

Contrast and Readability Audit for “Pretty Much” Site

Global Styles & Color Scheme

The site's base design uses light backgrounds with dark text, which is a sound starting point for readability. The primary brand colors defined – **Brand Navy** (`#0F1C2E`), **Brand Blue** (`#0066CC`), **Brand Gold** (`#FFB81C`), and a lighter **Brand Cream** accent (`#e9c77b`) – need to be applied with care to meet contrast standards ¹. In general, dark colors from the brand palette (navy or dark blue) should be used for text on light backgrounds, while light colors (white or very light tints) should be used for text on dark backgrounds ² ³. For example, using black or navy text on a white background yields excellent contrast (10:1 or higher), whereas using the gold or cream text on white is **insufficient** (only around ~1.7:1 contrast, far below the WCAG AA minimum 4.5:1). Indeed, previous audits found instances of gold text blending into light backgrounds ⁴. The remedy is to reserve gold/cream as accent **background** colors or for icons, and avoid them as body text on white. If a golden accent is needed in text, consider placing it on a dark backdrop or outlining it for contrast. The project has already moved in this direction by substituting many gold-on-light usages with darker text ⁵ ⁶. All new design tweaks should continue to **enforce at least AA contrast (4.5:1)** for normal text ⁷ using the centralized contrast variables and color system put in place.

Link Styles: By default, links use the dark brand blue/navy for text, which on light backgrounds is highly legible. However, the current hover style is to recolor links to the gold accent (e.g. a dark blue link on white changes to gold on hover) ⁸. This presents a contrast issue: gold (`#e9c476` – `#FFB81C`) against white has a ratio around 1.7:1 **[11†]**, making hovered links hard to read. To fix this, it's recommended to either **darken the hover color** or change the hover indication method. For example, use the darker gold variant (the “gold700” shade, e.g. `#8C7647`) or even Brand Navy for hover, which would provide ~9:1 contrast on white. Alternatively, keep the text color dark on hover and use an underline or subtle background highlight to indicate hover state. This ensures the link doesn't “disappear” when hovered. The main goal is to avoid any state where a light text color meets a light background. Consistently using the new `contrast-variables.css` (which maps “foreground” colors appropriate to each background) will help – e.g. on any light section, the CSS should automatically apply a dark hover color instead of the gold ⁷. In summary, **gold or cream text should only appear on sufficiently dark backgrounds**, and link hover styles should reflect that rule.

Navigation Bar

The header navigation must remain legible whether it overlays a hero image or sits on a colored bar. Currently, the site uses a transparent header that often overlays the page's hero section. To maintain readability, **all nav links are now rendered in white with a subtle text-shadow** ⁶ when over the hero. This is a good practice for contrast on a busy or dark image background – the white text has a ~12:1 contrast on a dark image or overlay, and the added shadow further improves readability against varying backgrounds. On pages where the header might sit over a solid color (for example, if a page header has a

Brand Blue background), the nav links are likewise forced to white text ⁹. This addresses past issues where nav items were inadvertently using dark text on a dark backdrop or vice versa ¹⁰. One thing to watch is the **hover state** of nav links: currently, on dark headers, hovering turns links to the Brand Gold color ⁹ ¹¹. On a very dark navy background this gold (\approx #FFB81C) is acceptable (contrast \sim 7.3:1 on navy), but on a mid-tone blue background (or over the lighter portion of a gradient) that same gold can drop to \sim 3:1 contrast. The design team should ensure that the header background is sufficiently dark wherever gold hover text is used, or else consider an alternative hover style. One solution is to keep nav text white on hover as well, perhaps underlining it or lightening the background behind it for feedback. In code, this could mean removing the forced gold hover in favor of an underline, or only applying the gold hover on pages where the header background is the very dark navy.

For the **mobile menu**, the recent changes are positive. When opened, the mobile nav now appears on a nearly black translucent backdrop (`bg-black/90` with backdrop-blur) with all menu text in white ¹². This high-contrast approach (white on black) ensures the menu items are easily readable on small screens, addressing any previous issue of text blending into a transparent background ¹³. The menu's close "X" button is also white on a dark circle, which is good. Each mobile nav item still uses a slight hover or active highlight (e.g. a lighter black background on hover, and the **Agent Portal** item is given a gold background as a call-to-action) ¹⁴ ¹⁵. These are implemented with Tailwind utility classes and inline styles – they generally follow contrast rules: for example, the Agent Portal button in the mobile menu is `bg-amber-500` with dark text ¹⁴ ¹⁶, meaning a gold-orange background with black text (which is high contrast). The one suggestion here is to ensure consistency: the desktop nav's Agent Portal uses the Brand Cream (`#e9c77b`) with navy text ¹⁷, whereas the mobile uses a slightly different gold (amber). It would be more consistent to use the same brand gold/cream color in both places. But from a purely contrast standpoint, both variants are working because they use **dark text on a light gold button**.

