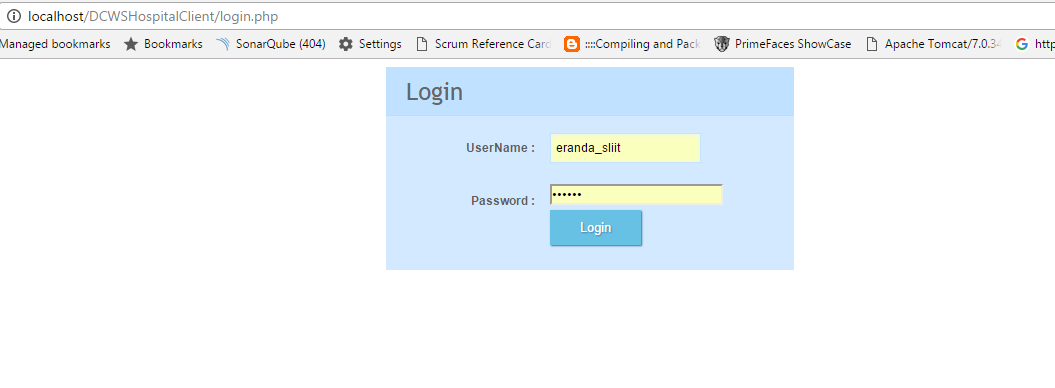
**DCWS Mid Assignment G.G.E.P Gunasena (MS16908284)**

**Work Flow of the System**

Go to below url and enter user name password. Here user details should enter as master data in to “hospitaluser” table since there is no user creation module.

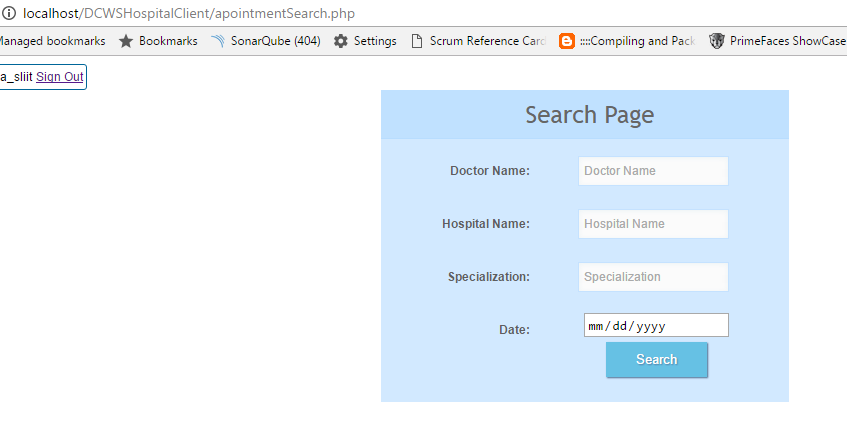
Note: In this assignment have used predefine css stylsheet for interface formatting

<http://localhost/DCWSHospitalClient/login.php>



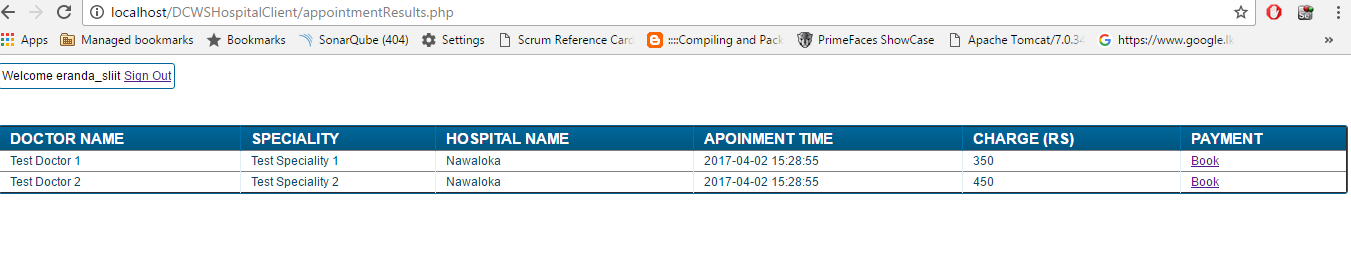
After success login user will direct to search page with search criteria form.

