

TÀI LIỆU

THIẾT KẾ HỆ THỐNG

DỰ ÁN ESB- NGÂN HÀNG MARITIME BANK

|  |  |
| --- | --- |
| Ngày hoàn thành | 30/03/2013 |
| Người lập | Nguyễn Tuấn Anh  Ngô Xuân Phong |
| Người xem xét | Ngô Xuân Phong  Quản trị dự án |

**Hà nội, tháng 03/2013**

Contents

[1 Thiết kế tổng thể 5](#_Toc355184223)

[1.1 Mô hình logic hệ thống 5](#_Toc355184224)

[1.2 Mô hình kiến trúc hệ thống 5](#_Toc355184225)

[1.2.1 Ứng dụng sử dụng dịch vụ 6](#_Toc355184226)

[1.2.2 Hệ thống tích hợp 6](#_Toc355184227)

[1.2.3 Ứng dụng cung cấp dịch vụ 7](#_Toc355184228)

[1.2.4 Các ứng dụng gateway (kết nối tới các ứng dụng bên ngoài MSB) 7](#_Toc355184229)

[1.3 Kiến trúc trao đổi dữ liệu 8](#_Toc355184230)

[1.3.1 Các thành phần tiến trình xử lý chính (Message flow) trong trục tích hợp 8](#_Toc355184231)

[1.3.2 Mô tả quy trình trao đổi dữ liệu của hệ thống 9](#_Toc355184232)

[2 Thiết kế chi tiết ứng dụng theo dõi, tra cứu, đối chiếu 10](#_Toc355184233)

[2.1 Thiết kế CSDL 10](#_Toc355184234)

[2.1.1 Bảng ESB\_PARAM 10](#_Toc355184235)

[2.1.2 Bảng ESB\_ERRDEF 10](#_Toc355184236)

[2.1.3 Bảng ESB\_APP 10](#_Toc355184237)

[2.1.4 Bảng ESB\_TRANS 11](#_Toc355184238)

[2.1.5 Bảng ESB\_ROUTING 12](#_Toc355184239)

[2.1.6 Bảng ESB\_SERVICE\_URL 12](#_Toc355184240)

[2.1.7 Bảng ESB\_TLLOG 12](#_Toc355184241)

[2.1.8 Bảng ESB\_MSGLOG 13](#_Toc355184242)

[2.1.9 Bảng ESB\_ERRLOG 14](#_Toc355184243)

[2.1.10 Bảng ESB\_CHANNEL 15](#_Toc355184244)

[2.1.11 Bảng ESB\_SEQMGR 15](#_Toc355184245)

[2.1.12 Bảng ESB\_CHUCNANG 16](#_Toc355184246)

[2.1.13 Bảng ESB\_CNANG\_NHOM 16](#_Toc355184247)

[2.1.14 Bảng ESB\_NHOM\_NSD 17](#_Toc355184248)

[2.1.15 Bảng ESB\_NSD 17](#_Toc355184249)

[2.1.16 Bảng ESB\_NSD\_NHOM 18](#_Toc355184250)

[2.2 Ứng dụng theo dõi, tra cứu, đối chiếu 18](#_Toc355184251)

[2.2.1 Quản trị người dùng 18](#_Toc355184252)

[2.2.2 Theo dõi, tra cứu 19](#_Toc355184253)

[2.2.3 Đối chiếu với giao dịch trong core 19](#_Toc355184254)

[3 Thiết kế chi tiết các luồng trao đổi, tích hợp 19](#_Toc355184255)

[3.1 Khuôn dạng MQ message chung 19](#_Toc355184256)

[3.1.1 MQ in message 19](#_Toc355184257)

[3.1.2 MQ out message 22](#_Toc355184258)

[3.2 Thiết kế MainFlow 23](#_Toc355184259)

[3.3 Thiết kế flow ghi log và flow xử lý exception 24](#_Toc355184260)

[3.3.1 Flow ghi log 24](#_Toc355184261)

[3.3.2 Flow xử lý exception 24](#_Toc355184262)

[3.4 Thiết kế chi tiết flow cho các ứng dụng cung cấp dịch vụ 24](#_Toc355184263)

[3.4.1 Flow cho Core bank (cung cấp API dạng TCP/IP socket) 24](#_Toc355184264)

[3.4.2 Flow cho RSA AM (cung cấp API dạng Java Call) 25](#_Toc355184265)

[3.4.3 Flow cho các ứng dụng SMS banking, SMS OTP, Fund transfer code, Fee, eBank (cung cấp API dạng web service) 25](#_Toc355184266)

[3.4.4 Flow cho Tsunami (cung cấp API dạng HTTP request) 27](#_Toc355184267)

[3.5 Thiết kế chi tiết flow cho các ứng dụng sử dụng dịch vụ 28](#_Toc355184268)

[3.5.1 Starfish 28](#_Toc355184269)

[3.5.2 MPayroll 29](#_Toc355184270)

[3.5.3 MTrading 29](#_Toc355184271)

[3.5.4 SMS banking 29](#_Toc355184272)

[3.5.5 Tsunami 30](#_Toc355184273)

[3.5.6 eBank 31](#_Toc355184274)

[3.5.7 eTax 33](#_Toc355184275)

[3.5.8 SMS Gateway 33](#_Toc355184276)

[3.5.9 Service Payment Gateway 34](#_Toc355184277)

[3.6 Thiết kế chi tiết Tax Gateway 34](#_Toc355184278)

[4 Phụ lục: các XSD cho từng flow của ứng dụng cung cấp dịch vụ 35](#_Toc355184279)

[4.1 Core bank (core.xsd) 35](#_Toc355184280)

[4.2 eBank (eBank.xsd) 52](#_Toc355184281)

[4.3 Fee (fee.xsd) 54](#_Toc355184282)

[4.4 SMS banking (SMSBanking.xsd) 56](#_Toc355184283)

[4.5 Tsunami (Tsunami.xsd) 57](#_Toc355184284)

[4.6 SMS OTP (otp.xsd) 58](#_Toc355184285)

[4.7 Fund Transfer code (Transfercode.xsd) 59](#_Toc355184286)

[4.8 RSA AM (rsa.xsd) 61](#_Toc355184287)

# Thiết kế tổng thể

## Mô hình logic hệ thống



## Mô hình kiến trúc hệ thống



Chúng ta xây dựng hệ thống tích hợpthực hiện việc trao đổi thông tin giữa các ứng dụng cung cấp dịch vụ, ứng dụng sử dụng dịch vụ và các gateway.

### Ứng dụng sử dụng dịch vụ

Đây là ứng dụng yêu cầu sử dụng dịch vụ của ứng dụng cung cấp dịch vụ thông qua hệ thống tích hợp.Trong ứng dụng sử dụng dịch vụ có 1 thành phần thực hiện việc tạo yêu cầu dưới dạng message rồi gửi tới hệ thống tích hợp thông qua giao thức MQ, Đông thời nhận kết quả trả về cũng thông qua giao thực MQ từ hệ thống tích hợp.

### Hệ thống tích hợp

Đây là hệ thống được xây dựng trên nền tảng chính là IBM Websphere Message Broker V8.Nó thực hiện chức năng nhận yêu cầu từ ứng dụng sử dụng dịch vụ dưới sạng message.Sau đó nó thực hiện chuyển đổi dữ liệu thành dạng dữ liệu phù hợp với yêu cầu của ứng dụng cung cấp dịch vụ tương ứng rồi gửi yêu cầu tới ứng dụng cung cấp dịch vụ.Sau nó nhận kết quả trả về từ ứng dụng cung cấp dịch vụ rồi thực hiện chuyển ngược lại thành message phù hợp rồi gửi về cho ứng dụng sử dụng dịch vụ.Hệ thống tích hợp được thiết kế gồm 1 số thành phần chính sau

1. ***Các luồng xử lý nghiệp vụ (message flow)***

Nó bao gồm các message flow thực hiện việc:

* Xử lý, chuyển đổi các loại message để phù hợp với việc trao đổi dữ liệu giữa ứng dụng sử dụng và ứng dụng cung cấp dịch vụ.
* Thực hiện việc định tuyến tới ứng dụng cung cấp dịch vụ tương ứng với ứng dụng sử dụng dịch vụ yêu cầu thông qua nội dung message đầu vào kết hợp với tham số lấy từ CSDL.Cũng như trả kết quả về cho ứng dụng sử dụng dịch vụ tương ứng với yêu cầu của nó.
* Trong quá trình thực hiện trao đổi.Hệ thống tích hợp sẽ thực hiện việc ghi log giao dịch qua lại hệ thống vào CSDL để phục vụ việc theo, dõi đối soát sau này.Nếu bất kỳ lỗi nào xảy ra(Hệ thống cũng như nghiệp vụ) trong quá trình thực hiện.Hệ thống sẽ ghi thông tin lỗi vào CSDL hoặc đưa ra Queue để tiện theo dõi, quản trị hệ thống.Đồng thời mọi xử lý sẽ được rollback về trạng thái ban đầu.Tùy theo yêu cầu mà Message sẽ được đưa ra 1 queue riêng để xử lý sau hoặc được huy bỏ và thông báo về cho ứng dụng sử dụng dịch vụ trạng thái không thành công của giao dịch và thực hiện việc ghi thông tin log của message thông báo không thành công vào CSDL.
* Thực hiện việc queing lại các message theo một độ trễ được khai báo trước, nhằm mục đích giảm tải cho hệ thống cung cấp dịch vụ trong trường hợp tần suất gửi yêu cầu tới quá khả năng đáp ứng của ứng dụng cung cấp dịch vụ.

1. ***Cơ sở dữ liệu***

Thực hiện việc các công việc chính sau:

* Lưu trữ Log giao dịch đi qua hệ thống, cũng như thông tin lỗi của hệ thống khi xảy ra.
* Chứa thông tin tham số hóa hệ thống nhằm mục đích phục vụ quá trình xử lý của các message flow.CSDL này tốt nhất được đặt trên cùng 1 máy với Message Broker để nâng cao hiệu suất đọc ghi dữ liệu của hệ thống tích hợp.
* Thực hiện truy vấn vào CSDL Core thông qua gateway để lấy thông tin giao dịch vào,ra core(đảm bảo reconcile).Công việc này thực hiện ngày 1 lần vào giờ thấp điểm(đêm chẳng hạn).

