

A FIELD PROJECT REPORT ON
“QR CODE BASED AUTOMATED ATTENDANCE SYSTEM”

Submitted

In partial fulfillment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE and ENGINEERING

Submitted By

CH.VARSHINI	231FA04B75
K.LAKSHMAN	231FA04C33
D.BHARATH	231FA04C47
A.RISHITHA	231FA04D01

Under the Guidance of
Dr. Nerella Sameera
Assistant Professor, CSE



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING AND INFORMATICS
VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University)
Vadlamudi, Guntur -522213, INDIA.

APRIL-2025.



(Deemed to be University) Estd. u/s 3 of UGC Act 1956

CERTIFICATE

This is to certify that the field project entitled "**QR CODE BASED AUTOMATED ATTENDANCE SYSTEM**" being submitted by Ch. Varshini (231FA04B75), K. Lakshman (231FA04C33), D. Bharath (231FA04C47), and A. Rishitha (231FA04D01) in partial fulfilment of the Bachelor of Technology in the Department of Computer Science and Engineering, Vignan's Foundation for Science Technology & Research (Deemed to be University), Vadlamudi, Guntur District, Andhra Pradesh, India.

This is a Bonafide work carried out by the aforementioned students under my guidance and supervision.

A handwritten signature in black ink, appearing to read "Sarv".

Guide

A handwritten signature in black ink, appearing to read "P. R. M." followed by a signature.

Project Review Committee

A handwritten signature in black ink, appearing to read "S. R. L." followed by a signature.

HoD, CSE

HoD
Dept. of Computer Science & Engg.
VFSTR Deemed to be Univ.
VADLAMUDI - 522 213
Guntur Dist., A.P., India.



VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH

(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

DECLARATION

Date: 26-04-2025

We hereby declare that the project work described in the field project titled "QR CODE BASED AUTOMATED ATTENDANCE SYSTEM", is the result of our own efforts and investigations.

This project is being submitted under the supervision of **Dr. Nerella Sameera, Assistant Professor** in partial fulfillment of the requirements for the Bachelor of Technology (B.Tech.) degree in Computer Science and Engineering at Vignan's Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur, Andhra Pradesh, India.

CH. VARSHINI	231FA04B75	ch.varshini
K. LAKSHMAN	231FA04C33	k.lakshman
D.BHARATH	231FA04C47	bharath
A.RISHITHA	231FA04D01	a.rishitha

TABLE OF CONTENTS

Chapter No.	Description	Page No.
1	Introduction	1-3
	1.1 Problem Definition	2
	1.2 Existing System	2
	1.3 Proposed System	2
	1.4 Literature Review	3
2	System Requirements	4-6
	2.1 Hardware & Software Requirements	5-6
	2.2 Software requirements specifications	6
3	System Design	7-11
	3.1 Module of Systems	8
	3.2 UML Diagrams	8-11
4	Implementation	12-15
	4.1 Sample Code	13-14
	4.2 Test Cases	15
5	Results	16-20
	5.1 Output screens	17-20
6	Conclusion	21-22
7	References	22

CHAPTER-01

INTRODUCTION

“QR CODE BASED AUTOMATED ATTENDANCE SYSTEM”

INTRODUCTION

In today's academic environment, traditional methods of attendance and login are often slow, insecure, and prone to human error. Manual processes using paper records or spreadsheets can lead to inefficiencies, lost data, and administrative delays. This project introduces a QR Code-Based Attendance System aimed at modernizing and streamlining faculty login and student attendance. The system uses QR codes for secure login and OTP's for password recovery, enhancing both usability and security. It enables real-time updates, centralized record-keeping, and automated reporting, reducing reliance on manual tasks. By integrating technologies like Node.js, Express.js, and MongoDB, the system ensures scalability and reliability across platforms. Ultimately, this project supports digital transformation in education by offering a smart, paperless, and efficient attendance solution.

1.1 Problem Definition

Manual systems are inefficient, prone to errors, and lack security. Attendance through registers or spreadsheets is time-consuming, especially for large groups. Forgotten passwords create administrative delays. There's a need for a secure, digital system with QR login, dynamic record handling, and OTP-based password reset.

1.2 Existing System

- Manual login and attendance using registers or basic software.
- Forgotten passwords need admin help.
- No real-time updates or analysis.
- Lacks QR and OTP-based secure authentication.
- Centralized data storage and automated reporting absent.

1.3 Proposed System

- Faculty login via scanned QR code storing ID and password.
- Backend matches credentials with database records.
- OTP email verification for password reset.
- Automatic QR regeneration after successful login or password change.
- Student attendance tracking and summary reports stored digitally.
- Real-time updates, audit logs, and secure access controls included.
- Reports can be exported for record-keeping.

1.4 Literature Review

- According to A. Sharma et al. (2020), QR code-based login systems provide faster and more secure authentication compared to traditional username-password methods. Their study highlighted improved accuracy and reduced login time in educational platforms.
- R. Kumar and S. Patel (2021) emphasized the effectiveness of OTP-based password recovery mechanisms, stating that such systems significantly reduce the risk of unauthorized access during credential recovery.
- As per Gupta and Singh (2019), technologies like Node.js and MongoDB are well-suited for real-time web applications due to their non-blocking I/O and flexible data handling, making them ideal for attendance management systems.
- A study by T. Verma et al. (2022) highlighted that automation in attendance tracking minimizes human error, ensures transparency, and improves administrative efficiency.
- Furthermore, J. Roy (2021) discussed how web-based systems with centralized management offer greater accessibility and scalability, especially in educational institutions aiming for digital transformation.

