

Aluno:

RA:

Data:

Projeto Mensageria

Entregáveis:

- demonstração do projeto funcionando;
- DER do Banco de Dados;
- Fontes do projeto no git;
- Deve ter commit de todos os membros.

1. Implemente um **consumidor para ler os dados de Reservas gerados por um sistema de hotelaria. Esses dados devem ser persistidos numa base relacional.**

- Deverá ter minimamente as tabelas: *reserva*, *quarto_reservado* e *cliente*;
- Registre a hora que a mensagem foi indexada na base de dados;
- Use a linguagem definida na divisão dos grupos;

2. Implemente uma rota de API que permita consultar as reservas [linguagem de preferência]

- Method: GET
- Path: /reserves
- Filtros: consulta por uuid, id do cliente, id do quarto, id do hotel;
- Retorno da API deve ter o contrato conforme modelo de payload;
- Deve-se adicionar um elemento que compute o **valor total da reserva** e de cada quarto;
- Atente-se a arquitetura RESTFull para modelagem da API.

Payload:

```
{
  "uuid": "3030-499f-39f949",
  "created_at": "2025-09-01T22:33:00Z",
  "type": "AB",
  "customer": {
    "id": 99494,
    "name": "joao da silva",
    "email": "joao.silva@email.com",
    "document": "123.456.789-00"
  },
  "hotel": {
    "id": 123,
    "name": "Hotel Exemplo",
    "city": "São Paulo",
    "state": "SP"
  },
  "rooms": [
    {
      "id": 1,
      "room_number": "101",
      "daily_rate": 300.00,
      "number_of_days": 3,
      "checkin_date": "2025-09-15",
      "checkout_date": "2025-09-18",
      "category": {
        "id": "AM",
        "name": "Apartamento Master",
        "sub_category": {
          "id": "BCRU",
          "name": "Bangalô Cruzeiro"
        }
      },
      "status": "confirmed",
      "guests": 2,
      "breakfast_included": true
    }
  ],
  "payment": {
    "method": "credit_card",
    "status": "approved",
    "transaction_id": "txn_123456789"
  },
  "metadata": {
    "source": "website",
    "user_agent": "Mozilla/5.0...",
    "ip_address": "192.168.1.1"
  }
}
```