1. ***Ứng dụng theo dõi hệ thống(Monitor client)***

Đây là ứng dụng client thực hiện việc theo dõi, quản lý các thông tin giao dịch đi qua hệ thống tích hợp, cũng như tình trạng lỗi của hệ thống.

### Ứng dụng cung cấp dịch vụ

Đây là ứng dụng thực hiện đáp ứng yêu cầu của ứng dụng sử dụng dịch vụ thông qua hệ thống tích hợp. Các ứng dụng cung cấp dịch vụ sẽ đưa ra các kêt nối WS, RSA, Socket để hệ thống tích hợp giao tiếp với nó. Nó thực hiện việc nhận yêu cầu từ hệ thống tích hợp gửi đến và trả lại kết quả cho ứng dụng yêu cầu sử dụng dịch vụ thông qua hệ thống tích hợp.

### Các ứng dụng gateway (kết nối tới các ứng dụng bên ngoài MSB)

Đây là các ứng dụng được đặt ở vùng DMZ. Làm cầu nối giữa cá hệ thống bên ngoài và các hệ thống ứng dụng bên trong MSB. Các ứng dụng gateway kế nối tới các ứng dụng cung cấp dịch vụ đều thông qua hệ thống tích hợp (ESB) sử dụng MQ. Ngược lại, các ứng dụng trong MSB kết nối tới gateway được giữ nguyên cách thức như hiện tại tức là kết nối trực tiếp thông qua Web Services.

## Kiến trúc trao đổi dữ liệu



### Các thành phần tiến trình xử lý chính (Message flow) trong trục tích hợp

Các thành phần xử lý chính (message low) gồm 1 số message flow cơ bản sau:

1. ***Message flow chính (Main Flow):***

Các flow này làm nhiệm vụ:

* Nhận yêu cầu từ ứng dụng sử dụng dịch vụ.Xử lý, chuyển đổi các loại message để phù hợp với việc trao đổi dữ liệu giữa ứng dụng sử dụng và ứng dụng cung cấp dịch vụ.
* Thực hiện việc định tuyến tới ứng dụng cung cấp dịch vụ tương ứng với ứng dụng sử dụng dịch vụ yêu cầu thông qua nội dung message đầu vào kết hợp với tham số lấy từ CSDL.Cũng như trả kết quả về cho ứng dụng sử dụng dịch vụ tương ứng với yêu cầu của nó.
* Trong quá trình thực hiện trao đổi.Hệ thống tích hợp sẽ thực hiện việc ghi log giao dịch qua lại hệ thống và CSDL bằng cách gửi thông tin đến 1 message subflow thực hiện ghi log(Log Writing Message Flow).
* Nếu bất kỳ lỗi nào xảy ra(Hệ thống cũng như nghiệp vụ) trong quá trình thực hiện.Hệ thống sẽ ghi thông tin lỗi vào CSDL hoặc đưa ra Queue bằng cách gửi thông tin lỗi đến 1 message subflow thực hiện ghi lỗi vào CSDL hoặc đưa ra queue(Exception Handling Message Flow).

1. ***Message flow thực hiện ghi log (Log Writing Message Flow):***

Log Writing Flow được thiết kế để cho tất cả message flow nghiệp vụ có thể sử dụng chúng.Nó làm nhiệm vụ thực hiện việc ghi, cập nhật thông tin giao dịch đi qua hệ thống tích hợp khi có yêu cầu từ các message flow nghiệp vụ.

1. ***Message flow thực hiện xử lý lỗi (Exception Handling Message Flow):***

Exception Handling Message Flow được thiết kế dùng dưới dạng subflow tất cả message flow nghiệp vụ có thể sử dụng chúng.Nó làm nhiệm vụ thực hiện việc ghi thông tin lỗi vào CSDL hoặc đưa ra Queue khi có yêu cầu từ các message flow nghiệp vụ.Đồng thời, tùy theo yêu cầu nghiệp vụ mà nó thực hiện đưa message gốc ra 1 queue chờ xử lý sau, hoặc hủy bỏ message gốc.Đồng thời gửi message kết quả không thành công về cho ứng dụng sử dụng dịch vụ, gọi Log Writing Message Flow thực hiện ghi thông tin message không thành công(có thông tin tham chiếu ứng với message yêu cầu) vào CSDL.

### Mô tả quy trình trao đổi dữ liệu của hệ thống

1. Ứng dụng sử dụng dịch vụ tạo message yêu cầu rồi gửi đến hệ thống tích hợp thông qua giao thức MQ.
2. Hệ thống tích hợp nhận message yêu cầu từ ứng dụng sử dụng dịch vụ.Sau đó, nó thực hiện gọi flow ghi log đánh dấu giao dịch đi qua hệ thống(2’).Rồi thực hiện xác định thông tin ứng dụng cung cấp dịch vụ,chuyển đổi message tương ứng với ứng dụng cung cấp dịch vụ, gửi yêu cầu tới ứng dụng cung cấp dịch vụ.
3. Ứng dụng cung cấp dịch vụ nhận yêu cầu từ hệ thống tích hợp, thực hiện xử lý, trả kết quả về cho hệ thống tích hợp.
4. Hệ thống tích hợp nhận kết quả trả về từ ứng dụng cung cấp dịch vụ
5. Hệ thống tích hợp thực hiện ghi log message kết quả vào csdl(5’), chuyển đổi message thành dạng message phù hợp rồi gửi về cho ứng dụng sử dụng dịch vụ.
6. Ứng dụng sử dụng dịch vụ nhận mesasge kết quả rồi xử lý nghiệp vụ tiếp.

Trong toàn bộ quá trình trao đổi tại hệ thống tích hợp.Nếu bất kỳ lỗi nào xảy ra(Hệ thống cũng như nghiệp vụ) trong quá trình thực hiện. Hệ thống sẽ ghi thông tin lỗi vào CSDL hoặc đưa ra Queue để tiện theo dõi, quản trị hệ thống.Đồng thời mọi xử lý sẽ được rollback về trạng thái ban đầu.Tùy theo yêu cầu mà Message gốc sẽ được đưa ra 1 queue riêng để xử lý sau hoặc được huy bỏ , tiếp theo nó gửi thông báo về cho ứng dụng sử dụng dịch vụ trạng thái không thành công của giao dịch và thực hiện việc ghi thông tin log của message thông báo không thành công vào CSDL.

# Thiết kế chi tiết ứng dụng theo dõi, tra cứu, đối chiếu

## Thiết kế CSDL

### Bảng ESB\_PARAM

* Mục đích: Lưu các tham số của ứng dụng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Not Null | Mô tả |
| 1 | GRP\_NAME | VARCHAR2(5) |  | x | Nhóm tham số  +SYS : nhóm tham số hệ thống không cho phép sửa ở giao diện  +URD : nhóm tham số do người dùng định nghĩa cho phép sửa ở giao diện |
| 2 | P\_NAME | VARCHAR2(50) | PK,Unique | x | Tên tham số |
| 3 | P\_VALUE | VARCHAR2(100) |  | x | Giá trị |
| 4 | P\_DESC | VARCHAR2(200) |  |  | Mô tả của tham số |
| 5 | LSTORD | NUMBER |  |  | Thứ tự hiển thị |

### Bảng ESB\_ERRDEF

* Mục đích: Định nghĩa các mã lỗi sử dụng trong chương trình,có thể mapping từ mã lỗi của hệ thống nguồn sang mã lỗi của hệ thống đích

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | AUTOID | NUMBER | PK |  | Mã số lỗi |
| 2 | SRC\_ERR\_NUM | NVARCHAR2 |  |  | Mã lỗi từ chương trình được gọi trả về (có thể là số hoặc chữ) |
| 3 | SRC\_ERR\_DESC | NVARCHAR2 |  |  | Mô tả lỗi do chương trình được gọi trả về |
| 4 | DES\_ERR\_NUM | NVARCHAR2 |  |  | Mã lỗi trả về cho chương trình đích |
| 5 | DES\_ERR\_DESC | NVARCHAR2 |  |  | Mô tả lỗi trả về cho chương trình đích |

### Bảng ESB\_APP

* Mục đích: Định nghĩa các ứng dụng kết nối đến ESB

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | APP\_ID | VARCHAR2(5) | PK |  | Mã ứng dụng  +SMS : sms  +IBS : ebank  +SML : payment gateway |
| 2 | APP\_NAME | VARCHAR2(200) |  |  | Tên ứng dụng |

|  |  |
| --- | --- |
| **APP\_ID** | **NAME** |
| SMS | SMS banking |
| IBS | eBank |
| SF | Starfish |
| CB | Core Bank |
| … |  |

### Bảng ESB\_TRANS

* Mục đích: Định nghĩa các giao dịch(method,operation…) được dùng bởi chương trình.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | TRAN\_CODE | VARCHAR2(5) | PK |  | Mã giao dịch |
| 2 | TRAN\_NAME | VARCHAR2(4) |  |  | Tên giao dịch |
| 3 | TRAN\_TYPE | VARCHAR2 |  |  | Loại giao dịch:  +M: method |
| 4 | APP\_ID | VARCHAR2 | FK đến bảng ESB\_APP |  |  |

Ví dụ dữ liệu:

|  |  |  |  |
| --- | --- | --- | --- |
| **TRAN\_CODE** | **TRAN\_NAME** | **TRAN\_TYPE** | **APP\_ID** |
| 1 | regSMSBanking |  | SMS |
| 2 | registerService |  | IBS |
| 3 | createFD |  | CB |
| 4 | linkMasterCard |  | CB |
| 5 | lockUpdateE\_Contact |  | CB |
| 6 | createSA |  | CB |
| 7 | joinSA |  | CB |
| 8 | joinCA |  | CB |
| 9 | searchCustomerById |  | CB |
| 10 | linkCard |  | CB |
| 11 | updateE\_Contact |  | CB |
| 12 | searchCustomerByCif |  | CB |
| 13 | addE\_Contact |  | CB |
| 14 | createCA |  | CB |
| 15 | getAccount |  | CB |
| 16 | createCif |  | CB |
| 17 | searchE\_ContactByCif |  | CB |