<http://localhost/DCWSHospitalClient/apointmentSearch.php>



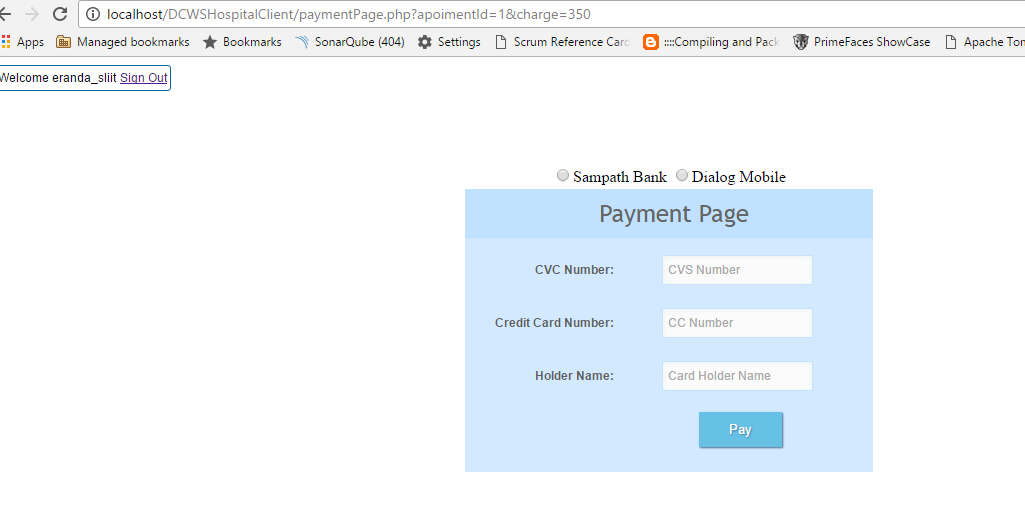
After search with criteria you will forward to below search results page.

<http://localhost/DCWSHospitalClient/appointmentResults.php>

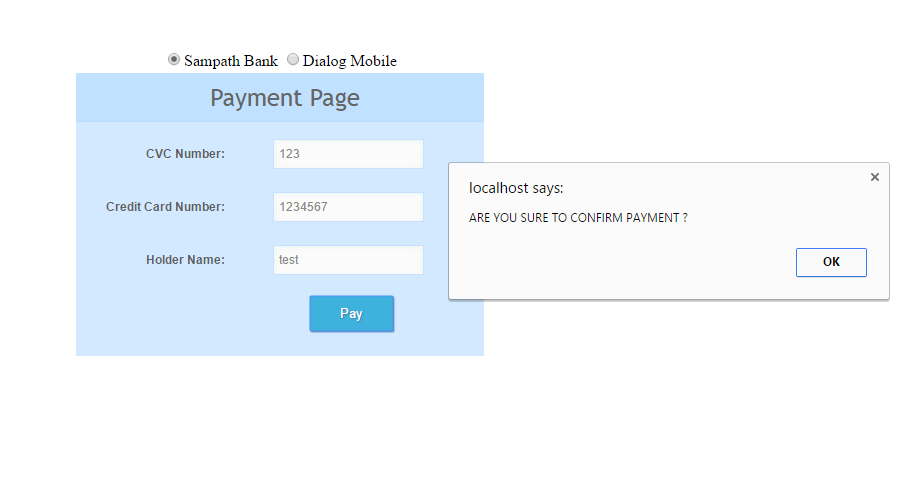


Each appointment record will have “Book” button and that button will redirect to payment page with payment options.

<http://localhost/DCWSHospitalClient/paymentPage.php>



Confirmation Page



Then user can select payment method as Sampath Bank PG or Dialog mobile payment.

After selecting payment option with details request will direct to specifi PG url in to the service layer and after that saving the request details request will pass to particular payment gateway url. (In this assignment this PG urls are test link).

After success payment request success response will display to client interface.

