy Updates" section with a "Pick & Edit" button and a note: "Update generations with speed and efficiency by selecting part of the rendered component and entering the new text description." Below it is a "Start generating" button. On the right, there's a "Projects" section listing several tasks:

- Touch up audio edits with Regenerate (Jack's Demo)
- Add stock media (Jack's Demo)
- Improve audio quality w/ audio Sound (Jacks Demo)
- Remove filter words (Jack's Demo)
- Transcribe audio to text automation (Jack's Demo)

A tooltip over one of the items says: "Select the text larger or change the colors of this element." There's also an "Update" button.

The screenshot shows the Tailwind AI interface integrated with VS Code. The left side shows the VS Code interface with a "Integration Options" panel connected to Tailwind AI. The main area displays a snippet of code:

```

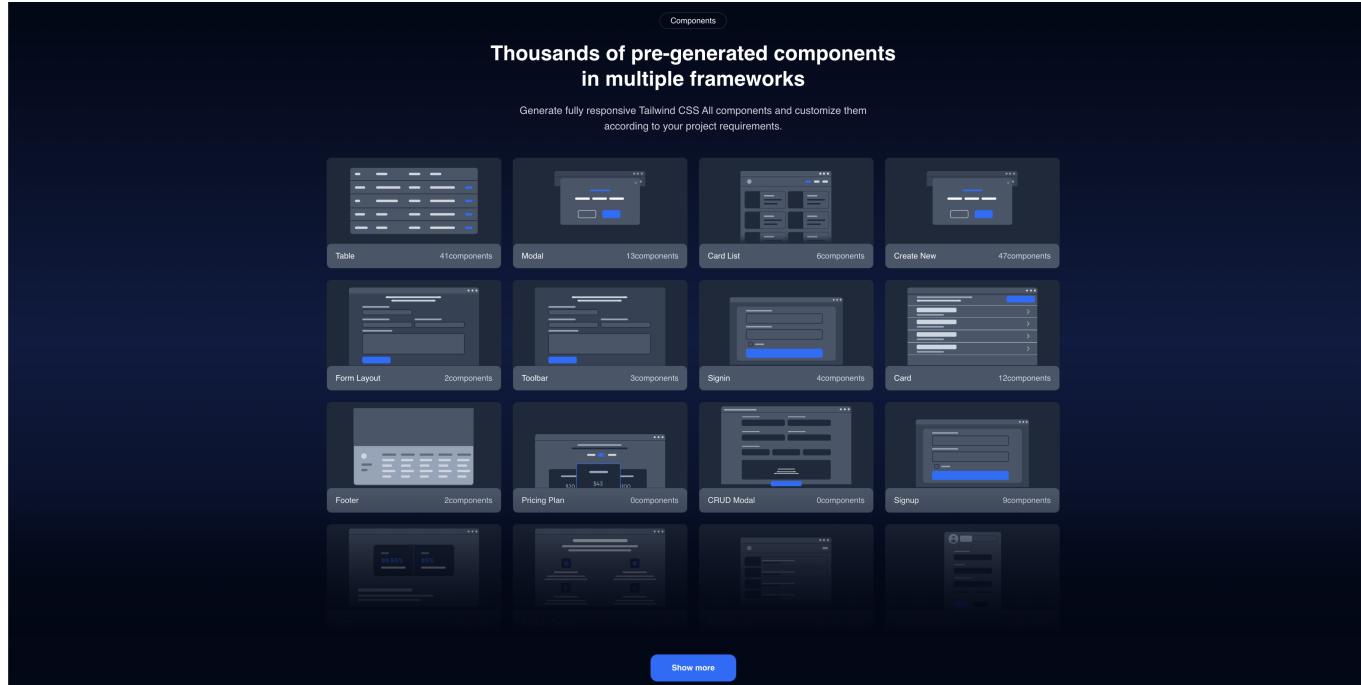
    // This file is auto-generated to help you code faster
    import { Head, Scripts, Tailwind } from '@tailwind/react';
    import { Mail, Facebook, Twitter, LinkedIn } from '@tailwind/react';
    import { MailIcon, FacebookIcon, TwitterIcon, LinkedInIcon } from '@tailwind/react';

    const App = () => {
      return (
        <Head>
          <title>Tailwind AI</title>
          <meta name="viewport" content="width=device-width, initial-scale=1" />
        </Head>
        <Scripts>
          <script src="https://unpkg.com/tailwindcss@^2/dist/tailwind.min.js" />
        </Scripts>
        <div>
          <h1>Hello World!</h1>
          <p>This page is generated by Tailwind AI. You can edit the code above and see the changes reflected here in real-time.</p>
          <div>
            <MailIcon /> <FacebookIcon /> <TwitterIcon /> <LinkedInIcon />
          </div>
        </div>
      );
    };
  