CHAPTER-02

SYSTEM REQUIREMENTS

SYSTEM REQUIREMENTS

2.1 Hardware & Software Requirements

1. Client-Side (User Device)

- o **Computer**
 - Processor: 1 GHz or higher (Intel/AMD)
 - RAM: 2 GB minimum (4 GB recommended)
 - Storage: 100 MB free space for caching
 - Display: 1024×768 resolution or higher
 - Camera: Integrated or external webcam for QR scanning
- o **Mobile Device**
 - Android 8.0 (Oreo) or later / iOS 12 or later
 - RAM: 2 GB minimum
 - Screen Size: 5 inches or larger (for optimal UI experience)
 - Camera: Rear-facing camera with autofocus for QR scanning
- o **Internet Connection**
 - Minimum: 5 Mbps for smooth operation
 - Recommended: 10 Mbps or higher for faster loading
- o **Browser**
 - Latest versions of chrome/firefox/edge
 - Camera access permission must be enabled

2. Server-Side (Hosting Environment)

- o Processor: 2 GHz dual-core or higher
- o RAM: 4 GB minimum (8 GB recommended for high traffic)
- o Storage: SSD with at least 20 GB (for OS, server software, and database)
- o Bandwidth: Minimum 100 GB/month (scalable based on user traffic)
- o Node.js runtime with Express.js framework
- o Security: SSL/TLS enabled, environment-based credential handling

Software Requirements

1. Client-Side (Web Browser)

- o **Supported Browsers**
 - Google Chrome (v90+)
 - Mozilla Firefox (v88+)
 - Microsoft Edge (v90+)
 - Safari (v14+)
 - Opera (v75+)
- o **Browser Features Required**

- JavaScript enabled
- HTML5 & CSS3 support
- WebRTC(for accessing camera during QR scan)
- Local Storage / Session Storage
- Camera Permissions (for QR scanning functionality)

2. Server-Side (Hosting & Backend)

· Runtime Environment

- Node.js v16+
- Express.js (Web Framework)

· Database System

- MongoDB v5.0+ (Cloud-based or Local)

· Operating System Compatibility

- Linux(Ubuntu 20.04+ recommended)
- Windows Server 2019+
- macOS (for development)

2.2 Software Requirements Specification (SRS)

Frontend Development

- **Languages & Technologies**
 - HTML5 (Semantic markup for accessibility & SEO)
 - CSS3 (Flexbox, Grid, Animations)
 - JavaScript (ES6+) (DOM manipulation, event handling)
 - AJAX (For dynamic content loading)
- **Frameworks & Libraries**
 - Bootstrap 5 (Responsive grid system, pre-built components)
 - jQuery (Optional) (For simplified DOM operations)
 - Font Awesome (Icons for UI enhancements)
- **Key Frontend Features**
 - Mobile-first, responsive design
 - Cross-browser compatibility
 - Form validation (client-side)
 - Dynamic UI updates without page reload