**Technology and Tools which are used to implement**

Front End – PHP 7/Java Script/JSON/Mysql

Service tier – Java/Spring/Hibernate/Mysql

ESB – wso2esb-5.0.0 / API

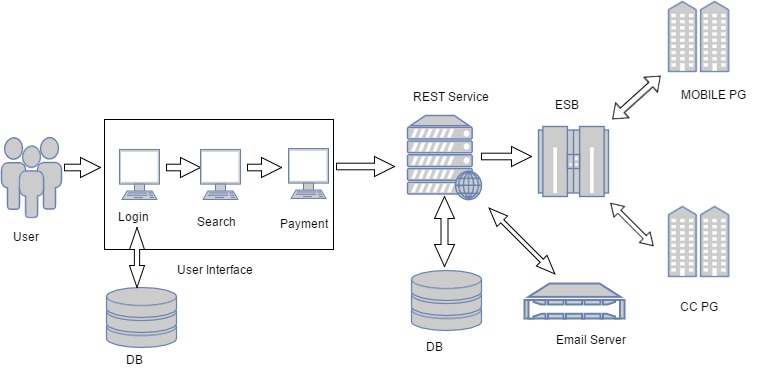
Build Tools – Maven / Eclipse

Log tools – Log back spring

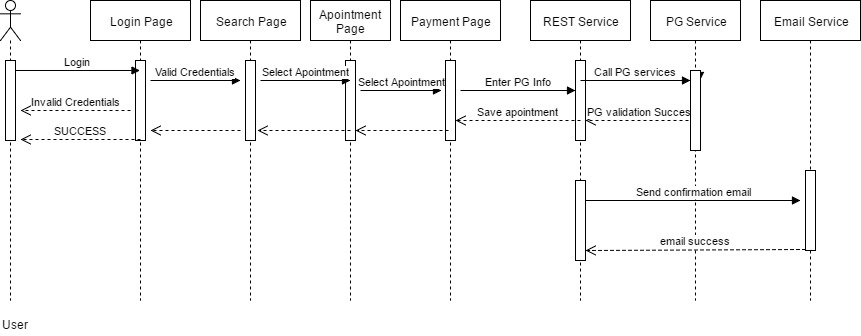
Servers – WsoEsb Server / Xamp

API documentation – Swagger documentation

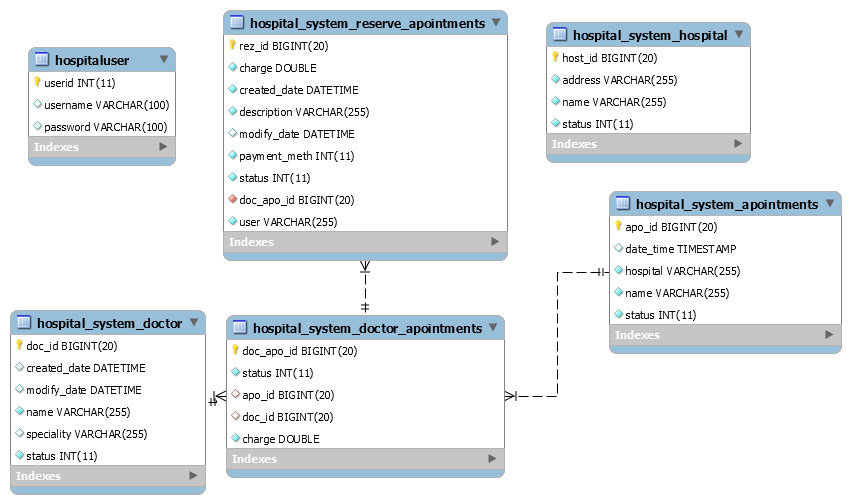
**High level architecture diagram of the system**



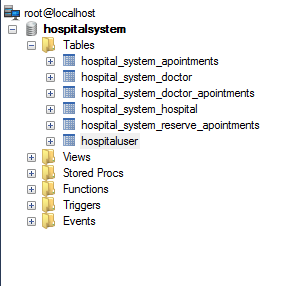
**Sequence diagram of the system**



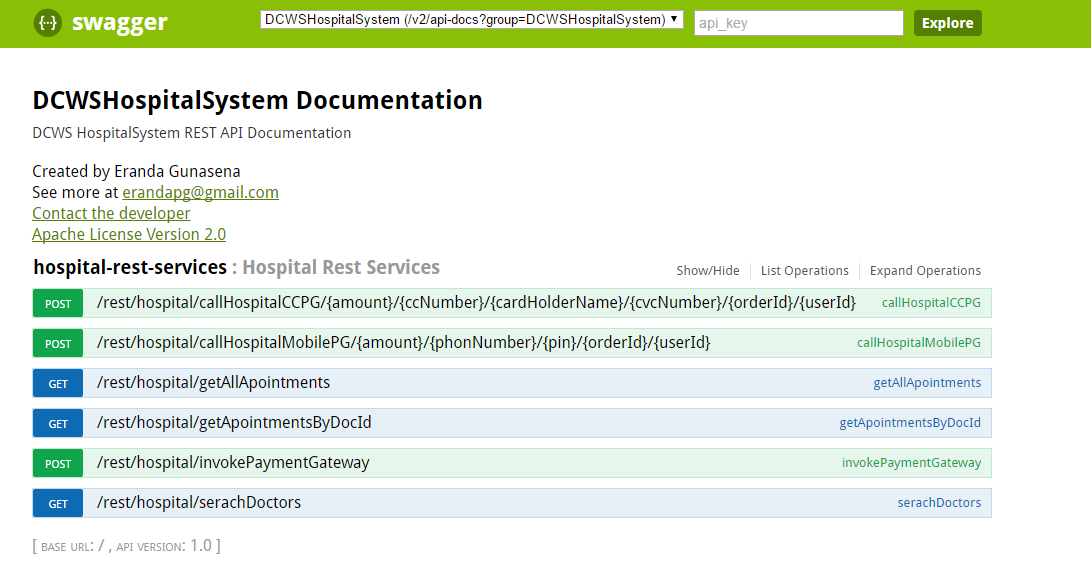
**Entity Relationship diagram of the system**



**Table structure relevant to the application**



**Rest APIs which are implemented**



**Deployment procedure (Server)**

Build eclipse project using below commands.

