

Workflow Automation for Recommendation Letter Approval for a University

Group Members:

P.M.M. Mayurajith 2020/ICT/43

S. Venuka 2020/ICT/70

M.J.H. Yeenas 2020/ICT/75

S. F. Shakeeka 2020/ICT/81

M.J. Ruzla 2020/ICT/82

M.N.I. Fernando 2020/ICT/106

J.A.M.P. Jayasinha 2020/ICT/108

Supervisor: Dr. S. Kirushanth, Senior Lecturer

Department of Physical Science

Faculty of Applied Science

University of Vavuniya

October 2024

Declaration

We hereby declare that the project proposal submitted for evaluation of course module IT3162 for the award of a Bachelor of Science in Information Technology is our own work, and contents from the work of others are cited and acknowledged. This proposal has not been submitted for any degree at this or any other institution.

Group Members:

- 2020/ICT/43 P.M.Mayurajith
- 2020/ICT/70 S. Venuka
- 2020/ICT/75 M.J. Yeenas
- 2020/ICT/81 S. F. Shakeeka
- 2020/ICT/82 M.J. Ruzla
- 2020/ICT/106 M.N.I. Fernando
- 2020/ICT/108 J.A.M.P. Jayasinha

Contents

1	Introduction	4
1.1	Introduction	4
1.2	Objectives	5
1.3	Benefits of this Research	6
2	Background	7
2.1	Background	7
2.2	Review on the Existing Systems	7
3	Material and Methods	8
3.1	Brief Description of Proposed Systems Design	8
3.2	Methodology	8
3.3	Tools	9
3.4	Requirements	9
4	Expected Results	10
5	References	12

1 Introduction

1.1 Introduction

A recommendation letter at a university level always involves multilevel processing by lecturers, heads, and administrators, which is a very cumbersome and time-consuming process. Such a manual approach results in delays in processing, but it also invites errors, inefficiencies, and miscommunications. In regard to the increasing need for smoothing the administrative workflow of educational institutions, there is a wide opportunity to implement the use of technology within their automation and refinement. We address the design of a workflow automation system that will facilitate simplification of the process of requesting approval for recommendation letters. Our contribution will be an easy-to-handle Workflow Automation System that is going to improve the process of requesting approval for recommendation letters and further integrate into university infrastructure in such a way as to enhance efficiency, transparency, and overall experience for students, lecturers, and administrators. The proposed system will enable students to place their requests in electronic form while providing the faculty and staff with a formalized efficient process to review, approve, or reject these requests. With regard to institutional policies, automatic notification and tracking of request statuses in real-time ensure that delays or bottlenecks are minimized. This automating will ensure that the highest efficiency in operations is achieved, and satisfaction for both students and employees of the university improves in the way desired by the university through modernizing its services and enhancing administrative capability.

1.2 Objectives

1. Simplify Document Approval Process:

Develop a system that automates the entire workflow for recommendation letter requests, reducing manual intervention and eliminating inefficiencies.

2. Provide Hierarchical Letter Approval Process

3. Enhance Accessibility for Students and Staff:

Provide an intuitive, user-friendly interface that allows students to submit their requests digitally and track the status of their applications in real-time.

4. Improve Administrative Efficiency:

Automate administrative tasks such as sending notifications, reminders, and approvals, which will reduce processing time and enable staff to focus on more critical tasks.

5. Ensure Transparency and Accountability:

Implement a tracking and logging mechanism that ensures every step of the approval process is recorded, providing transparency and accountability for both students and staff.

6. Increase the Feasibility of Integration with Existing University Systems:

Ensure that the workflow automation system can integrate with current university infrastructure, such as email systems, student information systems, and staff portals, to facilitate seamless operation.

7. Reduce Errors and Delays:

Minimize human errors and delays by automating manual processes, thereby reducing the chances of miscommunication or lost documents.

