

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



Nguyễn Nhật Anh

**NGHIÊN CỨU MÔ HÌNH MICROSERVICES VÀ ỨNG DỤNG
VÀO HỆ THỐNG TÍNH CƯỚC TẠI VIETTEL**

Chuyên ngành: Kỹ thuật viễn thông

Mã số: 8.52.02.08

TÓM TẮT ĐỀ ÁN TỐT NGHIỆP THẠC SĨ

HÀ NỘI - NĂM 2024

Đề án tốt nghiệp được hoàn thành tại:
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: TS. PHẠM ANH THU

Phản biện 1:

Phản biện 2:

Đề án tốt nghiệp sẽ được bảo vệ trước Hội đồng chấm đề án tốt nghiệp
thạc sĩ tại Học viện Công nghệ Bưu chính Viễn thông

Vào lúc: giờ ngày tháng năm

Có thể tìm hiểu đề án tốt nghiệp tại:

- Thư viện của Học viện Công nghệ Bưu chính Viễn thông.

MỞ ĐẦU

Trong thời đại số hóa hiện nay, dịch vụ viễn thông đóng vai trò quan trọng trong cuộc sống hàng ngày của chúng ta. Từ việc liên lạc, truy cập internet, đến sử dụng các ứng dụng di động, chúng ta không thể phủ nhận tầm quan trọng của viễn thông và hệ thống tính cước OCS (Online Charging System) trong việc cung cấp dịch vụ này.

Để đáp ứng nhu cầu ngày càng cao của người dùng và tận dụng tối đa tiềm năng của ngành viễn thông, các nhà cung cấp dịch vụ đã không ngừng nghiên cứu và phát triển hệ thống tính cước OCS. Việc quản lý cước phí một cách hiệu quả, đáng tin cậy và linh hoạt là một thách thức không nhỏ đối với các nhà cung cấp dịch vụ viễn thông.

Bên cạnh đó, với sự phát triển mạnh mẽ của công nghệ phần mềm thì mô hình vi dịch vụ trở thành xu hướng tất yếu để triển khai các hệ thống viễn thông và công nghệ thông tin. Với những lợi ích mang lại như tính linh hoạt trong việc triển khai và mở rộng độc lập, khả năng dễ dàng thay thế và nâng cấp đã thúc đẩy việc phát triển, tích hợp triển khai hệ thống tại các doanh nghiệp theo mô hình Microservices.

Tôi chọn đề tài nghiên cứu “Mô hình Microservices và ứng dụng vào hệ thống tính cước tại Viettel” không chỉ nằm ở việc cải thiện quá trình tính toán cước phí, mà còn ở khả năng đáp ứng nhu cầu người dùng và tối ưu hóa quy trình kinh doanh của các doanh nghiệp trong lĩnh vực này. Những lợi ích mà mô hình MS mang lại khi được tích hợp vào hệ thống tính cước OCS sẽ mang đến khả năng mở rộng độc lập, có thể mở rộng chỉ các dịch vụ cần thiết thay vì phải mở rộng toàn bộ ứng dụng. Bằng cách tìm hiểu, nghiên cứu về hệ thống OCS cũng như khả năng phát triển hệ thống theo mô hình MS, chúng ta có thể đóng góp vào sự phát triển và tiến bộ của ngành viễn thông, từ đó tạo ra những trải nghiệm tốt hơn cho người dùng và thúc đẩy sự cạnh tranh trên thị trường.

CHƯƠNG 1: TỔNG QUAN VỀ MÔ HÌNH MS

Chương này tập trung vào nghiên cứu mô hình MS, động lực thúc đẩy mô hình MS thay thế mô hình Monolith, kiến trúc tổng quan, những ưu, nhược điểm khi triển khai theo mô hình MS và khả năng ứng dụng trong các hệ thống viễn thông và công nghệ thông tin.

1.1. Động lực thúc đẩy mô hình Microservices

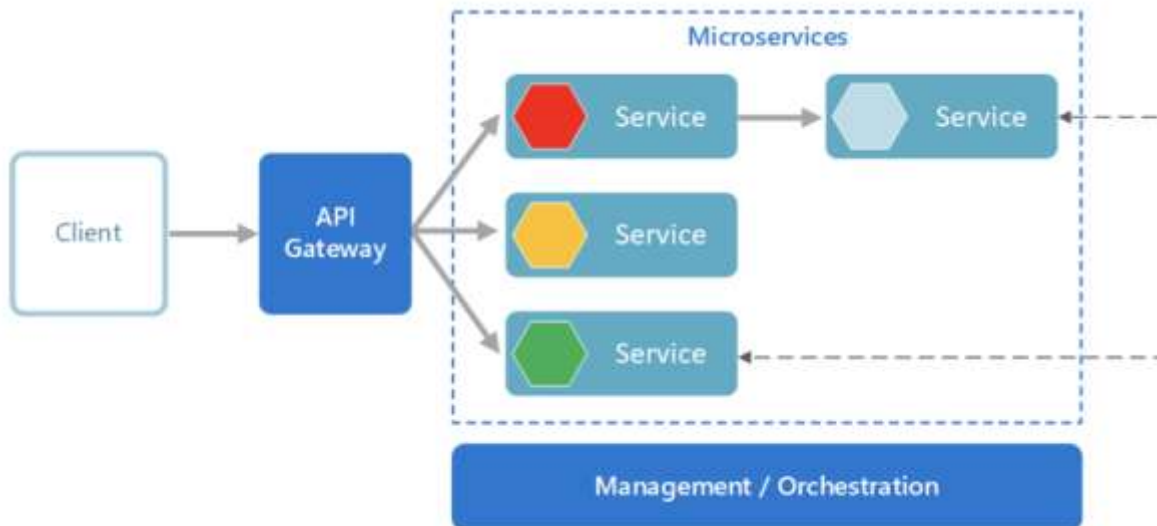
Các ứng dụng phần mềm doanh nghiệp được thiết kế để đáp ứng nhiều yêu cầu kinh doanh của doanh nghiệp. Ban đầu, các công ty thường xây dựng ứng dụng theo mô hình monolith truyền thống. Trong mô hình này, toàn bộ ứng dụng được triển khai và chạy trên một máy chủ hoặc một nhóm máy chủ duy nhất. Tất cả các chức năng và quy trình kinh doanh được tích hợp trong một mã nguồn duy nhất.

Theo thời gian, ứng dụng Monolith có thể trở nên phức tạp và khó quản lý hơn khi kích thước và quy mô của nó tăng lên do sự phức tạp về mặt nghiệp vụ của các hệ thống phần mềm doanh nghiệp. Việc thay đổi và triển khai các tính năng mới trở nên khó khăn do sự phụ thuộc mạnh mẽ giữa các thành phần trong monolith. Nhằm giải quyết những khó khăn này, hướng kiến trúc Microservices đã ra đời.