Finally, if the header is intended to become opaque on scroll (currently it remains `bg-transparent` with a shadow ¹⁸), consider adding a solid background color once the user scrolls past the hero. For example, a white or navy header on scroll would prevent the white nav text from ever sitting over unpredictable page content. Right now, the header uses a drop shadow to stand out when scrolling, but the text is still white – if the content it overlaps is white (imagine scrolling quickly so that the hero is gone and the header overlays a white section), the nav links would become invisible. A simple fix is to toggle a class like `.bg-white text-brand-blue` on the header when scrolled (or use a semi-transparent blur background) so that contrast is maintained in all scenarios. This would ensure no accidental white-on-white situations occur.

Hero Sections (Banner Text and Images)

Hero sections often have text over images or colored backgrounds, which is a prime area for contrast problems. The review identified that previously some hero banner text was hard to read against background visuals ⁴. The development team implemented several fixes to address this, which we can reinforce as best practices. First, **all hero titles and subtitles are now rendered in a high-contrast color with heavy weight and a shadow** ¹⁹. In practice, this means using either pure white or a very light brand color for the text, and adding a text-shadow to distinguish the letters from the background. For instance, the tagline/heading on the home page hero was switched from a thin, light font to a **bold face in Brand Cream (#e9c77b) with a soft black shadow** ²⁰ ²¹. This choice of a light-but-not-white text color over a darkened image provides good contrast while aligning with the brand's gold tone. The text-shadow (e.g. `0 2px 6px rgba(0,0,0,0.4)`) helps **outline the text** against any lighter patches in the background image, significantly improving readability ²⁰ ²¹. An alternative here could have been to use pure white text, but

the cream/gold still achieves a strong contrast due to the dark overlay (and yields ~13:1 contrast against near-black backgrounds). The key recommendation is to **always overlay hero images with either a semi-transparent dark layer or text shadow (or both) when placing light text on them**, which is exactly what's been done in code ²² ²³. The site now injects a **gradient overlay** on hero banners – for example, a black-to-transparent gradient is placed over hero images ²². This ensures the top of the image (behind the page title and nav bar) is sufficiently dark, and it gradually lets the image show more toward the bottom. This gradient overlay technique dramatically improves text legibility on heroes without entirely losing the visual background, and it's a textbook solution noted in the documentation ²⁴.

All hero section headings (usually `<h1>` or `<h2>`) and any hero callout text should meet at least 4.5:1 contrast against the backdrop. Given the large font sizes used in heroes, they technically could meet the AA large-text standard at 3:1, but aiming higher is wise. With the current styling – bold, light-colored text on dark overlays – the heroes meet or exceed these ratios. We should also check secondary text in heroes: if there's a tagline or smaller paragraph, it often appears in a slightly lighter shade (e.g. a light blue-gray `#EFF6FF` is used for hero paragraphs on dark backgrounds) ²⁵ ²⁶. That color (`#eff6ff`) is essentially a very pale blue that still contrasts ~4.5:1 on a navy backdrop, as it's almost white. This is good for a hierarchy (the heading in pure white or cream, and supporting text in an off-white) while still maintaining minimum contrast ²⁶. If any hero section ever uses a **light background** (less common, but perhaps a hero could be a white or pale section), then by the same token the text should switch to a dark color. The new “automatic contrast” system in the CSS attempts to handle this by defining CSS variables for `--text-light-primary` vs `--text-dark-primary` and applying them based on background classes ²⁷ ²⁸. Going forward, designers should leverage that: for example, if a hero were placed on a white background, adding a class like `.bg-white` on its container would trigger the script to set all child headings to dark text automatically ²⁷ ²⁹. Overall, the heroes are now in good shape: **bold, high-contrast typography plus background overlays** ensure the page title and calls-to-action are immediately visible. These changes address the earlier issues noted (where hero text sometimes blended in) ³⁰. Just remember to maintain these standards on any new hero graphics.

Buttons and Call-to-Action Elements

Buttons are critical UI elements, and their text must be easily readable against their backgrounds. The site has a few button styles, primarily **Primary (dark blue)** and **Secondary (gold)**. After the recent unification, the **Primary buttons** use the Brand Blue/Navy background (`#23374f` or `#0066CC`) with white text ³¹ ³². This yields a very strong contrast (white on the dark blue is >5.5:1 even for the lighter brand-blue shade, and up to ~12:1 on the navy shade), so primary buttons meet WCAG with room to spare. The hover state darkens the blue further, which only improves contrast ³³. No changes needed there – **white text on dark blue** is an accessible combination and aligns with the brand's look.

