Goal

Build a React **SplashSequence** that shows **F → M → F → B** (images or fallback letters) for ~5s, then reveals your main page — **in Laravel 9 + Vite**. Below are the exact files, commands, and the fixes for every issue you hit.

1) Prereqs (Windows)

- **Node/npm** installed (you have them).
- **PowerShell policy** (fix once so VS Code terminal works):

```
Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned –Force
```

(Alternative per-session: `Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass` — or use CMD instead of PowerShell.)

2) Install React + Vite React plugin

```
npm install react react-dom

npm install -D @vitejs/plugin-react
```

Avoid `npm audit fix --force` during setup (it jumps to breaking majors). If you already ran it but Vite still runs, you're fine.

3) Vite config

`vite.config.js` (project root)

```
import { defineConfig } from 'vite'

import laravel from 'laravel-vite-plugin'

import react from '@vitejs/plugin-react'



export default defineConfig({

  plugins: [
    laravel({
      input: ['resources/js/app.jsx'], // must match your real file
      refresh: true,
    }),
    react(),
  ],
})
```

Common problem we hit:

Failed to resolve rollupOptions.input … `resources/js/app.jsx`” → The file didn’t exist or path didn’t match. Create it and keep Blade path identical (next step).

4) Blade (load Vite & mount React)

`resources/views/welcome.blade.php` — inside `<head>`:

```
@viteReactRefresh
@vite('resources/js/app.jsx')
```

At the end of `<body>` (or anywhere in body):

```
<div id="root"></div>
```

Fixes we applied:

- **Preamble error** (“@vitejs/plugin-react can’t detect preamble”) → add `@viteReactRefresh` and ensure one valid `<head>...</head>`.
- **Bootstrap URL**: use the correct CDN (no `%40`):

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js" defer></script>
```

5) Project structure

```
resources/js/
 ├─ app.jsx        (entry — creates root & renders MainApp)
 ├─ mainApp.jsx      (your main component: shows Splash, then page)
 └─ components/
    └─ SplashSequence.jsx
public/images/
 ├─ F1.png
 ├─ M.png
 ├─ F2.png
 └─ B.png
```

On Windows you can't have both `App.jsx` and `app.jsx`. We use `mainApp.jsx` for the component and `app.jsx` for the entry.

6) Entry file

resources/js/app.jsx

```
import React from "react";
import { createRoot } from "react-dom/client";
import MainApp from "./mainApp";


const el = document.getElementById("root");
if (el) {
  createRoot(el).render(
    <React.StrictMode>
      <MainApp />  {/* Must be capitalized */}
    </React.StrictMode>
  );
}
```

Bug we fixed: `<mainApp />` (lowercase) won't render your component. Use `<MainApp />`.

7) Main component

resources/js/mainApp.jsx

```
import React, { useState } from "react";

import SplashSequence from "./components/SplashSequence";


function MainPage() {

  return (

    <div style={{ padding: 24 }}>

      <h1>FMFB — Main Dashboard</h1>

      <p>Welcome! The splash is done.</p>

    </div>

  );

}


export default function MainApp() {

  const [ready, setReady] = useState(false);

  return (

    <>

      {!ready && <SplashSequence onDone={() => setReady(true)} />}

      {ready && <MainPage />}

    </>

  );

}
```

8) Splash component (final, robust)

resources/js/components/SplashSequence.jsx

```
import React, { useEffect, useRef, useState, useMemo } from "react";


export default function SplashSequence({ onDone }) {
 // UseMemo so the reference doesn't change and re-trigger effects
 const images = useMemo(() => [
  "/images/F1.png",
  "/images/M.png",
  "/images/F2.png",
  "/images/B.png",
 ], []);

 const letters = ["F","M","F","B"]; // fallback while images load

 const [step, setStep] = useState(0);
 const [imgReady, setImgReady] = useState(false);
 const timerRef = useRef(null);

 // Preload once
 useEffect(() => {
  images.forEach(src => { const i = new Image(); i.src = src; });
 }, [images]);

 // Drive the sequence: ~5s total
 useEffect(() => {
  setStep(0);
  setImgReady(false);
  let i = 0;

  timerRef.current = setInterval(() => {
    i += 1;
    if (i < images.length) {
      setImgReady(false);
      setStep(i);
    } else {
      clearInterval(timerRef.current);
      onDone?.();
    }
  }, 1250);

  return () => { if (timerRef.current) clearInterval(timerRef.current); };
}, [onDone]); // don't depend on `images` unless memoized (we memoized it)

// Hide when finished
if (step >= images.length) return null;

return (
  <div style={{
    position: "fixed",
    inset: 0,
    display: "grid",
    placeItems: "center",
    background: "#341539",
    color: "#fff",
    zIndex: 9999
  }}>
    {/* Fallback letter so it's never blank while the image loads */}
    {!imgReady && (
      <div style={{ fontSize: "min(20vw,160px)", fontWeight: 800, opacity: .25 }}>
       {letters[step]}
      </div>
    )}

    <img
      src={images[step]}
      alt={`splash-${step}`}
      onLoad={() => setImgReady(true)}
      onError={() => console.warn("Image failed:", images[step])}
      style={{
       width: "min(60vw,420px)",
       height: "auto",
       objectFit: "contain",
       display: imgReady ? "block" : "none",
       userSelect: "none",
      }}
```

```
    />
  </div>
 );
}
```

Bugs we solved here:

- **Stuck on "F"** → effect was re-triggering because `images` (a new array each render) was in deps. We **memoized** `images` and removed it from the sequencer effect's deps.
- **Blank purple overlay** → added **fallback letters** and **preloading** so it's never visually empty, even on slow loads.

(StrictMode note: if dev double-mount bothers you, you can temporarily remove `<React.StrictMode>` in `app.jsx` to test.)

9) Run the app

Two terminals (recommended while developing):Zahra->(powershell)

```
npm run dev       # Vite dev server (hot reload)

php artisan serve  # Laravel at http://127.0.0.1:8000
```

"Vite manifest not found" in dev? You probably aren't running Vite, or `public/hot` is stale:

```
del public\hot

php artisan config:clear

php artisan cache:clear

php artisan view:clear

npm run dev
```

Production (no dev server):

```
npm run build

php artisan serve
```

10) Troubleshooting (everything you faced)

Symptom	Cause	Fix
`npm is not recognized`	Node/npm not in PATH or wrong shell	Reinstall Node; open new terminal
`npm.ps1 cannot be loaded`	PowerShell execution policy blocks npm	`Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned -Force` (or use CMD)
Vite: **Failed to resolve input**	Entry file path mismatch	Ensure `vite.config.js` and Blade both point to `resources/js/app.jsx` and file exists
React preamble can't be detected	Missing preamble or malformed `<head>`	Add `@viteReactRefresh` above `@vite(...)`; ensure one `<head>` pair
Bootstrap **nosniff**	URL had `%40` encoded `@`	Use `bootstrap@5.3.0` correct URL shown above
Splash never shows	`<mainApp />` lowercase tag	Use `<MainApp />`
Stuck on "F"	Effect restarts; `images` in deps	Memoize `images` (`useMemo`) and remove from sequencer deps
Blank overlay	Images slow/404	Put files in `public/images`, use leading `/`, add fallback letters, check Network 200 OK
Manifest not found	Running Laravel without Vite dev server	Use dev mode (`npm run dev`) or build (`npm run build`)