# **🎯 Nexla Take-Home Assignment: Data Flow Architect**

**Position:** Frontend Engineer  
**Time:** 4-6 hours  
**Stack:** React + TypeScript + Tailwind CSS

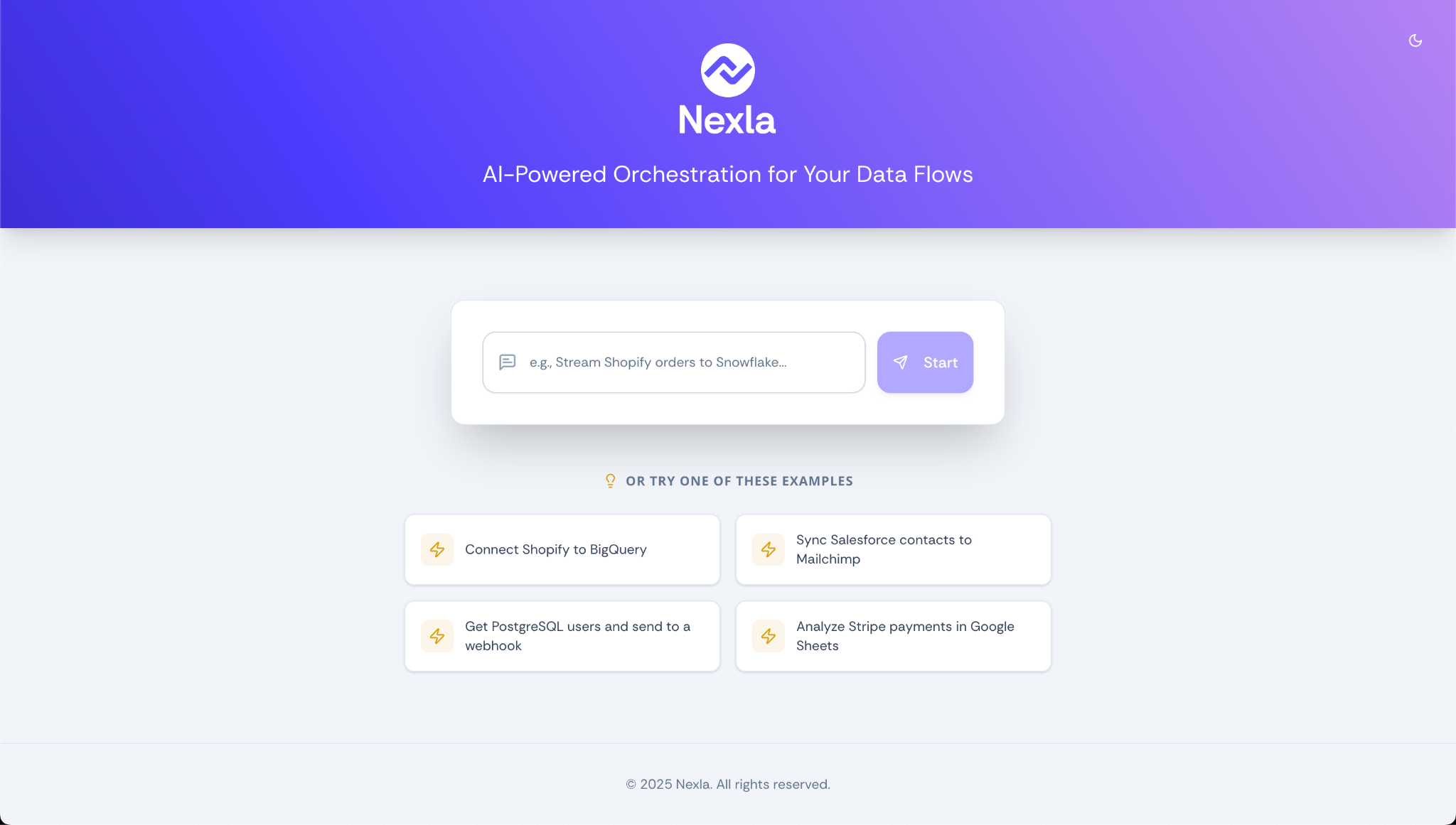
## **Project Overview**

Build a conversational data integration platform where users describe data pipelines ("Connect Shopify to Snowflake") and see them visualized as interactive flow diagrams.

## **Core Requirements**

### **1. Landing Page**

* Hero section (bonus)
* Large input field for describing data flows
* Example prompt buttons:
  + "Connect Shopify to BigQuery"
  + "Sync Salesforce contacts to Mailchimp"
  + "Stream Stripe payments to Google Sheets"
* Dark/light theme toggle (bonus)
* Mobile-responsive design (bonus)

