

WEB STAR STUDIO

MANUAL

MÉTODO DE DESENVOLVIMENTO DE SITES

MANUAL 3
Construção Visual

VERSÃO

v1.1 FINAL

DATA

11/09/2026

STATUS

FINAL — APROVADO

MANUAL

MANUAL 3

Construção Visual

VERSÃO	v1.1 FINAL
DATA	11/09/2026
STATUS	FINAL — APROVADO
ESCOPO DOCUMENTAL	primeira documentação ampliada — P0 + P1 + P2 relevante
PROJETO	WSS — Método de Desenvolvimento de Sites
FORMALIZAÇÃO	Pacote Corretivo Pós-Entrega v1.1 aprovado pela Coordenação em 11/09/2026; versão canônica vigente, com v1.0 FINAL preservado como baseline histórico.

NAVEGAÇÃO DO DOCUMENTO

Sumário

Estrutura do documento canônico v1.1 FINAL.

1. Objetivo	7
2. Escopo e fronteira	7
3. Posição deste Manual no fluxo	9
PARTE I — REFERÊNCIAS VISUAIS	10
4. Fonte principal de referências	11
5. Quem participa da escolha	11
6. Referências não precisam ser do mesmo setor	11
7. Fontes auxiliares	12
PARTE II — ANÁLISE DE REFERÊNCIA	13
8. A análise não é apenas “olhar o site”	14
9. O que tende a ser observado	14
10. DevTools não é etapa obrigatória de toda referência	14
11. Screenshots como evidência visual	15
PARTE III — TRADUÇÃO DA REFERÊNCIA PARA O PROJETO	16
12. Preservar essência, não fazer cópia cega	17
13. O que pode ser reaproveitado conceitualmente	17
14. Combinação de referências	17
15. Prompting visual que tende a funcionar	18
16. Anti-pattern de prompting visual	18
PARTE IV — HOME COMO ÂNCORA VISUAL	19
17. A Home é o principal ponto de partida	20
18. Primeira versão	20
19. O que é refinado na Home	20
20. Fidelidade e adequação	21
21. Quando avançar para páginas internas	21
PARTE V — PÁGINAS INTERNAS	22

22. Sitemap como guia de cobertura	23
23. Ordem usual	23
24. Paralelismo	23
25. Primeira geração de página interna	23
26. Elementos recorrentes	24
27. Elementos que podem variar	24
28. Peso visual diferente entre páginas	24
PARTE VI — CRITÉRIO PRÁTICO DE PÁGINA PRONTA	25
29. Três verificações principais	26
30. Não confundir "pronta visualmente" com "homologada"	26
31. AUSÊNCIA ATUAL DE PADRONIZAÇÃO	26
PARTE VII — COMPONENTIZAÇÃO E REUSO	27
32. Não existe regra humana formal de componentização	28
33. Reuso declarado	28
34. Reuso técnico confirmado	28
35. Formulação AS-IS segura	28
36. Componentização não deve apagar particularidade	29
PARTE VIII — DESIGN SYSTEM E CONSISTÊNCIA VISUAL	30
37. design.md como âncora operacional	31
38. O código também materializa consistência	31
39. Consistência híbrida	31
40. Não existe Design System único universal	32
41. O que a consistência busca preservar	32
PARTE IX — ASSETS E IMAGENS	33
42. Entrada de imagens do cliente	34
43. Catalogação com Codex	34
44. Quando falta material	34
45. Oportunidade futura sobre origem/autorização	34
46. Compressão, formato e naming	35
47. Imagens e performance	35
48. Imagens e responsividade	35
PARTE X — CONTEÚDO	36

49. IA participa fortemente da redação	37
50. Conteúdo pode chegar parcialmente pronto	37
51. Fatos sensíveis não devem ser inventados	37
52. Informação faltante	38
53. Validação factual	38
54. AUSÊNCIA ATUAL DE PADRONIZAÇÃO	38
PARTE XI — EFEITOS E ANIMAÇÕES	39
55. Efeitos fazem parte da linguagem visual	40
56. Tecnologias e técnicas observadas	40
57. Daniel P como evidência de sofisticação	41
58. Como investigar um efeito	41
59. Quando adaptar, simplificar, substituir ou remover um efeito	41
60. Problemas reais com efeitos	42
PARTE XII — RESPONSABILIDADE	43
61. Responsividade entra durante a construção	44
62. Tamanhos/dispositivos relevantes	44
63. Desktop e mobile podem exigir soluções diferentes	44
64. Preservar breakpoint já validado	44
65. AUSÊNCIA ATUAL DE PADRONIZAÇÃO	45
PARTE XIII — PERFORMANCE	46
66. Performance é tratada de forma pragmática	47
67. Primeiro foco quando há peso/lentidão	47
68. Sofisticação com contenção técnica	47
69. Gate de performance atual	48
70. Performance × referência	48
PARTE XIV — ACESSIBILIDADE AS-IS	49
71. Existe atenção consciente	50
72. Problemas reais detectados	50
73. Acessibilidade não é auditoria formal	50
74. Limite deste Manual	50
PARTE XV — RELAÇÃO COM NORMA TÉCNICA	52
75. NORMA TÉCNICA VIGENTE	53

