# 1. INTRODUCTION

## 1.1 OVERVIEW

The general view of this application is to manage the lab security which is created using RFID technology. Radio Frequency Identification (RFID) is a technology designed to allow objects, animals and even human beings to be identified, located, and tracked using radio frequency signals. The history of RFID can be traced back to World War II when it was used by the British to identify friendly aircraft. The technology has developed over the years and is now used in a wide range of applications such as Electronic Product Surveillance Tag, Access Control, Transportation and logistics etc. For example, an RFID tag attached to an automobile during production can be used to track its progress through the assembly line; RFID-tagged pharmaceuticals can be tracked through warehouses, and implanting RFID microchips in livestock and pets allows for positive identification of animals.

**Why RFID?**

Radio-frequency identification (RFID) uses electromagnetic fields 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. Active tags have a local power source (such as a battery) and may operate hundreds of meters from the RFID reader. Unlike a 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 (AIDC). Thus have the following advantages when compared to barcode such as:

- Automatic-identification technology - it does not require human intervention
- No line of insight – read contents in a few seconds
- Accurate
- Read multiple tags at a time
- All RFID tags can be detected instantly within range also cross-referenced against the database.

**Why Arduino Uno?**

Arduino is a single-board microcontroller meant to make the application more accessible. The Arduino Uno board is a microcontroller based on the ATmega328. It has 14 digital input/output pins in which 6 can be used as PWM (Pulse Width Modulation) outputs, a 16 MHz ceramic resonator, an ICSP header, a USB connection, 6 analog inputs, a power jack, and a reset button. This contains all the required support needed for the microcontroller.

Arduino was created in the year 2005 by two Italian engineers David Cuartielles and Massimo Banzi where "Uno" means one with the goal of keeping in mind about students to make them learn how to program the Arduino Uno microcontroller and improve their skills about electronics and use it in the real world.

Arduino Uno microcontroller can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino also simplifies the process of working with microcontrollers, but it offers some advantage such as,

**a) Inexpensive:** Arduino boards are relatively inexpensive compared to other microcontroller platforms
**b) Cross-platform:** The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems
**c) Simple, clear programming environment:** easy-to-use for beginners, yet flexible
**d)** Open source and extensible software and hardware

The reason for choosing Arduino Uno over raspberry pi for the project is that it needs to do a single processing over and over again that can be achieved using Arduino Uno.

## 1.2 SCOPE

The hardware tools that are included in this project are,

- ✓ Arduino Uno board
- ✓ Passive RFID Tag

- ✓ RFID Reader
- ✓ Ethernet Shield
- ✓ Breadboard
- ✓ Jumper wire for connection (male to female)
- ✓ LCD display & Buzzer

The software tools that are included in this project are,

- ✓ **Frontend:** Html, CSS and JavaScript
- ✓ **Backend:** PHP
- ✓ **Database:** MySQL
- ✓ **Programmable:** Arduino IDE

## 1.3 SYSTEM INTERFACES

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, mac OS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino board. The Arduino IDE supports the languages C and C++ using special rules of code structuring.

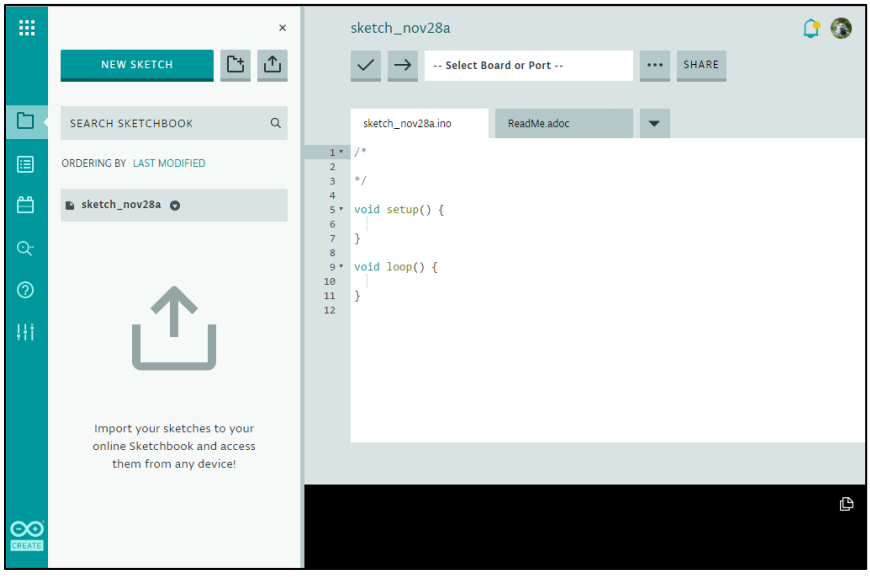



**Figure 1.1:** System Interface – Arduino IDE

The code for detecting each person using RFID tag is developed using this IDE. The working principle is that whenever the tag is shown near the RFID reader it sense and reads the tag details in which the information gets updated in the database. The information like tag ID, person ID, person name, date, time entered in and time out are recorded. The information can be viewed in the web page where the admin can sort according to the date/time/person ID.

## 1.4 USER PREREQUISITES

a) User should have basic knowledge about using the device.

b) User should know English language.

# 2. SOFTWARE AND HARWARE SPECIFICATION

## 2.1 Software Specification:

- **Arduino IDE:** It is open source software that is used to write and upload programs into Arduino board and supports languages like C and C++.
- **XAMPP:** It stands for Cross-Platform (X), Apache (A), MySQL (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing purposes.
- **Language used:**
    a) **PHP –** It stands for Hypertext Preprocessor. It's an open source, server-side, scripting language used for the development of web applications. By scripting language, we mean a program that is script-based (lines of code) written for the automation of tasks.
    b) **HTML –** The Hypertext Markup Language (HTML) is a standard for describing the structure and presentation of information via the Internet.
    c) **Java Script –** It is a client side scripting language used both on the client-side and server-side that allows you to make web pages interactive. It helps to improves user experience of the web page by converting it from a static page into an interactive one.

## 2.2 Hardware Specification:

- **Passive RFID card/tag:** Passive RFID systems use tags with no internal power source and instead are powered by the electromagnetic energy transmitted from an RFID reader. Passive RFID tags only have two main components – the tag's antenna and the microchip or integrated circuit (IC). It has the following attribute:

**Figure 2.1:** Passive RFID card/tag

- **RFID Reader (RC522):** This low cost MFRC522 based RFID Reader Module is easy to use and can be used in a wide range of applications. The MFRC522 is a highly integrated reader/writer IC for contactless communication at 13.56 MHz.


**Figure 2.2:** RFID Reader (RC522)

Its pin configuration with Ardunio is as follows:

| RFID-RC522 PIN | ARDUINO UNO PIN |
|---|---|
| SDA | 8 |
| SCK | 13 |
| MOSI | 11 |
| MISO | 12 |
| IRQ | UNUSED |
| GND | GND |
| RST | 9 |
| 3.3V | 3.3V |

**Table 2.1:** Pin Configuration of RFID Reader (RC522)

- **Arduino Uno Microcontroller (R2):** The components that make up an Arduino board and what each of their functions are given as follows,
    1. **Reset Button** – This will restart any code that is loaded to the Arduino board
    2. **AREF** – Stands for "Analog Reference" and is used to set an external reference voltage
    3. **Ground Pin** – There are a few ground pins on the Arduino and they all work the same
    4. **Digital Input/Output** – Pins 0-13 can be used for digital input or output
    5. **PWM** – The pins marked with the (~) symbol can simulate analog output
    6. **USB Connection** – Used for powering up your Arduino and uploading sketches
    7. **TX/RX** – Transmit and receive data indication LEDs
    8. **ATmega Microcontroller** – This is the brains and is where the programs are stored
    9. **Power LED Indicator** – This LED lights up anytime the board is plugged in a power source
    10. **Voltage Regulator** – This controls the amount of voltage going into the Arduino board
    11. **DC Power Barrel Jack** – This is used for powering your Arduino with a power supply
    12. **3.3V Pin** – This pin supplies 3.3 volts of power to your projects
    13. **5V Pin** – This pin supplies 5 volts of power to your projects
    14. **Ground Pins** – There are a few ground pins on the Arduino and they all work the same
    15. **Analog Pins** – These **pins can read the signal from an analog sensor and convert it to digital**

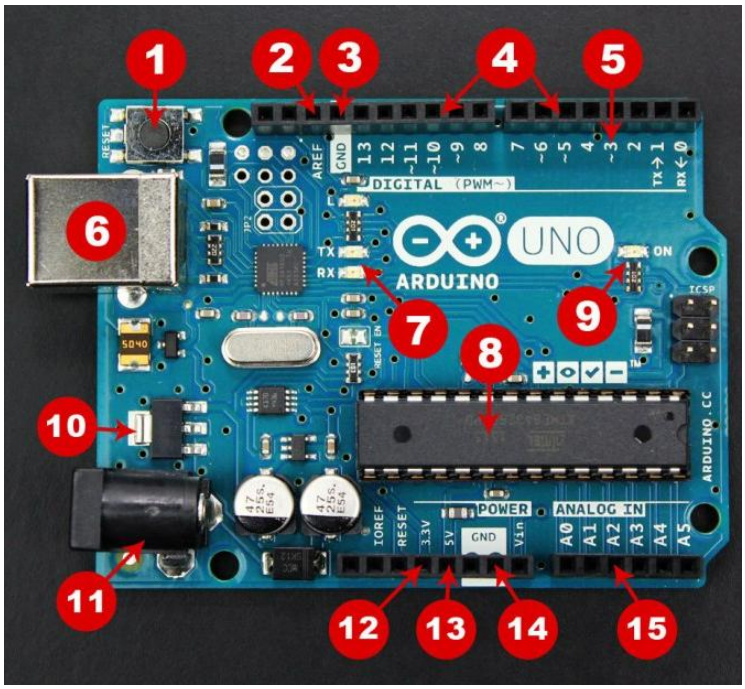



**Figure 2.3:** Arduino Uno Microcontroller (R2)

- **Ethernet Shield:** The Arduino Ethernet Shield allows an Arduino board to connect to the internet. It is based on the Wiznet W5100 ethernet chip (datasheet). The ethernet shield connects to an Arduino board using long wire-wrap headers which extend through the shield. It uses RJ45 Cable that needs to be connected to router for data transfer.