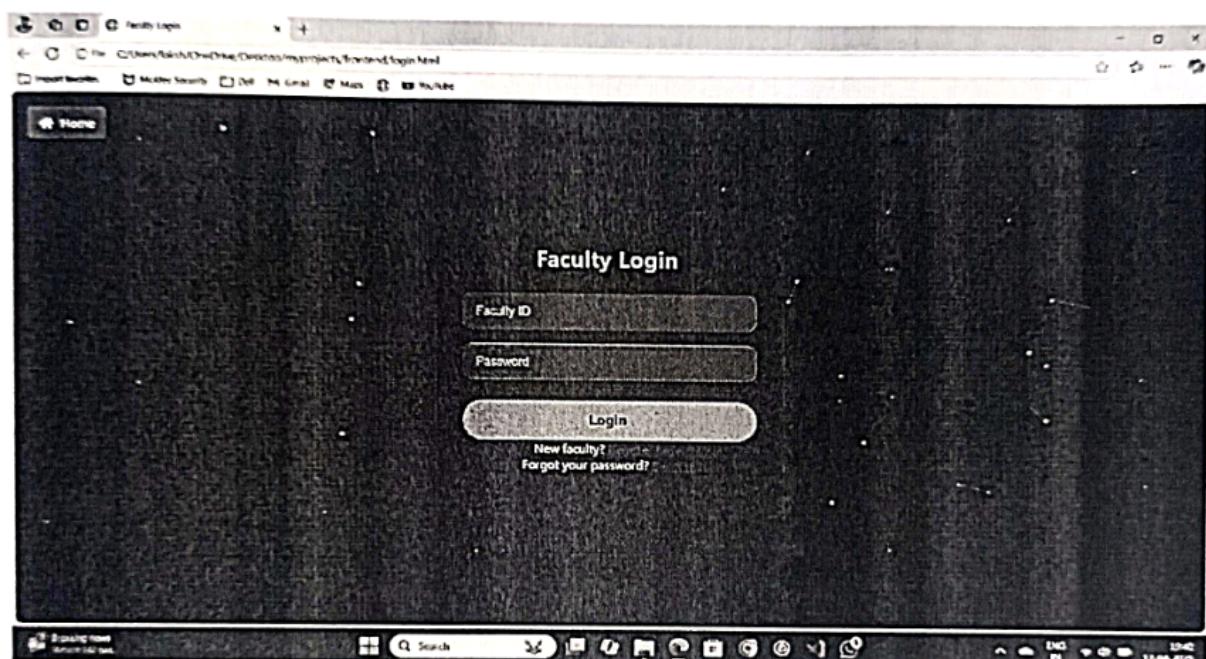
CHAPTER-03

SYSTEM DESIGN

SYSTEM DESIGN

3.1 Modules of the System

1. **Faculty Authentication Module** - QR-based login and session handling
2. **OTP and Password Reset Module** - Email OTP with time validation
3. **QR Generation Module** - Generates QR from updated credentials
4. **Student Attendance Module** - Records timestamped entries
5. **Admin Dashboard Module** - Views, filters, and exports attendance data

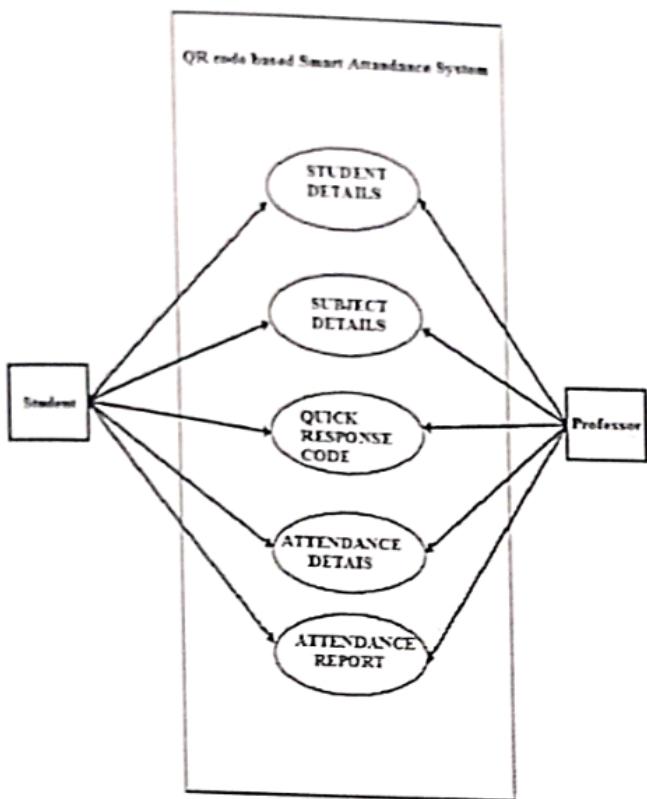


3.2 UML Diagrams

Includes system flow diagrams, use case diagrams, and sequence diagrams for better understanding of the project structure.

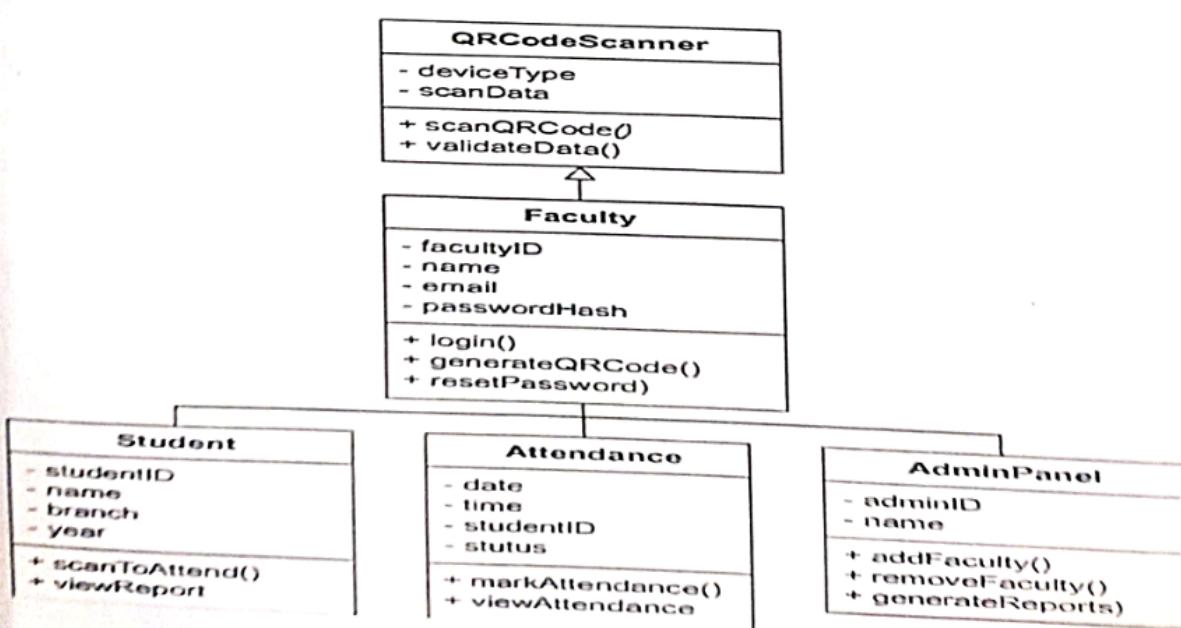
1. Use Case Diagram

"This diagram shows the main functionalities of your system from a user's perspective."



