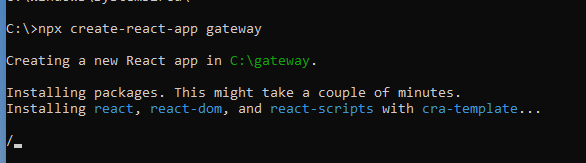
**LAB 3. 1. CREATING AN E-COMMERCE WEBSITE AND PAYMENT GATEWAY**

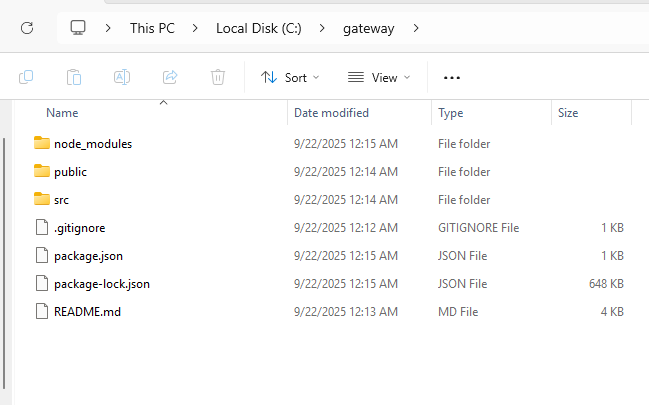
### ****Bước 1: Khởi tạo giao diện người dùng****

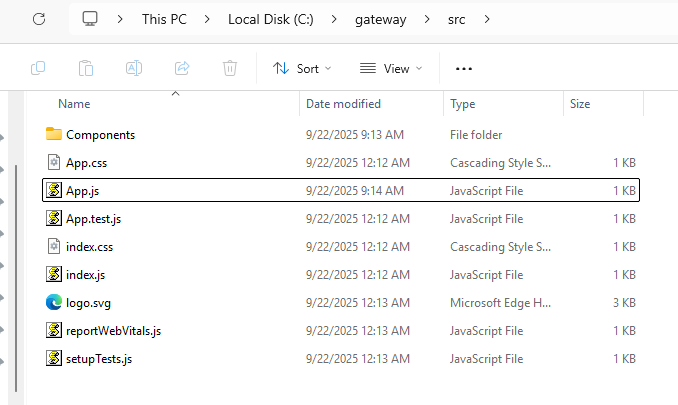
**1. Create the React app:**

npx create-react-app gateway



After installing, the project folder gateway will appear:

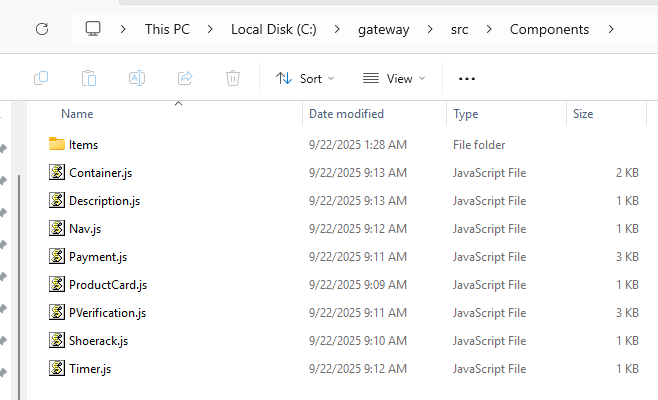




**2. Create the UI components:**

Below is a compact, practical list of the **UI component files** you should create for the gateway app (placed under src/Components), with a one-line description for each.

This is the minimal, runnable set that implements: **Browse → Buy Now → Payment → Verify** flow.



Place these files in src/Components/:

* Nav.js: top navigation bar (app title / simple header).
* Description.js: short site description/slogan under the nav.
* Container.js: page controller (keeps page state: shop | payment | verify and routes between components).
* Shoerack.js shop page: maps data array to product cards.
* ProductCard.js — single product card UI (image, name, price, **Buy Now** button).
* Items/all.js — product data array (5 shoe items; images can be placeholder URLs).
* Payment.js — payment screen: shows receive address, amount, QR/code, input for txHash and "Verify" button.
* PVerification.js — verification screen: polls confirmations (mock or real) and shows final result.
* Timer.js — countdown timer (used by verification page).

