International Journal for Research Trends in Social Science & Humanities
Volume 4, Issue 5
Sep-Oct 2026, PP 1-23

Intern Verify: A Secure Web-Based Internship Certificate Verification System Using Qr Codes and Sha-256 Hashing

K. Meena[1] M.Shravya[2] M.Sankeerthana[3] M.Poojaswi [4]

Department Of Computer Science And Engineering

Hyderabad Institute Of Technology And Management

ABSTRACT-Internship certificates help students demonstrate practical training and experience. However, certificates shared as PDFs or images can be copied, edited, or misused, and colleges or recruiters may find it difficult to confirm their authenticity through manual contact with the issuing organization. Intern Verify proposes a centralized digital process for submitting internship details and certificates, reviewing them through an authorized person, and checking approved records through a unique verification ID or QR code. Students can submit the organization name, internship dates, and supporting certificate. An authorized reviewer can approve, reject, or request corrections. After approval, a record can be registered so that a recruiter or institution can verify its status without relying only on the appearance of the document. The report describes the problem, related work, system requirements, proposed architecture, methodology, implementation plan, and functional testing approach. Since no completed prototype results were supplied, the report distinguishes proposed functions from measured results. Privacy, access control, accurate issuer review, and clear verification status are important considerations. The system aims to make internship verification more organized, transparent, and convenient for students, colleges, and recruiters.

I. Introduction

In today's competitive world, practical experience plays an important role in a student's academic and professional development. Internships provide students with an opportunity to apply their theoretical knowledge to real-world problems, develop technical skills, understand workplace environments, and gain experience in their chosen fields. Through internships, students can work on practical tasks, collaborate with professionals, improve their communication skills, and understand the expectations of industries. Internship experience can also help students build confidence and prepare for future employment opportunities. Therefore, maintaining reliable records of internship activities and providing proof of completed work are important for both students and organizations.

However, verifying internship experience can be challenging, particularly when students participate in short-term internships, remote internships, project-based training, or internships offered by small organizations. In some cases, internship details are maintained through informal documents, emails, spreadsheets, or manually prepared certificates. These methods may not provide a consistent way to record the tasks completed, working hours, internship duration, and mentor approval. As a result, students may face difficulties when presenting their experience to recruiters, educational institutions, or other organizations. Recruiters may also spend additional time contacting the issuing organization or mentor to confirm whether the submitted certificate is genuine.

Another important concern is the possibility of unauthorized modifications to digital certificates. For example, a person could change the internship duration, organization name, or other information mentioned in a certificate. If there is no reliable verification mechanism, identifying such modifications may be difficult. Similarly, organizations may find it challenging to maintain internship records and respond to verification requests, especially when they do not have a dedicated internship management system. These challenges highlight the need for a simple, affordable, and reliable platform that can maintain internship records and support certificate verification.

The Internship Verifier is proposed as a web-based application designed to address these challenges. The system provides a centralized platform for recording internship information, tracking assigned tasks, maintaining working hours, obtaining mentor approval, and generating digital internship certificates. It aims to connect students, mentors, and organizations through a straightforward process. By maintaining structured internship records, the proposed system helps reduce dependence on informal documentation and makes the verification process more convenient.

The system begins with the recording of essential internship details, including the student's name, organization name, assigned tasks, internship duration, total working hours, and mentor information. During the internship, completed tasks and working hours can be recorded for future reference. The mentor reviews the submitted work and provides approval after checking the student's progress. Once the required details have been recorded and the internship has been approved, the system generates a digital certificate containing the relevant internship information. This process helps establish a structured record of the student's internship activities.

A key feature of the proposed system is the use of the SHA-256 cryptographic hash functions to support the integrity verification of digital certificates. A hash function converts input data into a fixed-length value known as a hash. SHA-256 produces a 256-bit hash, commonly represented as a 64-character hexadecimal string. When the information included in a certificate changes, the resulting hash will generally be different. The system can use this property to compare the hash generated from the current certificate details with the hash associated with the original record. If the values do not match, the system can indicate that the information has changed. This provides a method for detecting modifications to certificate data.

The proposed certificate can also include a QR code to make certificate identification and verification more convenient. A recruiter or other authorized person can scan the QR code to access the certificate's identification details or verification page. The QR code can be associated with a unique certificate ID, allowing the corresponding record to be located more easily. When connected to a backend database and verification service, this feature can help users check the stored certificate information without manually entering all the details. The combination of certificate identification, QR-based access, and hash comparison is intended to make the verification process simpler and more efficient.

The system can be developed using Python and the Flask web framework for backend processing, SQLite for storing internship and certificate records, and the SHA-256 implementation available through Python's hash library. HTML and CSS can be used to design a simple and user-friendly interface. These technologies support the development of a lightweight application suitable for demonstrating the main functions of internship management and certificate verification. The system does not require blockchain technology, making the proposed approach comparatively straightforward to implement and maintain.

The main objective of the Micro-Internship Verifier is to provide a reliable and accessible method for documenting internship experience and verifying digital certificates. Its major functions include recording internship details, tracking tasks and working hours, obtaining mentor approval, generating digital certificates, creating SHA-256 hashes, assigning certificate IDs, and supporting QR-based verification. The system is intended to benefit students by providing organized proof of their practical experience, help mentors maintain records of completed work, and assist recruiters in checking certificate information.

The project focuses on developing a simple and cost-effective solution that can be used by educational institutions, startups, small and medium-sized organizations, and internship providers. The initial implementation concentrates on essential internship management and certificate verification features rather than advanced functionalities such as payment processing, blockchain integration, or a dedicated mobile application. Although SHA-256

helps detect changes in certificate data, the overall reliability of the system also depends on secure record storage, proper access control, and the authenticity of the organization issuing the certificate.

In conclusion, the Micro-Internship Verifier aims to improve the way short-term internship experiences are recorded, documented, and verified. By combining structured task tracking, mentor approval, digital certificate generation, SHA-256 hashing, and QR-code-based access, the proposed system provides a practical approach to internship verification. It can reduce the effort involved in maintaining records and checking certificates while helping students present their internship experience in a more organized manner. The project demonstrates how a lightweight web application can use digital technologies to support transparency, data integrity, and convenience in internship certificate management.

