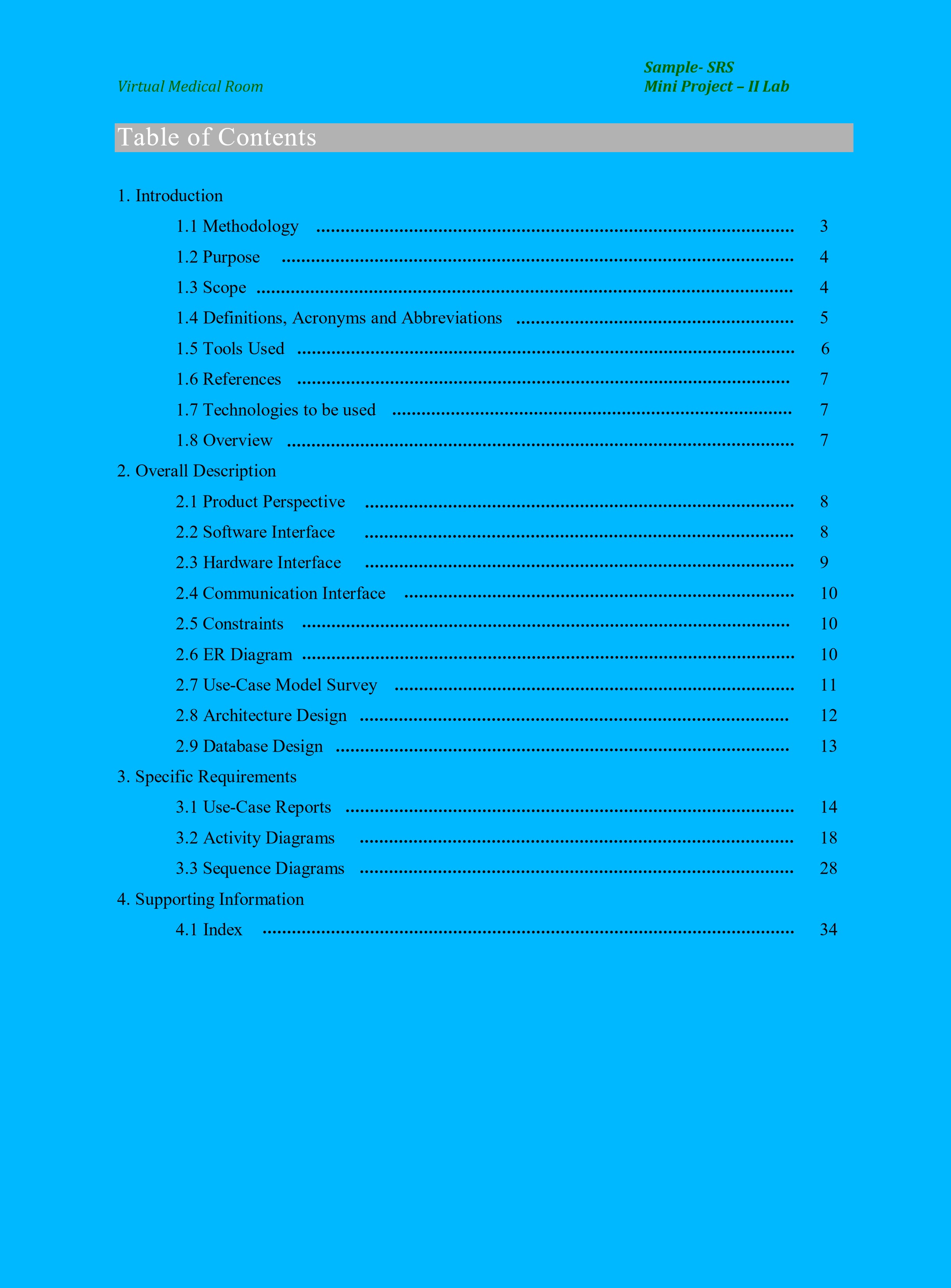
|  |
| --- |
| **MINI PROJECT- II**  **(Session)**  **Design and implementation of staff member availability using RFID technology**  **Software Requirement Specifications**      **Institute of Engineering & Technology**    **Team Members**  Shivangi Tripathi (161500278)  Samiksha Gupta (161500478)  Zaved Alam (161500646)  Ayaz Khan (161500154)  ***Mini Project Guide***  Ms. Debjani Ghosh  Assistant professor  Department of Computer Engineering & Applications |



**1.INTRODUCTION**

**Methodology**

### Waterfall (Traditional)

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

The following illustration is a representation of the different phases of the Waterfall Model.



