PAYMENT REQUEST APPROVAL SYSTEM
(For Construction Company)

Chit Su Zaw
Student, 5th Year, B.C.Sc.
University of Computer Studies, Hinthada

Supervised By
Daw Wint Kyu Kyu Ko
Faculty of Computer Science
University of Computer Studies, Hinthada

Abstract. Payment Request Approval System is an automated solution designed to manage the various functions of payment requests and approvals. It handles all processes involved in payment approvals seamlessly. This project provides comprehensive information about the payment request workflows. The system offers many flexible and convenient features, allowing approvers and requesters to minimize time and maximize efficiency.

Keywords: payment, request, approval, PowerApps, PowerAutomate, SharePoint lists, Outlook Mail, approvers, requesters

INTRODUCTION

Today, many organizations need to manage their data to improve performance, reliability and security. With advancements in technology, digital systems have gained popularity. These systems are widely used across various fields for replacing traditional paper-based method. Digital systems are more accessible, save space, easier to manage and operate faster than writing on paper.

PROBLEM

The Payment Request Approval System, built by using Microsoft Power Apps[1], Power Automate[4], and SharePoint Lists[3] to help organizations manage their payment approval processes more effectively. This system aims to automate and simplify the steps involved in approving payments, which traditionally require a lot of manual work and paperwork. By using these Microsoft tools, the system provides a user-friendly interface and integrates smoothly to offer a comprehensive solution.

APPROACH

The system includes features such as user authentication and role management to ensure that only authorized people can access it and perform specific tasks. Users can submit payment requests through a simple form that captures all necessary details, such as the payment amount, description, due date, and any supporting documents. These requests are stored in a SharePoint list, which serves as the database. Once a request is submitted, it enters an approval workflow managed by Power Automate, which can be customized to match the organization's approval hierarchy. This ensures that each request is reviewed and approved by

Accepted Date: 25.10.2024
www.ucsh.edu.mm

the right people, and outlook mails[2] are sent to remind approve or reject to act promptly to requester.

The flowchart illustrates the user journey within an application integrated with PowerApps and Power Automate [5]. The process starts with the user logging in using their Microsoft account [5]. After entering their username and password, the microsoft verifies the credentials. Upon successful login, the user can link in payment request approval system of PowerApps and the user is directed to the Welcome Screen, where the application checks their permissions. Based on these permissions, the user can access different parts of the application: Dashboard Screen, Request Screen, or Attachment Screen.

