IBDIA
INSTITUTO BRASILEIRO DE DADOS E INTELIGÊNCIA ARTIFICIAL
Inovação, Ciência e Impacto com Inteligência

MANUAL DE DESENVOLVIMENTO DE PRODUTOS DE PD&I

IBDIA

Da ideia à pesquisa, da evidência ao produto e à transferência tecnológica.

Documento 6 da Arquitetura Institucional de PD&I | Versão Pública 1.1 | Setembro de 2026

VERSÃO PÚBLICA — Documento Institucional. Conteúdo destinado à divulgação externa. Procedimentos internos, alçadas, controles operacionais e informações administrativas foram suprimidos ou resumidos quando necessário. O documento interno de referência permanece preservado.

1. Finalidade e escopo

Este Manual estabelece o processo institucional para transformar problemas, hipóteses, pesquisas e protótipos em ativos tecnológicos e, quando houver evidência suficiente, em produtos, pilotos, licenças ou transferências tecnológicas. Aplica-se aos projetos dos 12 Núcleos, aos 6 Programas Estratégicos (GeoAI, Forecast AI, OpenData AI, Responsible AI, AgroAI e Energy & Resources AI), à GeoAI Platform, FazAí, IBDIA Intelligence, SEGI, TerraLens AI, aos projetos transversais Retail Intelligence e AstroAI, ao Technology Core e a futuras iniciativas.

Princípio central
O IBDIA não deve confundir atividade de desenvolvimento com progresso de produto. Cada etapa precisa produzir evidência verificável. Um projeto pode terminar como publicação, dataset, método ou software científico sem se tornar produto comercial.

2. Fluxo institucional de desenvolvimento

Estágio	Fase	Pergunta principal	Saída mínima
E0	Ideia / Problema	Existe um problema ou oportunidade relevante?	Ficha de oportunidade.
E1	Conceito / Discovery	Para quem, por quê e qual valor pretendido?	Concept brief + escopo.
E2	Pesquisa / Experimento	Há caminho científico/técnico plausível?	Estado da arte, dados, baseline e experimento.
E3	PoC	A hipótese funciona tecnicamente?	Prova de conceito reproduzível.
E4	MVP	É possível entregar o fluxo de valor ponta a ponta?	MVP utilizável e instrumentado.
E5	Validação	Funciona com qualidade, segurança e dados adequados?	Relatório de validação.
E6	Piloto	Funciona em ambiente/usuário real?	Piloto monitorado e aceite técnico.
E7	Produto	É operável, documentado e suportável?	Release operacional.
E8	Escala / Transferência	Há modelo sustentável de adoção?	Licenciamento, operação, transferência ou spin-off.

3. Gates de decisão

Gate	Decisão	Critérios mínimos
G1	E0 → E1	Problema claro; aderência ao IBDIA; responsável; usuário/beneficiário identificável.
G2	E1 → E2	Hipótese; estado da arte inicial; dados/fontes possíveis; riscos; escopo controlado.
G3	E2 → E3	Baseline; método; experimento reproduzível; viabilidade técnica demonstrável.
G4	E3 → E4	PoC consistente; arquitetura mínima; backlog MVP; custo e dependências conhecidos.
G5	E4 → E5	Fluxo ponta a ponta; testes; métricas; logging; documentação; dados de validação.
G6	E5 → E6	Critérios de sucesso; parceiro/usuário; segurança; privacidade; plano de piloto.
G7	E6 → E7	Resultado do piloto; confiabilidade; operação; suporte; segurança; PI/licenças.
G8	E7 → E8	Demanda/adoção; custos; modelo jurídico/comercial; capacidade de escala e governança.