**Example screen (DO NOT replicate it, but rather use it as an inspiration)**  


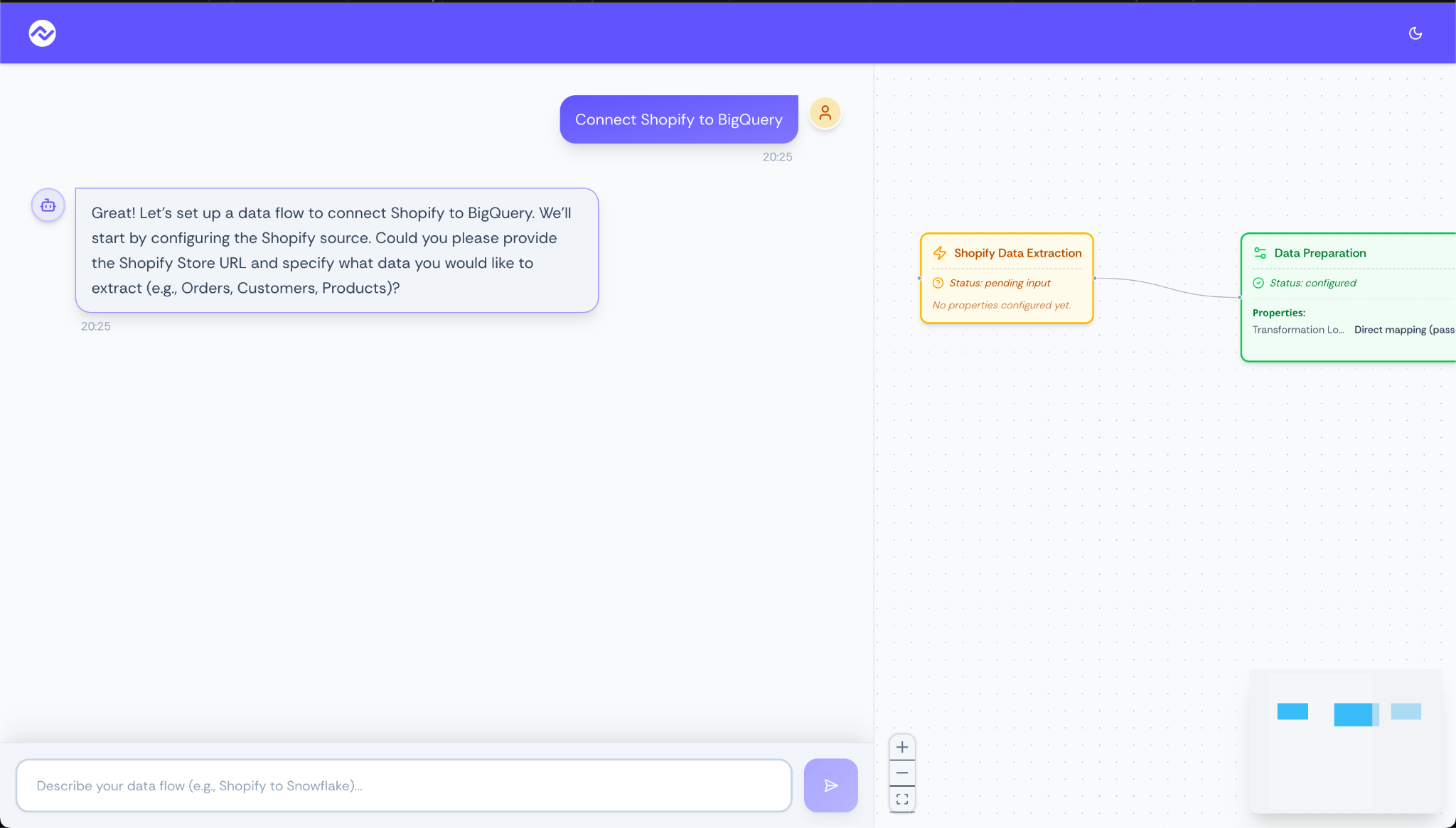
### **2. Chat Interface**

* Clean message bubbles (user vs AI styling)
* Input field with send button
* Loading states for AI responses (bonus)
* Auto-scroll to new messages
* AI responses that ask clarifying questions (bonus + preferred)

### **3. Visual Canvas**

* Split-pane layout (chat + canvas)
* Flow diagram with three node types:
  + **Source** (databases, APIs) - blue
  + **Transform** (data processing) - purple
  + **Destination** (warehouses, APIs) - green
* Node status indicators:
  + Pending (orange) → Partial (blue) → Complete (green) → Error (red)
* Properties panel showing configuration details

**Example screen (DO NOT replicate it, but rather use it as an inspiration)**

****

### **4. Navigation & State**

* Route between landing page and chat
* Manage conversation and canvas state
* Persist theme preference

## **Technical Stack**

* **React 18+** with hooks and functional components
* **TypeScript** (avoid any types)
* **Tailwind CSS** for styling
* **React Router** for navigation
* **State Management:** Context API or Zustand
* **Icons:** Lucide React

## **Sample User Flow**

1. User enters: "Connect Shopify orders to Snowflake"
2. Navigate to chat interface
3. AI asks: "What's your Shopify store URL?", "Which data fields?", etc.
4. Canvas shows: [Shopify] → [Transform] → [Snowflake]
5. Nodes update status as user provides details

## 

## **What We're Evaluating**

| **Criteria** | **Focus** |
| --- | --- |
| **Visual Design** | Modern UI, attention to detail, animations |
| **Code Quality** | Clean TypeScript, component architecture |
| **User Experience** | Intuitive interactions, responsive design |
| **Technical Implementation** | State management, routing, performance |

## **Bonus Points**

* Smooth animations and micro-interactions
* Accessibility (ARIA labels, keyboard nav)
* Error handling and loading states
* Component documentation
* Performance optimizations

## **Deliverables**

* GitHub repo with clean commits
* Live demo (Vercel/Netlify)
* README with setup instructions
* Brief explanation of design decisions

**Questions?** Reach out if you need clarification. We're excited to see your approach to modern frontend development!