MediCode Project Requirements

Project Scope

Build a web app that:
1) Accepts a user’s list of medications (brand/generic)
2) Normalizes names to **RxNorm** ingredients (IN)
3) Dedupes to unique ingredients
4) Builds unique pairs and fetches label interaction text for each ingredient from openFD*
5) Flags, mentions (A label mentions B, and vice versa), and displays source snippets + links

Note: RxNav’s own drug–drug interaction endpoint is discontinued; we rely on label text (openFDA) for our free MVP.

MVP Functional Requirements

- Input
  - Enter list of drugs (comma- or newline-separated)
  - Typos tolerated via RxNorm approximate search functionality

- Normalization
  - For each input, resolve: `rxcui`, preferred display name, ingredients

- Deduplication
  - Collapse multiple brand/generics into same ingredient bucket
  - Maintain mapping: `ingredient -> {sourceDrugs[]}`

- Pairing
  - Generate unordered unique pairs of ingredients

- Label fetch
  - Per ingredient, fetch `drug_interactions[]` from openFDA (cache results)

- Matching
  - For each pair:
    - A’s label text “mentions” B (substring check, min length threshold)
    - B’s label text “mentions” A

- Results UI
  - Show normalized data, deduped ingredients, pairs
  - ✓ / ✗ flags with expandable snippets and **source links**
  - Status messages and a small debug pane

- Error handling
  - Continue if some entries don’t match RxNorm
  - Show “no data” if no label interaction text found

Data Constructs (shapes we pass around to and from API)

```
type Normalized = {
  query: string;
  rxcui: string | null;
  display: string | null;
  ingredients: string[];
  error?: "no_match";
};
```

- Normalized is the user input for medication names normalized, returning following properties (JSON)
- rxcui is the drug unique identifier
- display is the normalized/preferred name
- rxnorm ingredients names string array
- passes

```
type IngredientNode = {
  ingredient: string;
  key: string;
  sourceDrugs: Set<string>;   // inputs that mapped here
};
```

- IngredientNode returns the ingredients from the Normalized rxcui
- Maps the source drugs from the input

```
type LabelPayload = {
  lines: string[];         // openFDA label drug_interactions[]
  source: string;          // linkable query URL
};
```

- LabelPayload is the return from openFDA with the label drug interaction array
- Returns the source from the openFDA drug interaction