Software Requirements Specification (SRS) Template

The document in this file is an annotated outline for specifying software requirements, adapted from the IEEE Guide to Software Requirements Specifications (Std 830-1993).

Tailor this to your needs, removing explanatory comments as you go along. Where you decide to omit a section, you might keep the header, but insert a comment saying why you omit the data.

CS3911

Team Number 10

Butler Bot

Software Requirements Specification

Document

Version: 1.0 **Date: (02/01/2022)**

Table of Contents

3.7.5 Stimulus

3.7.6 Response

3.7.7 Functional Hierarchy

3.8 Additional Comments

4. Change Management Process

5. Document Approvals

6. Supporting Information

1. Introduction

1.1 Purpose

The purposes of this document are:

1. To state the requirements for the Discord Bot Scheduling System.
2. To show the analysis of the requirements (dependencies, issues resolved, how the requirements have been understood, etc.)

1.2 Scope

This software system will be designed to utilize Discord API to send notification with Discord bot. This application will allow users to manually or from CSV files to input the message and schedule messages to the Discord bot channel.

1.3 Definitions, Acronyms, and Abbreviations.

Term	**Definition**
Database	Collection of all information monitor by this system
User	Application consumers

1.4 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

Software Requirements Specification,April 15, 2004,Web Publishing System.

1.5 Overview

The next chapter, the Overall Description section, of this document gives a general overview of the functions to feature the product. It describes the informal requirements and is used to establish a context for the technical requirements specification.

The third chapter, Specific Requirements section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

2. The Overall Description

This system is designed to be a facility for scheduling meetings and has many potential applications, such as scheduling reminders and keeping track of school work and deadlines.

The system is implemented with advanced technology (Discord API, Raspberry Pi and Python framework). With the ability to handle high demand levels, it also utilizes deploy automation CI/CD to update the hardware with the newest software at all times.

2.1 Product Perspective

The Discord bot system is a virtually self contained scheduling system; however, it will require users to have access to a web browser on their workstation computer. Users will have to invest in any kind of computer in order to access the browser to schedule their reminders. The system will also have the ability to send notifications on the Discord channel to remind users their due dates.

2.1.1 System Interfaces

The system is to be an internet-command terminal based system, meaning that all user interaction is done through the remote terminal of the system and also remote CSV,Excel files. The System interfaces required on the system server are the following:

- Network interface to a network with an internet connection
- Raspberry Pi is on all time in order to be a hosting server.

2.1.2 Interfaces

All interaction with Discord bot will occur through the pre-created Discord channel and pre-existing data sheet files.

2.1.3 Hardware Interfaces

Raspberry Pi

2.1.4 Software Interfaces

Software will interface with Discord through Discord API

2.1.5 Communications Interfaces

There are no communications interfaces to this system.

2.1.6 Memory Constraints

There are no memory constraints to this system.

2.1.7 Operations

The system will be updated through GitHub CI/CD. With any failed build, the software can be rolled back to the last stable version. This ensures the hosting machine will be available most of the time.

2.2 Product Functions

This section contains information about high level features of the software which has been reviewed and agreed among development team members.

Schedule reminder: Users should be able to schedule reminders in order to keep track of school work deadlines.

Meeting Scheduler: To make sure meetings happen on time, the software allows users to schedule meeting appointments and users should receive notification on Discord.

2.3 User Characteristics

There is only one type of user in this system (general user). General users will be able to use all the features of the system and modify any data they set.

All users should have some knowledge about GitHub, Excel, CSV and basic computer terminal commands.

2.4 Constraints

The system is hosted in a Raspberry Pi and this requires an available Raspberry Pi machine at all times.

2.5 Assumptions and Dependencies

The system is installed on a Raspberry Pi machine.

All the scheduled data is required from user input through remote command terminal or users can upload schedule sheet data on Discord bot repository. Incorrect format input will cause the software to behave unexpectedly.

2.6 Apportioning of Requirements.

Discord bot is fully intended to be shipped out with all operational and complete features with its first release. Future features will be released whenever the first version is fully stable and functional.

3. Specific Requirements

This section specifies the detailed requirements which the system shall meet.

3.1 External Interfaces

The system doesn’t have a GUI. Users' interaction is held through Raspberry Pi terminal. All new data is merged and saved in a pre-existing data sheet which allows Raspberry Pi to keep track of reminder date and time.

3.2 Functions

Name: Schedule reminder

Description: Users input time and date with message to schedule automatic reminders on Discord channel.

Source of input and destination of output: All users input data can be entered in Raspberry Pi terminal and data sheet can be uploaded on Discord Bot repository. All new data will be merged with the production build whenever new pull requests are merged. Results will be sent out as reminders on the Discord bot channel.

Name: Schedule meeting.

Description: Users input time and date with note for reminder meeting and scheduled messages are sent out on Discord bot channel as the results.

Source of input and destination of output: All users input data is entered through Raspberry Pi terminal and also data can be merged from excel, csv data sheet files. Discord bot sends messages as meeting reminders through the Discord bot channel.

3.4 Logical Database Requirements

This section specifies the logical requirements for any information that is to be placed into a database. This may include:

- *Types of information used by various functions*
- *Frequency of use*
- *Accessing capabilities*
- *Data entities and their relationships*
- *Integrity constraints*
- *Data retention requirements*

If the customer provided you with data models, those can be presented here. ER diagrams (or static class diagrams) can be useful here to show complex data relationships. Remember a diagram is worth a thousand words of confusing text.

All data is saved stored in csv or excel file with logical structure is given below:

Reminder:

Column Name	Data Type	Description
Id	int	Identity column
Title	varchar	Reminder title
Note	varchar	Reminder note which will be send in the Discord bot
Tag	varchar	Tag people or names in the reminder message
TimeStamp	datetime	Date and time which reminder is schedule to send out

3.5 Design Constraints

Discord bot doesn’t have any design constraints.

3.6 Software System Attributes

3.6.1 Reliability

The Raspberry Pi is available for 99.99% of the time without failure

3.6.2 Availability

New data can be updated all time throughout the data and GitHub repository can be used to merge new data with Raspberry Pi at all times.

3.6.3 Security

Software can only access with internal users who have access to the Discord bot repository.

3.6.4 Maintainability

Software is managed by GitHub. With any failed operation, the software always rolled back to the previous stable version.

3.6.5 Portability

Software is integrated with Raspberry Pi, so portability is very high and very expense efficient.