II. Literature Survey

The literature survey focuses on existing research and systems related to internship management, student work tracking, and certificate verification. Various studies have proposed web-based platforms to improve communication between students, educational institutions, and organizations. These systems help maintain student records, manage internship activities, and improve coordination between different stakeholders. However, many existing approaches have limitations in securely verifying internship certificates and detecting unauthorized modifications.Previous studies have highlighted the importance of digital platforms for managing internship information and reducing manual paperwork. Farid Ridwan and Abdull Hadi studied online internship management systems that support organized student records and internship coordination. Similarly, Nevetha P. and Amutha N. focused on web-based systems that improve communication between students and organizations. These studies show that digital platforms can make internship management easier and more efficient.Hyrmet Mydyti and Arbana Kadriu discussed management platforms that improve coordination between different stakeholders, including students and employers. Their work highlights the importance of maintaining proper communication and managing internship-related information in a structured manner. However, such systems may not provide sufficient security features for verifying the authenticity of internship certificates.Indah Naurah Fakhira et alpresented an incremental approach to internship management using student and placement records. This approach supports organized management of internship information and helps institutions maintain structured records. However, the absence of hash-based verification creates limitations in identifying unauthorized changes to stored certificate information.Donatus Edzebubu Ankah studied a web-based system focused on improving internship supervision and reducing manual tracking. The approach supports better monitoring of student activities and coordination with supervisors. However, its focus is mainly on university-based supervision and does not fully address secure digital certificate verification.The reviewed studies indicate that existing internship management systems mainly focus on record maintenance, communication, supervision, and coordination. Although these approaches improve the management of internship activities, challenges remain in verifying certificate authenticity, detecting modifications, and providing quick verification facilities for recruiters. Manual verification can also require additional time and effort from organizations.Based on the findings of the literature survey, the proposed INTERN VERIFY system aims to overcome these limitations by combining internship task tracking, mentor approval, digital certificate generation, QR code-based verification, and SHA-256 hashing. The system maintains structured internship records and allows recruiters to verify certificate details more conveniently. The use of SHA-256 hashing helps detect unauthorized modifications, while QR codes simplify access to certificate information. Thus, the proposed system aims to provide a more organized, secure, and reliable approach to internship management and certificate verification

Sl. No.	Approach / Study Area	Summary	Relevance and Limitations
1	Online internship management system (Farid Ridwan & Abdull Hadi, 2020)	Provides an online platform to maintain student records and manage internship-related information.	Reduces paperwork and improves internship management, but does not focus on secure certificate verification.
2	Web-based internship system (Nevetha P. & Amutha N., 2018)	Connects students and organizations through a web-based platform for managing internship information.	Improves communication between students and organizations, but lacks certificate security features.
3	Internship management platform (Hyrmet Mydyti & Arbana Kadriu, 2015)	Uses student and employer information to improve internship coordination and management.	Supports stakeholder coordination, but does not provide cryptographic certificate verification.
4	Incremental internship management (Indah Naurah Fakhira et al., 2015)	Uses student and placement records to maintain organized internship information.	Improves record management, but does not include hash-based verification.
5	Web-based internship supervision (Donatus Edzebubu Ankah, 2019)	Provides a web-based system to maintain supervisor records and monitor internship activities.	Reduces manual tracking and improves supervision, but is mainly focused on university-based management.

III. Online Internship Management System

The study conducted by Farid Ridwan and Abdull Hadi in 2020 focuses on an online internship management system designed to improve the management of student internship information. Traditional internship management often involves maintaining records manually, which can create difficulties in storing, updating, and retrieving student information. An online system provides a digital platform to organize internship-related details and make the management process more convenient.The system focuses on maintaining student records and organizing information related to internship activities. By using an online platform, important information can be stored systematically instead of depending entirely on paper-based records. This helps reduce paperwork and makes it easier to manage internship information. It also supports a more organized approach for institutions and organizations involved in internship programs.One of the major advantages of an online internship management system is that it improves the accessibility and organization of information. Student details and internship records can be maintained in a structured manner, reducing the effort required for manual record maintenance. It also helps simplify the management of internship-related activities and provides a digital approach to handling information.However, the study mainly focuses on online internship management and does not address secure internship certificate verification. It does not provide a specific cryptographic mechanism to identify unauthorized modifications to certificate information. Therefore, additional security features are required to ensure that internship certificates are genuine and reliable.This study is relevant to our proposed INTERN VERIFY project because it highlights the importance of maintaining internship records digitally. Our project extends this concept by introducing mentor approval, digital certificate generation, QR code-based verification, and SHA-256 hashing. These additional features aim to make internship records more secure and improve the verification process for recruiters.

IV. Web-Based Internship System

The study conducted by Nevetha P. and Amutha N. in 2018 focuses on a web-based internship management system designed to improve communication between students and organizations. Internship programs involve coordination between students and organizations, and managing this information manually can sometimes become difficult. A web-based platform provides a common place to manage internship-related information and improve communication between the people involved.The main purpose of this approach is to connect students and organizations through a digital platform. It supports the sharing and management of internship information and helps students and organizations coordinate their activities. By using a web-based system, information can be managed more conveniently, reducing the difficulties associated with traditional communication and record-handling methods.An important advantage of this approach is improved communication between students and organizations. It helps make the exchange of internship-related information easier and supports better coordination during internship activities. It also provides a structured way to manage information, which can improve the overall organization of internship programs.However, the system has limitations when it comes to certificate security and authenticity verification. Improving communication and maintaining internship information alone may not be sufficient to prove whether a certificate is genuine. Without suitable verification mechanisms, recruiters may still face difficulties in identifying fake or modified internship certificates.This study is relevant to our INTERN VERIFY project because it highlights the importance of connecting different participants through a web-based platform. Our proposed system follows a similar approach by providing facilities for students, mentors, and recruiters. In addition, it introduces mentor approval, QR code generation, and SHA-256 hashing to improve certificate verification and help identify unauthorized changes.