For **Secondary buttons**, which currently use the Brand Gold/Cream as the background, we need to ensure the text color is dark enough. Right now, the code shows two approaches: in some places the gold-background buttons have navy text (for example, the “Agent Portal” button is `background-color: #e9c77b` with text `color: #0F1C2E`) ¹⁷, but in other instances gold buttons were implemented with the medium blue text (e.g. Tailwind's `.bg-amber-500 text-gray-900` or custom `.btn-secondary` using `#FFB81C` bg and `#0066CC` text) ³⁴. **Navy or black text is the correct choice on a gold background** – it gives about a 9:1 contrast ratio ¹⁷. In contrast, the blue text (`#0066CC`) on gold yields only ~3.2:1 **[26+]**, which *fails* the 4.5:1 requirement for normal text. Our recommendation is to standardize the

Secondary button style to use the darkest possible text. The team already demonstrated this with the Agent Portal button (gold with navy text) and noted it as an improvement ¹⁷. To propagate this fix, any other occurrence of gold buttons (e.g. perhaps the “Start a Transaction” hover state, or old `.btn-secondary` definitions) should be updated to use `color: #0F1C2E` (the Brand Navy) or at least `#1E293B` (a very dark gray) for the text. For example:

```
/* Ensure gold secondary buttons have dark text for contrast */
.btn-secondary {
  background-color: #FFB81C; /* brand gold */
  color: #0F1C2E !important; /* brand navy text */
}
.btn-secondary:hover {
  background-color: #F0D28C !important; /* slightly lighter gold on hover */
  color: #0F1C2E !important;
}
```

With this rule, the gold button’s label will always be dark and legible. The hover style above uses a lighter gold (picked in code as `#f0d28c` ³⁵) which still keeps the text contrast high. Alternatively, one could invert on hover (dark blue background with white text), but that might confuse brand identity – sticking to gold with dark text is fine as long as contrast is addressed.

Specific CTA examples: The “Start a Transaction” button on the home hero currently has a unique styling – default state is a white background with blue text and a gold border ³⁶, and on hover it inverts to gold background with blue text ³⁷. The default state (white with blue text) is excellent for visibility on the dark hero (blue text `#0066CC` on white is ~5.6:1, and the bold font makes it even clearer). The hover state, however, reintroduces the low-contrast blue-on-gold issue. Since this is a large button, the text might be slightly above 18px and *just* meet the 3:1 contrast needed for large text – but it’s borderline. It would be safer to treat this similar to the Agent Portal button: use navy text on the gold hover. We suggest adjusting that hover in CSS to:

```
a[href*="Start a Transaction"]:hover {
  background-color: #FFB81C !important;
  color: #0F1C2E !important; /* navy text instead of blue */
}
```

This small tweak preserves the gold-on-hover effect but makes the text much more readable (contrast jumps from ~3.2:1 to ~9:1). It also matches the visual style of the Agent Portal button, which already uses navy text on gold ¹⁷. Consistency here will reinforce both branding and accessibility.

Beyond colors, the buttons have gained other improvements: the audit notes the addition of **border rings and shadows on focus/hover** to make them stand out ³⁸. For instance, a focus style (`outline: 2px solid #3b82f6`) was added globally for keyboard navigation ³⁹. This is great for usability and doesn’t affect color contrast of text, but it does ensure the button itself is visible when focused. All button states – default, hover, active, disabled – should be tested under high contrast mode and with colorblind filters to ensure they remain discernible. Color-wise, as long as we stick to **dark text on light buttons and light text**

on dark buttons, we will satisfy WCAG. The primary/secondary scheme covers this, with the secondary needing the text color fix as described.

One more detail: in some places, the code uses Tailwind's amber/yellow utilities for buttons (e.g. `bg-amber-500 text-gray-900` for Agent Portal in the nav on mobile ¹⁶). Amber 500 is a slightly orange-tinted gold. It too has sufficient contrast with `text-gray-900` (almost black) for the text. There's no harm in using the Tailwind preset here, but to maintain **styling consistency** you might consider defining `brand-gold-500` as that same color so all instances (desktop and mobile) use the exact same shade. Consistency aside, the **accessibility goal is achieved** because in all these cases the text on the button is dark enough ¹⁷.

