Low Level Design Document

Introduction

This Low Level Design (LLD) document outlines the core components and implementation details for **Evently - Event Listing Portal**. Evently is a React-based web application for browsing and searching local events, with filtering by date, category, and location.

---

1. System Components

Component	Description	Key Responsibilities
EventList	Displays list of events	Render events, handle pagination
EventFilter	UI for filtering events	Manage filter state, trigger search
EventCard	Shows event summary	Display event info, handle click
EventDetailModal	Shows detailed event info	Fetch and display event details
App	Main app container	Route components, manage state
API Service	Handles API requests	Fetch events, handle errors

---

2. Class/Interface Overview

Component/Class	Key Props/State/Methods	Relationships
`EventList`	`events`, `onEventClick`, `loading`	Uses `EventCard`
`EventFilter`	`filters`, `onFilterChange`	Used by `App`
`EventCard`	`event`, `onClick`	Used by `EventList`
`EventDetailModal`	`eventId`, `onClose`	Used by `App`
`App`	`events`, `filters`, `selectedEventId`	Parent of all components
`apiService`	`fetchEvents(filters)`, `fetchEventById(id)`	Used by `App`, `EventDetailModal`

Key Methods (Example):

```
// apiService.js
async function fetchEvents(filters) { /* ... */ }
async function fetchEventById(id) { /* ... */ }
```

---

3. Data Structure Overview

Model	Fields
Event	id, title, description, date, category, location, imageUrl
Filter	date (range), category (string), location (string)

Event Example:

```
{
  id: string,
  title: string,
  description: string,
  date: string (ISO),
  category: string,
  location: string,
  imageUrl: string
}
```

---

4. Algorithms / Logic

Event Filtering Flow (Pseudocode):

```
onFilterChange(newFilters):
  setFilters(newFilters)
  setLoading(true)
  events = await apiService.fetchEvents(newFilters)
  setEvents(events)
  setLoading(false)
```

Event Detail Modal Flow:

```
onEventClick(eventId):
  setSelectedEventId(eventId)
  // Modal opens, fetches event details by ID
```

---

5. Error Handling

Scenario	Handling Approach
API fetch failure	Show error message, allow retry
No events found	Show “No events found” message
Invalid filter input	Validate input, show inline error
Network error	Show notification, suggest reload

End of Document