4. Papéis e responsabilidades

Papel	Responsabilidade
Sponsor / Patrocinador interno	Defende relevância estratégica, remove impedimentos e valida decisões de portfólio.
Product Owner / Líder de Produto	Problema, usuários, prioridades, backlog, métricas e coordenação da evolução.
Research Lead	Método científico, estado da arte, desenho experimental e validade das

	conclusões.
Tech Lead / Arquiteto	Arquitetura, padrões, integração, segurança, qualidade e decisões técnicas.
Data/ML/AI Engineer	Pipelines, features, modelos, avaliação, inferência e MLOps.
Software Engineer	Backend, frontend, APIs, integrações, testes e operação.
UX/UI	Pesquisa de usuário, fluxos, protótipos, acessibilidade e experiência.
Especialista de domínio	Requisitos técnicos, interpretação, limites e validação setorial.
Responsible AI / Segurança	Riscos, privacidade, fairness, robustez, documentação e controles.
Parceiro piloto	Dados/ambiente real, feedback, validação e aceite conforme acordo.

5. E0 — Ideia e enquadramento

- Registrar problema, contexto, usuário/beneficiário e resultado desejado.
- Indicar Núcleo(s), Programa potencial e possível produto/família.
- Explicitar por que IA/dados/geotecnologia são necessários; evitar IA sem necessidade.
- Identificar soluções existentes e risco de duplicação com o portfólio interno.
- Estimar ordem de grandeza de dados, pessoas, tempo, cloud/GPU e dependências.

Entregável E0	Conteúdo mínimo
Ficha de oportunidade	Título; problema; usuário; hipótese de valor; Núcleo; Programa; responsável; riscos; próximos passos.
Decisão	Avançar, manter no backlog, incorporar a iniciativa existente ou arquivar.

6. E1 — Discovery e conceito

- Entrevistar ou observar usuários quando possível.
- Definir jobs-to-be-done, casos de uso e fora de escopo.
- Mapear concorrentes, alternativas e tecnologias internas reutilizáveis.
- Definir métricas de sucesso técnicas, de usuário e, quando aplicável, econômicas.
- Criar arquitetura conceitual e hipótese de dados.

Documento	Conteúdo
Concept Brief	Problema, público, proposta de valor, casos de uso, escopo e restrições.
Mapa de hipóteses	Hipótese de usuário, valor, dados, modelo, operação e adoção.
Backlog de pesquisa	Questões que precisam ser respondidas antes do MVP.
Registro de riscos	Dados, ética, segurança, PI, dependências e incertezas.

7. E2 — Pesquisa e experimentação

E2 é a fase em que a solução deve ganhar fundamento científico/técnico antes de ser transformada em software de produto.

- Revisão do estado da arte e definição de baseline.
- Aquisição/curadoria de dados com Data Card inicial.
- Experimentos versionados e reproduzíveis.
- Separação correta entre treino, validação e teste; prevenção de leakage.
- Registro de métricas, hiperparâmetros, artefatos e limitações.
- Análise de custo computacional e viabilidade operacional.

8. E3 — Prova de Conceito (PoC)

A PoC demonstra viabilidade técnica de uma hipótese específica. Não deve carregar requisitos de produto que não sejam necessários à demonstração.

Obrigatório na PoC	Critério
Reprodutibilidade	Código, ambiente, dados/amostra e instruções suficientes para repetir o resultado.
Baseline	Comparação com método simples ou referência relevante.
Métrica	Critério objetivo de sucesso definido antes da conclusão.
Limitações	Condições em que o resultado não deve ser generalizado.
Decisão	Avançar para MVP, repetir pesquisa, reformular, incorporar ou arquivar.

9. E4 — MVP

O MVP deve provar valor ponta a ponta com o menor conjunto de funcionalidades necessário. Não é sinônimo de software mal construído.

- Fluxo principal funcional do início ao fim.
- Autenticação e permissões mínimas quando houver usuários/dados protegidos.
- Backend/API e interface suficientes ao caso de uso.
- Logging, métricas de uso, custo e erros.
- Tratamento de falhas previsíveis e mensagens compreensíveis.
- Documentação de instalação, execução e limitações.
- Instrumentação para aprender com a validação.

10. E5 — Validação

Dimensão	Exemplos de validação
Técnica	Acurácia, erro, latência, throughput, robustez, disponibilidade e custo.
Dados	Qualidade, cobertura, drift, missingness, proveniência e atualização.
Usuário	Conclusão de tarefa, tempo, erros, satisfação, compreensão e utilidade.
IA responsável	Bias/fairness quando aplicável, explicabilidade, segurança, limites e supervisão.
Produto	Uso recorrente, retenção, conversão ou eficiência operacional quando pertinente.
Domínio	Validação por especialista e comparação com prática/referência técnica.

