# BSK Ops — Plano de Projeto (3–5 semanas)

**Objetivo**: Entregar uma PWA mobile‑first para gestão de tarefas operativas com vistas **Hoje** e **Esta semana**, **recorrências** (semanal/bisemanal), **offline‑first** (IndexedDB + outbox), **fotos** e **Fechar dia/Resumos**. **Ritmo**: 3–4h/dia, 5 dias/semana (ajusta conforme necessário). **Datas sugeridas** (exemplo): Semana 1 = 1–7 Set, S2 = 8–14 Set, S3 = 15–21 Set, S4 = 22–28 Set, S5 (buffer) = 29 Set–5 Out.

## Roadmap em 15 passos (elaborado)

Cada passo inclui **Objetivo**, **Tarefas**, **DoD** (Definition of Done) e **Estimativa**.

### 1) Fundações do projeto

**Objetivo**: Base sólida de tooling/código. **Tarefas**: - Vite + React + TypeScript + Tailwind + React Router - ESLint + Prettier + tsconfig estrito (noImplicitAny, strict) - Commitlint + Husky (pre‑commit: lint & typecheck) - README inicial + scripts npm **DoD**: npm run lint e tsc --noEmit a verde; projeto arranca localmente. **Estimativa**: 4–6h

### 2) PWA (base)

**Objetivo**: App instalável e shell em cache. **Tarefas**: - Manifest (name, short\_name, icons 192/512, display=standalone, orientation=portrait) - Service Worker (Workbox) com **stale‑while‑revalidate** para app‑shell - Página “Instalar app” + banner sugestão (A2HS) - UX de update: detetar SW novo, botão “Atualizar” (skipWaiting/clientsClaim) **DoD**: Lighthouse PWA ≥ 95; instalável num telemóvel. **Estimativa**: 4–6h

### 3) Modelação & dados (offline‑first)

**Objetivo**: Schema e persistência local robusta. **Tarefas**: - Definir tipos (Task, ChecklistTemplate, ChecklistRun, Summary, Attachment) - IndexedDB com **Dexie**: tabelas, índices (when, status, updatedAt), migrações v1 - kv (key‑value) para flags de UI (ex.: viu\_banner) **DoD**: CRUD local com Dexie; dados persistem após refresh. **Estimativa**: 6–8h

### 4) UI Shell & Navegação

**Objetivo**: Estrutura visual e navegação mobile. **Tarefas**: - AppBar + **menu hambúrguer (☰)** com drawer - **Home** (cards: Hoje, Semana, Checklists, Resumos, Bloqueios, Recorrências de hoje) - Tabs/Bottom bar **Hoje/Esta semana**; indicador **Online/Offline** **DoD**: Navegação fluida; links do menu funcionam. **Estimativa**: 4–6h

### 5) Tarefas — Hoje (MVP)

**Objetivo**: Operação diária funcional. **Tarefas**: - Lista com **pesquisa** e **chips** (Com hora / Bloqueadas / Com anexos / Só pendentes) - Ações rápidas: **✓ Feito**, **⏸ Bloquear** (motivo), **➜ Semana**, **Eliminar**, **📎 Foto (stub)** - **Marcar tudo feito** + **Fechar dia** (diálogo) - Empty state com CTA “Selecionar da Semana” **DoD**: Fluxo “abrir → marcar → fechar dia” completo localmente. **Estimativa**: 6–10h

### 6) Tarefas — Esta semana (MVP)

**Objetivo**: Planeamento semanal simples. **Tarefas**: - Lista com pesquisa e filtros: **Categoria**, **Dia**, **↻ Recorrentes**, **Transitados**, **Estado** - Seleção múltipla → **Enviar para Hoje**; ação rápida **[➜ Hoje]** **DoD**: Check‑in da manhã em 2–3 toques. **Estimativa**: 6–8h

### 7) Recorrências & seed

**Objetivo**: Geração automática das tarefas recorrentes. **Tarefas**: - Padrões: **Semanal** (Seg–Dom) e **Bisemanal** (pares/ímpares) - **Seed automático** à **segunda‑feira** (gera ocorrências desta semana) - Badge **Transitado** para carry‑over **DoD**: Recorrentes aparecem no dia correto; marcar “Feita” remove a ocorrência atual. **Estimativa**: 6–8h

### 8) Fechar dia & Resumos

**Objetivo**: Passagem de turno clara. **Tarefas**: - Diálogo: contagem **Feitas / Bloqueadas / Por fazer**; notas do dia - Mover pendentes para “Esta semana”; guardar **Summary** (refs de anexos) - Ecrã **Resumos** (lista + detalhe) **DoD**: Resumo guardado e visível; arranque seguinte mostra briefing (opcional). **Estimativa**: 6–8h

