

Story Using Design Thinking

Monitoring of interfaces is critical for any organization and is not restricted to any specific industry. There is a need to view the snapshot of interface processing stats for reporting and monitoring purposes. This task is often performed by the support managers/system managers and consumes majority of their time. The entire process of monitoring interfaces is time consuming (and frustrating at times) because

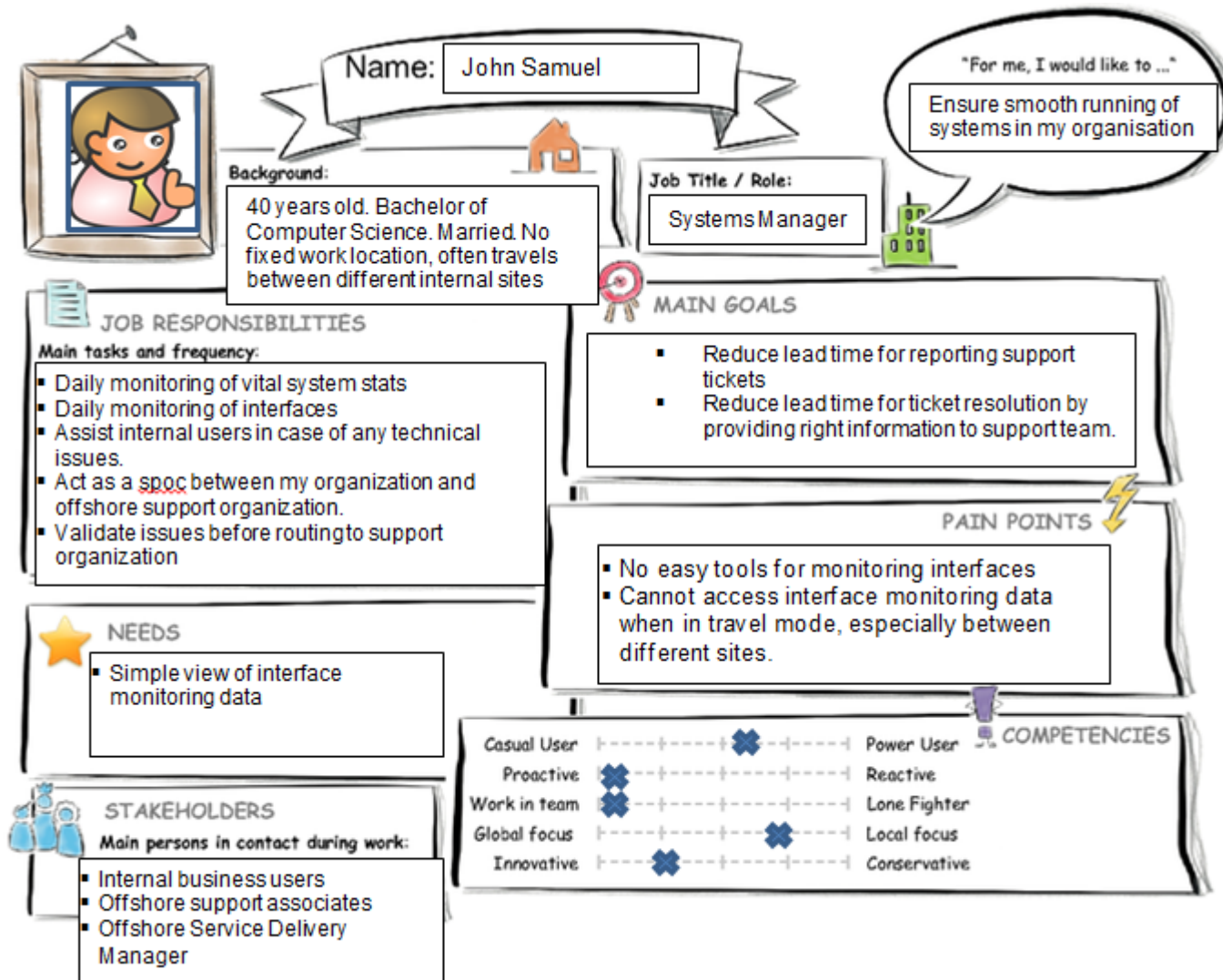
- Information is **scattered across different transactions** like
 - WE05 for IDoc monitoring
 - SMQ1,SMQ2 –For queued RFC monitoring
 - SM58 – For Transactional RFC monitoring
 - SXMB_MONI – Local Integration engine monitoring
- Navigating between transactions is not easy in SAP especially when we have **restriction on number of open sessions**.
- These transactions are **very complex and present a wide range of data** which at times confuses the end user. Often the data is presented in a cryptic and misleading way and relies on the experience of end user to figure out right information.
- Due to the nature of these transactions, they are **not available online**. Hence the end user will need a laptop/desktop with VPN connectivity to retrieve stats.
- Average time taken to complete this process is **20-30 mins** depending on the volume of interfaces.

