

Front-end Developer Task (Next.js)

Preconditions:

- Typescript;
- Tailwind CSS;
- React Functional Components;
- Reusable components;
- User intuitive interface, proper UI behavior (validation, spinners, errors, etc.);
- API base URL: <https://67862af6f80b78923aa5ace3.mockapi.io/api/v1/balance>

Task:

- **Part 1:**
 - Implement registration page;
 - Implement login page with OTP. (Enter email, password, and then render an OTP form);
 - Mock server request behavior using `setTimeout()`;
 - Add incorrect email UI behavior (Login) with hardcode value of `'incorrect@email.com'`;
 - Add incorrect password UI behavior (Login) with hardcode value of `'incorrect-password'`;
- **Part 2:**
 - **GET** data from `/currencies`;
 - Map currencyId to its abbreviation (check the last page);
 - Implement a table in the following way (see the screenshot) with a **dynamically adjustable column count** (increment, decrement);
 - Implement the second table, GET data from `/not-found`, and handle the error response;

Balances					
Name	Balance	Name	Balance	Name	Balance
EUR	52.22	CHF	534.00	ARS	567411.23
USD	723.31	INR	35671.33	GBP	23.23
JPY	534531.67				

Bonus Part:

- Add input to search by currencySymbol (client-side);
- Delete an item from the UI;
- Implement coverage report:
 - Write Unit Tests using Jest (Core logic should be covered at least);
 - Write Functional Tests using React Testing Library with snapshots (For proper rendering, API requests and responses should be mocked);

Currency Mapping:

```
{  
  "1": "AUD",  
  "2": "RSD",  
  "3": "CHF",  
  "4": "JPY",  
  "5": "EUR",  
  "6": "USD",  
  "7": "DZD",  
  "8": "ARS",  
  "9": "AZN",  
  "10": "BRL",  
  "11": "CNY",  
  "12": "GEL",  
  "13": "INR",  
  "14": "LVL",  
  "15": "OMR",  
  "16": "CUP",  
  "17": "ZAR",  
  "18": "ZWD",  
  "19": "QAR",  
  "20": "PLN",  
  "21": "GBP",  
  "22": "CAD",  
  "23": "SEK",  
  "24": "PHP",  
  "25": "IDR"  
}
```