Page-Specific Contrast Issues and Fixes

Home Page

The Home page combines a hero section, feature highlights, and possibly some summary sections. We've already covered the hero: the home hero text is now very readable thanks to bold, shadowed cream-colored text on a darkened image ²⁰ ⁴⁰. One specific element to note is the tagline or subheading in the hero (if present). It appears that any `<p>` text in the hero is styled as a very light blue (`text-blue-50` equivalent) ⁴¹ ⁴². This is effectively a near-white and should be fine on the dark overlay (contrast ~4.8:1 against a dark navy background). It might have been adjusted because pure white for both heading and sub-text can be visually jarring; using a slightly softer light color for subtext differentiates it without losing much contrast. This approach is good, as all hero text remains above the 4.5:1 ratio requirement now.

Scrolling down, the Home page likely features "feature cards" or service highlights. In earlier iterations, those feature blocks had contrast problems – the audit documentation noted **text sometimes had the same color as its background** in those cards ⁴. For example, there may have been a case of a card with a light background image where the text was also light. The development team responded by enforcing specific classes (`feature-card`) to define text colors. Now, **feature cards on the home page use dark text on a light card**: the headings on feature cards are set to the brand blue (which is a medium-dark blue) and the body text to a dark gray ⁴³. Brand Blue (#0066CC) as a heading on a white card gives ~5.3:1 contrast, which meets AA for all text sizes. The body text in gray-700 (#374151) is even darker (~10:1 contrast on white) ⁴⁴. These choices fix any prior illegibility that may have occurred when an accent color was used for text. If those feature cards have any colored background sections, the "contrast-system.css" ensures that headings would flip to white if the background is dark, or to nearly black if the background is light ²⁷ ⁴⁵. In short, **all text in the feature/intro sections on Home is now explicitly styled for contrast**.

Another Home section is the **Statistics section** (if present, showing some numeric stats). Based on the CSS, this section has a very light background (a gray-to-white subtle gradient) ⁴⁶. The stat numbers are in a large, dark-blue font and the labels in gray-600 ⁴⁷. Both are dark on a nearly white background, which is good. Gray-600 text on #F9FAFB (gray-50) is about 5:1 – enough for the slightly larger label text (and nearly enough for normal text, which is acceptable for secondary info). The numbers themselves in brand blue are even larger text and have ~5.5:1 contrast on that light background, so they are clearly readable. No issues there; the design choice to put stats on a light card with dark text avoids the common mistake of low-contrast "number on a background circle" that some sites do. They also added a border and shadow on

those cards ⁴⁸, which doesn't affect text contrast but does help visually separate the card from the background (useful if the background gradient is subtle).

One minor call-out: the **scroll call-to-action** (perhaps an arrow or "scroll down" prompt on the hero). There's a class `.scroll-btn` that was given a white background and blue text ⁴⁹. This likely appears at the bottom of the hero prompting the user to scroll. It's now a white button with blue icon/text – on a dark hero background, that's highly visible. Originally, if that was transparent or differently colored, it may have been overlooked. The current fix (white circle with brand-blue icon) is a good contrast solution.

About Us Page

The About page hero likely follows the global pattern, using the shared `PageHeroWrapper`. Indeed, in code the hero content for About is placed on a `.glass-card-navy` overlay ⁵⁰. This is a dark translucent card (navy background with blur) designed for white text. All text within that card is explicitly forced to white or nearly-white for maximum contrast ²⁶. For example, the "About Debbie O'Brien" title on that card is rendered in white (with maybe a slight text-shadow) so it pops against the navy glass background ²⁶. The content of the card (some bullet points or icons) will likewise be in a light color. This addresses the earlier empty/unstyled card issue that was noted for About's hero ⁵¹ – now it's populated and styled correctly.

Moving down, the About page often includes sections like "Our Team" or "Mission & Values". According to the CSS fixes, the **Team section** uses a light gray background (`bg-gray-50`) with white cards for each member ⁵² ⁵³. Each team member card has dark text (name in bold, role in gray-600) ⁵⁴. This is a solid accessible design: dark text on white for the cards, and even the background behind is only a very light gray, so contrast is still fine (gray-600 role text on gray-50 background is roughly 5:1, acceptable for that small subtitle). If any team member names were styled in brand colors, the code suggests they are not – they stuck to neutral dark colors, which is best for consistency and contrast.

The **Mission section** is listed with a white background and paragraph text ⁵⁵. Black or near-black text on white yields maximum readability, so as long as that's what's used (the CSS shows `.mission-content` just uses default text styling), it's perfect. Sometimes designers try to use a faint gray for large blocks of text; here I'd caution that if any body text is lighter than #757575, it might fall below 4.5:1 on white. However, since the global base text is set to #333333 by default ⁵⁶, we have a good dark base color for all paragraph content. That equates to about 13:1 contrast on white – excellent. So the mission text and any descriptions on About are in line with WCAG AA (even AAA in many cases, since AAA requires 7:1 for body text and we're above that).

