

# **SOFTWARE REQUIREMENT SPECIFICATION FOR SCHOOL ATTANDANCE SYSTEM**

# Software Requirements Specification (SRS)

Table of Contents.....

**1. Introduction .....1**

1.1 Purpose .....1

1.2 Document Conventions ..... 1

1.3 Product Scope .....1

**2. Overall Description .....2**

2.1 Product Perspective .....2

2.2 Product Functions .....2

2.3 Operating Environment .....2

2.4 Design and Implementation Constraints .....2

2.5 Assumptions and Dependencies .....3

2.6 User Documentation.....3

**3. External Interface Requirements  
.....3**

3.1 User Interfaces .....3

3.2 Hardware Interfaces .....3

3.3 Software Interfaces.....3

**4. System Features .....4**

4.1 System Feature 1 .....4

**5. Other Nonfunctional Requirements .....4**

5.1 Safety Requirements .....5

5.2 Security Requirements .....5

5.3 Software Quality Attributes .....5

# **1. Introduction**

## **1.1 Purpose**

The main purpose of this document is to get detailed overview of the SAS's features, workings & requirements. It will give users truly a new experience of taking attendance online. This document is created for users of the software and also potential developers.

## **1.2 Document Conventions**

This Document was created based on the IEEE template for System Requirement Specification Documents.

## **1.3 Product Scope**

SAS is the software which people can use for taking attendance of students online. User can give their attendance. They can check their attendance percentage. The whole process from giving the attendance to checking the percentage of attendance is very easy. Parents will be also able to check their child's attendance by logging in the Id provided to them.

# **2. Overall Description.**

## **2.1 Product Perspective**

SAS was developed for making the online attendance system more easy and effective. It can show you the daily attendance of the students. It is an open source project and it has a very active developer team to support it and provide feedback to users.

## **2.2 Product Functions.**

There are two kinds of process models for this system. There is the overview process mode and the conditional process model. Starting with the first one. The first step of this process is to have a fingerprint capture device. That will do the following steps

: 1- Students enter their fingerprints into the device.

2:- Here every fingerprint will be having a unique id which will be stored in the student database. The id will take the step to check whether it is matching or not.

3: - The system checks the finger print and sends it to the database.

4:- In the database the system check that if the fingerprint added is matching with any fingerprint stored in the database. If not then it will show the error message to the student and he/she have to enter the fingerprint again.

5:-The student has to re-enter the finger print and after the matching of 80% print, the system will mark the attendance of the student.

6:- After that system will immediately send the attendance to the faculty so that they can check who is attending their class and who is not.

7:- There is another process in this software too that is if a student have missed the class then the system will send the message to the parents and the respective faculty. If the student is able to provide the valid reason for not attending the class then faculty can refill the attendance of that student.

## **2.3 Operating environment**

This project will go through two steps: The first step is to have the automatic attendance device in every classroom in the school. These devices will be connected to the computer and its system. Students have to put their fingerprints on file in the registration office on their first day to save their fingerprint data in the database. The second step is to connect this system to the site. That is to connect the database to the system database to work as one system on the site. This step would complete the work, and the project will work in one system. That is because the attendance Report will be updated all the time. Also, the site will control all the students' attendance Reports not in a separate system or database. This system has some requirements to be accomplished. It needs hardware and software. Hardware requirements:

- 1) Biometric Fingerprint Scanners
  - 2) Cables for the device the current system work is already in existence.
- However, we need some system requirements:
- 1) Create new databases and indexes for students and class list by using mysql
  - 2) Make connation to the current database
  - 3) Design interfaces for the users
  - 4) Design an attendance page on
  - 5) Programming using JavaScript, PHP, and HTML.

## **2.4 Design and implementation constraints.**

SAS is developed in Java, PHP, HTML; it uses OpenGL for its visualization engine and has been built on top of the Net Beans Platform. It uses a modular design where every feature is wrapped into a separate module and the modules depend on each other through well-written APIs. There are several APIs available to make plug-in development easy.