The sequential phases in Waterfall model are −

* **Requirement Gathering and analysis** − All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
* **System Design** − The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
* **Implementation** − With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
* **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
* **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
* **Maintenance** − There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model, phases do not overlap.

1.2 Purpose

Employees, staff members or students in an institute face issues in searching a staff member throughout cabin rooms. Even though one finds a cabin, it isn’t necessary that he/she finds him/her. To address the discussed issues, we propose a solution that can check the availability of a staff member in his/her cabin.

1.3 scope

* There are two basic users- **Client, Admin.**
* Admin have their own profile.
* Admin has authority to add/delete/update faculty details.
* Client can see the cabin rooms and faculty availability.

1.4 Definitions, Acronyms and Abbreviations

**Admin**

Administrator. He has authority to add/delete/update faculty details.

**DB**

Database. A database management system that provides and efficient database platform to maintain records of faculty members.

**UML**

**Unified Modeling Language.** It’s a standard languagefor writing software blueprints. The UML may be used to visualize, specify, construct and document

**HTTP**

**Hypertext Transfer Protocol.** It’s a service protocol.

HTML

Hypertext Markup Language. It’s building block of html pages.

CSS

Cascading Style Sheets.It’s designed to enable the separation of presentation and content.

SQL

Structured Query Language. It’s a domain specific language used in programming and designed for managing data.

RFID

Radio-frequency identification. It’s uses [electromagnetic fields](https://en.wikipedia.org/wiki/Electromagnetic_field) to automatically identify and track tags attached to objects

PCB

 Printed circuit board mechanically supports and electrically connects [electronic components](https://en.wikipedia.org/wiki/Electronic_components).

**1.5 Tools Used**

**Hardware architecture- NodeMCU, RFID, PCB**

**NodeMCU**

NodeMCU is an open source [IoT](https://en.wikipedia.org/wiki/Internet_of_Things) platform. It includes [firmware](https://en.wikipedia.org/wiki/Firmware) which runs on the [ESP8266](https://en.wikipedia.org/wiki/ESP8266) [Wi-Fi](https://en.wikipedia.org/wiki/Wi-Fi) [SoC](https://en.wikipedia.org/wiki/System_on_a_chip) from [Espressif Systems](https://en.wikipedia.org/w/index.php?title=Espressif_Systems&action=edit&redlink=1), and hardware which is based on the ESP-12 module.[The term "NodeMCU" by default refers to the firmware rather than the development kits. The firmware uses the [Lua](https://en.wikipedia.org/wiki/Lua_(programming_language)) scripting language. It is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and spiffs.

**RFID**

Radio-frequency identification (RFID) uses [electromagnetic fields](https://en.wikipedia.org/wiki/Electromagnetic_field) to automatically identify and track tags attached to objects. The tags contain electronically-stored information. Passive tags collect energy from a nearby RFID reader's interrogating [radio waves](https://en.wikipedia.org/wiki/Radio_waves). Active tags have a local power source (such as a battery) and may operate hundreds of meters from the RFID reader. Unlike a [barcode](https://en.wikipedia.org/wiki/Barcode), the tag need not be within the line of sight of the reader, so it may be embedded in the tracked object. RFID is one method for [Automatic Identification and Data Capture](https://en.wikipedia.org/wiki/Automatic_Identification_and_Data_Capture) (AIDC).

**PCB**

Printed circuit board  mechanically supports and electrically connects [electronic components](https://en.wikipedia.org/wiki/Electronic_components) or [electrical](https://en.wikipedia.org/wiki/Electrical) components using [conductive](https://en.wikipedia.org/wiki/Electrical_conductor) tracks, pads and other features [etched](https://en.wikipedia.org/wiki/Industrial_etching) from one or more sheet layers of copper [laminated](https://en.wikipedia.org/wiki/Laminated) onto and/or between sheet layers of a [non-conductive](https://en.wikipedia.org/wiki/Insulator_(electricity)) [substrate](https://en.wikipedia.org/wiki/Substrate_(electronics)). Components are generally [soldered](https://en.wikipedia.org/wiki/Soldering) onto the PCB to both electrically connect and mechanically fasten them to it.

