# Projeto

A BiaBot é uma plataforma brasileira de ferramentas clínicas para médicos. A ferramenta principal é o **extrator de exames**: recebe laudos de exames de imagem e laboratoriais (PDF, imagem escaneada ou foto de celular) e os transforma num modelo estruturado e amigável — valores, unidades, faixas de referência, achados e conclusões organizadas de forma padronizada.

A proposta de valor central é reduzir o tempo que o médico gasta lendo e transcrevendo exames em formatos heterogêneos (cada laboratório tem seu layout) e entregar uma visão limpa, comparável e pronta para consulta.

Situação atual: a plataforma ainda não tem identidade visual própria, e o uso depende de acesso à aplicação web. Há a intenção de levar o extrator para o WhatsApp, onde o médico envia o documento por mensagem e recebe de volta o PDF estruturado.

**Objetivo da sprint INSPER:** construir um sistema funcional front-end + back-end, com foco forte em UX e UI, que receba documentos médicos e apresente o resultado estruturado — opcionalmente integrado ao WhatsApp. A análise e modelagem do documento **não** é escopo dos alunos: a BiaBot disponibiliza um serviço de extração pronto e um conjunto de exames simulados.

# Estrutura de usuários

| Perfil | Descrição |
|---|---|
| Médico (usuário final) | Clínico, especialista ou residente. Recebe exames de pacientes em formatos diversos, tem pouco tempo e usa muito o celular. Quer ver o essencial em segundos: o que está alterado e o que mudou desde o último exame. |
| Equipe de produto BiaBot | Define identidade, prioridades e narrativa comercial. Consome protótipos, métricas de uso e materiais de apresentação. |
| Time técnico BiaBot | Mantém o serviço de extração e vai incorporar o que a sprint produzir. Consome código, APIs, documentação e infraestrutura reproduzível, sem camada visual. |

# Insumos disponíveis

A BiaBot vai entregar à sprint:

- **Serviço de extração (API)** — recebe um documento e devolve o exame estruturado em JSON. Endpoint, autenticação, limites de uso e tempo médio de resposta: *a confirmar com o ponto focal*.
- **Conjunto de exames simulados** — laudos laboratoriais (hemograma, bioquímica, perfil lipídico etc.) e de imagem (ultrassom, tomografia, ressonância, raio-X), em PDF e imagem, incluindo fotos de celular com qualidade variada. Nenhum dado de paciente real.
- **Schema de saída** — documentação dos campos do JSON (paciente simulado, tipo de exame, data, laboratório, analitos com valor/unidade/referência, achados, impressão diagnóstica).

# Histórias

# NÍVEL FÁCIL — Identidade e porta de entrada

## História 1 — Identidade visual e design system

Como equipe de produto da BiaBot, quero uma identidade visual única e um design system documentado, para que todas as telas, PDFs e mensagens da plataforma tenham uma cara consistente, transmitam confiança clínica e sejam reconhecíveis pelo médico.

**Contexto do desafio**

Hoje a BiaBot não tem identidade própria; cada tela foi feita sem um padrão. Em saúde, a percepção de confiabilidade é decisiva — e um sistema visual bem definido também acelera o trabalho de todas as outras histórias desta sprint.

**Critérios de aceitação**

- Proposta de marca: logotipo (versões horizontal, símbolo e monocromática), paleta, tipografia e tom de voz.
- Paleta com cores semânticas para o contexto clínico (normal, atenção, alterado/crítico), testada para contraste WCAG AA e para daltonismo.
- Design system no Figma com tokens (cor, tipografia, espaçamento, raio) e no mínimo 15 componentes (botões, inputs, card de exame, tabela de analitos, badges de status, área de upload etc.).
- Tokens exportados para código (CSS variables ou tema Tailwind) e catálogo de componentes em Storybook ou equivalente.
- Pelo menos 2 rodadas de validação com médicos ou com a equipe BiaBot, com registro do feedback.

