

Waterfall Deliverables – Part -2

SMS TRAFFIC MANAGEMENT SYSTEM



February 14, 2025

COPED

**Document 6- Please prepare a use case diagram, activity diagram and a use case specification document.**

**Use Case 1: SMS Traffic Dashboard**

|  |  |
| --- | --- |
| **Use case Name** | **SMS Traffic Dashboard** |
| **Use case Description** | The SMS Traffic Dashboard use case provides a centralized interface for system administrators to monitor and analyse SMS traffic in real-time. The dashboard displays key performance indicators (KPIs) such as message volume, delivery rates, and error rates, allowing administrators to quickly identify issues and take corrective action. |
| **Actors Primary Actors Secondary actors** | Primary Actors: System Administrator Secondary Actors: None |
| **Basic Flow** | 1. The system administrator logs into the system and navigates to the SMS Traffic Dashboard. 2. The system displays the current SMS traffic data, including message volume, delivery rates, and error rates. 3. The system administrator views the data and identifies any issues or trends. 4. The system administrator takes corrective action as needed, such as adjusting system settings or contacting support. |
| **Alternative Flows** | 1. The system administrator clicks on a specific KPI to view more detailed data. 2. The system displays a detailed report or chart for the selected KPI. 3. The system administrator analyses the detailed data and takes corrective action as needed. |
| **Exceptional flows** | 1. Error: The system encounters an error while retrieving data, such as a database connection issue.  - The system displays an error message to the system administrator.  - The system administrator contacts support to resolve the issue. 2. No Data: The system has no data to display, such as when the system is first installed.  - The system displays a message indicating that there is no data available.  - The system administrator waits for data to be collected and then views the dashboard. |
| **Pre- Conditions** | The system administrator has a valid login account. The system has been configured to collect and display SMS traffic data. |
| **post-conditions** | The system administrator has viewed the SMS traffic data and taken corrective action as needed. The system has updated its data and is ready for the next user to view. |
| **Assumptions** | The system administrator has a basic understanding of SMS traffic and system administration. The system has been properly configured and is functioning correctly. |
| **Constraints** | The system must be able to handle a large volume of SMS traffic data. The system must be able to display data in real-time. The system must be secure and protect user data. |
| **Dependencies** | The system relies on the SMS traffic data collection system to provide accurate and up-to-date data. The system relies on the system administrator to configure and maintain the system. |
| **Inputs and Outputs** | Inputs: SMS traffic data, system administrator login credentials. Outputs: SMS traffic data display, error messages, detailed reports and charts. |
| **Business Rules** | The system must display data in a clear and concise manner. The system must provide real-time data to ensure timely decision-making. The system must be secure and protect user data. |
| **Miscellaneous Information** | The SMS Traffic Dashboard use case is a key component of the SMS Traffic Management System. The system administrator will use the dashboard to monitor and analyse SMS traffic on a regular basis. The dashboard will be customizable to meet the specific needs of the system administrator. |

**Use Case 2: View Dashboard**

|  |  |
| --- | --- |
| **Use case Name** | **View Dashboard** |
| **Use case Description** | The system administrator views the real-time SMS traffic dashboard to monitor and analyse SMS traffic data. |
| **Actors Primary Actors Secondary actors** | System Administrator |
| **Basic Flow** | 1. The system administrator logs into the system and navigates to the dashboard.  2. The system displays the real-time SMS traffic data.  3. The system administrator views the data and analyses it. |
| **ALTERNATE FLOW** | 1. The system administrator clicks on a specific data point to view more detailed information.  2. The system displays a detailed report or chart for the selected data point. |
| **Exceptional flows** | 1. Error: The system encounters an error while retrieving data.  - The system displays an error message to the system administrator.  2. No Data: The system has no data to display.  - The system displays a message indicating that there is no data available. |
| **Pre- Conditions** | The system administrator has a valid login account, and the system has been configured to collect and display SMS traffic data. |
| **post-conditions** | The system administrator has viewed the real-time SMS traffic data and analysed it. |
| **Assumptions** | The system administrator has a basic understanding of SMS traffic and system administration. |
| **Constraints** | The system administrator has a basic understanding of SMS traffic and system administration. |
| **Dependencies** | The system relies on the SMS traffic data collection system to provide accurate and up-to-date data. |
| **Inputs and Outputs** | Inputs: System administrator login credentials, SMS traffic data Outputs: Real-time SMS traffic data display |
| **Business Rules** | The system must display data in a clear and concise manner and provide real-time updates. |
| **Miscellaneous Information** | The view dashboard use case is a key component of the real-time SMS traffic dashboard. |