**Application architecture- HTML and CSS, PHP, MySQL, JAVASCRIPT, ARDUINO IDE**

**HTML and CSS**

Hypertext Markup Language is the standard markup language for creating web pages and web applications. With Cascading Style Sheets and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. [Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [web server](https://en.wikipedia.org/wiki/Web_server) or from local storage and [render](https://en.wikipedia.org/wiki/Browser_engine) the documents into multimedia web pages. HTML describes the structure of a web page [semantically](https://en.wikipedia.org/wiki/Semantic_Web) and originally included cues for the appearance of the document.

CSS is designed to enable the separation of presentation and content, including [layout](https://en.wikipedia.org/wiki/Page_layout), [colors](https://en.wikipedia.org/wiki/Color), and [fonts](https://en.wikipedia.org/wiki/Typeface). This separation can improve content [accessibility](https://en.wikipedia.org/wiki/Accessibility), provide more flexibility and control in the specification of presentation characteristics, enable multiple [web pages](https://en.wikipedia.org/wiki/Web_page) to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

**PHP**

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages. PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.

**MySQL**

MySQL is an [open-source](https://en.wikipedia.org/wiki/Open-source) [relational database management system](https://en.wikipedia.org/wiki/Relational_database_management_system). MySQL is a central component of the [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) open-source web application software stack (and other "[AMP](https://en.wikipedia.org/wiki/List_of_AMP_packages)" stacks). LAMP is an acronym for "[Linux](https://en.wikipedia.org/wiki/Linux), [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server), MySQL, [Perl](https://en.wikipedia.org/wiki/Perl)/[PHP](https://en.wikipedia.org/wiki/PHP)/[Python](https://en.wikipedia.org/wiki/Python_(programming_language))". Applications that use the MySQL database include:  [TYPO3](https://en.wikipedia.org/wiki/TYPO3), [MODx](https://en.wikipedia.org/wiki/MODx), [Joomla](https://en.wikipedia.org/wiki/Joomla), [WordPress](https://en.wikipedia.org/wiki/WordPress), [Simple Machines Forum](https://en.wikipedia.org/wiki/Simple_Machines_Forum), [phpBB](https://en.wikipedia.org/wiki/PhpBB), [MyBB](https://en.wikipedia.org/wiki/MyBB), and [Drupal](https://en.wikipedia.org/wiki/Drupal). MySQL is also used in many high-profile, large-scale [websites](https://en.wikipedia.org/wiki/Website), including [Google](https://en.wikipedia.org/wiki/Google) (though not for searches), [Facebook](https://en.wikipedia.org/wiki/Facebook), [Twitter](https://en.wikipedia.org/wiki/Twitter), [Flickr](https://en.wikipedia.org/wiki/Flickr), and [YouTube](https://en.wikipedia.org/wiki/YouTube).

**JAVASCRIPT**

JAVASCRIPT, is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [programming language](https://en.wikipedia.org/wiki/Programming_language). It is a language which is also characterized as [dynamic](https://en.wikipedia.org/wiki/Dynamic_programming_language), [weakly typed](https://en.wikipedia.org/wiki/Weak_typing), [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) and [multi-paradigm](https://en.wikipedia.org/wiki/Multi-paradigm_programming_language). Alongside [HTML](https://en.wikipedia.org/wiki/HTML) and [CSS](https://en.wikipedia.org/wiki/CSS), JavaScript is one of the three core technologies of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). JavaScript enables interactive [web pages](https://en.wikipedia.org/wiki/Web_page) and thus is an essential part of [web applications](https://en.wikipedia.org/wiki/Web_application). The vast majority of [websites](https://en.wikipedia.org/wiki/Website) use it, and all major [web browsers](https://en.wikipedia.org/wiki/Web_browser) have a dedicated [JavaScript engine](https://en.wikipedia.org/wiki/JavaScript_engine) to execute it.

**ARDUINO IDE**

The [Arduino](https://en.wikipedia.org/wiki/Arduino) integrated development environment ([IDE](https://en.wikipedia.org/wiki/Integrated_development_environment)) is a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) application (for [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS), [Linux](https://en.wikipedia.org/wiki/Linux)) that is written in the programming language [Java](https://en.wikipedia.org/wiki/Java_(programming_language)). It is used to write and upload programs to Arduino board. The source code for the IDE is released under the [GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License), version 2. The Arduino IDE supports the languages [C](https://en.wikipedia.org/wiki/C_(programming_language)) and [C++](https://en.wikipedia.org/wiki/C%2B%2B) using special rules of code structuring. The Arduino IDE supplies a [software library](https://en.wikipedia.org/wiki/Software_library) from the [Wiring](https://en.wikipedia.org/wiki/Wiring_(development_platform))project, which provides many common input and output procedures.

