

PROCEDIMENTO OPERACIONAL

MÉTODO DE DESENVOLVIMENTO DE SITES

SOP-002

Iniciar Tecnicamente um Projeto WSS: Base →
Lovable → GitHub → Ambiente Local

VERSÃO

v1.1 FINAL

DATA

11/09/2026

STATUS

FINAL — APROVADO

PROCEDIMENTO OPERACIONAL

SOP-002

Iniciar Tecnicamente um Projeto WSS: Base → Lovable → GitHub → Ambiente Local

VERSÃO	v1.1 FINAL
DATA	11/09/2026
STATUS	FINAL — APROVADO
MANUAL PROPRIETÁRIO	Manual 5 — Desenvolvimento com Codex, Lovable e GitHub
MANUAIS RELACIONADOS	Manual 2 — Entrada e Contexto do Projeto; Manual 4 — Projeto-Base, Admin e Funcionalidades
ESCOPO	AS-IS — preparação técnica de um novo projeto WSS até a primeira rodada local

NAVEGAÇÃO DO DOCUMENTO

Sumário

Estrutura do documento canônico v1.1 FINAL.

1. Finalidade	5
2. Gatilho — quando usar	5
3. Quando NÃO usar	5
4. Papéis	6
5. Pré-condições	6
6. Inputs	7
7. Ferramentas	7
PARTE I — PROCEDIMENTO	9
8. Passo 1 — Remixar a Base Admin no Lovable	10
9. Passo 2 — Aplicar o prompt master e os materiais	10
10. Passo 3 — Produzir a primeira base do institucional	10
11. Passo 4 — Chegar ao primeiro ou segundo protótipo	11
12. Passo 5 — Verificar se o protótipo corresponde ao contexto	11
13. Passo 6 — Conectar o projeto ao GitHub	12
14. Passo 7 — Clonar o repositório para a máquina	12
15. Passo 8 — Abrir o projeto na ferramenta local	13
16. Passo 9 — Validar entendimento local	13
17. Passo 10 — Confirmar branch e estado atual	13
18. Passo 11 — Atualizar o ambiente local antes da primeira rodada	14
PARTE II — VALIDAÇÃO DO SETUP	15
19. Checklist de prontidão técnica	16
20. Evidência mínima de conclusão	16
PARTE III — CONDICIONAIS	17
21. O Lovable gerou uma primeira versão fraca	18
22. O Lovable gerou requisito fora do escopo	18
23. O GitHub ainda não deve entrar	18
24. A tarefa exige mudança radical já no início	18

25. Codex e Claude divergem sobre o projeto	19
PARTE IV — STOP CONDITION	20
26. Condição de parada	21
27. O que NÃO é necessário para encerrar	21
PARTE V — CONTINGÊNCIA	22
28. Falha na conexão Lovable ↔ GitHub	23
29. Clone local desatualizado	23
30. Contexto não está acessível localmente	23
PARTE VI — AUSÊNCIAS ATUAIS DE PADRONIZAÇÃO	24
31. Não inventar	25
PARTE VII — NORMA TÉCNICA VIGENTE	26
32. Tratamento	27
PARTE VIII — ERROS CONHECIDOS / ANTI-PATTERNS	28
33. Começar fora do Lovable e tentar trazer depois	29
34. Permanecer todo o projeto apenas no Lovable	29
35. Clonar e editar sem atualizar o repositório	29
36. Editar localmente sem validar contexto	29
37. Criar branch/PR/tag como obrigação documental	29
PARTE IX — OUTPUT	30
38. Saída esperada	31
PARTE X — PRÓXIMO PASSO	32
39. Encaminhamento	33
40. Relação com o fluxo contínuo depois da inicialização	33
41. Referências	33

1. Finalidade

Transformar um projeto já contextualizado em um ambiente técnico pronto para o fluxo híbrido atual da WSS:

```
contexto pronto
→ Base Admin
→ Lovable
→ primeiros protótipos
→ GitHub
→ clone local
→ Codex / Claude
→ primeira rodada local pronta
```

O SOP termina quando o projeto:

- já existe como remix da Base no Lovable;
- já recebeu o contexto/prompt master;
- já possui primeira base/protótipo;
- já está conectado ao GitHub;
- já está clonado localmente;
- já pode ser lido pelas ferramentas locais;
- já está atualizado para iniciar trabalho local.