8. Provide Customizable Features for Policy Adherence:

Ensure that the system is customizable to comply with the university's specific policies and approval procedures for recommendation letters.

1.3 Benefits of this Research

- **Increased Efficiency:**

The automation of the recommendation letter approval process will significantly reduce the time spent on manual handling of requests, allowing faster turnaround times for both students and staff.

- **Reduced Administrative Workload:**

Automation of the sending of reminders and tracking of approvals lets university staff focus on strategic student-facing activities, with an overall gain in workload optimization.

- **Improved Student Satisfaction:**

Students benefit through more transparent, streamlined, and convenient procedures. With request submission now online and real-time tracking of application status, better response times would naturally result, reducing student anxiety.

- **More Transparent and Accountable:**

Automation that tracks the approval process ensures every action is documented, making the process easier to audit, keep track of progress, and hold all parties accountable.

- **Human Errors Are Minimized:**

Automation of tasks leads to fewer possible human errors; lost forms or misunderstandings between departments are largely avoided, achieving a more precise and consistent procedure.

- **Cost-Effectiveness:**

Although there are initial development costs, over the long run, there will be reductions in operational costs from manual paperwork, printing, and administrative overhead.

- **Scalable Future University Processes:**

The system can serve as a platform for other administrative functions within the university, making it scalable and adaptable to future workflow automation needs.

- **Real-Time Monitoring and Decision-Making:**

With automated notifications and status updates, decision makers can review and approve requests in a timely manner, avoiding unnecessary delays and enhancing operational decision-making.

2 Background

2.1 Background

Universities handle a large number of documents every day—ranging from student leave applications to staff requests, and other administrative forms. Typically, these submissions go through multiple layers of approval, involving different departments and individuals. Right now, this process is mostly manual, requiring physical paperwork and face-to-face interactions, which can be slow and inefficient. In a world where digital tools are transforming every aspect of our lives, automating these internal processes can help the university function more smoothly. By digitizing document submissions, we can save time, reduce errors, and improve transparency. This system aims to create a digital workflow where students, staff, and labourers can submit their forms easily, and approvals can happen quickly—without the need for paper or back-and-forth emails. With templates already set up, users will only need to fill in their details and the system will handle the rest. This would greatly simplify how we handle paperwork at the university

2.2 Review on the Existing Systems

Currently, most document submissions within the university are done manually. Students and staff typically have to write letters or fill out forms using word processing apps, then print them out, sign them, and deliver them physically to the right office. From there, the forms are passed from person to person for approval, often resulting in delays, lost documents, and confusion about where the submission is in the process. The existing system is not only time-consuming but also prone to human error. Approvals can take days or even weeks, especially if someone in the approval chain is unavailable. There's also a lack of transparency—once a document is submitted, users often have no way of knowing where it is in the approval process or who is responsible for the delay. Moreover, the heavy reliance on paper-based submissions has environmental costs and adds unnecessary complexity to managing records and archives. With the proposed automated system, these pain points can be addressed by digitizing and streamlining the entire process, providing a more efficient, user-friendly, and transparent solution for everyone involved.

3 Material and Methods

3.1 Brief Description of Proposed Systems Design

This system design aims to manage the letter submission workflow (submission, approval, etc.) within a university or organization. It includes a portal accessible by users (students, lecturers, and administrators) to submit, track, and manage the status of their submissions (e.g., approved or rejected). The system design consists of multiple portals to streamline different roles and responsibilities, as follows:

- **User Portal:**
Allows users to submit documents and track the status of approvals or rejections in real-time.
- **Admin Portal:**
Provides administrative users with tools to manage document types, edit submission hierarchies, and oversee the entire approval process.
- **Approval Portal:**
Designed for users with approval rights (such as lecturers, heads of departments, and deans) to accept or reject submissions. The approval workflow enables the document to pass through various hierarchical levels, based on document type and user roles.