76. Não reescrever o AS-IS com a norma	53
PARTE XVI — CASOS REAIS	55
77. Guilherme Mendonça	56
78. Daniel P	56
79. O que não generalizar a partir dos casos	56
PARTE XVII — AUSÊNCIAS ATUAIS DE PADRONIZAÇÃO	57
80. Lista consolidada deste Manual	58
PARTE XVIII — OPORTUNIDADES FUTURAS — FORA DO ESCOPO	59
81. Oportunidades já identificadas	60
PARTE XIX — RELAÇÃO COM OS OUTROS MANUAIS	61
82. Relação com o Manual 2	62
83. Relação com o Manual 4	62
84. Relação com o Manual 5	62
85. Relação com o Manual 6	62
86. Relação com o Manual 8	63
PARTE XX — SOPS DERIVADOS	64
87. SOP-003 — Construir e Refinar a Home a partir das Referências	65
88. SOP-004 — Construir Páginas Internas Mantendo Consistência Visual	65
89. Efeitos não viram SOP próprio nesta primeira versão	65
PARTE XXI — CRITÉRIO DE CONCLUSÃO	67
90. Critério de conclusão deste Manual	68
91. Fontes principais desta versão	68

1. Objetivo

Este Manual documenta como a WSS transforma referências, contexto, Sitemap e conteúdo em uma experiência visual implementada e coerente.

Seu foco é a construção visual do site público:

```
referências
→ análise
→ tradução para o cliente
→ Home
→ páginas internas
→ reuso / consistência
→ assets / conteúdo
→ efeitos
→ responsividade
→ performance
→ atenção à acessibilidade
```

Ao final, o leitor deve compreender:

- como as referências são escolhidas e utilizadas;
- como a Home estabelece a linguagem visual;
- como páginas internas derivam dessa linguagem;
- como o Dev trabalha fidelidade e adaptação;
- como reuso e componentização aparecem no AS-IS;
- como `design.md` se relaciona com tokens/componentes no código;
- como imagens e conteúdo entram na composição;
- como efeitos/animações são escolhidos ou descartados;
- como desktop, notebook e mobile influenciam a construção;
- como performance é tratada de forma pragmática;
- qual é a atenção atual à acessibilidade;
- o que ainda não possui padronização formal.

Este Manual registra o **AS-IS**. Ele não cria um Design System formal universal, não cria política nova de performance, não institui auditoria WCAG e não transforma um projeto visualmente sofisticado em padrão obrigatório para todos os clientes.

2. Escopo e fronteira

2.1 O que entra neste Manual

Este Manual cobre:

1. pesquisa de referências;
2. escolha da referência principal e referências secundárias;

3. análise visual;
4. screenshots e recortes;
5. investigação de comportamento/efeito;
6. tradução de referência para o cliente;
7. Home;
8. páginas internas;
9. ordem prática das páginas;
10. reuso;
11. componentização;
12. Design System híbrido;
13. `design.md`;
14. tokens/variáveis/componentes;
15. assets e imagens;
16. conteúdo textual;
17. efeitos e animações;
18. responsividade;
19. performance;
20. acessibilidade AS-IS;
21. critérios práticos de “pronto” na construção;
22. casos reais úteis para esta camada.

2.2 O que não entra neste Manual

Ficam fora:

- preparação inicial do contexto e prompt master em profundidade;
- Git/GitHub;
- clone, pull, commit e push;
- operação detalhada Codex/Lovable/Claude;
- configuração do Admin;
- Chatbot/RAG;
- QA final e regressão;
- debugging;
- SEO/AEO/GEO;
- segurança;
- publicação.

O M3 é proprietário de **como construir e adaptar visualmente**.

O M6 é proprietário de **como validar tecnicamente o que foi construído**.

3. Posição deste Manual no fluxo

Este Manual começa quando:

- o projeto já foi contextualizado;
- as referências já foram encaminhadas/validadas;
- existe Sitemap;
- existe direção do cliente;
- o projeto já possui primeira base no Lovable/local.

Seu recorte principal é:

```
primeira base do institucional  
→ comparar com referência  
→ refinar Home  
→ consolidar linguagem visual  
→ desenvolver páginas internas  
→ tratar conteúdo/assets/reuso  
→ adaptar efeitos  
→ ajustar responsividade/performance  
→ considerar página visualmente pronta
```

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE I — REFERÊNCIAS VISUAIS

4. Fonte principal de referências

A principal fonte declarada é:

Awwwards

A equipe busca referências modernas com qualidade de:

- UI;
- UX;
- design;
- performance percebida;
- composição;
- navegação;
- efeitos.