2. Gatilho — quando usar

Usar quando o **SOP-001 — Contextualizar um Novo Projeto WSS** estiver concluído.

Pré-condição conceitual:

```
contexto compreendido
+ prompt master pronto
→ iniciar SOP-002
```

3. Quando NÃO usar

Não usar:

- para projeto legado que já possui arquitetura própria e precisa receber capacidade da Base;
- para investigar bug em projeto existente;
- para reconstruir histórico técnico;
- para fazer QA;

- para publicar;
- para configurar todos os módulos do Admin;
- para construir a Home em profundidade.

Projetos antigos/legados devem seguir o procedimento de portabilidade correspondente.

4. Papéis

4.1 Desenvolvedor

Responsável por:

- remixar a Base;
- aplicar contexto;
- gerar primeira versão;
- conectar GitHub;
- clonar;
- abrir localmente;
- validar que o projeto está pronto para a primeira rodada técnica.

4.2 IA / ferramentas

Podem participar:

- Lovable;
- Codex;
- Claude.

Nenhuma ferramenta substitui o julgamento humano sobre:

- escopo;
 - risco;
 - versão correta;
 - contexto;
 - necessidade de mudança estrutural.
-

5. Pré-condições

Antes de iniciar:

1. SOP-001 concluído;
2. prompt master pronto;
3. materiais principais disponíveis;
4. Base Admin atual identificada como ponto de partida;

5. acesso ao Lovable;
 6. acesso ao GitHub;
 7. máquina/ambiente local disponível para clone;
 8. ferramenta local escolhida disponível, normalmente Codex e/ou Claude.
-

6. Inputs

Obrigatórios

- prompt master;
- Contexto de Desenvolvimento Consolidado;
- Sitemap;
- Briefing Final;
- referências visuais;
- demais arquivos essenciais do projeto.

Quando existirem

- `contexto.md`;
 - `design.md`;
 - outros `.md`;
 - assets;
 - materiais do cliente.
-

7. Ferramentas

Lovable

Usado para:

- remix da Base;
- primeiros prompts;
- primeira versão/protótipos;
- origem normal do projeto.

GitHub

Usado como ponte entre:

- Lovable;
- ambiente local;
- Codex/Claude.

Ambiente local

Usado para:

- leitura;
- desenvolvimento;
- refinamento;
- investigação.

Codex / Claude

Entram depois do clone conforme a tarefa.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE I — PROCEDIMENTO

8. Passo 1 — Remixar a Base Admin no Lovable

Criar o novo projeto a partir da **Base Admin WSS atual** usando remix.

Objetivo

Começar com:

- estrutura funcional reutilizável;
- Admin;
- Blog;
- módulos/capacidades já disponíveis na Base.

Importante

O projeto deve começar como novo projeto, sem depender do histórico de chat do projeto-base.

A Base é ponto de partida técnico, não produto final do cliente.

9. Passo 2 — Aplicar o prompt master e os materiais

No projeto remixado:

1. enviar o prompt master;
2. disponibilizar documentos e referências necessárias;
3. reforçar restrições importantes;
4. orientar a primeira geração do institucional.

Não fazer

- depender de memória implícita do Lovable;
- assumir que o remix trouxe contexto do cliente;
- tratar conteúdo demo da Base como conteúdo final;
- inventar fatos ausentes.

10. Passo 3 — Produzir a primeira base do institucional

Usar os primeiros prompts para gerar uma primeira versão utilizável.