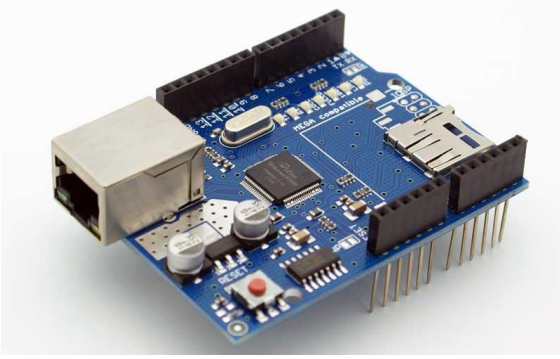


**Figure 2.4:** Ethernet Shield

- **LCD Display with I2C:** To provide user interfaces and commonly used to display data in devices. Its pin configuration with LCD i2c with Ardunio is

| LCD I2C 16X2 PIN | ARDUINO UNO PIN |
|---|---|
| SDA | A4 |
| SCL | A5 |
| GND | GND |
| VCC | 5.5V |

**Table 2.2:** Pin Configuration of LCD Display with I2C

**Figure 2.5:** LCD Display with I2C

# 3. SYSTEM DESIGN

## 3.1 INTRODUCTION

This purpose of this document is to understand the work flow of the model.

## 3.2 ARCHITECTURAL DESIGN

The representation of the proposed system is depicted in the following figures:

**a) Software Architecture:**

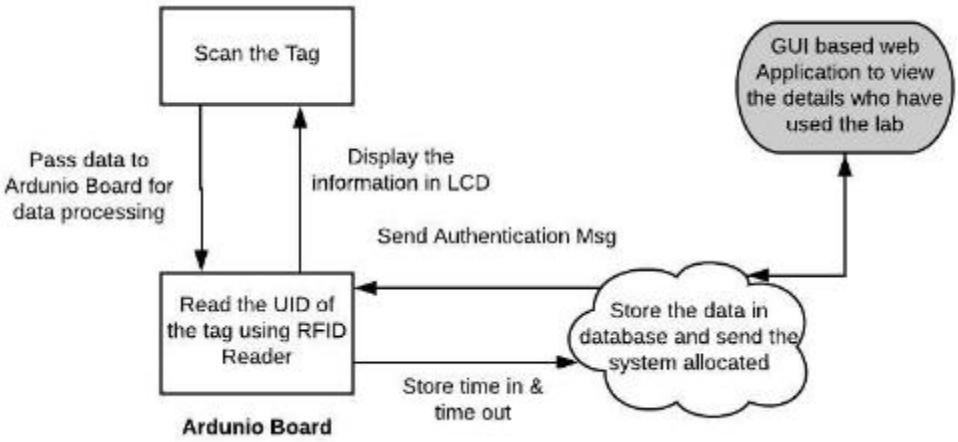



**Figure 3.1:** Software Architecture

**b) Hardware Architecture:**

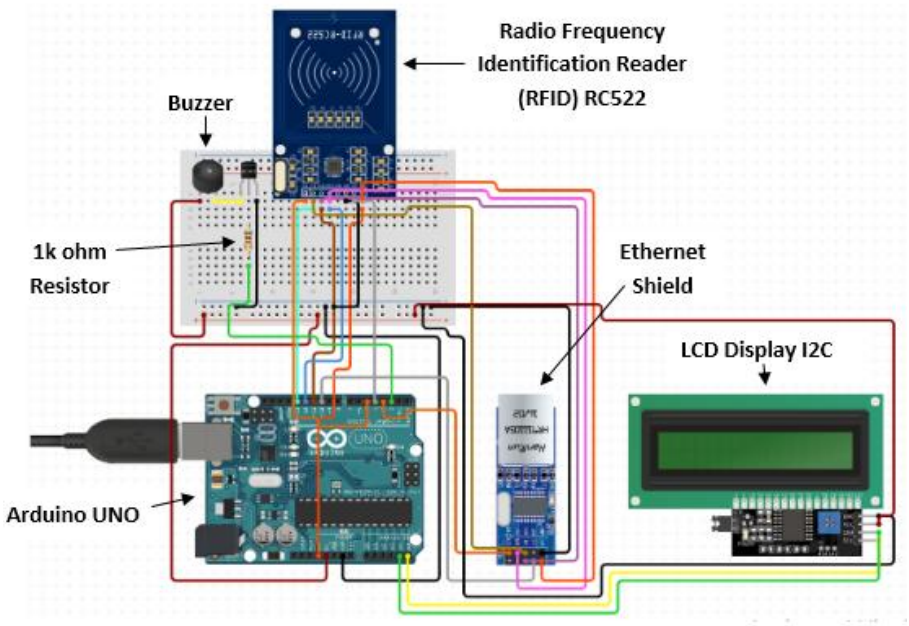



**Figure 3.2:** Hardware Architecture

## 3.3 USECASE DIAGRAM

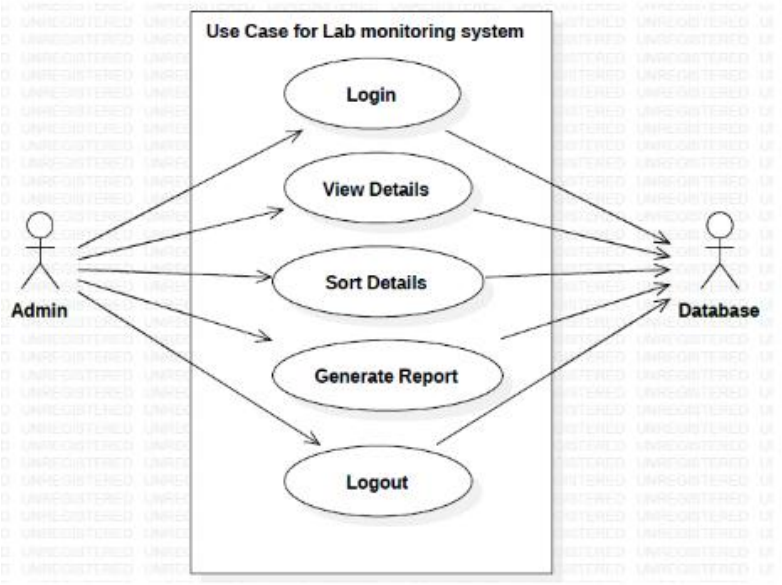



**Figure 3.3:** Use Case Diagram

## 3.4 ACTIVITY DIAGRAM

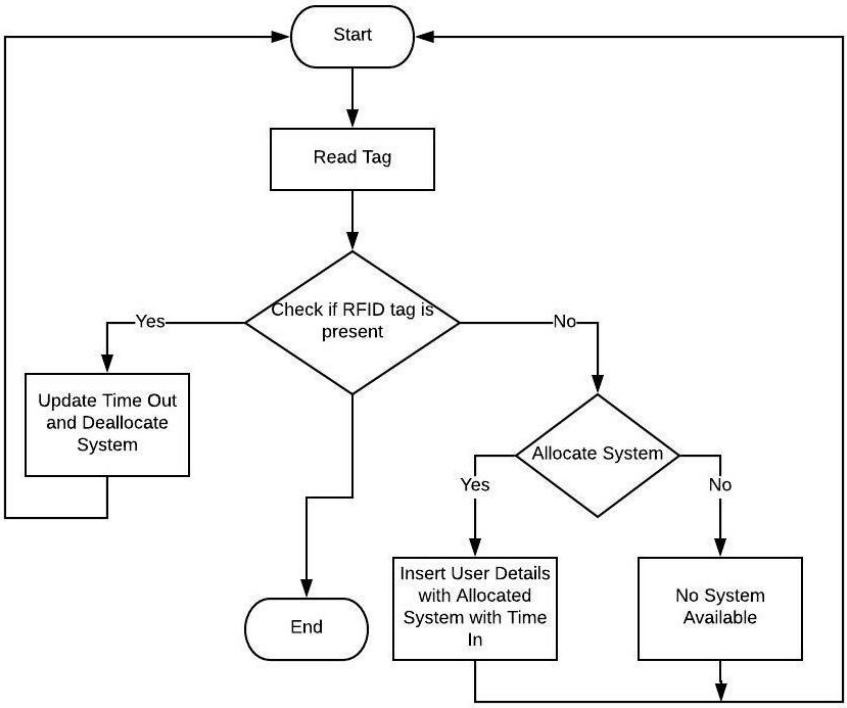



**Figure 3.4:** Activity Diagram

## 3.5 DATABASE DESIGN

### Table schemas:

| **Login** | | | |
|---|---|---|---|
| **Description** | Admin Authentication | | |
| **Attribute** | **Description** | **Type** | **Examples of values** |
| Login ID | Admin ID | Varchar | admin |
| Password | Admin Password | Varchar | admin123 |
| **Primary Key** | Login ID | | |

| **Lab Detail** | | | |
|---|---|---|---|
| **Description** | It stores the details of each lab that provides the total number of system which are working and not working. | | |
| **Attribute** | **Description** | **Type** | **Examples of values** |
| ID | ID of the lab | Varchar | L3 |
| Lab Name | Name of the lab | String | PG Lab |
| No of working system | The total number of systems working in this lab | Integer | 28 |
| No of non-working system | The total number of non-systems working in this lab | Integer | 1 |
| **Primary Key** | ID | | |
| **Foreign Keys** | - | | |

| **System Detail** | | | |
|---|---|---|---|
| **Description** | System Details | | |
| **Attribute** | **Description** | **Type** | **Examples of values** |
| SNo | Serial Number | Integer(auto) | 1 |

| | | | |
|---|---|---|---|
| System Name | Name of the each PC | Varchar | PGlab01 |
| Status | Status of system<br>0: available 1: in-use | Integer | 1 |
| **Primary Key** | SNo | | |
| **Foreign Keys** | - | | |

| **Lab User Detail** | | | |
|---|---|---|---|
| **Description** | It contains the details of the users who are currently using the lab also the system they use. | | |
| **Attribute** | **Description** | **Type** | **Examples of values** |
| SNo | Serial Number | Integer(Auto) | 1 |
| UID | Unique ID of each student ID card | Varchar | D365F7B8 |
| Name | It is the name(register number) of the student | Varchar | 17/PCSA/501 |
| System Allocated | It is the name of the system allocated for the student | Varchar | PGlab01 |
| Time In | It stores the current time of the system of user entered in | Datetime | 12:00 PM |
| Time Out | It stores the current time of the system of user entered out | Datetime | 01:30 PM |
| Date | It stores the current date of the system | Date | 12/04/2018 |
| Tag Seen | Status of tag<br>0: entered 1: exit | Integer | 0 |
| **Primary Key** | SNo | | |
| **Foreign Keys** | - | | |