Normalmente são selecionadas **3 a 5 referências**.

5. Quem participa da escolha

Participam da busca de referências:

- Felipe;
- Dev;
- Analice;
- Felipe/CEO.

No kickoff, o cliente pode validar:

- referência principal;
- características específicas de referências secundárias.

Quando o desenvolvimento começa, a direção já tende a estar registrada no Contexto de Desenvolvimento Consolidado.

6. Referências não precisam ser do mesmo setor

A referência pode vir de outro segmento.

O critério não é apenas semelhança setorial.

A equipe pode aproveitar:

- estrutura;
- ritmo;

- navegação;
- sofisticação;
- uso de tipografia;
- Hero;
- animações;
- composição de seções;
- experiência.

A adaptação ao cliente é feita depois.

7. Fontes auxiliares

Outras fontes observadas:

- Navbar Gallery;
- React Bits;
- 21st.dev;
- Framer;
- CodePen;
- X/Twitter;
- Instagram.

Elas aparecem principalmente para:

- navbars;
- componentes;
- animações;
- efeitos;
- microinterações;
- ideias visuais.

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não existe catálogo obrigatório de referências que toda equipe precise consultar.

Awwwards é a principal fonte declarada, e as demais são auxiliares.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE II — ANÁLISE DE REFERÊNCIA

8. A análise não é apenas “olhar o site”

O Dev combina diferentes formas de leitura:

- navegação visual;
 - observação de layout;
 - observação de comportamento;
 - screenshots;
 - comparação de proporções;
 - investigação com Codex quando o comportamento é complexo.
-

9. O que tende a ser observado

Na prática, a leitura pode considerar:

- Hero;
- proporções;
- hierarquia;
- tipografia;
- paleta;
- imagem;
- espaçamento;
- largura de conteúdo;
- cards;
- CTAs;
- transições;
- scroll;
- componentes;
- comportamento responsivo;
- ritmo visual entre seções.

Não existe checklist universal formal para essa análise.

10. DevTools não é etapa obrigatória de toda referência

Não há inspeção técnica fixa via DevTools para qualquer referência.

Quando necessário, o Dev ou Codex pode aprofundar:

- comportamento;
- animação;

- efeito;
- interação;
- estrutura aparente.

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

A profundidade da inspeção varia com a complexidade do que se quer reproduzir.

11. Screenshots como evidência visual

Screenshots são usados para reduzir ambiguidade.

Eles ajudam a indicar:

- qual estado;
- qual seção;
- qual enquadramento;
- qual detalhe;
- qual breakpoint.

Essa prática é especialmente importante no prompting visual.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE III — TRADUÇÃO DA REFERÊNCIA PARA O PROJETO

12. Preservar essência, não fazer cópia cega

A prática observada é buscar fidelidade visual, mas adaptar quando necessário.

A referência pode ser modificada quando:

- entra em conflito com o briefing;
- não combina com o cliente;
- prejudica usabilidade;
- não funciona bem em mobile;
- exigiria cópia literal inadequada;
- existe necessidade funcional diferente.

Formulação AS-IS

preservar a essência visual e comportamental útil, adaptando-a ao cliente e ao contexto real do projeto.

13. O que pode ser reaproveitado conceitualmente

Pode-se aproveitar, por exemplo:

- composição;
- ritmo;
- proporções;
- modelo de Hero;
- comportamento de seção;
- hierarquia;
- interação;
- transição;
- direção tipográfica;
- lógica de cards.

Isso não implica duplicação literal de conteúdo, identidade ou marca.

14. Combinação de referências

Um projeto pode ter:

- uma referência principal;
- características selecionadas de referências secundárias.

O resultado pode combinar elementos.

Não há exigência de reproduzir um único site de referência de ponta a ponta.

15. Prompting visual que tende a funcionar

A prática corrente favorece:

1. escolher uma seção;
2. fornecer screenshot;
3. fornecer link;
4. pedir estudo da referência;
5. limitar escopo;
6. buscar fidelidade;
7. adaptar depois ao cliente.

Esse processo reduz o grau de interpretação livre da IA.

16. Anti-pattern de prompting visual

Abordagem considerada fraca:

enviar apenas o link do site e pedir que a IA seja “extremamente fiel”.

O problema é a amplitude excessiva.

A IA pode:

- escolher detalhes errados;
- simplificar a composição;
- interpretar de forma genérica;
- alterar áreas não pedidas.

O catálogo formal de anti-patterns pertence ao Manual 8.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE IV — HOME COMO ÂNCORA VISUAL

17. A Home é o principal ponto de partida

A Home funciona como primeira grande materialização da direção visual.

Ela recebe mais refinamento inicial porque estabelece:

- identidade;
 - ritmo;
 - linguagem;
 - proporções;
 - atmosfera;
 - padrões de seções;
 - comportamento;
 - referência para as páginas internas.
-

18. Primeira versão

A primeira versão normalmente nasce no Lovable a partir do prompt master.