O objetivo aqui não é concluir o visual.

É criar uma base suficiente para:

- materializar o projeto;

- confirmar que o contexto foi aplicado;
- permitir o próximo estágio híbrido.

11. Passo 4 — Chegar ao primeiro ou segundo protótipo

No AS-IS, o GitHub entra normalmente depois do primeiro ou segundo protótipo.

Critério prático

O projeto deve possuir base suficiente para seguir localmente.

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Não existe:

- número rígido de prompts;
- score;
- checklist automatizado;
- momento exato universal.

A decisão é prática e humana.

12. Passo 5 — Verificar se o protótipo corresponde ao contexto

Antes de conectar o GitHub, confirmar que o projeto não saiu claramente do escopo.

Verificar, no mínimo:

- cliente correto;
- direção inicial coerente;
- Sitemap/estrutura reconhecível;
- ausência de requisito inventado evidente;
- Base correta.

Se houver desvio forte

Corrigir contexto antes de avançar.

Não propagar um projeto claramente mal contextualizado para o fluxo local.

13. Passo 6 — Conectar o projeto ao GitHub

Depois da primeira base:

1. conectar o projeto Lovable ao GitHub;
2. criar/confirmar o repositório correspondente;
3. verificar que o projeto está representado no repositório.

Objetivo

Estabelecer a ponte:

```
Lovable  
↔ GitHub  
↔ local
```

AUSÊNCIA ATUAL DE PADRONIZAÇÃO

Este SOP não define:

- convenção universal de nome;
- organização obrigatória de repositórios;
- PR obrigatório;
- tags obrigatórias.

14. Passo 7 — Clonar o repositório para a máquina

Após a conexão, clonar o repositório para o ambiente local.

O Dev faz isso cedo no fluxo.

Resultado esperado

O projeto local deve conter:

- código;
- documentos;
- configurações versionadas;
- demais artefatos presentes no repositório.

Limite

Este SOP não prescreve comando Git específico.

15. Passo 8 — Abrir o projeto na ferramenta local

Abrir o repositório em:

- Codex;
- Claude;
- ou ambos, conforme necessidade.

Pedir à ferramenta

- ler o projeto;
- entender a estrutura;
- ler os `.md`;
- identificar contexto do cliente;
- resumir antes de editar.

Se um arquivo necessário não estiver acessível

Reenviar ou disponibilizar o arquivo.

Não presumir acesso.

16. Passo 9 — Validar entendimento local

Antes da primeira alteração local, confirmar que a ferramenta consegue explicar:

- o que é o projeto;
- quem é o cliente;
- qual é a direção;
- qual é a estrutura;
- quais áreas já existem;
- quais restrições principais devem ser preservadas.

Se houver perda de contexto

1. interromper;
 2. reforçar documentos;
 3. pedir nova leitura;
 4. pedir novo resumo;
 5. só então iniciar alteração.
-

17. Passo 10 — Confirmar branch e estado atual

No AS-IS, a prática predominante é trabalhar na:

`main`

Regra

Registrar isso como prática atual, não como norma universal de Git.

Branch separada

Pode ser usada em mudança radical.

Quando uma rodada usa branch ou Pull Request, o Codex costuma criar esses artefatos sob instrução/supervisão do Dev. Isso não torna branch/PR obrigatórios, não cria etapa manual fixa para o Dev e não estabelece convenção de nomenclatura.

Não é necessária para a inicialização normal.

18. Passo 11 — Atualizar o ambiente local antes da primeira rodada

Antes de iniciar a primeira rodada local:

- trazer a versão mais recente disponível do repositório.

No AS-IS, o Dev faz pull antes de iniciar trabalho local.