11. E6 — Piloto

- Definir escopo, duração, participantes, ambiente, dados e responsáveis.
- Assinar instrumentos necessários para confidencialidade, dados, PI e responsabilidades.
- Congelar critérios de sucesso antes do início.
- Monitorar qualidade, incidentes, custos e comportamento do usuário.
- Registrar mudanças realizadas durante o piloto.
- Emitir relatório final com decisão E6 → E7 ou retorno a estágio anterior.

12. E7 — Produto

Área	Requisito para E7
Arquitetura	Componentes documentados, integrações, limites e estratégia de evolução.
Operação	Deploy reproduzível, ambientes, observabilidade, backups e plano de incidentes.
Segurança	Segredos, autenticação, autorização, dependências, vulnerabilidades e logs tratados.
Qualidade	Testes automatizados adequados ao risco e critérios de release.
IA/ML	Model/Data Cards, avaliação, versionamento, monitoramento e rollback.
UX	Fluxos revisados, acessibilidade proporcional ao público e tratamento de erros.
Suporte	Responsáveis, documentação, SLA/SLO quando aplicável e processo de manutenção.
Jurídico/PI	Licenças, direitos sobre dados/código/modelos e forma de exploração definidos.

13. E8 — Escala e transferência tecnológica

- Definir se a tecnologia será operada pelo IBDIA, licenciada, transferida, coexplorada ou destinada a spin-off.
- Estabelecer custos unitários, capacidade, limites e estratégia de infraestrutura.
- Automatizar provisionamento, observabilidade e processos repetitivos.
- Formalizar treinamento, documentação e pacote de transferência.
- Definir versionamento de longo prazo, suporte, depreciação e encerramento.

14. Gestão de repositórios e GitHub

Diretriz pública: os projetos utilizam repositórios controlados, versionamento e rastreabilidade compatíveis com sua maturidade e criticidade. Os procedimentos operacionais detalhados são de uso interno.

15. Estratégia de versionamento

Diretriz pública: versões de software, modelos, dados e documentação devem ser identificáveis e reproduzíveis. Regras internas de nomenclatura e release foram suprimidas.

16. Gestão de dados

- Todo dataset relevante deve possuir proprietário/responsável, origem, licença/base de uso, período, esquema, qualidade e restrições.
- Separar dados brutos, processados, features e resultados.
- Evitar copiar dados sensíveis para notebooks, repositórios ou ambientes pessoais.
- Aplicar minimização, controle de acesso e retenção conforme risco.
- Registrar transformações para permitir rastreabilidade.
- Datasets de demonstração devem ser públicos, autorizados, sintéticos ou devidamente anonimizados.

Artefato	Conteúdo mínimo
Data Card	Origem, finalidade, cobertura, licença, variáveis, qualidade, limitações, sensibilidade e atualização.
Data Dictionary	Campos, tipos, unidades, domínios e regras.
Pipeline record	Ingestão, transformações, validações e versão.
Split record	Critério de treino/validação/teste e prevenção de leakage.

17. Desenvolvimento de modelos de IA/ML

Etapa	Padrão mínimo
Baseline	Começar com referência simples e mensurável.
Experimento	Configuração rastreável e reproduzível.
Avaliação	Métricas adequadas ao problema e análise de erro.
Robustez	Testes fora do cenário ideal e dados degradados quando pertinente.
Explainability	Aplicar quando necessária ao risco, domínio ou decisão.
Model Card	Uso pretendido, dados, métricas, limitações, riscos e versão.
Deploy	Artefato versionado, dependências fixadas e estratégia de rollback.
Monitoramento	Qualidade, drift, latência, custo, erros e comportamento relevante.

18. IA Generativa, LLMs, RAG e agentes

- Versionar prompts, templates, ferramentas e políticas de sistema.
- Separar avaliação de recuperação (retrieval) da avaliação da resposta gerada.
- Registrar fontes/evidências em aplicações de RAG sempre que a experiência exigir verificabilidade.
- Criar testes contra prompt injection, vazamento de contexto e uso indevido de ferramentas.
- Controlar custos por requisição, tokens, modelo e usuário/projeto.
- Não assumir que troca de modelo preserva comportamento; reavaliar antes do release.
- Para agentes, limitar permissões e exigir confirmação humana para ações de impacto relevante.

