

PROJETO E ABORDAGEM DE DESENVOLVIMENTO			
NextWave Digital Ventures			
Loja de Bebês Reborn			

Controle de Versões			
Versão	Data	Autor	Notas da Revisão
1.1	22/08/2025	Luana Giovana Monteiro Luan Marqueti	Nada a comentar

1 Descrição do projeto a ser desenvolvido

Nome do projeto: Loja de Bebês Reborn (MVP local)

Objetivo:

Demonstrar um e-commerce com **back-end simples** e **fluxo direto de compra** (lista → produto → checkout), persistindo **pedidos** em arquivo local, sem integrações externas.

Justificativa:

Escopo muito pequeno, implementável em 4 semanas:

- Front **estático** consumindo **API local**.
- **Sem carrinho, sem auth, sem deploy**.
- Persistência por **arquivo JSON** (ou SQLite opcional).

Escopo do MVP (apenas o essencial):

1. **Catálogo** de até **6 SKUs** (nome, preço, 1 imagem) servidos por GET /api/products (dados seedados em data/products.json).
2. **Página de produto** consumindo GET /api/products/:id com botão “**Comprar agora**” (opcional: campo quantidade).
3. **Checkout direto**: formulário (nome, e-mail, endereço, produtold, quantidade) que chama POST /api/orders.
4. **Back-end local** (Node/Express):
 - GET /api/products
 - GET /api/products/:id
 - POST /api/orders → gera orderId, calcula total e **salva** em data/orders.json.
5. **Confirmação**: exibe orderId e resumo; opção “**Baixar pedido (.json)**” (gerado no front).
6. **Execução 100% local**: npm start sobe o servidor que serve estáticos e a API (sem deploy).

Fora do escopo :

Carrinho, variações/estoque, cupons, pagamento real, frete/rastreamento, e-mail transacional, relatórios, perfis/perm xões, SEO, LGPD completa, testes automatizados, qualquer forma de deploy.

Premissas/restricções:

- Time 1–2 pessoas; Node 18+; sem dados sensíveis reais.
- Catálogo mantido por arquivo (data/products.json) e pedidos gravados em data/orders.json.
- Alternativa opcional: **SQLite** com 2 tabelas (products, orders) se preferirem.

Critérios de sucesso (acadêmico):

- Catálogo de até **6 produtos** aparece via API e na interface.
- Fluxo **Produto → Checkout → Confirmação** funcional.
- **Pedido persistido** em data/orders.json e disponível para download .json.

Plano de 4 semanas (macro):

- **S1:** Estrutura do projeto; seed products.json; GET /api/products; lista e página de produto.
- **S2:** GET /api/products/:id; tela de produto com **Comprar agora**; esqueleto do checkout.
- **S3:** POST /api/orders (validações mínimas, persistência em arquivo); tela de confirmação + download .json.
- **S4:** Testes manuais, pequenos ajustes visuais, README curto (como rodar e onde ficam os dados).

2 Funcionalidades

1.1 [Catálogo]

- **[MVP] API:** GET /api/products retorna até 6 itens do seed (id, name, price, imageUrl). **Critérios de aceite:** A rota responde em <200ms localmente; lista aparece na **home**.
- **[MVP] Página de lista:** grid com cards (nome, preço, imagem) clicáveis para o detalhe.
- **Produto**
- **[MVP] API:** GET /api/products/:id retorna 1 item.

PROJETO E ABORDAGEM DE DESENVOLVIMENTO		
NextWave Digital Ventures		
Loja de Bebês Reborn		

- [MVP] Página de produto: exibe nome, preço, imagem e botão “Comprar agora” (opcional: **quantidade**, default = 1). Critérios de aceite: Ao clicar “Comprar agora”, o usuário é levado ao checkout levando productId (e quantidade).
- Checkout direto (sem carrinho)
- [MVP] Formulário: **nome, e-mail, endereço** (mínimo), mais **productId** (oculto) e **quantity** (se usada).
- [MVP] API: POST /api/orders
 - Request (JSON): { productId: string, quantity?: number=1, customer: { name:string, email:string, address:string } }
 - Validações mínimas: productId existente; quantity >= 1; name/email/address não vazios.
 - Processo: busca preço do produto, calcula total = price * quantity, gera orderId (ts-rand), grava em data/orders.json (append).
 - Response (201): { orderId, product, quantity, total, customer: {name,email}, createdAt }
 - Erros: 400 (payload inválido), 404 (produto não encontrado).
- [MVP] Tela de confirmação: mostra orderId, resumo do pedido e botão “Baixar pedido (.json)” (front gera o arquivo). Critérios de aceite: Pedido aparece no arquivo data/orders.json; download .json contém os mesmos dados da resposta.
- Não-funcionais (mínimos)
- [MVP] Execução local via npm start (Express serve páginas estáticas e API no mesmo processo).
- [MVP] Logs simples de acesso/erros no console; **sem autenticação e sem TLS**.

3 Abordagem de Desenvolvimento

3.1 Resposta das questões da ferramenta Agile Suitability Filter Tool

2 X3.4.1 CATEGORIA: CULTURA

X3.4.1.1 Adesão (Buy-in) à Abordagem

Há entendimento e apoio do patrocinador sênior para usar ágil neste projeto?

Escala (exemplo): Sim = 1 · Parcial = 5 · Não = 10.

5

X3.4.1.2 Confiança no Time

Considerando os patrocinadores e representantes de negócio que trabalharão com o time: essas partes **confiam** que a equipe pode transformar sua visão e necessidades em um produto/serviço bem-sucedido — com **apoio e feedback contínuos de mão dupla**?

10

X3.4.1.3 Poderes de Tomada de Decisão do Time

O time terá **autonomia** para tomar **decisões locais** sobre como realizar o trabalho?

