

Guida Tecnica allo Sviluppo Frontend

Sintesi dei concetti: Redux Toolkit, React, i18n e Git Workflow

1. State Management con Redux Toolkit (RTK)

L'architettura utilizza RTK per gestire lo stato globale in modo modulare e prevedibile.

Azioni Sincrone vs Asincrone

Funzione	Tipo	Utilizzo Principale
createAction	Sincrona	Segnali semplici o trigger (es. reset dello store).
createAsyncThunk	Asincrona	Chiamate API, logica complessa con side-effects.
createActionTypesMap	Utility	Generazione di nomi univoci per le azioni in ambito MFE.

Reducers vs ExtraReducers: I reducers gestiscono azioni create internamente allo slice, mentre gli extraReducers rispondono ad azioni esterne (come i Thunk) o segnali da altri moduli.

2. Flusso Asincrono (Thunks)

Il Thunk gestisce il ciclo di vita di una richiesta al server: **Pending** → **Fulfilled** / **Rejected** .

```
// Struttura di un Thunk intelligente (filterByText)
const filterByText = createAsyncThunk(name, async (text, thunkAPI) => {
  const state = thunkAPI.getState();
  const filters = selectors.getFilters(state);
  const response = await api.call({ ...filters, text });
  return response.data; // Questo diventa il Payload
});
```

La proprietà condition

Permette di bloccare una chiamata prima che parta (es. se i dati sono già presenti o se la ricerca è identica alla precedente), ottimizzando le performance.

3. Componenti Custom & Virtualizzazione

Componenti come `VirtualizedSelect` sono progettati per gestire enormi moli di dati (Budget, Categorie) senza rallentare il browser.

Pattern: FetchMoreItems

Utilizza la tecnica dell'**Infinite Scroll**. Il componente delega il caricamento dei dati a Redux tramite una callback:

- **Componente:** Rileva lo scroll e chiama `fetchMoreItems`.
- **Redux:** Riceve la richiesta, esegue il Thunk e aggiorna lo store.
- **Risultato:** L'utente vede nuovi dati apparire fluidamente.

4. Internazionalizzazione (i18n)

Il sistema è basato su `i18next` e utilizza il **Lazy Loading** per caricare i testi solo quando necessario.

Funzione	Descrizione
<code>loadTranslations</code>	Carica i bundle JSON per un namespace specifico. Non sovrascrive se già presente.
<code>overwriteTranslations</code>	Carica i testi e forza l'aggiornamento se il namespace esiste già.
<code>interpolation</code>	Permette di formattare testi dinamicamente (es. UPPERCASE) nei file JSON.

Ogni modulo ha la sua cartella `i18n/` per garantire l'isolamento (Namespacing) ed evitare conflitti tra etichette simili.

5. Git Workflow & Best Practices

Collaborare in team richiede standard di comunicazione chiari tramite Git.

Conventional Commits

Tipo	Esempio di utilizzo
feat	Nuova funzionalità (es. aggiunta filtro).
fix	Correzione di un bug (es. errore di traduzione).
refactor	Pulizia o miglioramento del codice senza cambi logici.
chore	Manutenzione (es. aggiornamento dipendenze).

Pull Requests (PR)

Le PR sono lo spazio di revisione del codice. Una buona PR deve essere piccola, ben descritta e accompagnata da screenshot se modifica la UI.

6. Tips per VS Code

- **Preview Mode:** Disattiva Workbench > Editor: Enable Preview per evitare che i file aperti sostituiscano il tab corrente.
- **Shortcuts:** Usa `Cmd + \` per dividere lo schermo e confrontare Actions e Reducers contemporaneamente.
- **GitLens:** Fondamentale per capire chi ha modificato cosa e quando (Git Blame).