19. APIs e integrações

Tema	Diretriz
Contrato	OpenAPI ou documentação equivalente para APIs externas/internas relevantes.
Autenticação	Método adequado ao contexto; nunca depender de segredo embutido no cliente.
Erros	Códigos e mensagens consistentes; não expor detalhes sensíveis.
Idempotência	Aplicar em operações em que repetição possa gerar efeitos duplicados.
Rate limit	Definir limites quando houver custo, abuso ou capacidade restrita.
Observabilidade	Request IDs, métricas, logs e tracing quando necessário.
Compatibilidade	Mudanças quebráveis devem seguir política de versão/depreciação.

20. UX, interface e acessibilidade

- Projetar a partir das tarefas do usuário, não das capacidades do modelo.
- Exibir incerteza, fontes, limitações ou necessidade de revisão humana quando relevante.

- Evitar interfaces que induzam confiança excessiva em resultados probabilísticos.
- Usar protótipos de baixa fidelidade antes de investir em interface completa.
- Testar com usuários representativos sempre que possível.
- Aplicar princípios de acessibilidade e responsividade proporcionais ao público e ao estágio.

21. Testes e qualidade de software

Diretriz pública: testes devem ser proporcionais ao risco, ao estágio E0–E8 e ao uso previsto da tecnologia. Checklists e critérios internos de aceite foram suprimidos.

22. CI/CD e ambientes

Diretriz pública: automação de integração, entrega e ambientes é adotada quando adequada à maturidade do projeto. Configurações e procedimentos internos foram suprimidos.

23. Segurança e privacidade

Controle	Diretriz
Segredos	Secret manager ou variáveis protegidas; rotação e menor privilégio.
Acesso	RBAC/ABAC conforme necessidade; revisão periódica.
Dependências	Atualização, scanning e inventário de componentes.
Dados	Criptografia em trânsito e repouso quando apropriado; minimização e segregação.
Logs	Não registrar senhas, tokens ou dados sensíveis desnecessários.
Incidentes	Canal, responsáveis, contenção, registro e aprendizado.
Terceiros	Avaliar APIs/modelos/cloud quanto a dados, retenção, termos e riscos.

24. IA responsável por estágio

Estágio	Ação mínima
E0–E1	Identificar impacto, usuários afetados e usos indevidos previsíveis.
E2	Avaliar dados, representatividade, limitações e riscos do método.
E3	Documentar limitações e testar falhas relevantes.
E4	Criar Model/Data Cards iniciais e controles de interface.
E5	Executar avaliação de fairness/robustez/explicabilidade conforme risco.
E6	Monitorar incidentes e efeitos em uso real.
E7–E8	Governança contínua, auditoria, drift, revisão de risco e processo de retirada.

25. Documentação obrigatória por maturidade

Artefato	E2	E3	E4	E5	E6	E7
README técnico	●	●	●	●	●	●
Concept/Research Brief	●	●	●	●	●	●
Data Card	●	●	●	●	●	●
Model Card		●	●	●	●	●
Arquitetura		○	●	●	●	●
Plano/testes		○	●	●	●	●
Security/Privacy Review			○	●	●	●
Relatório de validação				●	●	●
Plano de piloto					●	●
Runbook/Operação					○	●
Licenças/PI		●	●	●	●	●

Legenda: ● obrigatório; ○ recomendado/obrigatório conforme risco.

26. Propriedade intelectual e open source

- Antes de publicar código, pesos, datasets ou documentação, verificar titularidade e obrigações contratuais.
- Registrar bibliotecas, modelos e datasets de terceiros e suas licenças.
- Definir se o ativo será aberto, parcialmente aberto, fechado, licenciado ou protegido.
- Contribuições de pesquisadores e parceiros devem possuir regras claras de titularidade e exploração.
- Não misturar código incompatível com a estratégia de licenciamento do produto.