### Bảng ESB\_ROUTING

* Mục đích: Với mỗi loại transaction khác nhau sẽ được định tuyến đến các queue khác nhau để xử lý.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | TRAN\_CODE | Varchar2(10) |  |  | Mã giao dịch |
| 2 | SENDER\_ID | Varchar2(20) |  |  | Mã ứng dụng nguồn |
| 3 | RECEIVER\_ID | Varchar2(20) |  |  | Mã ứng dụng đích |
| 4 | QMGR\_NAME | Varchar2(50) |  |  | Tên queue manager |
| 5 | QUEUE\_NAME | Varchar2(50) |  |  | Tên queue |

### Bảng ESB\_SERVICE\_URL

* Mục đích: Định nghĩa các tham số kết nốicủa các dịch vụ sẽ được sử dụng bởi chương trình.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | SERVICE\_ID | VARCHAR2(5) | PK |  | Mã dịch vụ |
| 2 | SERVICE\_TYPE | VARCHAR2(4) |  |  | Loại dịch vụ:  +W: webservice  +T: TCP |
| 3 | SERVICE\_NAME | VARCHAR2 |  |  | Mô tả service |
| 4 | SERVICE\_URL | VARCHAR2 |  |  | Chuỗi kết nối  Webservice: sẽ là chuỗi URL  TCP: ip máy chủ,port… |
| 5 | SERVICE\_TIMEOUT | NUMBER |  |  | Thời gian timeout cho phép khi gọi service |

### Bảng ESB\_TLLOG

* Mục đích: Lưu các message xml vào/ra ở ESB.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | TX\_NUM | VARCHAR(20) | PK |  | Số bút toán giao dịch |
| 2 | TX\_DATE | VARCHAR(8) |  |  | Ngày bút toán |
| 3 | MESSAGE\_SN | VARCHAR(20) |  |  | Số thứ tự message,sinh ra từ chương trình nguồn  (Sẽ được dùng để đối chiếu mức message XML giữa chương trình nguồn và hệ thống ESB) |
| 4 | TRAN\_CODE | VARCHAR(10) | FK đến bảng ESB\_TRANS |  | Mã giao dịch |
| 5 | SENDER\_ID | VARCHAR(5) |  |  | Mã ứng dụng gửi |
| 6 | SENDER\_TRAN\_SN | VARCHAR(20) |  |  | Số giao dịch |
| 7 | SEND\_DATE | VARCHAR(8) |  |  | Ngày gửi |
| 8 | SEND\_TIME | VARCHAR(8) |  |  | Giờ gửi |
| 9 | RECEIVER\_ID | VARCHAR(5) |  |  | Mã ứng dụng nhận |
| 10 | RECEIVER\_TRAN\_SN | VARCHAR(20) |  |  | Số giao dịch |
| 11 | RECEIVE\_DATE | VARCHAR(8) |  |  | Ngày nhận giao dịch |
| 12 | RECEIVE\_TIME | VARCHAR(8) |  |  | Giờ nhận giao dịch |
| 13 | RESP\_DATE | VARCHAR(8) |  |  | Ngày phản hồi giao dịch |
| 14 | RESP\_TIME | VARCHAR(8) |  |  | Giờ phản hồi |
| 15 | RESP\_CODE | VARCHAR(3) |  |  | Mã lỗi |
| 16 | RESP\_MSG | VARCHAR(200) |  |  | Mô tả lỗi |
| 17 | IN\_XML\_MSG | CLOB(4000) |  |  | Message XML gửi đi |
| 18 | OUT\_XML\_MSG | CLOB(4000) |  |  | Message XML nhận về |

### Bảng ESB\_MSGLOG

* Mục đích: Lưu các chi tiết các message request/response khi gọi các dịch vụ.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | TX\_NUM | VARCHAR(20) | FK |  | Số bút toán giao dịch |
| 2 | TX\_DATE | VARCHAR(8) |  |  | Ngày bút toán |
| 3 | TRAN\_SN | VARCHAR(20) |  |  | Số giao dịch sinh bởi chương trình nguồn  Trong trường hợp message input XML chỉ gồm 1 giao dịch thì đây chính là SENDER\_TRAN\_SN  Trong trường hợp message input XML gồm nhiều giao dịch trong 1 message thì với mỗi giao dịch này chương trình nguồn phải sinh ra một số tran\_sn |
| 4 | SND\_TRAN\_SN | VARCHAR(20) |  |  | Số giao dịch (sinh bởi ESB)  Các hệ thống bị gọi sẽ lưu lại thông tin này dùng làm đối chiếu ngược lại với ESB  (Vào core thì đây là số sequence,số ref …) |
| 5 | SND\_TRAN\_DATE | VARCHAR(8) |  |  | Ngày gửi (sinh bởi ESB) |
| 6 | SND\_TIME | VARCHAR(8) |  |  | Giờ gửi (sinh bởi ESB) |
| 7 | RCV\_TRAN\_DATE | VARCHAR(8) |  |  | Ngày nhân message được phản hồi |
| 8 | RCV\_TIME | VARCHAR(8) |  |  | Giờ nhận được message phản hồi  (So sánh giữa giờ gửi/nhận để biết được thời gian xử lý message) |
| 9 | RESP\_CODE | VARCHAR(3) |  |  | Mã lỗi trả về |
| 10 | RESP\_MSG | VARCHAR(200) |  |  | Mô tả lỗi trả về |
| 11 | IN\_MESSAGE | CLOB(4000) |  |  | Nội dung nessage gửi đi |
| 12 | OUT\_MESSAGE | CLOB(4000) |  |  | Nội dung message nhận về |
| 13 | TELLER\_ID | VARCHAR(10) |  |  | Mã giao dịch viên thực hiện giao dịch  (Trong trường hợp gửi vào core) |
| 14 | APPROVER\_ID | VARCHAR(10) |  |  | Mã kiểm soát viên thực hiện giao dịch  (Trong trường hợp gửi vào core) |
| 15 | REF\_APP\_ID | VARCHAR(10) |  |  | Tham chiếu đến ứng dụng nào? |
| 16 | REF\_SERVICE | VARCHAR(10) |  |  | Tham chiếu đến dịch vụ nào |
| 17 | REF\_PORT | VARCHAR(10) |  |  | Sử dụng cổng nào (giao dịch với core) |
| 18 | REF\_CHANNEL | VARCHAR(10) |  |  | Sử dụng kênh nào (giao dịch với core) |
| 19 | REF\_TRAN\_CODE | VARCHAR(10) |  |  | Mã trancode thực tế đi vào hệ thống đích là gì  Ví dụ: chuyển khoản vào core là EB8277… |
| 20 | REF\_CIF\_ACCT | VARCHAR(20) |  |  | Số cif/account (dùng để tra cứu nhanh giao dịch) |
| 21 | REF\_AMT | VARCHAR(20) |  |  | Số tiền (dùng để tra cứu nhanh giao dịch) |

### Bảng ESB\_ERRLOG

* Mục đích: Ghi log lỗi,exception…phát sinh bởi hệ thống

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | AUTOID | NUMBER | PK |  | Số tự tăng. |
| 2 | LOG\_DATE | DATE |  |  | Ngày,giờ phát sinh lỗi. |
| 3 | ERR\_SOURCE | NVARCHAR2(200) |  |  | Nguồn phát sinh lỗi (hàm,lớp…) |
| 4 | ERR\_MSG | NVARCHAR2(300) |  |  | Mô tả lỗi |

### Bảng ESB\_CHANNEL

* Mục đích: Thông tin về port đi vào core

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | CHANNEL\_ID | VARCHAR2(10) | PK |  | Tên kênh |
| 2 | CHANNEL\_NAME | VARCHAR2(100) |  |  | Mô tả |
| 3 | REF\_PORT | NUMBER |  |  | Cổng thực tế đi vào core. |
| 4 | STATUS | VARCHAR2(1) |  |  | Trạng thái sử dụng |

### Bảng ESB\_SEQMGR

* Mục đích: Quản lý thông tin sequence dùng để sinh thẻ tích kiệm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Tên trường | Loại dữ liệu | Ràng buộc | Null | Mô tả |
| 1 | AUTOID | NUMBER | PK |  | SỐ tự tăng |
| 2 | VERSION | VARCHAR2(5) |  |  | Phiên bản |
| 3 | PARAM1 | VARCHAR2(20) |  |  | Tham số 1 |
| 4 | PARAM2 | VARCHAR2(20) |  |  | Tham số 2 |
| 5 | SEQ\_LIMIT | NUMBER |  |  | Maximum số seq cho phép |
| 6 | SEQ\_NUMBER | NUMBER |  |  | Số seq hiện tại |
| 7 | BRANCH\_CODE | VARCHAR2(3) |  |  | Mã chi nhánh |
| 8 | GROUP\_CODE | VARCHAR2(300) |  |  | Mã nhóm thẻ |
| 9 | CCY\_CODE | VARCHAR2(3) |  |  | Mã tiền tệ |