## 2.5 Assumptions and Dependencies.

SAS is developed in Java, PHP and HTML and therefore requires Java to be installed on the user's system. The latest stable version of SAS requires Java version 7 or higher. This applies to Windows and Linux users. On Mac OS X, Java is bundled with the application.

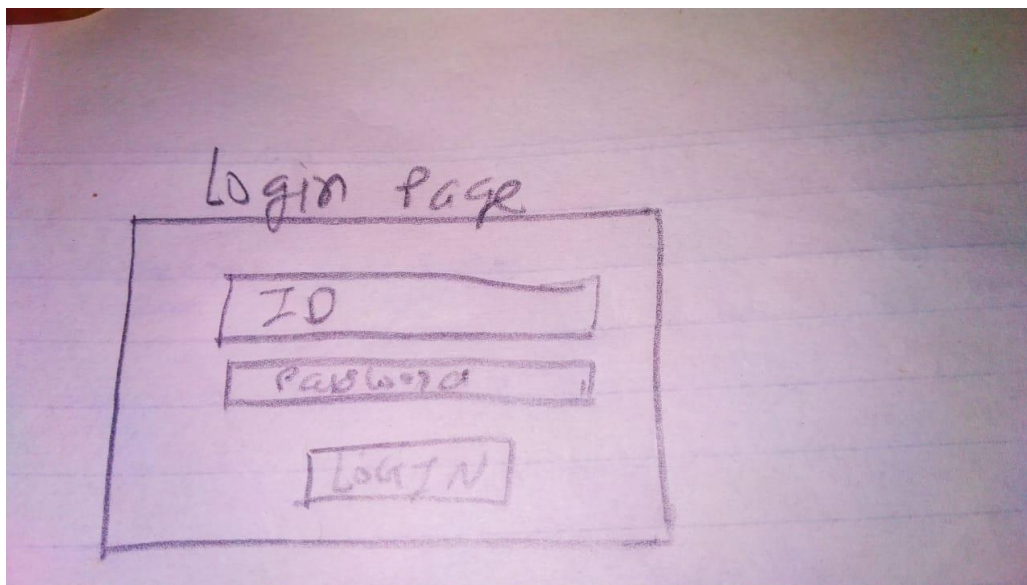
## 2.6 User Documentation.

One help page is provided in web app in that page all the feature like -how to see the attendance -how to contact faculty for any query -how to update attendance -how to provide the valid reason for the absence. All the thing mention on that page with appropriate guidance and instruction with all the pages screenshot.

## 3. External Interface Requirements

### 3.1 User Interface

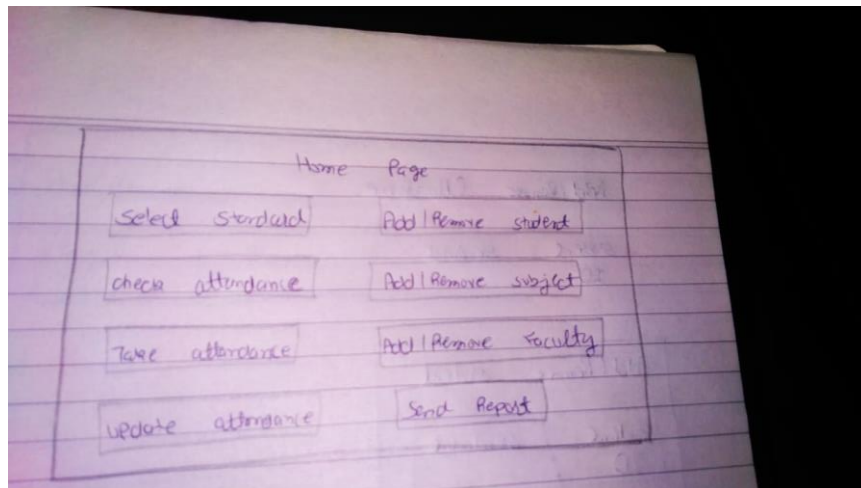
➤ Login page



➤ Sign up page

Hand-drawn sketch of a sign-up form on lined paper. The form is enclosed in a rectangular box and contains labels for Name, Stud. ID, Email, Phone no., Parent no., Password, and Confirm pw., each followed by a horizontal input field. At the bottom of the box are two buttons labeled 'signup' and 'cancel'. Below the box, the text 'Sign Up Page' is written.