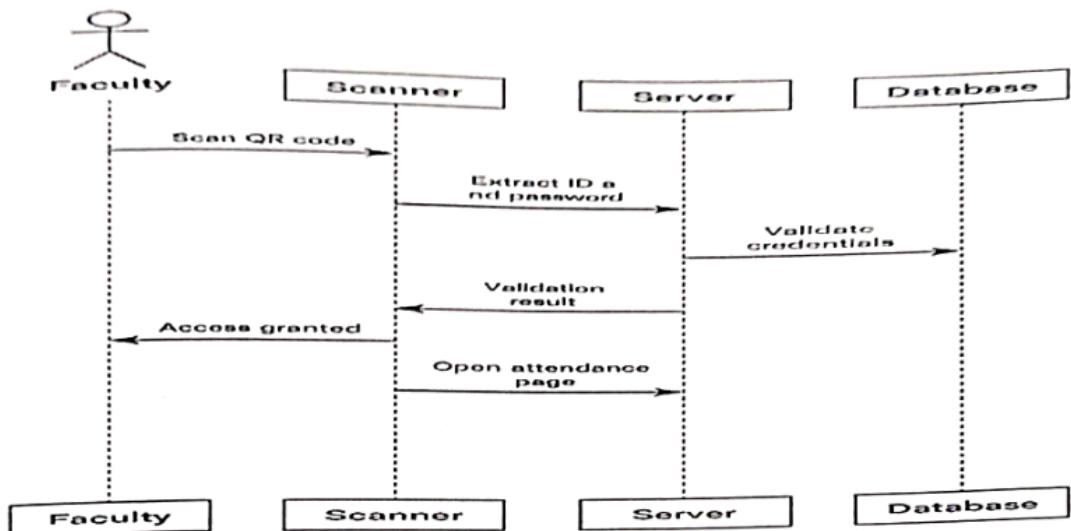
2. Class Diagram

"This represents the main entities in your system and their relationships."



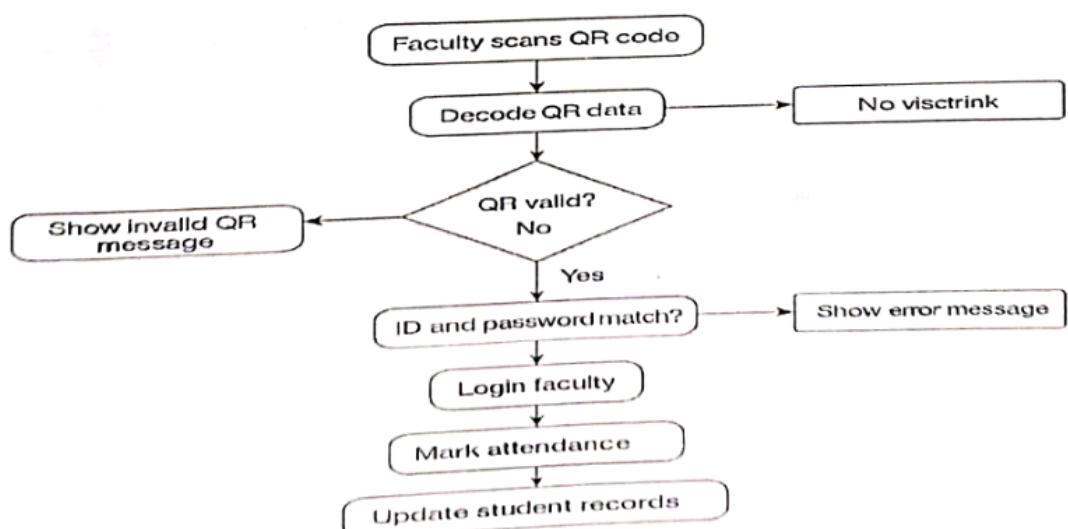
3. Sequence Diagram

It visually represents how faculty members log in using QR code authentication, how the OTP verification process works, and how student attendance is recorded.



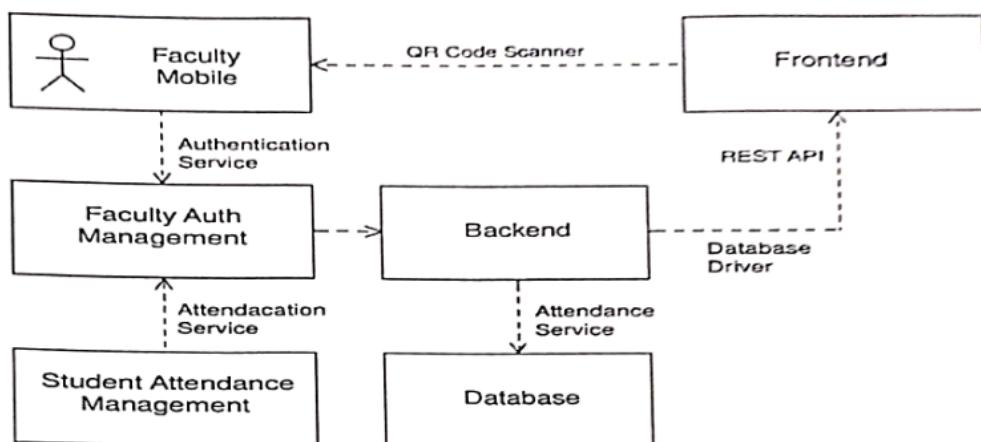
4. Activity Diagram

“This shows the workflow for recording faculty attendance”