3.2 Methodology

The system will be developed using the Agile methodology, incorporating an iterative development process to enable flexibility and continuous improvement. The key stages involved in this methodology are as follows:

1. **Requirement Gathering:** Conduct interviews with stakeholders (students, lecturers, administrators) to gather detailed requirements for the system.
2. **System Design:** Break down the system into distinct modules, such as submission, tracking, approval, and user management, to create a structured architecture.
3. **Implementation:** Develop the system in iterative cycles, starting with basic features like submission tracking, and add approval workflows in subsequent iterations.
4. **Testing:** Conduct various testing stages, including unit testing, integration testing, and user acceptance testing, to ensure functionality, usability, and reliability.
5. **Deployment:** Roll out the portal with continuous monitoring, gathering user feedback, and updating the system as needed to maintain functionality and user satisfaction.

3.3 Tools

- **Programming Languages:**

- **MongoDB:** Uses JavaScript or JSON-like queries.
- **Express.js:** Written in JavaScript.
- **React.js:** Written in JavaScript (often with JSX).
- **Node.js:** Runs JavaScript code on the server.

- **Database Management:**

- MongoDB for managing and storing user submissions, approval histories, and other related data.

- **Version Control:**

- GitHub for managing the development process with version control and collaboration.

3.4 Requirements

Functional Requirements:

1. The system should allow users to submit documents and track their status in real-time.
2. Administrators should be able to modify submission hierarchies and document types.
3. Role-based access control should be implemented to ensure only authorized users can approve or reject submissions.

Non-functional Requirements:

1. Scalability: The system should be able to handle a large number of simultaneous users.
2. Security: Sensitive data such as student submissions and approvals must be protected using encryption and role-based access.
3. Usability: The system should have an intuitive user interface with clear instructions on submission and tracking processes.

4 Expected Results

- **Simplify Document Approval Process**

- **Real-time Approval Process Tracking:**

Users can easily track where their document is in the approval process. Whether it is waiting for approval or already approved, the visibility provided reduces confusion and the need for follow-ups.

- **Provide Resubmission Service for Rejected Documents**

- **Smooth Submission and Approval:**

Once a user submits a document, it automatically flows through the appropriate hierarchy for approval, such as lecturers or department heads. The system removes the need to manually manage or chase approval contacts.

- **Admin-Controlled Templates:**

Administrators have the authority to create and manage templates, ensuring consistency across the university. They can also set predefined paths that each document follows during the approval process, making operations organized and efficient.

- **Role-Based Access:**

Access to different templates and submission options varies based on the user's role (student, academic staff, non-academic staff, etc.), ensuring that users interact only with relevant options and keeping the process streamlined.

- **Quick and Clear Approvals:**

Users can see the current status of their document, whether pending approval or approved, making the process more transparent and reducing the need for status inquiries.

- **Saves Time and Prevents Mistakes:**

Templates and automation reduce errors, such as incorrect submissions or missing information, ensuring the process is faster and more accurate.

- **Flexible Approval Paths:**

The admin can customize the approval steps for various document types, ensuring each submission is reviewed by the appropriate personnel. This flexibility allows the system to adapt to different departments and requirements.

- **Environmentally Friendly:**

A paperless approach reduces the need for physical documents, promoting sustainability and supporting the university's eco-friendly initiatives.

- **Admin Authority:**

Administrators have full control over user roles, workflows, and templates, allowing them to ensure the system operates smoothly and efficiently for all users.

ACTIVITIES	October			November				December			
	2nd Week	3rd week	4th week	1st week	2nd week	3rd week	4th week	1st Week	2nd Week	3rd Week	4th Week
Title											
Proposal											
Gathering Information											
Planning											
Prototype											
Designing											
Developing											
Testing											
Documentation											

Research Timeline

5 References

- <https://react.dev/>
- <https://expressjs.com/>
- <https://www.mongodb.com/>
- <https://www.figma.com/>
- <https://www.mysql.com/>