Faculty of Engineering and Technology
Master of Software Engineering (SWEN)

SWEN6307: SERVICE-ORIENTED SOFTWARE ENGINEERING
Second Semester 2016/2017

Assignment2: Simulation for Web Services Architecture on Paltel Telecommunication Company

Instructor: Dr. Nariman Ammar

Date: 29 April 2017

Team Members

Student ID	Student Name
1155124	Ibrahim Assi
1155073	Nadeem Al Kilani
1155072	Tarik Hannoun

ABSTRACT:

With the rapid improvement of the technologies, PALTEL find that it's become necessary to stratify the customer's needs, and tailor system to their customers in order to allow them to interest with many services and campaigns, for this reason Paltel decide to make a serious changes on the integration environment between the legacy systems and the new required systems for their improvement plan.

What we try to do on this project is to simulate how that Paltel as a telecommunication company decided few years ago to utilize the existing technologies to build their architecture design by take a decision to build a service oriented architecture after they were using not effective technologies for years such as a Database links, which prevented them from improving their business needs and Keep pace with development.

So we introduce some of the enterprise systems like CRM and billing systems, and showed how service oriented technology (web servers) used as an integration point between Paltel systems to interact effectively and used as a secure layer to expose Paltel systems to public internet and how they integrate with BANK system.

KEYWORDS: CRM System, Billing System, Legacy Systems, LAN network, DMZ network, SOAP, Web server, Portal, Mobile app

DESCRIPTION:

PALTEL company is a big telecommunication company deals with Hundreds of thousands customers and operates millions of transactions, so that it has enterprise systems to deals with customer's bills, payments, installations, etc. They have CRM and billing system to manage the customer's fixed lines transactions, in addition to many types of "Value Added Services" and other new services, also they used to manage complex campaigns to encourage their customers to subscribe with these new services.

Of course all of these things operates by an enterprise system that all integrates with each other In order to facilitate the customer to deal with these services in terms of purchase, activate, deactivate services, installment payments, apply faults, complaints with one click either via mobile app or portal we site.

Since PALTEL allows the external customers to access their applications, they divided their networks to two networks:
1. LAN network (Local secured network), which contains:
    a. Legacy systems (CRM and billing) with their local Databases.
    b. Local web server
    c. Portal and Mobile app Administration

2. DMZ network (Public network, exposed to public internet), which contains:
   a. Paltel web site (portal)
   b. Public web server

Public network DMZ opened for public users with least privilege that is enabled them to brows the portal, and act as a mediation secure layer include (Public web server) to allow mobile application to access local network securely. Paltel has 3-tier architecture style (backend – middleware – client), **Figure1,**
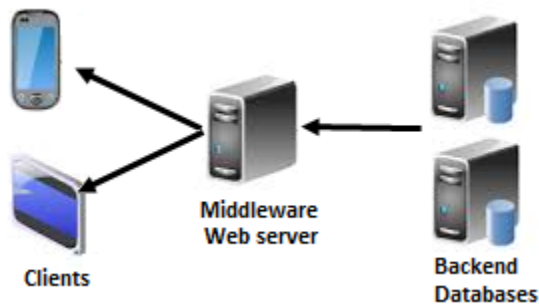

Figure1

But when deciding the network to LAN and DMZ and including the Internet as tier for mobile apps and integrated system such as the bank, we can say that we have N-tier architecture style,
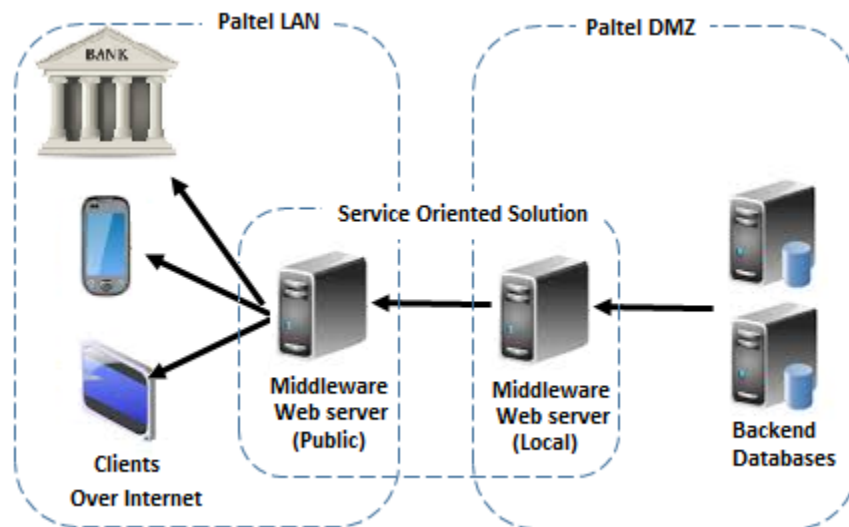

Figure2

All in order to provide the customer with an easy and effective mechanism to manage his accounts and interesting with PALTEL service in a secure way without having to visit the showroom. of course these DMZ applications integrates with the CRM and Billing system through web services.