5. Component Diagram

"This shows the high-level architecture of your system"



RESULT

The screenshot displays the following components of the system:

- QR Code Based Automated Attendance Sys**: The main title.
- Faculty Login via QR Code**: A section with a QR code for faculty login.
- Scan the QR code to login**: Instructions for faculty login.
- Enter OTP**: A modal window for entering an OTP.
- Enter OTP**: Input field for the OTP.
- Reset Password**: Link to reset password.
- A new QR code has been generated**: Confirmation message.
- Admin Dashboard**: A summary dashboard with:
 - Total Attendance: 120
 - Attendance by Subject
- Admin Dashboard**: Label for the dashboard.
- QR Code Regeneration**: A section with a QR code for regeneration.
- A new QR code has been generated**: Confirmation message for QR code regeneration.
- Student-wise Attendance Report**: A report for student Aditya Rao with:

Subject	Date	Status
Mathematics	01-02-2024	Present
Physics	24-01-2024	Present
Chemistry	15-01-2024	Absent
- Student-wise Attendance Report**: Label for the report.

CHAPTER-04

IMPLEMENTATION

IMPLEMENTATION

4.1 Sample Code

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>QR Code Login</title>
<link rel="stylesheet" href="style.css">
<script src="https://unpkg.com/html5-qrcode" defer></script>
<script src="login.js" defer></script>
</head>
<body>
<div class="container">
<h2>Faculty Login via QR Code</h2>
<div id="reader"></div>
<p id="status"></p>
</div>
</body>
</html>
```

CSS

```
body {
font-family: Arial, sans-serif;
background: #f1f1f1;
display: flex;
justify-content: center;
align-items: center;
height: 100vh;
margin: 0;
}
```

```
.container {
background: #ffffff;
padding: 20px;
border-radius: 12px;
box-shadow: 0 0 10px rgba(0,0,0,0.1);
```

```
    text-align: center;  
}  
  
#reader {  
    width: 300px;  
    margin: 0 auto;  
    margin-top: 20px;  
}  
  
#status {  
    margin-top: 15px;  
    color: green;  
    font-weight: bold;  
}
```

JAVASCRIPT

```
window.onload = function () {  
    const status = document.getElementById("status");  
  
    function onScanSuccess(decodedText, decodedResult) {  
        status.textContent = "QR Code matched: " + decodedText;  
  
        // Save session or simulate login  
        setTimeout(() => {  
            window.location.href = "dashboard.html"; // Redirect to dashboard  
        }, 2000);  
    }  
  
    const html5QrCode = new Html5Qrcode("reader");  
  
    html5QrCode.start(  
        { facingMode: "environment" },  
        {  
            fps: 10,  
            qrbox: 250  
        },  
        onScanSuccess,  
        (errorMessage) => {
```

```

    status.textContent = "Scanning...";

}

).catch(err => {
    status.textContent = "Camera access denied or error occurred!";
    console.error(err);
});

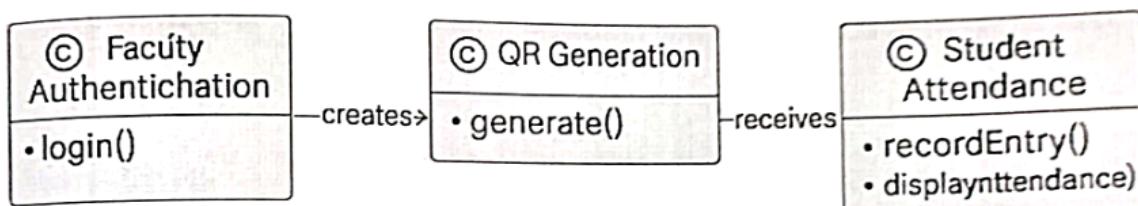
};