The **Professional Journey / Timeline** component on the About page has a unique design with colored icons and perhaps timeline cards. Each timeline event in the code has a pair of colors: a darker color for an icon background and a lighter variant for hover, with the icon itself set to white ⁵⁷ ⁵⁸. For example, one event uses a purple background circle (#7a2046) and the icon (graduation cap) is white ⁵⁷. This is great: white icon on a colored circle is high contrast. When you hover, the circle might expand (they have an overlay effect) and the icon might change color – according to the code, on hover they actually **fade the icon's white color to the dark color** (group-hover text becomes `var(--bg-color)` ⁵⁹). But simultaneously, a larger overlay circle expands behind in the original color. The effect is more visual flair, but importantly, at the end of it the icon might not remain white. We should double-check that the icon doesn't become low-contrast when hovered. The CSS suggests the icon, which is white initially in a smaller circle, becomes colored when the bigger colored overlay expands (which might leave the icon now on a white background of the small

circle). They did ensure the small circle itself is white (`group-hover:bg-white`)⁶⁰, so when the icon turns from white to, say, purple, it is then on a white circle – purple on white likely has enough contrast if the purple is dark. Indeed, the colors for icons (like #7a2046) are all quite rich/dark, and on a white circle they would be fine (~6.5:1 for that purple on white [58†]). So the timeline interactivity still respects contrast – an example of thoughtful design. The text for each timeline entry (years, titles, descriptions) presumably is just normal black/gray text on the page’s default background (likely white or a very light section background). There’s no sign of the timeline text being a problem in the CSS, and given the rest of the page’s attention to color, it’s safe to assume they are using dark text for those details as well.

In summary, the About page now uses **light section backgrounds with dark text** for content, and **dark overlay cards with light text** for the hero – both approaches meet or exceed accessibility contrast guidelines. No remaining contrast issues were observed on this page after the fixes.

Services Page

The Services page appears to follow a similar pattern. The hero would again be a dark glass card with white text (pageType “services” likely uses the same `glass-card-navy` in the hero)⁵⁰. That ensures the page title “Services” is immediately visible. For the content, the Services page CSS defines **service cards** as white (`bg-white`) with dark text for each service⁶¹⁶². Each service title is an `text-xl font-bold text-gray-900` (nearly black)⁶³, and the description is `text-gray-700`⁶². This mirrors the approach on the home feature cards – using neutral dark colors on a clean background. Users will have no trouble reading these. If there are any icons or decorative elements for each service, the CSS snippet shows a `.feature-icon` class for service features that uses `text-brand-blue` for the icon color⁶⁴. That means any bullets or checkmarks in a list of service features are colored in the medium blue. On a white background, those icons have sufficient contrast to be seen (brand blue #0066CC on white is ~5.6:1, so even an icon glyph is clearly visible). In fact, this adds a bit of color to the page without sacrificing readability. The accompanying feature text is in gray-700⁶⁵, which as noted is a safe dark gray.

One check: sometimes pages like Services might include colored section bands (e.g. a call-out banner or testimonial with a different background). If present, we should ensure the text in those is handled. The provided CSS hints at a generic rule: any section with IDs like “work-together” or “how-we-work” was forced to `bg-gray-100 !important`⁶⁶. This suggests that perhaps on the Work With Me page (or possibly a section within Services), there was a section with an off-brand background that caused contrast concerns. By overriding it to a light gray background, the developers ensured that **all text in that section could be dark** and readable. This is a wise fallback – rather than fighting with a background image or gradient that made text hard to read, they simply use a clean, light backdrop. We see a consistent philosophy now: whenever in doubt, **use light backgrounds with dark text or dark backgrounds with light text**. The Services page adheres to this, and no outstanding contrast issues are evident after the fixes (the UI improvements doc specifically mentions the Services page had gradient overlays removed in favor of clean cards and improved contrast in the form and features)⁶⁷.

“Work With Me” (Contact) Page

The “Work With Me” page (presumably a contact or lead capture page) had some of the more problematic contrast combinations originally – likely due to decorative gradients or background images in the contact

section. The overhaul has significantly improved it. Now, the contact page is split into a **Contact Info card** and a **Contact Form**, each using a glassmorphic card for contrast and aesthetics ⁶⁸ ⁶⁹ .

