**Overview**

**Old system structure: the old system built using java EE, and struts framework.**

**Struts: is a one of the first MVC frameworks for developing java applications.**

**The whole logic of the old system located in:**

**WEB-INF/classes/com/raya/nsgb/business/**

**In business directory: you’ll find main domains like: admin, reporting, subscription, and so on.**

**-----------------------------------------------------------------------------------------------**

**Subscription Management**

**In business/subscription/pipeline/: you’ll find all the actions related to subscription management.**

**In pipeline directory: there are classes and interfaces, each interface names like:**

**I<Action>Pipeline 🡪 I for interface.**

**The implementation of this interface: you can navigate to it easily using IntelliJ, or you will find it named like: <Action>Pipeline.**

**Finally, in <Action>Pipeline class, you will find the logic of this action.**

**-----------------------------------------------------------------------------------------------**

**Request Management**

**In business/subscription/pipeline/IIssueRequestPipeline, IGeneratePinCodePipeline, IListPendingIssue: you will find the logic of the request management.**

**Request Tracking and Resolution**

**In business/subscription/pipeline/IRequestsPipeline: some of request tracking methods written there.**

**-----------------------------------------------------------------------------------------------**

**Staff Activity Audit**

**In business/reporting/pipeline: I think this part is related to the reports, I’m not sure btw, I’ll check it again tomorrow ISA.**

**-----------------------------------------------------------------------------------------------**

**Reports Requirements**

1. **Customer interaction report:**

You will find the queries for this report in: **IvrAuditDetailsReportPipeline**,

**IvrAuditReportPipeline.**

The base query in **IvrAuditDetailsReportPipeline** is:

FROM MwAuditActions A JOIN FETCH A.actionId C + whereClause + order by A.id

String where = " WHERE ";

if (criteria.get("Call ID") != null && !criteria.get("Call ID").equals("")) {

where = where + " A.callId =" + criteria.get("Call ID") ;

criteria.remove("Call ID"); }

if (criteria.get("Call UUID") != null && !criteria.get("Call UUID").equals("")) {

where = where + " A.callUUID =" + criteria.get("Call UUID") ;

criteria.remove("Call UUID"); }

then 🡪 this report gets all the actions related to a call using the call id or uuid or both.

The base query in **IvrAuditReportPipeline** is:

FROM MwAuditCalls A LEFT JOIN FETCH A.clientID C"+ whereClause+" order by A.callID

String where = " WHERE ";

if (criteria.get("From Date") != null && criteria.get("To Date") != null) {

if (criteria.get("Monthly Date") == null || ((String) criteria.get("Monthly Date")).equals("false"))

where = where + " A.startDate >= TO\_DATE('" + criteria.get("From Date") + "', 'dd/mm/yyyy')" + " AND A.startDate <= TO\_DATE('" + DateUtil.nextDayOrMonth((String) criteria.get("To Date")) + "', 'dd/mm/yyyy') ";

else

where = where + " AND A.startDate between TO\_DATE('" + criteria.get("From Date") + "', 'dd/mm/yyyy')" + " and TO\_DATE('" + DateUtil.nextDayOrMonth((String) criteria.get("To Date")) + "', 'mm/yyyy') ";

criteria.remove("Monthly Date");

criteria.remove("From Date");

criteria.remove("To Date");

}

if (criteria.get("Client ID") != null && !criteria.get("Client ID").equals("")) {

where = where + " AND A.clientID like '%" + criteria.get("Client ID") + "%' ";

criteria.remove("Client ID");

}

if (criteria.get("Subscription ID") != null && !criteria.get("Subscription ID").equals("")) {

where = where + " AND C.subscriptionId like '%" + criteria.get("Subscription ID") + "%' ";

criteria.remove("Subscription ID");

}

if (criteria.get("Mobile No") != null && !criteria.get("Mobile No").equals("")) {

where = where + " AND A.mobileNo like '%" + criteria.get("Mobile No") + "%' ";

criteria.remove("Mobile No");

}

if (criteria.get("Branch Code") != null && !criteria.get("Branch Code").equals("")) {

where = where + " AND A.branchCode like '%" + criteria.get("Branch Code") + "%' ";

criteria.remove("Branch Code");

}

if (criteria.get("Call UUID") != null && !criteria.get("Call UUID").equals("")) {

where = where + " AND A.callUUID = '" + criteria.get("Call UUID") +"'";

criteria.remove("Call UUID");

}

return where;

then 🡪 this report gets all the calls by some criteria fields, includes:

date range (from:to), client id, mobile no, subscription id, call uuid, branch code,