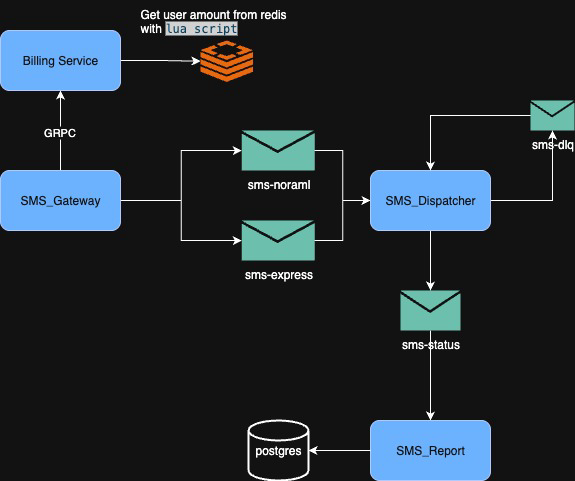
این پروژه یک سیستم SMS Gatewayل با معماری میکروسرویس است که قابلیت ارسال پیامک، مدیریت موجودی کاربران، گزارش گیری و پردازش پیام های فوری (Express) و (Normal) را فراهم میکند.

همه سرویس ها از طریق کافکا و درصورت لزوم از grpc برای ارتباط با یکدیگر استفاده میکنند.

**معماری سرویس‌ها**



سرویس‌ها

**1. Billing Service**

* مسئول مدیریت موجودی کاربران.
* ذخیره‌سازی موجودی در **Redis**.
* ارتباط با SMS Gateway از طریق **gRPC**.

**2. SMS Gateway Service**

* دریافت درخواست ارسال SMS از کلاینت‌ها.
* اعتبارسنجی درخواست و موجودی با Billing Service (gRPC).
* ارسال اس ام اس به تاپیک sms-express و یا sms-normal

**3. SMS Dispatcher Service**

* Consumer از Kafka برای sms-normal و sms-express.
* ارسال پیام به اپراتور مناسب بر اساس شماره موبایل (MCI/MTN/Rightel) با استفاده از **operator.json**.
* پیاده‌سازی DLQ:
  1. پیام‌های خطادار به تاپیک sms-dlq می‌روند.
  2. DLQ Consumer تا ۳ بار Retry می‌کند.
  3. نتیجه ارسال (چه موفق چه ناموفق) به تاپیک sms-status می‌رود.

**4. Report Service**

* Consumer از Kafka برای sms-status
* ذخیره در PostgreSQL.
* ارائه API های گزارش‌گیری

**تکنولوژی های استفاده شده**

* **Go:**  زبان پیاده سازی سرویس‌ها
* **Kafka:** برای message broker
* **gRPC:** برای ارتباط بین سرویس‌ها
* **Redis** : برای موجودی کاربران
* **PostgreSQL** : برای گزارشات