- The **Contact Info** card (with address/phone/email) is implemented with the `.contact-info` class, which applies a `glass-card-navy` style background ⁶⁸ . This is a dark, semi-transparent navy panel. All text and icons inside are accordingly styled light-on-dark: the `.contact-text` class makes text white ⁷⁰ , and the icons (like a phone or email icon) are given `text-blue-200` (a very light blue) ⁷¹ . White text on the navy glass is obviously high contrast (white on #0F1C2E is about 17:1), and even the light blue icons on navy are around 12:1 contrast **[59†]** . This means phone numbers, email addresses, and labels in that card will be easy to read. Previously, if these were plain gold text on some patterned background, it would have been an issue – but that’s no longer the case ⁷² . The use of a **dark translucent card with white text** also gives a nice visual consistency with other dark sections while solving the readability concerns.
- The **Contact Form** itself uses a `.contact-form` class on a `glass-card-frost` background ⁶⁹ . “Frost” here means a light, semi-transparent white background (the CSS in contrast-system defined `glass-card-white` / `frost` with an 80% white bg) ⁷³ ⁷⁴ . On this kind of background, the text inside should be dark. Indeed, the form labels are `text-gray-700` ⁷⁵ and the input text would naturally be dark (inputs by default render black text). The inputs have a light border (gray-300) which is sufficient to outline them on a white card ⁷⁶ . One possible improvement: the placeholder text in inputs, if any, might be a lighter gray. We should ensure it’s not too light – typically browser default placeholders are a gray around #767676 which is about 4.8:1 on white, so that’s borderline but acceptable for non-essential text. It might be worth explicitly styling placeholders to `#6B7280` (Tailwind gray-500) or darker to ensure they’re visible. However, since placeholders are short-lived hints, this is a minor point. The main content (what the user types) will be dark on white – perfectly readable. The form’s submit button is styled as a Brand Blue button with white text (`bg-brand-blue text-white`) ⁷⁷ . As noted, that’s an ideal combo for contrast. On hover, it becomes a slightly darker blue which is fine.

Additionally, any supporting text or disclaimers on the contact page likely follow the same pattern: if on the navy card, they’re white; if on the frosted card, they’re dark. The design team also removed any busy background from the form section – earlier notes indicated a gradient overlay on the “Work With Me” page that didn’t match the site aesthetic ⁶⁷ . Now, by using these two contrasting cards on a presumably simple background, they have achieved both clarity and a modern look.

One thing to check is how the **form section background** interacts with the rest. The CSS forced sections with “work-together” or similar IDs to gray-100 (a very light gray) ⁶⁶ , which suggests the overall page background behind the glass cards is a light gray. That’s good because the frosted white card will still stand out slightly, and the navy card obviously stands out. Both cards are easily distinguishable, and the light gray page background poses no problem for the black text outside the cards (if any). If there are any decorative separators or images, those have been toned down so as not to interfere with text.

In summary, the Contact/Work With Me page now complies with contrast best practices: **dark-on-light for form fields, light-on-dark for info card**, no low-contrast combinations left. Users should be able to fill out the form and read contact info without straining.

Footer

The footer is a common area where contrast issues can creep in (designers often use lighter text for a subtle look, which can backfire on light backgrounds). In this site's case, the footer appears to have a **light background** (likely white or very pale gray) after the revisions, with all footer text in a relatively dark color. The audit fix file explicitly set footer paragraphs and normal links to `color: #374151` (Tailwind gray-700) ⁷⁸. That's a medium-dark gray which on white yields about 10:1 contrast **[21†]**, more than sufficient. Previously, if any footer text was using the gold or a light gray, that would have been illegible on a light background. Now, whether it's the address, copyright, or list of quick links, they're using at least gray-700 (or darker) which passes AA easily.

The **footer link hover** style, however, still uses the Brand Gold as an accent ⁷⁹. So a link in the footer (say "Privacy Policy") will be dark gray initially and turn gold on hover. As discussed earlier, gold on white is not AA-compliant. This is a minor risk because a user only sees that color momentarily on hover. But to be thorough, we recommend adjusting the footer link hover to either use a darker accent or an underline. For example, footer links could simply remain gray-700 but underline on hover (this would actually be more consistent with typical accessibility guidelines, which suggest not relying solely on color changes). If a color change is desired, perhaps use the darker gold (`#8C7647`) on hover – this color would yield ~4.4:1 on white, which is just shy of 4.5:1 but much closer than the bright gold. Even better, something like the field-drab brown (`#75592d`, which is within the brand's earthy palette) gives over 6:1 contrast on white **[58†]**. Selecting an accessible hover color does *not* mean abandoning brand standards; it can be framed as using the "gold 700" or "gold 800" tone from the extended palette. Implementing that in CSS might look like:

```
footer a:hover {  
  color: #8C7647 !important; /* dark gold */  
}
```

This small tweak would ensure even on hover the text meets contrast. The rest of the footer elements (section headers like "Quick Links", icons if any for social media, etc.) should also be checked. If, for instance, social media icons are present in the footer and are using Brand Gold fill on a white background, that's the same issue – they might be hard to see. Ideally those icons should be in gray or navy, and only change color on hover if at all. The general rule: **in the light-colored footer, keep all text/icons at least dark gray** (and use the gold sparingly). Given the changes we see, the team has already moved in this direction. The fixes summary explicitly mentions footer contrast was fixed ⁸⁰, so likely they caught things like making sure even small text (copyright notices, etc.) are sufficiently dark (often `#6B7280` or darker on white). We've verified that with gray-700 and up being used, which is good.