By adopting Service Oriented Architecture, and to achieve this architecture, PALTEL implemented 2 web servers, as mentioned previously, one on LAN and the other on DMZ for security issues and used a SOAP technology as they have many legacy systems integrated with CRM and billing such as financial system, pointing, etc., so it makes deal with SOAP technology more flexible and easy.

Also by using SAOP technology PLATEL increase their benefits form the existing legacy system that were built on different technologies and increased the security level, so each team is responsible only to his job with no need to know more about the needed integration system, or to know how the other system built , what is only to get an API , also as it known by using web services we support the LESKOV principle which is apply open –close design , so each system doesn't affect with any change happened to the other system , also we can rebuilt the other system and changed the technology with any affect to the other systems.

The below figure (**Figure3**) shows PALTEL SOA architecture that describe how local network zone, consists of CRM and billing systems, interconnect with DMZ zone via web services implemented on web servers,

The customer will connect with Paltel system through the Portal web application in Paltel DMZ network or mobile application on public internet. The web server in DMZ zone contains all services related to customer requests. Customer can perform multi services such as view his information, view his bills and pay unpaid bills using VISA, these operations are complex and done on different level of integration between PALTEL organization itself (LAN, DMZ), the BANK organization and Public Internet.

Below in the SCOPE section a more detailed description exists about the operation that might take place as in the architecture figure3
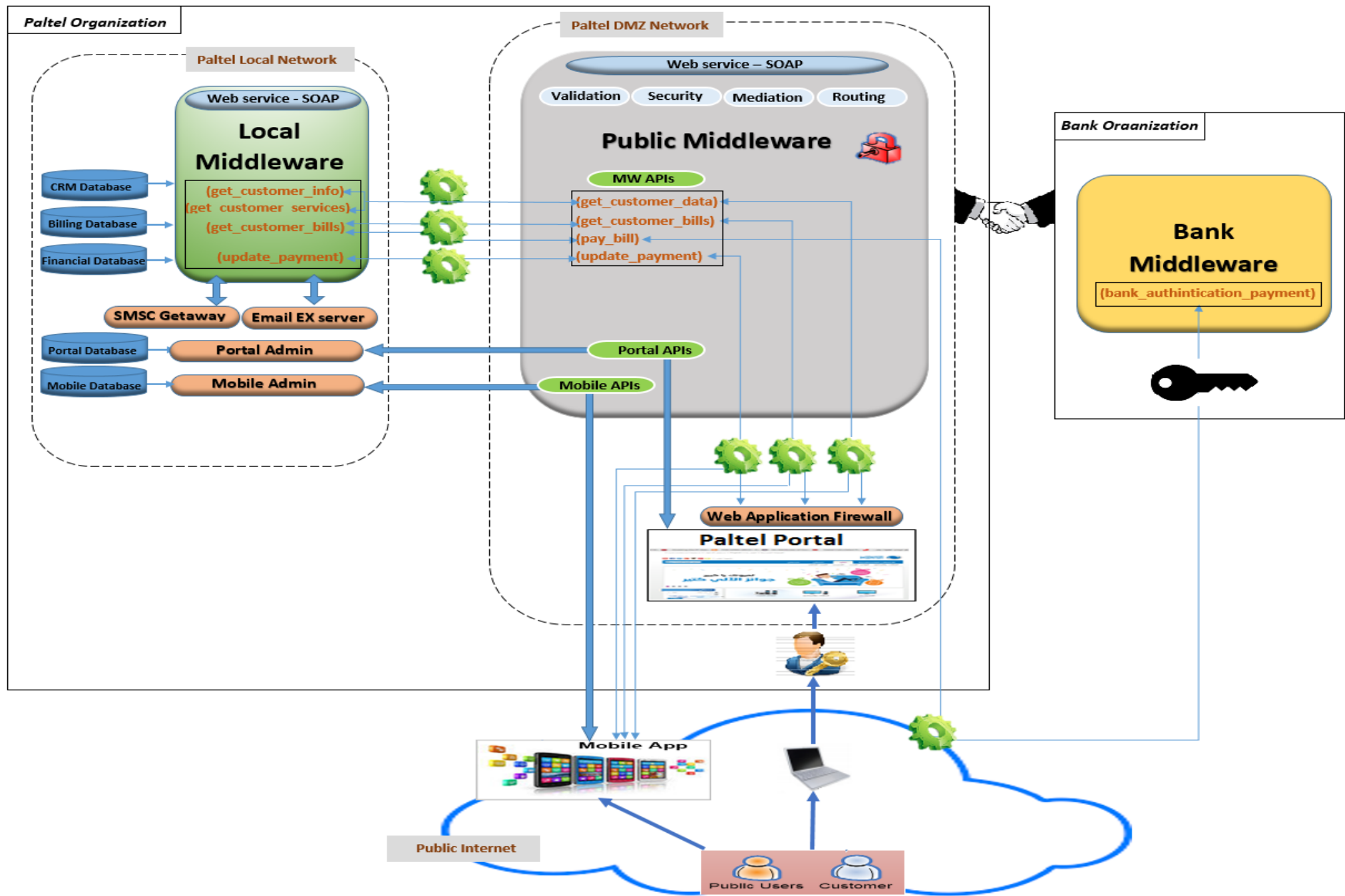
Figure3: Simulation for Paltel Service Oriented Architecture – middleware (Web Services Architecture), includes networks LAN & DMZ