**Use Case 3: Configure Dashboard**

|  |  |
| --- | --- |
| **Use case Name** | **Configure Dashboard** |
| **Use case Description** | The system administrator configures the real-time SMS traffic dashboard to display the desired data and layout. |
| **Actors Primary Actors Secondary actors** | - Primary Actors: System Administrator  - Secondary Actors: None |
| **Basic Flow** | 1. The system administrator logs into the system and navigates to the dashboard configuration page. |
| 2. The system administrator selects the data to display on the dashboard, such as message volume, delivery rates, and error rates. |
| 3. The system administrator configures the layout of the dashboard, including the placement of charts, tables, and other visualizations. |
| **ALTERNATE FLOW** | 1. The system administrator uses the system's built-in configuration tools to customize the dashboard. |
| 2. The system administrator imports a pre-built configuration template. |
| **Exceptional flows** | 1. Error: The system encounters an error while saving the configuration changes. |
| - The system displays an error message to the system administrator. |
| 2. Invalid Configuration: The system administrator enters an invalid configuration. |
| - The system displays an error message to the system administrator and prompts them to re-enter the configuration. |
| **Pre- Conditions** | The system administrator has a valid login account and the system has been configured to allow dashboard configuration. |
| **post-conditions** | The system administrator has configured the dashboard to display the desired data and layout. |
| **Assumptions** | The system administrator has a basic understanding of dashboard configuration and system administration. |
| **Constraints** | The system must be able to handle a large volume of configuration changes and display the configured dashboard in real-time. |
| **Dependencies** | The system relies on the dashboard configuration system to provide accurate and up-to-date configuration options. - Inputs and Outputs:  - Inputs: System administrator login credentials, configuration changes  - Outputs: Configured dashboard display |
| **Inputs and Outputs** | Inputs: System administrator login credentials, SMS traffic data |
| Outputs: Real-time SMS traffic data display |
| **Business Rules** | The system must provide a user-friendly and intuitive configuration interface and validate all configuration changes. |
| **Miscellaneous Information** | The configure dashboard use case is a key component of the real-time SMS traffic dashboard. |