```

On the right, there's a "VS Code Extensions" section with a "Do it all in your favorite code editor." heading, a note: "Generate, update and preview the component along with code. All with-in VS code.", and a "Download extension" button.

- No comment aside from a suggestion to add some subtle animation to the blue button on hover. Ship it! 🚚

Pre-Generated Components Example Grid



- Looks great. Ship it.

Testimonial Card "Carousel"

Don't just take our word for it

Remarkable developer experiences that enable success stories, empower businesses, and fuel growth across industries and individuals.

PureCode AI to instantly make production ready components cuts out hours of the development work ...

Tucker Front End Developer

Uploading our project theme and instantly generating components has been a game changer for the team. A must...

Julia DeBoer Front End Engineer

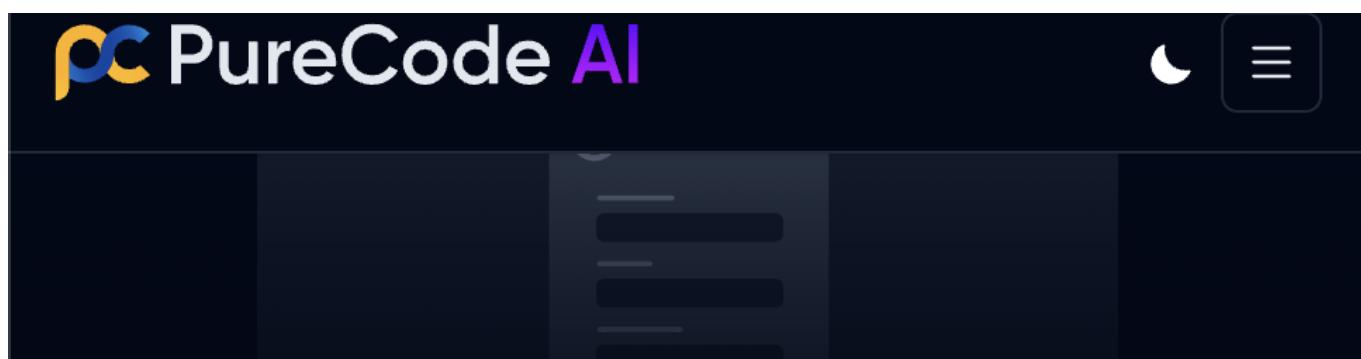
I'm delighted with PureCode.ai! The VS Code extension is incredibly useful for developers looking to save time writing...

Suman Chopra Senior Full Stack Developer

This tool saves me considerable time and effort on the front end. After prompting, it creates exactly what I need, ...

Olivia Marcello React.js Developer

- No comment on desktop, but on smaller viewports it would be nice if the "carousel" was in fact functional (i.e. having the cards actually slide left-right when the user swipes across the carousel row)



Account Overview

0 components

Show more

Don't just take our word for it

Remarkable developer experiences that enable success stories, empower businesses, and fuel growth across industries and individuals.

“Using PureCode AI to instantly generate production ready components cuts out hours of mundane development work on projects.”

Rob Tucker 

Lead MERN Developer

“Uploading generating changer for React.js de

Julia DeB

Front End En



Pricing Comparison Chart

Start your 7 day **UNLIMITED FREE TRIAL**

Monthly Annual -20% off

Pro	Premium	Enterprise
\$19 /monthly	\$39 /monthly	Contact Us
Start free trial	Start free trial	Talk to Sales
Custom Theming	✗	✓
Unlimited component generations	✓	✓
Unlimited component iterations/updates	✓	✓
Private generations	✓	✓
Add multiple users	✗	✓
SAML SSO	✗	✓
Data Privacy assurance	✓	✓
Flexible Commercial Terms	✓	✓

- Ship it. Love it. Buttons have hover animations. ❤️

FAQ Section

FAQ

What frameworks does PureCode AI support?