Objetivo

Evitar começar sobre cópia desatualizada.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE II — VALIDAÇÃO DO SETUP

19. Checklist de prontidão técnica

Antes de encerrar o SOP, confirmar:

- [] projeto foi remixado da Base Admin atual;
 - [] prompt master foi aplicado;
 - [] documentos/referências essenciais foram fornecidos;
 - [] primeira base/protótipo existe;
 - [] contexto não apresenta desvio material evidente;
 - [] projeto está conectado ao GitHub;
 - [] repositório foi clonado;
 - [] ferramenta local consegue ler o projeto;
 - [] contexto local foi resumido/validado;
 - [] branch/estado atual foram entendidos;
 - [] ambiente local foi atualizado antes da primeira rodada.
-

20. Evidência mínima de conclusão

O executor deve conseguir mostrar ou confirmar:

1. projeto Lovable existente;
 2. repositório GitHub correspondente;
 3. cópia local do repositório;
 4. primeira versão/protótipo;
 5. ferramenta local com compreensão suficiente;
 6. ambiente pronto para alteração.
-

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE III — CONDICIONAIS

21. O Lovable gerou uma primeira versão fraca

Não é motivo automático para abandonar o fluxo.

Perguntar:

- o contexto está correto?
- a base foi aplicada?
- o protótipo é suficiente para seguir localmente?

Se sim:

seguir para GitHub/local e refinar depois.

O visual final pertence ao Manual 3 / SOP-003.

22. O Lovable gerou requisito fora do escopo

Corrigir antes de seguir.

Não conectar ao fluxo local uma base que já carregue erro estrutural evidente sem tratá-lo.

23. O GitHub ainda não deve entrar

Se o projeto ainda não possui base minimamente utilizável:

- continuar os primeiros prompts no Lovable.

Limite

Não prolongar indefinidamente.

O fluxo atual leva GitHub/local cedo.

24. A tarefa exige mudança radical já no início

Branch separada pode ser considerada.

DECISÃO HUMANA NECESSÁRIA

Não existe threshold formal que obrigue branch.

25. Codex e Claude divergem sobre o projeto

Não escolher automaticamente pelo nome da ferramenta.

Voltar às fontes:

- código;
- contexto;
- documentos;
- estado real do projeto.

Preservar a leitura sustentada pelo projeto.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE IV — STOP CONDITION

26. Condição de parada

O SOP está concluído quando:

o projeto já existe no Lovable a partir da Base, recebeu o contexto, possui primeira base/protótipo, está conectado ao GitHub, foi clonado localmente, a ferramenta local compreende o projeto e o ambiente está atualizado para iniciar a primeira rodada local.

27. O que NÃO é necessário para encerrar

Não exigir nesta etapa:

- Home final;
- páginas internas concluídas;
- Admin totalmente configurado;
- módulos testados;
- QA completo;
- build de produção;
- SEO;
- segurança;
- publicação;
- domínio.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE V — CONTINGÊNCIA

28. Falha na conexão Lovable ↔ GitHub

Se a conexão não se estabelece:

- não criar fluxos paralelos improvisados;
- verificar a conexão/configuração correspondente;
- resolver o bloqueio técnico antes de tratar o projeto local como integrado.

Limite

Este SOP não inventa procedimento técnico específico de recuperação que não esteja documentado nas fontes.

29. Clone local desatualizado

Antes de editar:

- atualizar.

Não começar deliberadamente sobre estado antigo.

30. Contexto não está acessível localmente

Disponibilizar os arquivos necessários.

Não assumir que o código sozinho substitui:

- Briefing;
- Sitemap;
- Contexto;
- `design.md`.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE VI — AUSÊNCIAS ATUAIS DE PADRONIZAÇÃO

31. Não inventar

O AS-IS não possui:

- nomenclatura universal de repositório;
 - número rígido de prompts antes do GitHub;
 - comando Git obrigatório documentado;
 - política formal universal de branching;
 - PR obrigatório;
 - tags obrigatórias;
 - convenção universal de commit;
 - build obrigatório como parte do setup;
 - uso obrigatório simultâneo de Codex e Claude;
 - checklist automático para decidir “hora de clonar”.
-

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE VII — NORMA TÉCNICA VIGENTE

32. Tratamento

Se o projeto estiver sujeito a norma técnica vigente:

- ela continua válida independentemente da ferramenta;
- não deve ser removida pelo remix;
- não deve ser ignorada no primeiro protótipo;
- deve permanecer acessível ao desenvolvimento posterior.