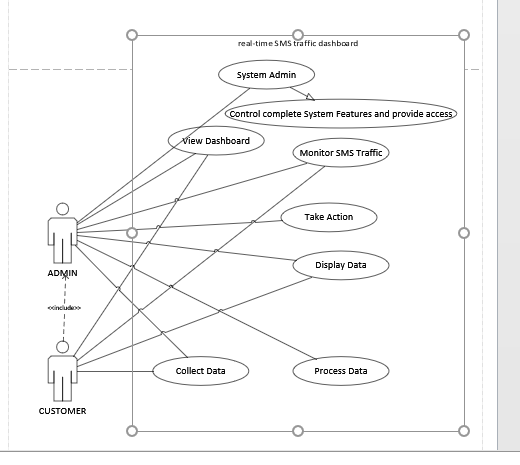
**Use Case 4: Generate Reports**

|  |  |
| --- | --- |
| **Use case Name** | **Generate Reports** |
| **Use case Description** | The system administrator generates reports on the SMS traffic data to analyse trends, issues, and areas for improvement. |
| **Actors Primary Actors Secondary actors** | - Primary Actors: System Administrator  - Secondary Actors: None |
| **Basic Flow** | 1. The system administrator logs into the system and navigates to the report generation page.  2. The system administrator selects the report type, such as a summary report or a detailed report.  3. The system administrator selects the data to include in the report, such as message volume, delivery rates, and error rates.  4. The system generates the report and displays it to the system administrator. |
|  |
|  |
| **ALTERNATE FLOW** | 1. The system administrator uses the system's built-in report generation tools to customize the report.  2. The system administrator schedules the report to be generated automatically at a specified interval. |  |
|  |
| **Exceptional flows** | 1. Error: The system encounters an error while generating the report.  - The system displays an error message to the system administrator.  2. No Data: The system has no data to include in the report.  - The system displays a message indicating that there is no data available. |  |
|  |
|  |
|  |
| **Pre- Conditions** | The system administrator has a valid login account and the system has been configured to generate reports. |  |
| **post-conditions** | The system administrator has generated a report on the SMS traffic data. |  |
| **Assumptions** | The system administrator has a basic understanding of report generation and system administration. |  |
| **Constraints** | The system must be able to handle a large volume of report generation requests and display the reports in real-time. |  |
| **Dependencies** | The system relies on the report generation system to provide accurate and up-to-date report options. |  |
| **Inputs and Outputs** | - Inputs: System administrator login credentials, report type, data selection  - Outputs: Generated report |  |
|  |
| **Business Rules** | The system must provide a user-friendly and intuitive report generation interface and validate all report generation requests. |  |
| **Miscellaneous Information** | The generate reports use case is a key component of the real-time SMS traffic dashboard. |  |

**Use Case 5: Configure Real-time SMS Traffic Dashboard**

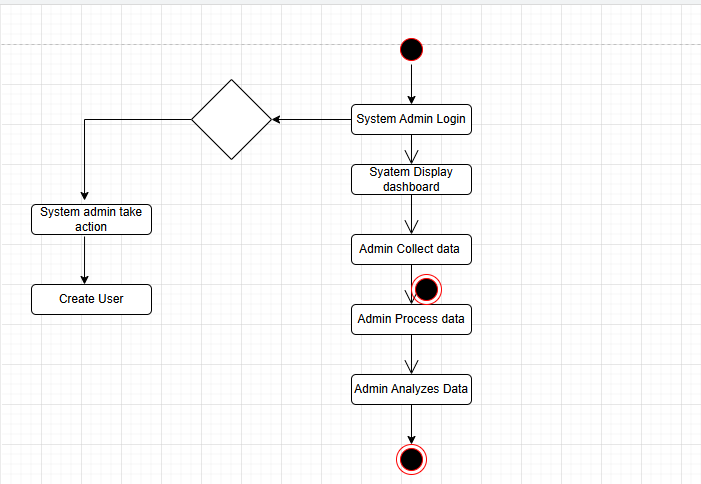
|  |  |
| --- | --- |
| **Use case Name** | **Configure Real-time SMS Traffic Dashboard** |
| **Use case Description** | The system administrator configures the real-time SMS traffic dashboard to display the desired data and metrics. |
| **Actors Primary Actors Secondary actors** | - Primary Actors: System Administrator  - Secondary Actors: None |
| **Basic Flow** | 1. The system administrator logs into the system and navigates to the real-time SMS traffic dashboard configuration page.  2. The system administrator selects the data and metrics to display on the dashboard.  3. The system administrator configures the dashboard layout and design.  4. The system administrator saves the configuration changes. |
|  |
|  |
| **ALTERNATE FLOW** | 1. The system administrator uses the system's built-in configuration tools to configure the dashboard.  2. The system displays a preview of the configured dashboard. |  |
|  |
| **Exceptional flows** | 1. Error: The system encounters an error while saving the configuration changes.  - The system displays an error message to the system administrator.  2. Invalid Configuration: The system administrator enters an invalid configuration.  - The system displays an error message to the system administrator and prompts them to re-enter the configuration. |  |
|  |
|  |
|  |
| **Pre- Conditions** | The system administrator has a valid login account and the system has been configured to display real-time SMS traffic data. |  |
| **post-conditions** | The system administrator has configured the real-time SMS traffic dashboard to display the desired data and metrics. |  |
| **Assumptions** | The system administrator has a basic understanding of system administration and configuration. |  |
| **Constraints** | The system must be able to handle a large volume of SMS traffic data and display it in real-time. |  |
| **Dependencies** | The system relies on the SMS traffic data collection system to provide accurate and up-to-date data. |  |
| **Inputs and Outputs** | - Inputs: System administrator login credentials, configuration changes  - Outputs: Configured real-time SMS traffic dashboard |  |
|  |
| **Business Rules** | The system must provide a user-friendly configuration interface and validate user input. |  |
| **Miscellaneous Information** | The configure real-time SMS traffic dashboard use case is a key component of the real-time SMS traffic dashboard. |  |

