Token Management System

Software Requirement Specifications

GROUP NO: 18

RINESH S

AADITYA CK

MUHAMMAD SINAN

Contents

5. Functional Requirements

i. Input Requirements

ii. Output Requirements

iii. Processing Requirements

6. Non-Functional Requirements

i. Performance Requirements

ii. Security Requirements

iii. Usability Requirements

iv. Accessibility Requirements

7. Constraints

i. Hardware integration constraints

8. Assumptions and Dependencies

i. Assumptions

ii. Dependencies

9. Conclusion

Introduction

■ Purpose

The purpose of this Software Requirements Specification SRS document is to provide a comprehensive overview of the functional and non-functional requirements for a software application that functions as a token booking and live token tracking from users side and on the token providers side token incrementing and managing other token function. This document serves as a guide for the developers and stakeholders involved in the development of the software application, ensuring that all requirements are clearly documented and understood. The SRS document outlines the scope and limitations of the software application, as well as the constraints, assumptions, and dependencies that may affect its development and implementation. By providing a clear and concise description of the software requirements, this document ensures that the software application meets the needs and expectations of its users.

■ Scope

This SRS document describes a software application that functions as a token booking and live token tracking from users side and on the token providers side token incrementing and managing other token function. Anyone who gives value to time and simplicity in life will use this application From the hospitals side when other hospitals have implemented this apps facilities other hospitals have to use this app to stand in the compiling environment. It will reduce congestion in hospitals, Those hospitals using our app get more patients than others because up to a limit everyone values time more than distance, Can be upgraded for other areas like Bank & financial institutions, School, Hotel, Saloon etc, Can give ads in app, token display, also in token advertised recommendation while selecting hospitals according to country , state ,etc collaboration with hardware manufactures of display and token printing machine in future.

■ Overview

The remaining sections of this document will describe the functional and non-functional requirements for the software application. The requirements will be organized into sections that describe the general requirements, functional requirements, non-functional requirements, constraints, and assumptions and dependencies.

Stakeholder Analysis

■ Identification of Stakeholders

1. Primary Stakeholders

   All patients and there bystanders(apparently all),Doctors, receptionist, nurses and all staffs in hospitals.

2. Secondary Stakeholders

   All the other users and token providers of bank, Financial bank & financial institutions School, Hotels , saloons etc

➢ Requirements Proposed By Stakeholders

1. The application should work 24/7 even if there any maintenance it should be at night time and should be done fast as can
2. There should be options to consider the late comers and those who want to take tests and revisit the doctor
3. There should be minimal delay in live token displaying and should not contain any errors

■ Requirement Gathering Methodologies

1. Contacting different hospital boards and staffs of the hospitals Through linked in ,calling and direct visiting
2. Public opinion survey In TKMCE, Google forms was circulated to analyze the general public considerations while going to hospitals

General Requirements

■ Product Perspective

It is a platform that enables hospitals to host there token management in it and there by the patients of there hospital can use these facilities. This application reduces the congestion in hospitals through the alarm system that notify before particular number of tokens and notify To start travel and reach destination before the token.

■ Product Functions

➢ User

1. Login
2. Select country,state,district,hospital and doctor
3. Book token
4. Select time of travel to hospital
5. Payment option to book token
6. User data filling at the time of registering
7. Request for reconsidering late coming
8. Request to accept revisit after tests
9. Alarm setting according to time and token
10. Disable default alarm

➢ DOCTOR

1. Login
2. Activate/deactivate user option to select this doctor token booking
3. Increment token
4. Call late comers
5. Calling again after tests
6. Add revisit option for the current token user
7. Current token ,Token left ,Total tokens showing

- **Staff**

    1. Login
    2. Setting hospital secret code
    3. Add token

■ User Classes and Characteristics

The target audience for this software application is the patients and there relatives. The users of this application will be those who use smart phones and familiar with using applications, even if the patient is not a smartphone user the relatives or family members Other audiences are hospitals doctors and the staffs who who will be able to use the application and website.

■ Operating Environment

The software application will be designed to work on any modern web browser, including Chrome, Firefox, and Safari. The application will require an internet connection to access external data sources, application is available in Android, iOS, HTML, HTML Canvas, Linux, Windows, macOS. In the production environment, the web application should have access to a MongoDB database.