➤ Home page



➤ Select standard

A hand-drawn UI sketch on lined paper. It features a rectangular container with a title 'Select standard' at the top. Below the title is a dropdown menu with the word 'select' and a small downward-pointing arrow on the right side.

➤ Check attendance

A hand-drawn UI sketch on lined paper for a 'check attendance' form. The form is enclosed in a large rectangle. At the top, it is titled 'check attendance'. Below the title are two input fields: 'student name :' and 'student ID :'. Under these fields are two buttons labeled 'Go' and 'Cancel'. At the bottom of the form is a label 'Attendance Percentage :' followed by an empty rectangular box for the value.

➤ Add/Remove student



Add / Remove student

Student ID	Standard
------------	----------

Add Cancel

➤ Add/Remove faculty

Add / Remove Faculty

Faculty ID	Name	Standard
------------	------	----------

Add Cancel

➤ Add/Remove subject

Add / Remove Subject

Subject Code	Subject name	Faculty name	Standard

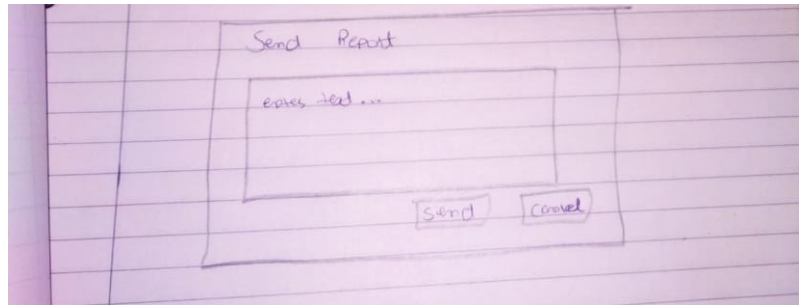
Add / Remove Cancel

➤ Taking attendance

Take attendance				
Student ID	Subject Name	Date	P	A

- 
- Updating attendance

Update Attendance				
Student ID	Subject Name	Date	P	A
		Update	Cancel	



➤ Send report

## 3.2 Hardware Interfaces

The system will use:

- 1) Biometric Fingerprint Devices display software
- 2) Web pages for the forms HTML, PHP
- 3) Server
- 4) Programming using JavaScript
- 5) Database uses with mysql

## 4. System Features

This section demonstrates SAS's most useful and unique features and explains how they can be used and the results they will give back to the user.

### 4.1 Adding a New student:

To add a new user to the system, all of them should have registered in the admission office before they can register in their classes. On the orientation day, all students must scan their thumbs in the input device for only one time to save the fingerprint data in the registration office to sign up.

## **4.2 Use the system to attend to classes**

When students have a class, they must scan their thumbs in the fingerprint input device. If the scan matches, students can enter the class, and they will be checked on the page. If the scan does not match, the student must check with the registration office to figure out the checking device.

## **4.3 Report students**

When the students have enrolled in the class, they are now able to check on their current attendance situation through the page. In the system, they will be shown a page that gives them the whole attendance status in the semester.

## **4.4 Faculty receive a report**

The system will send a message after ten minutes of the class time to the faculty. Faculty will have the all students' attendance reports in the particular class. Faculty can modify some of the attendance grades if he/she needs.

## **4.5 Students missed classes**

When a student misses a class, he/she will receive a message via email and page. Students must long in to the page and go to the attendance page to write the reason for missing the class. Students have to submit the form to wait for the response of the faculty's decision.

