

**PROCESS CONTROLLER**

**Software Requirement Specification (SRS) Document**

**Sprint 2 Implementation**

**Project Timeline: 02.01.2023 to 11.01.2023**

**INDEX**

1. **Introduction**

* 1. Purpose ------------------------------------------------------------------------------------------------- 4
  2. Intended use --------------------------------------------------------------------------------------- 4

1.3 Scope ---------------------------------------------------------------------------------------------------4

1. **Overall description**

2.1 Project Features --------------------------------------------------------------------------------6

2.2 User needs ----------------------------------------------------------------------------------------6

2.3 Operating Environment ------------------------------------------------------------------------6

2.4 Assumptions and dependency --------------------------------------------------------------- 6

1. **System feature and requirements** 
   1. **Functionality** 
      1. FR\_01-> Process Manager --------------------------------7
      2. FR\_02-> PM application --------------------------------7
      3. FR\_03-> User input --------------------------------7
      4. FR\_04-> Employee Logout --------------------------------7
      5. FR\_05-> Appropriate menu options --------------------------------7
      6. FR\_06-> Statistics information ------------------------------- 7
      7. FR\_07-> PM app should be able to exit --------------------------------7
      8. FR\_08-> Display statistics for each process --------------------------------7
      9. FR\_09-> Running a process --------------------------------7
      10. FR\_10-> Control user program --------------------------------7
      11. FR\_11-> Parameter’s validation --------------------------------7
      12. FR\_12-> Control user program --------------------------------7
      13. FR\_13-> Control user program --------------------------------7
      14. FR\_14-> Control user program --------------------------------7
   2. **System requirement** -----------------------------------------------------------------------7

**3.3 Functional Requirements** --------------------------------------------------------------------- 8

1. **Data Flow Diagram** 
   1. DFD level 0 ----------------------------------------------------------------------------------------- 9
   2. DFD level 1 ------------------------------------------------------------------------------------------10

# 1. INTRODUCTION: -

Process controller is a system software which takes the input and display the output accordingly, the user can give the program input and can perform multiple operations on the given program and can display various statistics information of that process by giving the process name.

## 1.1 Purpose: -

The system works as a client - server based application. User/Client is required to enter the user input and then they can select the options to perform task of the selected program, they can view the statical information by giving the process name and can get the various information of the selected program such as memory occupied space and CPU time

and the user can add or delete the program and can exit when the process is completed.

**1.2 Intended Use: -**

* Development Team
* Maintenance Team
* Clients

Since this a general-Purpose Software Thus any one Can access it.

## 1.3 Scope: -

The scope of the project is to create a process controller. This system consists of process which can be started and stopped according to user wish and perform multiple operations on it.

# 2. Overall Description: -

**2.1 Project Features:**

### The user interfaces attractive and user-friendly.

### The entire process is menu driven.

### The user can give the input and the process displays the output.

### The client will have the menu option and the user can choose the option.

### The server will perform the main task such as starting a process, stopping a process, temporarily suspend the program.

### The client can view the various statistics information of the selected program such as current running process and paused process.

### The client can select a program to view the information such as memory occupied by that process and CPU time.

* After completing the process, the user can exit.

**2.2 User Needs**

1. User Characteristics: The user should be familiar with menu-driven Applications.
2. General Constraints: A full internet connection is required for Linux (Operating System).
3. Intended audience:
4. Developers
5. Project Manager

**2.3 Operating Environment**

The operating environment for the application is listed below

* Operating system: Any Linux-based OS

**2.4 Assumptions and Dependency: -**

* System should have any flavour of Linux installed.

* System should have either 4GB or more RAM.

* The service is used preferably on a desktop or laptop.

# 3. System Features and Requirements: -

**3.1 Functionality: -**

**3.1.1 FR\_01-> Process Manager:** that monitors and controls different processes.

**3.1.2 FR\_02-> PM application:** It will have a menu-based solution for accepting

user input and displaying output.

**3.1.3 FR\_03-> User input**: A list of programs can be given as input by the user.

**3.1.4 FR\_04-> Employee Logout:** It is used to capture the employee’s logout details that who has been logged out.

**3.1.5 FR\_05-> Appropriate menu options:** The PM program will provide appropriate menu options for the user.

**3.1.6 FR\_06-> Statistics information:** The PM program will display the statistics information of that program such as current running process.

**3.1.7 FR\_07-> PM app should be able to exit:** It should be able to exit once the current program is completed.

**3.1.8 FR\_08-> Display statistics for each process:** It should be able to display the statistical information based on the given program name.

**3.1.9 FR\_09-> Running a process:** At any time only one process should be running.

**3.1.10 FR\_10-> Control user program:** The PM app will be able to control any custom user programs as well as generic system programs.

**3.1.11 FR\_11-> Parameter’s validation:** The PM app will be able to validate the parameters which are passed.

**3.1.12 FR\_12-> Control user program:** The PM app will be able to control any custom user programs as well as generic system programs.

**3.1.13 FR\_13-> Control user program:** The PM app will be able to control any custom user programs as well as generic system programs.

**3.1.14 FR\_14-> Control user program:** The PM app will be able to control any custom user programs as well as generic system programs.

## 3.2 System Requirements: -

System Requirements are types of functional requirements. These are features that are required for a system to function.

**Software Interface:**

* Operating System: Linux OS which supports networking.
* Connect protocol: TCP protocol

**Hardware Interface:**

Hardware requirements are:

* Processor: i3 or above
* ROM: 1TB (SSD/HDD)
* RAM: 4 GB or above

**3.3 Functional Requirements:**

**3.3.1 G6\_TR01 - Process Synchronization:** It is the way by which processes that share the same memory space are managed in an operating system. Here, it ensures that multiple clients accessing the common data i.e., hotel data is synchronized, thereby avoiding conflicts.

**3.3.2 G6\_TR02 - Shared Memory in Linux:** All data related to the hotel such as room types, price, etc. are shared by multiple clients. Mutex can be used for locking and unlocking the shared resources to avoid data corruptions and booking errors.

**3.3.3 G6\_TR03 - Socket Programming in C - TCP:** Socket programming is a way of connecting two nodes, here the client and server, on a network to communicate with each other and coordinate the hotel booking activities.

**3.3.4 G6\_TR04 - Support for statistics:** Server is responsible for the display of statistics related to availability of rooms such as number of rooms booked and vacant.

**3.3.5 G6\_TR05 - I/O Multiplexing:** I/O multiplexing is the ability to perform I/O operations on multiple file descriptors.

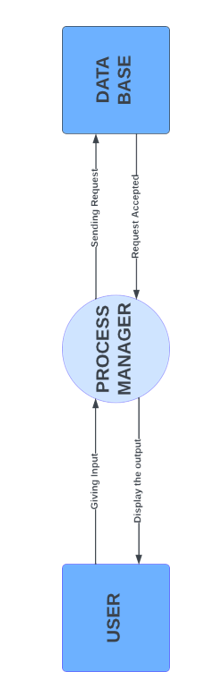
**3.3.6 G6\_TR06 - Logging and Debugging Framework:** Linux logs provide a timeline of events for a valuable troubleshooting tool when encountering issues. When issues arise, analyzing log files facilitates debugging.

**4. Data Flow Diagram:**

#### 4.1 DFD Level 0 –

This Level 0 is also known as context diagram, it is an abstract view with the

mechanism represented as a single process with external parties. This DFD for the system depicts the overall structure as a single bubble. The incoming, outgoing indicators showing input and output data.



#### 4.2 DFD Level 1: -

The context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses.