V. Internship Management Platform

The study conducted by Hyrmet Mydyti and Arbana Kadriu in 2015 focuses on an internship management platform designed to improve coordination between students and employers. Internship programs require proper communication and information sharing between different stakeholders. Maintaining these details in an organized manner is important for managing internship activities effectively. A management platform helps organize student and employer information and supports better coordination.The approach uses student and employer data to manage internship-related information. It provides a structured method for maintaining records and coordinating activities between the people involved in internship programs. By organizing information in a common platform, it becomes easier to manage internship details and maintain communication between different stakeholders.One of the important advantages of this approach is improved coordination and organized information management. It supports communication between students and employers and helps maintain internship-related records systematically. Such platforms can reduce difficulties in handling information manually and provide a more structured approach to internship management.However, the approach has limitations in the area of digital certificate security. Although it supports stakeholder management and coordination, it does not include cryptographic verification techniques for checking certificate authenticity. Without such mechanisms, identifying unauthorized changes to digital certificate information can become difficult.This study is useful for our INTERN VERIFY project because it highlights the importance of maintaining proper coordination between students, organizations, and other stakeholders. Our project builds on this idea by introducing a structured internship verification process. It includes task tracking, mentor approval, digital certificates, QR code-based verification, and SHA-256 hashing to improve the reliability and security of internship records.

VI. Incremental Internship Management

The study by Indah Naurah Fakhira and others in 2015 focuses on an incremental approach to internship management using student and placement records. Managing student and placement information is an important part of internship programs. When these records are

maintained manually or without proper organization, retrieving and updating information can become difficult. A structured management approach helps institutions organize internship-related details more effectively.The approach focuses on maintaining student information and placement records in an organized manner. It supports the management of internship-related data and helps institutions maintain structured information about students and their placements. By organizing these records, the process of managing internship information becomes more convenient and systematic.One of the major advantages of this approach is improved record management. It helps maintain student and placement information in an organized format and supports better handling of internship-related records. This can reduce the difficulties involved in managing information manually and helps institutions maintain proper records for future reference.However, the approach has limitations regarding digital certificate security. It does not include hash-based verification techniques to identify unauthorized changes to certificate information. Maintaining organized records is important, but additional security mechanisms are also necessary to ensure that digital certificates remain authentic and unchanged.This study is relevant to our INTERN VERIFY project because it emphasizes the importance of maintaining structured student and internship records. Our proposed system uses this concept and further extends it through task tracking, mentor approval, digital certificate generation, QR codes, and SHA-256 hashing. These features aim to provide better record management and help recruiters verify the authenticity of internship certificates.

VII. Web-Based Internship Supervision

The study conducted by Donatus Edzebubu Ankah in 2019 focuses on a web-based internship supervision system designed to improve the monitoring and management of internship activities. During an internship, students perform different tasks under the guidance of supervisors or mentors. Maintaining proper supervision records is important to understand student progress and ensure that internship activities are properly monitored. A web-based system provides a digital approach to maintaining supervision-related information.The system focuses on maintaining student and supervisor records and improving the process of monitoring internship activities. It helps organize supervision information and provides a structured way to manage internship-related details. By using a web-based platform, the effort involved in manual tracking can be reduced, and internship supervision can become more organized.One of the major advantages of this approach is that it reduces manual tracking and improves supervision. It supports better monitoring of student activities and helps maintain proper coordination between students and supervisors. This approach highlights the importance of recording internship progress and maintaining supervision-related information throughout the internship period.However, the study mainly focuses on university-based internship supervision and does not fully address secure digital certificate verification. It does not provide specific mechanisms such as QR code-based certificate verification or cryptographic hashing to detect unauthorized modifications. Therefore, additional features are required to improve the security and authenticity of internship certificates.This study is relevant to our INTERN VERIFY project because it highlights the importance of monitoring student internship activities and maintaining supervisor records. Our project extends this concept by allowing mentors to review and approve completed tasks before certificate generation. It also introduces QR code verification and SHA-256 hashing to help recruiters verify certificates and identify possible modifications.

VIII. Problem Statement

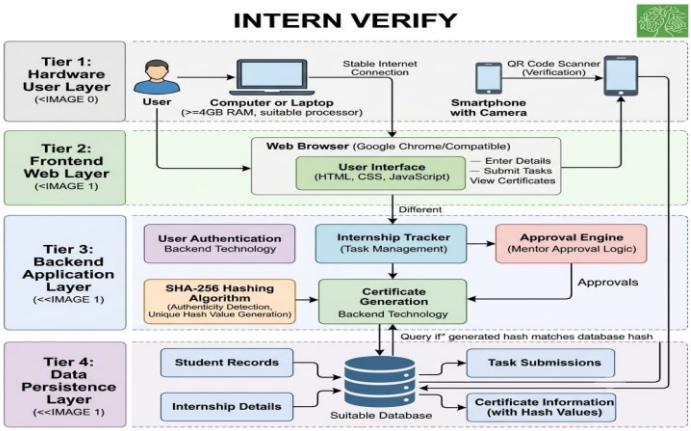
Students increasingly participate in short-duration internships, project assignments, practical training activities, and other experiences intended to build job-related skills. These experiences can be valuable even when they do not follow the structure of a long-term internship. However, the value of an experience may be difficult to demonstrate if the details of the work are not recorded consistently. Students may receive a certificate that mentions only the organization name and duration, while details about the assigned tasks,

completed work, working hours, and mentor approval may be missing or stored separately. When the student later needs to demonstrate what was completed, the supporting information may be difficult to retrieve.The problem becomes more significant during recruitment or academic evaluation. A recruiter or institution may receive a certificate but have no immediate way to check whether it was issued by the stated organization or whether its details have been changed after issuance. The verifier may need to contact the organization manually, locate the relevant mentor, and wait for confirmation .Another challenge is the lack of a unified record connecting the internship activity to the final certificate.Tasks, attendance or working-hour information, mentor feedback, and certificate files may be maintained in different places. If these records are not linked, it can be difficult to establish which tasks were approved and which details were used to generate a certificate. Without a clear approval step, the system may also generate a certificate before a mentor has confirmed the work. The application should return a clear status, such as verified, not found, or mismatch, without exposing unnecessary personal information.The proposed use of SHA-256 addresses one part of the problem: checking whether the certificate data currently being examined matches the data represented by a stored hash. However, hashing alone cannot confirm that the original information was truthful, that the organization is legitimate, or that an unauthorized person did not change both a record and its hash. The system must also use a controlled approval process and protect access to its database. Issuer authentication, role-based permissions, secure backups, and audit records are important considerations for any real deployment.Therefore, the main problem addressed by this project is the absence of a simple, integrated, and low-cost mechanism for recording micro-internship activities and making approved certificate information easier to verify. The proposed system aims to connect student and organization details, task and hour records, mentor approval,certificate generation, timestamps,hashing, and a verification page in one application.