}

```

4.2 Test Cases

- Ensures proper functionality for faculty authentication, password reset, QR generation, and student attendance recording



- These are the scenarios, input and the expected output for each scenario

Test Case ID	Scenario	Input	Expected Output
TC01	Faculty login using valid QR	Valid QR code containing credentials	Login successful—Invalid QR Code
TC02	Faculty login using Invalid QR	QR-code with incorrect/or-credentials	Error message "Invalid QR Code"
TC03	Faculty attempts to log in expired QR	Old/used QR code	OTP verified—New QR generate
TC05	Password reset with incorrect OTP	Registered email wrong OTP	Error message "Invalid OTP. Try again."
TC06	Student attendance via QR	Valid student QR scanned	Timestamped attendance saved in DB
TC07	QR scan fails due to dash/board	No camera access	Alert: Camera access denied or blocked
TC08	Admin login to dashboard	Valid admin credentials	Access to dashboard View attendance logs
TC10	QR generation after password enge	Click export in dashboard	File downloaded in CSV/PDF format

CHAPTER-05

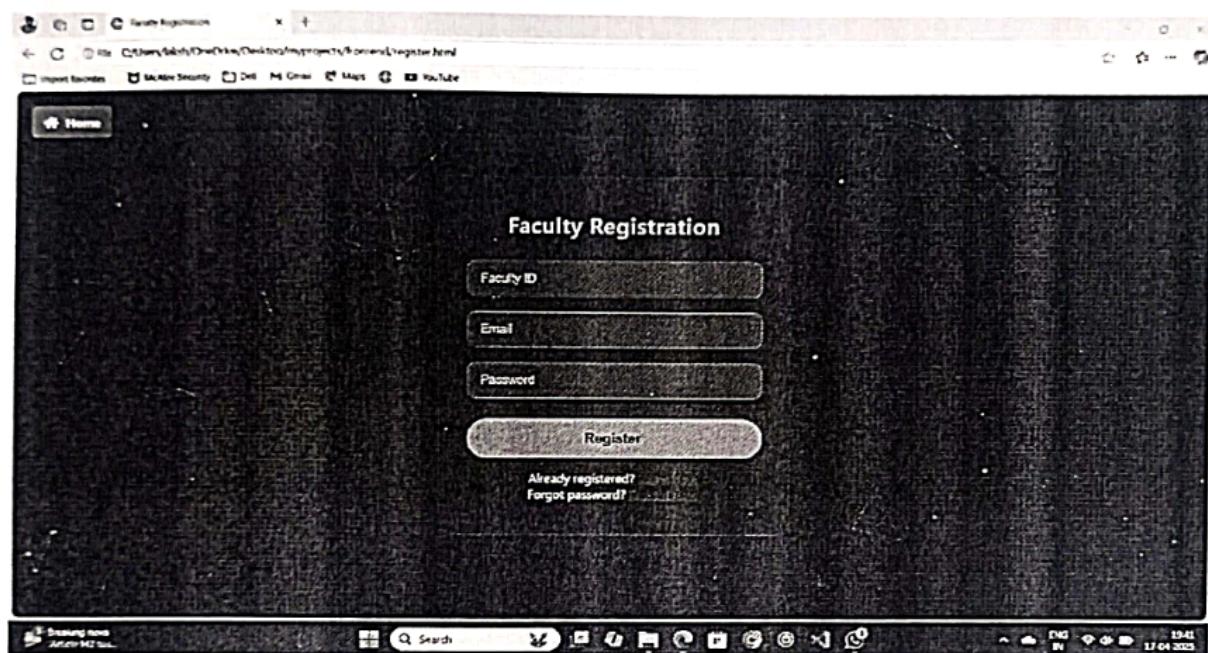
RESULTS

RESULTS

5.1 Output Screens

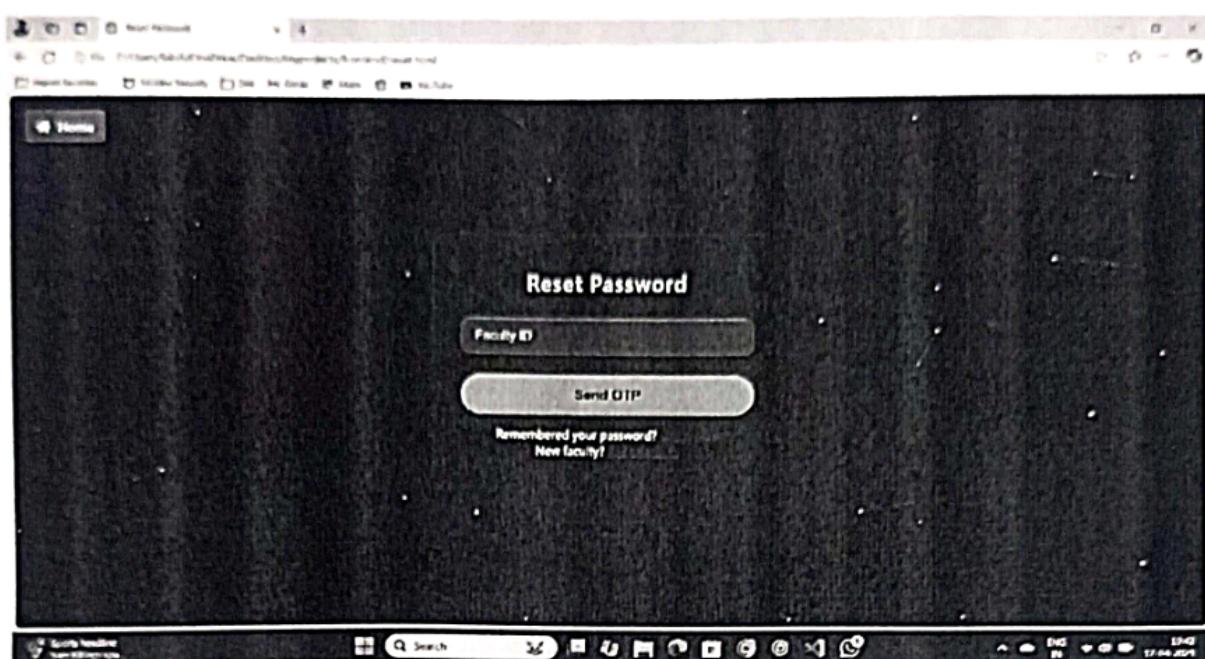
Faculty Registration

It ensures faculty login, password reset, and student attendance tracking through QR codes and dynamic database handling. This is for a faculty who is not registered from them, then can enter their facultyid, email, password and then register. And there are other options like already registered and forgot password; these are helpful if they want to change password or to login.



