Software Requirement Specification Document

Of

"Atmiya Alumni Association" Project

Author: Brijraj R. Kacha

E-Mail: brkacha553@gmail.com

Table of content:

Purpose:

The main objective of this document is to outline and define the interaction between the application and the end user. It encompasses both functional and non-functional requirements, ensuring a complete and thorough understanding of what the application must achieve to be considered complete. This document plays a vital role in establishing effective communication between different stakeholders, including users and developers, fostering a shared comprehension of the scope and goals of the A^3 - Atmiya Alumni Association software. By providing a comprehensive and detailed overview, it serves as a crucial guide to ensure that all necessary functionalities and objectives are met.

Scope:

This Software Requirements Specification (SRS) is a comprehensive document encompassing various potential use cases that users may encounter, providing an overview of the project's scope and intended applications. Additionally, it offers valuable insights into the project's User Interface (UI) sketches, backend data flows, and other crucial software processes. The primary focus of this document is to provide detailed descriptions of the anticipated use cases, ensuring a thorough understanding of the software's functionalities.

Moreover, the SRS includes well-defined information on functional requirements, non-functional requirements, design blueprints, maintenance guidelines, validation procedures, testing protocols, version controls, and other relevant aspects. This extensive coverage ensures

that both technical and non-technical individuals can gain a precise understanding of the software's features and capabilities.

By offering clarity on all essential aspects, this SRS serves as a valuable resource for anyone involved in the project, facilitating seamless collaboration and ensuring a shared vision of the software's goals and objectives. It acts as a comprehensive guide, guiding the development team, stakeholders, and users in aligning their efforts and expectations, thereby contributing to the successful implementation and delivery of the software.

Overview:

In the year 2018, the renowned educational institute in Rajkot underwent a significant transformation and became a unitary University - Atmiya University. Over the span of more than 50 years, this esteemed institution has been imparting education to students from various fields and diverse branches of study. As a result, a vast community of alumni has emerged, forming an integral part of the university's legacy.

In order to establish a strong connection with each and every alumnus, the university initiated the circulation of a Google Form link. This form was intended for all the alumni to fill out and submit their updated details. However, as responses started pouring in, they came in an overwhelming random manner, reaching an impressive count of thousands.

Unfortunately, the task of bifurcating and sorting these numerous entries according to various branches has proven to be an extremely time-consuming endeavour for a specific faculty. To address this formidable challenge and streamline the process of data bifurcation and sorting, a cutting-edge software solution is being developed. This innovative software aims to achieve all the necessary bifurcation and sorting while providing comprehensive data analysis, all with a single click of a button. The implementation of this software is anticipated to save countless hours for all the departments of Atmiya University, revolutionizing the way data is handled and enhancing the efficiency of the entire system.

Product Perspective:

Concept of Operations:

This application is specifically designed for a particular purpose, making it a specialized tool tailored to the needs of its users. Its primary function revolves around helping the relevant faculty or responsible individuals to efficiently sort and bifurcate the responses collected from a circulated Google Form link. Beyond this, the application offers valuable features, including comprehensive data analysis with an array of visually informative graphs and charts.

One of the application's noteworthy capabilities is enabling users to effortlessly export the output analysis and segmented data. Additionally, users can conveniently attach the analytical graphs, charts, and bifurcated data to various platforms, such as G-mail, WhatsApp, and PowerPoint presentations, ensuring seamless sharing of valuable insights.

This application prides itself on being incredibly user-friendly, ensuring a hassle-free experience. Users merely need to select the corresponding excel file containing alumni responses from their computer system and then initiate the process with a simple click of the run button. Incredibly swift, the application's program can bifurcate thousands of data entries within a mere 10 to 15 seconds.

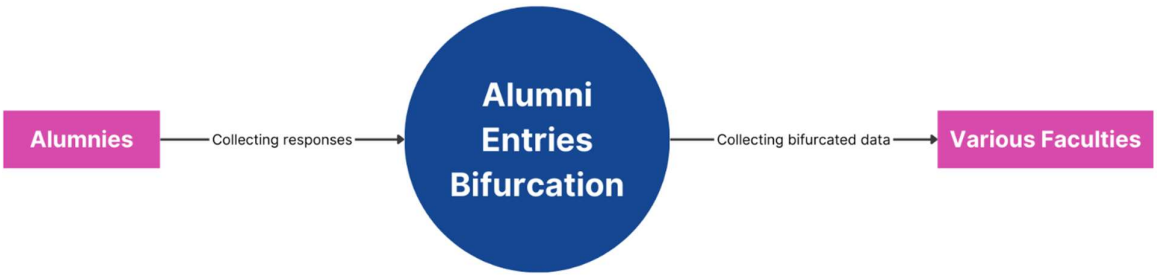
To facilitate organization and accessibility, the application generates a comprehensive distribution of data, categorized by faculty. For example, within a list of 2000 alumni, the application will create separate files for each faculty, effectively storing data for each group. Furthermore, within each faculty, the data will be further distributed by branch, ensuring a detailed breakdown.