How can I bring my own theme?

Why should I use the VS Code Extension?

How can I generate better components?

How does the pick and edit feature work?

- no comment

Miscellaneous

- Final comment: Where's a footer? Privacy Policy, Terms of Service, etc. Low priority I'm sure because the landing page is not the product. 

Tablet (768px viewport - iPadMini)[^1]

Mobile (430px viewport - iPhone 15 Pro Max)[^1]

Authenticated User

Desktop (1440px viewport - MacBook Air MX)[^2]

- Comments to come in a few hours

Tablet (768px viewport - iPadMini)[^1]

Mobile (430px viewport - iPhone 15 Pro Max)[^1]

General Positive Aspects of Landing Page (my bias - *strong opinions, weakly held*)

- Stay tuned! Quick words that pop into my head on first impression. Modern. Great dark color choices and use of gradients.
- 98% mobile responsive at first glance

Bugs/Possible Areas to Improve/General Critiques (my bias - *strong opinions, weakly held*)

- More to come here soon.

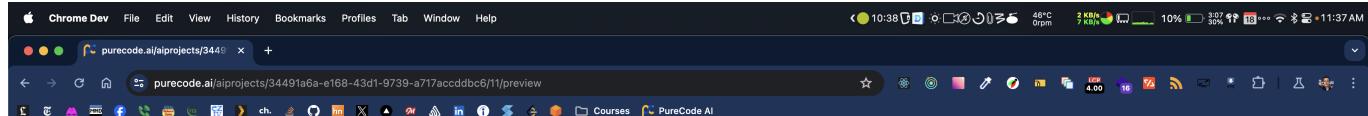
Product Thoughts (web) - (very rough draft of my thoughts atm)

- The meat and potatoes of PureCode.ai and where I first tried to give feedback, but when I started to poke around the web app I was unable to make use of its generative powers because the Azure backend was generating a **502 – Bad Gateway** error that rendered the app totally unusable.

The screenshot shows the PureCode AI web interface. On the left, there's a component editor titled "Select Component" with several card designs listed. On the right, there are tabs for "Canvas" and "Code". Below the editor is a large text area for "Describe your component here." At the bottom right of the editor area is a button labeled "Download VS Code Extension".

At the bottom of the screen, the Firefox Developer Edition DevTools are open, specifically the Network tab. This tab lists numerous network requests, many of which are failing with a status of "502 Bad Gateway". Some requests are successful, showing JSON responses like "tokens7_clerk_js_version=4.71.4". The failed requests are all to the URL "https://purecode-next-backend.azurewebsites.net/api/v1/chat-new-chat". The failed requests show various initiator details (e.g., "clerk.browser.js:2 fetch", "fetch", "clerk.browser.js:2 (etc.)") and response headers indicating CORS issues ("CORS Missing Allow ...").

- The web app now works. I've played with it some. I'll post my feedback in a video walk along. Quick thoughts: Impressive, versioning of multi-step prompts has a lot of possibility to make this a very useful product. Web app, however, does seem a bit brittle and I find it necessary to operate always with DevTools open to obtain sufficient feedback on how the generative process is progressing. App-level crashes happen quite often, although typically a hard browser refresh enables one to continue building on previous prompt steps. At times, a new step will cause previously working functionality to vanish (ai forgetfulness is a thing)
- I wonder how long the web app has been in development?
- It's a shame that manually editing generated code is quite buggy at the moment (difficult with cursor placement and any attempts at editing generated code leads to garbled mess.)
- It currently feels like a developer-level of knowledge is needed by a user to extract useful functionality from the app.
- It would be nice to add an appropriate React error boundary to limit effective crash area when the error below takes place (caused by an attempt to access undefined fields on render (`<ErrorBoundary/>` where are you?)). Hard-refresh usually allows for progress to be made.



VSCODE Extension Thoughts

- I installed the VSCode extension, but it's currently not publicly functional: "*We are rolling out new feature in this extension*"
 - Extension released on 23 March 2024. Last updated 16 April. Alpha stage.
-

[^1]: Thoughts not included for brevity (and given the use case for the site I suspect most users will be on a laptop) - I would be willing to explore in further detail if requested [^2]: For sake of brevity, I will avoid any nit-picky stylist observations and only focus on bugs in the UI/UX - again I am willing to explore in more detail upon request