**USE CASE DIAGRAM**



**ACTIVITY DIAGRAM**

**Admin:**



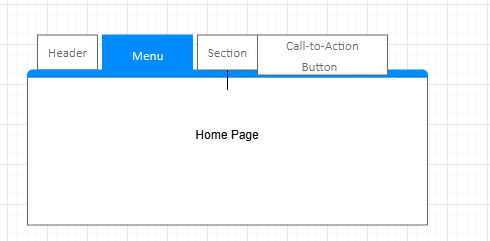
**Customer:**

A diagram of a system

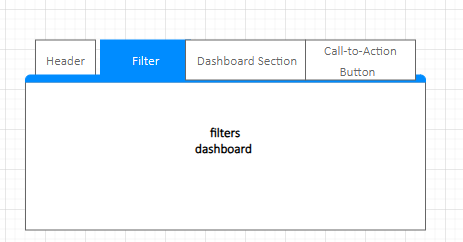
AI-generated content may be incorrect.

**Document 7- Screens and pages**

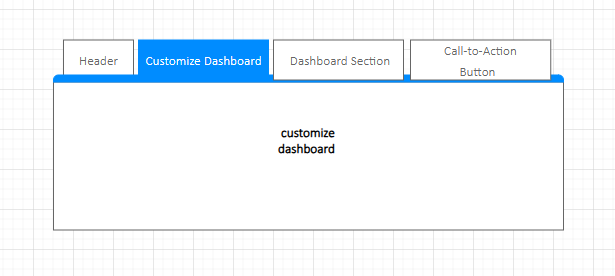
**home page can be designed as follows:**



**The filters dashboard can be designed as follows:**



**The customize dashboard can be designed as follows:**



**The alert notifications page can be designed as follows:**

A screenshot of a computer screen

AI-generated content may be incorrect. **Document 8- Tools-Visio and Axure**

Throughout the project, I had the opportunity to utilize both Visio and Axure to create comprehensive mock-ups and prototypes of the Real-time SMS Traffic Dashboard. My experience with Visio was instrumental in creating detailed and precise wireframes, which helped to visualize the layout and functionality of the dashboard. I was able to leverage Visio's extensive library of shapes and templates to create a professional-looking design, and its collaboration features allowed me to easily share and review my work with the team. On the other hand, Axure proved to be a powerful tool for creating interactive prototypes, enabling me to bring the dashboard to life and test its usability and functionality. Axure's drag-and-drop interface and robust feature set made it easy to create complex interactions and animations, and its ability to generate HTML and CSS code allowed me to quickly transition from prototype to production-ready design. Overall, my experience with both Visio and Axure was highly positive, and I was able to effectively utilize these tools to create a well-designed and functional Real-time SMS Traffic Dashboard that met the project's requirements.

**Document 9- BA experience**

**Requirement Gathering:**

To gather requirements, we employed the MOSCOW technique, which involves prioritizing requirements as Must-Haves, Should-Haves, Could-Haves, and Won't-Haves. This technique helped us to focus on the most critical requirements and allocate resources accordingly.

However, the client was not available for some period during this phase, which posed a significant challenge. To overcome this, I had to source out point of contacts from the client's side and get the information as soon as possible. This required effective communication and project management skills to ensure that the requirements were gathered accurately and efficiently.

**Validation of Requirements:**

To validate the requirements, I used the FURPS technique, which stands for Functionality, Usability, Reliability, Performance, and Supportability. This technique helped me to ensure that the requirements were complete, consistent, and met the stakeholder's needs.

**Removal of Duplicated Requirements:**

During the requirement gathering phase, I noticed that there were many requirements that were duplicated or repeated. To avoid confusion and ensure that the requirements were accurate, I removed these duplicated requirements immediately. This helped to streamline the requirement gathering process and ensure that the project stayed on track.

**Prototyping:**

To give more specific requirements, we used prototyping techniques. Prototyping helped to visualize the requirements and identify any gaps or inconsistencies. It also enabled us to validate the requirements with the stakeholders and ensure that everyone was on the same page.

Overall, my experience as a BA in the requirement gathering phase was challenging but rewarding. I had to use my skills and techniques to gather, validate, and prioritize the requirements, and ensure that the project stayed on track. The use of MOSCOW, FURPS, and prototyping techniques helped to ensure that the requirements were accurate, complete, and met the stakeholder's needs.

