

MANUAL DA EQUIPE – PROJETO ÓCULOS INTELIGENTE

1. Objetivo do Manual

Este manual define **funções, responsabilidades, regras internas, fluxos de trabalho e procedimentos padrão** para a equipe do projeto “Óculos Inteligente de Baixo Custo”.

Serve como referência para manter **organização, alinhamento e consistência** durante todo o ciclo de desenvolvimento.

2. Estrutura da Equipe e Papéis

Product Owner – Fabio Brasileiro

- Define prioridades do backlog.
- Aprova critérios de aceitação.
- Alinha requisitos com stakeholders.

Scrum Master – Heloísa Cativo

- Garante que o time siga Scrum.
- Remove impedimentos.
- Facilita reuniões e boa comunicação.

Tech Lead / Arquiteto – José Filho

- Define arquitetura do sistema.
- Aprova decisões técnicas.
- Orienta devs sobre boas práticas.

Desenvolvedores Backend – Victor Fernandes, Victor Abreu, Guilherme Reis

- Implementam lógica embarcada e APIs.
- Integram sensores, IA e BLE.
- Criam testes de unidade.

Desenvolvedores Frontend/Android – Luis Oliveira, Eric Silva, João Pereira

- Desenvolvem app Android.
- Criam telas, acessibilidade e TTS.
- Integram BLE e calibradores do sistema.

QA / Testador – Mateus Miranda

- Cria planos e casos de testes.
- Valida módulos e sprint releases.
- Registra bugs e acompanha correções.

Config Manager – José Filho

- Cuida de versionamento (Git).
- Organiza branches e releases.
- Garante integridade do repositório.

CCB – Victor Abreu

- Aprova solicitações de mudança.
- Avalia impacto técnico e de cronograma.

3. Regras Internas da Equipe

3.1 Comunicação

- Canal principal: **WhatsApp + Discord**
- Reuniões:
 - **Daily:** 15 min
 - **Revisão de Sprint:** ao final de cada sprint
 - **Retrospectiva:** após review
- Chamadas urgentes: marcar @todos no Discord.

3.2 Regras de Commit (Git)

Sempre seguir o padrão:

type: descrição curta

- o que foi feito
- motivo da mudança

Types permitidos:

feat, fix, docs, test, refactor, perf, style

Branch pattern:

- **main** → versão estável
- **develop** → versão de integração
- **feature/nome** → novas funções
- **hotfix/nome** → correções urgentes

3.3 Boas Práticas de Código

- Nome de variáveis claro e objetivo.
- Um commit por tarefa.
- Evitar código duplicado.
- Sempre documentar funções principais.
- Criar testes unitários onde possível.

4. Procedimentos Operacionais

4.1 Desenvolvimento de Hardware

- Validar sensores antes de integrar.
- Testar HC-SR04 em bancada com régua.
- Testar vibracall com PWM em três níveis.
- Testar câmera com snapshot antes da IA.
- Registrar consumo da bateria a cada iteração.

4.2 Desenvolvimento do Firmware (ESP32)

Fluxo padrão:

1. Implementar leitura de sensores.
2. Garantir tratamento de erros.
3. Enviar eventos via BLE.
4. Testar latência e estabilidade.

5. Integrar com app Android.

Checklist interno:

- Distância
- Vibração proporcional
- Captura de imagem
- Inferência TFLite
- BLE

4.3 Desenvolvimento do App Android

- Criar telas simples e acessíveis.
- Implementar TTS responsivo.
- Garantir sincronização BLE.
- Incluir calibração (thresholds e vibração).
- Feedback por voz e botões grandes.

4.4 Testes

QA deve validar:

- Funcionalidade dos sensores
- Reconhecimento de objetos
- Latência ponta-a-ponta
- Uso com 1–5 usuários reais
- Exportação de logs
- Estabilidade da conexão BLE

5. Organização do Repositório

```
/docs
  ERS/
  diagramas/

/hardware
  esp32/
  esquemáticos/

/app
  android app/

/firmware
  modules/
  ble/
  vision/

/tests
  unit/
  integration/
```

6. Solicitações de Mudança (CCB)

Qualquer alteração no sistema deve seguir:

1. Abrir issue: "Solicitação de Mudança".
2. CCB avalia impacto.
3. Decisão registrada no GitHub.
4. Implementação inicia somente após aprovação.

7. Fluxo de Sprint (Scrum)

Duração: 7 dias

Dia 1 – Planejamento

Definição do backlog da sprint.

Dias 2–6 – Execução

Desenvolvimento, testes e integração.

Dia 7 – Review + Retrospectiva

8. Regras de Qualidade

- Latência ≤ 400 ms
- Acurácia $\geq 70\%$
- Autonomia ≥ 2 horas
- BLE estável
- Vibração perceptível
- Logs funcionando

9. Checklist Final da Sprint (interno)

- Código testado
- Testes feitos pelo QA
- Documentação atualizada

- Bugs corrigidos
- Branch mergeada corretamente
- Versão registrada



10. Contatos e Papéis Rápidos

- **PO:** decisões de produto
- **SM:** processos ágeis
- **Tech Lead:** decisões técnicas
- **QA:** testes
- **Config Manager:** Git
- **CCB:** aprova mudanças