

This is a step-by-step guide to splitting your large `shop-admin/src/App.jsx` file into a clean, standard React folder structure.

Your goal is to change your folder structure from this:

```
shop-admin/src/  
├─ App.jsx    (Everything is in here)  
├─ main.jsx  
├─ firebaseConfig.js  
└─ index.css
```

...to this:

```
shop-admin/src/  
├─ components/  
│   ├── AdminLayout.jsx  
│   ├── ProductModal.jsx  
│   └── ProtectedRoute.jsx  
├─ pages/  
│   ├── DashboardPage.jsx  
│   ├── InventoryPage.jsx  
│   └── LoginPage.jsx  
├─ App.jsx          (Will become your main router)  
├─ main.jsx         (No changes needed)  
├─ firebaseConfig.js (No changes needed)  
└─ index.css        (No changes needed)
```

Step 1: Create Your New Folders

In your `shop-admin/src` directory, create two new folders:

1. `components`
2. `pages`

Step 2: Create `pages/LoginPage.jsx`

1. Create a new file at `shop-admin/src/pages/LoginPage.jsx`.
2. Go to your original `App.jsx` file.
3. **Cut** the *entire* `function LoginPage() { ... }` (lines 80-143 in your file) and **paste** it into `LoginPage.jsx`.
4. At the very top of `LoginPage.jsx`, add these imports:

```
import React, { useState } from 'react';  
import { useNavigate } from 'react-router-dom';
```

```
import { db, auth } from '../firebaseConfig.js';
import { signInWithEmailAndPassword, signOut } from 'firebase/auth';
import { doc, getDoc } from 'firebase/firestore';
```

5. On the very first line of the file (before the function), add `export default`:

```
export default function LoginPage() {
  // ... all your code
}
```

Step 3: Create `components/ProtectedRoute.jsx`

1. Create a new file at `shop-admin/src/components/ProtectedRoute.jsx`.
2. **Cut** the `function ProtectedRoute({ ... }) { ... }` (lines 145-151) and **paste** it into the new file.
3. Add these imports at the top:

```
import React from 'react';
import { Navigate, useLocation } from 'react-router-dom';
```

4. Add `export default` to the function:

```
export default function ProtectedRoute({ user, isAdmin, children }) {
  // ... all your code
}
```

Step 4: Create `components/AdminLayout.jsx`

1. Create a new file at `shop-admin/src/components/AdminLayout.jsx`.
2. **Cut** the `function AdminLayout() { ... }` (lines 153-200) and **paste** it in.
3. Add these imports at the top:

```
import React from 'react';
import { Routes, Route, Link, useNavigate } from 'react-router-dom';
import { auth } from '../firebaseConfig.js';
import { signOut } from 'firebase/auth';
import { InboxStackIcon, CreditCardIcon, ArrowLeftOnRectangleIcon } from '@heroicons';
import DashboardPage from '../pages/DashboardPage.jsx';
import InventoryPage from '../pages/InventoryPage.jsx';
```

4. Add `export default` to the function.

Step 5: Create `pages/DashboardPage.jsx`

1. Create a new file at `shop-admin/src/pages/DashboardPage.jsx`.
2. **Cut** the `function DashboardPage() { ... }` (lines 205-321) and **paste** it in.
3. Add these imports at the top:

```
import React, { useState, useEffect } from 'react';
import { db } from '../firebaseConfig.js';
import {
  collection, doc, onSnapshot, updateDoc,
  query, orderBy, runTransaction, getDoc
} from 'firebase/firestore';
```

4. Add `export default` to the function.

Step 6: Create `components/ProductModal.jsx`

1. Create a new file at `shop-admin/src/components/ProductModal.jsx`.
2. **Cut** the `function ProductModal({ ... }) { ... }` (lines 434-526) and **paste** it in.
3. Add these imports at the top:

```
import React, { useState } from 'react';
import { db } from '../firebaseConfig.js';
import {
  doc, updateDoc, addDoc, collection, serverTimestamp
} from 'firebase/firestore';
import { XMarkIcon } from '@heroicons/react/24/outline';
```

4. Add `export default` to the function. (Note: Your file has a copy-paste error where `LoginPage`'s code is inside this modal. You'll need to delete that extra code from lines 528-562.)

Step 7: Create `pages/InventoryPage.jsx`

1. Create a new file at `shop-admin/src/pages/InventoryPage.jsx`.
2. **Cut** the `function InventoryPage() { ... }` (lines 323-432) and **paste** it in.
3. Add these imports at the top:

```
import React, { useState, useEffect } from 'react';
import { db } from '../firebaseConfig.js';
import {
  collection, query, orderBy, onSnapshot, doc, deleteDoc
} from 'firebase/firestore';
import { PencilIcon, TrashIcon, PlusIcon } from '@heroicons/react/24/outline';
```

```
import ProductModal from '../components/ProductModal.jsx';
```

4. Add `export default` to the function.

Step 8: Clean Up Your `shop-admin/src/App.jsx`

After cutting all those functions out, your `App.jsx` file is almost empty. **Replace its entire contents** with this new, clean router setup. This is the only code you need to copy.

```
import React, { useState, useEffect } from 'react';
import { Routes, Route, Navigate } from 'react-router-dom';
import { db, auth } from './firebaseConfig.js';
import { doc, getDoc } from 'firebase/firestore';
import { onAuthStateChanged, signOut } from 'firebase/auth';

// --- Our New Imports ---
import LoginPage from './pages/LoginPage.jsx';
import AdminLayout from './components/AdminLayout.jsx';
import ProtectedRoute from './components/ProtectedRoute.jsx';

// =====
// 0. Main App Component (Router)
// =====
export default function App() {
  const [user, setUser] = useState(null);
  const [loadingAuth, setLoadingAuth] = useState(true);
  const [isAdmin, setIsAdmin] = useState(false);

  useEffect(() => {
    const unsubscribe = onAuthStateChanged(auth, async (currentUser) => {
      if (currentUser) {
        setUser(currentUser);
        try {
          const userDocRef = doc(db, 'users', currentUser.uid);
          const userDoc = await getDoc(userDocRef);
          if (userDoc.exists() && userDoc.data().isAdmin === true) {
            setIsAdmin(true);
          } else {
            setIsAdmin(false);
            signOut(auth);
          }
        } catch (error) {
          console.error("Error checking admin status:", error);
          setIsAdmin(false);
          signOut(auth);
        }
      } else {
        setUser(null);
        setIsAdmin(false);
      }
      setLoadingAuth(false);
    });
    return () => unsubscribe();
  }, []);
```

```
if (loadingAuth) {
  return (
    <div className="flex items-center justify-center min-h-screen bg-gray-100">
      <div className="w-12 h-12 border-4 border-dashed rounded-full animate-spin border-gray-300">
      </div>
    </div>
  );
}

return (
  <Routes>
    <Route path="/login" element={<LoginPage />} />
    <Route
      path="/*"
      element={
        <ProtectedRoute user={user} isAdmin={isAdmin}>
          <AdminLayout />
        </ProtectedRoute>
      }
    />
  </Routes>
);
}
```

Step 9: Final Step

You're done! Run `npm run dev` in your `shop-admin` folder. Your app will work exactly as before, but now your files are clean, separate, and easy to manage.