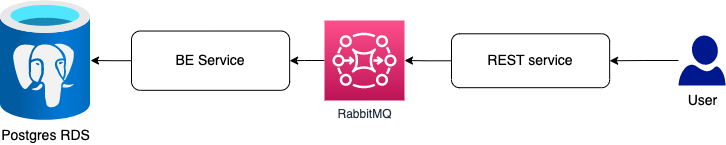
**Home Assignment**

As the next step in the process, we’d like you to complete a home assignment that focuses on both understanding complex system architecture and demonstrating your coding skills.

**Task Overview:**

Your assignment is to implement a **backend-only** To-Do application. Swagger will serve as the client interface for this task.



**Requirements:**

1. **Implement RESTful APIs** for managing both User and Item entities.
2. A User can have multiple Items (one-to-many relationship).
3. Design the models using a **minimal set of fields** necessary for the task.
4. **Soft delete** should be implemented for deleting an item.
5. Use **Entity Framework (EF)** with the **Code-First approach**.
6. Create two separate services:
   * **Web Service**:
     + Exposes the APIs using Swagger.
     + Handles only **input validation and verification logic**.
   * **Worker Service**:
     + Consumes messages from the web service.
     + Responsible for **persisting data** to the database.
7. **Inter-service communication** between the Web Service and the Worker Service should be implemented using **RabbitMQ** with an **RPC-style pattern** (i.e., request-reply over message queues).

**Technologies:**

* **Database**: PostgreSQL
* **Message Broker**: RabbitMQ

Focus on designing a clean and minimal **database schema** and defining a robust communication pattern with RabbitMQ (e.g., queue types, policies, error handling, etc.).

**Deliverables:**

* A **Docker Compose file** that brings up:
  + The web service
  + The worker service
  + PostgreSQL
  + RabbitMQ
* The Docker Compose setup should allow for running and testing the entire system locally with minimal configuration.
* Code should follow **best practices** and clean architecture principles as much as possible.

Feel free to reach out via email if you have any questions or need further clarification.

Looking forward to your submission!