These transactions may be helpful for technical consultants for debug & trouble shooting but definitely are not meant for support managers/system managers. Support /system managers are **mainly interested in**

- Number of successful/failed/waiting IDocs in a given timeframe
- RFC queue stats with number of waiting/failed entries in the queue in a given timeframe.
- Number of successful/failed/waiting messages in local integration engine for a specific time frame
- More importantly access all the above **information from a single view over internet**. This will help them to access the information anywhere anytime and instantly route errors to support desk using remedy tickets/calls.

All the support /system managers wish to have an application to monitor interfaces which is simple, coherent and delightful. The good news is we can fulfill their wish by building a SAP Fiori application for monitoring interfaces

Persona of John Samuel

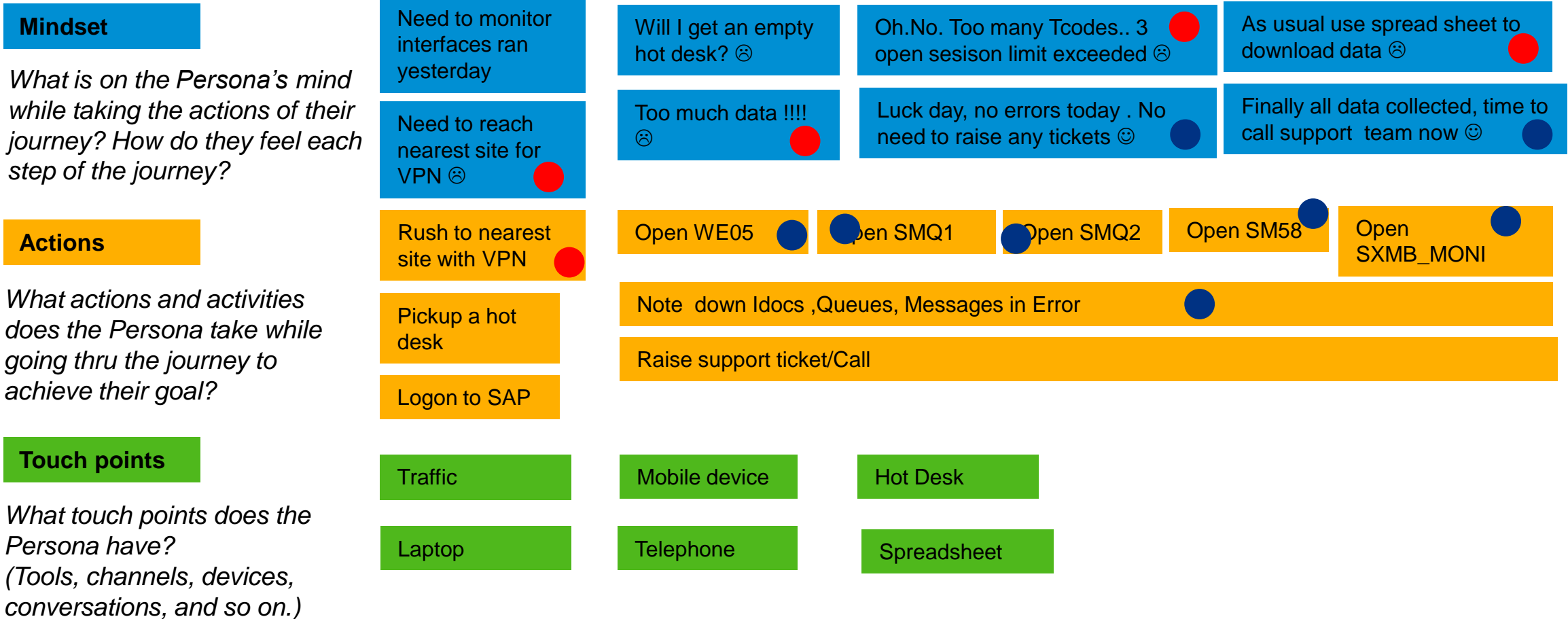


Name	John Samuel
"For me, I would to...":	Ensure smooth running of systems in my organisation
Background	40 years old. Married, Bachelor of Computer Science. No fixed work location, often travels between different internal sites of the organisation.
Job Title / Role	Systems Manager
Job Responsibilities (Main tasks and frequency)	<ul style="list-style-type: none"> • Daily monitoring of vital system stats • Daily monitoring of interfaces • Assist internal users in case of any technical issues. • Act as a spoc between my organization and offshore support organization. • Validate issues before routing to support organization
Main Goals	<ul style="list-style-type: none"> • Reduce lead time for reporting support tickets • Reduce lead time for ticket resolution by providing right information to support team
Needs	<ul style="list-style-type: none"> • Simple view of interface monitoring data
Pain Points	<ul style="list-style-type: none"> • No easy tools for monitoring interfaces • Cannot access interface monitoring data when in travel mode, especially between different sites.
Stakeholders	<ul style="list-style-type: none"> • Internal business users • Offshore support associates • Offshore Service Delivery Manager
Competencies	<ul style="list-style-type: none"> • Slightly more than a casual user • Team player • Mostly local focus

User Experience Journey

Current User Experience Journey

Duration of the Journey: 20 - 30 min



Mockup – Landing Page

Filters in header tool bar area. Default filter is current date

IDoc Processing Stats

Direction	Status	Basic Type	Count	
Outbound	▲ Waiting	DEBMAS	20	>
		MATMAS	10	>
		CREMAS	10	>
	■ Success	MATMAS	100	>
		CREMAS	200	>
		DEBMAS	250	>
Inbound	● Error	ORDERS	2	>
	■ Success	ORDERS	100	>

Drill down

Application Type	Monitoring
Screen Pattern	Full Screen
Design Elements Used	<ol style="list-style-type: none"> 1. Icon Tab 2. Table 3. ObjectNumber 4. ObjectStatus 5. VizFrame (Charts) 6. Toolbar