Other supporting files (project root or other folders):

* src/App.js — app root renders <Container />.
* src/index.js — app entry (already present in CRA).

src/

└── Components/

└──Items/

├── Shoes1.js

├── Shoes2.js

├── Shoes3.js

├── Shoes4.js

├── Shoes5.js

└── all.js

src/Components/Items/all.js:

|  |
| --- |
| // src/Components/Items/all.js  const Items = [  {  id: 1,  name: "Air Runner",  price: "0.05 ETH",  img: "https://via.placeholder.com/200x140?text=Air+Runner",  desc: "Lightweight running shoe."  },  {  id: 2,  name: "City Walker",  price: "0.04 ETH",  img: "https://via.placeholder.com/200x140?text=City+Walker",  desc: "Comfortable daily walker."  },  {  id: 3,  name: "Trail Pro",  price: "0.06 ETH",  img: "https://via.placeholder.com/200x140?text=Trail+Pro",  desc: "Durable trail shoe."  },  {  id: 4,  name: "Street Classic",  price: "0.03 ETH",  img: "https://via.placeholder.com/200x140?text=Street+Classic",  desc: "Casual classic style."  },  {  id: 5,  name: "Sport Elite",  price: "0.08 ETH",  img: "https://via.placeholder.com/200x140?text=Sport+Elite",  desc: "High performance shoe."  }  ];  export default Items; |

src/Components/Items/Shoes1.js:

|  |
| --- |
| // src/Items/Shoes1.js  export default {  price: 200,  name: "Badidas",  logo: "Badidas.png",  image: "Shoe1.jpg"  }; |

src/Components/Items/Shoes2.js:

|  |
| --- |
| // Shoes2.js  export default {  price: 220,  name: "Pumani",  logo: "Pumani.png",  image: "Shoe2.jpg"  }; |

src/Components/Items/Shoes3.js:

|  |
| --- |
| // Shoes3.js  export default {  price: 240,  name: "Nikey",  logo: "Nikey.png",  image: "Shoe3.jpg"  }; |

src/Components/Items/Shoes4.js:

|  |
| --- |
| // Shoes4.js  export default {  price: 180,  name: "Convars",  logo: "Convars.png",  image: "Shoe4.jpg"  }; |

src/Components/Items/Shoes5.js:

|  |
| --- |
| // Shoes5.js  export default {  price: 210,  name: "Rebax",  logo: "Rebax.png",  image: "Shoe5.jpg"  }; |

src/Components/ProductCard.js:

|  |
| --- |
| // src/Components/ProductCard.js  import React from "react";  export default function ProductCard({ item, onBuy }) {  return (  <div style={cardStyle}>  <img src={item.img} alt={item.name} style={{ width: "100%", height: 120, objectFit: "cover" }} />  <h3 style={{ margin: "8px 0 4px" }}>{item.name}</h3>  <div style={{ color: "#666", fontSize: 14 }}>{item.desc}</div>  <div style={{ marginTop: 8, display: "flex", justifyContent: "space-between", alignItems: "center" }}>  <strong>{item.price}</strong>  <button onClick={() => onBuy(item)} style={btnStyle}>Buy Now</button>  </div>  </div>  );  }  const cardStyle = {  border: "1px solid #e6e6e6",  borderRadius: 8,  padding: 12,  width: 220,  boxSizing: "border-box",  background: "#fff"  };  const btnStyle = {  padding: "6px 10px",  background: "#0b79f7",  color: "#fff",  border: "none",  borderRadius: 6,  cursor: "pointer"  }; |

src/Components/Shoerack.js:

|  |
| --- |
| // src/Components/Shoerack.js  import React from "react";  import Items from "./Items/all";  import ProductCard from "./ProductCard";  export default function Shoerack({ onBuy }) {  return (  <div>  <h2>Shop</h2>  <div style={{ display: "flex", gap: 12, flexWrap: "wrap" }}>  {Items.map(it => (  <ProductCard key={it.id} item={it} onBuy={onBuy} />  ))}  </div>  </div>  );  } |

src/Components/Payment.js:

|  |
| --- |
| // src/Components/Payment.js  import React, { useState } from "react";  export default function Payment({ product, onProceedToVerify, onCancel }) {  // mock invoice: orderId + receiveAddress + expectedAmount  const [order] = useState({  orderId: `ORD-${Date.now() % 100000}`,  address: "0xDEMOmerchantAddress000000000000000000",  amount: product?.price || "0.00 ETH",  expiresAt: Date.now() + 15 \* 60 \* 1000  });  const [txHash, setTxHash] = useState("");  return (  <div>  <h2>Payment</h2>  <div style={{ display: "flex", gap: 20 }}>  <div style={{ minWidth: 260 }}>  <div style={panel}>  <div><strong>Product</strong></div>  <div style={{ marginTop: 8 }}>{product?.name}</div>  <div style={{ color: "#666", marginTop: 4 }}>{product?.desc}</div>  <div style={{ marginTop: 8 }}><strong>Amount:</strong> {order.amount}</div>  <div style={{ marginTop: 8 }}><strong>Receive address:</strong>  <div style={{ background: "#fafafa", padding: 8, borderRadius: 4, marginTop: 6 }}>  <code style={{ fontSize: 12 }}>{order.address}</code>  </div>  </div>  </div>  </div>  <div style={{ flex: 1 }}>  <div style={panel}>  <div><strong>Pay instructions</strong></div>  <ol style={{ marginTop: 8 }}>  <li>Send the exact amount to the receive address.</li>  <li>Copy txHash and paste into the box below (if you used another wallet).</li>  <li>Click "Verify payment".</li>  </ol>  <div style={{ marginTop: 10 }}>  <input  placeholder="Paste txHash (e.g. 0x...)"  value={txHash}  onChange={(e) => setTxHash(e.target.value)}  style={{ width: "100%", padding: 8, boxSizing: "border-box" }}  />  </div>  <div style={{ marginTop: 10, display: "flex", gap: 8 }}>  <button onClick={() => onProceedToVerify({ orderId: order.orderId, txHash })} style={okBtn}>Verify payment</button>  <button onClick={onCancel} style={cancelBtn}>Cancel</button>  </div>  </div>  </div>  </div>  </div>  );  }  const panel = { border: "1px solid #eee", padding: 12, borderRadius: 8, background: "#fff" };  const okBtn = { padding: "8px 12px", background: "#0b79f7", color: "#fff", border: "none", borderRadius: 6 };  const cancelBtn = { padding: "8px 12px", borderRadius: 6 }; |

src/Components/PVerification.js:

|  |
| --- |
| // src/Components/PVerification.js  import React, { useEffect, useState } from "react";  import Timer from "./Timer";  /\*\*  \* This is a mock verification component.  \* It simulates confirmations: if txHash non-empty, increase confirmations every 2s.  \*/  export default function PVerification({ order, txHash, onBackToShop }) {  const [confirmations, setConfirmations] = useState(0);  const required = 3; // mock small number to demo  useEffect(() => {  if (!txHash) return;  setConfirmations(0);  const iv = setInterval(() => {  setConfirmations(c => {  const nc = c + 1;  if (nc >= required) {  clearInterval(iv);  }  return nc;  });  }, 2000);  return () => clearInterval(iv);  }, [txHash]);  return (  <div>  <h2>Payment Verification</h2>  <div style={{ display: "flex", gap: 16 }}>  <div style={{ minWidth: 300 }}>  <div style={{ border: "1px solid #eee", padding: 12, borderRadius: 8 }}>  <div><strong>Order:</strong> {order?.orderId}</div>  <div style={{ marginTop: 8 }}><strong>TxHash:</strong> <code style={{ fontSize: 12 }}>{txHash || "-"}</code></div>  <div style={{ marginTop: 8 }}><strong>Confirmations:</strong> {confirmations} / {required}</div>  <div style={{ marginTop: 8 }}><Timer seconds={15\*60} /></div>  {confirmations >= required ? (  <div style={{ marginTop: 12, color: "green" }}>  ✅ Payment confirmed. You may ship the product.  <div style={{ marginTop: 8 }}>  <button onClick={onBackToShop} style={{ padding: "8px 12px" }}>Back to shop</button>  </div>  </div>  ) : (  <div style={{ marginTop: 12, color: "#333" }}>  Waiting for confirmations...  </div>  )}  </div>  </div>  <div style={{ flex: 1 }}>  <div style={{ border: "1px solid #eee", padding: 12, borderRadius: 8 }}>  <strong>Debug / Note</strong>  <p style={{ color: "#666", marginTop: 8 }}>  This demo verifies client-side only (mock). For production verify server-side against a trusted node and use more confirmations.  </p>  </div>  </div>  </div>  </div>  );  } |