Microservices cung cấp cho các nhà phát triển một hướng tiếp cận mới trong việc xây dựng phần mềm theo hướng dịch vụ bằng cách tổ chức các thành phần phần mềm thành những module nhỏ, phân tách ứng dụng thành các dịch vụ nhỏ hơn. Các module này được coi là những ứng dụng riêng rẽ, được phát triển và triển khai độc lập, giao tiếp với nhau thông qua các API. Mỗi Microservice cung cấp một API để cho phép các Microservices khác gọi và tương tác với nó. API định nghĩa các giao thức và các tập lệnh mà các Microservices có thể sử dụng để trao đổi dữ liệu và thực hiện các chức năng cần thiết.

1.2. Kiến trúc Microservices

Kiến trúc Microservices là một kiến trúc phần mềm phân tán, trong đó ứng dụng được xây dựng dưới dạng một tập hợp các dịch vụ nhỏ, độc lập và phối hợp với nhau. Mỗi dịch vụ trong kiến trúc Microservices giải quyết một khía cạnh và chức năng cụ thể của ứng dụng như ghi log, tìm kiếm dữ liệu, đóng vai trò độc lập và có thể được triển khai, mở rộng và quản lý độc lập.



Hình 1.1 Mô hình Microservices

Các thành phần trong kiến trúc Microservices:

Service: Kiến trúc Microservices xây dựng ứng dụng dưới dạng một tập hợp các dịch vụ nhỏ, tồn tại độc lập và có khả năng thực hiện một chức năng cụ thể. Mỗi dịch vụ có thể được triển khai và quản lý độc lập, sử dụng ngôn ngữ lập trình và công nghệ phù hợp cho chức năng của mình. Các dịch vụ trong kiến trúc Microservices giao tiếp với nhau thông qua các giao diện lập trình ứng dụng (API). Các API này định nghĩa các giao thức và tập lệnh mà các dịch vụ sử dụng để trao đổi dữ liệu và thực hiện các chức năng cần thiết. Giao tiếp qua API giúp các dịch vụ hoạt động độc lập và linh hoạt trong việc phát triển và triển khai.

Client: Khách hàng có thể sử dụng giao diện người dùng để tạo yêu cầu. Đồng thời, một hoặc nhiều vi dịch vụ được ủy quyền thông qua cổng API để thực hiện nhiệm vụ được yêu cầu. Kết quả là, ngay cả những vấn đề phức tạp lớn hơn đòi hỏi sự kết hợp của các vi dịch vụ cũng có thể được giải quyết tương đối dễ dàng.

API Gateway: Cổng API là điểm nhận các yêu cầu từ khách hàng. Thay vì yêu cầu trực tiếp các dịch vụ, khách hàng kết nối đến cổng API để chuyển tiếp các yêu cầu đến các dịch vụ thích hợp ở phía sau.

1.3. Những ưu điểm và nhược điểm của mô hình Microservices

1.3.1. Ưu điểm của mô hình Microservices

Với những đặc điểm trong mô hình kiến trúc, hệ thống MS mang đến những lợi ích cho các nhà phát triển và các kỹ sư trong việc phát triển và vận hành mà mô hình monolith không thể mang lại. Những ưu điểm của hệ thống MS bao gồm:

- + Giảm thiểu công sức phát triển
- + Khả năng mở rộng linh hoạt
- + Khả năng triển khai độc lập
- + Khả năng cô lập lỗi
- + Khả năng tích hợp với các công nghệ và ngăn xếp công nghệ khác nhau

1.3.2. Nhược điểm của mô hình Microservices

Bên cạnh những lợi ích mà mô hình kiến trúc MS mang lại thì kèm theo đó là những nhược điểm của kiến trúc này. Những nhược điểm của hệ thống MS bao gồm:

- + Sự phức tạp trong việc quản lý và triển khai
- + Chi phí phát triển và vận hành

1.4. Khả năng ứng dụng của mô hình Microservices

1.4.1. Ứng dụng web và di động

Kiến trúc Microservices thích hợp cho việc xây dựng các ứng dụng web và di động phức tạp. Với việc phân tách các chức năng và logic kinh doanh thành các dịch vụ độc lập, các nhóm phát triển có thể làm việc độc lập trên từng dịch vụ. Điều này giúp tăng tốc độ phát triển và khả năng mở rộng của hệ thống. Ví dụ, một ứng dụng thương mại điện tử có thể sử dụng các dịch vụ riêng biệt cho quản lý sản phẩm, quản lý đơn hàng, thanh toán và xác thực người dùng.

1.4.2. Hệ thống tài chính và ngân hàng

Các hệ thống tài chính và ngân hàng thường có nhiều chức năng và dịch vụ phức tạp. Kiến trúc Microservices cho phép phân tách các chức năng như quản lý tài khoản, quản lý giao dịch, xử lý thanh toán và phân tích dữ liệu thành các dịch vụ độc lập. Điều này giúp tăng tính linh hoạt và khả năng mở rộng của hệ thống, đồng thời giảm thiểu sự phụ thuộc giữa các phần tử.

1.4.3. Hệ thống y tế

Trong lĩnh vực y tế, kiến trúc Microservices có thể được áp dụng để tách riêng các chức năng như quản lý bệnh nhân, quản lý lịch hẹn, xử lý hồ sơ y tế và gửi thông báo. Điều này giúp tăng tính linh hoạt và khả năng mở rộng của hệ thống y tế, đồng thời cho phép tích hợp dữ liệu và tương tác với các ứng dụng và thiết bị y tế khác.

1.5. Kết luận chương 1

Mô hình Microservices với những ưu điểm vượt trội trong việc phát triển phát mềm và vận hành hệ thống đang ngày càng trở nên phổ biến. Nó giúp tăng tính linh hoạt, khả năng mở rộng và sử dụng công nghệ đa dạng giúp các nhà phát triển và vận hành tiết kiệm thời gian và công sức hơn rất nhiều so với các hệ thống triển khai theo mô hình Monolith. Tuy nhiên, mô hình Microservices cũng có những thách thức như việc quản lý và triển khai nhiều dịch vụ độc lập mang lại nhiều sự phức tạp hơn do đó đòi hỏi sự quản lý và triển khai kỹ lưỡng của đội ngũ kỹ sư công nghệ thông tin. Dù vậy thì với những lợi ích lớn mà mô hình Microservices mang lại, nó vẫn hứa hẹn sẽ là kiểu mô hình được áp dụng rộng rãi trong kiến trúc phần mềm trong tương lai.

CHƯƠNG 2 HỆ THỐNG TÍNH CƯỚC TRỰC TUYẾN OCS

Chương 2 tập trung vào nghiên cứu hệ thống tính cước trực tuyến OCS hiện tại đang triển khai theo mô hình Monolith, kiến trúc hệ thống và luồng xử lý tính cước các dịch vụ viễn thông tại Viettel.