IDoc Details

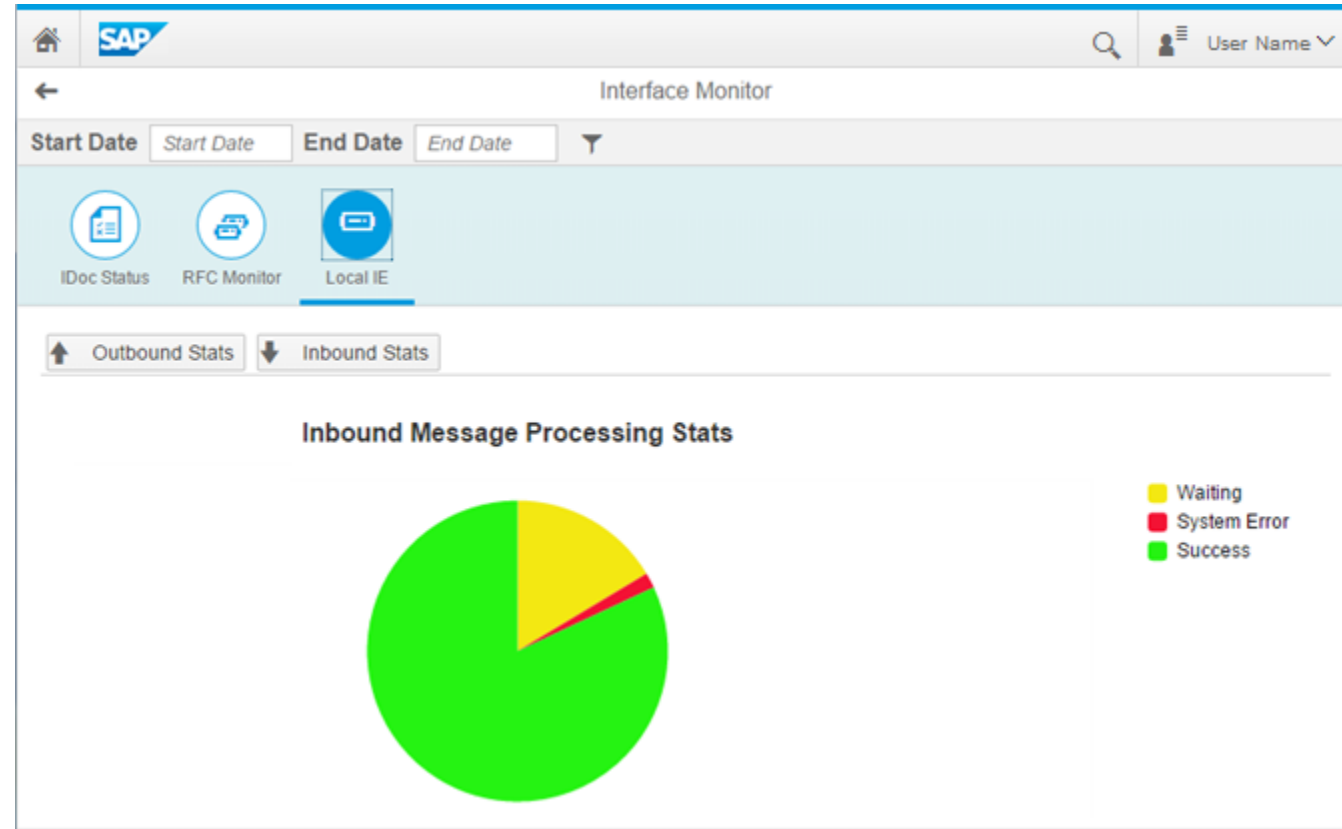
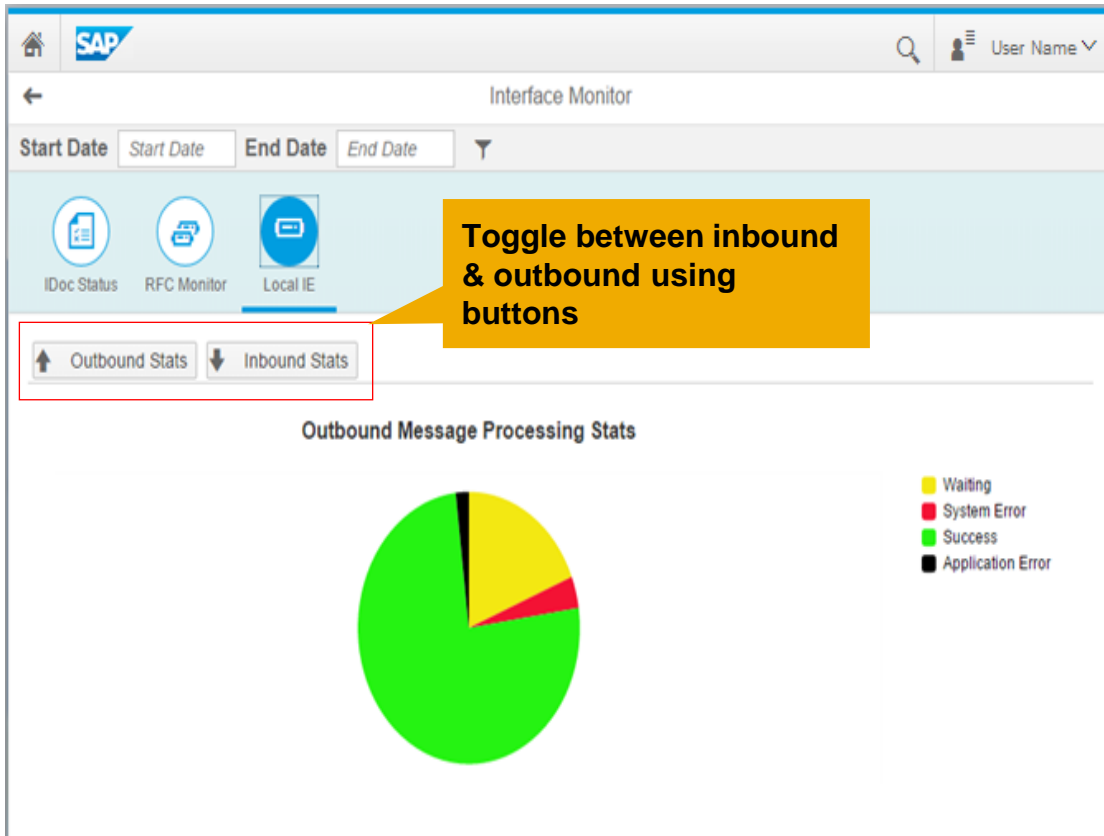
IDoc Number	Basic Type	Sender	Receiver	Type	Error Text
60001	ORDERS	3PL	ZGB090	LS	No Inbound Partner Profile defined for ZGB090
60002	ORDERS	3PL	ZGB090	LS	No Inbound Partner Profile defined for ZGB090
60003	ORDERS	3PL	ZGB090	LS	No Inbound Partner Profile defined for ZGB090
60004	ORDERS	3PL	ZGB090	LS	No Inbound Partner Profile defined for ZGB090
60005	ORDERS	3PL	ZGB090	LS	No Inbound Partner Profile defined for ZGB090
60006	ORDERS	3PL	ZGB090	LS	No Inbound Partner Profile defined for ZGB090