**Perguntas orientadoras**

- Como transmitir confiança clínica sem cair no visual genérico "hospital azul e branco"?
- Como indicar valores alterados sem depender só de cor?
- Faz sentido ter modo escuro (plantão noturno)?
- Como a identidade se adapta a canais com pouco controle visual, como o WhatsApp?

**Entregáveis**

- Brand book (PDF ou página) com usos corretos e incorretos da marca.
- Arquivo Figma do design system.
- Pacote de tokens + Storybook publicado.
- Registro das validações e decisões de design.

## História 2 — Site institucional com demonstração

Como médico que ouviu falar da BiaBot, quero acessar um site que explique em poucos segundos o que a plataforma faz e me deixe testar com um exame de exemplo, para que eu decida se vale criar uma conta.

**Contexto do desafio**

Sem uma porta de entrada clara, o médico não entende o valor do extrator antes de se cadastrar. A demonstração com exame simulado — laudo bagunçado entrando, exame organizado saindo..

**Critérios de aceitação**

- Home page responsiva aplicando a identidade da História 1 (ou uma identidade provisória acordada com a BiaBot).
- Seções: proposta de valor, como funciona (3 passos), antes × depois (laudo bruto × exame estruturado), segurança e privacidade, chamada para cadastro.
- Demonstração interativa com pelo menos 3 exames simulados, sem necessidade de cadastro.
- Lighthouse ≥ 90 em performance e acessibilidade no mobile.
- Formulário de lista de espera/cadastro com persistência e eventos de analytics (visita, demo iniciada, cadastro).

**Perguntas orientadoras**

- Qual mensagem convence um médico: tempo economizado, menos erro de transcrição ou visão histórica do paciente?
- Como comunicar segurança e LGPD sem jargão jurídico?

# NÍVEL INTERMEDIÁRIO — Sistema funcional front + back

## História 3 — Upload e pipeline de processamento de documentos

Como médico, quero enviar um ou vários exames (PDF, imagem ou foto do celular) e acompanhar o status do processamento, para que eu não fique esperando numa tela travada e saiba exatamente quando o resultado está pronto.

**Contexto do desafio**

O serviço de extração pode levar de segundos a minutos e pode falhar (foto borrada, documento ilegível, arquivo que nem é exame). Uma chamada síncrona simples não escala. Esta história constrói o back-end assíncrono que conecta o upload ao serviço da BiaBot.

**Critérios de aceitação**

- Upload com arrastar e soltar e captura pela câmera no mobile; múltiplos arquivos; validação de tipo e tamanho.
- Back-end com fila de processamento (ex.: Redis + Celery/RQ/BullMQ) e workers que chamam o serviço de extração.
- Status por documento (recebido → processando → concluído → erro) atualizado em tempo real
- Tratamento de falhas: retry com backoff, timeout e mensagem de erro compreensível ("não conseguimos ler esta foto — tente com mais luz").

**Perguntas orientadoras**

- Fila, cron ou chamada síncrona: quando cada um se justifica?
- Como evitar processar o mesmo exame duas vezes?
- Vale pré-processar a imagem (rotação, corte, contraste) antes de enviar ao serviço?

## História 4 — Visualização de exames laboratoriais

Como médico, quero ver um exame laboratorial organizado por grupos, com os valores fora da referência destacados, para que eu identifique o que está alterado em segundos, sem ler o laudo inteiro.

**Contexto do desafio**

Laudos laboratoriais misturam dezenas de analitos, unidades e faixas de referência num layout que muda a cada laboratório. O JSON do serviço já traz os dados estruturados; o desafio aqui é inteiramente de apresentação e usabilidade.

**Critérios de aceitação**

- Tela de resultado agrupada (hemograma, função renal, lipídios...) com valor, unidade, faixa de referência e indicador visual.
- Resumo no topo ("X de Y analitos fora da referência"), com os alterados listados primeiro.
- Indicador da posição do valor dentro/fora da faixa (ex.: barra de referência), sem depender só de cor.