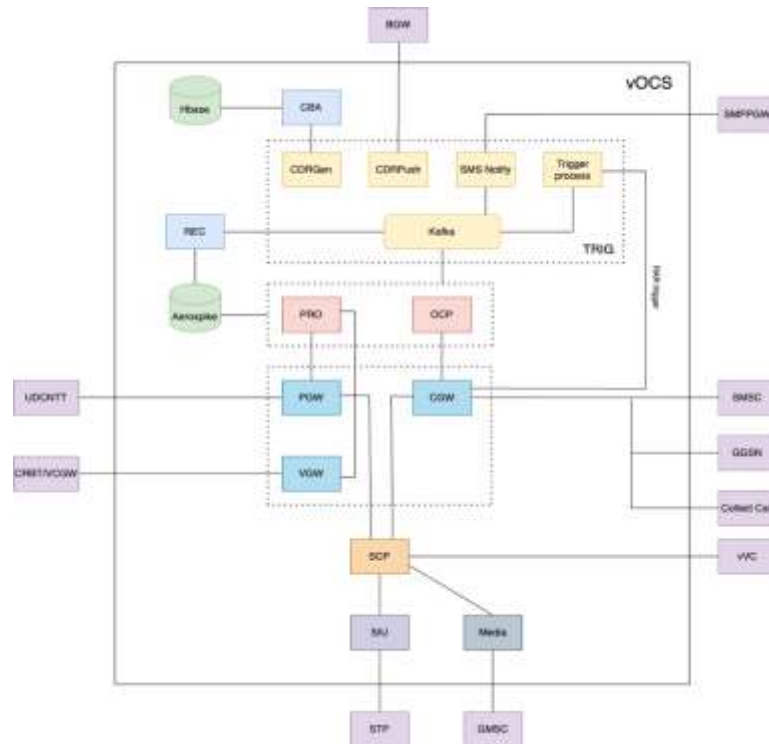
2.1. Tổng quan về hệ thống OCS

Hệ thống tính cước trực tuyến OCS (Online Charging System) là một hệ thống trong lĩnh vực viễn thông và dịch vụ di động, được sử dụng để tính toán và quản lý thông tin về cước phí và sử dụng dịch vụ của khách hàng.

Hệ thống OCS đóng vai trò là trái tim của mạng lưới và thực hiện một số chức năng như sau:

- Quản lý cước phí
- Quản lý tài khoản và sử dụng dịch vụ
- Quản lý thời gian và tương tác với thời gian thực
- Tích hợp với các hệ thống khác
- Quản lý sự kiện và báo cáo

2.2. Kiến trúc hệ thống OCS



Hình 2.1. Mô hình hệ thống OCS theo kiến trúc Monolith

2.2.1. Các thành phần trong hệ thống

SIU (Signaling Interface Unit): cung cấp kết nối giữa mạng lõi SS7 và các máy chủ ứng dụng phân tán.

SCP (Service Control Point): là một tập hợp các cơ sở dữ liệu lưu giữ các thông tin cần thiết để cung cấp các dịch vụ phức tạp hơn so với điều khiển cuộc gọi thông thường (cung cấp VAS, các dịch vụ tổng đài 1800), kết nối mạng SS7 với hệ thống cơ sở dữ liệu.

Media: là thành phần lưu trữ và xử lý các file âm báo của tất cả luồng nghiệp vụ trên OCS.

vVC (Voucher Card): hệ thống quản lý thẻ cào, chứa các thông tin khuyến mại, ưu đãi.

CGW (Charging Gateway): gateway kết nối tới các thành phần tính cước chính trong khối OCU, chuyển đổi các giao thức Diameter/SMPP+ sang giao thức nội bộ trong hệ thống OCS, chuyển các bản tin thành bản tin craMsg.

PGW (Provisioning Gateway): gateway kết nối tới các hệ thống TRAN, VAS, Provisioning, UDCNTT, chuyển đổi các giao thức thành giao thức nội bộ, chuyển các bản tin thành bản tin craMsg.

VGW (VAS Gateway): chuyển tiếp các tin nhắn dành cho các dịch vụ VAS.

OCP (Online Charging Process): chạy theo cơ chế Active-Active, xử lý các bản tin craMsg nhận được từ CGW, thực hiện chức năng tính cước thời gian thực cho tất cả các dịch vụ và điều khiển băng thông cho các dịch vụ data (Mobile data, FTTH, ADSL), tất cả các server có vai trò như nhau và được nhận roundrobin bản tin từ Gateway.

PRO (Provisioning): thực hiện các chức năng đấu nối, đăng ký/hủy thuê bao, cộng/trừ tiền tài khoản, xử lý các dịch vụ VAS, USSD, IVR, Web-Service.

REC (Recurring): thực hiện gia hạn, chuyển đổi vòng đời, xuất dữ liệu thuê bao (trạng thái thuê bao, số dư tài khoản).

TRIG: xử lý các bản tin trigger, chuyển đổi thành CDR, CraMsg, RAR, smsnotify

PCRF (Policy and Charging Rules Function): hỗ trợ detect luồng dịch vụ data, thực thi chính sách và tính cước dựa trên luồng (flow-based), cho phép kiểm soát dịch vụ tốt hơn phù hợp với nguồn lực tài chính.

CBA (Customer Behavior Analysis): quản lý và phân tích thông tin hành vi khách hàng, đưa ra các chính sách hoặc các gói cước phù hợp cho khách hàng.

OAM (Operation and Administration Management): phân hệ hỗ trợ vận hành, thống kê và giám sát toàn bộ tiến trình, node mạng của hệ thống OCS.

2.2.2. Các giao thức trong hệ thống

SMPP (Short Message Peer-to-Peer): giao thức truyền một lượng lớn tin nhắn SMS, được sử dụng để kết nối các hệ thống bên ngoài với trung tâm xử lý lượng lớn SMS (cụ thể là SMSC).

SMPP+: là một phiên bản độc quyền của SMPP dành cho các dịch vụ USSD.

MML (Man-Machine Language): hỗ trợ kết nối tới phân hệ SCU của nghiệp vụ USSD, IVR và hỗ trợ kết nối của các tổng đài TRAN/VAS và tổng đài thẻ nạp.

DCC (Diameter Credit-Control): giao thức Diameter được sử dụng để tính cước thời gian thực cho các dịch vụ người dùng (truy nhập mạng, dịch vụ khởi tạo phiên, dịch vụ tin nhắn và các dịch vụ truy cập dữ liệu khác □ điều khiển tín dụng và giá cước thời gian thực.