## 3.6 MODEL DESIGN

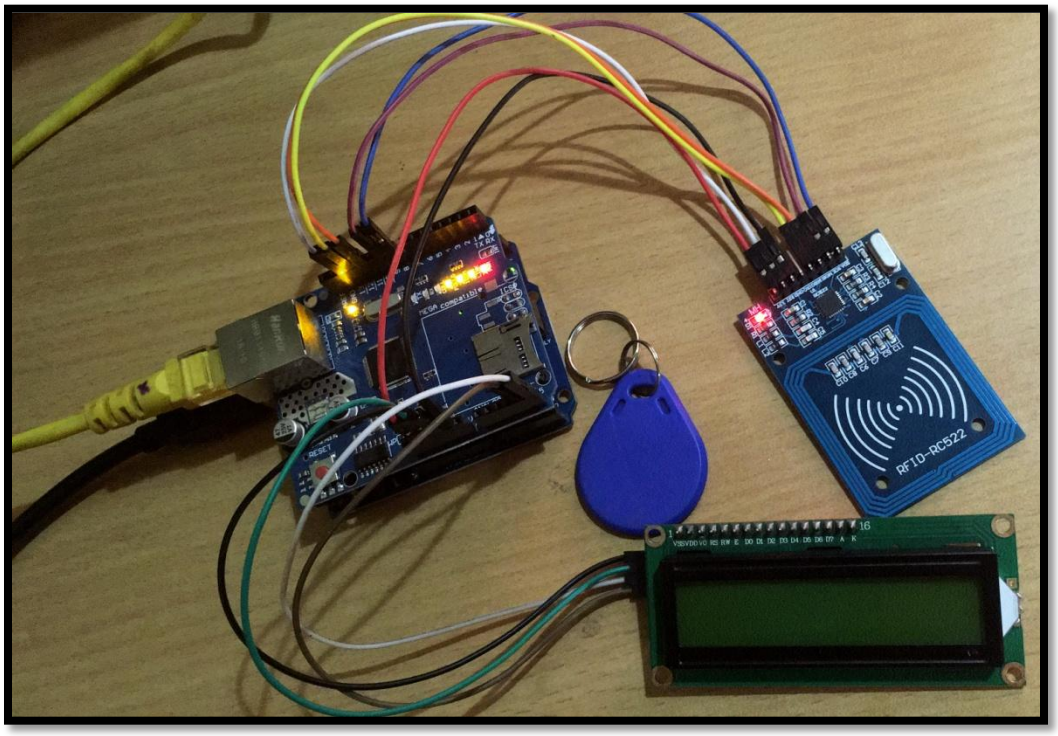
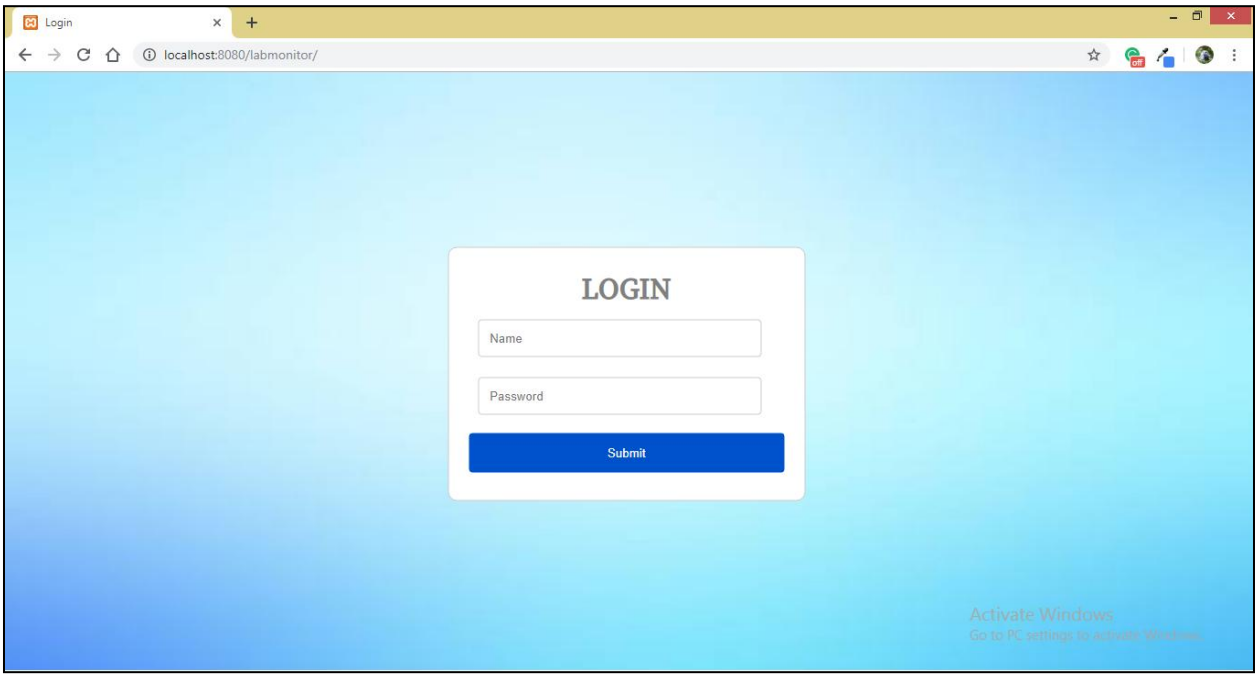



**Figure 3.5:** Model Design

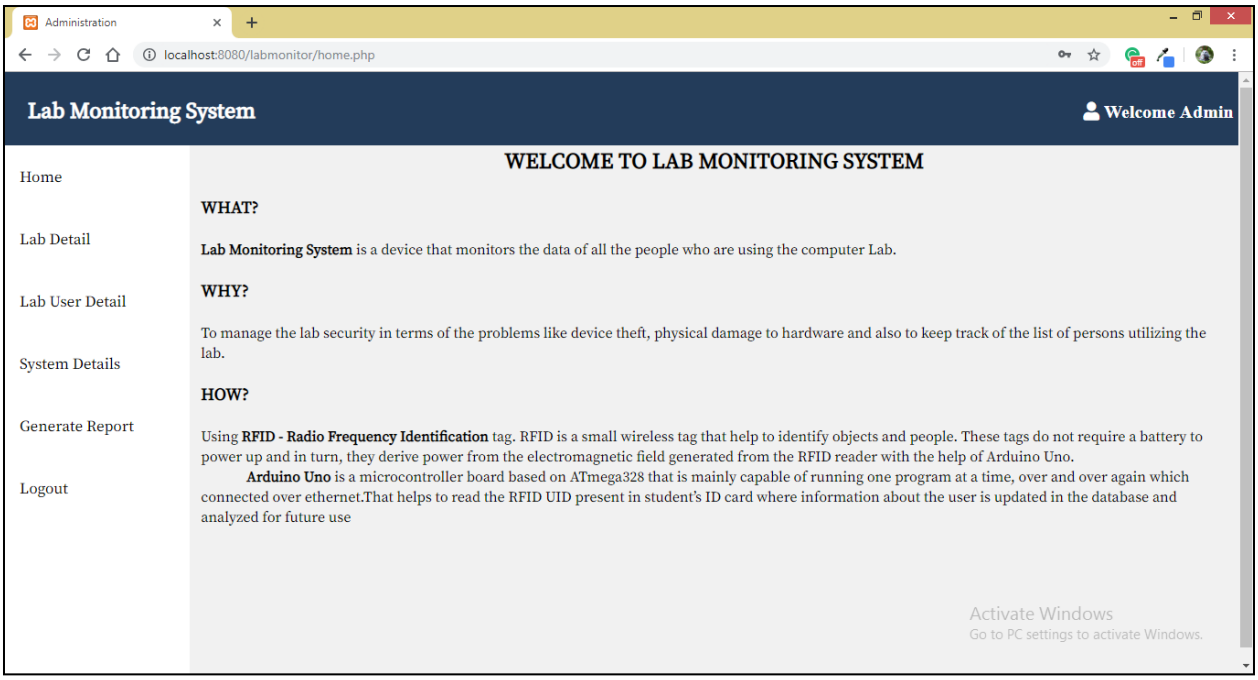
# 4. GRAPHICAL USER INTERFACES

## Admin Modules

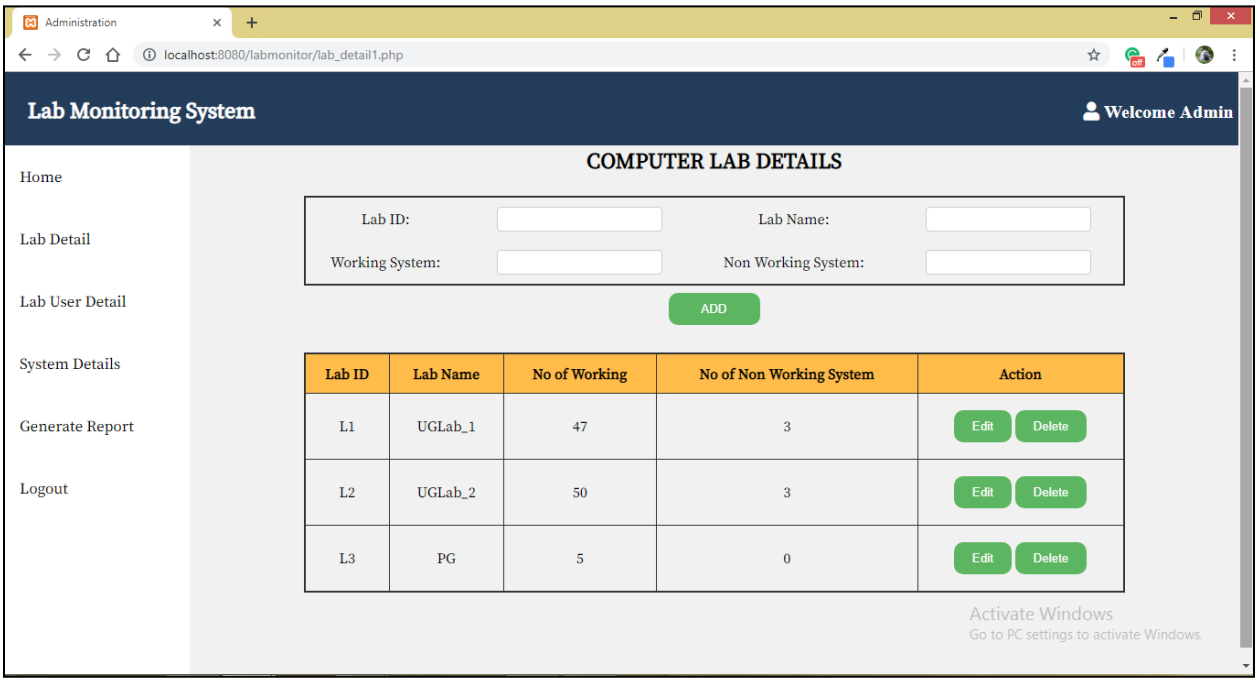
### Login page:



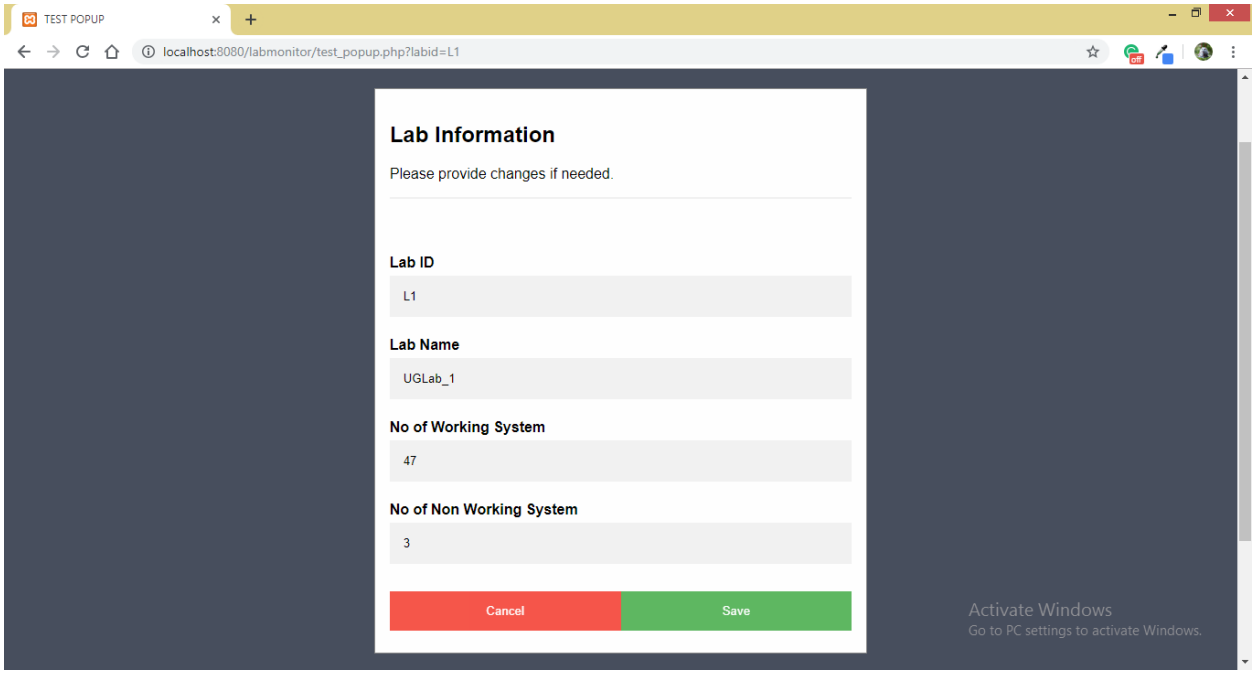

### Home Page:



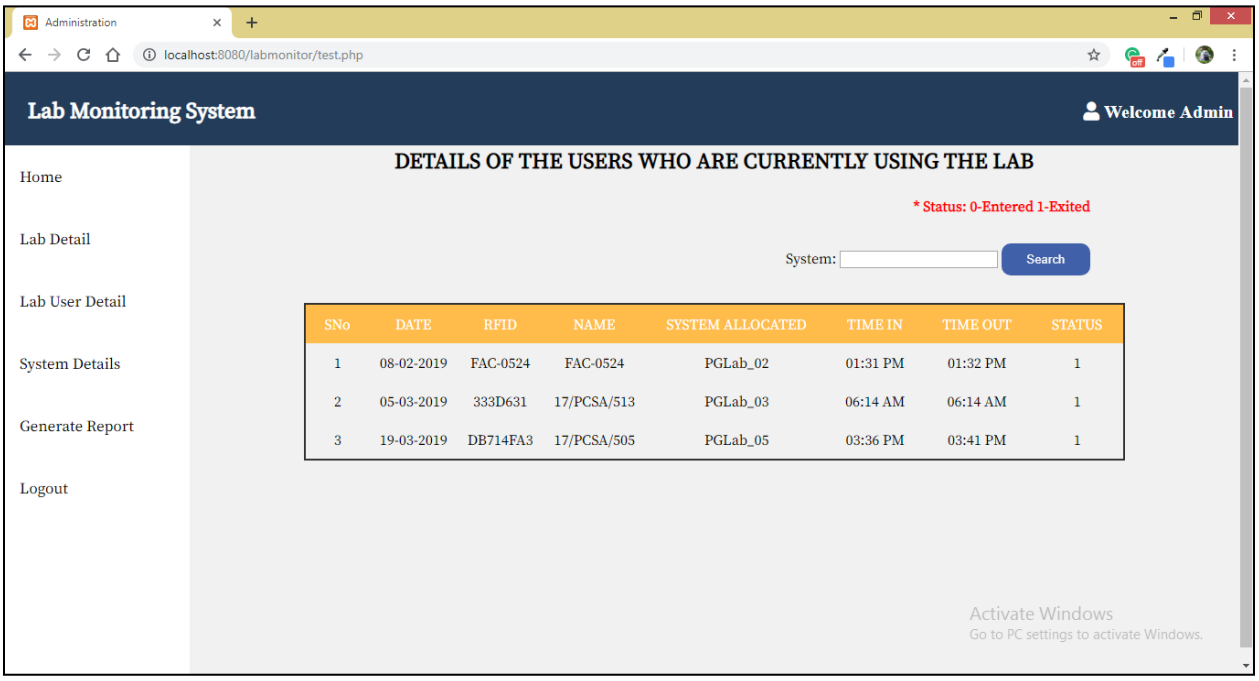
# Lab Detail Page:



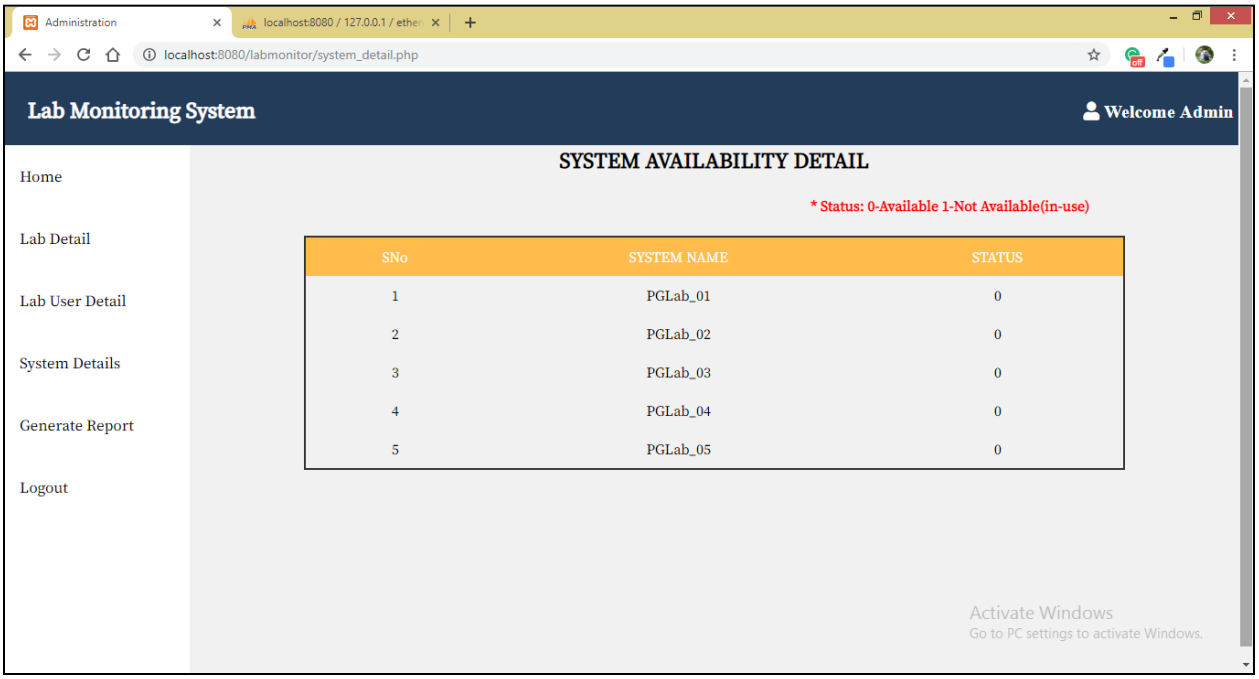

# Edit Lab Detail Page:



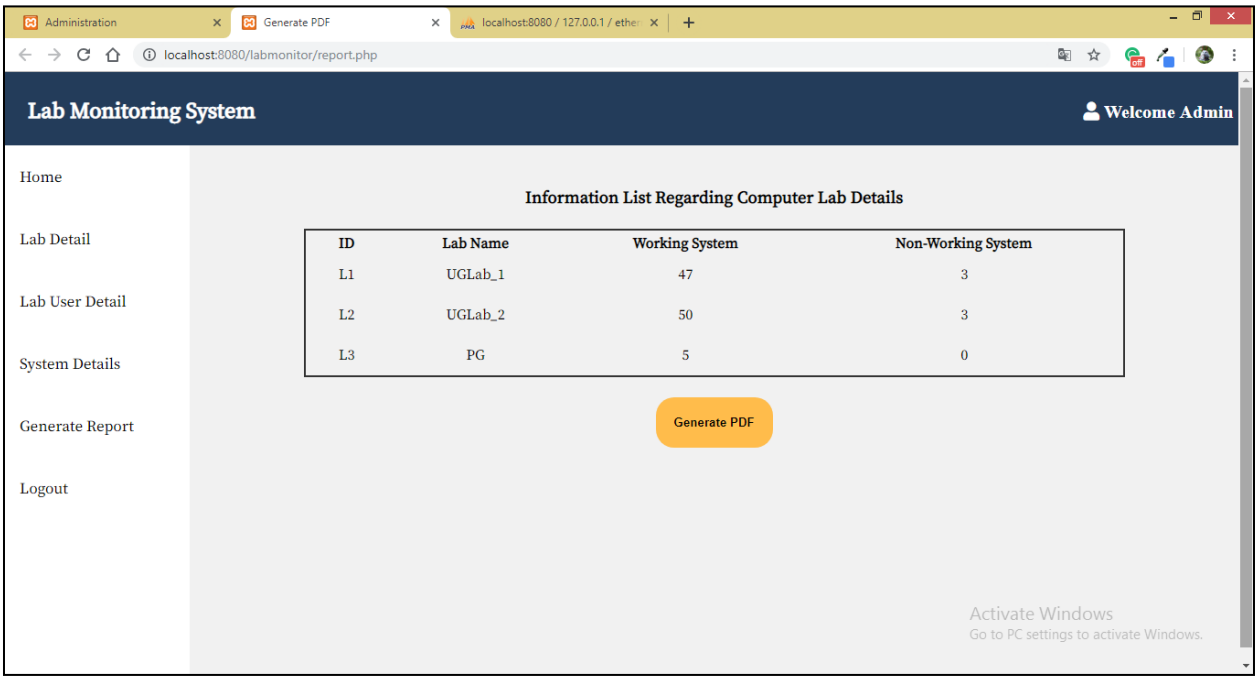
## Lab User Detail Page:



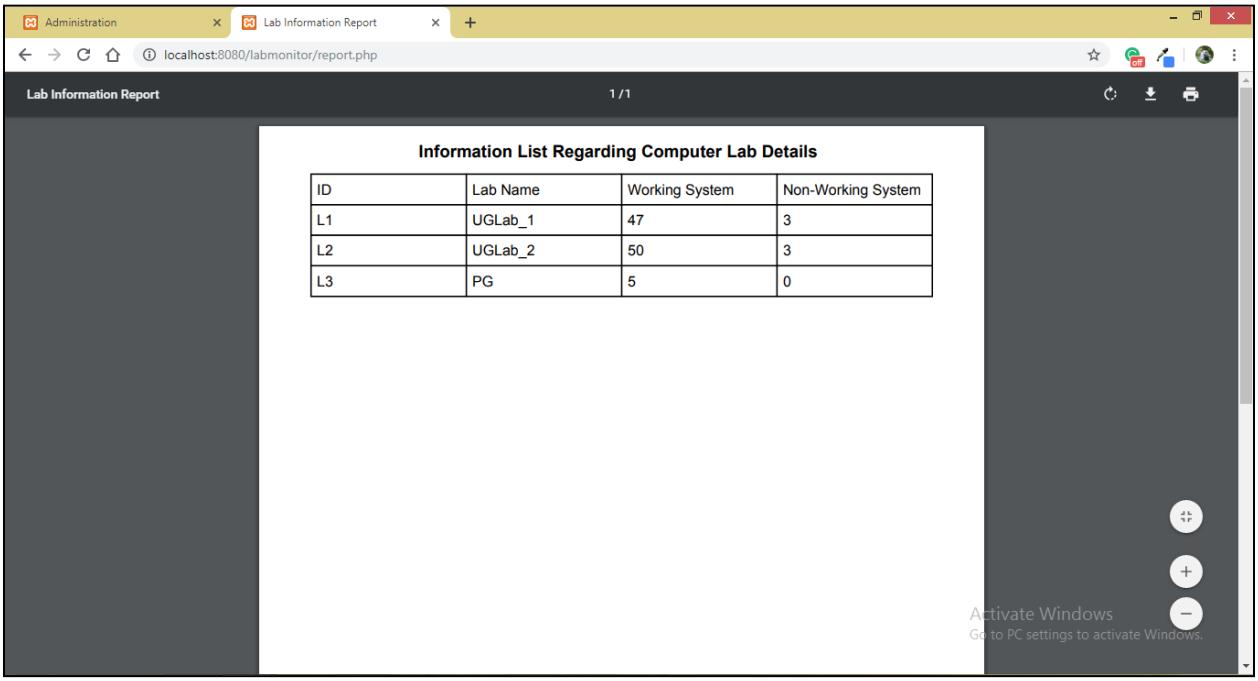

## System Detail Page:



## Report Generator Page:

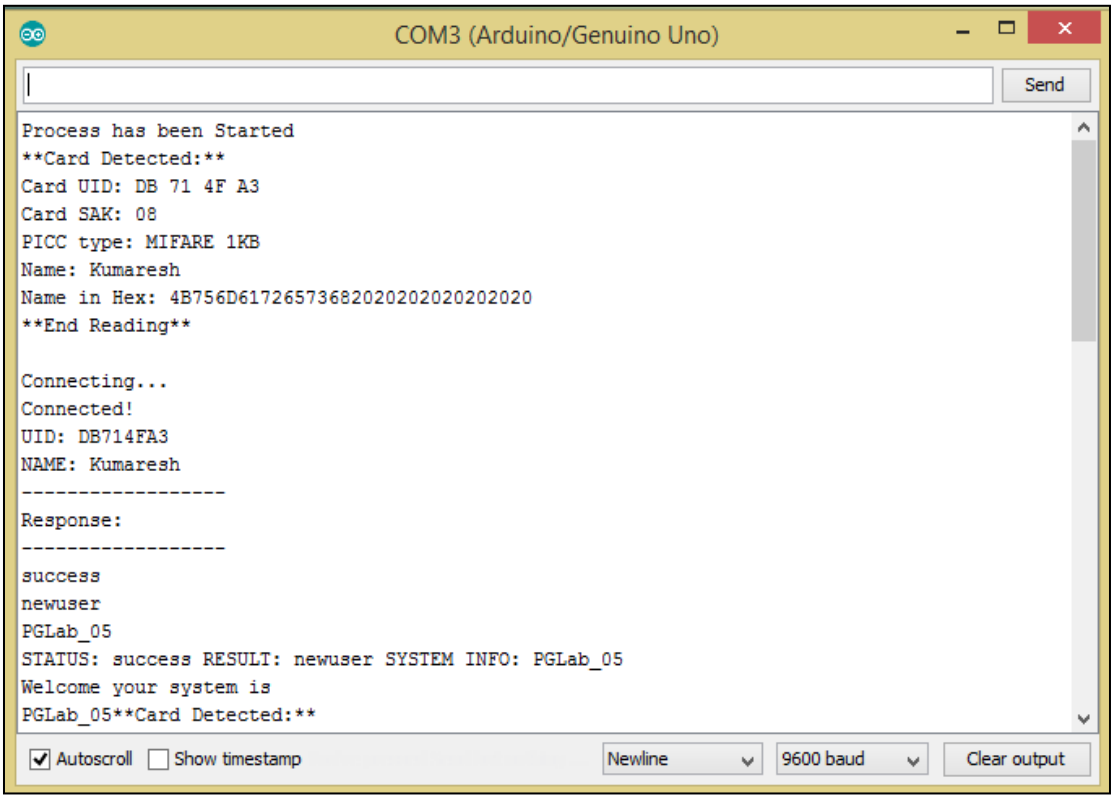



## PDF Page View:

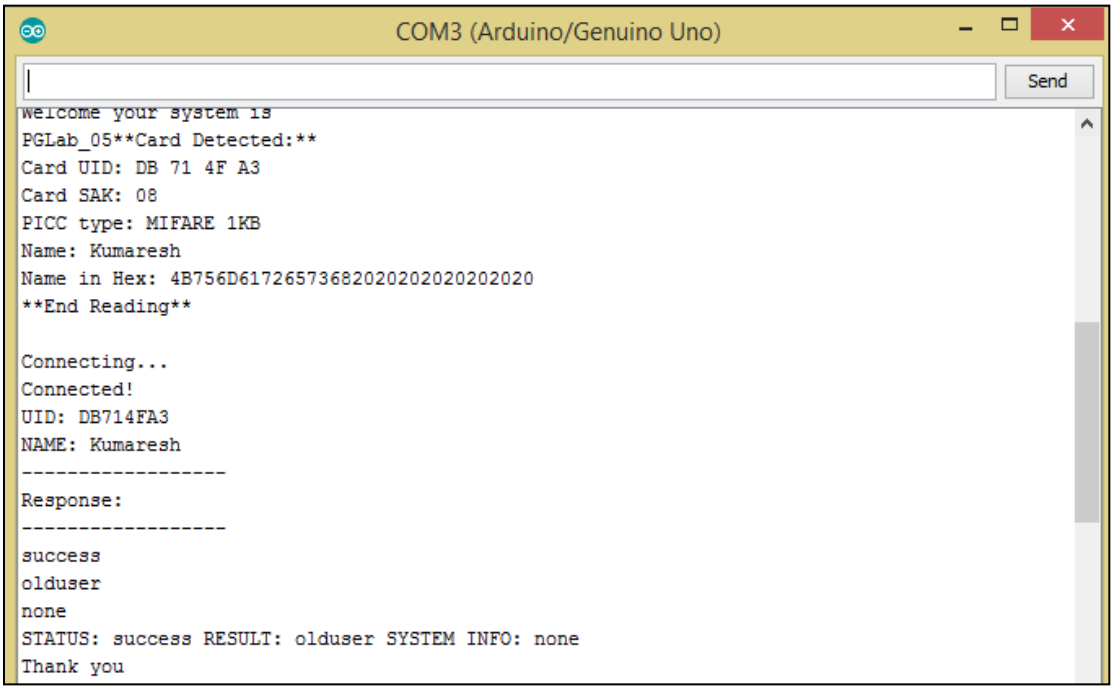


## Ardunio IDE Serial Monitor:

## Entry of new user in computer lab and system allocation process -




## Entry of old user in lab and system de-allocation process -



# 5. TESTCASE SPECIFICATION

**Screen Name:** Admin Login Page

**Test case id:** 1

**Test Name:** Verify Authentication

**Purpose:** Ensure that user can log into the application.

| Step | User Action | Expected Result | Actual Result | Status (Pass/Fail) |
|---|---|---|---|---|
| 1 | Launch the application | The Login screen appears | The Login screen appears | Pass |
| 2 | Type in **admin** as the username and **admin123** as the password | Valid user – The Home screen appears | The Home screen appears | Pass |
| 3 | Type in **sample** as the username and **123** as the password | Invalid Username and Password | Invalid Username and Password | Pass |

**Screen Name:** Hardware Check

**Test case id:** 2

**Test Name:** Verify User System Allocation

**Purpose:** Allocate system to user who enters the lab and show the information in LCD

| Step | User Action | Expected Result | Actual Result | Status (Pass/Fail) |
|---|---|---|---|---|
| 1 | Launch the application | LCD Display – Process Started | Process Started | Pass |
| 2 | Show the Tag to RFID Reader.<br>New User – Allocate system which is free | LCD Display -<br>Welcome User<br>System Allocated:<br>PGLab_01 | Welcome User<br>System Allocated:<br>PGLab_01 | Pass |
| 3 | Show the Tag to RFID Reader. Old User – Deallocate system which was allocated to that user. | LCD Display – Thank You | Thank You | Pass |
| 4 | Show the Tag to RFID Reader. If no system available. | Sorry No System Available | Sorry No System Available | Pass |

# 6. CODING

## a) Hardware Code:

**//Includes all packages required for each hardware**

```
#include <SPI.h>
#include <Ethernet.h>
#include <MFRC522.h>
#include <ArduinoJson.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
```

**//Defining the Pins for MFRC522 that process the data channel**

```
#define RST_PIN 9
#define SS_PIN 8
```

**//Creating instance of each header**

```
MFRC522 mfrc522(SS_PIN, RST_PIN); //create MFRC522 instance
```

**// Set the LCD address to 0x3F for a 16 chars and 2 line display**

```
LiquidCrystal_I2C lcd(0x3F, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

EthernetClient client;
```

**//Variable used in program**

```
byte buffer2[30];       //reads name
int readsuccess;        //to check the presence of tag
//String uid="DB714FA3";    //testing variable
//String uname="sai";       //testing variable
char serv[] = "172.16.1.110";          //IP Address of the server


void setup() {
```

