**Vision Document for “SALMANS”**

**Team members:**

*Chinedu Urbanus Ugwu -*

*John Imhonikhe Lawal - 610098*

*Nhu Quy Nguyen - 986604*

*Mohammed Seud - 109362*

**1. Introduction**

Mr. Luns owns a beauty salon where he performs unisex hairstyles. Initially, he used to get 5-10 customers every week that want to get their hair done. Gradually the number of customers exponentially increased and so he started giving out piece of papers stating the position of members in the waiting queue. The issuing of papers was done on a first-come first-served basis, and customers would have to respond when their numbers were called. This process became very inefficient as the number of customers continued to increase and some customers would lie about their position – even going to the point of coming with falsified papers. He began to record customer names and numbers every day in excel sheets to reduce such occurrences but as the number of customers increased, it was just too difficult to manage manually. Currently, he has employed new hands to assist in the hairstyling and priority management but with larger customer numbers, this approach is still failing. Also, due to the large number of customers, it is difficult for him to get customer feedback ensure that his employees are delivering satisfactory services.

SALMANS is a new salon management software tool that will address these challenges. Mr. Luns customers will now be able to register with his business via the software and make reservations for hairstyling. They will be able to choose from available dates and times, as well as select the style they want from a variety of styles available. They can also provide information on unique styles that they may want to get. They can also choose to be open to rescheduling (which is cheaper) or they can insist on receiving the service at the exact date and time. The software will also let the customers apply for an expedited service, which will come at a higher fee. When they do that, the system will check for customers who are open to rescheduling and reschedule them for other dates or times. This software will also assign hairstylists to customers automatically as well as let customers rate the quality of the service they received and provide reviews. Each hairstylist will also be able to view the details of customers assigned to them and the time for each person. Mr. Luns will also be able to view all reservations and allocations. He is responsible adding new hairstylists to the system and viewing the reviews of customers. He can also view statistics of his business and see how much money he is making at any point in time.

**2. Positioning**

**2.1 Problem Statement**

|  |  |
| --- | --- |
| The problem of | *Managing customers’ priority and getting customers feedback on service delivery.* |
| Affects | *Mr. Luns, his staff and his customers.* |
| The impact of which is | *Scheduling is complex, must be manually maintained, and updated frequently.* |
| A successful solution would be | *One tool which manages customers schedule and monitors service delivery.*  *This tool will provide a Database and a user interface that is easy to use for Mr. Luns, staff, and customers.* |

**2.2 Product Position Statement**

|  |  |
| --- | --- |
| For | *Mr Luns* |
| Who | *Needs to manage the priority of customers in his salon and get feedback on the service they receive.* |
| SALMANS | *Is a business management system* |
| That | *Helps customers make reservations, keeps track of their priorities and enables to them to make reviews on the service they received remotely.* |
| Unlike | *Having customers come to the salon to compete for priority and managing the priority with Excel spreadsheets and physical evaluation of service quality.* |
| Our product | *Is web-based, so customers can make reservations from anywhere, and the business owner can evaluate his business performance remotely as well.* |

**3. Stakeholder Descriptions**

**3.1 Stakeholder Summary**

****

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Responsibilities** |
| Admin (Mr. Luns) | Admin add, edit, delete salon schedule. He also can manage income of the salon. | Admin is responsible for setting up, insert initial data and managing system. |
| Luns’s customer | Customers can see the schedule of the shop and then select the appropriate time. He/she also can change the schedule. | Customers just search and select appropriate time for him/her and then book that time with Luns. |
| Developers | Developers develop system on the basis of given document | Developers are responsible for developing system features, fixing bug, and maintaining the system’s availability. |
| Testers | Testers use JUnit tool to test system or integration test. | Testers are responsible for integration testing. |

**3.2 User Environment**

*[Detail the working environment of the target user. Here are some suggestions:*

*Number of people involved in completing the task? Is this changing?*

*How long is a task cycle? Amount of time spent in each activity? Is this changing?*

*Any unique environmental constraints: mobile, outdoors, in-flight, and so on?*

*Which system platforms are in use today? Future platforms?*

*What other applications are in use? Does your application need to integrate with them?*

*This is where extracts from the Business Model could be included to outline the task and roles involved,*

*and so on.]*

**4. Product Overview**

**4.1 Product Perspective**

*[This subsection of the* ***Vision*** *document puts the product in perspective to other related products and the*

*user’s environment. If the product is independent and totally self-contained, state it here. If the product is a*

*component of a larger system, then this subsection needs to relate how these systems interact and needs to*

*identify the relevant interfaces between the systems. One easy way to display the major components of the*

*larger system, interconnections, and external interfaces is with a block diagram.]*

**4.2 Assumptions and Dependencies**

*[List each factor that affects the features stated in the* ***Vision*** *document. List assumptions that, if changed,*

*will alter the* ***Vision*** *document. For example, an assumption may state that a specific operating system will*

*be available for the hardware designated for the software product. If the operating system is not available,*

*the* ***Vision*** *document will need to change.]*

**4.3 Needs and Features**

*[Avoid design. Keep feature descriptions at a general level. Focus on capabilities needed and why (not*

*how) they should be implemented.]*























**4.4 Alternatives and Competition**

*[Identify alternatives the stakeholder perceives as available. These can include buying a competitor’s*

*product, building a homegrown solution, or simply maintaining the status quo. List any known competitive*

*choices that exist or may become available. Include the major strengths and weaknesses of each competitor*

*as perceived by the stakeholder or end user.]*

**5. Other Product Requirements**

*[At a high level, list applicable standards, hardware, or platform requirements; performance requirements;*

*and environmental requirements.*

*Define the quality ranges for performance, robustness, fault tolerance, usability, and similar*

*characteristics that are not captured in the Feature Set.*

*Note any design constraints, external constraints, or other dependencies.*

*Define any specific documentation requirements, including user manuals, online help, installation,*

*labeling, and packaging requirements.*

*Define the priority of these other product requirements. Include, if useful, attributes such as stability,*

*benefit, effort, and risk.]*