Not stopping there, the application goes a step further by providing a year-wise distribution of alumni data, allowing for deeper insights into trends and patterns. An overall count of data across all branches, faculties, and years is also automatically generated for a holistic perspective.

The application boasts a valuable feature to eliminate duplicate entries. If any alumni attempt to fill the circulated Google Form multiple times with identical data, the application will efficiently disregard previous entries, considering only the most recent submission. This ensures accuracy and reliability in the data analysis process.

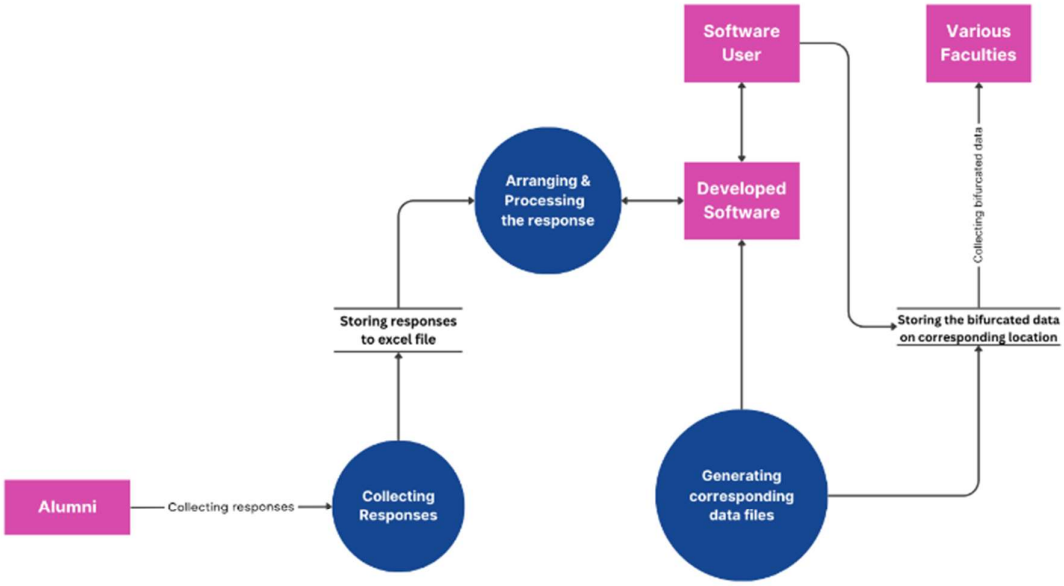
Data Flow Diagrams:

Level 0 DFD:



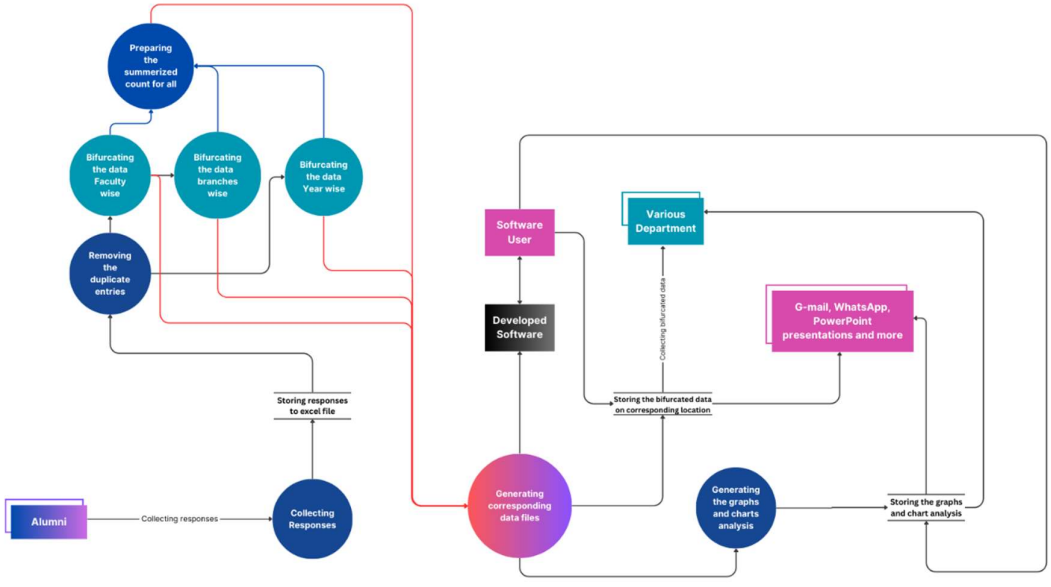
Data Flow Diagram - Level 0
(PROJECT NAME : ATMIYA ALUMNI ASSOCIATION)