## **4.6 Students missed two classes and more**

When students miss more than one class, students will receive a warning message for missing two classes or more. The warning message should be for the missed class for the whole semester and their status in a danger level.

## **4.7 Faculty check the report and the attendance control panel page**

Faculty has to check on the report and give the final submission. Faculty has full control of modifying any grades and looking at students who have excuses to modify their grades. Faculty receives messages from the system about students who missed classes. For students who submit the note for the missed class, faculty members could look at the note and give a decision on the student's grades

## **5 Other non-functional requirements.**

### **5.1 Safety requirements.**

The current system is build upon user name and password access. Students/Faculty can access through page & they control it. System has its own rules and regulations of use. Strong security is the part of policy's purpose.

### **5.2 Security Requirements.**

To ensure that no one of SAS's users loses any data while using SAS (due to a crash or a bug of some kind) the developer team updates SAS regularly. There is a bug tracker available where users can report any bugs they have encountered so that the developers can fix it in the next release.

### **5.3 Software Quality Attributes.**

SAS provides the users simple features. Due to its well designed and easy to use interface it can be used by both experts and typical users.

## **Gantt Chart for School Attendance System**

Project Name: School Attendance System

Manager:Vadnagara Dhairya

Start Date: April 23th, 2022

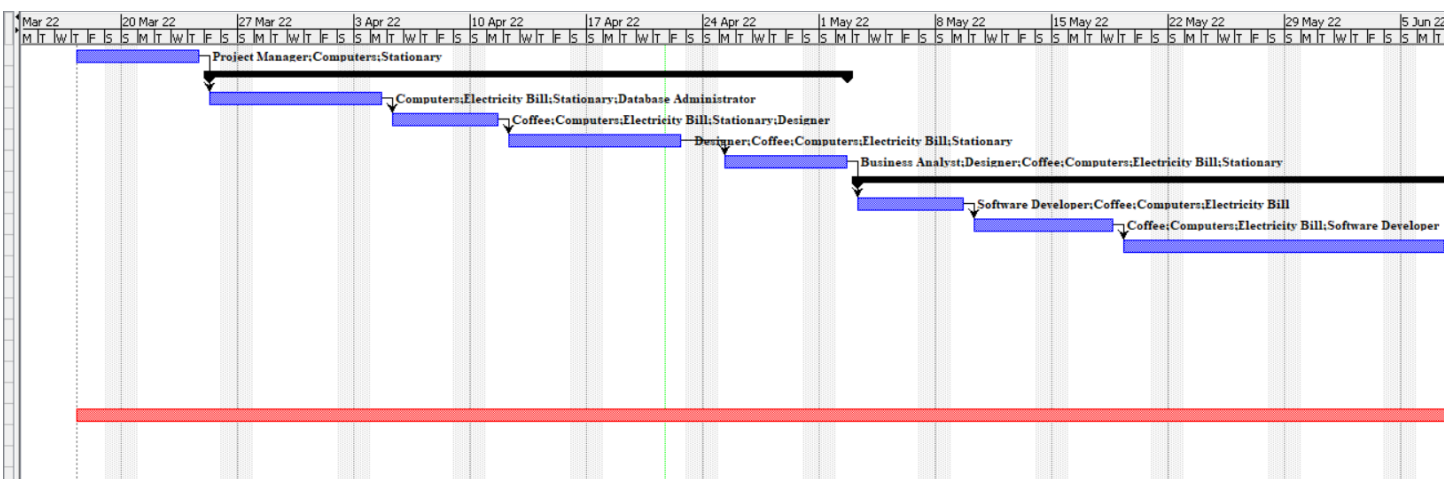
Notes: This is termwork project

# 1. Tasks

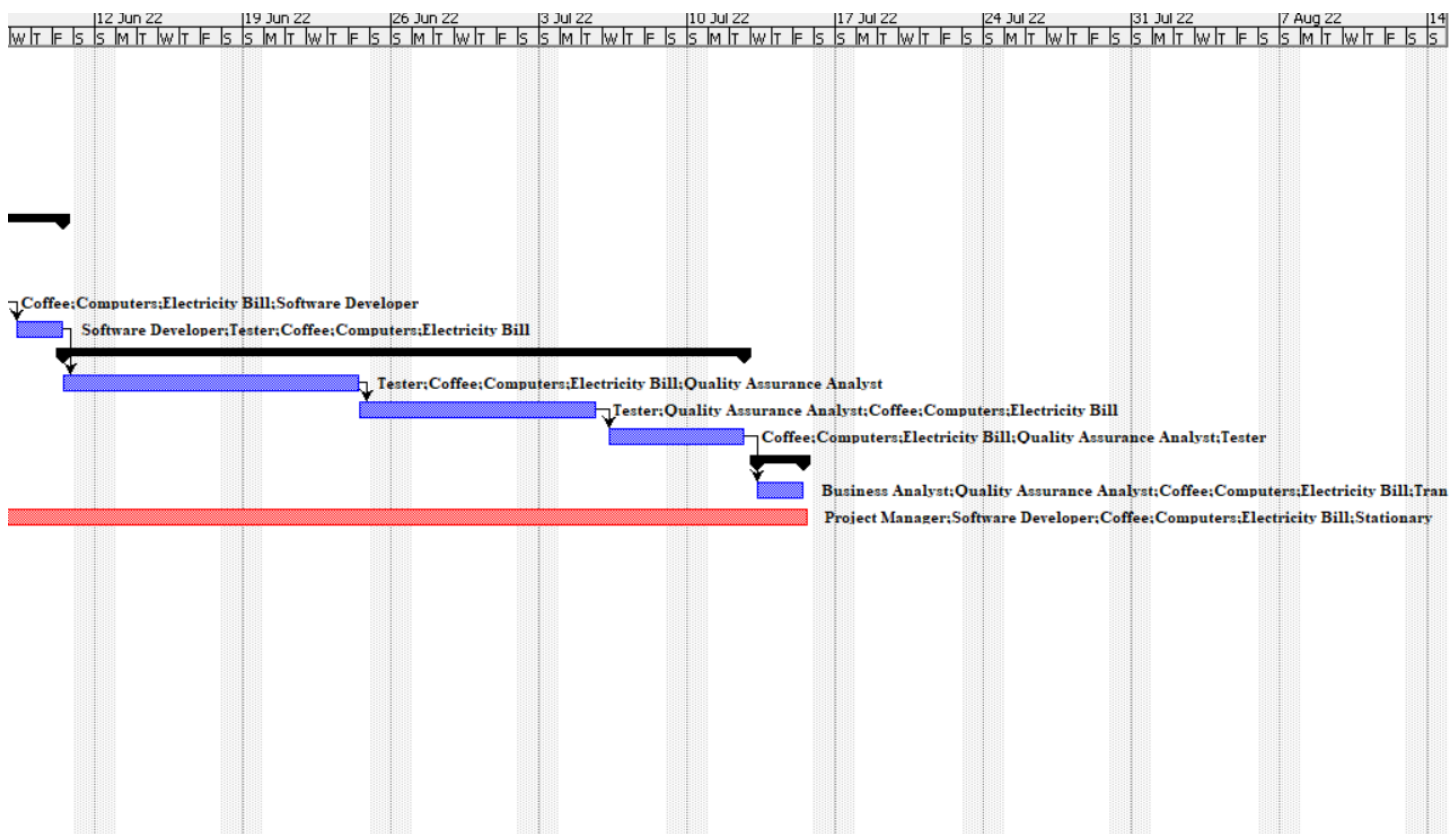
	📌	Name	Duration	Work	Start	Finish	Predecessors	Resource Names
1		System Documentation	6 days	48 hours	17/3/22 8:00 AM	24/3/22 5:00 PM		Project Manager;Computers;Stationary
2	📁	Design	27 days	264 hours	25/3/22 8:00 AM	2/5/22 5:00 PM		
3		Design Database	7 days	56 hours	25/3/22 8:00 AM	4/4/22 5:00 PM	1	Computers;Electricity Bill;Stationary;Database Administrator
4		Software Design	5 days	40 hours	5/4/22 8:00 AM	11/4/22 5:00 PM	3	Coffee;Computers;Electricity Bill;Stationary;Designer
5		Interface Design	9 days	72 hours	12/4/22 8:00 AM	22/4/22 5:00 PM	4	Designer;Coffee;Computers;Electricity Bill;Stationary
6		Create Design Sprecification	6 days	96 hours	25/4/22 8:00 AM	2/5/22 5:00 PM	5	Business Analyst;Designer;Coffee;Computers;Electricity Bill;Stationary
7	📁	Development	28.5 days	248 hours	3/5/22 8:00 AM	10/6/22 1:00 PM		
