Workspace PC Usage Dashboard - Real Time

Software Requirement Specification



Centre for Computational Technologies

Transforming human life by democratization of technology

<https://www.cctech.co.in>





# Introduction

## Purpose

The purpose of this document is to outline the requirements for the development of a system that monitors CPU, RAM, Disk usage, login status etc., of PCs in a LAN and provides real-time data visualization.

## Scope

The system will include a client app on PCs, a server app for data processing, and a dashboard monitoring app on an admin's PC.

## Definitions

* WMI: Windows Management Instrumentation
* WPF: Windows Presentation Foundation
* C#: Programming language used for development
* Visual Studio: Integrated Development Environment (IDE)

# Overall Description

## Product Perspective

The monitoring system is a standalone application that interacts with remote PCs in the LAN and centralizes the data on the admin’s PC for visualization.

## Product Functions

* Client App
* Collects system information using WMI.
* Sends collected data to the server app.
* Server App
* Receives data from client apps.
* Processes and organizes the received data.
* Forwards processed data to the dashboard monitoring app.
* Dashboard Monitoring App
* Receives and visualizes real-time data from the server app.
* Displays CPU, RAM, disk usage, login status, etc., using WPF controls and charts.

## User Classes and Characteristics

* Admin User: Monitors PCs within the LAN.
* End Users: Users of the PCs being monitored.

## Operating Environment

* Client App: Windows OS on target PCs (supports Windows 7 and later).
* Server App: Windows OS (supports Windows Server 2012 and later).
* Dashboard Monitoring App: Windows OS on admin's PC.

## Design and Implementation Constraints

* All components developed using C# and WPF.
* Use of WMI for system information retrieval.

# Specific Requirement

## External Interface Requirements

* Client-Server Communication: TCP/IP sockets for data transmission.
* Dashboard Interface: WPF framework for real-time visualization.

## Functional Requirements

* Client App
* Must collect CPU, RAM, disk usage, and login status.
* Should transmit data securely to the server app.
* Server App
* Must receive data from clients.
* Should process and organize data efficiently.
* Dashboard Monitoring App
* Must display real-time data in graphical format.
* Should provide clear visualization of system stats.

## Non-functional Requirements

* Performance: System should provide real-time data updates within a 5-second interval.
* Security: Data transmission should be encrypted using SSL/TLS.
* Usability: User interface elements should be intuitive and easy to understand.

## System Features

* Client App
* Collects and transmits system information
* Server App
* Receives, processes, and forwards data.
* Dashboard Monitoring App
* Displays real-time data using charts and graphs.

# Software Tools

* Visual Studio 2022
* C# programming Language
* WPF(Windows Presentation Foundation)

# Milestone

| **SR No.** | **Milestone** | **Description** | **Timeline** |
| --- | --- | --- | --- |
| 1. | SRS Approval | SRS Approval by client | 05/01/2024 EOD |
| 2. | Development of Client Application | Development and Testing of Client and application | 06/01/2024 10:00 AM |
| 4. | Development of Server Application | Development and Testing of Server Application | 06/01/2024 EOD |
| 6. | Development of Dashboard Application with UI | Development and Testing of Dashboard Application with responsive UI | 07/01/2024 8:00 PM |
| 7. | Final Testing of Product | Final Testing of Client, Server and Dashboard Apps | 07/01/2024 EOD |
| 9. | Product and PPT presentation | Final Presentation of product | 08/01/2024 2:00 PM |

# Conclusion

This software comprises three standalone applications, Client app which would be installed on PCs to be monitored. The server app would be running on an IP address which will collect data from PCs in LAN via Client app. Lastly Dashboard app will fetch that data in real time from server and shows it in the form of graphs and charts.