Level 1 DFD:



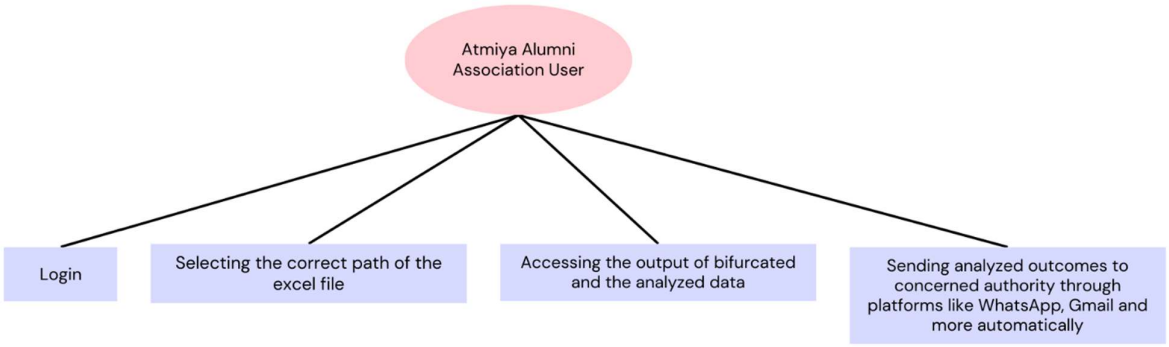
Data Flow Diagram - Level 1
(PROJECT NAME : ATMIYA ALUMNI ASSOCIATION)

Level 2 DFD:



Data Flow Diagram - Level 2
(PROJECT NAME : ATMIYA ALUMNI ASSOCIATION)

Product Functions:



Use Cases:

Normal case:

Upon the user's selection of an Excel file path that contains comprehensive alumni data, the program will diligently verify the designated path to ensure its accuracy. Once confirmed, the program will proceed to extract and retrieve the data from the chosen file, and subsequently initiate the process of methodically segregating each data entry based on specific categories. Concurrently, it will actively identify and remove any redundant or duplicate entries encountered within the dataset.

Following this meticulous categorization and data refinement, the program will diligently generate separate and distinct bifurcated files, each organized according to the predefined categories. These bifurcated files will be stored in the very location where the primary file, containing the unsegregated data, was originally located. As a result, the user will have easy access to both the non-bifurcated data and the newly created bifurcated files.

Beyond the fundamental data management aspects, the program will also furnish the user with comprehensive data analysis, presenting the information in a visually appealing manner through a diverse array of charts. These charts will effectively illustrate key insights and trends derived from the alumni data, providing the user with valuable information for informed decision-making and analysis.

Wrong data file selection:

Every time the user chooses a path for an Excel file, the program will attentively check if the selected file contains alumni data in the required format. If the file does not meet these criteria or if the user selects a different type of file altogether, the program will not continue with its processing. Instead of generating a malfunction or proceeding with erroneous data, the program will respond with an error message that politely prompts the user to select the appropriate file. The error message will be clear and concise, conveying the message "Please select the appropriate file," allowing the user to rectify the situation

and proceed with the correct file selection for successful data processing.

User Characteristics:

This application has been meticulously crafted with a distinct and specific purpose in mind. Its intended audience comprises individuals who hold key positions as faculty or authorities within Atmiya University. These privileged users are equipped with the responsibility of managing and overseeing the analysis of the university's alumni data. Hence, this application caters to their unique needs and requirements, facilitating seamless data handling and enabling comprehensive analysis to derive valuable insights about the alumni community of the university.

