Online Scholarship Form Verification for

GNDEC

PROJECT SYNOPSIS

OF MINOR PROJECT

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE & ENGINEERING

SUBMITTED BY:

MANISHA GUPTA(2203496)

KARMANDEEP KAUR(2203486)

NAVJOT KAUR(2203510)

Akshal
19/2/25

GURU NANAK DEV ENGINEERING COLLEGE

LUDHIANA-141013

Table of Content

1. Introduction

Scholarship management system of our college needs to track which students have applied for scholarships. To address this need, we propose the development of a scholarship management website that enables students to register and submit their scholarship details, specifically their OTR (One-Time Registration) number issued by the National Scholarship Portal (NSP). This OTR number is a unique identifier assigned to students after they apply for a scholarship. Our system will allow students to upload their OTR card, input their OTR number, along with additional personal details such as father's name, mother's name, and date of birth. The system will then extract text from the uploaded OTR card and validate whether the provided OTR number matches the extracted one. If the numbers match, the student will receive a success email; otherwise, an alert will be issued for incorrect submissions.

2. Rationale

The current process of tracking scholarship applications is inefficient and lacks automation, leading to potential errors, delays, and mismanagement. Our website aims to streamline the verification process by automating the OTR number validation, thereby reducing manual efforts and enhancing accuracy. Additionally, it ensures transparency and accountability in scholarship application tracking, helping the college maintain an organized record of applicants.

3. Objectives

1. Design and implementation of a web-based platform for scholarship form submission and verification.

2. Facilitate the college scholarship department to check the status of Student's scholarship application.

3. Auto-generate and send confirmation messages to the user's email upon successful verification and submission.

4. Literature Review

The development of scholarship management systems has been widely explored in academic research and industry practices. Several studies highlight the significance of digital transformation in scholarship management.

- **Automated Scholarship Systems**: Research indicates that automation in scholarship verification processes improves efficiency and reduces administrative workload. Digital platforms have been successfully implemented in universities to enhance transparency and accuracy in tracking applications.
- **Use of OCR Technology**: Optical Character Recognition (OCR) has been effectively utilized in document verification and text extraction processes. Studies demonstrate that OCR can significantly improve the accuracy of data extraction from official documents.
- **Web-Based Student Portals**: Various universities have adopted web-based portals for student registration and document submission. Literature suggests that these platforms not only streamline application tracking but also enhance the overall user experience.
- **Secure Database Management**: The importance of secure, cloud-based database systems in managing sensitive student information is emphasized in recent studies. MongoDB and other NoSQL databases are commonly recommended for their scalability and security features.

5. Feasibility Study

A feasibility study ensures that the project is viable and achievable within the given constraints. The key aspects considered include

5.1 Technical Feasibility

- The project utilizes modern web development technologies, including **HTML, Tailwind CSS, JavaScript, React.js, Node.js, and MongoDB**, ensuring a scalable and efficient system.
- The use of **OCR technology** for extracting text from the OTR card is feasible with readily available libraries such as **Tesseract.js**.
- The system's email notification feature can be implemented using **Nodemailer** in Node.js.

5.2 Economic Feasibility

- The cost of development is minimal as open-source technologies are used.
- The cloud-based hosting and database services, such as **MongoDB Atlas**, provide cost- effective storage solutions.
- Automation reduces the need for manual verification, saving time and resources for the college.

5.3 Operational Feasibility

- The system is designed to be **user-friendly**, allowing students to complete the registration process with ease.
- College administration can efficiently monitor the list of students who have successfully applied for scholarships.
- The project aligns with institutional goals by ensuring an accurate and efficient tracking system.

6. Methodology

The development process will follow the **Agile methodology**, ensuring iterative development and testing. The project will be completed in the following phases:

1. **Requirement Analysis**
    - Understanding the needs of the scholarship management system.
    - Defining key functionalities.
2. **Design Phase**
    - Creating wireframes and UI/UX design using Tailwind CSS.
    - Designing the database schema using MongoDB.
3. **Development Phase**
    - Implementing the **registration form** using React.js.
    - Developing file upload functionality for the **OTR card**.
    - Integrating OCR technology for text extraction.
    - Implementing Node.js backend for processing and validating OTR numbers.
    - Setting up email notification using Nodemailer.
4. **Testing Phase**
    - Conducting unit testing for individual components.
    - Performing integration testing to ensure system functionality.
    - Running security tests to validate data protection measures.
5. **Deployment & Maintenance**
    - Hosting the website on a cloud-based platform.
    - Conducting user training sessions.
    - Providing continuous updates and support

7. Facilities Required for Proposed Work

The successful execution of the project requires the following resources:

1. **Software Requirements:**
    - Operating System: Windows/Linux/MacOS.Development .
    - Tools: Visual Studio Code, Postman.
    - Programming Languages: JavaScript, HTML, CSS.
    - Frameworks & Libraries: React.js, Node.js, Tailwind CSS, Tesseract.js (OCR).
    - Database: MongoDB (Atlas or Local Setup).
    - Email Service: Nodemailer.
2. **Hardware Requirements:**
    - A computer with a modern processor (Intel i5/i7 or equivalent AMD Ryzen).
    - Minimum 8GB RAM.
    - Stable internet connection.
    - Cloud hosting services (AWS, Vercel, or Firebase).

9.References

1. Convert Images to Text in Node.js with Tesseract.js by Medium

   https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://medium.com/%40abhishekchamoli007/convert-images-totext-in-node-js-with-tesseract-js-a-step-by-step-guideb4fa5f5ee809&ved=2ahUKEwjRuP29rqLAxVxzTgGHTUcO6gQFnoECB8QAQ&usg=AOvVaw1IPmH7vdxL8FPL-j5qkES

2. Image To Text Conversion With React And Tesseract.js Article by Smashing magazine.

   https://www.smashingmagazine.com/2021/06/image-text-conversionreact-tesseract-js- ocr/