Sigtran: truyền các dữ liệu báo hiệu thời gian thực qua mạng IP, cho phép các nút phía mạng IP giao tiếp với các nút phía mạng SS7, cho phép các nút SS7 có thể giao tiếp với nhau qua các link IP.

Gx: kết nối với GGSN, hỗ trợ các nghiệp vụ PCRF.

Gy: kết nối với GGSN, hỗ trợ các nghiệp vụ liên quan đến tính cước nghiệp vụ DATA.

FTP (File Transfer Protocol): trao đổi file CDR với các hệ thống khác.

2.2.3. Các tiến trình trong hệ thống

ProvisioningProcess: phân hệ thực hiện xử lý các nghiệp vụ đầu nối, kích hoạt/hủy thuê bao, đăng ký các dịch vụ VAS đăng ký, thay đổi, hủy gói cước, tạo/hủy nhóm, thêm thành viên, nạp tiền, thay đổi/truy vấn tài khoản, quản lý thông tin zone, các parameter của thuê bao và hệ thống, recurring.

OfflineProcess: thực hiện tính cước offline cho khách hàng hoặc refund bằng cách tạo ra các bản tin tính cước từ CDROffline hoặc CDRRefund.

TriggerProcess: phân loại trigger liên quan đến Notify tới khách hàng và các trigger liên quan đến tạo ra các event tác động tới profile khách hàng (công tích lũy, đăng ký gói cước, đầu ra là các sự kiện thông báo (SMS_Notify, USSD_Notify) để gửi tới phân hệ SMS Notify (bổ sung các thông tin còn thiếu cho bản tin thông báo, phân

loại bản tin thông báo theo thời gian gửi) hoặc các sự kiện tính cước để gửi đến phân hệ Gateway.

CDRGen: xử lý tin nhắn trong hàng đợi xử lý trigger Kafka liên quan đến các CDR Event, tạo CDR file và đẩy đến CBA.

CDRPush: đẩy các file CDR được tạo tới BGW qua giao thức FTP.

SMS Notify: đẩy SMS qua SMSC đến khách hàng đồng thời đẩy tin nhắn đến hàng đợi Kafka trong CBA để lưu trữ lịch sử tin nhắn.

2.2.4. Các cơ sở dữ liệu

MySQL: lưu trữ chính sách nghiệp vụ, cấu hình hệ thống, gói cước, thao tác liên quan đến lịch sử của thuê bao, ưu điểm là miễn phí và đảm bảo được hết các tính năng của 1 cơ sở dữ liệu quan hệ.

Aerospike (NoSQL): lưu trữ thông tin khách hàng (số thuê bao, trạng thái, gói cước, tài khoản, tham số khác).

Hbase: cơ sở dữ liệu có khả năng lưu trữ và truy xuất dữ liệu lớn với hiệu suất cao và khả năng mở rộng linh hoạt nên được dùng để lưu trữ các CDR sinh ra từ các nghiệp vụ tính cước của OCP, PRO, REC.

2.3. Các luồng xử lý tính cước trong hệ thống OCS

2.3.1. Luồng tính cước dịch vụ thoại

Luồng tính cước thoại sẽ đi từ hệ thống mạng lõi MSC đến OCS. Các thành phần trong hệ thống OCS chịu trách nhiệm tính cước thoại bao gồm SCP, CGW, OCP. Cơ sở dữ liệu Aerospike đóng vai trò cung cấp các thông tin của thuê bao như trạng thái thuê bao, gói cước, tài khoản thoại đang sử dụng để tính cước chính xác. Sau khi quá trình tính cước hoàn thành, OCP sẽ gửi các sự kiện sinh CDR đến Kafka để sau đó xử lý sinh ra CDR thoại.

2.3.2. Luồng tính cước dịch vụ SMS

Luồng tính cước SMS sẽ đi từ hệ thống mạng lõi SMSC đến OCS. Các thành phần trong hệ thống OCS chịu trách nhiệm tính cước thoại bao gồm CGW, OCP. Cơ sở dữ liệu Aerospike đóng vai trò cung cấp các thông tin của thuê bao như trạng thái thuê bao, gói cước, tài khoản tin nhắn đang sử dụng để tính cước chính xác. Sau khi quá trình tính cước hoàn thành, OCP sẽ gửi các sự kiện sinh CDR đến Kafka để sau đó xử lý sinh ra CDR SMS.

2.3.3. Luồng tính cước dịch vụ Data

Luồng tính cước Data sẽ đi từ hệ thống mạng lõi GGSN đến OCS. Với luồng tính cước data sẽ tính cước trên mặt phẳng giao diện gy. Các thành phần trong hệ thống OCS chịu trách nhiệm tính cước thoại bao gồm CGW, OCP. Cơ sở dữ liệu Aerospike đóng vai trò cung cấp các thông tin của thuê bao như trạng thái thuê bao, gói cước, tài khoản data đang sử dụng để tính cước chính xác. Sau khi quá trình tính cước hoàn thành, OCP sẽ gửi các sự kiện sinh CDR đến Kafka để sau đó xử lý sinh ra CDR Data.

2.3.4. Luồng tính cước dịch vụ PCRF

Luồng tính cước PCRF sẽ đi từ hệ thống mạng lõi GGSN đến OCS. Với luồng tính cước data sẽ tính cước trên mặt phẳng giao diện gx. Các thành phần trong hệ thống OCS chịu trách nhiệm tính cước thoại bao gồm CGW, OCP. Cơ sở dữ liệu Aerospike đóng vai trò cung cấp các thông tin của thuê bao như trạng thái thuê bao, gói cước, tài khoản PCRF đang sử dụng để tính cước chính xác. Sau khi quá trình tính cước hoàn thành, OCP sẽ gửi các sự kiện sinh CDR đến Kafka để sau đó xử lý sinh ra CDR PCRF.

2.3.5. Luồng tính cước dịch vụ Webservice

Luồng tính cước Webservice sẽ đi từ hệ thống ứng dụng CNTT đến OCS. Các thành phần trong hệ thống OCS chịu trách nhiệm tính cước Webservice bao gồm PGW, PRO. Cơ sở dữ liệu Aerospike đóng vai trò cung cấp các thông tin của thuê bao như trạng thái thuê bao, gói cước, tài khoản gốc đang sử dụng để tính cước chính xác. Sau khi quá trình tính cước hoàn thành, PRO sẽ gửi các sự kiện sinh CDR đến Kafka để sau đó xử lý sinh ra CDR HistBal.

2.4. Kết luận chương 2

Trong chương này đã nêu được tổng quan về hệ thống OCS theo mô hình Monolith bao gồm kiến trúc hệ thống, các thành phần trong hệ thống và luồng xử lý tính cước trực tuyến của các dịch vụ viễn thông đang triển khai tại Viettel hiện tại.