Mockup – Icon Tab- RFC Monitor

The screenshot shows the SAP Interface Monitor application. At the top, there is a navigation bar with the SAP logo, a search icon, and a user profile icon labeled 'User Name'. Below this is a breadcrumb trail 'Interface Monitor'. A filter bar contains 'Start Date' and 'End Date' input fields. A tabbed interface below the filter bar shows three tabs: 'IDoc Status', 'RFC Monitor' (which is selected and highlighted with a blue underline), and 'Local IE'. The main content area displays the title 'RFC Processing Stats - Equivalent TCodes (SMQ1,SMQ2,SM58)' above a table. The table has four columns: 'RFC Type', 'Direction', 'Status', and 'Count'. It lists data for 'qRFC' (Inbound and Outbound) and 'tRFC' (N/A), with various status values like 'READY', 'RUNNING', 'SYSFAIL', 'WAITING', and 'ARETRY'.

RFC Type	Direction	Status	Count
qRFC	Inbound	READY	100
		RUNNING	250
		SYSFAIL	5
		WAITING	1
	Outbound	ARETRY	3
		SYSFAIL	5
	N/A	WAITING	5
		RECORDED	40
		SYSFAIL	100

Mockup – Icon Tab- Local Integration Engine Stats



Demo Application URL : <https://interfacemonitorapp-p260966trial.dispatcher.hanatrial.ondemand.com/>

SAP Web IDE App

The screenshot displays the SAP Web IDE interface. The browser address bar shows the URL: `https://webide-p260966trial.dispatcher.hanatrial.ondemand.com`. The application menu includes File, Edit, Run, Search, View, Tools, and Help. The user is logged in as 'Hi Jithin!' with a 'Logout' option.

The left sidebar shows a project structure for 'Interface_Monitor_App' with folders for 'css', 'i18n', 'Mock_Data', 'model', 'util', and 'view'. The 'util' folder is selected.

The main editor displays the XML code for 'Master.view.xml'. The code includes namespaces for 'sap.viz.ui5.controls' and 'sap.ui.table', and defines a controller 'com.jr.interface.monitor.view.Master'. The XML structure includes a 'Page' with a 'subHeader' containing a 'Toolbar' and an 'IconTabBar'. The 'Toolbar' contains labels for 'Start Date' and 'End Date', and a 'DateTimeInput' for 'End Date'. The 'IconTabBar' contains three 'IconTabFilter' items: 'IDoc Processing Stats', 'RFC Monitor', and 'Local IE'. The 'RFC Monitor' filter contains a 'Table' and two 'Button' elements for 'Outbound Stats' and 'Inbound Stats'. The 'Local IE' filter contains a 'Toolbar' with two 'Button' elements for 'Outbound Stats' and 'Inbound Stats'. The 'Local IE' filter also contains a 'Popover' and two 'ScrollContainer' elements, each containing a 'VizFrame' for a chart.

Yellow callout boxes highlight specific design components in the code:

- Design Component #1 - Toolbar**: Points to the `<Toolbar>` element and its child elements.
- Design Component #2 - IconTabBar**: Points to the `<IconTabBar id="idIconTabBarMulti" expanded="{device}>isNoPhone}" class="sapUiResponsiveContentPadding">` element.
- Design Component #3 - Table**: Points to the `<Table>` element within the 'RFC Monitor' filter.
- Design Component #4 - Button**: Points to the `<Button text="Outbound Stats" icon="sap-icon://arrow-top" press="onViewOutboundPressed"></Button>` element.
- Design Component #5 - Chart & Scrollcontainer**: Points to the `<ScrollContainer id="idViz_Out_Scroll" height="100%" width="100%" horizontal="false" vertical="true" focusable="true">` element and its child `<viz:VizFrame id="idVizFramePie_Out" uiConfig="{applicationSet:'fiori'}" vizType="pie" width="100%" height="300px"></viz:VizFrame>` element.
- Design Component #6 - Details View**: Points to the 'view' folder in the project structure.