IX. Proposed Methodology

The proposed methodology begins by identifying the difficulties faced by students in proving their internship experience and recruiters in verifying the authenticity of internship certificates. The system is designed to provide a simple and secure platform for students, mentors, and recruiters. Students can enter their internship details, record completed tasks, working hours, duration, and submit their work for mentor verification. SHA-256 hashing is applied to generate a unique hash value from the certificate information, which helps identify unauthorized modifications. The generated QR code allows recruiters to quickly access the certificate verification details and check its authenticity without contacting the organization manually. The development process includes problem identification, requirement analysis, system design, website development, integration of task tracking, mentor verification, certificate generation, QR code creation, and SHA-256 hashing. Functional testing is carried out to check student detail entry, task submission, mentor approval, certificate generation, QR code scanning, and hash verification. During evaluation, the team checks whether the system correctly records internship details, generates certificates, detects changes in certificate information, and provides accurate verification results. This methodology aims to develop a reliable and user-friendly internship verification system that improves trust and reduces the possibility of fake or modified internship certificates.

International Journal for Research Trends in Social Science & Humanities
Volume 4, Issue 5
Sep-Oct 2026, PP 1-23



X. Hardware Requirements

Sl. No.	Component	Experimental Configuration
1	Computer or Laptop	Computer or laptop for developing and running the web application
2	Processor	Suitable processor to support application development and execution
3	RAM	Minimum 4 GB RAM recommended
4	Storage Device	Sufficient storage for project files, internship records, and digital certificates
5	Internet Connection	Stable internet connection for accessing the web application and online verification
6	Smartphone / Camera	Smartphone or camera-enabled device for scanning the QR code on digital certificates

XI. Software Requirements

Sl. No.	Software / Component	Configuration
1	Operating System	Windows or Linux operating system for development and execution
2	Frontend Technologies	HTML, CSS, and JavaScript for website design and user interaction
3	Backend Technology	Suitable backend technology to manage student details, internship tasks, and mentor approvals
4	Database	Database for storing student records, internship details, task submissions, and certificate information
5	SHA-256 Hashing	Generates a unique hash value to detect unauthorized modifications in certificate information
6	Web Browser	Google Chrome or any compatible browser to access and test the application

XII. Implementation

The implementation of the proposed Internship Verification System begins with developing a web-based platform that connects students, mentors, and recruiters. The student module allows users to enter their internship details, including organization name, internship duration, working hours, and completed tasks. Students can submit their work details for mentor review and approval. The mentor module provides facilities to review submitted tasks, verify the work completed, and approve the internship details. Once the required information is verified, the system generates a digital internship certificate containing the student's details, organization information, and internship completion status. A unique QR code is generated and linked to the certificate to enable easy online verification. SHA-256 hashing is implemented to generate a unique hash value based on the certificate information, helping detect any unauthorized changes to the certificate. The recruiter or verifier can scan the QR code to access the certificate details and check its authenticity. The system also maintains internship records and verification details for future reference. Finally, the complete application is tested to ensure that task submission, mentor approval, certificate generation, QR code scanning, and hash verification work correctly. The implementation aims to provide a simple, secure, and reliable platform for managing and verifying student internship experiences.

Sample Code:

1. CERTIFICATEAVIEW.JS :-

```
import { downloadCertificatePdf, formatDate } from '../lib/pdf.js';
import { findCertificateById } from '../lib/database.js';
import { buildVerificationUrl } from '../lib/certificate.js';
import { makeQrDataUrl } from '../lib/qr.js';
import { emptyState, flowerSprig, icons } from '../ui/illustrations.js';
import { escapeHtml, verifyHref } from '../ui/html.js';
function sheetHtml(certificate) {
 return `
   <article class="certificate-sheet" aria-label="Certificate preview">
     <p class="sheet-title">Certificate of Completion</p>
     <p>This certifies that</p>
     <p class="sheet-name">${escapeHtml(certificate.studentName)}</p>
     <p>has successfully completed an internship at</p>
     <p class="sheet-company">${escapeHtml(certificate.companyName)}</p>
     <p>as ${escapeHtml(certificate.internshipRole)}, completing
${certificate.workingHours} working hours.</p>
     <p>Completed on ${escapeHtml(formatDate(certificate.completionDate))}</p>
     <p class="sheet-meta">ID ${escapeHtml(certificate.id)}</p>
     <p class="sheet-meta">SHA-256 ${escapeHtml(certificate.sha256Hash)}</p>
   </article>`;
}
function notFoundMarkup() {
```

```
  return `
    <section class="card empty-state">
      ${emptyState()}
      <h2>We could not find that certificate</h2>
      <p class="muted">Check the link or ID, or browse your saved certificates.</p>
      <div class="success-actions">
        <a class="btn btn-primary" href="#/certificates">My Certificates</a>
        <a class="btn btn-secondary" href="#/verify">Verify an ID</a>
      </div>
    </section>`;
}
export async function renderCertificateView(container, params)
  const id = params.get('id') ?? '';
  container.innerHTML = '<p class="muted">Loading certificate...</p>';
   let certificate;
  try {
    certificate = id ? await findCertificateById(id.trim().toUpperCase()) : null;
  } catch (error) {
    console.error(error);
    container.innerHTML = '<p class="alert alert-error">Sorry, this certificate could not be
loaded.</p>';
    return;
  }
  if (!certificate) {
    container.innerHTML = notFoundMarkup();
    return;
  }
 const verifyUrl = buildVerificationUrl(certificate.id);
  const qrDataUrl = await makeQrDataUrl(verifyUrl, 400);
  container.innerHTML = `
    <header class="page-header">
      ${flowerSprig({ size: 44 })}
      <div>
        <h1>Certificate details</h1>
        <p><a href="#/certificates">Back to My Certificates</a></p>
      </div>
    </header>
```

```
    <div class="certificate-layout">
      ${sheetHtml(certificate)}
      <aside class="card qr-card">
        <h3>Scan to verify</h3>
        <img src="${qrDataUrl}" alt="QR code linking to the verification page for ${escapeHtml(certificate.id)}" />
        <a class="btn btn-small btn-secondary" href="${verifyHref(certificate.id)}">${icons.verify} Open verification</a>
        <button class="btn btn-accent" type="button" id="download-pdf">${icons.download} Download PDF</button>
        <p class="form-hint" id="download-status" role="status"></p>
        <dl class="detail-list">
          <dt>Student</dt><dd>${escapeHtml(certificate.studentName)}</dd>
          <dt>Company</dt><dd>${escapeHtml(certificate.companyName)}</dd>
          <dt>Role</dt><dd>${escapeHtml(certificate.internshipRole)}</dd>
          <dt>Hours</dt><dd>${certificate.workingHours}</dd>

<dt>Completed</dt><dd>${escapeHtml(formatDate(certificate.completionDate))}</dd>
        </dl>
      </aside>
    </div>
  const downloadButton = container.querySelector('#download-pdf');
  const status = container.querySelector('#download-status');
  downloadButton.addEventListener('click', async () => {
    downloadButton.disabled = true;
    status.textContent = 'Preparing your PDF...';
try {
      await downloadCertificatePdf(certificate);
      status.textContent = 'Your PDF has been downloaded.';
    } catch (error) {
      console.error(error);
      status.textContent = 'Sorry, the PDF could not be created. Please try again.';
    } finally {
      downloadButton.disabled = false;
    }
  });
}
```

2. DASHBOARD.JS

```
import { downloadCertificatePdf, formatDate } from '../lib/pdf.js';
import { listCertificates } from '../lib/database.js';
import { emptyState, icons, sparkle } from '../ui/illustrations.js';
import { certificateHref, escapeHtml, verifyHref } from '../ui/html.js';
function rowHtml(certificate) {
  return `
    <li class="card cert-row">
      <div>
        <span class="label">Student</span>
        <span class="value">${escapeHtml(certificate.studentName)}</span>
      </div>
      <div>
        <span class="label">Company</span>
        <span class="value">${escapeHtml(certificate.companyName)}</span>
      </div>
      <div>
        <span class="label">Completed</span>
        <span
class="value">${escapeHtml(formatDate(certificate.completionDate))}</span>
        <span class="muted">ID ${escapeHtml(certificate.id)}</span>
      </div>
      <div class="actions">
        <a                class="btn                btn-small                btn-secondary"
href="${certificateHref(certificate.id)}">${icons.view} View</a>
        <a                class="btn                btn-small                btn-secondary"
href="${verifyHref(certificate.id)}">${icons.verify} Verify</a>
        <button class="btn btn-small btn-accent" type="button" data-action="download" data-
id="${escapeHtml(certificate.id)}">${icons.download} PDF</button>
      </div>
    </li>`;
}
function emptyMarkup(title, text) {
  return `
    <div class="card empty-state">
      ${emptyState()}
      <h2>${title}</h2>
      <p class="muted">${text}</p>
```

```
      <a class="btn btn-primary" href="#/generate">${sparkle({ size: 16, color: '#4A3B32' })} Generate a certificate</a>
    </div>`;
}
function renderList(listArea, certificates, query) {
  const needle = query.trim().toLowerCase();
  if (certificates.length === 0) {
    listArea.innerHTML = emptyMarkup(
      'No certificates yet',
      'Once you generate a certificate, it will appear here with view, verify, and download options.',
    );
    return;
  }
const matches = certificates.filter(
    (certificate) =>
      !needle ||
      certificate.studentName.toLowerCase().includes(needle) ||
      certificate.id.toLowerCase().includes(needle),
  );
 if (matches.length === 0) {
    listArea.innerHTML = `<p class="card muted">No certificates match "${escapeHtml(query.trim())}".</p>`;
    return;
  }
  listArea.innerHTML = `<ul class="cert-list">${matches.map(rowHtml).join('')}</ul>`;
}
export async function renderDashboard(container) {
  container.innerHTML = `
    <header class="page-header">
      ${sparkle({ size: 34, color: '#9CAF88' })}
      <div>
        <h1>My Certificates</h1>
        <p>Search, view, verify, and download every certificate you have created.</p>
      </div>
    </header>
    <div class="stats-row">
      <div class="card stat">
```

```
        <span>Total certificates generated</span>
        <strong id="stat-total">0</strong>
      </div>
    </div>
    <div class="toolbar">
      <div class="field search-field">
        <label for="certificate-search">Search by student name or certificate ID</label>
        <input id="certificate-search" type="search" placeholder="e.g. Ayesha or CERT-" autocomplete="off" />
      </div>
    </div>
    <p class="form-hint" id="dashboard-status" role="status"></p>
    <div id="list-area" aria-live="polite"></div>
  `;
 const listArea = container.querySelector('#list-area');
  const status = container.querySelector('#dashboard-status');
  const search = container.querySelector('#certificate-search');
let certificates;
  try {
    certificates = await listCertificates();
  } catch (error) {
    console.error(error);
    listArea.innerHTML = '<p class="alert alert-error">Sorry, your certificates could not be loaded.</p>';
    return;
  }
 container.querySelector('#stat-total').textContent = String(certificates.length);
  renderList(listArea, certificates, '');
search.addEventListener('input', () => renderList(listArea, certificates, search.value));
listArea.addEventListener('click', async (event) => {
    const button = event.target.closest('[data-action="download"]');
    if (!button) return;
    const certificate = certificates.find((item) => item.id === button.dataset.id);
    if (!certificate) return;
  button.disabled = true;
    status.textContent = 'Preparing your PDF...';
    try {
```

```
    await downloadCertificatePdf(certificate);
    status.textContent = `Downloaded ${certificate.id}.`;
   } catch (error) {
    console.error(error);
    status.textContent = 'Sorry, the PDF could not be created. Please try again.';
   } finally {
    button.disabled = false;
   }
  });
}
```