**//Initialization**

```
  Serial.begin(9600);
  SPI.begin();        // Init SPI bus
  mfrc522.PCD_Init();       // Init MFRC522 card
  //mfrc522.PCD_DumpVersionToSerial(); //show details of card reader module
  while (!Serial) continue;
```

```
  // Initialize Ethernet library
  byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
  if (!Ethernet.begin(mac)) {
    Serial.println(F("Failed to configure Ethernet"));
    return;
  }

  //------------------------------------------------------------------

  Serial.println(F("Process has been Started")); //shows in serial

  //shows in LCD display
  lcd.begin(16,2);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Process Has Been");
  lcd.setCursor(0,1);
  lcd.print("Started");
  delay(2000);
  lcd.clear();
  lcd.print("Show Your Tag");
}

void loop() {
  readsuccess=check_for_rfid_tag(); //method return true(1) if tag is present and if not false(0)  if(readsuccess){
      // Prepare key - all keys are set to FFFFFFFFFFFFh at chip delivery from the factory.
  MFRC522::MIFARE_Key key;
  for (byte i = 0; i < 6; i++) key.keyByte[i] = 0xFF;

  byte block;
  byte len;
  MFRC522::StatusCode status;

  Serial.println(F("**Card Detected:**"));

  mfrc522.PICC_DumpDetailsToSerial(&(mfrc522.uid)); //dump some details about the card

  //mfrc522.PICC_DumpToSerial(&(mfrc522.uid));      //uncomment this to see all blocks in hex

  Serial.print(F("Name: "));
  block = 1;
```

```
  status = mfrc522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_A, 1,
&key, &(mfrc522.uid)); //line 834
  if (status != MFRC522::STATUS_OK) {
    Serial.print(F("Authentication failed: "));
    Serial.println(mfrc522.GetStatusCodeName(status));
    return;
  }

  status = mfrc522.MIFARE_Read(block, buffer2, &len);
  if (status != MFRC522::STATUS_OK) {
    Serial.print(F("Reading failed: "));
    Serial.println(mfrc522.GetStatusCodeName(status));
    return;
  }

  //Print Name
  String myname;
  for (uint8_t i = 0; i < 16; i++) {
    Serial.write(buffer2[i]);
    //readcard[i]=buffer2[i];
  }
  Serial.println(" ");
  Serial.print("Name in Hex: ");
  for (uint8_t i = 0; i < 16; i++) {
    Serial.print(buffer2[i],HEX);
  }
  Serial.println(F("\n**End Reading**\n"));

  delay(1000); //change value if you want to read cards faster

  mfrc522.PICC_HaltA();
  mfrc522.PCD_StopCrypto1();
  //performing http request
  delay(1000);

  Serial.println(F("Connecting..."));

  //Creating a HTTP Request Connection

  client.setTimeout(10000);
  if (!client.connect(serv, 8080)) {
    Serial.println(F("Connection failed"));
    return;
  }
  Serial.println(F("Connected!"));
```

**// Send HTTP request**

```
client.print(String("GET ") + "/labmonitor/arduino/checkuser.php?");
  client.print("rfid=");
      Serial.print("UID: ");
      for(int i=0;i<4;i++)
      {
        Serial.print(mfrc522.uid.uidByte[i],HEX);
        client.print(mfrc522.uid.uidByte[i],HEX);
      }
      //client.print(uid);
      client.print("&name=");
      Serial.println(" ");
      Serial.print("NAME: ");
      for (uint8_t i = 0; i < 16; i++) {
         Serial.write(buffer2[i]);
         client.print(buffer2[i],HEX);
        }
      //client.print(uname);

  client.print(" HTTP/1.1\r\nHost: " );
  client.print(serv);
  client.print("\r\nConnection: close\r\n\r\n"); //GET request for server response.
if (client.println() == 0) {
  Serial.println(F("Failed to send request"));
  return;
}
```

**// Check HTTP status**

```
char httpstatus[32] = {0};
client.readBytesUntil('\r', httpstatus, sizeof(httpstatus));
if (strcmp(httpstatus, "HTTP/1.1 200 OK") != 0) {
  Serial.print(F("Unexpected response: "));
  Serial.println(httpstatus);
  return;
}
```

**// Skip HTTP headers**

```
char endOfHeaders[] = "\r\n\r\n";
if (!client.find(endOfHeaders)) {
  Serial.println(F("Invalid response"));
  return;
}
```

```
// Allocate JsonBuffer
  const size_t capacity = JSON_OBJECT_SIZE(3) + JSON_ARRAY_SIZE(2) + 60;
DynamicJsonBuffer jsonBuffer(capacity);

// Parse JSON object
JsonObject& root = jsonBuffer.parseObject(client);
if (!root.success()) {
  Serial.println(F("Parsing failed!"));
  return;
}

// Extract values
Serial.println();
Serial.println("-------------------");
Serial.println(F("Response:"));
Serial.println("-------------------");
Serial.println(root["status"].as<char*>());
Serial.println(root["result"].as<char*>());
Serial.println(root["system_info"].as<char*>());
String s=root["status"].as<char*>();
String r=root["result"].as<char*>();
String sinfo=root["system_info"].as<char*>();
Serial.print("STATUS: ");
Serial.print(s);
Serial.print(" RESULT: ");
Serial.print(r);
Serial.print(" SYSTEM INFO: ");
Serial.print(sinfo);
Serial.println();
if(s.equals("success"))
  {
    if(r.equals("newuser"))
    {
      Serial.println("Welcome your system is");
      Serial.print(sinfo);
      lcd.clear();
      lcd.print("Welcome User");
      delay(2000);
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("System Allocated:");
      lcd.setCursor(0,1);
      //delay(1000);
      //lcd.clear();
      lcd.print(sinfo);
      delay(3000);
```

```
      }
      else
      {
        Serial.println("Thank you");
        lcd.clear();
        lcd.print("Thank you");
        delay(1000);
      }
    }
    else
    {
      Serial.println("Sorry no system is free");
      lcd.clear();
      lcd.print("Welcome User");
      delay(2000);
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print("Sorry No System");
      lcd.setCursor(0,1);
      lcd.print("is Free!!");
      delay(2000);
    }

  // Disconnect

  client.stop();
  lcd.clear();
  lcd.print("Show Your Tag");
  }
}

//Method that checks for the presence of the tag

int check_for_rfid_tag(){
  // Look for new cards
  if ( ! mfrc522.PICC_IsNewCardPresent()) {
    return 0;
  }

  if ( ! mfrc522.PICC_ReadCardSerial()) {
    return 0;
  }
  return 1;
}
```

## b) Software Code:

### Arduino Part

### connection.php

```
<?php

$username = "root";
$pass = "";
$host = "localhost";
$db_name = "ethernet";
$con = mysqli_connect ($host, $username, $pass,$db_name);

// Check connection
if (mysqli_connect_errno())
{
 echo "Failed to connect to MySQL: " . mysqli_connect_error();
}
?>
```

### checkuser.php

```
<?php

include("includes/connection.php");

$msg=0;
$uid=$_GET['rfid'];
$name=hex2str($_GET['name']);
$data=null; //check if it is a new user or old user

$sql_select = "SELECT sys_allocated, tag_seen FROM lab_checkin WHERE
rfid='$uid'";
if($result=mysqli_query($con,$sql_select))
{
    while($row = mysqli_fetch_array($result))
    {
            if($row['tag_seen']==0)
            {
                    //check out process
                    $data="olduser";
                    $msg = 2;
                    //echo "<b>Status</b> : old user<br>";
                    $sysname = $row['sys_allocated'];
```

```
                        //update the given allocated system -> 0
                        $conn = mysql_connect("localhost","root","");
                        if (!$conn)
                          {
                          die('Could not connect: ' . mysql_error());
                          }
                        mysql_select_db("ethernet", $conn);

                        //echo "<b>Updating System Allocated</b><br>";
                        //echo "User Allocated System is ".$sysname."<br>";

                        mysql_query("UPDATE system_detail SET status = '0' WHERE
sys_name = '$sysname'");

                        //echo "Updating Tag Seen<br>";
                        //updating tag_seen -> 1
                        $conn = mysql_connect("localhost","root","");
                        if (!$conn)
                          {
                          die('Could not connect: ' . mysql_error());
                          }
                        mysql_select_db("ethernet", $conn);

                        mysql_query("UPDATE lab_checkin SET tag_seen = '1' WHERE
rfid = '$uid'");
                        mysql_query("UPDATE lab_checkin SET timeout = NOW()
WHERE rfid = '$uid'");
                        mysql_close($conn);
                        //echo "Successfully updated - tag_seen is checked out<br>";
                        break;
                }
    }
}
else
{
echo "error is ".mysqli_error($con);
}

if($data==null)
{
    $msg = 1;
    //echo "<b>Status</b> : new user<br>";
    //include ('getname.php');
    include ('allocate_system.php');
    $result = $_SESSION['result'];
```

```
    if($result == "success")
    {
            $con = mysql_connect("localhost","root","");
            if (!$con)
            {
                    die('Could not connect: ' . mysql_error());
            }
            mysql_select_db("ethernet", $con);
            mysql_query("INSERT INTO lab_checkin (rfid,name,sys_allocated)
VALUES ('$uid', '$name','$sysname')");
            //echo "Successfully Inserted<br>";

            mysql_close($con);
    }
}
```

**//method that converts hex values into ascii or alphabetic values**

```
function hex2str($hex)
{
   $str = '';
   for($i=0;$i<strlen($hex);$i+=2) $str .= chr(hexdec(substr($hex,$i,2)));
   return $str;
   //echo "My Name is ".$str;
}
```

**//sending response message to Arduino**

```
if($msg==1)
{
    $status="success";
    $output="newuser";
    $system_info=$sysname;
    //$extra="labmonitor/arduino/response.php?status=".$status."&output=".$output."
&system_info".$system_info;
    //header( 'Location: $extra' );
    echo "{\"status\": \"$status\",\"result\": \"$output\", \"system_info\":
\"$system_info\"}";

}
else if($msg==2)
{
    $status="success";
    $output="olduser";
    $system_info="none";
    //$extra="labmonitor/arduino/response.php?status=".$status."&output=".$output."
&system_info".$system_info;
```

```
    //header( 'Location: $extra' );
    echo "{\"status\": \"$status\",\"result\": \"$output\", \"system_info\":
\"$system_info\"}";
}
else
{
    $status="fail";
    $output="newuser";
    $system_info="no system available";
    //$extra="labmonitor/aduino/response.php?status=".$status."&output=".$output."
&system_info".$system_info;
    //header( 'Location: $extra' );
    echo "{\"status\": \"$status\",\"result\": \"$output\", \"system_info\":
\"$system_info\"}";

}

?>
```

## allocate_system.php

```
<?php
include("includes/connection.php");

$sysname=" ";

$sql_select = "SELECT * FROM system_detail";
if($result=mysqli_query($con,$sql_select))
{
    while($row = mysqli_fetch_array($result))
    {
            if($row['status']==0)
            {
                    $sysname = $row['sys_name'];
                    //echo "My Free System is ".$sysname."<br>";
                    $sql_update = "UPDATE system_detail SET status = '1' WHERE
sys_name = '$sysname'";
                    if(mysqli_query($con,$sql_update))
                    {
                    //echo "Successfully updated<br>";
                    session_start();

                    $result = "success";
                    //$system_info = $sysname;

                    $_SESSION['result'] = $result;
```

```
                    mysqli_close($con);
                    }
                    else
                    {
                    echo "error is ".mysqli_error($con );
                    }
                    break;
                }
    }
}
else
{
echo "error is ".mysqli_error($con );
}
if($sysname == " ")
{
    echo "Sorry!! No System Available";
    session_start();
    $msg = 3;
    $_SESSION['result'] = $result;

}
?>
```