SCOPE

In this project we will try to show SOA-based solution as a simulation environment on Paltel problem domain, including hosting organizations and client applications, this done through implement a few services (composite services involved atomic services) such as get customer's information , get customer's bills , get customer's services and Add Payment , so we will implement our idea through portal Application and mobile Application.

Properties Client applications (portal and mobile systems):

A. Portal application will be hosted under Apache Application server, using php programming language and SAOP web services.

B. Mobile application will be hosted under JBoss Application server, using Android framework, and SAOP web services.

So the customer will follow the following procedure if he want to manage his services either by using mobile App or the portal App (referenced as a Client App) :-

Basic Operations:

1. Customer type his Tel no and password
2. authenticate the customer.
3. Client App shows the customer information like his name, address, email, etc., and his services
4. Client App shows the active services of the customers and the subscription date of each service
5. Customer request to see his bills.
6. Client App shows unpaid bills.
7. Once the customer request to pay his bill, the system will make authentication on the bank gateway and call the web service to perform transaction payment.
8. Client App check the returned value from the Bank web services in order to know the status of the payment.
9. Either of the payment was success or fail, the system will send email and SMS to the customer to inform him with transaction status
10. If the payment was success the additional step will be performed which is to register the successful payment to the customer through call web service to transfer this payment

Basic Operations (detailed description of the architecture operations):

Note: duplicated web services on both web servers (LAN & DMZ), because of security issues.

1- Login to the portal (Tel no and password)
2- Retrieve Customer Information automatically:

   a. Portal main page retrieve automatically customer information via composite service "get_customer_data" on the DMS web server.
   b. composite service "get_customer_data" on the DMS web server, call 2 atomic web services from LAN web server:
      i. 1st atomic service "get_customer_info", retrieve basic customer information (customer name, address, mobile, email, telephone number, area code, fax number, customer classification)
      ii. 2nd atomic service "get_customer_services", retrieve customer active services (telephone number, service name)

3- Retrieve Billing information:

   Customer request to view his bills by click a button, then portal call web service "get_customer_bills" from DMZ web server, which call a copy of the same web service on the LAN web server, this service retrieve (bill number, telephone number, bill period, bill amount, currency, due date, service name, amount)

4- Bill Payment:

   a. Customer request to pay for a specific bill via choose a bank payment
   b. Portal request to pay specific bill through web service "pay_bill" on DMZ web server, this is a composite web service that call 2 other web services:
      i. 1st web services from LAN web server that retrieve customer bill information to send it to bank web service "get_customer_bills"
      ii. 2nd web service, is the bank web service '' "bank_authintication_payment", 'pay_bill' service call this service to process the payment on the bank, portal call this web service to authenticate customer bank account and send customer bill info.
   c. Bank web service "bank_authintication_payment", return payment status.
   d. Portal update payment status as paid via web service "update_payment", from DMZ web server, which call a copy of the same web service on the LAN web server, the LAN web service update the payment with the payment type and bank transaction number.