3. HOME.JS

```
import { flowerSprig, heroIllustration, icons, sparkle } from '../ui/illustrations.js';
export function renderHome(container) {
 container.innerHTML = `
  <section class="hero">
   <div class="hero-copy">
    <div class="eyebrow">
     ${flowerSprig({ size: 40 })}
     <span class="badge">${sparkle({ size: 14, color: '#56694A' })} Generate, store, verify</span>
    </div>
    <h1>Your Internship Journey, Beautifully Certified!</h1>
    <p class="lede">
     Create polished internship certificates, keep every record safely stored, and check
     that any certificate is registered and its details are unchanged.
    </p>
    <div class="hero-actions">
     <a class="btn btn-primary" href="#/generate">Get Started</a>
     <a class="btn btn-secondary" href="#/certificates">Browse certificates</a>
    </div>
   </div>
   <div class="hero-illustration">${heroIllustration()}</div>
  </section>
<section class="feature-grid" aria-label="What you can do">
   <a class="card feature-card pink" href="#/generate">
    <span class="icon-bubble">${icons.generate}</span>
    <h3>Generate Certificate</h3>
```

```
      <p>Enter the internship details and create a beautiful, downloadable certificate.</p>
      <span class="card-link">Start a certificate</span>
    </a>
    <a class="card feature-card sage" href="#/verify">
      <span class="icon-bubble">${icons.verify}</span>
      <h3>Verify Certificate</h3>
      <p>Check whether a certificate is registered and its stored details are unchanged.</p>
      <span class="card-link">Check a certificate</span>
    </a>
    <a class="card feature-card cream" href="#/certificates">
      <span class="icon-bubble">${icons.library}</span>
<h3>My Certificates</h3>
      <p>Browse, search, view, and download every certificate you have generated.</p>
      <span class="card-link">Open my certificates</span>
    </a>
  </section>
 `;
}
```

4. VERIFY.JS

```
import { verifyCertificate } from '../lib/service.js';
import { formatDate } from '../lib/pdf.js';
import { badgeIcon, magnifier, sparkle } from '../ui/illustrations.js';
import { certificateHref, escapeHtml } from '../ui/html.js';
const ID_PATTERN = /^CERT-[A-Z0-9]{5}-[A-Z0-9]{5}$/;
function verifiedMarkup({ certificate }) {
 return `
  <section class="result result-verified">
   <div class="result-header">
    ${badgeIcon('verified')}
    <h2>Certificate Verified</h2>
   </div>
   <p>This certificate is registered, and its stored details still match its saved SHA-256 fingerprint.</p>
   <dl class="detail-list">
    <dt>Student</dt><dd>${escapeHtml(certificate.studentName)}</dd>
    <dt>Company</dt><dd>${escapeHtml(certificate.companyName)}</dd>
    <dt>Role</dt><dd>${escapeHtml(certificate.internshipRole)}</dd>
```

```
      <dt>Hours</dt><dd>${certificate.workingHours}</dd>

<dt>Completed</dt><dd>${escapeHtml(formatDate(certificate.completionDate))}</dd>
      <dt>Certificate ID</dt><dd>${escapeHtml(certificate.id)}</dd>
      <dt>SHA-
256</dt><dd><code>${escapeHtml(certificate.sha256Hash)}</code></dd>
    </dl>
    <a class="btn btn-small btn-secondary" href="${certificateHref(certificate.id)}">View
certificate</a>
  </section>`;
}
function invalidMarkup(title, message, details = '') {
 return `
  <section class="result result-invalid">
   <div class="result-header">
    ${badgeIcon('invalid')}
    <h2>${title}</h2>
   </div>
   <p>${message}</p>
   ${details}
  </section>`;
}
function outcomeMarkup(outcome, id) {
 if (outcome.status === 'verified') return verifiedMarkup(outcome);
 if (outcome.status === 'mismatch') {
  return invalidMarkup(
5.      'INVALID CERTIFICATE',
    'This record exists, but its details no longer match the fingerprint saved when it was
created. The record may have been altered, so do not rely on it.',
    `<dl class="detail-list">
      <dt>Certificate ID</dt><dd>${escapeHtml(id)}</dd>
      <dt>Saved                                                                    SHA-
256</dt><dd><code>${escapeHtml(outcome.certificate.sha256Hash)}</code></dd>
      <dt>Recalculated                                                             SHA-
256</dt><dd><code>${escapeHtml(outcome.recalculatedHash)}</code></dd>
    </dl>`,
  );
 }
 return invalidMarkup(
```