Depois, o Dev compara com a referência e inicia refinamento.

Quase nunca a primeira geração é considerada suficiente em fidelidade.

19. O que é refinado na Home

O Dev trabalha principalmente:

- estrutura;
- proporções;
- tipografia;
- imagens;
- espaçamentos;
- alinhamentos;
- comportamento;
- efeitos;
- responsividade.

O processo é iterativo.

20. Fidelidade e adequação

A Home precisa equilibrar:

- referência visual;
- briefing;
- identidade do cliente;
- conteúdo real;
- conversão/CTA;
- usabilidade;
- mobile.

Quando esses elementos entram em conflito, não há algoritmo formal de decisão.

O julgamento permanece humano.

21. Quando avançar para páginas internas

Quando a Home já estabelece bem a linguagem visual, o Dev avança.

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não existe score objetivo universal que diga:

“Home pronta = X pontos”.

A decisão é predominantemente visual e funcional.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE V — PÁGINAS INTERNAS

22. Sitemap como guia de cobertura

O Sitemap continua sendo referência para saber:

- quais páginas precisam existir;
- qual estrutura pública deve ser coberta;
- quais páginas são prioritárias.

Ele não determina sozinho o grau de sofisticação visual de cada página.

23. Ordem usual

Prática predominante observada:



```
graph LR; Home[Home] --> Contato[Contato]; Home --> Blog[Blog];
```

Contato e Blog costumam vir logo após a Home por serem, em geral, páginas mais simples e rápidas de estabilizar. Depois, as demais páginas avançam conforme:

- complexidade;
- referência;
- ideia visual;
- importância estratégica.

Essa sequência não é mandatória e pode mudar por prioridade estratégica, conteúdo, referência ou dependência técnica.

24. Paralelismo

O trabalho pode ser paralelo.

Se surge uma boa ideia para uma seção de outra página, o Dev pode trabalhá-la antes de terminar a página atual.

Isso é compatível com a natureza visual do processo.

25. Primeira geração de página interna

A IA pode receber instrução semelhante a:

com base na Home e no Sitemap, crie a página X.

A primeira versão é avaliada pelo Dev e normalmente refinada.

26. Elementos recorrentes

Podem ser reutilizados:

- Footer;
 - CTA final;
 - FAQ;
 - identidade geral;
 - linguagem;
 - padrões visuais.
-

27. Elementos que podem variar

Podem mudar conforme a página:

- Hero;
- depoimentos;
- estrutura de seções;
- efeitos;
- grau de sofisticação;
- composição.

A consistência não significa que todas as páginas precisam ter o mesmo layout.

28. Peso visual diferente entre páginas

Prática observada:

- Sobre tende a receber bastante atenção;
- página geral de Especialidades também;
- páginas individuais podem ser mais diretas;
- página prioritária para o negócio pode receber refinamento adicional.

Isso é um critério prático, não regra formal universal.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE VI — CRITÉRIO PRÁTICO DE PÁGINA PRONTA

29. Três verificações principais

Na fase de construção, o Dev considera principalmente:

1. todas as seções previstas no Sitemap estão presentes;
 2. a qualidade visual está satisfatória e não parece simples demais;
 3. botões e ações principais funcionam.
-

30. Não confundir “pronta visualmente” com “homologada”

Esses critérios não substituem:

- QA;
- testes completos;
- regressão;
- SEO;
- segurança;
- produção.

Essas etapas pertencem ao Manual 6 e Manual 7.

31. AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não existe checklist formal detalhado de “página pronta” para a construção visual.

O método atual preserva julgamento humano.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE VII — COMPONENTIZAÇÃO E REUSO

32. Não existe regra humana formal de componentização

O Dev não declarou uma disciplina formal do tipo:

“se um elemento aparece duas vezes, deve virar componente”.

A prática é pragmática.

33. Reuso declarado

O Dev tende a:

- copiar/adaptar elementos entre páginas;
- preservar padrão;
- alterar localmente quando a página precisa de solução própria;
- evitar repetir em excesso efeitos muito elaborados.

34. Reuso técnico confirmado

A VAL-001 confirmou reuso real no código do Daniel P.

Exemplos:

- `FaqSection` compartilhado;
- `PillarCards` parametrizado;
- `SpecialtyPageLayout` orientado por dados;
- componentes específicos quando a solução é local.

Isso demonstra que o AS-IS não é apenas copiar e colar visualmente.

35. Formulação AS-IS segura

Reuso pragmático e híbrido: componentes compartilhados quando há ganho claro, layouts parametrizados quando páginas possuem estrutura semelhante e componentes locais quando a solução é específica.

36. Componentização não deve apagar particularidade

O fato de existir um componente compartilhado não significa que toda página deva ser forçada ao mesmo layout.

