Ria Software Engineer Intern - Coding Challenge

Currency Exchange Dashboard

Overview

Build a currency exchange dashboard using Next.js that allows users to convert currencies and view live exchange rates. This project evaluates your technical skills, code quality, and product thinking. You can use any publicly available tool including ai, but if you use ai tools you should document in README your way of usage.

Time Expectation

2–4 hours

You don't need to spend more than this. We value quality over quantity—a well-structured, clean solution with fewer features is better than a rushed implementation with many features.

Technical Requirements

- **Framework:** Next.js 14+ (App Router)
- **Language:** TypeScript
- **Styling:** Your choice (Tailwind CSS, CSS Modules, styled-components, etc.)
- **API:** Frankfurter API (free, no API key required)

Functional Requirements

Your application must include the following features:

1. Currency Converter

- User can enter an amount
- User can select a "from" currency and a "to" currency
- Display the converted amount

2. Exchange Rates Overview

- Display current exchange rates for at least 10 major currencies against a selected base currency
- Allow the user to change the base currency

Bonus ⭐

- **Add one feature that you think would make this tool more useful for someone sending money internationally.**

This is intentionally open-ended. We want to see how you think about product and user needs. In your README, briefly explain:

- What feature you added
- Why you chose it
- How it improves the user experience

There's no right or wrong answer here—we're interested in your reasoning and creativity.

- **Deploy your application** to Vercel, Netlify, or similar (free tier is fine)

API Reference

Frankfurter API — https://www.frankfurter.app/docs/

Useful endpoints:

Endpoint	Description
`GET /latest`	Latest exchange rates
`GET /latest?from=USD`	Rates with USD as base
`GET /latest?from=USD&to=EUR,GBP`	Specific currency pairs
`GET /currencies`	List of available currencies
`GET /2024-01-15`	Historical rates for a specific date
`GET /2024-01-01..2024-01-31`	Time series data

Submission Instructions

1. **Create a GitHub repository** with your solution
2. **Include a README.md** with:
   - Brief description of your project
   - Setup instructions (`npm install`, `npm run dev`, etc.)
   - Your innovation feature explanation
   - If you used AI in this project, how did you use it?
   - Any assumptions or trade-offs you made
   - (Optional) What you would improve with more time
3. **Send to the same mail that you received this instruction from, with CC to zyan@riamoneytransfer.com and cpozas@riamoneytransfer.com:**
   - Title: Ria Intern Project-[your name]
   - Link to your GitHub repository
   - Link to the deployed application if deployed

Evaluation Criteria

Criteria	What We Look For
Functionality	All functional requirements working correctly—converter calculates accurately, rates display properly, base currency switching works
Code Quality	Clean component architecture, proper TypeScript typing, meaningful naming, separation of concerns (e.g., API logic separate from UI), DRY principles
User Experience	Loading states handled appropriately, errors displayed gracefully (API failures, network issues, invalid inputs), responsive design (mobile and desktop), UI quality
Innovation Feature	Creativity and user-centric thinking, quality of implementation, clear explanation of reasoning
Documentation	Clear README with setup instructions, explanation of decisions and trade-offs, code comments where helpful, AI usage explanations

Questions?

If you have any questions about the requirements, please reach out to zyan@riamoneytransfer.com with the title [Intern challenge question] We're happy to clarify anything. Good luck and have fun programming!