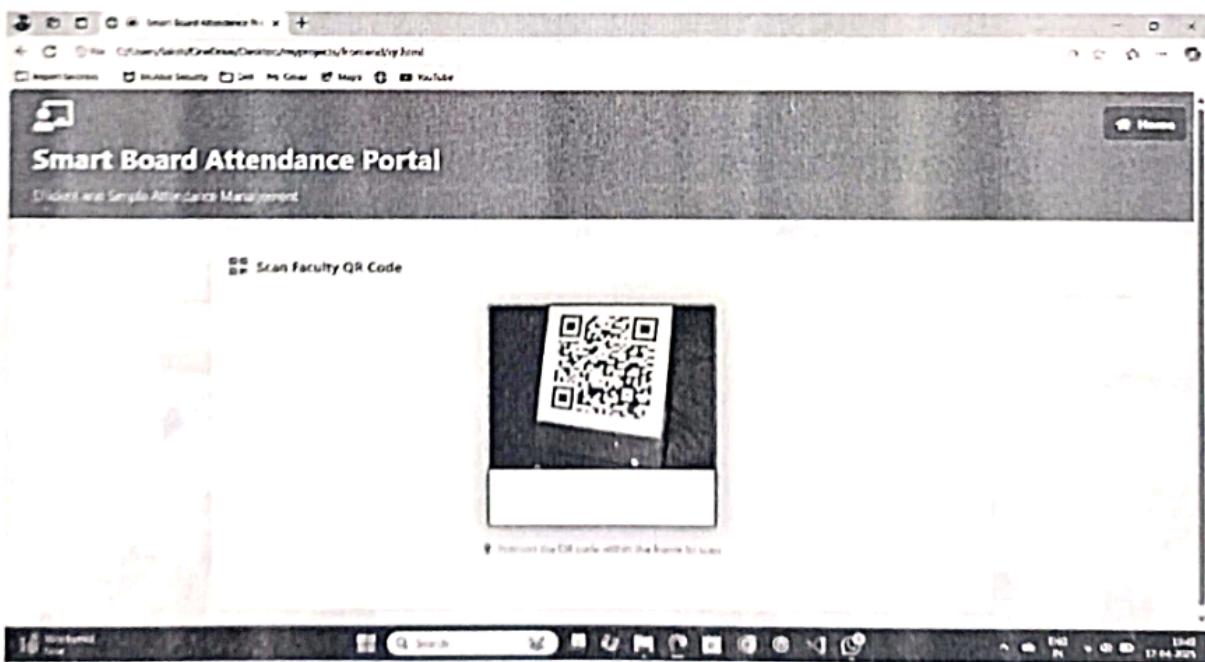
Reset Password

If faculty wants to change their password, they can reset their password by entering their faculty id and next click on send otp. The otp will be sent to the registered mail which is linked to faculty id. So next by entering the otp, they can change the password, then automatically new qr code is generated.



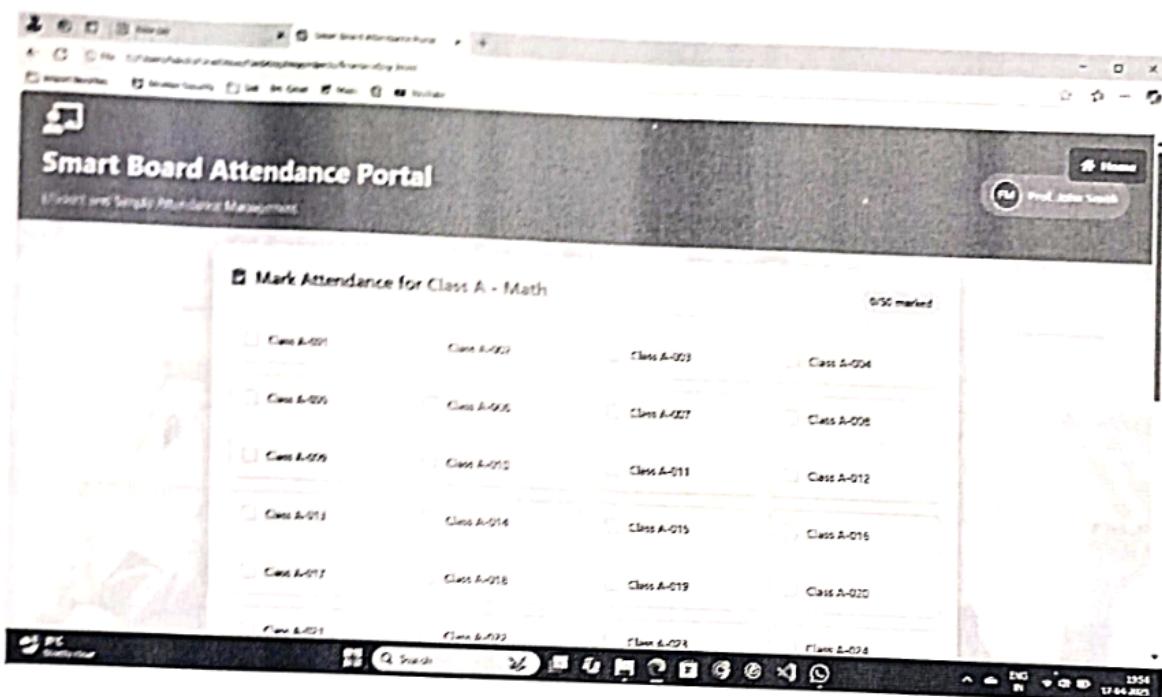
Scan Faculty QR Code

This is the page where faculty can login by using the qr code. Which is generated when they registered using their faculty id and password based on that the qr code is generated. They can store the qrcode either by taking a photo or by taking a print copy of qr code.



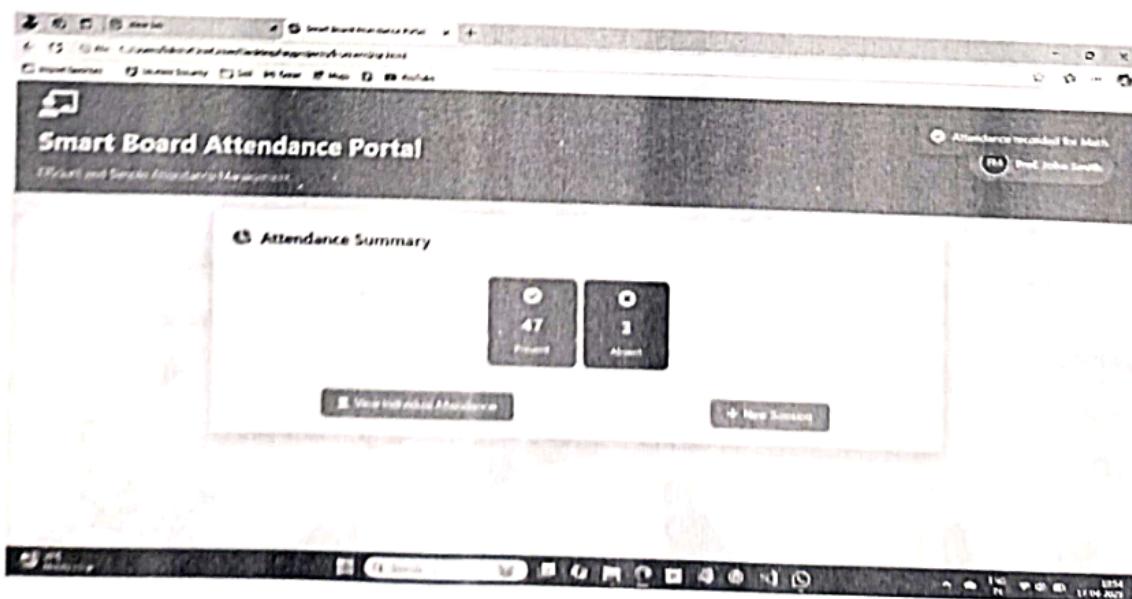
Mark Attendance

Here the faculty can mark attendance of students by calling their respective regno.then can mark absentees by clicking on their regno and next submit attendance.



Attendance Summary

After submitting it will display no of presentees and no of absentees and there is another option to view individual attendance and to start a new session to mark attendance for another class or another subject.



Subject-wise Attendance

Here they can view total attendance and subject wise attendance for every individual .if a faculty click on Class A -001 they can view their total attendance and their subject wise attendance .