Quando a página exige solução própria, o projeto pode manter componente específico.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE VII — DESIGN SYSTEM E CONSISTÊNCIA VISUAL

37. `design.md` como âncora operacional

O `design.md` é a principal referência persistente declarada pelo Dev para:

- tipografia;
- paleta;
- espaçamento;
- direção visual;
- referências;
- comportamento esperado.

Codex e Claude podem reler esse arquivo entre páginas/chats.

38. O código também materializa consistência

A VAL-001 confirmou no Daniel P:

- tokens CSS;
- variáveis globais;
- temas;
- componentes compartilhados;
- layouts parametrizados.

Portanto, a consistência não depende apenas do `design.md`.

39. Consistência híbrida

Formulação segura:

`design.md` orienta intenção e continuidade; o código pode materializar tokens, utilitários, componentes e variantes, coexistindo com decisões locais.

40. Não existe Design System único universal

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não há evidência de:

- um Design System único;
- completamente formal;
- centralizado;
- idêntico em todos os projetos WSS.

Também foram encontradas constantes locais específicas de subsistemas/páginas.

41. O que a consistência busca preservar

No AS-IS, a consistência procura evitar:

- tipografias desconectadas;
- espaçamentos arbitrários;
- cores sem relação com a direção;
- componentes visualmente incoerentes;
- linguagem diferente em páginas que deveriam pertencer ao mesmo sistema.

Mas isso não significa uniformidade total.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE IX — ASSETS E IMAGENS

42. Entrada de imagens do cliente

Quando o cliente envia imagens, normalmente chegam por Google Drive.

O Dev baixa o material.

43. Catalogação com Codex

O Dev pode pedir ao Codex:

- analisar foto por foto;
- catalogar;
- sugerir em qual página/seção cada imagem funciona melhor.

Isso transforma o conjunto bruto de imagens em mapa de uso.

44. Quando falta material

O Dev pode:

- procurar imagens públicas do profissional;
- consultar Doctoralia;
- consultar Google;
- gerar imagens com IA;
- usar Codex para geração contextualizada.

Canva aparece pontualmente.

45. Oportunidade futura sobre origem/autorização

OPORTUNIDADE FUTURA — FORA DO ESCOPO

Foi identificada necessidade futura de revisar:

- uso de imagens públicas;
- autorização;
- origem;
- geração de imagens a partir do cliente.

Este Manual não cria política nova.

46. Compressão, formato e naming

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Compressão, formato e naming:

- são tratados quando necessário;
- mas não constituem hoje uma etapa formal separada do método.

Isso deve ser documentado como ausência, não substituído por pipeline ideal.

47. Imagens e performance

Quando existe peso/lentidão, imagens são um dos primeiros pontos de atenção.

O Dev relatou uso de:

- otimização;
- WebP;
- ajustes de assets.

Mas não existe pipeline universal formal de imagem para todos os projetos.

48. Imagens e responsividade

O enquadramento pode variar por breakpoint.

O Dev pode:

- reposicionar foco;
- ajustar crop;
- tratar mobile separadamente;

sem alterar o desktop já validado.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE X — CONTEÚDO

49. IA participa fortemente da redação

O Dev estimou que cerca de **90% dos textos** são criados ou adaptados por:

- Claude;
- Codex;
- Lovable.

Isso inclui:

- estruturação;
 - adaptação;
 - complementação;
 - ajustes de linguagem.
-

50. Conteúdo pode chegar parcialmente pronto

Copys e direcionamentos podem chegar do:

- briefing;
- contexto;
- decisões do projeto;
- conteúdo fornecido pelo cliente.

A IA não cria obrigatoriamente tudo do zero.

51. Fatos sensíveis não devem ser inventados

O Dev evita inventar ou alterar sem validação:

- formação;
 - títulos;
 - credenciais;
 - procedimentos;
 - resultados;
 - especialidades;
 - locais;
 - números;
 - afirmações médicas;
 - outros fatos sensíveis.
-

52. Informação faltante

Quando falta dado confiável:

- pode ser usado placeholder em casos simples;
 - pode ser registrada pendência em `.md`;
 - o conteúdo pode avançar sem transformar o placeholder em fato validado.
-

53. Validação factual

O cliente funciona como principal validação factual final.

Limite documental

Isso não equivale a:

- revisão editorial formal universal;
 - medical review formal obrigatório;
 - workflow de aprovação de conteúdo padronizado.
-

54. AUSÊNCIA ATUAL DE PADRONIZAÇÃO

O controle de pendências em `.md` depende de acompanhamento humano.

O Dev reconheceu risco de esquecimento.

Esse é um achado AS-IS.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XI — EFEITOS E ANIMAÇÕES

55. Efeitos fazem parte da linguagem visual

Efeitos podem ser usados quando:

- sustentam a referência;
 - combinam com o cliente;
 - aumentam sofisticação;
 - não prejudicam o uso.
-

56. Tecnologias e técnicas observadas

Nos projetos principais aparecem:

- Framer Motion / Motion;
- GSAP;
- ScrollTrigger;
- SplitText;
- Lenis;
- anime.js;
- AOS;
- Three.js;
- React Three Fiber;
- Drei;
- Lottie;
- Swiper;
- CSS animations;
- CSS transitions;
- keyframes;
- IntersectionObserver;
- requestAnimationFrame;
- parallax;
- SVG filters;
- masks;
- clip-path;
- WebGL;
- shaders;
- smooth scroll;
- text reveal.