Functional Requirements:

1. Gathering Responses: The application begins by efficiently collecting responses from all the alumni of Atmiya University, utilizing a user-friendly Google Form to ensure ease and convenience for the respondents.
2. Categorization by Faculty: After the data collection phase, the application skilfully bifurcates the received responses, effectively categorizing them based on the respective faculties to which the alumni belong. This categorization helps in organizing and analysing the data according to the different academic divisions within the university.
3. Categorization by Department: Furthermore, the application performs another level of bifurcation by organizing the responses according to the specific departments within each faculty. This breakdown provides valuable insights into the individual areas of study and research pursued by the alumni.
4. Categorization by Graduation Year: Additionally, the application ensures a comprehensive analysis by classifying the responses based on the year of graduation. This categorization aids in understanding trends and changes over time within the alumni community.
5. Count Sheet Preparation: To present a concise overview, the application generates a detailed count sheet that summarizes the number of entries at various levels, including departmental, faculty-wise, and across different graduation years. This count sheet provides a comprehensive snapshot of the alumni data and their distribution across various parameters.
6. Data Storage: For enhanced organization and accessibility, the application creates dedicated storage locations on the system for each bifurcated dataset. This segregation ensures easy retrieval and streamlined management of data subsets.
7. Excel File Generation: To facilitate further data exploration and sharing, the application automatically generates Excel files for each bifurcated dataset. Each file is meticulously saved in a corresponding and unique location, thereby maintaining data integrity and facilitating seamless navigation.

8. Plot Generation for Data Analysis: To enhance the understanding of the data, the application employs data visualization techniques by generating plots and charts based on the bifurcated datasets. These visual representations offer a comprehensive and insightful analysis of the alumni data, allowing users to make informed decisions and identify significant patterns and trends.

Through these elaborate steps, the application aims to provide the designated faculty and authorities with a powerful tool to efficiently manage, analyse, and gain valuable insights from the extensive alumni data of Atmiya University, fostering better engagement, decision-making, and future planning for the institution.

Non-functional requirements:

In the context of the application's development, several essential non-functional requirements have been identified to ensure its overall effectiveness and performance. These non-functional requirements encompass a range of critical aspects that directly impact the application's functionality, usability, and overall user experience.

1. System Security:

A top priority is the robust implementation of stringent security measures to safeguard the application and its valuable data from potential threats and unauthorized access. The application must employ state-of-the-art encryption techniques, secure authentication protocols, and access controls to protect sensitive alumni data and maintain the integrity of the system.

2. Data Privacy:

Given the sensitive nature of the alumni data being handled, strict adherence to data privacy regulations and policies is paramount. The application must incorporate robust data privacy measures to ensure that only authorized users have access to specific data and that all data is handled in compliance with applicable data protection laws.

3. Resource Requirements:

The application should be designed to optimize resource utilization, minimizing hardware and software dependencies while ensuring efficient usage of system resources. It should be scalable to accommodate an increasing volume of data and user interactions without compromising performance.

4. Maintainability:

To facilitate seamless updates, bug fixes, and enhancements, the application must be designed with a focus on maintainability. This involves clear and well-documented code, modular design principles, and a systematic approach to testing and debugging.

5. Portability:

The application should be developed with cross-platform compatibility in mind, allowing it to function smoothly across various operating systems and devices. A portable design ensures that users can access the application from different devices without any loss of functionality.

6. Reliability:

The application must exhibit a high degree of reliability and robustness. It should be resilient to errors, crashes, or data loss, and be capable of recovering gracefully from unexpected failures. Ensuring a reliable application is crucial to building trust among users and promoting long-term usage.

7. Efficiency:

Optimizing the application's performance and response time is crucial to providing a smooth and satisfying user experience. The application should be designed to handle large datasets efficiently, perform complex operations quickly, and ensure minimal lag or delays during data processing and visualization.

In conclusion, incorporating these non-functional requirements into the development process will help create a secure, reliable, and efficient application that upholds data privacy, ensures system stability, and offers a seamless user experience. By addressing these key aspects, the application can achieve a high level of performance, usability, and user satisfaction, ultimately contributing to the success of managing and analysing the alumni data of Atmiya University.

Performance Requirements:

Performance requirements for the Atmiya University Alumni Association project are essential to ensure that the application functions efficiently, delivers a seamless user experience, and handles data processing effectively. Here are some performance requirements that could be considered for the project:

1. Response Time:

The application should respond to user interactions promptly, with minimal delay. The maximum acceptable response time for common operations, such as data retrieval, chart generation, or file generation, should be specified to provide a smooth user experience.

2. Scalability:

As the number of alumni and data grows over time, the application should scale gracefully to handle the increased workload. It should be able to accommodate a growing user base and a larger volume of data without significant performance degradation.

3. Data Processing Speed:

The time required for data processing, including data categorization, bifurcation, and generation of plots, should meet specific performance benchmarks. The application should efficiently manage large datasets and complex operations to avoid any slowdowns.

4. Resource Utilization:

The application should be optimized to make efficient use of system resources, such as CPU, memory, and more. It should avoid excessive resource consumption, ensuring that other system processes and applications can run smoothly alongside it.