### Bảng ESB\_CHUCNANG

| **Tên truờng** | **Kiểu dữ liệu** | **Rỗng?** | **P/F Key?** | **Mặc định** | **Diễn giải** |
| --- | --- | --- | --- | --- | --- |
| ID | NUMBER (22) |  | P |  | PK |
| CNANG\_CHA | NUMBER (22) | X |  |  | FK tham chieu ESB\_CHUCNANG.ID |
| TEN\_CNANG | VARCHAR2 (50) | X |  |  | Ten chuc nang |
| MO\_TA | VARCHAR2 (150) | X |  |  | Mo ta chuc nang |
| TEN\_ACTION | VARCHAR2 (100) | X |  |  | Ten action tuong ung voi chuc nang |
| HIEN\_THI | VARCHAR2 (1) | X |  |  | Co hien thi tren menu hay khong. Y - Co, N - Khong |
| SAP\_XEP | NUMBER (22) | X |  |  | Thu tu sap xep |
| TRANG\_THAI | VARCHAR2 (1) | X |  |  | Trang thai chuc nang. Y - Dang su dung, N - Khong su dung |
| MA\_CNANG | VARCHAR2 (5) |  | U |  | Ma chuc nang |
| LOAI\_CNANG | VARCHAR2 (1) | X |  |  | T: them moi; S: sua; X: xoa; D: duyet |
| KY\_HIEU\_CNANG | VARCHAR2 (50) |  | U |  | Ky hieu chuc nang |

### Bảng ESB\_CNANG\_NHOM

| **Tên truờng** | **Kiểu dữ liệu** | **Rỗng?** | **P/F Key?** | **Mặc định** | **Diễn giải** |
| --- | --- | --- | --- | --- | --- |
| ID | NUMBER (22) |  | P |  | PK |
| CNANG\_ID | NUMBER (22) | X |  |  | FK tham chieu toi ESB\_CHUCNANG.ID |
| NHOM\_ID | NUMBER (22) | X |  |  | FK tham chieu toi ESB\_NHOM\_NSD.ID |

### Bảng ESB\_NHOM\_NSD

| **Tên truờng** | **Kiểu dữ liệu** | **Rỗng?** | **P/F Key?** | **Mặc định** | **Diễn giải** |
| --- | --- | --- | --- | --- | --- |
| ID | NUMBER (22) |  | P |  | PK |
| TEN\_NHOM | VARCHAR2 (50) | X | U |  | Ten nhom NSD |
| LOAI\_NHOM | VARCHAR2 (10) | X |  |  | Phan loai nhom NSD |

### Bảng ESB\_NSD

| **Tên truờng** | **Kiểu dữ liệu** | **Rỗng?** | **P/F Key?** | **Mặc định** | **Diễn giải** |
| --- | --- | --- | --- | --- | --- |
| ID | NUMBER (22) |  | P |  | PK |
| TEN\_NSD | VARCHAR2 (100) | X |  |  | Ten NSD |
| CHUC\_DANH | VARCHAR2 (100) | X |  |  | Chuc danh cua NSD |
| MA\_NSD | VARCHAR2 (50) | X |  |  | Ma NSD, su dung de Logon |
| TRANG\_THAI | VARCHAR2 (2) | X |  |  | Trang thai cua NSD, 01-Active, 02-Khoa, 03-ngung hoat dong |
| NGAY\_TAO | DATE (7) | X |  |  | Ngay tao |
| NGUOI\_TAO | NUMBER (22) | X |  |  | Khoa ngoai cua esb.esb\_nsd |
| IP\_TRUYCAP | VARCHAR2 (20) | X |  |  | IP dang truy cap |
| SESSION\_ID | VARCHAR2 (100) | X |  |  | Session ID |
| TGIAN\_TRUYCAP | DATE (7) | X |  |  | Thoi diem truy cap gan nhat |
| EMAIL | VARCHAR2 (80) | X |  |  | Email nguoi dung |
| PASSWORD | VARCHAR2 (80) | X |  |  | Mat khau nguoi dung |
| SO\_DTHOAI | VARCHAR2 (44) | X |  |  | So dien thoai nguoi dung |

### Bảng ESB\_NSD\_NHOM

| **Tên truờng** | **Kiểu dữ liệu** | **Rỗng?** | **P/F Key?** | **Mặc định** | **Diễn giải** |
| --- | --- | --- | --- | --- | --- |
| ID | NUMBER (22) |  | P |  | PK |
| NSD\_ID | NUMBER (22) | X |  |  | FK tham chieu toi ESB\_NSD.ID |
| NHOM\_ID | NUMBER (22) | X |  |  | FK tham chieu toi ESB\_NHOM\_NSD.ID |
| CREATED\_DATE | DATE (7) | X |  |  | Ngay phan quyen |
| CREATED\_BY | NUMBER (22) | X |  |  | FK tham chieu toi ESB\_NSD.ID |

## Ứng dụng theo dõi, tra cứu, đối chiếu

### Quản trị người dùng

Bao gồm các chức năng:

* Login
* Logout
* Đổi mật khẩu
* Reset mật khẩu
* Tạo người dùng
* Phân quyền cho người dùng

### Theo dõi, tra cứu

Các điều kiện tra cứu:

* Theo ứng dụng sử dụng
* Theo loại giao dịch: chuyển khoản, tạo sổ tiết kiệm, tất toán …
* Theo trạng thái
* …

### Đối chiếu với giao dịch trong core

Thông tin logging trong DB ghi log sẽ được đối chiếu với thông tin giao dịch lấy từ Core Bank. Thông tin giao dịch sẽ được lấy từ Core Bank sang ESB DB một cách tự động, định kì; sử dụng Oracle Gateway tương tự như eBank đang dùng.



# Thiết kế chi tiết các luồng trao đổi, tích hợp

## Khuôn dạng MQ message chung

### MQ in message

<?xml version="1.0" encoding="utf-8" ?>

<Data>

<Header>

<req\_id>req\_id</req\_id>

<req\_app>app\_id</req\_app>

<srv>srv\_id</srv>

<srv\_app>srv\_app\_id</srv\_app>

<req\_time>yyyy-mm-dd hh:mi:ssssss</req\_time>

</Header>

<Body>

<!--Thay đổi theo giá trị của thẻ srv trên header, tuân thủ theo cấu trúc của đầu vào của method -->

</Body>

</Data>

**Ví dụ 1:**

<?xml version="1.0" encoding="utf-8" ?>

<Data>

<Header>

<req\_id>1234</req\_id>

<req\_app>3 starfish</req\_app>

<srv>2 registerService của eBank</srv>

<req\_time>yyyymmdd hhmiss</req\_time>

</Header>

<Body>

<registerService>

<!-- lấy theo tên của service/method -->

<!-- các cấu trúc element sẽ lấy theo cấu trúc của đầu vào của method -->

<userInfo>

<cifNo>cif</cifNo>

<email>email</email>

<gender>Mr.</gender>

<groupId>VIP</groupId>

<mobile>84904132550</mobile>

<openMbs>y</openMbs>

<securityType>b</securityType>

<userName>phongnx</userName>

</userInfo>

<tokenNo></tokenNo>

<acctNoList>

<RelatedAccountWS>

<acctNo>acctNo</acctNo>

<alias>myacc</alias>

</RelatedAccountWS>

<RelatedAccountWS>

<acctNo>acctNo</acctNo>

<alias>myacc</alias>

</RelatedAccountWS>

</acctNoList>

</registerService>

</Body>

</Data>

**Ví dụ 2:**

<?xml version="1.0" encoding="utf-8" ?>

<Data>

<Header>

<req\_id>1234</req\_id>

<req\_app>3 starfish</req\_app>

<srv>1 regSMSBanking của SMS Banking</srv>

<req\_time>yyyymmdd hhmiss</req\_time>

</Header>

<Body>

<regSMSBanking>

<cifNumber> cifNumber </cifNumber>

<accountNumber> accountNumber </accountNumber>

<serviceType>1/2</serviceType>

<alert>1/0</alert>

<phoneNumber> phoneNumber </phoneNumber>

<branchCode></branchCode>

<appReg> appReg </appReg>

<regSMSBanking>

</Body>

</Data>

**Ví dụ 3:**

<?xml version="1.0" encoding="utf-8" ?>

<Data>

<Header>

<req\_id>1234</req\_id>

<req\_app>3 starfish</req\_app>

<srv>3 createFD của Core Bank</srv>

<req\_time>yyyymmdd hhmiss</req\_time>

</Header>

<Body>

<createFD>

<!-- lấy theo tên của service/method -->

<!-- các cấu trúc element sẽ lấy theo cấu trúc của đầu vào của method -->

<channel> channel </channel>

<hostDate> hostDate </hostDate>

<hostId> hostId </hostId>

<branchCode> branchCode </branchCode>

<currencyType> currencyType </currencyType>

<cifNumber> cifNumber </cifNumber>

<accountNumber> accountNumber </accountNumber>

<accountType> accountType </accountType>

<accountName> accountName </accountName>

<modeOfOperation> modeOfOperation </modeOfOperation>

</createFD>

</Body>

</Data>

### MQ out message

<?xml version="1.0" encoding="utf-8" ?>

<Data>

<Header>

<res\_id>res\_id</res\_id>

<req\_id>req\_id</req\_id>

<req\_app>app\_id</req\_app>

<srv>srv\_id</srv>

<res\_time> yyyy-mm-dd hh:mi:ssssss </res\_time>

</Header>

<Body>

<!--Thay đổi theo giá trị của thẻ srv trên header, tuân thủ theo cấu trúc của đầu ra của method -->

</Body>

</Data>

**Ví dụ 1:**

<?xml version="1.0" encoding="utf-8" ?>

<Data>

<Header>

<res\_id>res\_id</res\_id>

<req\_id>1234</req\_id>

<req\_app>3 starfish</req\_app>

<srv>2 registerService của eBank</srv>

<req\_time>yyyymmdd hhmiss</req\_time>

</Header>

<Body>

<return>0-- thành công</return>

</Body>

</Data>

**Ví dụ 2:**

<?xml version="1.0" encoding="utf-8" ?>

<Data>

<Header>

<res\_id>res\_id</res\_id>

<req\_id>1234</req\_id>

<req\_app>3 starfish</req\_app>

<srv>1 regSMSBanking của SMS Banking</srv>

<req\_time>yyyymmdd hhmiss</req\_time>

</Header>

<Body>

<out>1/-1/-2/-3/-4/-5</out>

</Body>

</Data>

## Thiết kế MainFlow

* Các ứng dụng sử dụng dịch vụ put request msg và Q-in (chung cho tất cả các ứng dụng sử dụng dịch vụ)
* MainRequestFlow sẽ được kích hoạt: flow này bao gồm 1 số validation đơn giản dựa trên header của msg và bước routing: định tuyến msg tới các flow tương ứng với từng ứng dụng cung cấp dịch vụ. MainRequestFlow sẽ gọi tới các LoggingFlow, ExceptionFlow để ghi log và xử lý lỗi nếu cần
* Flow của ứng dụng cung cấp dịch vụ được kích hoạt. Flow của ứng dụng cung cấp dịch vụ cũng sẽ gọi tới tới các LoggingFlow, ExceptionFlow để ghi log và xử lý lỗi nếu cần
* Response sẽ được trả về Response Routing node để định tuyến trả kết quả về cho Q-out riêng của mỗi ứng dụng sử dụng dịch vụ