A lista demonstra variedade, não obrigação.

57. Daniel P como evidência de sofisticação

A VAL-001 confirmou no projeto Daniel P:

- Framer Motion;
- GSAP;
- ScrollTrigger;
- SplitText;
- Three;
- React Three Fiber;
- Drei;
- efeitos de retrato;
- scroll customizado;
- motion em componentes.

Esse caso sustenta que a WSS consegue operar com soluções visuais sofisticadas.

58. Como investigar um efeito

A prática pode incluir:

1. identificar a seção/efeito;
2. fornecer screenshot;
3. fornecer link;
4. pedir estudo do comportamento;
5. investigar técnica/biblioteca;
6. reproduzir de forma isolada;
7. adaptar ao projeto;
8. validar desktop/mobile.

Esse procedimento ficará incorporado ao Manual e aos SOPs de Home/página, não como SOP próprio nesta primeira versão.

59. Quando adaptar, simplificar, substituir ou remover um efeito

Quando um efeito apresenta problema, a preferência prática é preservar seu valor visual sempre que possível:

1. adaptar;
2. simplificar ou substituir;
3. remover apenas quando a solução continuar inviável, instável ou prejudicial.

Essa ordem é aplicada quando o efeito:

- está quebrado;
 - não combina com o cliente;
 - conflita com briefing;
 - prejudica usabilidade;
 - prejudica mobile;
 - gera instabilidade;
 - pesa demais.
-

60. Problemas reais com efeitos

Foram relatados:

- objeto 3D instável;
- borda de Hero quebrando em notebook;
- scroll horizontal prejudicando usabilidade.

Esses exemplos mostram que sofisticação não é objetivo absoluto.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XII — RESPONSABILIDADE

61. Responsividade entra durante a construção

O Dev não espera necessariamente a QA final para pensar em mobile.

Responsividade faz parte do refinamento visual.

62. Tamanhos/dispositivos relevantes

No fluxo atual aparecem como referências práticas:

- desktop/monitor;
- notebook/MacBook;
- mobile físico.

O QA final desses tamanhos pertence ao Manual 6.

63. Desktop e mobile podem exigir soluções diferentes

Um efeito ou composição aprovado no desktop não precisa ser reproduzido de forma idêntica no mobile.

O Dev pode:

- reposicionar/adaptar;
 - simplificar;
 - substituir quando necessário;
 - remover o efeito apenas quando a adaptação/simplificação não resolver de forma aceitável;
 - mudar crop;
 - alterar composição;
 - adaptar navegação.
-

64. Preservar breakpoint já validado

No prompting, o Dev explicita quando:

- desktop já está validado;
- mobile é o único foco;
- ou o contrário.

Isso reduz regressão lateral.

65. AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não existe matriz extensa obrigatória de todos os breakpoints intermediários.

O fluxo atual prioriza tamanhos/dispositivos reais e de maior impacto.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XIII — PERFORMANCE

66. Performance é tratada de forma pragmática

O Dev considera:

- tempo percebido;
- fluidez;
- peso visual;
- efeitos;
- imagens;
- 3D.

A meta é preservar qualidade visual sem tornar a experiência lenta ou instável.

67. Primeiro foco quando há peso/lentidão

Práticas relatadas:

- otimizar imagens;
 - usar WebP;
 - revisar efeitos;
 - revisar animações;
 - revisar 3D;
 - pedir ao Codex otimização sem perder a proposta visual.
-

68. Sofisticação com contenção técnica

No Daniel P foram observados:

- lazy loading;
- chunking;
- separação de `framer-motion`.

Isso demonstra preocupação técnica real em conter o custo de uma interface sofisticada.

Não deve ser transformado em receita universal de bundling.

69. Gate de performance atual

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não existe score numérico universal de aprovação.

O gate declarado é principalmente perceptivo:

- carregamento aceitável;
- fluidez;
- experiência sem travamentos relevantes.

Auditorias técnicas adicionais podem ocorrer, mas não substituem esse achado AS-IS.

70. Performance x referência

Quando o efeito desejado prejudica a experiência, o Dev prioriza:

1. otimizar/adaptar;
2. simplificar ou substituir;
3. remover somente se a solução continuar inadequada.

Fidelidade não é prioridade absoluta diante de quebra/usabilidade/performance.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XIV — ACESSIBILIDADE AS-IS

71. Existe atenção consciente

O AS-IS inclui atenção a:

- contraste;
- tamanho de fonte;
- legibilidade;
- mobile;
- público idoso;
- alt text.