## Admin Part

## labdetail.php

```
<?php session_start();
error_reporting(0);
if(isset($_POST['add'])) {
  $id=$_POST['labid'];
  $name=$_POST['labname'];
  $working_sys=$_POST['working_sys'];
  $non_working_sys=$_POST['non_working_sys'];
  include("includes/connection.php");
  $query="INSERT INTO lab_detail
(id,lab_name,num_working_sys,num_non_working_sys) VALUES
('$id','$name','$working_sys','$non_working_sys')";
  if(mysqli_query($con,$query))
  {
    //echo "Affected rows: " . mysqli_affected_rows($con);
    if(mysqli_affected_rows($con)>0)
    {
      //echo "<br><div class='successmsg'>SUCCESSFULLY
REGISTERED!!</div>";
      //$_SESSION['errmsg']="Successfully Added!!";
    }
```

```
  }
  else
  {
    //echo("Error description: " . mysqli_error($con));
    $_SESSION['errmsg']="Error description: " . mysqli_error($con);
  }
}
if(isset($_GET['del']))
{
  $labid = $_GET['labid'];
  include("includes/connection.php");
  $delete = "DELETE FROM lab_detail WHERE id='$labid'";
  if(mysqli_query($con,$delete)){
    //echo "Successfully Deleted!!";
    echo '<script>alert("Successfully Deleted!!");</script>';
  }
  else{
    echo "Error updating record: " . mysqli_error($con);
    echo '<script>alert("Error in connection");</script>';
  }
}
?>
<html>
<head>
  <title>Administration</title>
  <link rel="stylesheet" href="https://use.typekit.net/wqe7gdm.css">
  <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.7.2/css/all.css" integrity="sha384-
fnmOCqbTlWIlj8LyTjo7mOUStjsKC4pOpQbqyi7RrhN7udi9RwhKkMHpvLbHG9S
r" crossorigin="anonymous">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <link rel="stylesheet" href="lab_detail_style.css">
  <style type="text/css">

  </style>
</head>
<body>
  <header><h2>Lab Monitoring System <i class="fas fa-user" style="padding-left:
900px;
    font-size: 20px;">   Welcome Admin</i></h2></header>
  <section class="menu">
    <nav>
      <a class="sidebar" href="home.php" class="menu-items">Home</a>
      <a class="sidebar" href="lab_detail1.php" class="menu-items">Lab Detail</a>
      <!--<a href="lab_user_detail.php" class="menu-items">Lab User Detail</a>-->
```

```
    <a class="sidebar" href="test.php" class="menu-items">Lab User Detail</a>
    <a class="sidebar" href="system_detail.php" class="menu-items">System
Details</a>
    <a class="sidebar" href="report.php" class="menu-items"
target="_blank">Generate Report</a>
    <a class="sidebar" href="index.html" class="menu-items">Logout</a>
   </nav>
  </section>

  <section class="menu-content">
   <div align="center">
    <h1>Computer Lab Details</h1><br>
    <form method="post">
     <table border="0">
      <tr>
       <td>Lab ID:</td>
       <td><input type="text" name="labid" required></td>
       <td>Lab Name:</td>
       <td><input type="text" name="labname" required></td>
      </tr>
      <tr>
       <td>Working System:</td>
       <td><input type="text" name="working_sys" pattern="[0-9]+"
required></td>
        <td>Non Working System:</td>
        <td><input type="text" name="non_working_sys" pattern="[0-9]+"
required></td>
      </tr>
     </table>
     <input type="submit" name="add" id="add" value="ADD" style="width:
100px;"><br>
     <span class="errmsg" >
      <b><?php echo htmlentities($_SESSION['errmsg']); ?>
      <?php echo htmlentities($_SESSION['errmsg']="");?></b>
      </span>
    </form>
    <!--<div class="result" style="color:red;padding: 10px;"></div>-->
    <?php
     include("includes/connection.php");

     $select =mysqli_query($con,"SELECT * FROM lab_detail");
     ?>
     <br><table border="1" id="user_table">
     <tr>
     <th>Lab ID</th>
     <th>Lab Name</th>
```

```
        <th>No of Working </th>
        <th>No of Non Working System</th>
        <th>Action</th>
        </tr>
        <?php
        while ($row=mysqli_fetch_array($select))
        {
         ?>
         <tr>
          <td><?php echo $row['id'];?></td>
          <td><?php echo $row['lab_name'];?></td>
          <td><?php echo $row['num_working_sys'];?></td>
          <td><?php echo $row['num_non_working_sys'];?></td>
          <td>
           <button style="width:auto;" id="<?php echo $row['id'];?>"
            onclick="edit(this.id);" value="<?php echo $row['id'];?>">Edit</button>
            <!--<button style="width:auto;" id="<?php echo $row['id'];?>"
            onclick="Delete(this.id);" value="<?php echo
$row['id'];?>">Delete</button>-->
            <a class=".delete" href="lab_detail1.php?labid=<?php echo
$row['id']?>&del=delete" onClick="return confirm('Are you sure you want to
delete?')">
              <button style="width:auto;">Delete</button>
            </a>
          </td>
         </tr>
         <?php
        }
        ?>
        </table>
    </div>
  </section>
  <!--<footer>
    <h3 class="footer">Copyright &copy; 2019, MK Preethi of Stella Maris College II
MSc IT</h3>
  </footer>-->
  <script>
function edit(clicked_id)
{
    window.location.href = "test_popup.php?labid="+clicked_id;
}
</script>
</body>
</html>
```

## edit_labdetail.php

```
<?php
session_start();
error_reporting(0);
if(isset($_GET['save']))
{
    $id=$_GET['labid'];
    $name=$_GET['labname'];
    $num_working_sys=$_GET['num_working_sys'];
    $num_non_working_sys=$_GET['num_non_working_sys'];

    include 'includes/connection.php';
    $update = "UPDATE lab_detail SET id='$id' , lab_name='$name' ,
num_working_sys='$num_working_sys',
num_non_working_sys='$num_non_working_sys' WHERE id='$id'";
    if (mysqli_query($con, $update)) {
    //echo "Successfully Updated!!";
    $_SESSION['errmsg']="Successfully Updated!!";
  } else {
    echo "Error updating record: " . mysqli_error($con);
    $_SESSION['errmsg']="Error updating record: " . mysqli_error($con);
  }
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>TEST POPUP</title>
    <link rel="stylesheet" type="text/css" href="popup.css">
    <style type="text/css">
            .errmsg{
                    color: red;
                    padding: 5px;
            }
    </style>

</head>
<body>
<div id="id01" class="modal">
   <span onclick="exit();" class="close" title="Close">&times;</span>
   <form class="modal-content" method="GET">
    <div class="container">
      <h2>Lab Information</h2>
      <p>Please provide changes if needed.</p>
```

```
<hr>
<span class="errmsg" >
<?php echo htmlentities($_SESSION['errmsg']); ?>
<?php echo htmlentities($_SESSION['errmsg']="");?>
</span>
<?php
if(isset($_GET['labid']))
{
    $id=$_GET['labid'];
  include('includes/connection.php');
  $query="SELECT * FROM lab_detail WHERE id='$id'";
  if($res=mysqli_query($con,$query))
  {
   while($row=mysqli_fetch_array($res))
   {
     //echo $row['name'] ."<br>";
  ?>
  <br><br><label><b>Lab ID</b></label>
  <input type="text" id="labid" name="labid" value="<?php echo $row['id']
?>" readonly>
  <label><b>Lab Name</b></label>
  <input type="text" id="labname" name="labname" value="<?php echo
$row['lab_name'] ?>">
  <label><b>No of Working System</b></label>
  <input type="text" id="num_working_sys" name="num_working_sys"
value="<?php echo $row['num_working_sys'] ?>">
  <label><b>No of Non Working System</b></label>
  <input type="text" id="num_non_working_sys"
name="num_non_working_sys" value="<?php echo $row['num_non_working_sys']
?>">

 <?php    }
  }
  else
  {
   echo("Error description: " . mysqli_error($res));
  }
 } ?>

<div class="clearfix">
 <button type="button" onclick="exit();" class="cancelbtn">Cancel</button>
 <button type="submit" class="signupbtn" name="save" value="save"
onclick="save();">Save</button>
</div>
</div>
</form>
```

```
  </div>

<script type="text/javascript">
function PopUp(){
     document.getElementById('id01').style.display='block';
}
window.onload = function () {
   setTimeout(function () {
      PopUp('show');
   },500);
}

function exit(){
     document.getElementById('id01').style.display='none';
     window.location.href = "lab_detail1.php";
}

function save(){
     document.getElementById('id02').style.display='block';
}
</script>
</body>
</html>
```