6. 'INVALID CERTIFICATE',

```
    `No certificate is registered with the ID <strong>${escapeHtml(id)}</strong>. Check the ID for typos, or make sure you are using the same browser where it was generated.`,
  );
}
function renderResultArea(resultArea, html) {
  resultArea.innerHTML = html;
}
async function runVerification(resultArea, rawId) {
  const id = rawId.trim().toUpperCase();
  if (!id) {
    renderResultArea(resultArea, invalidMarkup('Invalid Certificate', 'Please enter a certificate ID to check.'));
    return;
  }
  if (!ID_PATTERN.test(id)) {
    renderResultArea(
      resultArea,
      invalidMarkup('Invalid Certificate', 'Certificate IDs look like CERT-XXXXX-XXXXX. Please check the ID and try again.'),
    );
    return;
  }
renderResultArea(resultArea, '<p class="muted" role="status">Checking the record and recalculating its fingerprint...</p>');
  try {
    const outcome = await verifyCertificate(id);
    renderResultArea(resultArea, outcomeMarkup(outcome, id));
  } catch (error) {
    console.error(error);
    renderResultArea(resultArea, invalidMarkup('Invalid Certificate', 'Sorry, the check could not be completed. Please try again.'));
  }
}
export function renderVerify(container, params) {
  const initialId = params.get('id') ?? '';
  container.innerHTML = `
    <header class="page-header">
```

```
      ${sparkle({ size: 34, color: '#9CAF88' })}
      <div>
        <h1>Verify a certificate</h1>
        <p>Enter a certificate ID, or open the link from its QR code.</p>
      </div>
    </header>
    <div class="verify-layout">
      <aside class="card verify-illustration">
        ${magnifier({ size: 150 })}
        <p class="muted">We look up the record, recalculate its SHA-256 fingerprint from the stored details, and compare it with the saved one.</p>
      </aside>
      <section class="verify-panel">
        <form class="card lookup-form" id="verify-form" novalidate>
          <div class="field">
            <label for="verify-id">Certificate ID</label>
            <input id="verify-id" name="id" type="text" placeholder="CERT-XXXXX-XXXXX" autocomplete="off" value="${escapeHtml(initialId)}" />
          </div>
          <button class="btn btn-primary" type="submit">Verify certificate</button>
        </form>
        <div id="verify-result" aria-live="polite"></div>
        <p class="note">
          SHA-256 shows that a record's details have not changed since it was saved. It does not prove who issued
          the certificate. Records are stored in this browser.
        </p>
      </section>
    </div>
  `;
const form = container.querySelector('#verify-form');
  const resultArea = container.querySelector('#verify-result');
  form.addEventListener('submit', (event) => {
    event.preventDefault();
    runVerification(resultArea, form.elements.namedItem('id').value);
  });
  if (initialId) {
    runVerification(resultArea, initialId);
```

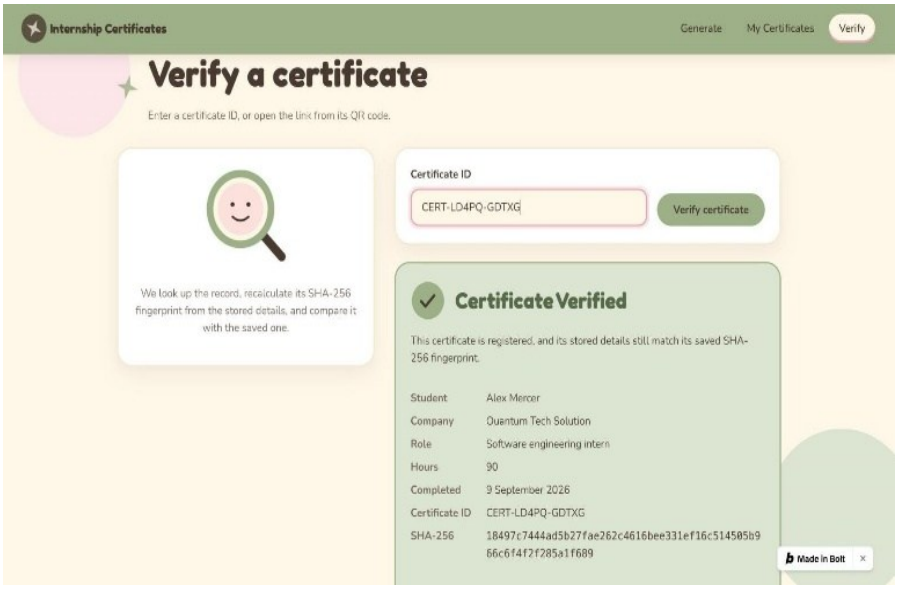
International Journal for Research Trends in Social Science & Humanities
Volume 4, Issue 5
Sep-Oct 2026, PP 1-23

}

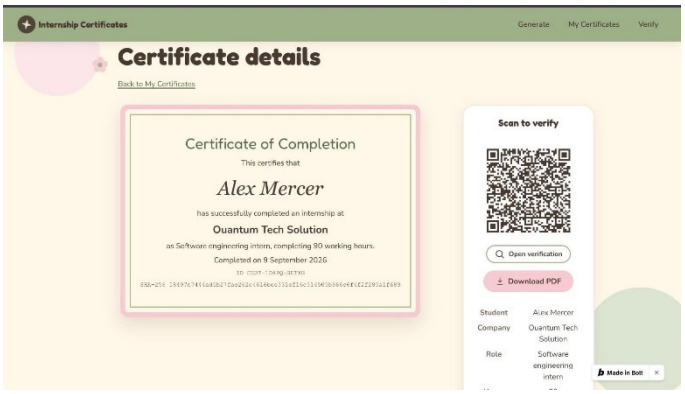
}

XIII. Result

The intended result of the INTERN VERIFY system is to develop a secure and user-friendly platform that allows students to record their internship details, maintain completed task information, and obtain verified digital certificates. Students should be able to enter their personal details, organization information, internship duration, working hours, and completed tasks. The mentor verification component is intended to allow mentors to review the submitted tasks, check the student's work progress, and approve the completed internship activities. Based on the approved details, the system generates a digital internship certificate containing the student's information, organization name, internship details, and completion status. The SHA-256 hashing mechanism is intended to generate a unique hash value based on the certificate information and help identify unauthorized modifications. Functional evaluation should check whether student details are recorded correctly, internship tasks are submitted successfully, mentors can approve or reject submitted work, and certificates are generated with the correct information. Additional testing should examine whether QR codes direct recruiters to the appropriate verification details and whether SHA-256 hashing identifies changes made to certificate information.. During evaluation, each test case should include the expected outcome, observed outcome, and any errors identified during testing. The system should be examined under different conditions, including incomplete student details, unapproved internship tasks, incorrect certificate information, and modified certificate records. These tests will help identify possible errors and areas that require improvement. The supplied project material describes the proposed features and expected outcomes but does not provide completed test logs, measured verification accuracy, or confirmed security-testing results. Therefore, no numerical performance claims can be made at this stage. After implementation and testing, this section should be updated with actual observations, screenshots, and test results, identifying which functions worked correctly and which require further improvement. Until such evidence is available, the outcome should be presented as a proposed internship verification system rather