72. Problemas reais detectados

No Daniel P foram encontrados, por exemplo:

- texto de 7px em mobile;
- botão-ícone sem nome acessível.

Esses itens foram tratados como correções específicas.

73. Acessibilidade não é auditoria formal

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não existe hoje auditoria formal de acessibilidade obrigatória em todos os projetos.

Também não existe checklist WCAG universal incorporado ao método atual.

Esse fato precisa permanecer explícito.

74. Limite deste Manual

Este Manual registra o que é praticado hoje.

Ele não introduz:

- certificação;
- score;
- nível WCAG;
- auditoria formal;

- gate de acessibilidade novo.
-

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XV — RELAÇÃO COM NORMA TÉCNICA

75. NORMA TÉCNICA VIGENTE

A construção visual permanece descrita acima como **AS-IS**. Em paralelo, o padrão canônico vigente de SEO/AEO/GEO é:

- `WSS_Padrao_Tecnico_Operacional_SEO_AEO_GEO_v0.2_FINAL.md`.

Quando aplicável à superfície pública, essa norma adiciona requisitos e referências que **não substituem** o julgamento visual do Dev nem reescrevem a prática histórica.

Performance normativa

O padrão define como metas **recomendadas**, no percentil 75 quando houver dados adequados:

- $LCP \leq 2,5 \text{ s}$;
- $INP \leq 200 \text{ ms}$;
- $CLS \leq 0,1$.

Dados de campo/CrUX devem ser priorizados quando disponíveis; PageSpeed Insights e Lighthouse permanecem instrumentos de diagnóstico e regressão. Essas metas não são promessa de ranking e **não viram o gate visual AS-IS deste Manual**.

Terceiros, Chat e CTA

Scripts, widgets, Chat/chatbot/CTA e outros terceiros não devem bloquear a renderização nem causar degradação desproporcional da experiência. A funcionalidade central da página deve continuar disponível diante de falha de terceiro.

Imagens

Na camada normativa:

- `alt` útil/contextual é obrigatório para imagens informativas;
- imagens responsivas/eficientes são recomendadas;
- mídia não deve ser a única portadora de informação essencial quando isso comprometer compreensão/rastreabilidade.

Semântica e acessibilidade

HTML semântico e acessibilidade possuem valor próprio e podem melhorar robustez para tecnologias assistivas e agentes. O padrão **não** autoriza afirmar que ARIA ou acessibilidade geram ranking/GEO direto.

Portanto, permanecem simultaneamente verdadeiras:

- **AS-IS**: não existe auditoria WCAG formal universal incorporada hoje ao método do Dev;
- **NORMA**: requisitos técnicos específicos de semântica, imagens, experiência e acessibilidade devem ser respeitados quando classificados como obrigatórios/aplicáveis pelo padrão vigente.

76. Não reescrever o AS-IS com a norma

A norma pode exigir mais formalização do que o Dev historicamente executa.

Este Manual deve preservar as duas camadas:

AS-IS

Como o Dev constrói hoje.

NORMA

O que o padrão técnico exige.

O detalhamento de SEO/AEO/GEO fica no Manual 6.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XVI — CASOS REAIS

77. Guilherme Mendonça

Uso:

- principal exemplo do fluxo atual;
- Home como âncora;
- desenvolvimento a partir de referências;
- evolução para páginas internas;
- adaptação visual do institucional.

Exclusão

O esboço antecipado antes de todas as definições do cliente não integra o método.

78. Daniel P

Uso:

- caso visualmente sofisticado;
- componentização;
- tokens;
- motion;
- 3D;
- lazy loading;
- chunking.

Esse caso demonstra amplitude técnica, não padrão mínimo obrigatório.

79. O que não generalizar a partir dos casos

Não afirmar que:

- todo site precisa de 3D;
- todo projeto deve usar GSAP;
- todo projeto precisa do mesmo nível de motion;
- todo projeto deve seguir os mesmos tokens;
- toda página interna deve usar layout parametrizado;
- toda Home deve ter a mesma sofisticação.

O caso demonstra capacidade e práticas possíveis.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XVII — AUSÊNCIAS ATUAIS DE PADRONIZAÇÃO

80. Lista consolidada deste Manual

O AS-IS não possui:

- checklist formal universal de análise de referência;
- regra humana formal de componentização;
- Design System único completamente centralizado;
- checklist detalhado universal de “página pronta”;
- pipeline formal separado de compressão/formato/naming de assets;
- revisão editorial/medical review formal universal;
- mecanismo automático para garantir resolução de todas as pendências de conteúdo;
- matriz obrigatória de efeitos por tipo de projeto;
- matriz extensa obrigatória de breakpoints intermediários;
- score numérico universal de performance;
- auditoria formal obrigatória de acessibilidade.

