JSON | MANIFEST

{

  "name": "APPFinance",

  "short\_name": "APPFinance",

  "start\_url": ".",

  "display": "standalone",

  "background\_color": "#ffffff",

  "theme\_color": "#4CAF50",

  "description": "Gestão de portfólio e simulações de investimentos",

  "icons": [

    {

      "src": "icons/icon-192.png",

      "sizes": "192x192",

      "type": "image/png"

    },

    {

      "src": "icons/icon-512.png",

      "sizes": "512x512",

      "type": "image/png"

    }

  ]

}

JS | SERVICE-WORKER

const CACHE\_NAME = "appfinance-cache-v1";

const ASSETS = [

  "/",

  "/index.html",

  "/style.css",

  "/js/main.js",

  "/manifest.json",

  "/icons/icon-192.png",

  "/icons/icon-512.png"

];

// Instala e guarda ficheiros no cache

self.addEventListener("install", event => {

  event.waitUntil(

    caches.open(CACHE\_NAME).then(cache => {

      return cache.addAll(ASSETS);

    })

  );

});

// Ativa e limpa caches antigos

self.addEventListener("activate", event => {

  event.waitUntil(

    caches.keys().then(keys => {

      return Promise.all(keys.filter(k => k !== CACHE\_NAME).map(k => caches.delete(k)));

    })

  );

});

// Responde com cache ou vai buscar à rede

//self.addEventListener("fetch", event => {

  //event.respondWith(

    //caches.match(event.request).then(response => {

     // return response || fetch(event.request);

    //})

 // );

//});

self.addEventListener("install", (event) => {

  event.waitUntil(

    (async () => {

      const cache = await caches.open("app-cache-v1");

      const urls = [

        "/",

        "/index.html",

        "/style.css",

        "/js/main.js",

        "/screens/atividade.html",

        "/screens/dashboard.html", // etc...

        // evita listar módulos dinâmicos se não tiveres a certeza

      ];

      await Promise.all(

        urls.map(async (url) => {

          try {

            const resp = await fetch(url, { cache: "no-cache" });

            if (!resp.ok) throw new Error(`${url} -> ${resp.status}`);

            await cache.put(url, resp);

          } catch (e) {

            console.warn("[SW] skip cache:", e.message);

          }

        })

      );

    })()

  );

});