8		Deploy Development Envir	5 days	40 hours	3/5/22 8:00 AM	9/5/22 5:00 PM	6	Software Developer;Coffee;Computers;Electricity Bill
9		Deploy system modules	7 days	56 hours	10/5/22 8:00 AM	18/5/22 5:00 PM	8	Coffee;Computers;Electricity Bill;Software Developer
10		Integrate System modules	14 days	112 hours	19/5/22 8:00 AM	7/6/22 5:00 PM	9	Coffee;Computers;Electricity Bill;Software Developer
11		Perform Initial Testing	2.5 days	40 hours	8/6/22 8:00 AM	10/6/22 1:00 PM	10	Software Developer;Tester;Coffee;Computers;Electricity Bill
12	📁	Testing	22.5 days	280 hours	10/6/22 1:00 PM	12/7/22 5:00 PM		
13		Deploy Test Environment	10 days	80 hours	10/6/22 1:00 PM	24/6/22 1:00 PM	11	Tester;Coffee;Computers;Electricity Bill;Quality Assurance Analyst
14		Perform System Test	7.5 days	120 hours	24/6/22 1:00 PM	5/7/22 5:00 PM	13	Tester;Quality Assurance Analyst;Coffee;Computers;Electricity Bill
15		Document Issues	5 days	80 hours	6/7/22 8:00 AM	12/7/22 5:00 PM	14	Coffee;Computers;Electricity Bill;Quality Assurance Analyst;Tester
16	📁	Deployment	2.5 days	40 hours	13/7/22 8:00 AM	15/7/22 1:00 PM		
17		Acceptance Testing	2.5 days	40 hours	13/7/22 8:00 AM	15/7/22 1:00 PM	15	Business Analyst;Quality Assurance Analyst;Coffee;Computers;Electricity Bill;Transportation
18		User Documentation	87 days	1,392 hours	17/3/22 8:00 AM	15/7/22 5:00 PM		Project Manager;Software Developer;Coffee;Computers;Electricity Bill;Stationary