## labuserdetail.php

```
<?php
$url=$_SERVER['REQUEST_URI'];
header("Refresh: 10; URL=$url"); // Refresh the webpage every 5 seconds
?>
<html>
<head>
  <title>Administration</title>
  <link rel="stylesheet" href="https://use.typekit.net/wqe7gdm.css">
  <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.7.2/css/all.css" integrity="sha384-
fnmOCqbTlWIlj8LyTjo7mOUStjsKC4pOpQbqyi7RrhN7udi9RwhKkMHpvLbHG9S
r" crossorigin="anonymous">
  <link rel="stylesheet" href="style.css">
  <style type="text/css">
  input[type=submit]{
  background-color: #2c50a3;
  color: white;
  padding: 10px 20px;
  margin: 8px 0;
  border: none;
```

```
  cursor: pointer;
  width: 10%;
  opacity: 0.9;
  border-radius: 10px;
}
input[type=submit]:hover {
  opacity:1;
}
  </style>
</head>
<body>
  <header><h2>Lab Monitoring System <i class="fas fa-user" style="padding-left:
900px;
    font-size: 20px;">  Welcome Admin</i></h2></header>
  <section class="menu">
    <nav>
      <a href="home.php" class="menu-items">Home</a>
      <a href="lab_detail1.php" class="menu-items">Lab Detail</a>
      <!--<a href="lab_user_detail.php" class="menu-items">Lab User Detail</a>-->
      <a href="test.php" class="menu-items">Lab User Detail</a>
      <a href="system_detail.php" class="menu-items">System Details</a>
      <a href="report.php" class="menu-items" target="_blank">Generate Report</a>
      <a href="index.html" class="menu-items">Logout</a>
    </nav>
  </section>

  <section class="menu-content">
    <div align="center">
      <h1>Details of the users who are currently using the lab</h1><br>
      <p align="right" style="padding-right: 150px;color: red"><b>* Status: 0-Entered
1-Exited</b></p>
      <br>
      <form>
    <!--    <table>
          <tr>
            <td>Date:</td>
            <td><input type="date" name="date" id="date"></td>
            <td>Name:</td>
            <td><input type="text" name="name" id="name"></td>
            <td>System:</td>
            <td><input type="text" name="sysname" id="sysname"></td>
            <td><input type="submit" name="search" value="Search"></td>
          </tr>
        </table>  -->
        <div align="right" style="margin-right: 150px;">
        <span>
```

```
        <label>System:</label>
        <input type="text" name="sysname" id="sysname" autocomplete="off">
        <input type="submit" name="search" value="Search">
      </span>
    </div>
    </form><br>
    <?php
    if(isset($_GET['search']))
    {
     //$date=$_GET['date'];
     //$name=$_GET['name'];
     $sysname=$_GET['sysname'];
     if(/*$date!=null || $name!=null ||*/ $sysname!=null)
     {
      $Tformat="%h:%i %p";
     include("includes/connection.php");
      /*$query="SELECT DATE_FORMAT(mydate,'%m-%d-%Y') mydate,
rfid,name,sys_allocated,TIME_FORMAT(timein,'$Tformat')
timein,TIME_FORMAT(timeout,'$Tformat') timeout,tag_seen FROM lab_checkin
WHERE name='$name' OR sys_allocated='$sysname' OR mydate='$date'";*/
       $query="SELECT DATE_FORMAT(mydate,'%m-%d-%Y') mydate,
rfid,name,sys_allocated,TIME_FORMAT(timein,'$Tformat')
timein,TIME_FORMAT(timeout,'$Tformat') timeout,tag_seen FROM lab_checkin
WHERE sys_allocated='$sysname'";

      if($result = mysqli_query($con, $query))
      {
       if (mysqli_num_rows($result) > 0)
       {?>
        <table border="0" cellspacing="0" cellpadding="4">
          <tr>
            <td class="table_titles">Date</td>
            <td class="table_titles">RFID</td>
            <td class="table_titles">NAME</td>
            <td class="table_titles">SYSTEM ALLOCATED</td>
            <td class="table_titles">TIME IN</td>
            <td class="table_titles">TIME OUT</td>
            </tr>
       <?php while($row = mysqli_fetch_assoc($result))
        {?>

          <tr>
            <td><?php echo $row['mydate']?></td>
            <td><?php echo $row['rfid']?></td>
            <td><?php echo $row['name']?></td>
            <td><?php echo $row['sys_allocated']?></td>
```

```
                <td><?php echo $row['timein']?></td>
                <td><?php echo $row['timeout']?></td>
               </tr>
          <?php
             } ?>
              </table><br>
        <?php  }
        else{
          echo "<script>alert('No Record Found!!');</script>";
          }
        }
        }
       }
       ?>

     <table border="0" cellspacing="0" cellpadding="4">
    <tr>
     <td class="table_titles">SNo</td>
     <td class="table_titles">DATE</td>
     <td class="table_titles">RFID</td>
     <td class="table_titles">NAME</td>
     <td class="table_titles">SYSTEM ALLOCATED</td>
     <td class="table_titles">TIME IN</td>
     <td class="table_titles">TIME OUT</td>
     <td class="table_titles">STATUS</td>
     </tr>
    <?php
     $con = mysql_connect("localhost","root","");
     if (!$con)
       {
       die('Could not connect: ' . mysql_error());
       }

     $count = 1;
     mysql_select_db("ethernet", $con);
     $Tformat="%h:%i %p";
     //$Dformat="%y-%m-%d";
     $result = mysql_query("SELECT DATE_FORMAT(mydate,'%d-%m-%Y')
mydate, rfid,name,sys_allocated,TIME_FORMAT(timein,'$Tformat')
timein,TIME_FORMAT(timeout,'$Tformat') timeout,tag_seen FROM lab_checkin");
     while($row = mysql_fetch_array($result))
     {

        echo "<tr>";
        echo "<td>" . $count . "</td>";
        echo "<td>" . $row['mydate'] . "</td>";
```

```
        echo "<td>" . $row['rfid'] . "</td>";
        echo "<td>" . $row['name'] . "</td>";
        echo "<td>" . $row['sys_allocated'] . "</td>";
        echo "<td>" . $row['timein'] . "</td>";
         echo "<td>" . $row['timeout'] . "</td>";
        echo "<td>" . $row['tag_seen'] . "</td>";
        echo "</tr>";
        $count ++;
    }
    mysql_close($con);
  ?>
 </table>
  </div>
 </section>
 <!--<footer>
  <h3 class="footer">Copyright &copy; 2019, MK Preethi of Stella Maris College II MSc IT</h3>
 </footer>-->

</body>
</html>
```

## report.php

```
<?php
 function fetch_data()
 {
     $output = '';
     $connect = mysqli_connect("localhost", "root", "", "ethernet");
     $sql = "SELECT * FROM lab_detail ORDER BY id ASC";
     $result = mysqli_query($connect, $sql);
     while($row = mysqli_fetch_array($result))
     {
     $output .= '<tr>
                     <td>'.$row["id"].'</td>
                     <td>'.$row["lab_name"].'</td>
                     <td>'.$row["num_working_sys"].'</td>
                     <td>'.$row["num_non_working_sys"].'</td>
                </tr>
                    ';
     }
     return $output;
 }
 if(isset($_POST["create_pdf"]))
 {
     require_once('tcpdf/tcpdf.php');
```

```
    $obj_pdf = new TCPDF('P', PDF_UNIT, PDF_PAGE_FORMAT, true, 'UTF-8',
false);
    $obj_pdf->SetCreator(PDF_CREATOR);
    $obj_pdf->SetTitle("Lab Information Report");
    $obj_pdf->SetHeaderData('', '', PDF_HEADER_TITLE,
PDF_HEADER_STRING);
    $obj_pdf->setHeaderFont(Array(PDF_FONT_NAME_MAIN, '',
PDF_FONT_SIZE_MAIN));
    $obj_pdf->setFooterFont(Array(PDF_FONT_NAME_DATA, '',
PDF_FONT_SIZE_DATA));
    $obj_pdf->SetDefaultMonospacedFont('helvetica');
    $obj_pdf->SetFooterMargin(PDF_MARGIN_FOOTER);
    $obj_pdf->SetMargins(PDF_MARGIN_LEFT, '4', PDF_MARGIN_RIGHT);
    $obj_pdf->setPrintHeader(false);
    $obj_pdf->setPrintFooter(false);
    $obj_pdf->SetAutoPageBreak(TRUE, 10);
    $obj_pdf->SetFont('helvetica', '', 12);
    $obj_pdf->AddPage();
    $content = '';
    $content .= '
    <h3 align="center">Information List Regarding Computer Lab Details</h3><br
/><br />
    <table border="1" cellspacing="0" cellpadding="5">
        <tr>
            <th>ID</th>
            <th>Lab Name</th>
            <th>Working System</th>
            <th>Non-Working System</th>
        </tr>
    ';
    $content .= fetch_data();
    $content .= '</table>';
    $obj_pdf->writeHTML($content);
    $obj_pdf->Output('sample.pdf', 'I');
}
?>
<!DOCTYPE html>
<html>
    <head>
        <title>Generate PDF</title>
        <title>Administration</title>
         <link rel="stylesheet" href="https://use.typekit.net/wqe7gdm.css">
         <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.7.2/css/all.css" integrity="sha384-
fnmOCqbTlWIlj8LyTjo7mOUStjsKC4pOpQbqyi7RrhN7udi9RwhKkMHpvLbHG9S
r" crossorigin="anonymous">
```

```
        <link rel="stylesheet" href="style.css">
        <style type="text/css">
          input[type="submit"]{
            background-color: #ffbc4b;
            border:none;
            border-radius: 20px;
            padding: 20px;
            cursor: pointer;
            font-weight: bold;
          }
        </style>
    </head>
    <body>
      <header><h2>Lab Monitoring System <i class="fas fa-user" style="padding-
left: 900px;
   font-size: 20px;">   Welcome Admin</i></h2></header>
  <section class="menu">
   <nav>
    <a href="home.php" class="menu-items">Home</a>
    <a href="lab_detail1.php" class="menu-items">Lab Detail</a>
    <!--<a href="lab_user_detail.php" class="menu-items">Lab User Detail</a>-->
    <a href="test.php" class="menu-items">Lab User Detail</a>
    <a href="system_detail.php" class="menu-items">System Details</a>
    <a href="report.php" class="menu-items">Generate Report</a>
    <a href="index.html" class="menu-items">Logout</a>
   </nav>
  </section>

  <section class="menu-content">
   <br /><br />
        <div class="container" align="center">
            <h3 align="center">Information List Regarding Computer Lab
Details</h3><br />
            <div class="table-responsive">
                <table class="table table-bordered">
                    <tr>
                        <th>ID</th>
                        <th>Lab Name</th>
                        <th>Working System</th>
                        <th>Non-Working System</th>
                    </tr>
                <?php
                echo fetch_data();
                ?>
                </table>
                <br />
```

```
                <form method="post">
                    <input type="submit" name="create_pdf" class="btn btn-danger"
value="Generate PDF" />
                </form>
            </div>
        </div>
  </section>
     </body>
 </html>
```

# 7. CONCLUSION

Initially there was no security concern available in our college computer laboratory and keeping this in mind automation to lab monitoring system was build using Ardunio with RFID technology in the form of web based monitoring system which sense the term Internet Of things (IOT) using PHP (a server side scripting). It provides easy implementation, installation portability. Ardunio being the low cost helps us to integrate the application at affordable cost.

In this project only tag of every user who enters the lab are sensed which may lead to misuse of the tag and because of this drawback in future a face recognition setup could be done thereby providing at most security in lab.

# 8. REFERENCES


- https://www.cs.toronto.edu/~sme/CSC340F/2005/slides/tutorial-classes_ERDs.pdf
- A Framework For Structural Modeling Of An RFID-Enabled Intelligent Distributed Manufacturing Control System by A.V. Barenji, R.V. Barenji & M. Hashemipour
- https://www.it.iitb.ac.in/~sri/talks/rfid-05.pdf
- https://store.arduino.cc/usa/arduino-uno-rev3
- https://www.mschoeffler.de/2017/02/07/how-to-use-the-rfid-rc522-module-rfid-reader-with-the-arduino-uno/
- https://blog.atlasrfidstore.com/rfid-basics-read-write-tags
- https://www.inc.com/guides/2010/09/how-to-use-rfid-technology.html
- https://www.makerspaces.com/arduino-uno-tutorial-beginners/
- https://www.infineon.com/dgdl/FAQs_RFID_e_0304.pdf?fileId=db3a304412b91b910112baad0cc021bd
- https://electronicshobbyists.com/logging-data-to-database-using-arduino-ethernet-shield/