**1.6 References**

* NodeMCU Development Workshop- Agus Kurniawan
* HTML: [www.W3school.com](http://www.W3school.com)
* PHP: [www.Javatpoint.com](http://www.Javatpoint.com)
* RFID: A Guide to Radio Frequency Identification
* Wikipedia: [www.wikipedia.com](http://www.wikipedia.com)

**1.7 Technologies to be used**

* Relational Database Management System.
* Arduino (Esp8266).
* Xampp.

**1.8 Overview**

**Existing System:**

* Description of faculty member.
* Availability Status.

**Drawbacks:**

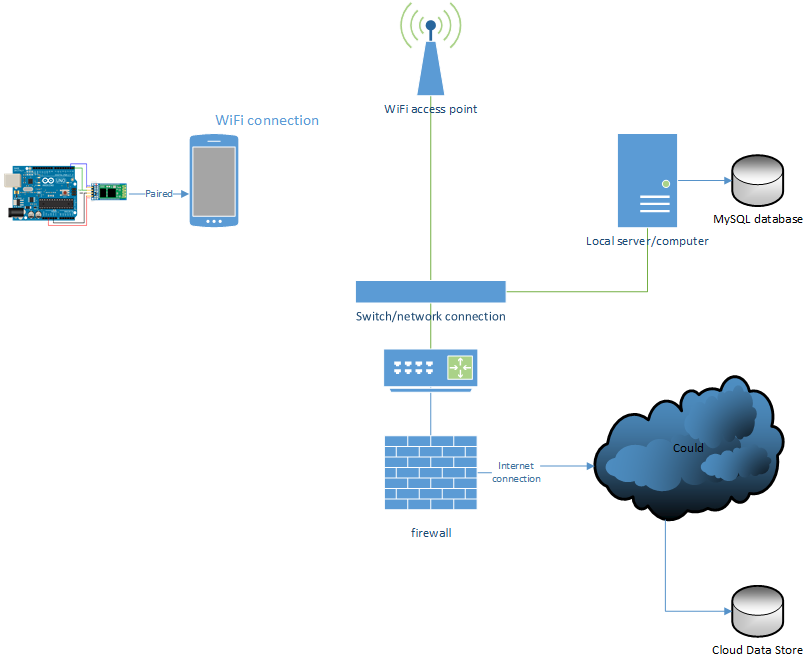
* Card required.

**Proposed System:**

* Details of faculty members could be modified.
* Client could see the availability status of each faculty member.

**2. Overall Description**

**2.1 Product Perspective**



**Fig 2.1:product Prespective**

**2.2 Software Interface**

**Client on Internet:**

Web Browser, Operating System

**Client on Intranet:**

Web Browser, Operating System

**Web Server:**

WASCE, Operating System

**Data Base Server**

DB, Operating System

**Development End**

RAD,DB, Operating System, webserver

**2.3 Hardware Interface**

Minimum Requirements:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Client Side** |  |  |  |  |
|  | **Processor** | **RAM** | **Disk Space** |  |
| Internet Explorer - 6 | Intel Pentium III or AMD - 800 MHz | 128 MB | 100 MB |  |
| **Server Side** |  |  |  |  |
|  | **Processor** | **RAM** | **Disk Space** |  |
| RAD | Intel Pentium III or AMD - 800 MHz | 1 GB | 3.5 GB |  |
| DB2 - 9.5 | 256 MB | 500 MB (Excluding  Size) | Data |

Recommended Requirements:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Client Side** |  |  |  |  |
|  | **Processor** | **RAM** | **Disk Space** |  |
| Internet Explorer - 6 | Intel Pentium III or AMD - 800 MHz | 128 MB | 100 MB |  |
| **Server Side** |  |  |  |  |
|  | **Processor** | **RAM** | **Disk Space** |  |
| RAD | Intel Pentium III or AMD - 800 MHz | 1 GB | 3.5 GB |  |
| DB2 - 9.5 | 256 MB | 500 MB (Excluding  Size) | Data |

