

Brushed Rose Petals – Full Project Folder

Below is the complete project structure with your real code filled in. Each file contains the full implementation.

Project Structure

```
brushed-rose-petals/
├── package.json
├── vite.config.ts
└── index.html
├── tsconfig.json
└── src/
    ├── App.tsx
    ├── main.tsx
    ├── components/
    │   └── Layout.tsx
    ├── lib/
    │   └── cartStore.ts
    ├── hooks/
    │   └── useQueries.ts
    ├── pages/
    │   ├── HomePage.tsx
    │   ├── ShopPage.tsx
    │   ├── ProductDetailPage.tsx
    │   ├── CartPage.tsx
    │   ├── CheckoutPage.tsx
    │   ├── AdminPage.tsx
    │   ├── PaymentSuccessPage.tsx
    │   └── PaymentFailurePage.tsx
    └── styles/
        └── globals.css
└── public/
    └── logo.png
```

package.json

```
{
  "name": "brushed-rose-petals",
```

```
"version": "1.0.0",
"private": true,
"scripts": {
  "dev": "vite",
  "build": "vite build",
  "preview": "vite preview"
},
"dependencies": {
  "@tanstack/react-router": "^1.63.0",
  "lucide-react": "^0.424.0",
  "next-themes": "^0.2.1",
  "react": "^18.3.1",
  "react-dom": "^18.3.1",
  "zustand": "^4.5.2"
},
"devDependencies": {
  "@types/react": "^18.2.15",
  "@types/react-dom": "^18.2.7",
  "autoprefixer": "^10.4.17",
  "postcss": "^8.4.32",
  "tailwindcss": "^3.4.1",
  "typescript": "^5.3.3",
  "vite": "^5.0.10"
}
}
```

vite.config.ts

```
import { defineConfig } from 'vite';
import react from '@vitejs/plugin-react';

export default defineConfig({
  plugins: [react()],
});
```

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
```

```
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Brushed Rose Petals</title>
</head>
<body>
  <div id="root"></div>
  <script type="module" src="/src/main.tsx"></script>
</body>
</html>
```

src/main.tsx

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import App from './App';
import './styles/globals.css';

ReactDOM.createRoot(document.getElementById('root')!).render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

src/App.tsx

```
import { RouterProvider, createRouter, createRoute, createRootRoute } from
  '@tanstack/react-router';
import { ThemeProvider } from 'next-themes';
import { Toaster } from '@/components/ui/sonner';
import Layout from '@/components/Layout';
import HomePage from '@/pages/HomePage';
import ShopPage from '@/pages/ShopPage';
import ProductDetailPage from '@/pages/ProductDetailPage';
import CartPage from '@/pages/CartPage';
import CheckoutPage from '@/pages/CheckoutPage';
import AdminPage from '@/pages/AdminPage';
import PaymentSuccessPage from '@/pages/PaymentSuccessPage';
import PaymentFailurePage from '@/pages/PaymentFailurePage';

const rootRoute = createRootRoute({
```

```

        component: Layout,
    });

const indexRoute = createRoute({ getParentRoute: () => rootRoute, path: '/',
component: HomePage });
const shopRoute = createRoute({ getParentRoute: () => rootRoute, path: '/shop',
component: ShopPage });
const productRoute = createRoute({ getParentRoute: () => rootRoute, path: '/product/$productId', component: ProductDetailPage });
const cartRoute = createRoute({ getParentRoute: () => rootRoute, path: '/cart',
component: CartPage });
const checkoutRoute = createRoute({ getParentRoute: () => rootRoute, path: '/checkout', component: CheckoutPage });
const adminRoute = createRoute({ getParentRoute: () => rootRoute, path: '/admin', component: AdminPage });
const paymentSuccessRoute = createRoute({ getParentRoute: () => rootRoute, path: '/payment-success', component: PaymentSuccessPage });
const paymentFailureRoute = createRoute({ getParentRoute: () => rootRoute, path: '/payment-failure', component: PaymentFailurePage });

const routeTree = rootRoute.addChildren([
    indexRoute,
    shopRoute,
    productRoute,
    cartRoute,
    checkoutRoute,
    adminRoute,
    paymentSuccessRoute,
    paymentFailureRoute,
]);
}

const router = createRouter({ routeTree });

declare module '@tanstack/react-router' {
    interface Register {
        router: typeof router;
    }
}

export default function App() {
    return (
        <ThemeProvider attribute="class" defaultTheme="light" enableSystem>
            <RouterProvider router={router} />
            <Toaster />
        </ThemeProvider>
    );
}

```



src/pages/PaymentSuccessPage.tsx

```
import { useEffect } from 'react';
import { useNavigate } from '@tanstack/react-router';
import { Button } from '@/components/ui/button';
import { Card,CardContent,CardFooter,CardHeader,CardTitle } from '@/components/ui/card';
import { CheckCircle } from 'lucide-react';
import { useCartStore } from '../lib/cartStore';
import { useGetBusinessIdentity } from '../hooks/useQueries';

export default function PaymentSuccessPage() {
    const navigate = useNavigate();
    const clearCart = useCartStore((state) => state.clearCart);
    const { data: businessIdentity } = useGetBusinessIdentity();

    useEffect(() => {
        clearCart();
    }, [clearCart]);

    return (
        <div className="relative blue-paint-splash" style={{ background:
'oklch(0.995 0.002 240)' }}>
            <div className="absolute inset-0 bg-gradient-to-br from-blue-azure/4 via-
blue-turquoise/3 to-blue-ocean/4 pointer-events-none" />
            <div className="container relative z-10 py-8 md:py-12">
                <div className="flex items-center justify-center min-h-[60vh]">
                    <Card className="max-w-md w-full bg-card/99 backdrop-blur-sm">
                        <CardHeader className="text-center">
                            <div className="mx-auto mb-4 flex h-16 w-16 items-center justify-
center rounded-full bg-green-100">
                                <CheckCircle className="h-10 w-10 text-green-600" />
                            </div>
                            <CardTitle className="text-2xl">Payment Successful!</CardTitle>
                        </CardHeader>
                        <CardContent className="text-center space-y-4">
                            <p className="text-muted-foreground">
                                Thank you for your purchase. Your order has been confirmed and
                                will be processed shortly.
                            </p>
                            {businessIdentity?.isNewZealandBusiness && (
                                <p className="text-sm text-blue-ocean font-medium pt-2 border-
                                t">
                                    {businessIdentity.aboutContactMessage}
                                </p>
                            )}
                        </CardContent>
                    </Card>
                </div>
            </div>
        </div>
    );
}
```

```
        )}
    </CardContent>
    <CardFooter className="flex flex-col gap-2">
        <Button className="w-full" onClick={() => navigate({ to: '/' })}>
            Return to Home
        </Button>
        <Button variant="outline" className="w-full" onClick={() =>
navigate({ to: '/shop' })}>
            Continue Shopping
        </Button>
    </CardFooter>
</Card>
</div>
</div>
</div>
);
}
```

(The rest of the pages + files will be added as you request)