**Parse Message API**

**Overview**

Parse Message API is a .NET Core-based backend service that processes user messages, extracts structured information, caches results using Redis, and schedules delayed tasks using RabbitMQ.

**Technologies Used**

* **.NET Core** for API development
* **Entity Framework Core** with **PostgreSQL** as the database
* **Redis** for caching
* **RabbitMQ** for delayed messaging
* **Serilog** for logging

**API Endpoints**

**1. Parse Message**

**Endpoint:** POST /api/Message/ParseMessage

**Request Body:**

{

   "message": "`Docter appointment` on next sunday at 2pm remind sriram, tamil via Teams and repeat everyday"

}

**Response:**

{

"id": "dceba3d1-8b30-4c0e-9720-8ca314079ae4",

"title": "Docter appointment",

"repeat": true,

"app": "Teams",

"time": "14:00:00",

"date": "30-03-2025",

"trigger\_At": "2025-03-30T14:00:00",

"created\_At": "2025-03-26T10:22:55.0198192Z",

"send\_to": [

{

"axusersid": 1059440000000,

"username": "Tamil",

"email": "tamil@agile-labs.com",

"mobile": null

},

{

"axusersid": 1059330000000,

"username": "Sriram",

"email": "sriram.s@agile-labs.com",

"mobile": null

}

]

}

**2. Send Delayed Message**

**Endpoint:** POST /api/Message/sendDely

**Request Body:**

{

"id": "dceba3d1-8b30-4c0e-9720-8ca314079ae4",

"title": "Docter appointment",

"repeat": true,

"app": "Teams",

"time": "14:00:00",

"date": "30-03-2025",

"trigger\_At": "2025-03-30T14:00:00",

"created\_At": "2025-03-26T10:22:55.0198192Z",

"send\_to": [

{

"axusersid": 1059440000000,

"username": "Tamil",

"email": "tamil@agile-labs.com",

"mobile": null

},

{

"axusersid": 1059330000000,

"username": "Sriram",

"email": "sriram.s@agile-labs.com",

"mobile": null

}

]

}

**Response:**

Message 'Docter appointment' scheduled successfully for 14:00:00.

**Architecture and Components**

**1. Database Schema (PostgreSQL)**

Tables:

* Messages: Stores parsed messages
* AxUsers: Stores user information

**2. Caching (Redis)**

* Used to store parsed messages for quick retrieval.
* Implemented in RedisCacheServices.cs.

**3. Message Queue (RabbitMQ)**

* Used to schedule delayed message delivery.
* Implemented in MessageProducer.cs and MessageConsumer.cs.

**4. Logging**

* Implemented using Serilog (Program.cs).
* Logs are stored in logs/errors.log.

**Configuration (appsettings.json)**

{

"Redis": {

"ConnectionString": "localhost:6379"

},

"ConnectionStrings": {

"PostgresConnection": "Host=localhost;Port=5432;Database=axpertdb;Username=postgres;Password=log"

}

}

**Dependencies**

* Microsoft.EntityFrameworkCore
* StackExchange.Redis
* RabbitMQ.Client
* Serilog

**Conclusion**

This API efficiently processes user messages, caches results, and schedules delayed tasks. It integrates caching and messaging queues for optimized performance.