Tuy nhiên, với các lợi ích của mô hình MS đã đề cập trong chương 1 thì việc tích hợp triển khai hệ thống OCS theo mô hình MS thay thế cho mô hình Monolith truyền thống được xem là giải pháp triển khai phù hợp với xu hướng công nghệ tương lai và các đề xuất cũng như giải pháp tích hợp hệ thống OCS với mô hình MS sẽ được đề cập trong nội dung của chương 3.

CHƯƠNG 3 ÁP DỤNG MÔ HÌNH MICROSERVICES VÀO HỆ THỐNG TÍNH CƯỚC TẠI VIETTEL

Chương 3 tập trung nghiên cứu định hướng phát triển hệ thống tính cước hệ thống OCS theo mô hình Microservices thay thế cho mô hình monolith truyền thống. Với các ưu điểm vượt trội mà mô hình Microservices mang đến sẽ giúp hệ thống OCS tăng tính linh hoạt và hoạt động độc lập hơn giữa các dịch vụ với nhau tránh xung đột giữa các thành phần xử lý tính cước trong hệ thống.

3.1. Giới thiệu chung

Hệ thống tính cước trực tuyến OCS đóng vai trò đặc biệt quan trọng, góp phần nâng cao trải nghiệm của khách hàng và mang lại nguồn thu lớn cho các doanh nghiệp viễn thông, công nghệ thông tin. Vì vậy, nếu áp dụng được những ưu điểm của mô hình MS tích hợp vào hệ thống OCS sẽ mang lại những lợi ích vô cùng lớn.

Một số lợi ích khi triển khai hệ thống OCS theo mô hình Microservices bao gồm:

- + Phân tách chức năng
- + Độc lập triển khai và cập nhật
- + Tích hợp linh hoạt
- + Tăng khả năng mở rộng
- + Tăng tính sẵn sàng và khả năng phục hồi

Trong quá trình triển khai mô hình MS trong hệ thống OCS sẽ bao gồm các bước như phân tích chức năng, thiết kế kiến trúc, triển khai dịch vụ, giao tiếp và tích hợp, quản lý và giám sát, quản lý phiên bản và cập nhật, kiểm thử và triển khai, theo dõi và tối ưu.

3.2. Đề xuất hệ thống OCS của Viettel theo mô hình Microservices

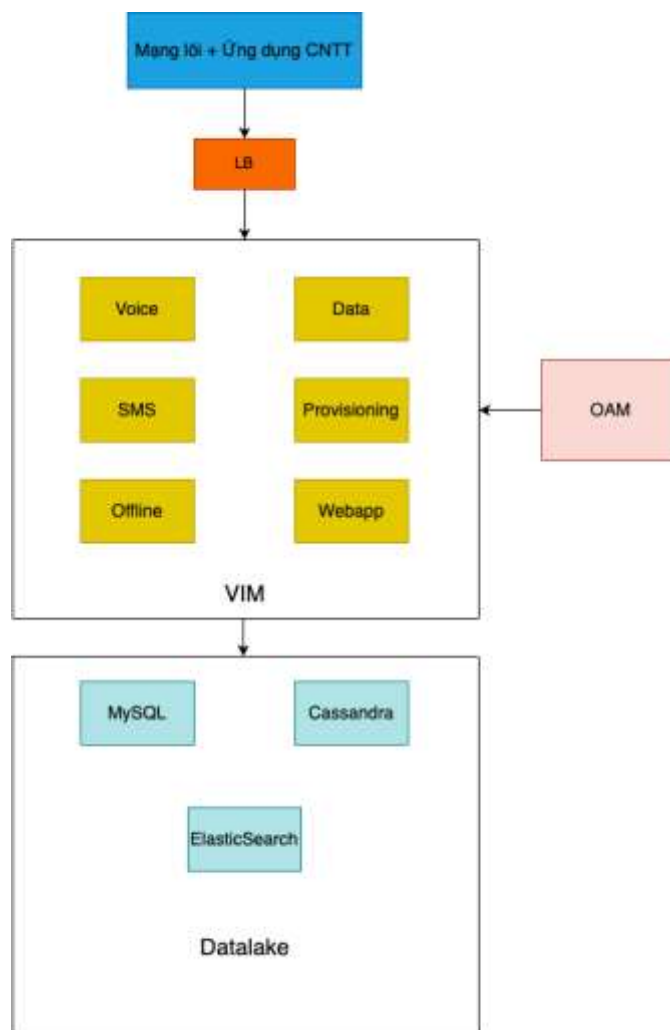
Hệ thống OCS tại Viettel đang triển khai theo mô hình Monolith, các dịch vụ viễn thông sẽ được gộp thành một khối duy nhất để tính cước. Với việc triển khai theo mô hình Monolith truyền thống nên hệ thống OCS có kiến trúc đơn giản, không yêu cầu tính linh hoạt và mở rộng cao giúp đơn giản hoá quá trình triển khai và quản lý.

Tuy nhiên với nhu cầu sử dụng ngày càng cao và phức tạp của khách hàng, yêu cầu giao tiếp của hệ thống OCS với các hệ thống khác ngày càng lớn thì việc triển khai theo mô hình Monolith truyền thống sẽ gặp khó khăn trong việc mở rộng quy mô. Khi tải hệ thống lớn, việc mở rộng mô hình Monolith có thể đòi hỏi triển khai và quản lý

hiều bản sao của toàn bộ ứng dụng, không thể mở rộng chỉ một phần cụ thể. Ngoài ra, các yêu cầu sửa lỗi và nâng cấp liên tục có thể ảnh hưởng đến toàn bộ hệ thống OCS, làm tăng rủi ro và giới hạn tính linh hoạt trong việc triển khai các thay đổi và cải thiện.

Để đáp ứng được những yêu cầu thay đổi này thì việc định hướng phát triển hệ thống theo mô hình Microservices là đặc biệt quan trọng khi sẽ giải quyết được các nhược điểm của mô hình Monolith.

Hệ thống OCS là một trong những hệ thống đặc biệt quan trọng của Viettel. Vì vậy, các giải pháp được đưa ra để tích hợp mô hình Microservices vào hệ thống trở thành mục tiêu hàng đầu của doanh nghiệp.