Escala (exemplo): Sim = 1 · Provável/Talvez = 5 · Improvável = 10.

1

3 X3.4.2 CATEGORIA: TIME

X3.4.2.1 Tamanho do Time

Qual o tamanho do time núcleo?

Use esta escala: 1–9 = 1, 10–20 = 2, 21–30 = 3, 31–45 = 4, 46–60 = 5, 61–80 = 6, 81–110 = 7, 111–150 = 8, 151–200 = 9, 201+ = 10.

1

X3.4.2.2 Níveis de Experiência

Considerando a **experiência e habilidade** dos papéis-chave do time núcleo: é **normal** haver mistura de experientes e iniciantes; porém, para que projetos ágeis fluam melhor, **ajuda** quando **cada papel tem pelo menos um membro experiente**. Escala (exemplo): Sim = 1 · Parcial = 5 · Não = 10.

10

X3.4.2.3 Acesso ao Cliente/Negócio

O time terá **acesso diário** a pelo menos **um representante** de negócio/cliente para **perguntas e feedback**? Escala (exemplo): Sim = 1 · Parcial = 5 · Não = 10.

10

4 X3.4.3 CATEGORIA: PROJETO

PROJETO E ABORDAGEM DE DESENVOLVIMENTO		PMO Project Management Office
NextWave Digital Ventures		
Loja de Bebês Reborn		

X3.4.3.1 Probabilidade de Mudança

Que percentual de requisitos provavelmente **mudará** ou será **descoberto por mês**?

Escala (exemplo): 50% = 1, 25% = 5, 5% = 10.

5

X3.4.3.2 Criticidade do Produto ou Serviço

Para estimar o nível de **verificação e rigor documental** possivelmente necessário, avalie a **criticidade** do produto/serviço considerando a **perda** em caso de **defeitos**. O espectro ilustrativo vai de impacto em **tempo** (baixo) a **fundos discricionários / essenciais**, até risco a **uma vida** ou a **muitas vidas** (alto).

1

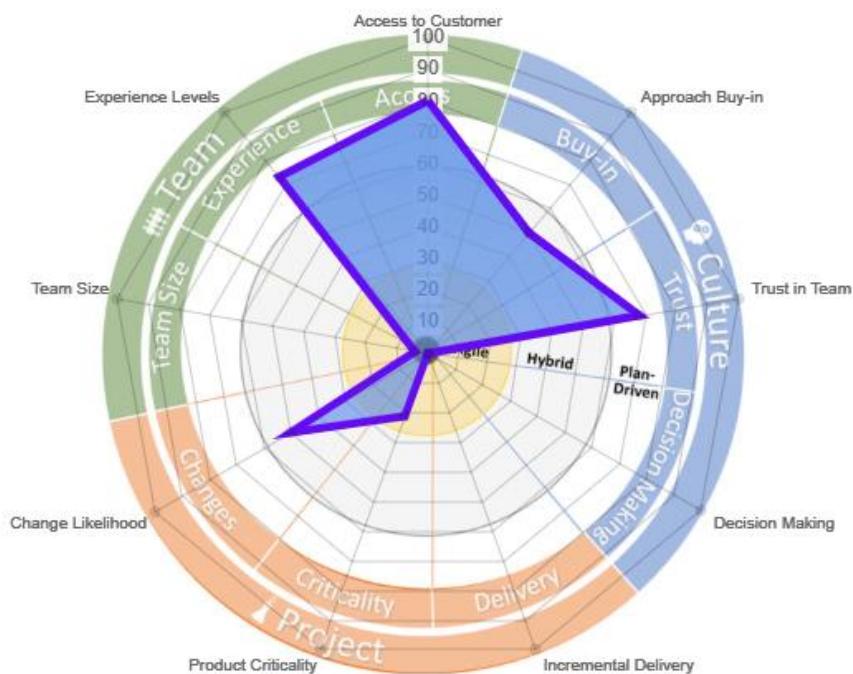
X3.4.3.3 Entrega Incremental

É possível **construir e avaliar** o produto/serviço **em partes**? Haverá representantes de negócio/cliente **disponíveis** para dar **feedback em tempo hábil** sobre os incrementos entregues?

Escala (exemplo): *Sim* = 1 · *Talvez/As vezes* = 5 · *Improvável* = 10.

1

3.2 Gráfico da ferramenta Agile Suitability Filter Tool



3.3 Abordagem escolhida

Abordagem: Híbrida — Ágil na execução (iterações curtas e entrega incremental) com controles preditivos leves na governança/escopo.

PROJETO E ABORDAGEM DE DESENVOLVIMENTO	
NextWave Digital Ventures	
Loja de Bebês Reborn	

Justificativa pelo gráfico (radar de adequação):

Os escores Trust=10, Access=10 e Experience=10 ficaram na periferia, indicando baixa confiança no time, pouco acesso ao cliente/negócio e maturidade técnica reduzida — fatores que desaconselham Ágil puro. Por outro lado, Decision Making=1, Deli-very=1, Team Size=1 e Criticality=1 ficaram próximos ao centro, mostrando autonomia do time, alta viabilidade de entrega incremental, time pequeno e baixa criticidade, todos favoráveis ao Ágil. Buy-in=5 e Changes=5 ficaram na faixa intermediária, reforçando a necessidade de combinar práticas.

Resultado: **Híbrido** maximiza os benefícios de entrega iterativa e ajustes rápidos sem ignorar as restrições organizacionais evidentes no gráfico.

O **híbrido** equilibra o cenário: preserva cadência e valor incremental onde o gráfico indica força para Ágil (autonomia, entrega, tamanho, baixa criticidade) e adiciona contenção e previsibilidade onde o gráfico expõe fraquezas (acesso, confiança e experiência).