src/Components/Timer.js:

|  |
| --- |
| // src/Components/Timer.js  import React, { useEffect, useState } from "react";  export default function Timer({ seconds = 900 }) {  const [t, setT] = useState(seconds);  useEffect(() => {  const iv = setInterval(() => setT(s => Math.max(0, s - 1)), 1000);  return () => clearInterval(iv);  }, []);  const mm = Math.floor(t / 60).toString().padStart(2, "0");  const ss = (t % 60).toString().padStart(2, "0");  return <div>Time left: {mm}:{ss}</div>;  } |

src/Components/Nav.js:

|  |
| --- |
| // src/Components/Description.js  import React from "react";  export default function Description() {  return (  <div style={{ marginBottom: 12, color: "#444" }}>  Simple demo: browse shoes, buy, paste txHash and verify (mock).  </div>  );  } |

src/Components/Description.js:

|  |
| --- |
| // src/Components/Description.js  import React from "react";  export default function Description() {  return (  <div style={{ marginBottom: 12, color: "#444" }}>  Simple demo: browse shoes, buy, paste txHash and verify (mock).  </div>  );  } |

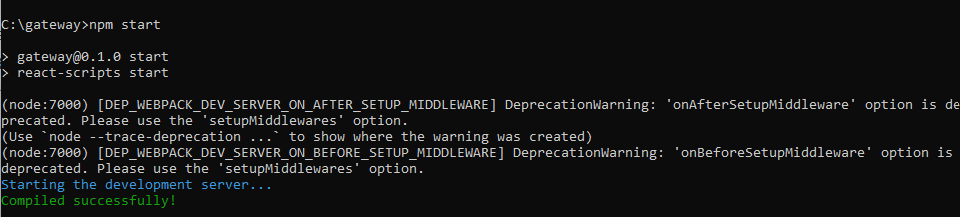
src/Components/Container.js:

|  |
| --- |
| // src/Components/Container.js  import React, { useState } from "react";  import Nav from "./Nav";  import Description from "./Description";  import Shoerack from "./Shoerack";  import Payment from "./Payment";  import PVerification from "./PVerification";  export default function Container() {  const [page, setPage] = useState("shop"); // shop | payment | verify  const [selected, setSelected] = useState(null);  const [currentOrder, setCurrentOrder] = useState(null);  const [pendingTx, setPendingTx] = useState("");  function handleBuy(item) {  setSelected(item);  setPage("payment");  }  function handleCancelPayment() {  setSelected(null);  setPage("shop");  }  function handleProceedToVerify({ orderId, txHash }) {  // save order info and txHash; here orderId is mocked from Payment component  setCurrentOrder({ orderId });  setPendingTx(txHash || "0xMOCK\_TX\_HASH");  setPage("verify");  }  return (  <div style={{ padding: 16 }}>  <Nav />  <Description />  {page === "shop" && <Shoerack onBuy={handleBuy} />}  {page === "payment" && selected && (  <Payment  product={selected}  onProceedToVerify={handleProceedToVerify}  onCancel={handleCancelPayment}  />  )}  {page === "verify" && currentOrder && (  <PVerification  order={currentOrder}  txHash={pendingTx}  onBackToShop={() => { setPage("shop"); setSelected(null); setCurrentOrder(null); setPendingTx(""); }}  />  )}  </div>  );  } |

src/App.js

|  |
| --- |
| // src/App.js  import React from "react";  import Container from "./Components/Container";  function App() {  return (  <div className="App" style={{ fontFamily: "Arial, sans-serif", background: "#f5f7fb", minHeight: "100vh" }}>  <Container />  </div>  );  }  export default App; |

**1. Start the React app:**



**Then show in localhost3000:**