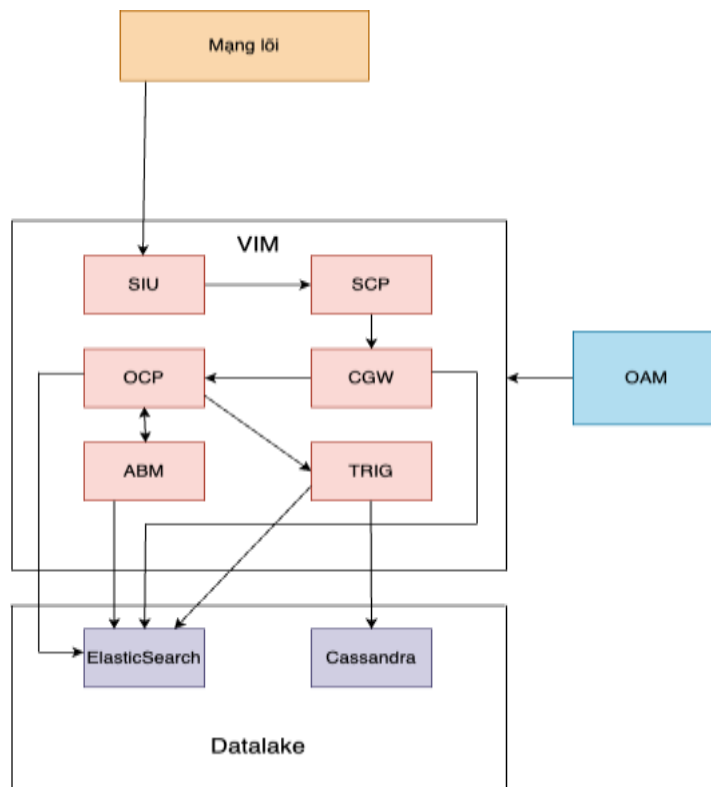
Hình 3.1. Mô hình hệ thống OCS theo kiến trúc Microservices được triển khai tại Viettel

Hệ thống OCS triển khai theo mô hình MS sẽ chia các dịch vụ để tính cước thành từng dịch vụ nhỏ để xử lý bao gồm các thành phần:

+ *Khối dịch vụ Voice*: xử lý các nghiệp vụ liên quan đến quá trình tính cước dịch vụ thoại.

- + Khối dịch vụ Data: xử lý các nghiệp vụ liên quan đến quá trình tính cước dịch vụ Data, PCRF.
- + Khối dịch vụ SMS: xử lý các nghiệp vụ liên quan đến quá trình tính cước dịch vụ SMS.
- + Khối dịch vụ Provisioning: xử lý các nghiệp vụ liên quan đến quá trình quản lý thông tin thuê bao.
- + Khối dịch vụ Offline: xử lý các nghiệp vụ liên quan đến quá trình gia hạn gói cước, tài khoản sử dụng của thuê bao.
- + Khối dịch vụ Webapp: xử lý các nghiệp vụ liên quan đến quá trình vận hành khai thác các dịch vụ trên Web như cấu hình chính sách kinh doanh (khai báo, thay đổi chính sách gói cước), tra cứu thông tin, lịch sử sử dụng dịch vụ của thuê bao.
- + Khối lưu trữ Datalake: xử lý các nghiệp vụ liên quan đến quá trình lưu trữ bản ghi cước dịch vụ (CDR), log tập trung của dịch vụ, chính sách kinh doanh của hệ thống.
- + Khối giám sát OAM: xử lý các nghiệp vụ liên quan đến quá trình giám sát tiến trình bao gồm CPU, RAM, tỷ lệ mã lỗi, tỷ lệ xử lý bản tin thành công, số lượng bản tin đầu vào.

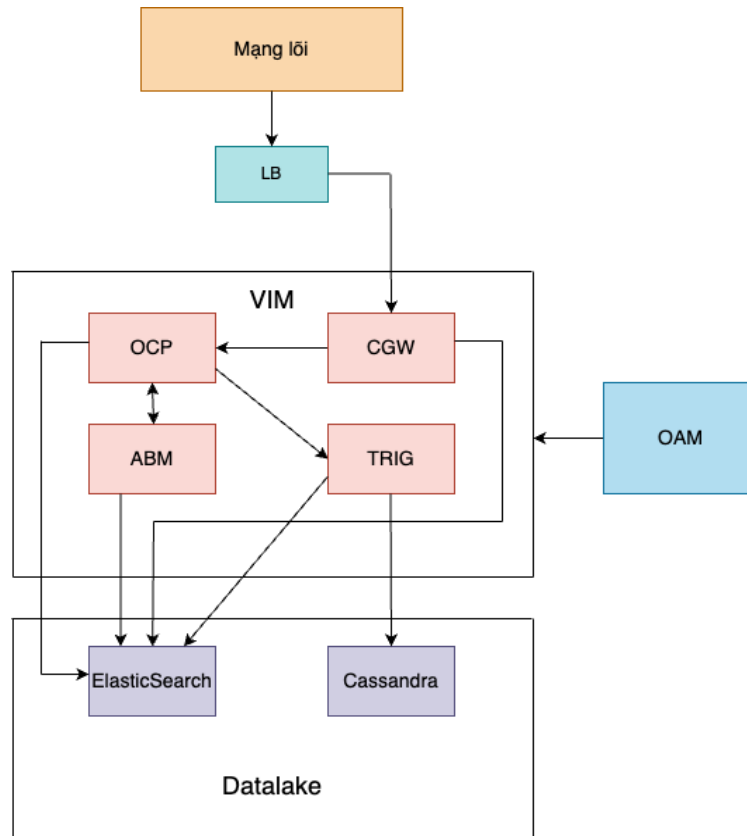
3.2.1. Khối dịch vụ Voice



Hình 3.2. Mô hình khối dịch vụ Voice theo kiến trúc Microservices

Khối dịch vụ Voice xử lý các nghiệp vụ liên quan đến quá trình tính cước dịch vụ thoại, sinh các bản ghi CDR thoại để lưu trữ vào trong cơ sở dữ liệu Cassandra phục vụ cho việc tra cứu các bản ghi CDR xử lý khiếu nại của khách hàng hay đối soát cước chênh lệch giữa hệ thống OCS với hệ thống khác. Ngoài ra, các log xử lý của từng thành phần trong khối sẽ được lưu trữ vào cơ sở dữ liệu ElasticSearch, sử dụng cho việc tra cứu quá trình xử lý trong trường hợp có lỗi xảy ra.