Mainly, there are five tasks available, inside the task's subtasks are present with there working hours, total number of working days required to complete that particular task and with its starting and ending dates. Predecessors shows the connection between the tasks like after completing the task which task should start. Resource name column shows the assignment of there source to that particular task.

## 2. Gantt Chart







This Gantt Charts shows the timeline of the project. In which it displays the timeline of the particular tasks and its subtasks with their source assignment labeling in it and even connectors are also present in it which shows about the connection between the tasks like after complete a particular task which task needs to get started.

### 3. Resources

	Ⓢ	Name	RBS	Type	E-mail A...	Material Label	Initials	Group	Max. Units	Standard Rate	Overtime Rate	Cost Per Use	Accrue At	Base Calendar
1	Ⓢ	Project Manager		Work			P		100%	Rs. 5000.00/hour	Rs. 0.00/hour	Rs. 0.00 Prorated		Standard
2	Ⓢ	Software Developer		Work			S		100%	Rs. 4000.00/hour	Rs. 0.00/hour	Rs. 0.00 Prorated		Standard
3	Ⓢ	Business Analyst		Work			B		100%	Rs. 4500.00/hour	Rs. 0.00/hour	Rs. 0.00 Prorated		Standard
4	Ⓢ	Database Administrator		Work			D		100%	Rs. 2000.00/hour	Rs. 0.00/hour	Rs. 0.00 Prorated		Standard
5	Ⓢ	Tester		Work			T		100%	Rs. 2000.00/hour	Rs. 0.00/hour	Rs. 0.00 Prorated		Standard
6	Ⓢ	Designer		Work			D		100%	Rs. 3000.00/hour	Rs. 0.00/hour	Rs. 0.00 Prorated		Standard
7	Ⓢ	Quality Assurance Analyst		Work			Q		100%	Rs. 0.00/hour	Rs. 0.00/hour	Rs. 0.00 Prorated		Standard
8	Ⓢ	Coffee		Material			C			Rs. 3000.00		Rs. 0.00 Prorated		
9	Ⓢ	Computers		Material			C			Rs. 6000.00		Rs. 0.00 Prorated		
10	Ⓢ	Electricity Bill		Material			E			Rs. 5000.00		Rs. 0.00 Prorated		
11	Ⓢ	Stationery		Material			S			Rs. 1000.00		Rs. 0.00 Prorated		
12	Ⓢ	Transportation		Material			T			Rs. 6000.00		Rs. 0.00 Prorated		

This image shows all the resources available in the project in the form of work and material. Work resources include project manager, software developer, business analyst, database administrator, tester, designer, and quality assurance analyst. Material resources include coffee, computer, electricity bill, stationery, and transportation charges to be done while the development of the project.



## 4. Resource assignment

1	Project Manager	120 hours				
	Onsite meetings	8 hours Flat		0 days	0 days	Rate A
	Stakeholders Requirement	40 hours Flat		0 days	0 days	Rate A
	Finalize Stakeholders requi	24 hours Flat		0 days	0 days	Rate A
	System Documentation	48 hours Flat		0 days	0 days	Rate A
2	Software Developer	228 hours				
	Deploy Development Envir	40 hours Flat		0 days	0 days	Rate A
	Deploy system modules	56 hours Flat		0 days	0 days	Rate A
	Integrate System modules	112 hours Flat		0 days	0 days	Rate A
	Perform Initial Testing	20 hours Flat		0 days	0 days	Rate A
3	Business Analyst	68 hours				
	Create Design Sprecificatio	48 hours Flat		0 days	0 days	Rate A
	Acceptance Testing	20 hours Flat		0 days	0 days	Rate A
4	Database Administrator	56 hours				
	Design Database	56 hours Flat		0 days	0 days	Rate A
5	Tester	160 hours				
	Perform Initial Testing	20 hours Flat		0 days	0 days	Rate A
	Deploy Test Environment	40 hours Flat		0 days	0 days	Rate A
	Perform System Test	60 hours Flat		0 days	0 days	Rate A
	Document Issues	40 hours Flat		0 days	0 days	Rate A
6	Designer	160 hours				
	Software Design	40 hours Flat		0 days	0 days	Rate A
	Interface Design	72 hours Flat		0 days	0 days	Rate A
	Create Design Sprecificatio	48 hours Flat		0 days	0 days	Rate A
7	Quality Assurance Analyst	160 hours				
	Deploy Test Environment	40 hours Flat		0 days	0 days	Rate A
	Perform System Test	60 hours Flat		0 days	0 days	Rate A
	Document Issues	40 hours Flat		0 days	0 days	Rate A
	Acceptance Testing	20 hours Flat		0 days	0 days	Rate A
8	Coffee	0 hours				
	Software Design	1 Flat		0 days	0 days	Rate A
	Software Design	1 Flat		0 days	0 days	Rate A