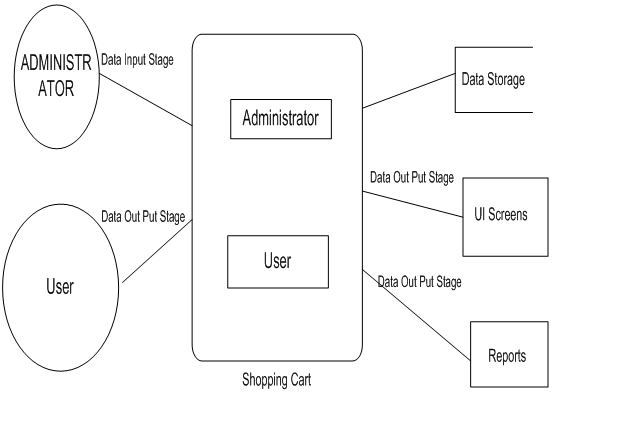
**2.4 Communication Interface**

* Client on internet will be using HTTP/HTTPS protocol.
* Admin on internet will be using HTTP/HTTPS protocol.

**2.5 Constraints**

* + GUI is only in English.
  + Login id and password is used for the identification of admins.
  + Users will be able to see the cabin or staff room members and their availability status.

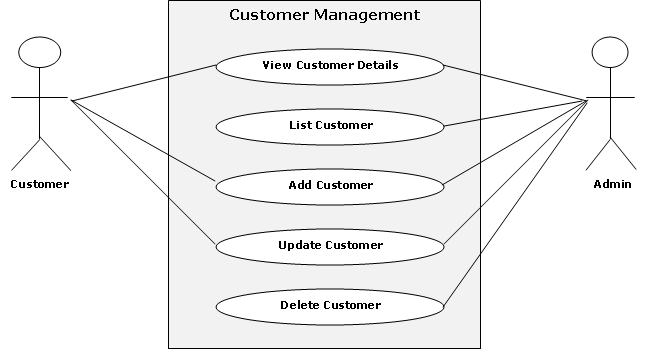
**2.6 ER Diagram**



Data Storage

LOGIN

**2.7 Use Case Model Survey**



Faculty availability

**2.8 Architecture Diagram**

**Application Layer BUSSINESS LAYER DATABASELAYER**

Login UI

Login

Login UI

Registration UI

Registration UI

Faculty Manager

Search UI

Search UI

Report UI

Report

Report UI

Web Cam Interaction UI

Web Cam Interaction UI

FacultyManager

**2.9 Database Design**

report

Admin\_detail

faculty\_detail

comments

s\_no

name

user\_id

password

s\_no

room\_no

name

availability

**3. SPECIFIC REQUIREMENTS**

**3.1 Use Case Reports**

Report

USER

View Faculty Details

Search faculty Details

Add Faculty Details

Login

Admin

Delete Details

Update Details

**3.2 Activity Diagram**

**3.2.1­­­**

Enter Faculty Details

Verification of User Name

Invalid

Faculty Name Already Exists

Fill The Mandatory field

Valid

Verification of Mandatory field

Invalid

Valid

Approved

Return to the Respective page

**3.2.2 Admin Login**

Enter userid and password

Verification of userid and password

Invalid

Valid

Admin Login Successful

Admin home Screen

**3.3 Sequence Diagram**

**3.3.1 Admin Login Sequence Diagram**

Administrator

System

Database

Enter ID and Password

Verify ID and Password

Display Admin Page

Message 1

**4. Supporting Information**

**4.1 index**

Overview

Overall

P

Product

Proposed system

Purpose

R

RAD

References

Reports

**S**

Scope

Sequence

Software

**T**

Transition

Technology

**E**

Elaboration

ER Diagram

Existing System

**H**

Hardware

HTML

HTTP

HTTPS

**I**

Interface

Introduction

**J**

JavaScript

**M**

Methodology

**O**

Operating System

Our Plan

**A**

Abbreviations-

Acronyms

Administrator

Analysis

**C**

Client on Internet

Client on Intranet

Communication

Constraints

Constructions

**D**

Database

DB

Definition

Development

Drawbacks

Introduction

**U**

Usecase

UML

**W**

Waterfall

**SPECIAL THANKS**

DEPARTMENT OF COMPUTER ENGINEERING AND APPLICATIONS

We convey a special thanks to our department and our college and also convey a special thanks to all these software’s and websites, they have been helping a lot in completing the project.