Contrast Ratio Summary and WCAG Compliance

Below is a brief table of some key color combinations from the site, their contrast ratios, and whether they meet WCAG AA guidelines (for normal text size):

Text Color vs Background	Contrast Ratio	AA Compliant?	Recommendation/Status
Brand Gold (#FFB81C) on White	~1.73:1 【28†】	No (Fail)	Avoid this combination for text. Use dark text on white instead ² .
Brand Blue (#0066CC) on White	~5.6:1	Yes	Good for headings, links, etc. (used for links and icons) ⁸¹ .
Brand Blue (#0066CC) on Brand Gold	~3.2:1 【23†】	No (Fail)	Use Brand Navy text on Brand Gold background for 9+:1 contrast ¹⁷ .
Brand Navy (#0F1C2E) on Brand Gold	~9.9:1 【27†】	Yes	Excellent contrast (used in gold buttons now) ¹⁷ .
White (#FFFFFF) on Brand Blue	~5.57:1 【24†】	Yes	Good contrast for nav text, buttons, etc. (in use) ³¹ .
White on Brand Navy / Black	>12:1	Yes (AAA)	Ideal for hero text, nav on dark, etc. ²⁶ .
Gray-700 (#374151) on White	~10.3:1 【21†】	Yes (AAA)	Used for body text and footer text ⁷⁸ .
Gray-500 (#6B7280) on Gray-50	~4.8:1	~Yes (borderline)	Acceptable for larger or secondary text (e.g. form placeholders). If smaller text, use Gray-600 or darker.
Light Blue (#EFF6FF) on Navy	~4.5:1	Yes	Used for secondary hero text ⁴² ; just meets AA for normal text.
Brand Cream (#E9C77B) on Black	~13:1	Yes (AAA)	Very high contrast; used for hero heading text with shadow ²⁰ .

(Ratios calculated per WCAG 2.1 luminosity formula. AA requires ≥ 4.5 for normal text, ≥ 3.0 for large text. AAA requires ≥ 7.0 for normal text.)

As shown above, the problematic combos (gold-on-light, blue-on-gold) have been corrected in the new styling by either changing the text color or the background. The site now largely uses **only high-contrast pairings**. One should remain cautious of a few edge cases – e.g. gold link hover on white (1.7:1, fails) is an outlier that can be adjusted as discussed. But all **core content** (headers, body text, buttons, nav, etc.) now meets at least AA contrast, and in many cases AAA. The developers explicitly mention that they **“Ensured all color combinations meet WCAG AA standards”** going forward ⁷ , which is an excellent guiding principle.

Recommendations & Next Steps

The recent fixes have vastly improved the site's accessibility by addressing previous contrast issues ³⁰ ³⁸ . To maintain this high standard, here are a few recommendations:

- **Adopt the Contrast Variable System Globally:** The repository introduced a centralized system of CSS custom properties for colors (e.g. `--text-dark-primary`, `--text-light-primary`, etc.) to automatically handle contrast based on background context ⁸² ²⁷ . It's important to continue using those classes and patterns consistently. For example, whenever a new section or component is added, assign it a semantic background class (like `.bg-gray-800` or `.bg-white`) so that the global CSS will apply appropriate text colors. This reduces the chance of human error where someone might pick an unsafe color. The "contrast-system.css" and related files provide a blueprint for this automated approach ²⁷ ²⁹ .
- **Beware of New Accent Usage:** If design tweaks reintroduce brand accent colors (gold/cream or red) in text, always cross-check their contrast. As a rule, **light gold or red text should go on dark backgrounds only**. (The brand red `#EF4444` on white is actually OK at ~5.4:1 [11†] , but gold is not.) If you ever want a lighter text on a light background for aesthetic reasons, strongly consider increasing font weight/size and adding a text-shadow or outline – but even then, meeting 4.5:1 is required for normal text. It's easier to just use a darker variant of the color. The documentation suggests using the new `contrast-variables.css` "as a source of truth for all color values" ⁸³ , which implies designers should stick to those pre-vetted colors instead of arbitrary ones.
- **Finalize Link Hover Styles:** As noted, adjust the remaining low-contrast link states (like footer and perhaps main nav hover) to an accessible color. This could be as simple as using the existing dark navy for hover underlines, or using the brand gold **as a background highlight** behind the link text instead of as the text color. For instance, a hover style where the link gets a gold **underline** or a gold highlight box with dark text could convey the brand feel without sacrificing contrast. Any of these changes will keep you safely within guidelines.
- **Maintain Contrast in Imagery:** Continue using overlays on images and consider the placement of text. The team's addition of gradient overlays on heroes ⁴⁰ and even applying slight background colors on transparent sections ⁸⁴ has made a big difference. Whenever new hero images or section background images are added, ensure a similar 40–50% dark overlay or blur is applied if text will be over them. It's an easy step to forget, but the current code provides a pattern (the `[data-hero-component="true"] : :before` rule injecting a gradient) that you can reuse ⁴⁰ .
- **Test Responsively:** On small screens, sometimes layouts change and background colors might overlap differently. The mobile menu was a good catch – originally perhaps it lacked a background, but now it has one ¹² . Similarly, test forms and cards on mobile to make sure, for example, a card that was dark doesn't accidentally end up on a differently colored background when stacked. It appears the site handles this by using consistent container backgrounds (light gray page background for the contact section, etc.), so it should be fine. Just be mindful that **any dynamic or conditionally rendered content keeps the contrast rules**. An example might be a popup modal or tooltip – if any appear, they should follow the same dark-on-light or light-on-dark scheme.

- **Periodic Accessibility Audits:** As a future recommendation (echoing the project’s own notes ⁸³), integrate an accessibility linter or use browser tools to periodically check contrast. Tools like Lighthouse, axe, or WCAG Contrast Checker can flag if someone accidentally introduces “#e9c767” instead of #e9c77b, etc., which might not have been accounted for. Since the team has done a great job fixing issues (the “Text Visibility Fixes” and “Contrast and Usability Improvements” docs enumerate these fixes ⁸⁵ ¹⁰), ensuring ongoing compliance will protect that effort. Even adding automated tests for a few critical pages’ color contrast could be worthwhile.

In conclusion, **text readability on the site has improved tremendously** after addressing the CSS styling issues. We identified and corrected areas where insufficient contrast was a problem: navigation links, hero text, glass card content, button labels, and footer links have all been adjusted to meet accessibility standards ⁸⁶ ⁸⁷. By implementing the above recommendations – especially standardizing hover states and sticking to the contrast-first style definitions – the site will not only pass WCAG guidelines but also provide a consistently clear visual experience to all users. The brand’s colors are still in use, but now in a way that **enhances** rather than undermines content visibility. Going forward, any new design elements should follow the established pattern of **“dark text on light backgrounds, light text on dark backgrounds”**, using the provided palette and contrast variables. This mantra will ensure that the site remains readable and professional-looking across all pages and devices. ⁷ ⁸⁴

1 **tailwind.config.cjs**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/tailwind.config.cjs>

2 4 51 67 **VISUAL_FIXES.md**

https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/VISUAL_FIXES.md

3 7 24 80 83 **TEXT_VISIBILITY_FIXES.md**

https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/TEXT_VISIBILITY_FIXES.md

5 **accent-colors.css**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/styles/accent-colors.css>

6 17 19 20 21 22 23 35 36 37 40 **hero-nav-contrast.css**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/styles/hero-nav-contrast.css>

8 31 33 **index.ts**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/theme/index.ts>

9 11 25 26 32 34 41 42 43 44 49 78 79 **text-contrast-fixes.css**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/styles/text-contrast-fixes.css>

10 13 30 38 72 84 85 86 87 **CONTRAST_AND_USABILITY_IMPROVEMENTS.md**

https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/CONTRAST_AND_USABILITY_IMPROVEMENTS.md

12 14 15 16 18 **Header.tsx**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/components/Header.tsx>

27 28 29 45 73 74 82 **contrast-system.css**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/styles/contrast-system.css>

39 **globals.css**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/app/globals.css>

46 47 48 52 53 54 55 61 62 63 64 65 66 68 69 70 71 75 76 77 81 **page-specific.css**

https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/styles_new/page-specific.css

50 **PageHeroWrapper.tsx**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/components/PageHeroWrapper.tsx>

56 **globalStyles.tsx**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/theme/globalStyles.tsx>

57 58 59 60 **Timeline.tsx**

<https://github.com/haydnphilipdesign/pretty-much/blob/d453b3d301eedc97726cf433c7e07a22b58a6e09/src/components/Timeline.tsx>