Agile Aggregator – Class Documentation

# AggregationController

This is the main API controller.  
- Route: `/api/aggregation`  
- Accepts query parameters such as `keyword`, `count`, `page`, `sortBy`, `fromDate`, and `services`.  
- Delegates calls to one or more `IAggregatorService` implementations.  
- Supports sequential and parallel service execution based on number of requested services.  
- Returns a dictionary mapping service name to `AggregatedResponse`.

# IAggregatorService

Interface implemented by all services (e.g., News, Weather, Spotify).  
- Method: `Task<AggregatedResponse> FetchDataAsync(...)`  
- Property: `string Name { get; }` used for service identification.

# NewsService

Fetches news from NewsAPI.org.  
- Supports filtering by keyword, date, and sorting (`publishedAt`, `relevancy`, `popularity`).  
- Handles API errors gracefully and returns fallback messages.  
- Implements `IAggregatorService`.

# SpotifyService

Fetches tracks from Spotify Web API using client credentials OAuth.  
- Returns top tracks matching the search keyword.  
- Includes image URL and artist metadata.  
- Implements `IAggregatorService`.

# WeatherService

Fetches current weather from OpenWeatherMap.  
- Uses city name as the keyword.  
- Only returns one result (current weather snapshot).  
- Implements `IAggregatorService`.

# AggregatedResponse

Model returned by all services.  
- Contains `List<AggregatedResult>` and a `TotalCount`.  
- Used to unify output of different APIs.

# AggregatedResult

Represents a single result from any service.  
- Fields: `Title`, `Description`, `Date`, `Url`, `ImageUrl`, `Source`.  
- Used in both frontend display and API output.