Data models

- **CRM data model:**
  - **customer information:** (account number, customer name, address, mobile, email, telephone number, area code, fax number, customer classification, customer type "residential, corporate", subscription date, line status)
  - **Payments:** (bill number "unique", account number, telephone number, payment due date, bill amount, bill currency, payment amount, payment currency, currency rate, payment type "cash, bank", payment location name, bank trans number)
  - **Services:** (account number, telephone number, service name, service status, subscription date)
- **Billing data model:**
  - **Bill header:** (bill number, account number, telephone number, bill period, bill amount, currency, due date)
  - **Bill summary:** (bill number, account number, telephone number, service name, amount)
- **Bank data model:**
  - **Bank trans:** (bank name, bank trans number, amount, currency)

Technologies:

1- Three backend systems as a database schemes on "MySQL" to provide data:
   a. CRM (2tables customer info, services and payment)
   b. Billing (bill information)
   c. Bank (transaction)
2- Two middleware servers:
   a. First middleware exists on LAN environment, built on Apache/php technology and has web services based on SOAP technology and provide WSDL to help client to connect its services.
   b. Second middleware exists on DMZ (public to Internet) environment, built on JBoss technology and has web services based on SOAP technology, provide WSDL to help client to connect its services.
3- Two Clients, each will use web services from the second middleware DMZ (JBoss):
   a. First client is a web application built on php technology
   b. Second client is a mobile application built on android technology

   Both clients depend on WSDL information from second middleware web server to connect the web services

Paradigm to be used :

1- SOAP: will be implemented on the LAN web server.
2- SOAP: will be implemented on the DMZ web server.

Data representation:

1- In SOAP: xml data

Appendix of Web services used

WS-01 Get Customer Information :

This process returns all information related to the customer

WSDL URL: http://....... /.... /... /getCustomerinfo?wsdl
Inputs

Name	Data Type	Comment
areaCode	String	Like 09 , 02
TelNo	String	Like 2810202 (without area code)

Outputs:

The output will be Text XML object

Name	Data Type	Comment
Account_number	String	This account used as umbrella to hold many tel number related to the same customer
Customer_name	String	
Address	String	
Mobile	String	
Email	String	
Telephone_number	String	
Area_code	String	
Fax_number	String	
Customer_classification	String	Platinum ,Golden ,Sliver,…ect.
Customer_type	String	Residential, Corporate
Subscription_date	String	
Line_status	String	Active or inactive

WS-02 Get Customer services :

This process returns all active services that is installed on customer's Tel number

WSDL URL: http://....... /.... /... /getCustomerServices?wsdl
Inputs

Name	Data Type	Comment
areaCode	String	Like 09 , 02
TelNo	String	Like 2810202 (without area code)

Outputs:

The output will be Text XML object

Name	Data Type	Comment
Account_number	String	This account used as umbrella to hold many tel number related to the same customer
Telephone_number		

Service_name		Like caller ID ,caller ID+ , Follow me ,…etc
Service_status		Active .inactive
Subscription_date		

WS-03 Get Bill Header :

This process returns the last 6 bills of the customers with a brief information as mentioned on the table below

WSDL URL: http://……. /…. /… /getBillHeader?wsdl

Inputs

Name	Data Type	Comment
areaCode	String	Like 09 , 02
TelNo	String	Like 2810202 (without area code)

Outputs:

The output will be Text XML object

Name	Data Type	Comment
Account_number	String	This account used as umbrella to hold many tel number related to the same customer
Bill_number	String	
Telephone_number	String	
Bill_period	String	
Bill_amount	String	
Currency	String	
Due_date	String	

WS-04 Get Bill Summary :

This process returns summary of the bills based on each bill period

WSDL URL: http://……. /…. /… /getBillSummary?wsdl

Inputs

Name	Data Type	Comment
areaCode	String	Like 09 , 02
TelNo	String	Like 2810202 (without area code)
Bill_period	String	Like 072017

Outputs:

The output will be Text XML object

Name	Data Type	Comment
Account_number	String	This account used as umbrella to hold many tel number related to the same customer

Bill_number	String	
Telephone_number	String	
Service_name	String	
Amount	String	

WS-05 Pay Bill :

This process is responsible to add payment to the bank

WSDL URL: http://....... /.... /... /payBill?wsdl
Inputs

Name	Data Type	Comment
areaCode	String	Like 09 , 02
TelNo	String	Like 2810202 (without area code)

Outputs:

The output will be Text XML object

Name	Data Type	Comment
Account_number	String	This account used as umbrella to hold many tel number related to the same customer
Bill_number	String	
Telephone_number	String	
Amount	String	
Currency		