8	Coffee	0 hours				
	Software Design	1 Flat	0 days	0 days	Rate A	
	Interface Design	1 Flat	0 days	0 days	Rate A	
	Create Design Specification	1 Flat	0 days	0 days	Rate A	
	Deploy Development Environment	1 Flat	0 days	0 days	Rate A	
	Deploy system modules	1 Flat	0 days	0 days	Rate A	
	Integrate System modules	1 Flat	0 days	0 days	Rate A	
	Perform Initial Testing	1 Flat	0 days	0 days	Rate A	
	Deploy Test Environment	1 Flat	0 days	0 days	Rate A	
	Perform System Test	1 Flat	0 days	0 days	Rate A	
	Document Issues	1 Flat	0 days	0 days	Rate A	
	Acceptance Testing	1 Flat	0 days	0 days	Rate A	
9	Computers	0 hours				
	System Documentation	1 Flat	0 days	0 days	Rate A	
	Design Database	1 Flat	0 days	0 days	Rate A	
	Software Design	1 Flat	0 days	0 days	Rate A	
	Interface Design	1 Flat	0 days	0 days	Rate A	
	Create Design Specification	1 Flat	0 days	0 days	Rate A	
	Deploy Development Environment	1 Flat	0 days	0 days	Rate A	
	Deploy system modules	1 Flat	0 days	0 days	Rate A	
	Integrate System modules	1 Flat	0 days	0 days	Rate A	
	Perform Initial Testing	1 Flat	0 days	0 days	Rate A	
	Deploy Test Environment	1 Flat	0 days	0 days	Rate A	
	Perform System Test	1 Flat	0 days	0 days	Rate A	
	Document Issues	1 Flat	0 days	0 days	Rate A	
	Acceptance Testing	1 Flat	0 days	0 days	Rate A	

10	Electricity Bill	0 hours				
	Design Database	1 Flat	0 days	0 days	Rate A	
	Software Design	1 Flat	0 days	0 days	Rate A	
	Interface Design	1 Flat	0 days	0 days	Rate A	
	Create Design Specification	1 Flat	0 days	0 days	Rate A	
	Deploy Development Environment	1 Flat	0 days	0 days	Rate A	
	Deploy system modules	1 Flat	0 days	0 days	Rate A	
	Integrate System modules	1 Flat	0 days	0 days	Rate A	
	Perform Initial Testing	1 Flat	0 days	0 days	Rate A	
	Deploy Test Environment	1 Flat	0 days	0 days	Rate A	
	Perform System Test	1 Flat	0 days	0 days	Rate A	
	Document Issues	1 Flat	0 days	0 days	Rate A	
	Acceptance Testing	1 Flat	0 days	0 days	Rate A	
11	Stationary	0 hours				
	Stakeholders Requirements	1 Flat	0 days	0 days	Rate A	
	Finalize Stakeholders requirements	1 Flat	0 days	0 days	Rate A	
	System Documentation	1 Flat	0 days	0 days	Rate A	
	Design Database	1 Flat	0 days	0 days	Rate A	
	Software Design	1 Flat	0 days	0 days	Rate A	
	Interface Design	1 Flat	0 days	0 days	Rate A	
	Create Design Specification	1 Flat	0 days	0 days	Rate A	
12	Transportation	0 hours				
	Onsite meetings	1 Flat	0 days	0 days	Rate A	
	Stakeholders Requirements	1 Flat	0 days	0 days	Rate A	
	Finalize Stakeholders requirements	1 Flat	0 days	0 days	Rate A	
	Acceptance Testing	1 Flat	0 days	0 days	Rate A	

This resource assignment shows about the tasks which has been assign to there source in the form of hours and cost.

NAME: DHAIRYAVADNAGARA  
ROLL NO: MA059