■ Design Implementation and Constraints

The application web view is implemented using HTML, CSS, JavaScript, php, php myadmin and any necessary web application frameworks. The application will be implemented using a modern web application server, the application is build using flutter.

➢ External Interface Requirements

➔ User Interfaces

- **<u>Login and authentication:</u>**

    1. The user should be able to log into the application using his email and password. Once logged, the user will be greeted with a corresponding dashboards.

2. User has to fill some personal details that are needed for the prescription and aadhaar no before going to dashboard
3. Doctor login through email password and the secret code of hospital set by staff
4. Staffs has to register the hospital with details of hospital and has to set secret code also then goes to dashboard

- **User dashboard 1**

1. User selects hospital by searching and selects doctor to get into the token booking dashboard

- **User dashboard 2**

1. Where booking of token,current token displaying ,alarm setting,request for revisit after tests.
2. request to consider late comer

- **Doctor dashboard**

1. It contains side bar to activate or deactivate token booking
2. In main screen displaying current token,tokens left ,total tokens, re visitors number pending ,late comers number pending.
3. Option to increment token,calling late comers,calling re visitors option.
4. Allow revisit request option for the current token user
5. Current visitors name and age are there in the doctor dashboard

- **Staff dashboard**

1. Add new token,set time limit for token registration,set amount for token booking.

➔ Hardware Interfaces

1. Computer system: This includes the hardware components used for running the handwriting recognition software and performing the evaluation. This can

include processors, memory, storage, and other components like keyboard, mouse, touch screen and a display.
2. Network connection: A network connection is required to upload and download files, access online resources, and communicate with systems.

➔ Software Interfaces

1. Graphical User Interface (GUI): The graphical user interface of the software is the primary interface for users to interact with the software. It includes features like menus, icons, buttons, and other graphical elements that allow users to navigate and interact with the software.
2. Database Interface: The database interface allows the software to interact with a database or data storage system to store and retrieve data related to the users, hospitals, doctors, and token.
3. External API: mainly used for transaction of money

Functional Requirements

■ Input Requirements

The software application will require the following input from the user:

1. **From user:** personal details such as (name,aadhaar number,sex,age,etc)selection of hospital and doctor ,in alarm setting number of tokens before alarm are the data needed to be entered by user
2. **From doctor:** only hospital secret code , and. Login details
3. **Staff:** details of hospitals such as name,district,state,country,and many details related to the hospital such as establishment ,pictures,evident details to confirm its valid hospital.

■ Output Requirements

1. Show current token number,tokens left,total tokens,alarm on reaching token, and a remainder for doctor to consider the late comers and revisit patients.

■ Processing Requirements

1. Retrieve and store personal information,hospital registration details,token details.
2. Alarm before particular token and on reaching token

Nonfunctional Requirements

■ Performance Requirements

1. The software application must be capable of updating live tokens at intervals of no more than 2 seconds, while also being able to efficiently handle up to 10,000 simultaneous user requests.
2. The application must be able to handle up to 10,000 simultaneous user requests.

■ Security Requirements

1. The software application must ensure that user data is protected from unauthorized access or theft.
2. The application must implement proper authentication.
3. authorization mechanisms to ensure that only authorized users can access the system.

■ Usability Requirements

1. The software application must be easy to use .
2. navigate for users with basic computer literacy.
3. The application must provide clear and concise instructions to guide the user through the process booking the tokens.

■ Maintainability Requirements

1. The software application must be designed and implemented in a modular and extensible manner to enable easy maintenance and updates.
2. The application must be well-documented to facilitate future development and maintenance efforts.

Constraints

■ Hardware integration constraints

1. Integrating our application with the current token display systems and token printing system is difficult because each hospitals use different systems

Assumptions and Dependencies

■ Assumptions

1. The software application assumes that the users and hospitals will provide correct informations.

■ Dependencies

1. The software application is dependent on a modern web application server and web application frameworks to implement the required functionality.

Conclusion

1. This application is made as a platform for all hospitals to eradicate all there token related problems
2. It is designed with maintainability, and performance , scalability in mind.
3. And for user ease, convenience, save time and effort, eradicate queuing problem and related troubles in all hospitals, improve selection of hospitals.
4. This platform can increase the overall experience in hospital, reduce number of staffs in hospitals.