**SPRINTS BACKLOG**

**Sprint 1 (Semana 1-2) – Setup & infraestrutura básica**

**Objetivo:** Preparar o ambiente, arquitetura, e criar funcionalidades mínimas para abertura de chamados.

**Histórias/Itens:**

Como funcionário, quero poder criar uma nova conta/login no app para poder abrir chamados.

**Tarefa (Luis):** Setup autenticação (por exemplo via Firebase/Auth ou backend escolhido).

**Critérios de aceite:** usuário cadastrado, login/logout funciona.

Como funcionário, quero poder abrir um novo chamado com título e descrição para que possa reportar um problema.

**Tarefa (Gustavo):** Tela “Novo Chamado” com formulário, validações.

**Critérios:** ao enviar, chamado é salvo no backend e visível no painel “Meus Chamados”.

Estabelecer estrutura do projeto (repositório, branches, CI).

**Tarefa (Danilo):** Configuração do repositório para deployment, configurar expo/React Native + TypeScript (já mencionado) .

**Tarefa (Felipe):** Criar guideline de commit, branch naming, revisão de PR.

Criar o backend mínimo: API para “criar chamado” e “listar meus chamados”.

**Testes básicos:** criar testes unitários para função de envio de chamado.

**Documentação:** atualizar README com instruções de ambiente, rodar localmente.

**Sprint 2 (Semana 3-4) – Listagem, detalhes e fluxo de usuário técnico**

**Objetivo:** Permitir que o funcionário veja lista de chamados, e que o técnico visualize e assuma chamados.

**Histórias/Itens:**

Como funcionário, quero ver em minha app a lista de chamados abertos por mim para acompanhar o status.

**Tarefa (Caio):** Tela “Meus Chamados” com listagem, filtros (ex: status “aberto”, “em andamento”).

Como técnico, quero ver todos os chamados pendentes para que eu possa assumir um.

**Tarefa (Luis):** Backend “listar todos chamados pendentes”, app técnico com tela correspondente.

Como técnico, quero poder “assumir” um chamado (mudar status para “em andamento”) para representar que estou cuidando dele.

**Tarefa (Luis):** API “assumir chamado”, UI botão “Assumir”.

Como funcionário, quero receber notificação/in-app quando meu chamado for assumido pelo técnico para acompanhar.

**Tarefa (Gustavo):** Setup de push notifications ou ao menos notificação in-app/local.

**Testes de fluxo:** criar pull request, rodar testes, code review.

Documentar endpoints da API (ex: via Swagger ou README).

**Sprint 3 (Semana 5-6) – Integração de IA para sugestão imediata + melhorias de UI/UX**

**Objetivo:** Integrar módulo de IA para sugerir soluções e refinar interface, navegação e feedback de usuário.

**Histórias/Itens:**

Como funcionário, quero que a IA analise a descrição do chamado e sugira uma solução automática para que meu problema possa se resolver mais rápido.

**Tarefa (Endly):** Integrar serviço (por ex: OpenAI ou outro) para analisar texto e retornar sugestão.

**Critérios:** a sugestão aparece, usuário pode aceitá-la ou ignorar.

Como técnico, quero poder ver a sugestão da IA no chamado assumido para acelerar resolução.

**Tarefa (Caio):** Mostrar sugestão na tela de “detalhe do chamado”.

Melhorias de UI/UX: Tela splash, navegação entre telas, menu lateral ou tab bar.

**Tarefa (Felipe):** Definir design baseline (cores, tipografia) e aplicar nas telas implementadas.

**Validação de entrada:** descrever corretamente campos, mostrar mensagens de erro, feedback visual.

**Documentação de módulo IA:** descrever como funciona, dependências, limites.

**Testes de integração:** simular envio de chamado + sugestão IA + feedback do usuário.

**Sprint 4 (Semana 7-8) – Resolução de chamado, histórico, relatórios e integração web/desktop**

**Objetivo:** Permitir a resolução completa do chamado, geração de histórico e acesso via web ou desktop.

**Histórias/Itens:**

Como técnico, quero encerrar o chamado (marcar como “resolvido”) e adicionar comentário final para que o funcionário veja que foi atendido.

**Tarefa (Danilo):** API “fechar chamado”, UI “Encerrar”.

Como funcionário, quero acessar o histórico dos meus chamados (abertos, em andamento, resolvidos) para consulta.

**Tarefa (Felipe):** Tela “Histórico de chamados”.

Como gestor, quero relatório simples (ex: número de chamados por status, tempo médio de resolução) para acompanhar desempenho.

**Tarefa (Caio):** Endpoint de relatório + UI web ou dashboard simples.

Criar versão web ou desktop (por ex: via React web) para gestor/ técnico operarem em computador.

**Tarefa (Luis):** Setup projeto web minimal, login, listagem de chamados.

**Testes:** cobrir fluxo completo, UI responsiva.

**Documentação:** manual do sistema atualizado com novo módulo histórico/relatório.

**Sprint 5 (Semana 9-10) – Polimento, segurança, deploy e melhorias finais**

**Objetivo:** Refino final, segurança, deploy para produção, feedback dos usuários e monitoramento.

**Histórias/Itens:**

Como usuário (qualquer perfil), quero autenticação segura (por exemplo JWT, refresh token) para garantir confidencialidade.

**Tarefa (Luis):** Revisar autenticação, aplicar best-practices, criptografia etc.

Como administrador, quero configurar roles/permissões (ex: funcionário, técnico, gestor) para controlar acessos.

**Tarefa (Gustavo):** Definição de roles no backend + UI para atribuição.

Como equipe de TI, quero deploy em ambiente de produção (por ex: AWS, Azure, Heroku) para disponibilizar o app real.

**Tarefa (Felipe):** Pipeline CI/CD, configuração ambiente, docs de deploy.

Como equipe de produto, quero coletar feedback de usuários piloto e aplicar melhorias (bugs, UX, performance).

**Tarefa (Danilo):** Definir checklist de métricas, corrigir bugs prioritários.

**Monitoramento:** configurar logs, métricas de uso, crash-reporting, para acompanhar estabilidade.

Documentar “release” final: atualizar README, gravar vídeo/demo, publicar no store (se mobile) ou release web.

**Observações finais**

Antes de cada sprint, façam refinamento do backlog: detalhar histórias, estimar, definir prioridades.

Garantir que cada história esteja pronta para ser trabalhada (DoR) antes de entrar no sprint e que quando estiver concluída cumpra a DoD.

Após cada sprint, façam retrospectiva e demo para stakeholders.