## Thiết kế flow ghi log và flow xử lý exception

### Flow ghi log

### Flow xử lý exception

## Thiết kế chi tiết flow cho các ứng dụng cung cấp dịch vụ

### Flow cho Core bank (cung cấp API dạng TCP/IP socket)



* Request msg được put vào queue Core-in. Req msg sẽ được kiểm tra định dạng XML dựa trên XSD thông qua MessagSet.
* CoreFlow được kích hoạt
* Java node lấy req msg, phân tích req message đầu vào để xác định phương thức cần gọi, chuyển đổi định dạng message, mã hóa theo yêu cầu của Core bank. Gọi tới các LoggingFlow, ExceptionFlow để ghi log và xử lý lỗi.
* Thực hiện gửi yêu cầu tới core bank thông qua TCPIP Output node.
* Nhận kết quả trả về từ core bank thông qua TCPIP Receive Node.
* Java node xử lý message kết quả trả về, chuyển đổi định dạng message.
* Put response message vào queue Core-out.

### Flow cho RSA AM (cung cấp API dạng Java Call)



* Request msg được put vào queue RSA-in. Req msg sẽ được kiểm tra định dạng XML dựa trên XSD thông qua MessagSet.
* RSAFlow được kích hoạt
* Java node lấy req msg, lấy dữ liệu trong req msg, gọi tới các API của RSA AM sử dụng các thư viện của RSA agent để sử dụng các hàm, thủ tục thao tác với RSA AM . Gọi tới các LoggingFlow, ExceptionFlow để ghi log và xử lý lỗi.
* Nhận kết quả trả về từ RSA AM.
* Xử lý message kết quả trả về, chuyển đổi định dạng message phù hợp với ứng dụng sử dụng dịch vụ.
* Put response message vào queue RSA-out.

### Flow cho các ứng dụng SMS banking, SMS OTP, Fund transfer code, Fee, eBank (cung cấp API dạng web service)

Mỗi ứng ụng cung cấp dịch vụ này sẽ có 1 flow riêng, tên là SMSBFlow, SMSOTPFlow, FTFlow, FeeFlow, eBankFlow. Các tên này sẽ sử dụng tên chung WsFlow trong các hình vẽ và mô tả dưới đây.



* Request msg được put vào queue Ws-in. Req msg sẽ được kiểm tra định dạng XML dựa trên XSD thông qua MessagSet.
* WsFlow tương ứng được kích hoạt
* Transform req msg từ dạng quy định của ESB thành dạng quy định khi gọi web service. Gọi tới các LoggingFlow, ExceptionFlow để ghi log và xử lý lỗi.
* Thực hiện gửi yêu cầu tới ứng dụng cung cấp dịch vụ thông qua HTTP Request node.
* Nhận kết quả trả về từ ứng dụng cung cấp dịch vụ.
* Xử lý message kết quả trả về, chuyển đổi định dạng message chung theo quy định của ESB
* Put response msg vào queue Ws-out.

### Flow cho Tsunami (cung cấp API dạng HTTP request)



* Request msg được put vào queue Tsunami-in. Req msg sẽ được kiểm tra định dạng XML dựa trên XSD thông qua MessagSet.
* TsunamiFlow tương ứng được kích hoạt
* Transform req msg từ dạng quy định của ESB thành dạng quy định khi gọi HTTP RESTFull API của Tsunami. Gọi tới các LoggingFlow, ExceptionFlow để ghi log và xử lý lỗi.
* Thực hiện gửi yêu cầu tới Tsunami thông qua HTTP Request node.
* Nhận kết quả trả về từ Tsunami.
* Xử lý message kết quả trả về, chuyển đổi định dạng message chung theo quy định của ESB
* Put response msg vào queue Tsunami -out.

## Thiết kế chi tiết flow cho các ứng dụng sử dụng dịch vụ

### Starfish



Chỉ sử dụng dịch vụ của Core Bank, eBank và SMS banking.

Chi tiết các service mà Starfish sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| **Ứng dụng sử dụng dịch vụ** | **Method** | **Ứng dụng cung cấp dịch vụ** |
| Ebanking | checkOtp | Otp |
| getOtp |
| sendSMS | InternalGW |
| sendSMSObj |
| sendSMSArr |
| transationInquiry | InReceiverMoney |
| pickupValidate |
| pickupReciveMoney |
| calFee | Fee |
| calFeeObj |
| queryBilling | InternalEGW |
| payBilling |
| payBillingField |
| createFD | Core |
| HostMessageSending |
| createFDReceipt |
| transferFromCASAToCASA |
| getAccount |
| cardActivation |
| getFDReceiptAccount |
| checkOtp | Transfercode |
| getOtp |

### MPayroll

Chỉ sử dụng dịch vụ của Core và SMS banking.

Chi tiết các service mà MPayroll sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| Mpayroll | regSMSBanking | SMSBanking |
| linkCard | Core |
| createCA |
| getAccount |
| createCif |

### MTrading

Chỉ sử dụng dịch vụ của Core.

Chi tiết các service mà MTrading sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| Mtrading | debitAdvice | Core |
| accountInquiry |
| creditAdvice |
| transferGLtoGL |

### SMS banking

Chỉ sử dụng dịch vụ của Fee, core, transfer code, SMS gateway

Chi tiết các service mà SMS Banking sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| SMS banking | sendSMS | InternalGW |
| sendSMSObj |
| sendSMSArr |
| calFee | Fee |
| calFeeObj |
| transferFromCASAToCASA | Core |
| getInformationByIDForIBSRegistration |
| searchCustomerById |
| ddMasterInquiry |
| searchAccountMoreByCif |
| loanAccountInquiry |
| getInformationByCIFForIBSRegistration |
| miniStatement |
| searchCustomerByCif |
| searchAccountByCif |
| getPassCode | Transfercode |
| createPassCode |
| createPassCodeLength |
| resetPassCode |
| updatePassCode |
| checkPassCode |

### Tsunami

Sử dụng dịch vụ của SMS gateway, SMS banking, eBank, core

Chi tiết các service mà Tsunami sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| Tsunami | regSMSBanking | SMSBanking |
| sendSMS | InternalGW |
| registerService | EBanking |
| lockUpdateMainternanceCA | Core |
| linkMasterCard |
| lockUpdateE\_Contact |
| lockTransferAccountFromCifToAnother |
| searchCustomerById |
| linkCard |
| ddMasterInquiry |
| updateE\_Contact |
| updateMainternanceCA |
| searchCustomerByCif |
| addE\_Contact |
| createCA |
| getAccount |
| updateMainternanceNewCA |
| transferAccountFromCifToAnother |
| createCif |
| lockUpdateMainternanceNewCA |
| searchE\_ContactByCif |

### eBank



Sử dụng dịch vụ của SMS OTP, SMS banking, Fee, Core, Transfer code, RSA, SMS , gatewayService payment gateway

Chi tiết các service mà eBank sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| Ebanking | checkOtp | Otp |
| getOtp |
| sendSMS | InternalGW |
| sendSMSObj |
| sendSMSArr |
| transationInquiry | InReceiverMoney |
| pickupValidate |
| pickupReciveMoney |
| calFee | Fee |
| calFeeObj |
| queryBilling | InternalEGW |
| payBilling |
| payBillingField |
| createFD | Core |
| HostMessageSending |
| createFDReceipt |
| transferFromCASAToCASA |
| getAccount |
| cardActivation |
| getFDReceiptAccount |
| checkOtp | Transfercode |
| getOtp |
| Tạo user | RSA |
| Assign token cho user |
| Kiểm tra token OTP |
| Unassign token |
| Xóa user |

### eTax



Chỉ sử dụng dịch vụ của Core, Tax Gateway.

Chi tiết các service mà SMS Gateway sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| eTax | Chuyển khoản liên ngân hàng (từ CA/SA của khách hàng sang TK của Kho bạc mở tại Ngân hàng khác) | Core |
|  | Truy vấn thông tin khách hàng (theo số tài khoản) |  |
|  | GetPublicKey | Tax Gateway |
|  | WSProcess |  |
|  | ReconcileProcess |  |
|  | TraCuuThongTinDKT |  |
|  | TraCuuThongTinThuNop |  |
|  | TraCuuThongTinCT |  |
|  | GuiThongTinCT  GuiThongTinChungTu |  |
|  | CapNhatThongTinChungTu |  |
|  | HuyChungTuThue  HuyThongTinCT |  |

### SMS Gateway

Chỉ sử dụng dịch vụ của Tsunami và SMS banking.

Chi tiết các service mà SMS Gateway sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| SMS GW | putSMS | SMSBanking |
| SendMessage | Tsunami |

### Service Payment Gateway

Chỉ sử dụng dịch vụ của Core.

Chi tiết các service mà SMS Gateway sử dụng như sau:

|  |  |  |
| --- | --- | --- |
| Ứng dụng sử dụng dịch vụ | Method | Ứng dụng cung cấp dịch vụ |
| Services GW | transferFromCASAToCASA | Core |
| reversableTransaction |
| transferFromCASAToGL |
| transferFromWUToCA |
| reversableOL2Transaction |

## Thiết kế chi tiết Tax Gateway

Hiện tại ứng dụng eTax- Thu ngân sách đang kết nối trực tiếp sang Tổng cục Thuế và Tổng cục Hải quan. Để tuân thủ các nguyên tắc bảo mật và thiết kế của ESB, ta phải tạo ra 1 gateway để kết nối sang Tổng cục Thuế và Tổng cục Hải quan, gateway này được triển khai trên 1 máy chủ đặt trong vùng DMZ là vùng cho phép kết nối sang các hệ thống bên ngoài ngân hàng. Gateway này được đặt tên là Tax Gateway.