- Filtro "só alterados", busca por analogia e visualização do documento original lado a lado.
- Responsivo e mobile-first.
- Teste de usabilidade com pelo menos 5 usuários (médicos ou estudantes de medicina), medindo o tempo para encontrar os valores alterados.

**Perguntas orientadoras**

- Como lidar com faixas de referência que variam por sexo/idade ou que não vieram no laudo?
- Como mostrar que o extrator teve baixa confiança num campo?
- Tabela densa ou cards? O que o médico prefere — e como descobrir?

## História 5 — Visualização de laudos de exames de imagem

Como médico, quero ver laudos de imagem (ultrassom, tomografia, ressonância, raio-X) organizados em técnica, achados por estrutura anatômica e impressão diagnóstica, para que eu encontre rapidamente a conclusão e os achados relevantes.

**Contexto do desafio**

Laudos de imagem são majoritariamente texto livre; o serviço os estrutura em seções. A visualização precisa respeitar o texto do radiologista (sem reescrever) e, ao mesmo tempo, torná-lo fácil de escanear.

**Critérios de aceitação**

- Layout com a impressão diagnóstica em destaque e achados agrupados por órgão/região.
- Destaque dos achados que exigem atenção conforme a marcação vinda do serviço — a plataforma não cria interpretação própria.
- Navegação por seções e link para o trecho correspondente no documento original.
- Suporte a pelo menos 4 modalidades de exame entre os simulados.
- Validação com pelo menos 3 usuários do público-alvo.

**Perguntas orientadoras**

- Onde termina "organizar" e começa "interpretar"? Qual o risco de cada?
- Um mapa anatômico simplificado ajuda ou distrai?
- Como exibir medidas (ex.: dimensões de um nódulo) de forma comparável?

## História 6 — Geração do PDF estruturado com a identidade BiaBot

Como médico, quero baixar ou receber um PDF limpo e padronizado do exame estruturado, para que eu possa anexá-lo ao prontuário, imprimir ou enviar a um colega.

**Contexto do desafio**

O PDF é o artefato que circula fora da plataforma — inclusive é o que volta pelo WhatsApp (História 8). Ele precisa ser bonito, legível, impresso em preto e branco e rastreável até o documento de origem.

**Critérios de aceitação**

- Serviço de geração de PDF a partir do JSON
- Templates para exame laboratorial e para exame de imagem, aplicando a identidade visual.
- Legível em impressão P&B (alterados distinguíveis sem cor), formato A4, paginação, cabeçalho e rodapé.
- Rodapé com origem (laboratório, data do exame, data de geração) e aviso de que o documento reorganiza o laudo original e não o substitui.
- Geração em menos de 5s para um exame típico; endpoint documentado.

**Perguntas orientadoras**

- HTML→PDF ou biblioteca nativa? Trade-offs de fidelidade e performance?
- Incluir o laudo original como anexo no mesmo PDF?
- Como versionar templates sem quebrar PDFs já emitidos?

## História 7 — Área do médico: autenticação, pacientes e histórico

Como médico, quero ter uma conta segura onde meus exames ficam organizados por paciente, para que eu encontre qualquer exame enviado antes e mantenha os dados protegidos.

**Contexto do desafio**

Dados de saúde são dados pessoais sensíveis pela LGPD (art. 11). Mesmo trabalhando só com dados simulados, a arquitetura precisa nascer preparada para dados reais.

**Critérios de aceitação**

- Cadastro e login (e-mail/senha + pelo menos um segundo fator ou login social) e recuperação de senha.
- Cadastro de pacientes (dados mínimos, simulados) e vínculo de exames a pacientes.
- Lista de exames com busca, filtros (tipo, data, paciente) e paginação.
- Controle de acesso: um médico nunca vê dados de outro — com testes automatizados que provem isso.
- Log de auditoria de acessos e ações; exclusão de paciente apaga os dados associados.
- Termo de uso e consentimento (simulados).