Here are some of the key takeaways from my experience:

- Effective communication and project management skills are essential for successful requirement gathering.

- The use of techniques such as MOSCOW, FURPS, and prototyping can help to ensure that the requirements are accurate, complete, and met the stakeholder's needs.

- Removing duplicated requirements is crucial to avoid confusion and ensure that the project stays on track.

- Prototyping can help to visualize the requirements and identify any gaps or inconsistencies.

By following these best practices, BAs can ensure that the requirement gathering phase is successful, and the project is set up for success.

**Requirement Analysis:**

In this phase, I was responsible for drawing UML (Unified Modeling Language) diagrams to visually describe the requirements. UML diagrams, such as use case diagrams, class diagrams, and sequence diagrams, helped to provide a clear and concise representation of the system's functionality and behavior.

In addition to UML diagrams, I also used activity diagrams to describe the process flow and identify the key steps involved in the system's operations. Activity diagrams helped to illustrate the workflow and identify any potential bottlenecks or areas for improvement.

Communication and Collaboration:

Once the diagrams were created, I communicated them to the team, including stakeholders, developers, and testers. However, some team members might not have agreed with the diagrams, and changes were suggested. As a BA, it was my responsibility to consider these points, discuss them with the team, and make modifications as necessary.

This collaborative approach ensured that the diagrams accurately reflected the system's requirements and that everyone was on the same page. It also helped to identify and resolve any potential issues or conflicts early on, which saved time and effort in the long run.

**Preparation of BRS and SRS:**

As part of the requirement analysis phase, I was also responsible for preparing the Business Requirement Specification (BRS) and System Requirement Specification (SRS) documents.

The BRS document outlined the business needs and requirements of the system, including the functional and non-functional requirements. It provided a clear understanding of what the system needed to do and how it would meet the business objectives.

The SRS document, on the other hand, provided a detailed description of the system's technical requirements, including the hardware, software, and infrastructure needed to support the system. It outlined the system's architecture, components, and interfaces, as well as the data models and database design.

By preparing these documents, I ensured that the requirements were thoroughly documented and that everyone involved in the project had a clear understanding of what was expected.

**Key takeaways from my experience in this phase include:**

- UML diagrams and activity diagrams are essential tools for visually describing requirements and process flows.

- Collaboration and communication with the team are critical to ensuring that the diagrams accurately reflect the system's requirements.

- Considering and incorporating feedback from team members is essential for creating accurate and effective diagrams.

- Preparing BRS and SRS documents is crucial for ensuring that the requirements are thoroughly documented and that everyone involved in the project has a clear understanding of what is expected.

Overall, my experience in the requirement analysis phase was instrumental in ensuring that the requirements were thoroughly examined, documented, and communicated to the team. It laid the foundation for the successful development and implementation of the Real-time SMS Traffic Dashboard system.

**Design Phase:**

In the design phase, I worked closely with the development team to ensure that the solution met the requirements gathered during the previous phase. My key responsibilities included:

**- Preparing Test Cases:** From the use case diagrams, I prepared test cases to ensure that the solution was thoroughly tested. I made sure to cover all the possible scenarios, including positive and negative test cases.

**- Communicating with Client:** I communicated with the client on the design and solution documents to ensure that they were aware of the progress and that their requirements were being met.

**- Writing Negative Test Cases:** In addition to positive test cases, I also wrote negative test cases to ensure that the solution handled errors and exceptions correctly.

**- Comprehensive Test Cases:** I made sure to not miss a single test case, as this could have a huge impact on the project development in later stages. I reviewed and refined the test cases to ensure that they were comprehensive and covered all the requirements.

**- Preparing Test Data:** I prepared test data for testing to ensure that the solution was tested with realistic data.

**- Updating RTM (Requirements Traceability Matrix):** I updated the RTM to ensure that all the requirements were met. This involved mapping the requirements to the test cases and ensuring that all the requirements were covered.

The design phase was critical in ensuring that the solution was developed to meet the stakeholder's requirements. By preparing comprehensive test cases, communicating with the client, and updating the RTM, I ensured that the solution was thoroughly tested and met all the requirements.