Cd {workspace}\DCWSHospitalSystem

Mvn –Dmaven.test.skip=true clean package

Copy war file in project target falder in to below path

{wso2as-5.3.0\_home}\repository\deployment\server\webapps

Start wso2as server using below command

Cd {wso2as-5.3.0\_home}\bin

./ wso2server.sh start

**Deploy Procedure (Client)**

PHP Client need to be deploy in XAMP server.

**Deploy ESB API**

Go to ESB carbon admin console

Go to API link

Create New API and copy paste below configurations.

<api xmlns="http://ws.apache.org/ns/synapse" name="paymentGatewayTest" context="/dcws/hospitalsystem">

<resource methods="OPTIONS POST" uri-template="/ccpayment/requests/{amount}/{cardHolderName}/{ccNumber}" faultSequence="fault">

<inSequence>

<log level="custom">

<property name="amount" expression="get-property('uri.var.amount')"/>

</log>

<payloadFactory media-type="json">

<format>{ "amount": "$1", "cardHolderName": "$2", "ccNumber": "$3", "currency": "$4", "cvcNumber": "$5", "language": "$6", "orderId": "$7", "paymentType": "$8", "responseUrl": "$9", "userId": "$10" } </format>

<args>

<arg evaluator="json" expression="get-property('uri.var.amount')"/>

<arg evaluator="json" expression="get-property('uri.var.cardHolderName')"/>

<arg evaluator="json" expression="get-property('uri.var.ccNumber')"/>

<arg evaluator="json" expression="get-property('uri.var.currency')"/>

<arg evaluator="json" expression="get-property('uri.var.cvcNumber')"/>

<arg evaluator="json" expression="get-property('uri.var.language')"/>

<arg evaluator="json" expression="get-property('uri.var.orderId')"/>

<arg evaluator="json" expression="get-property('uri.var.paymentType')"/>

<arg evaluator="json" expression="get-property('uri.var.responseUrl')"/>

<arg evaluator="json" expression="get-property('uri.var.responseUrl')"/>

</args>

</payloadFactory>

<send>

<endpoint>

<http method="POST" uri-template="http://localhost:8080/rest/hospital/invokePaymentGateway"/>

</endpoint>

</send>

</inSequence>

<outSequence>

<log level="custom">

<property name="status" expression="json-eval($.status)"/>

<property name="transactionId" expression="json-eval($.transactionId)"/>

</log>

<property name="NO\_ENTITY\_BODY" scope="axis2" action="remove"/>

<property name="messageType" value="application/json" scope="axis2" type="STRING"/>

<log level="full"/>

<send/>

</outSequence>

</resource>

<resource methods="OPTIONS POST" uri-template="/mobilepayment/requests/{amount}/{cardHolderName}/{ccNumber}" faultSequence="fault">

<inSequence>

<log level="custom">

<property name="amount" expression="get-property('amount')"/>

</log>

<payloadFactory media-type="json">

<format>{ "amount": "$1", "cardHolderName": "$2", "ccNumber": "$3", "currency": "$4", "cvcNumber": "$5", "language": "$6", "orderId": "$7", "paymentType": "$8", "responseUrl": "$9", "userId": "$10" } </format>

<args>

<arg evaluator="json" expression="get-property('uri.var.amount')"/>

<arg evaluator="json" expression="get-property('uri.var.cardHolderName')"/>

<arg evaluator="json" expression="get-property('uri.var.ccNumber')"/>

<arg evaluator="json" expression="get-property('uri.var.currency')"/>

<arg evaluator="json" expression="get-property('uri.var.cvcNumber')"/>

<arg evaluator="json" expression="get-property('uri.var.language')"/>

<arg evaluator="json" expression="get-property('uri.var.orderId')"/>

<arg evaluator="json" expression="get-property('uri.var.paymentType')"/>

<arg evaluator="json" expression="get-property('uri.var.responseUrl')"/>

<arg evaluator="json" expression="get-property('uri.var.responseUrl')"/>

</args>

</payloadFactory>

<send>

<endpoint>

<http method="POST" uri-template="http://localhost:8080/rest/hospital/invokePaymentGateway"/>

</endpoint>

</send>

</inSequence>

<outSequence>

<log level="custom">

<property name="status" expression="json-eval($.status)"/>

<property name="transactionId" expression="json-eval($.transactionId)"/>

</log>

<property name="NO\_ENTITY\_BODY" scope="axis2" action="remove"/>

<property name="messageType" value="application/json" scope="axis2" type="STRING"/>

<log level="full"/>

<send/>

</outSequence>

</resource>

</api>