3.2.2. Khối dịch vụ Data



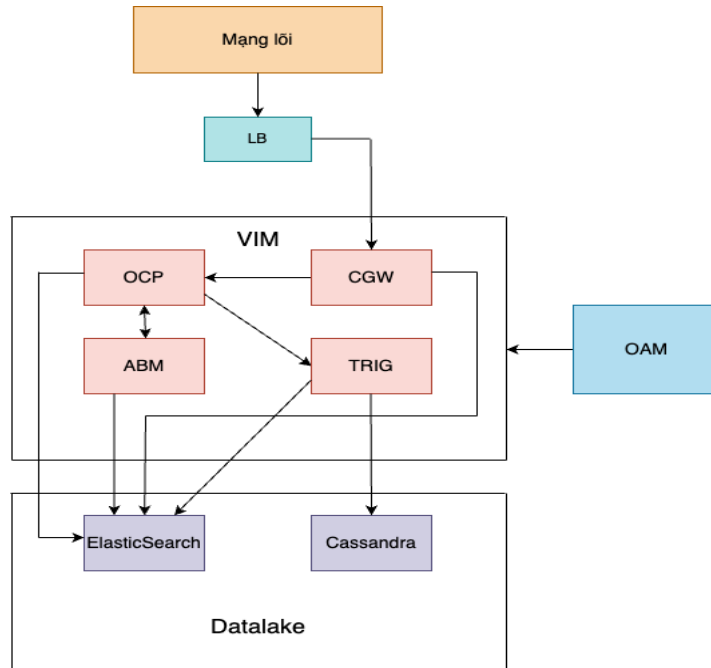
Hình 3.3. Mô hình khối dịch vụ Data theo kiến trúc Microservices

Khối dịch vụ Data xử lý các nghiệp vụ liên quan đến quá trình tính cước dịch vụ data và pcrf, sinh các bản ghi CDR data để lưu trữ vào trong cơ sở dữ liệu Cassandra phục vụ cho việc tra cứu các bản ghi CDR xử lý khiếu nại của khách hàng hay đối soát cước chênh lệch giữa hệ thống OCS với hệ thống khác tương tự như với chức năng của khối dịch vụ Voice. Các log xử lý của từng thành phần trong khối cũng sẽ được lưu trữ vào cơ sở dữ liệu ElasticSearch, sử dụng cho việc tra cứu quá trình xử lý trong trường hợp có lỗi xảy ra.

3.2.3. Khối dịch vụ SMS

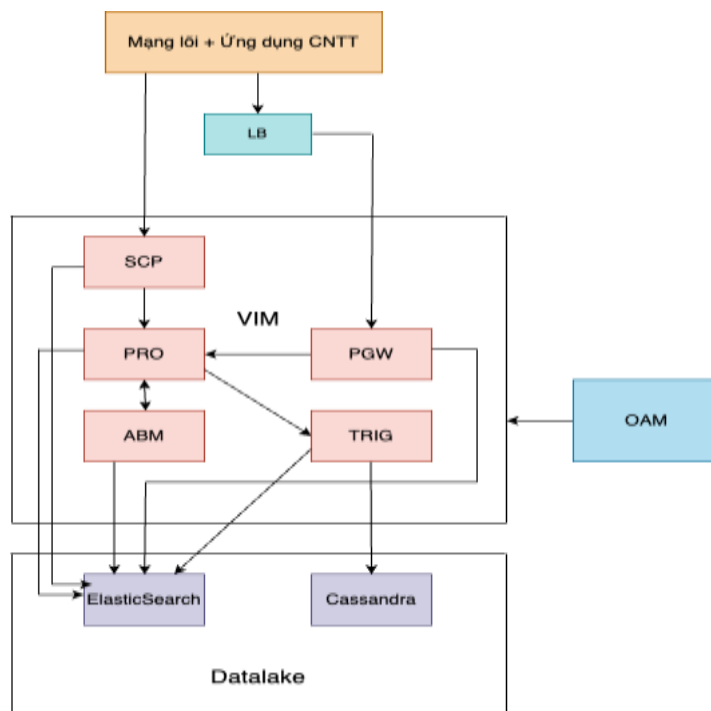
Khối dịch vụ SMS xử lý các nghiệp vụ liên quan đến quá trình tính cước dịch vụ tin nhắn SMS và MMS, sinh các bản ghi CDR sms, mms để lưu trữ vào trong cơ sở dữ

liệu Cassandra phục vụ cho việc tra cứu các bản ghi CDR xử lý khiếu nại của khách hàng hay đối soát cước chênh lệch giữa hệ thống OCS với hệ thống khác tương tự như với chức năng của khối dịch vụ Voice và Data. Các log xử lý của từng thành phần trong khối cũng sẽ được lưu trữ vào cơ sở dữ liệu ElasticSearch, sử dụng cho việc tra cứu quá trình xử lý trong trường hợp có lỗi xảy ra.



Hình 3.4. Mô hình khối dịch vụ SMS theo kiến trúc Microservices

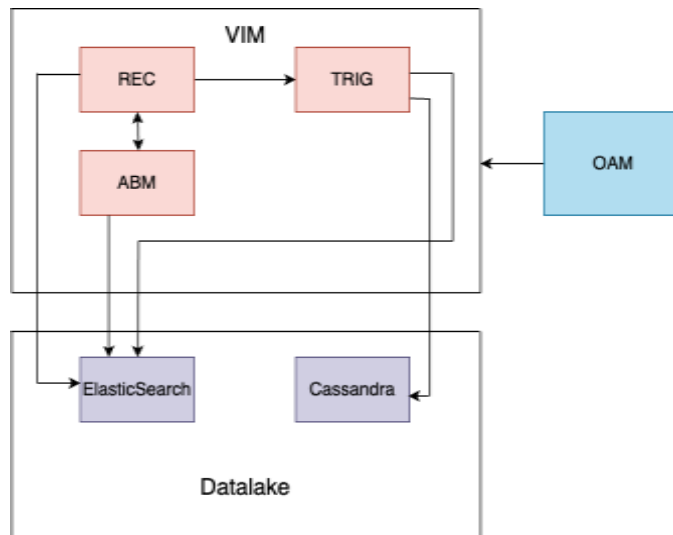
3.2.4. Khối dịch vụ Provisioning



Hình 3.5. Mô hình khối dịch vụ Provisioning theo kiến trúc Microservices

Khối dịch vụ Provisioning xử lý các nghiệp vụ liên quan đến quá trình quản lý thông tin thuê bao bao gồm trạng thái thuê bao, các phiên truy cập đang tồn tại, các gói cước đang sử dụng và tài khoản tương ứng với từng gói. Cũng giống như các khối dịch vụ tính cước trực tuyến, khối Provisioning cũng sinh các bản ghi CDR để lưu trữ vào trong cơ sở dữ liệu Cassandra phục vụ cho việc tra cứu các bản ghi CDR xử lý khiếu nại của khách hàng hay đối soát cước chênh lệch giữa hệ thống OCS với hệ thống khác. Ngoài ra, các log xử lý của từng thành phần trong khối cũng sẽ được lưu trữ vào cơ sở dữ liệu ElasticSearch, sử dụng cho việc tra cứu quá trình xử lý trong trường hợp có lỗi xảy ra.