The screenshot shows a web browser window titled "Smart Board Attendance Portal". The main content area displays "Attendance Details for Class A-001". On the left, there's a large circular icon containing "100%" and a link to "Subject wise Attendance". The right side lists subjects with their respective attendance percentages: Math (98%), Science (100%), English (100%), History (100%), and Geography (100%). At the bottom, there's a "Back to Student List" button.

Subject	Attendance (%)
Math	98%
Science	100%
English	100%
History	100%
Geography	100%

CHAPTER-06

CONCLUSION

CONCLUSION

The QR code-based automated attendance system delivers a secure, efficient, and user-friendly method for faculty authentication and student attendance tracking. By combining dynamic QR code generation for faculty login, OTP-protected password reset via email, and a MongoDB-backed record of student presences and absences, it minimizes manual errors and administrative overhead. Leveraging Node.js, Express.js, and frontend QR-scanning libraries ensures a responsive, real-time experience, while the modular design supports easy expansion (e.g., per-subject reporting or integration with existing MIS). Overall, this solution streamlines attendance workflows, enhances data accuracy, and boosts security for academic institutions.

REFERENCE

- Research papers on QR code-based attendance management systems and secure authentication methods.
- Online sources on Node.js, Express.js, and MongoDB for backend development.
- Documentation and tutorials for jsQR and the HTML5 getUserMedia API for in-browser QR scanning.
- Articles on using Nodemailer (or similar libraries) for OTP generation and email delivery in password-reset workflows.

Project Link

<https://github.com/Lakshman-2558/QR-Attendance-Portal.git>