5. Data Retrieval Efficiency:

The time taken to retrieve data from the storage locations should be minimized to reduce waiting times for users. The application should utilize optimized data retrieval techniques and caching mechanisms to expedite data access.

6. Error Handling and Recovery:

In case of errors or exceptions, the application should handle them gracefully and provide clear error messages to users. It should be capable of recovering from errors and restoring its functionality without causing data loss or corruption.

7. Testing and Benchmarking:

Performance testing and benchmarking should be conducted to assess the application's performance under various conditions. This includes simulating different user loads, data volumes, and stress scenarios to identify potential bottlenecks and areas for improvement.

By incorporating these performance requirements into the design and development of the Atmiya University Alumni Association project, the application can deliver a high-performing, reliable, and user-friendly experience for the designated faculty and authorities, enabling them to efficiently manage, analyse, and gain valuable insights from the extensive alumni data.

Advantages:

The application boasts a plethora of remarkable features that enhance data management and analysis, saving valuable time and effort for its users. Let's elaborate on these exceptional qualities:

1. Swift and Automatic Data Management:

The application leverages automation to handle data efficiently, ensuring that data management tasks are completed in mere fractions of a minute.

2. Time-Saving Efficiency:

By automating data processes, the application dramatically reduces the manual workload, leading to significant time savings for users.

3. Autonomous Data Analysis:

With its built-in data analysis capabilities, the application independently conducts comprehensive data analysis, enabling users to glean valuable insights effortlessly.

4. Automated Analysis Distribution:

The application autonomously distributes data analysis, ensuring relevant insights reach the appropriate recipients seamlessly.

5. Automatic Bifurcated Data Distribution:

The application effortlessly distributes bifurcated data to concerned authorities, promoting efficient collaboration and decision-making.

6. Workload Reduction through Automation:

Data handling automation lightens the user's workload, streamlining the entire data management process.

7. Prompt Bifurcation and Analysis Updates:

Users can access the latest bifurcation and analysis results at any time, enabling them to stay up-to-date with the most recent information.

8. Versatility and Compatibility:

The application demonstrates flexibility by running smoothly on older systems with fewer features, ensuring accessibility for a wider range of users.

9. Compact Size:

Its small size makes the application easy to download, install, and operate efficiently, without imposing heavy resource demands.

10. Secure Data Handling:

Data privacy is a top priority, as the application disconnects from global networks after data collection, eliminating the risk of data leakage through network vulnerabilities.

11. Elimination of Duplicate Entries:

The application intelligently removes duplicate entries, ensuring that data accuracy is maintained and no redundant information is included.

12. User-Friendly Interface:

Navigating the application is effortless, as users only need to select the input data Excel file and submit it to the application for seamless data processing.

13. Convenient Bifurcation Storage:

Bifurcated data is automatically stored in the same location as the input file, ensuring organized and easily accessible data subsets.

14. Robust Error Handling:

In the event of an incorrect format or input file, the application gracefully handles errors without causing any malfunctions, maintaining stability and user confidence.

Testing and Versional controls:

During the development phase of the application, rigorous testing will be conducted using test datasets to ensure the correctness and functionality of each individual operation. Subsequently, real data will be employed to further validate the application's performance and accuracy, ensuring its suitability for real-world scenarios.

Before the application is deployed, comprehensive testing will be carried out to assess its resilience under various challenging conditions. All possible edge cases and scenarios where the application might encounter issues will be thoroughly evaluated and addressed.

Recognizing that innovation is an ongoing process, the application will continuously evolve and improve. After its initial deployment, updates and enhancements will be introduced through the release of new versions. Each new version will cater to the diverse requirements of different users, offering a range of features to accommodate varying needs.

Users will be encouraged to use the latest versions of the application to access the most up-to-date functionalities and improvements. However, the older versions of the application will not be abruptly discontinued without proper consideration. They will continue to be accessible to users, ensuring continuity and preventing any disruption for those who prefer to stay on previous versions.

By adopting this approach, the application aims to provide a seamless and progressive user experience. As the user base and requirements evolve, the continuous release of new versions will ensure that the application remains relevant and aligned with users' changing needs, solidifying its position as a valuable tool for managing and analysing the alumni data of Atmiya University.

Conclusion:

In conclusion, this SRS (Software Requirements Specification) serves as a comprehensive blueprint for the A^3 – Atmiya Alumni Association project. It outlines the essential features, functionalities, and constraints necessary for successful development. With this roadmap in place, we are poised to create a remarkable software solution that meets user needs and exceeds expectations.