Tax Gateway sẽ kết nối sang các hệ thống của Tổng cục Thuế và Tổng cục Hải quan và cung cấp dịch vụ dạng web service để ứng dụng eTax sử dụng.

Tổng cục Hải quan cung cấp 2 web service:

* TTDTPortal: dùng cho trao đổi thông tin
* DCTTDTPortal: dùng cho đối chiếu các thông tin đã trao đổi.

Tổng cục Thuế cung cấp web service:

* GIPBank: trao đổi thông tin đăng ký thuế, thông tin thu nộp, chứng từ thuế với các ngân hàng.

TaxGateway sẽ là web service client của các web service trên là là web service để cho eTax gọi.



|  |  |  |
| --- | --- | --- |
| **Web service** | **Method** | **Mô tả** |
| TTDTPortal | GetPublicKey | Lấy thông tin Public Key của chứng thư số của TC hải quan. Phục vụ cho việc mã hóa bản tin truyền từ ngân hàng sang Hải quan |
|  | WSProcess | Thực hiện lấy thông tin thuế xuất nhập khẩu từ Hải quan |
| DCTTDTPortal | ReconcileProcess | Thực hiện đối chiếu giữa ngân hàng và hải quan |
| GIPBank | TraCuuThongTinDKT |  |
|  | TraCuuThongTinThuNop | Dịch vụ tra cứu thông tin thu nộp theo mã số thuế  Dịch vụ tra cứu thông tin đăng ký thuế theo số chứng minh thư |
|  | TraCuuThongTinCT | Dịch vụ tra cứu thông tin chứng từ thuế |
|  | GuiThongTinCT  GuiThongTinChungTu | Dịch vụ gửi chứng từ thuế |
|  | CapNhatThongTinChungTu | Dịch vụ cập nhật thông tin chứng từ thuế |
|  | HuyChungTuThue  HuyThongTinCT | Dịch vụ hủy một chứng từ thuế đã gửi |

# Phụ lục: các XSD cho từng flow của ứng dụng cung cấp dịch vụ

## Core bank (core.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createHoldMessage\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="account" type="xs:string"/>

<xs:element name="expiryDate" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

<xs:element name="amount" type="xs:string"/>

<xs:element name="code" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createFD\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="hostDate" type="xs:string"/>

<xs:element name="hostId" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="currencyType" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="accountType" type="xs:string"/>

<xs:element name="accountName" type="xs:string"/>

<xs:element name="modeOfOperation" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="lockUpdateMainternanceCA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="debitAdvice\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="effectiveDate" type="xs:string"/>

<xs:element name="crAmount" type="xs:string"/>

<xs:element name="glAccount" type="xs:string"/>

<xs:element name="drAmount" type="xs:string"/>

<xs:element name="buyRate" type="xs:string"/>

<xs:element name="sellRate" type="xs:string"/>

<xs:element name="debitCurrency" type="xs:string"/>

<xs:element name="glCurrency" type="xs:string"/>

<xs:element name="remark" type="xs:string"/>

<xs:element name="transcode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="linkMasterCard\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="accountTag" type="xs:string"/>

<xs:element name="accountType" type="xs:string"/>

<xs:element name="usage" type="xs:string"/>

<xs:element name="currency" type="xs:string"/>

<xs:element name="branchTag" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="idNumber" type="xs:string"/>

<xs:element name="idType" type="xs:string"/>

<xs:element name="name" type="xs:string"/>

<xs:element name="address" type="xs:string"/>

<xs:element name="product" type="xs:string"/>

<xs:element name="annualFee" type="xs:string"/>

<xs:element name="vip" type="xs:string"/>

<xs:element name="addressLine2" type="xs:string"/>

<xs:element name="addressLine3" type="xs:string"/>

<xs:element name="addressLine4" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="HostMessageSending\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="message" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="lockUpdateE\_Contact\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="sequenceNo" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="changeCardService\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="cardNumber" type="xs:string"/>

<xs:element name="serviceName" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createFDReceipt\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="journalSeq" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="fdGroupAccount" type="xs:string"/>

<xs:element name="depositAmt" type="xs:string"/>

<xs:element name="effectiveDate" type="xs:string"/>

<xs:element name="fdReceipt" type="xs:string"/>

<xs:element name="rate" type="xs:string"/>

<xs:element name="intPaymentToAcctno" type="xs:string"/>

<xs:element name="printTranferToAcctno" type="xs:string"/>

<xs:element name="currency" type="xs:string"/>

<xs:element name="productCode" type="xs:string"/>

<xs:element name="autoRenew" type="xs:string"/>

<xs:element name="intPaymentMode" type="xs:string"/>

<xs:element name="fdType" type="xs:string"/>

<xs:element name="remark" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="transferFromCASAToCASA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="creditAccount" type="xs:string"/>

<xs:element name="creditAmount" type="xs:string"/>

<xs:element name="creditCurrency" type="xs:string"/>

<xs:element name="creditRate" type="xs:string"/>

<xs:element name="debitAccount" type="xs:string"/>

<xs:element name="debitAmount" type="xs:string"/>

<xs:element name="debitCurrency" type="xs:string"/>

<xs:element name="debitRate" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="transCode" type="xs:string"/>

<xs:element name="vatFee" type="xs:string"/>

<xs:element name="serviceFee" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="lockTransferAccountFromCifToAnother\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="sourceCif" type="xs:string"/>

<xs:element name="accountToMove" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createSA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="hostDate" type="xs:string"/>

<xs:element name="hostId" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="currencyType" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="accountType" type="xs:string"/>

<xs:element name="accountName" type="xs:string"/>

<xs:element name="modeOfOperation" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="TFMessageSending\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="message" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="joinSA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="strteller" type="xs:string"/>

<xs:element name="strBankcode" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="strAccount" type="xs:string"/>

<xs:element name="strCifnumber" type="xs:string"/>

<xs:element name="strNameAcount" type="xs:string"/>

<xs:element name="strcurrCode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="joinCA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="strteller" type="xs:string"/>

<xs:element name="strBankcode" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="strAccount" type="xs:string"/>

<xs:element name="strCifnumber" type="xs:string"/>

<xs:element name="strNameAcount" type="xs:string"/>

<xs:element name="strcurrCode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="accountInquiry\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="account" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="markHotCard\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="cardNumber" type="xs:string"/>

<xs:element name="typeLock" type="xs:string"/>

<xs:element name="comment" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="getInformationByIDForIBSRegistration\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cert\_code" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="searchCustomerById\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="idNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="creditAdvice\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="effectiveDate" type="xs:string"/>

<xs:element name="crAmount" type="xs:string"/>

<xs:element name="glAccount" type="xs:string"/>

<xs:element name="drAmount" type="xs:string"/>

<xs:element name="buyRate" type="xs:string"/>

<xs:element name="sellRate" type="xs:string"/>

<xs:element name="debitCurrency" type="xs:string"/>

<xs:element name="glCurrency" type="xs:string"/>

<xs:element name="remark" type="xs:string"/>

<xs:element name="transcode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="linkCard\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cardNumber" type="xs:string"/>

<xs:element name="accountTag" type="xs:string"/>

<xs:element name="accountType" type="xs:string"/>

<xs:element name="usage" type="xs:string"/>

<xs:element name="currency" type="xs:string"/>

<xs:element name="branchTag" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="idNumber" type="xs:string"/>

<xs:element name="idType" type="xs:string"/>

<xs:element name="name" type="xs:string"/>

<xs:element name="address" type="xs:string"/>

<xs:element name="product" type="xs:string"/>

<xs:element name="annualFee" type="xs:string"/>

<xs:element name="vip" type="xs:string"/>

<xs:element name="addressLine2" type="xs:string"/>

<xs:element name="addressLine3" type="xs:string"/>

<xs:element name="addressLine4" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="reActiveCard\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="cardNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="reversableTransaction\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="strFromAccount" type="xs:string"/>

<xs:element name="strToAccount" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="strDescription" type="xs:string"/>

<xs:element name="strTranferAmount" type="xs:string"/>

<xs:element name="strJournalSeq" type="xs:string"/>

<xs:element name="orgJournalSeq" type="xs:string"/>

<xs:element name="strTransCode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="ddMasterInquiry\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="accountType" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="searchAccountMoreByCif\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="lastAccountType" type="xs:string"/>

<xs:element name="lastAccountNumber" type="xs:string"/>

<xs:element name="relationShip" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="loanAccountInquiry\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="transferGLtoGL\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="refID" type="xs:string"/>

<xs:element name="debitGLBranch" type="xs:string"/>

<xs:element name="debitGLAccount" type="xs:string"/>

<xs:element name="creditGLBranch" type="xs:string"/>

<xs:element name="creditGLAccount" type="xs:string"/>

<xs:element name="amount" type="xs:string"/>

<xs:element name="currency" type="xs:string"/>

<xs:element name="comments" type="xs:string"/>

<xs:element name="referenceText" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="hostDate" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="transferFromCASAToGL\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="hostIP" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="fromAccount" type="xs:string"/>

<xs:element name="amount" type="xs:string"/>

<xs:element name="fromCif" type="xs:string"/>

<xs:element name="GLAccount" type="xs:string"/>

<xs:element name="vatFee" type="xs:string"/>

<xs:element name="serviceFee" type="xs:string"/>

<xs:element name="remarks" type="xs:string"/>

<xs:element name="fromName" type="xs:string"/>

<xs:element name="fromId" type="xs:string"/>

<xs:element name="toAccount" type="xs:string"/>

<xs:element name="toName" type="xs:string"/>

<xs:element name="toId" type="xs:string"/>

<xs:element name="toAddress" type="xs:string"/>

<xs:element name="toIdIssueDate" type="xs:string"/>

<xs:element name="toIdIssuePlace" type="xs:string"/>

<xs:element name="transCode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="updateE\_Contact\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="sequenceNo" type="xs:string"/>