3.2.5. Khối dịch vụ Offline



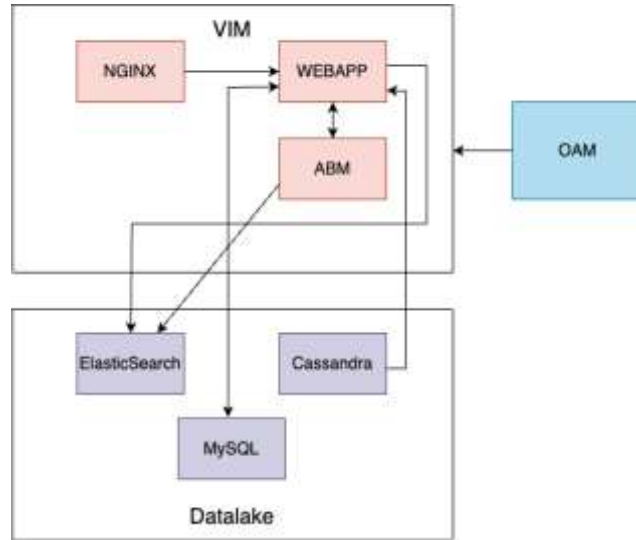
Hình 3.6. Mô hình khối dịch vụ Offline theo kiến trúc Microservices

Khối dịch vụ Offline xử lý các nghiệp vụ liên quan đến quá trình gia hạn gói cước, tài khoản sử dụng của thuê bao theo các chu kỳ gia hạn đã được cấu hình trên web chính sách kinh doanh. Khối Offline sau khi hoàn thành quá trình gia hạn sẽ sinh ra các CDR để lưu trữ vào trong cơ sở dữ liệu Cassandra. Tương tự với các khối dịch vụ khác trong hệ thống OCS, các CDR này phục vụ cho việc tra cứu xử lý khiếu nại của khách hàng hay đối soát cước chênh lệch giữa hệ thống OCS với hệ thống khác. Với mỗi quá trình xử lý của từng thành phần trong khối sẽ tạo ra log và được lưu trữ vào cơ sở dữ liệu ElasticSearch, sử dụng cho việc tra cứu quá trình xử lý trong trường hợp có lỗi xảy ra.

3.2.6. Khối dịch vụ Webapp

Khối dịch vụ Webapp xử lý các nghiệp vụ liên quan đến quá trình vận hành khai thác các dịch vụ trên Web như cấu hình chính sách kinh doanh (khai báo, thay đổi chính

sách gói cước), tra cứu thông tin, lịch sử sử dụng dịch vụ, tác động thay đổi thông tin của thuê bao.



Hình 3.7. Mô hình khối dịch vụ Webapp theo kiến trúc Microservices

3.3. Đánh giá kết quả thử nghiệm hệ thống OCS theo mô hình Microservices

3.1.1. Những ưu điểm về chỉ tiêu kỹ thuật so với hệ thống cũ

Hệ thống OCS thử nghiệm triển khai theo kiến trúc Microservices đem lại những ưu điểm khi số lượng server cần sử dụng để vận hành chỉ bằng 1/4 so với hệ thống OCS theo mô hình Monolith, giúp tiết kiệm chi phí đầu tư và bảo dưỡng định kỳ hàng năm.

Hệ thống mới cũng cho phép tăng năng lực chịu tải của các thành phần trong từng luồng dịch vụ, đáp ứng trải nghiệm dịch vụ của nhiều khách hàng hơn nữa.

Ngoài ra, thời gian phản hồi xử lý của các module cũng là điểm nổi bật khi so sánh giữa hai hệ thống. Với hệ thống OCS mới cho phép thời gian phản hồi của từng quá trình xử lý ít hơn rất nhiều, từ đó xử lý được nhiều tài giao dịch hơn.

3.1.2. Kết quả đạt được trong quá trình thử nghiệm hệ thống

Hệ thống OCS mới tại Viettel tích hợp triển khai theo mô hình Microservices đang trong giai đoạn thử nghiệm và tiếp tục phát triển. Hiện tại, hệ thống mới đang cung cấp dịch vụ tính cước, điều khiển băng thông đáp ứng cho 02 triệu thuê bao cắt chuyển từ hệ thống cũ và phục vụ đánh giá các yêu cầu tính năng theo bộ chỉ tiêu kỹ thuật để làm căn cứ báo cáo đầu tư hệ thống có năng lực xử lý toàn bộ thuê bao trên hệ thống chính.

Với những ưu điểm của mô hình Microservices, hệ thống OCS mới triển khai theo mô hình này đang mang lại những thuận lợi như hạn chế lỗi đa dịch vụ hoặc toàn mạng,

luồng tính cước dịch vụ tường minh và thuận lợi cho công tác vận hành khai thác. Bên cạnh đó, nó còn tối ưu được hiệu năng phần mềm do chỉ xử lý cho từng service cụ thể.

3.4. Kết luận chương 3

Hệ thống tính cước trực tuyến OCS là một trong những hệ thống đặc biệt quan trọng của mỗi doanh nghiệp viễn thông, công nghệ thông tin. Do vậy, việc triển khai thử nghiệm tích hợp hệ thống OCS với mô hình MS là xu hướng tất yếu trong lĩnh vực công nghệ phần mềm. Những ưu điểm mà mô hình MS mang lại như việc phân tách độc lập giữa các dịch vụ với nhau, tăng tính sẵn sàng trong việc cập nhật phiên bản mới hoặc xử lý lỗi đã giúp các dịch vụ viễn thông trong hệ thống OCS linh hoạt hơn, thuận lợi hơn cho công tác phát triển và vận hành khai thác tại Viettel.

KẾT LUẬN

Mô hình Microservices đã trở thành một xu hướng phổ biến trong việc phát triển các hệ thống phần mềm phức tạp. Bằng cách chia nhỏ ứng dụng thành các dịch vụ nhỏ, độc lập và có khả năng hoạt động độc lập, mô hình MS mang lại nhiều lợi ích quan trọng như tăng tính linh hoạt và khả năng mở rộng, cho phép sử dụng công nghệ và ngôn ngữ phù hợp nhất cho từng dịch vụ, tạo điều kiện cho việc phát triển phân tán và mở rộng ngang.

Trong khi đó, hệ thống tính cước trực tuyến OCS là một hệ thống đóng vai trò quan trọng trong việc tính toán cước phí dịch vụ viễn thông và quản lý thông tin thuê bao nên việc tận dụng những ưu điểm mà mô hình Microservices mang lại sẽ đem đến nhiều giá trị như hạn chế lỗi đa dịch vụ, dễ dàng khoanh vùng lỗi khi xảy ra và thuận lợi cho công cuộc vận hành khai thác tại Viettel.

Do thời gian nghiên cứu còn có hạn nên nhiều vấn đề trong đề án có thể chưa sâu sắc và đầy đủ, vì vậy tôi mong muốn nhận được sự thông cảm và đóng góp quý báu của các Thầy, Cô để bổ sung cho những nghiên cứu sau này.

Hướng phát triển tiếp theo của Đề án là hoàn thiện và phát triển hệ thống OCS tích hợp mô hình Microservices cho toàn bộ thuê bao trong mạng lưới của Viettel trong tương lai.