**Perguntas orientadoras**

- Autenticação pronta (Auth0, Clerk, Supabase) ou própria? Por quê?

- Quais dados o sistema realmente precisa guardar (minimização)?
- Criptografia em repouso e em trânsito: o que é viável em 3 semanas?

# NÍVEL AVANÇADO — Integrações e produto em produção

## História 8 — Extrator via WhatsApp

Como médico, quero enviar a foto ou o PDF de um exame pelo WhatsApp e receber de volta o PDF estruturado na mesma conversa, para que eu use a BiaBot sem abrir outro aplicativo, inclusive entre um atendimento e outro.

**Contexto do desafio**

O médico já vive no WhatsApp. Integrar o extrator ao canal elimina atrito, mas traz desafios novos: webhooks, download de mídia, janela de conversa de 24h, templates de mensagem aprovados, identificação do usuário e segurança da conversa. É também a ferramenta "extra" com mais chance de ser inédita para a turma.

**Critérios de aceitação**

- Integração com a WhatsApp Business Platform (Cloud API) ou com o sandbox do Twilio, recebendo mensagens via webhook.
- Receber PDF e imagem, baixar a mídia, enviar ao pipeline (reaproveitando a História 3 ou chamando o serviço diretamente) e devolver o PDF estruturado (História 6 ou um template simples).
- Mensagens de status ("recebi seu exame, estou processando") e de erro amigáveis.
- Vínculo do número de WhatsApp a uma conta (cadastro/verificação); números não cadastrados recebem orientação.
- Validação de assinatura dos webhooks, idempotência para mensagens repetidas e fila para picos.
- Demonstração ao vivo ponta a ponta em menos de 60 s por exame.

**Perguntas orientadoras**

- Cloud API da Meta, Twilio ou outro provedor: custo, burocracia e limites de cada um?
- Como lidar com várias fotos de um mesmo exame enviadas em sequência?
- Que fluxo conversacional faz sentido: menu, comandos ou linguagem livre?
- Que riscos de privacidade existem em trafegar exame pelo WhatsApp, e como mitigá-los?

## História 9 — Evolução do paciente ao longo do tempo

Como médico, quero comparar exames do mesmo paciente feitos em datas e laboratórios diferentes, para que eu veja a tendência de cada analito (ex.: glicemia, creatinina, LDL) sem montar planilha à mão.

**Contexto do desafio**

O maior valor de estruturar exames aparece quando eles podem ser comparados. O problema: laboratórios usam nomes, unidades e faixas de referência diferentes para o mesmo analito.

**Critérios de aceitação**

- Normalização de analitos (ex.: "Glicose" e "Glicemia de jejum" → mesmo analito), com tabela de equivalência documentada e conversão de unidades quando necessário.
- Linha do tempo por paciente e gráfico de evolução por analito, com a faixa de referência ao fundo.
- Destaque de mudanças relevantes entre exames consecutivos, com regra explícita e documentada.
- Funciona com pelo menos 3 exames por paciente para 5 pacientes simulados (fornecidos pela BiaBot ou gerados pela equipe a partir dos modelos).
- Exportação do gráfico/tabela para o PDF da História 6 (se feita) ou para PNG.

**Perguntas orientadoras**

- Como tratar faixas de referência diferentes entre laboratórios no mesmo gráfico?
- Normalização por regra, por tabela padronizada? Quanto de cada?
- O que é "mudança relevante" — e quem decide?

## História 10 — Infraestrutura, deploy e observabilidade

Como time técnico da BiaBot, queremos que o sistema da sprint rode em ambiente reproduzível, com deploy automatizado e monitoramento, para que possamos incorporar o trabalho à plataforma sem reescrever a infraestrutura.

**Contexto do desafio**

Protótipos de sprint costumam funcionar "só na máquina de quem fez". Esta história trata o sistema como produto: containers, CI/CD

**Critérios de aceitação**

- Pipeline de CI/CD (GitHub Actions)