<xs:element name="eContactType" type="xs:string"/>

<xs:element name="newEcontact" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="updateMainternanceCA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="introducerCode" type="xs:string"/>

<xs:element name="expense" type="xs:string"/>

<xs:element name="accountName" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="getInformationByCIFForIBSRegistration\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cif\_no" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createUnHoldMessage\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="account" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

<xs:element name="amount" type="xs:string"/>

<xs:element name="code" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

<xs:element name="actionCode" type="xs:string"/>

<xs:element name="expireDate" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="transferFromWUToCA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="hostIP" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="toAccount" type="xs:string"/>

<xs:element name="crAmount" type="xs:string"/>

<xs:element name="ttAmount" type="xs:string"/>

<xs:element name="toCif" type="xs:string"/>

<xs:element name="WUAccount" type="xs:string"/>

<xs:element name="vatFee" type="xs:string"/>

<xs:element name="serviceFee" type="xs:string"/>

<xs:element name="buyRate" type="xs:string"/>

<xs:element name="sendRate" type="xs:string"/>

<xs:element name="ttSendRate" type="xs:string"/>

<xs:element name="remarks" type="xs:string"/>

<xs:element name="toName" type="xs:string"/>

<xs:element name="toId" type="xs:string"/>

<xs:element name="toAddress" type="xs:string"/>

<xs:element name="toIdIssueDate" type="xs:string"/>

<xs:element name="toIdIssuePlace" type="xs:string"/>

<xs:element name="transCode" type="xs:string"/>

<xs:element name="refWU" type="xs:string"/>

<xs:element name="sendingCurrencyType" type="xs:string"/>

<xs:element name="receivingCurrencyType" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="miniStatement\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="recordNumber" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="accountType" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="searchCustomerByCif\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="addE\_Contact\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="eContactType" type="xs:string"/>

<xs:element name="newEcontact" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createCA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="hostDate" type="xs:string"/>

<xs:element name="hostId" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="currencyType" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="accountType" type="xs:string"/>

<xs:element name="accountName" type="xs:string"/>

<xs:element name="modeOfOperation" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="reversableOL2Transaction\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="hostIP" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="manager" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

<xs:element name="oldSequence" type="xs:string"/>

<xs:element name="transDate" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="fromAccount" type="xs:string"/>

<xs:element name="amount" type="xs:string"/>

<xs:element name="amountMustPay" type="xs:string"/>

<xs:element name="fromCif" type="xs:string"/>

<xs:element name="GLAccount" type="xs:string"/>

<xs:element name="vatFee" type="xs:string"/>

<xs:element name="serviceFee" type="xs:string"/>

<xs:element name="remarks" type="xs:string"/>

<xs:element name="fromName" type="xs:string"/>

<xs:element name="fromId" type="xs:string"/>

<xs:element name="toAccount" type="xs:string"/>

<xs:element name="toName" type="xs:string"/>

<xs:element name="toId" type="xs:string"/>

<xs:element name="toAddress" type="xs:string"/>

<xs:element name="toIdIssueDate" type="xs:string"/>

<xs:element name="toIdIssuePlace" type="xs:string"/>

<xs:element name="strTransCode" type="xs:string"/>

<xs:element name="refId" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="getAccount\_type">

<xs:sequence>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="currencyCode" type="xs:string"/>

<xs:element name="groupCode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="updateMainternanceNewCA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="introducerCode" type="xs:string"/>

<xs:element name="expense" type="xs:string"/>

<xs:element name="accountName" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="transferAccountFromCifToAnother\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="sourceCif" type="xs:string"/>

<xs:element name="accountToMove" type="xs:string"/>

<xs:element name="destinationCif" type="xs:string"/>

<xs:element name="destinationName" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="cardActivation\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cardNumber" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="idNo" type="xs:string"/>

<xs:element name="idType" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="searchAccountByCif\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createCif\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="IdNumber" type="xs:string"/>

<xs:element name="IdType" type="xs:string"/>

<xs:element name="IdDateIssued" type="xs:string"/>

<xs:element name="IdCountryIssued" type="xs:string"/>

<xs:element name="IdPlaceIssued" type="xs:string"/>

<xs:element name="strSurname" type="xs:string"/>

<xs:element name="afterSurname" type="xs:string"/>

<xs:element name="addressLine1" type="xs:string"/>

<xs:element name="addressLine2" type="xs:string"/>

<xs:element name="addressLine3" type="xs:string"/>

<xs:element name="addressLine4" type="xs:string"/>

<xs:element name="addressLine5" type="xs:string"/>

<xs:element name="nationality" type="xs:string"/>

<xs:element name="birthday" type="xs:string"/>

<xs:element name="raceCode" type="xs:string"/>

<xs:element name="maritalStatus" type="xs:string"/>

<xs:element name="gender" type="xs:string"/>

<xs:element name="occupationCode\_2" type="xs:string"/>

<xs:element name="placeOfBirth" type="xs:string"/>

<xs:element name="typeElectronicAddress1" type="xs:string"/>

<xs:element name="typeElectronicAddress2" type="xs:string"/>

<xs:element name="typeElectronicAddress3" type="xs:string"/>

<xs:element name="electronicAddress1" type="xs:string"/>

<xs:element name="electronicAddress2" type="xs:string"/>

<xs:element name="electronicAddress3" type="xs:string"/>

<xs:element name="hostDate" type="xs:string"/>

<xs:element name="birthday\_Full" type="xs:string"/>

<xs:element name="idDateIssued\_Full" type="xs:string"/>

<xs:element name="strOccupationCode\_1" type="xs:string"/>

<xs:element name="strProvince" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="lockUpdateMainternanceNewCA\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createLockForUnHoldMessage\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="account" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="searchE\_ContactByCif\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="searchCustomerInformationByID\_type">

<xs:sequence>

<xs:element name="channel" type="xs:string"/>

<xs:element name="teller" type="xs:string"/>

<xs:element name="hostName" type="xs:string"/>

<xs:element name="bankCode" type="xs:string"/>

<xs:element name="cifNumber" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="getFDReceiptAccount\_type">

<xs:sequence>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="currencyCode" type="xs:string"/>

<xs:element name="groupCode" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="createHoldMessage" type="createHoldMessage\_type" minOccurs="0"/>

<xs:element name="createFD" type="createFD\_type" minOccurs="0"/>

<xs:element name="lockUpdateMainternanceCA" type="lockUpdateMainternanceCA\_type" minOccurs="0"/>

<xs:element name="debitAdvice" type="debitAdvice\_type" minOccurs="0"/>

<xs:element name="linkMasterCard" type="linkMasterCard\_type" minOccurs="0"/>

<xs:element name="HostMessageSending" type="HostMessageSending\_type" minOccurs="0"/>

<xs:element name="lockUpdateE\_Contact" type="lockUpdateE\_Contact\_type" minOccurs="0"/>

<xs:element name="changeCardService" type="changeCardService\_type" minOccurs="0"/>

<xs:element name="createFDReceipt" type="createFDReceipt\_type" minOccurs="0"/>

<xs:element name="transferFromCASAToCASA" type="transferFromCASAToCASA\_type" minOccurs="0"/>

<xs:element name="lockTransferAccountFromCifToAnother" type="lockTransferAccountFromCifToAnother\_type" minOccurs="0"/>

<xs:element name="createSA" type="createSA\_type" minOccurs="0"/>

<xs:element name="TFMessageSending" type="TFMessageSending\_type" minOccurs="0"/>

<xs:element name="joinSA" type="joinSA\_type" minOccurs="0"/>

<xs:element name="joinCA" type="joinCA\_type" minOccurs="0"/>

<xs:element name="accountInquiry" type="accountInquiry\_type" minOccurs="0"/>

<xs:element name="markHotCard" type="markHotCard\_type" minOccurs="0"/>

<xs:element name="getInformationByIDForIBSRegistration" type="getInformationByIDForIBSRegistration\_type" minOccurs="0"/>

<xs:element name="searchCustomerById" type="searchCustomerById\_type" minOccurs="0"/>

<xs:element name="creditAdvice" type="creditAdvice\_type" minOccurs="0"/>

<xs:element name="linkCard" type="linkCard\_type" minOccurs="0"/>

<xs:element name="reActiveCard" type="reActiveCard\_type" minOccurs="0"/>

<xs:element name="reversableTransaction" type="reversableTransaction\_type" minOccurs="0"/>

<xs:element name="ddMasterInquiry" type="ddMasterInquiry\_type" minOccurs="0"/>

<xs:element name="searchAccountMoreByCif" type="searchAccountMoreByCif\_type" minOccurs="0"/>

<xs:element name="loanAccountInquiry" type="loanAccountInquiry\_type" minOccurs="0"/>

<xs:element name="transferGLtoGL" type="transferGLtoGL\_type" minOccurs="0"/>

<xs:element name="transferFromCASAToGL" type="transferFromCASAToGL\_type" minOccurs="0"/>

<xs:element name="updateE\_Contact" type="updateE\_Contact\_type" minOccurs="0"/>

<xs:element name="updateMainternanceCA" type="updateMainternanceCA\_type" minOccurs="0"/>

<xs:element name="getInformationByCIFForIBSRegistration" type="getInformationByCIFForIBSRegistration\_type" minOccurs="0"/>

<xs:element name="createUnHoldMessage" type="createUnHoldMessage\_type" minOccurs="0"/>

<xs:element name="transferFromWUToCA" type="transferFromWUToCA\_type" minOccurs="0"/>

<xs:element name="miniStatement" type="miniStatement\_type" minOccurs="0"/>

<xs:element name="searchCustomerByCif" type="searchCustomerByCif\_type" minOccurs="0"/>

<xs:element name="addE\_Contact" type="addE\_Contact\_type" minOccurs="0"/>

<xs:element name="createCA" type="createCA\_type" minOccurs="0"/>

<xs:element name="reversableOL2Transaction" type="reversableOL2Transaction\_type" minOccurs="0"/>