Essas ausências são parte da documentação.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XVIII — OPORTUNIDADES FUTURAS — FORA DO ESCOPO

81. Oportunidades já identificadas

Sem desenvolver solução nesta fase:

- formalizar políticas adicionais de componentização/design system;
- formalizar auditoria de acessibilidade;
- revisar política de imagens públicas/geradas;
- formalizar tratamento de assets;
- aperfeiçoar controle de pendências de conteúdo.

Nenhuma dessas oportunidades altera retroativamente o método atual.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XIX — RELAÇÃO COM OS OUTROS MANUAIS

82. Relação com o Manual 2

O Manual 2 entrega:

- contexto;
- referências;
- `design.md`;
- restrições.

O Manual 3 transforma isso em experiência visual.

83. Relação com o Manual 4

O Manual 4 é proprietário das capacidades do Admin.

Quando Blog, Chat ou outro módulo precisa de adaptação visual, o M3 fornece os princípios visuais; a configuração funcional permanece no M4.

84. Relação com o Manual 5

O Manual 5 explica como:

- Codex;
- Claude;
- Lovable;
- GitHub;

são operados.

O Manual 3 explica **o que o Dev busca construir visualmente com essas ferramentas**.

85. Relação com o Manual 6

O Manual 3 trata:

- responsividade durante construção;
- performance durante construção;
- atenção básica à acessibilidade.

O Manual 6 trata:

- QA;
- regressão;

- testes;
 - validação final;
 - auditorias.
-

86. Relação com o Manual 8

O Manual 8 concentrará:

- caso Daniel P em profundidade;
 - anti-patterns visuais;
 - problemas conhecidos;
 - práticas depreciadas.
-

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XX — SOPs DERIVADOS

87. SOP-003 — Construir e Refinar a Home a partir das Referências

O SOP formaliza deste Manual:

- inputs;
 - leitura da referência;
 - recorte;
 - primeira versão;
 - refinamento;
 - conteúdo;
 - assets;
 - efeitos;
 - responsividade;
 - critério de conclusão.
-

88. SOP-004 — Construir Páginas Internas Mantendo Consistência Visual

O SOP formaliza:

- Sitemap;
 - linguagem da Home;
 - reuso;
 - variação;
 - conteúdo;
 - consistência;
 - breakpoint;
 - critério de pronto.
-

89. Efeitos não viram SOP próprio nesta primeira versão

O procedimento de:

- investigar;
- reproduzir;
- adaptar;

fica incorporado a M3/SOP-003/SOP-004.

Isso evita fragmentação.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE XXI — CRITÉRIO DE CONCLUSÃO

90. Critério de conclusão deste Manual

Este Manual cumpre seu objetivo quando o leitor consegue responder:

1. onde buscar referências;
2. como uma referência é analisada;
3. como ela é adaptada ao cliente;
4. por que a Home é a âncora visual;
5. quando avançar para páginas internas;
6. como o Sitemap orienta cobertura;
7. como reuso e variação coexistem;
8. como a componentização aparece no AS-IS;
9. o que significa Design System híbrido;
10. qual é o papel de `design.md`;
11. como imagens são recebidas/catalogadas;
12. o que acontece quando faltam assets;
13. como IA participa do conteúdo;
14. quais fatos não devem ser inventados;
15. como efeitos são escolhidos;
16. quando um efeito deve ser descartado;
17. como mobile pode divergir do desktop;
18. como performance é tratada;
19. qual é a atenção atual à acessibilidade;
20. quais ausências não devem ser preenchidas com um TO-BE.

91. Fontes principais desta versão

Baseline canônico

1. `WSS_Arqueologia_Tecnica_ASIS_v0.3.md`;
2. `WSS_Matriz_Cobertura_ASIS_P0_P1_P2_v0.3_FINAL.md`;
3. `WSS_Handoff_Consolidacao_Manuais_SOPs_Skills_v0.2.md`;
4. `WSS_Metodo_Desenvolvimento_Sites_Plano_Conducao_v0.3.md`;
5. `WSS_Metodo_Desenvolvimento_Sites_Indice_Mestre_v0.1_DRAFT.md`.

Evidência técnica complementar

- `WSS_VAL_001_DanielP_Componentizacao_DesignSystem_Visual_v0.1.md`.

Manuais já estabilizados usados para preservar fronteiras

- `WSS_Manual_01_Visao_Geral_Toolchain_v1.1_FINAL.md`;

- `WSS_Manual_02_Entrada_Contexto_Projeto_v1.1_FINAL.md` ;
- `WSS_Manual_04_Projeto_Base_Admin_Funcionalidades_v1.1_FINAL.md` ;
- `WSS_Manual_05_Development_Codex_Lovable_GitHub_v1.1_FINAL.md` .

Norma técnica vigente relacionada

- `WSS_Padrao_Tecnico_Operacional_SE0_AE0_GEO_v0.2_FINAL.md` .

Versões anteriores/superadas de AS-IS, Matriz, Handoff e Plano não foram usadas como baseline ativo.

Status: FINAL — APROVADO