27. Arquitetura e reutilização — Technology Core

Antes de construir componente novo, a equipe deve verificar se o Technology Core já oferece capacidade equivalente.

Capacidade	Ativos preferenciais
Usuários/projetos	Identity & Project Core
LLMs	LLM Orchestration
Documentos	Document Generation / Reporting Engine
Geoespacial	Geospatial Engine / GeoAI Engine
Forecast	Forecast Engine
Anomalias	Anomaly Engine
RAG	RAG Engine
Responsible AI	Responsible AI Engine
Recomendação	Recommendation Engine

Regra de arquitetura
Duplicação só deve ser aceita quando houver justificativa técnica documentada. Um novo engine deve possuir consumidor real e perspectiva de reutilização.

28. Métricas de produto e PD&I

O IBDIA acompanha indicadores científicos, tecnológicos, de uso, qualidade e impacto de forma proporcional ao estágio do projeto. Metas e painéis operacionais internos não integram esta versão.

29. Critérios de pausa, incorporação e encerramento

Decisão	Quando aplicar
PAUSAR	Dependência temporária de dados, parceiro, equipe, infraestrutura ou financiamento.
INCORPORAR	A iniciativa é melhor tratada como módulo/engine de ativo existente.
REFORMULAR	Problema é válido, mas hipótese, arquitetura, UX ou escopo falhou.
ARQUIVAR	Baixa evidência, duplicação, custo desproporcional ou perda de relevância.
ENCERRAR COM SUCESSO	Pesquisa atingiu objetivo científico/técnico mesmo sem produto.
TRANSFERIR	Tecnologia madura deve passar a parceiro, empresa, operação dedicada ou spin-off.

30. Processo de release

O processo de release requer versionamento, documentação, validação e autorização compatíveis com o estágio e risco. O fluxo operacional detalhado é de uso interno.

31. Checklist para entrada em piloto

A entrada em piloto depende de evidências técnicas, segurança, dados, responsabilidades e plano de acompanhamento adequados. O checklist operacional foi suprimido.

32. Checklist para classificação como produto E7

A classificação como produto E7 exige validação suficiente, documentação, suporte, governança e condições de operação definidas. O checklist interno foi suprimido.

33. Aplicação do Manual ao portfólio atual

Família	Aplicação imediata do Manual
GeoAI Platform	Separar módulos por estágio; benchmarks e Data/Model Cards; pilotos por domínio.
FazAí	Priorizar Core + Livro; medir custo por geração, ativação e retenção antes de multiplicar apps.
Forecast AI	Construir baseline/Engine antes das verticais; backtesting rigoroso.
OpenData AI	Priorizar proveniência, evidência, avaliação de RAG e conectores governados.
Responsible AI	Usar o próprio framework para avaliar produtos internos desde E4/E5.
Retail Intelligence	Validar com dados adequados e medir incremento real em pilotos.
SEGI	Reformular sobre Technology Core e validar fluxo principal antes de expandir módulos.
TerraLens AI	Decidir incorporação/posicionamento em relação à GeoAI antes de duplicar desenvolvimento.
AstroAI	Priorizar reprodutibilidade, publicação e contribuição científica; produto comercial não é requisito.

34. Governança e cadência

A governança acompanha o desenvolvimento por ciclos periódicos e gates proporcionais à maturidade. A cadência administrativa detalhada é de uso interno.

35. Templates institucionais mínimos

O IBDIA utiliza templates padronizados para projeto, risco, dados, modelos, validação, release e transferência. Os modelos operacionais internos não integram a versão pública.

36. Conclusão

Este Manual cria uma disciplina única para o desenvolvimento tecnológico do IBDIA. A ideia central é simples: não acelerar pulando etapas, mas acelerar reduzindo retrabalho. Problemas são enquadrados antes do código; pesquisa gera evidência antes do produto; componentes comuns são reutilizados; dados e modelos são versionados; validação precede escala; e cada tecnologia possui uma decisão explícita de continuidade, incorporação, encerramento ou transferência. Com esse processo, o crescimento do portfólio deixa de depender de projetos isolados e passa a construir patrimônio científico e tecnológico cumulativo.