Para SEO/AEO/GEO, a autoridade vigente é:

- WSS_Padrao_Tecnico_Operacional_SEO_AEO_GEO_v0.2_FINAL .

Lovable, Codex, Claude ou outra ferramenta não devem degradar, por conveniência de implementação, os requisitos públicos aplicáveis definidos nessa norma.

Este SOP não redefine a norma.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE VII — ERROS CONHECIDOS / ANTI-PATTERNS

33. Começar fora do Lovable e tentar trazer depois

Prática depreciada como fluxo normal.

O AS-IS atual começa no Lovable/Base.

34. Permanecer todo o projeto apenas no Lovable

Prática depreciada.

O fluxo atual incorpora GitHub/local cedo.

35. Clonar e editar sem atualizar o repositório

Risco:

- divergência;
 - sobrescrita;
 - conflito.
-

36. Editar localmente sem validar contexto

Risco:

- mudança tecnicamente correta para o projeto errado;
 - perda de direção;
 - requisito inventado.
-

37. Criar branch/PR/tag como obrigação documental

Não pertence ao AS-IS universal.

Pode existir caso a caso.

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE IX — OUTPUT

38. Saída esperada

Ao final, devem existir:

No Lovable

- projeto remixado da Base;
- contexto aplicado;
- primeira base/protótipo.

No GitHub

- repositório conectado ao projeto.

Localmente

- clone do repositório;
- ferramenta local com acesso;
- contexto compreendido;
- estado atualizado.

Estado operacional

- pronto para primeira rodada de desenvolvimento local.
-

WSS · MÉTODO DE DESENVOLVIMENTO DE SITES

PARTE X — PRÓXIMO PASSO

39. Encaminhamento

Depois do SOP-002:

Construção visual

Seguir para:

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

quando o foco for Home/direção visual.

Configuração operacional

O Admin pode evoluir em paralelo conforme:

- Manual 4;
- SOP-005 — Configurar Admin e Módulos Aplicáveis.

40. Relação com o fluxo contínuo depois da inicialização

Depois deste SOP, o ciclo normal passa a ser:

```
confirmar contexto
→ pull
→ implementar/investigar
→ validar
→ commit
→ push
→ Lovable atualiza
→ nova rodada
```

Esse ciclo pertence ao Manual 5 e se repete durante o desenvolvimento.

41. Referências

Manual proprietário

- WSS_Manual_05_Desenvolvimento_Codex_Lovable_GitHub_v1.1_FINAL.md

Manuais relacionados

- WSS_Manual_02_Entrada_Contexto_Projeto_v1.1_FINAL.md ;
- WSS_Manual_04_Projeto_Base_Admin_Funcionalidades_v1.1_FINAL.md ;
- WSS_Manual_08_Boas_Praticas_AntiPatterns_Depreciadas_Casos_Reais_v1.1_FINAL.md .

SOP anterior

- WSS_SOP_001_Contextualizar_Novo_Projeto_WSS_v1.1_FINAL.md

Baseline

- WSS_Arqueologia_Tecnica_ASIS_v0.3.md ;
- WSS_Matriz_Cobertura_ASIS_P0_P1_P2_v0.3_FINAL.md ;
- WSS_Handoff_Consolidacao_Manuais_SOPs_Skills_v0.2.md ;
- WSS_Metodo_Desenvolvimento_Sites_Plano_Conducao_v0.3.md .

Status: FINAL — APROVADO