<xs:element name="getAccount" type="getAccount\_type" minOccurs="0"/>

<xs:element name="updateMainternanceNewCA" type="updateMainternanceNewCA\_type" minOccurs="0"/>

<xs:element name="transferAccountFromCifToAnother" type="transferAccountFromCifToAnother\_type" minOccurs="0"/>

<xs:element name="cardActivation" type="cardActivation\_type" minOccurs="0"/>

<xs:element name="searchAccountByCif" type="searchAccountByCif\_type" minOccurs="0"/>

<xs:element name="createCif" type="createCif\_type" minOccurs="0"/>

<xs:element name="lockUpdateMainternanceNewCA" type="lockUpdateMainternanceNewCA\_type" minOccurs="0"/>

<xs:element name="createLockForUnHoldMessage" type="createLockForUnHoldMessage\_type" minOccurs="0"/>

<xs:element name="searchE\_ContactByCif" type="searchE\_ContactByCif\_type" minOccurs="0"/>

<xs:element name="searchCustomerInformationByID" type="searchCustomerInformationByID\_type" minOccurs="0"/>

<xs:element name="getFDReceiptAccount" type="getFDReceiptAccount\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

## eBank (eBank.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="userInfo\_type">

<xs:sequence>

<xs:element name="cifNo" type="xs:string"/>

<xs:element name="email" type="xs:string"/>

<xs:element name="gender" type="xs:string"/>

<xs:element name="groupId" type="xs:int"/>

<xs:element name="mobile" type="xs:string"/>

<xs:element name="openMbs" type="xs:string"/>

<xs:element name="securityType" type="xs:string"/>

<xs:element name="userName" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="RelatedAccountWS\_type">

<xs:sequence>

<xs:element name="acctNo" type="xs:string"/>

<xs:element name="alias" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="acctNoList\_type" >

<xs:sequence>

<xs:element name="RelatedAccountWS" type="RelatedAccountWS\_type" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="registerService\_type">

<xs:sequence>

<xs:element name="userInfo" type="userInfo\_type"/>

<xs:element name="tokenNo" type="xs:string"/>

<xs:element name="acctNoList" type="acctNoList\_type"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="registerService" type="registerService\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

## Fee (fee.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="calFee\_type">

<xs:sequence>

<xs:element name="feeCode" type="xs:string"/>

<xs:element name="transID" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="toBranchCode" type="xs:string"/>

<xs:element name="transType" type="xs:string"/>

<xs:element name="fromAcc" type="xs:string"/>

<xs:element name="toAcc" type="xs:string"/>

<xs:element name="fromOrgBranchCode" type="xs:string"/>

<xs:element name="toOrgBranchCode" type="xs:string"/>

<xs:element name="amount" type="xs:string"/>

<xs:element name="currencyCode" type="xs:string"/>

<xs:element name="customerType" type="xs:string"/>

<xs:element name="productType" type="xs:string"/>

<xs:element name="businessType" type="xs:string"/>

<xs:element name="officeCode" type="xs:string"/>

<xs:element name="micNumber" type="xs:string"/>

<xs:element name="channel" type="xs:string"/>

<xs:element name="bankType" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="feeRequest\_type">

<xs:sequence>

<xs:element name="amount" type="xs:string"/>

<xs:element name="bankType" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="businessType" type="xs:string"/>

<xs:element name="channel" type="xs:string"/>

<xs:element name="currencyCode" type="xs:string"/>

<xs:element name="customerType" type="xs:string"/>

<xs:element name="feeCode" type="xs:string"/>

<xs:element name="fromAcc" type="xs:string"/>

<xs:element name="fromOrgBranchCode" type="xs:string"/>

<xs:element name="micNumber" type="xs:string"/>

<xs:element name="officeCode" type="xs:string"/>

<xs:element name="productType" type="xs:string"/>

<xs:element name="toAcc" type="xs:string"/>

<xs:element name="toBranchCode" type="xs:string"/>

<xs:element name="toOrgBranchCode" type="xs:string"/>

<xs:element name="transID" type="xs:string"/>

<xs:element name="transType" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="calFeeObj\_type">

<xs:sequence>

<xs:element name="feeParams" type="feeRequest\_type"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="calFee" type="calFee\_type" minOccurs="0"/>

<xs:element name="calFeeObj" type="calFeeObj\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

## SMS banking (SMSBanking.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="SMSRequest\_type">

<xs:sequence>

<xs:element name="content" type="xs:string"/>

<xs:element name="numberGW" type="xs:string"/>

<xs:element name="phoneNumber" type="xs:string"/>

<xs:element name="sequence" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="putSMS\_type">

<xs:sequence>

<xs:element name="SMSRequest" type="SMSRequest\_type"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="regSMSBanking\_type">

<xs:sequence>

<xs:element name="cifNumber" type="xs:string"/>

<xs:element name="accountNumber" type="xs:string"/>

<xs:element name="serviceType" type="xs:string"/>

<xs:element name="alert" type="xs:string"/>

<xs:element name="phoneNumber" type="xs:string"/>

<xs:element name="branchCode" type="xs:string"/>

<xs:element name="appReg" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="putSMS" type="putSMS\_type" minOccurs="0"/>

<xs:element name="regSMSBanking" type="regSMSBanking\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

## Tsunami (Tsunami.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="SendMessage\_type">

<xs:sequence>

<xs:element name="number"/>

<xs:element name="command"/>

<xs:element name="content"/>

<xs:element name="sender"/>

<xs:element name="sign"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="SendMessage" type="SendMessage\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

## SMS OTP (otp.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="checkOtp\_type">

<xs:sequence>

<xs:element name="transactionID" type="xs:string"/>

<xs:element name="otpTransactionId" type="xs:long"/>

<xs:element name="iValueTime" type="xs:int"/>

<xs:element name="sOtp" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="getOtp\_type">

<xs:sequence>

<xs:element name="transactionID" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="checkOtp" type="checkOtp\_type" minOccurs="0"/>

<xs:element name="getOtp" type="getOtp\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

## Fund Transfer code (Transfercode.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="checkOtp\_type">

<xs:sequence>

<xs:element name="transactionID" type="xs:string"/>

<xs:element name="otpTransactionId" type="xs:long"/>

<xs:element name="iValueTime" type="xs:int"/>

<xs:element name="sOtp" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="getOtp\_type">

<xs:sequence>

<xs:element name="transactionID" type="xs:string"/>

<xs:element name="length" type="xs:int"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="getPassCode\_type">

<xs:sequence>

<xs:element name="customerId" type="xs:string"/>

<xs:element name="clientId" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createPassCode\_type">

<xs:sequence>

<xs:element name="customerId" type="xs:string"/>

<xs:element name="clientId" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="createPassCodeLength\_type">

<xs:sequence>

<xs:element name="customerId" type="xs:string"/>

<xs:element name="clientId" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

<xs:element name="length" type="xs:int"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="resetPassCode\_type">

<xs:sequence>

<xs:element name="customerId" type="xs:string"/>

<xs:element name="clientId" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

<xs:element name="length" type="xs:int"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="updatePassCode\_type">

<xs:sequence>

<xs:element name="customerId" type="xs:string"/>

<xs:element name="clientId" type="xs:string"/>

<xs:element name="newTranferPass" type="xs:string"/>

<xs:element name="oldTranferPass" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="checkPassCode\_type">

<xs:sequence>

<xs:element name="customerId" type="xs:string"/>

<xs:element name="clientId" type="xs:string"/>

<xs:element name="transferCode" type="xs:string"/>

<xs:element name="description" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="checkOtp" type="checkOtp\_type" minOccurs="0"/>

<xs:element name="getOtp" type="getOtp\_type" minOccurs="0"/>

<xs:element name="getPassCode" type="getPassCode\_type" minOccurs="0"/>

<xs:element name="createPassCode" type="createPassCode\_type" minOccurs="0"/>

<xs:element name="createPassCodeLength" type="createPassCodeLength\_type" minOccurs="0"/>

<xs:element name="resetPassCode" type="resetPassCode\_type" minOccurs="0"/>

<xs:element name="updatePassCode" type="updatePassCode\_type" minOccurs="0"/>

<xs:element name="checkPassCode" type="checkPassCode\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

## RSA AM (rsa.xsd)

<?xml version="1.0"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="http://www.w3schools.com"

xmlns="http://www.w3schools.com"

elementFormDefault="qualified">

<xs:complexType name="header\_type">

<xs:sequence>

<xs:element name="req\_id" type="xs:int"/>

<xs:element name="req\_app" type="xs:string"/>

<xs:element name="srv" type="xs:int"/>

<xs:element name="srv\_app" type="xs:string"/>

<xs:element name="req\_time" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="checkOTP\_type">

<xs:sequence>

<xs:element name="userId" type="xs:string"/>

<xs:element name="otp" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="deleteUser\_type">

<xs:sequence>

<xs:element name="userId" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="Unassign\_type">

<xs:sequence>

<xs:element name="sn" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="Assign\_type">

<xs:sequence>

<xs:element name="userId" type="xs:string"/>

<xs:element name="sn" type="xs:string"/>

<xs:element name="pin" type="xs:int"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="CreateUser\_type">

<xs:sequence>

<xs:element name="userId" type="xs:string"/>

<xs:element name="pwd" type="xs:string"/>

<xs:element name="fname" type="xs:string"/>

<xs:element name="lname" type="xs:string"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="body\_type">

<xs:sequence>

<xs:element name="CreateUser" type="CreateUser\_type" minOccurs="0"/>

<xs:element name="Assign" type="Assign\_type" minOccurs="0"/>

<xs:element name="Unassign" type="Unassign\_type" minOccurs="0"/>

<xs:element name="deleteUser" type="deleteUser\_type" minOccurs="0"/>

<xs:element name="checkOTP" type="checkOTP\_type" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:element name="Data">

<xs:complexType>

<xs:sequence>

<xs:element name="Header" type="header\_type"/>

<xs:element name="Body" type="body\_type"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>