**Development Phase:**

In the development phase, I played a key role in facilitating the development process and ensuring that the solution was developed to meet the requirements. My key responsibilities included:

**1. Organizing JAD (Joint Application Development) Sessions:** I organized JAD sessions with the technical team to ensure that everyone was on the same page and that the solution was developed to meet the requirements.

**2. Clarifying Queries:** I clarified queries of the technical team during coding to ensure that they had a clear understanding of the requirements and the solution.

**3. Handling Conflicts:** There were some team members who didn't agree with the concept or didn't cooperate during JAD sessions. As a BA, I handled the situation gently and had one-on-one discussions with them to understand their concerns and address them. I explained how their actions were going to affect the project and worked with them to find a solution that met everyone's needs.

**4. Setting up a Healthy Environment:** I set up a healthy environment within the team by promoting open communication, collaboration, and respect for each other's opinions. This helped to build trust and ensured that the team worked together effectively to deliver the project.

**5. Referring Diagrams:** I referred to the diagrams created during the design phase to ensure that the unit was coded correctly.

**6. Conducting Regular Meetings:** I conducted regular meetings with the technical team and client to ensure that everyone was on track and that the project was progressing as planned. This was challenging at times, as some team members might not be available for the meeting. To address this, I recorded the session and provided it to the missed team member and then had a one-on-one discussion with them later to ensure they were caught up.

The development phase was critical in ensuring that the solution was developed to meet the stakeholder's requirements. By organizing JAD sessions, clarifying queries, handling conflicts, setting up a healthy environment, referring to diagrams, and conducting regular meetings, I ensured that the project was delivered on time and to the required quality.

**Testing Phase:**

In the testing phase, I worked closely with the testing team to ensure that the solution was thoroughly tested and met the requirements gathered during the previous phases. My key responsibilities included:

**1. Preparing Test Cases:** I prepared test cases from the use cases to ensure that the solution was tested against the requirements. I made sure to cover all the possible scenarios, including positive and negative test cases.

**2. Performing High-Level Testing:** I performed high-level testing to ensure that the solution met the overall requirements and functionality.

**3. Requesting Test Data:** I requested test data from the client to ensure that the solution was tested with realistic data. This helped to identify any issues or defects that may not have been caught with dummy data.

**4. Updating RTM (Requirements Traceability Matrix):** I updated the RTM to ensure that all the requirements were met and that the solution was developed to meet the stakeholder's needs.

**5. Taking Signoff from Client:** I took signoff from the client to ensure that they were satisfied with the testing and that the solution met their requirements.

**6. Preparing Client for UAT (User Acceptance Testing):** I prepared the client for UAT by providing them with the necessary information and support to ensure that they could test the solution effectively.

The testing phase was critical in ensuring that the solution was thoroughly tested and met the stakeholder's requirements. By preparing comprehensive test cases, performing high-level testing, and updating the RTM, I ensured that the solution was tested against the requirements and that any defects or issues were identified and addressed.

**Deployment Phase:**

In the deployment phase, I worked closely with the project team to ensure that the solution was successfully deployed and that the end-users were properly trained and supported. My key responsibilities included:

**1. Forwarding RTM to Client:** I forwarded the updated RTM (Requirements Traceability Matrix) to the client, which was attached to the project closure document. This ensured that the client had a clear understanding of how the solution met their requirements.

**2. Coordinating End-User Manuals:** I coordinated with the development team to complete and share end-user manuals with the client. These manuals provided detailed instructions on how to use the solution and troubled shooting tips.

**3. Planning and Organizing Training Sessions:** I planned and organized training sessions for the end-users to ensure that they were properly trained on the solution. This included scheduling the training sessions, preparing the training materials, and ensuring that all the necessary equipment was available.

**4. Ensuring Attendance:** I made sure that all the candidates attended the training sessions by sending out reminders and follow-up emails. I also worked with the client to ensure that all the necessary stakeholders were aware of the training sessions and were able to attend.

The deployment phase was critical in ensuring that the solution was successfully deployed and that the end-users were properly trained and supported. By forwarding the RTM to the client, coordinating end-user manuals, planning and organizing training sessions, and ensuring attendance, I ensured that the client was able to use the solution effectively and efficiently.