International Journal for Research Trends in Social Science & Humanities
Volume 4, Issue 5
Sep-Oct 2026, PP 1-23



XIV. Discussion

The Internship Verification System provides an organized and reliable approach to managing and verifying student internship experiences. It addresses common problems faced by students and recruiters, such as fake internship certificates, incomplete records, and difficulties in confirming the authenticity of internship experience. The system creates a structured platform where students can record their internship details, completed tasks, working hours, and duration in an organized manner.The student module allows students to submit their internship activities and work details for verification. The mentor module helps review the submitted tasks, monitor student progress, and approve completed activities. It also helps maintain proper records of students' practical work and internship completion.The integration of digital certificates and QR codes makes the verification process easier and more convenient for recruiters. SHA-256 hashing plays an important role in improving the security of the system. It generates a unique hash value based on the certificate information, which helps identify unauthorized modifications. If any changes are made to the original certificate details, the hash values can be compared to detect possible tampering. This improves the reliability and integrity of the digital certificates.The system also improves transparency and trust between students, mentors, organizations, and recruiters by maintaining verified internship information. It reduces paperwork, simplifies record management, and provides an efficient method for checking student internship experiences.Overall, the proposed Internship Verification System offers a simple, secure, and user-friendly solution for managing internship records and verifying digital certificates. By combining task tracking, mentor approval, QR code generation, and SHA-256 hashing, the system aims to reduce fake certificates and provide reliable proof of students' practical work experience.

XV. Conclusion

The Internship Verification System provides a secure, simple, and reliable solution for managing and verifying student internship experiences. It addresses the common challenges of fake internship certificates, incomplete records, and time-consuming manual verification. By maintaining student details, internship activities, completed tasks, working hours, and mentor approvals, the system ensures that internship information is organized and easily accessible.The system uses mentor verification to ensure that students' completed tasks are reviewed and approved before generating the final digital certificate. The integration of QR code technology allows recruiters to quickly access and verify certificate details without depending entirely on manual communication with organizations. SHA-256 hashing adds an additional layer of security by helping detect unauthorized modifications to certificate information and maintaining data integrity. Further more, the proposed system reduces paperwork, saves verification time, and improves transparency between students, mentors, organizations, and recruiters. It provides a convenient way to maintain internship records and establish trust in students' practical work experience.Overall, INTERN VERIFY aims to improve the credibility of internship certificates through digital verification and secure

record management. It offers an effective approach to reducing fake certificates and making the internship verification process easier, faster, and more trustworthy.

XVI. Future Work

The Internship Verification System can be further improved by introducing advanced features to make internship management and certificate verification more secure and efficient. In the future, blockchain technology can be integrated to provide stronger security and maintain tamper-proof internship records. Artificial Intelligence (AI) can be introduced to evaluate students' skills, performance, and internship activities based on their submitted work.The system can also be enhanced by developing a mobile application to provide easy access for students, mentors, and recruiters. Integration with professional platforms like LinkedIn can allow students to add their verified internship certificates directly to their profiles. Smart recommendation features can also be introduced to suggest suitable internship opportunities based on students' skills, interests, and qualifications.Furthermore, additional security features and improved verification methods can be implemented to strengthen certificate authenticity and protect student information. The system can be expanded to support multiple organizations, educational institutions, and internship programs.Overall, future improvements aim to make INTERN VERIFY a more advanced, accessible, and trustworthy platform that simplifies internship management, improves certificate security, and helps students showcase their genuine practical experience.

References

1. National Institute of Standards and Technology, “Secure Hash Standard (SHS),” FIPS PUB 180-4, Aug. 2015. doi: 10.6028/NIST.FIPS.180-4.
2. International Organization for Standardization (ISO), ISO/IEC 18004:2024 – QR Code Bar Code Symbology Specification, 2024.
3. A. Rustemi et al., “A Systematic Literature Review on Blockchain-Based Systems for Academic Certificate Verification,”IEEE Access, vol. 11, pp. 64679–64696, 2023. doi: 10.1109/ACCESS.2023.3289598.
4. F. Ridwan and A. Hadi, “Intership management system,” Universiti Malaysia Pahang, 2012. Available: https://umpir.ump.edu.my/id/eprint/7021/. Verify repository metadata before submission.
5. F. Fei, A. Waheeb, and S. Sulaiman, “Student Internships: A Selective Review,” Journal of Applied Technology andInnovation, vol. 4, no. 1, 2020. doi: 10.65136/jati.v4i1.260.
6. Python Software Foundation, “Hashlib – Secure Hashes and Message Digests,” Python Documentation.https://docs.python.org/3/library/hashlib.html
7. Controller of Certifying Authorities (CCA), Government of India, “Digital Signature Certificate Verification.” https://cca.gov.in/signature_verification.html
8. SQLite Consortium, “SQLite Documentation – Database Management System.”https://www.sqlite.org/docs.html
9. OWASP Foundation, “File Upload Cheat Sheet – Secure File Upload Guidelines.” https://cheatsheetseries.owasp.org/cheatsheets/File_Upload_Cheat_Sheet.html
10. DENSO WAVE Incorporated, “Types of QR Code and Their Features.” https://www.qrcode.com/en/codes/index.html
11. National Institute of Standards and Technology (NIST), “Digital Identity Guidelines – Identity Proofing and Enrollment,” SP 800-63A-4, 2025. https://csrc.nist.gov/pubs/sp/800/63/a/4/final
12. Mozilla Developer Network (MDN), “SubtleCrypto: digest() Method.” https://developer.mozilla.org/en-US/docs/Web/API/SubtleCrypto/digest

13. OWASP Foundation, “Logging Cheat Sheet – Application Security and Audit Logs.” https://cheatsheetseries.owasp.org/cheatsheets/Logging_Cheat_Sheet.html