### 9) Anexos (fotos)

**Objetivo**: Captura de evidências no telemóvel. **Tarefas**: - <input capture="environment" accept="image/\*"> - Compressão leve; guardar **Blob** em IndexedDB; preview no cartão - Marcar para envio posterior (link com outbox) **DoD**: Foto anexada offline e persiste após restart. **Estimativa**: 6–8h

### 10) Sync com backend (padrão Outbox)

**Objetivo**: Fonte de verdade no servidor sem perder offline. **Tarefas**: - **Outbox** (create/update/delete) com updatedAt e origin (clientId) - **Push** quando online (retry/backoff); **Pull** incremental (/changes?since) - Conflitos: **last‑write‑wins** + aviso - Upload de anexos → URL final **DoD**: Desligar/ligar rede sem perder alterações; servidor reflete mudanças. **Estimativa**: 10–14h (depende do backend)

### 11) Checklists (Abertura/Fecho/…)

**Objetivo**: Rotinas rápidas e padronizadas. **Tarefas**: - **Templates** (Abertura, Fecho, Limpeza, Receção) - **Run** do dia com progresso (tick único + nota/foto por item) **DoD**: Completar Abertura/Fecho num turno típico. **Estimativa**: 6–8h

### 12) Ajuda & Informação

**Objetivo**: Onboarding e procedimentos consultáveis offline. **Tarefas**: - Guia rápido (Hoje/Semana/Fechar dia/Recorrências) - Procedimentos internos; contactos (tel/mailto) **DoD**: Página concisa disponível offline. **Estimativa**: 3–4h

### 13) Acessibilidade & Performance

**Objetivo**: UX sólida para todos e rápida. **Tarefas**: - Focus rings, labels/ARIA, contraste - Perf budget: FCP < 1.5s (mid‑range), JS shell < 200KB gzip **DoD**: Lighthouse A11y ≥ 95; métricas dentro do budget. **Estimativa**: 3–5h

### 14) Testes & Qualidade

**Objetivo**: Confiança nas principais rotas. **Tarefas**: - Jest + RTL: utilitários (filtros/seed), reducers, componentes críticos - Cypress: “Hoje → Fechar dia”, “Recorrentes → Hoje”, **offline** (modo avião) - GitHub Actions: lint + typecheck + testes **DoD**: CI a verde; 2 cenários E2E a passar. **Estimativa**: 6–10h

### 15) Deploy

**Objetivo**: App pública, com updates fiáveis. **Tarefas**: - Vercel/Netlify (HTTPS), cabeçalhos PWA, preview deploys - Página “Estado & Sync” (debug: fila, espaço IndexedDB) **DoD**: Instalável num telemóvel real; update funciona. **Estimativa**: 3–5h

**Total estimado**: 56–82h → **3–5 semanas** a 3–4h/dia.

## Timeline sugerida (por semanas)

### Semana 1 — Base + PWA + Shell (≈ 12–18h)

* Passos **1, 2, 4**.
* Entrega: navegação completa (Home, Hoje, Semana, Drawer, indicador online) + instalável.

### Semana 2 — Tarefas (Hoje & Semana) (≈ 12–18h)

* Passos **5, 6**.
* Entrega: fluxo diário completo local (sem sync), fechar dia básico.

### Semana 3 — Recorrências + Resumos + Fotos (≈ 18–24h)

* Passos **7, 8, 9**.
* Entrega: seed semanal, resumos guardados, anexos offline.

### Semana 4 — Sync + Qualidade + Deploy (≈ 14–22h)

* Passos **10, 11, 12, 13, 14, 15** (foco no 10; os restantes em paralelo conforme tempo).
* Entrega: sync com backend (mínimo viável), checklists, testes, deploy.

**Buffer**: usa a Semana 5 para refinamentos, bugs e polimento.

## Critérios transversais (Definition of Ready/Done)

* **Ready**: user story clara, critérios de aceitação, mocks/wireframes definidos, impacto no SW/IndexedDB considerado.
* **Done**: código tipado, lint a verde, testes mínimos, Lighthouse ≥ 95 (PWA/A11y), documentação no README/CHANGELOG.

## Riscos & mitigação

* **Complexidade de sync** → começar offline‑first e ligar sync só no Passo 10; logs e “Estado & Sync” para debug.
* **SW a servir versão antiga** → rotina de update com skipWaiting + aviso ao utilizador.
* **IndexedDB espaço/limites** → expirar anexos antigos; quotas visíveis em “Estado & Sync”.
* **Scope creep** → manter foco no **MVP**; extras ficam no backlog pós‑v1.

## Checklist de arranque (copiar para issues)

* Repo criado com CI (lint/typecheck/test)
* Paleta & ícone (manifest)
* Wireframe validado (Home/Hoje/Semana/Fechar dia)
* Esquema Dexie v1 definido
* Plano de testes (2 E2E) definido

## Próximos passos

1. Confirmar esta timeline e estimativas.
2. Abrir issues para **Semana 1** (Passos 1,2,4) com critérios de aceitação.
3. Iniciar implementação e registar progresso diário (3–4h).