**مراحل اجرا**

# 1. run infra pre request

docker-compose up -d

# 2. run billing -service

cd billing-service

go mod tidy

go run main.go

# 3. sms-gateway

cd sms-gateway

go mod tidy

go run main.go

# 4. run sms-dispatcher

cd sms-dispatcher

go mod tidy

go run main.go

# 5. run report-service

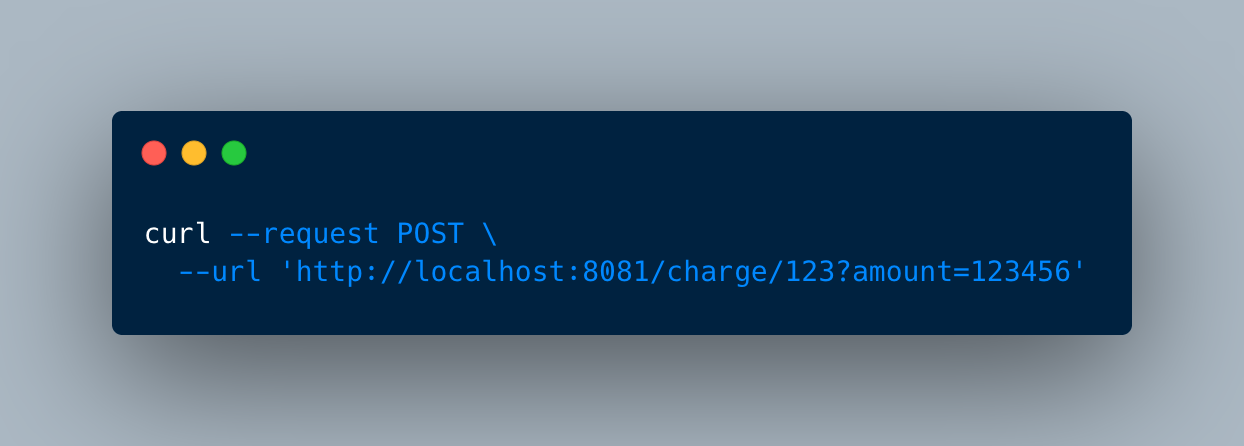
cd report-service

go mod tidy

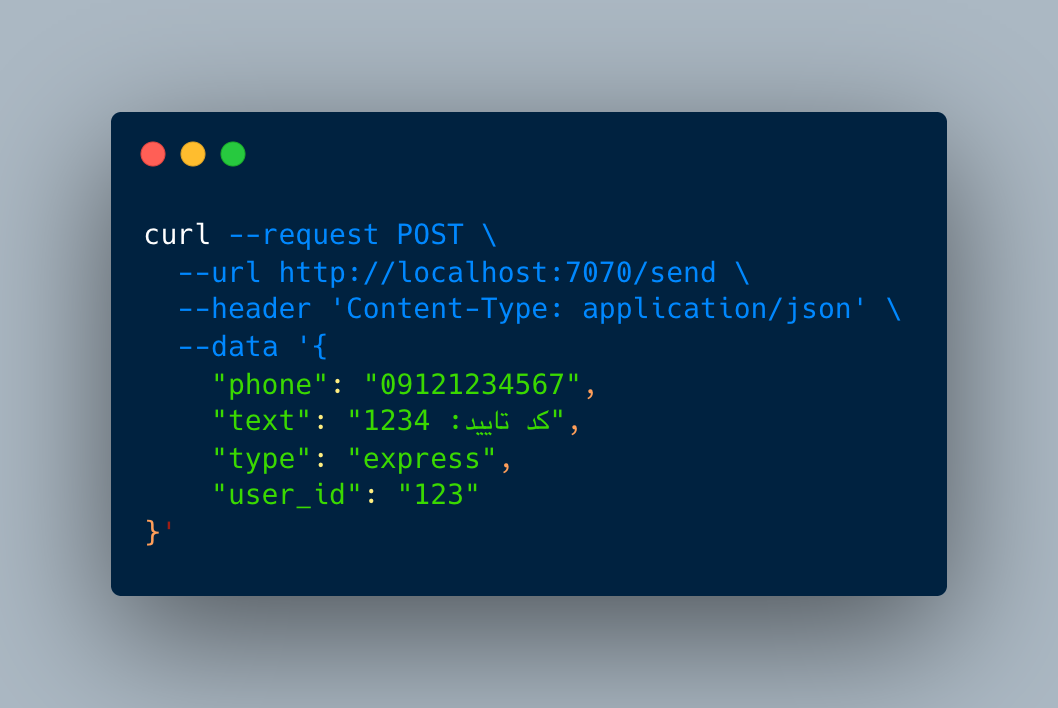
go run main.go

**Example API Calls**

**شارژ حساب کاربر**



**ارسال SMS**



**دریافت گزارش پیامک‌ها**

**A blue rectangle with white text

AI-generated content may be incorrect.**