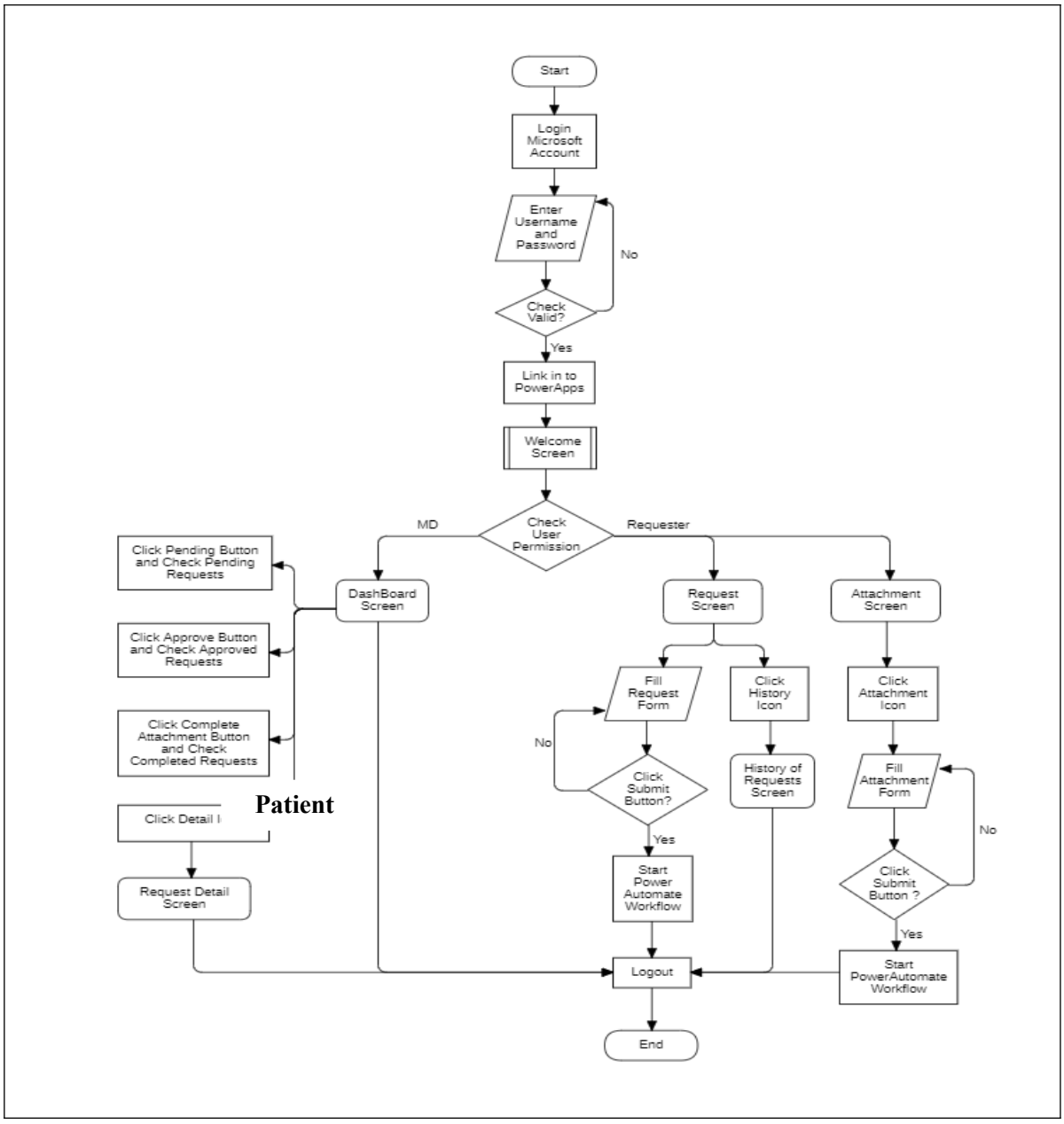

Figure 1: System Flow Diagram

Accepted Date: 25.10.2024
www.ucsh.edu.mm

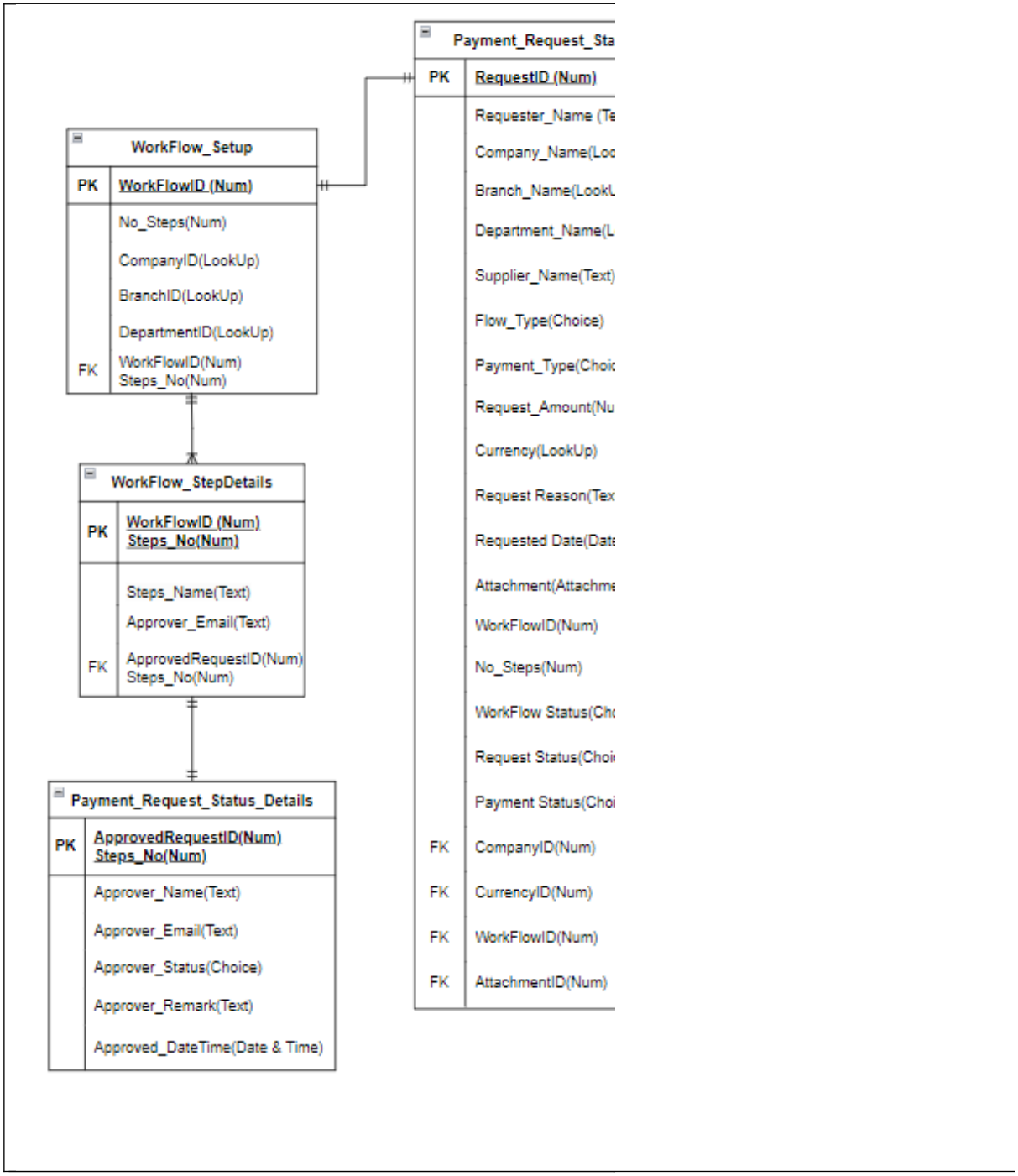

Figure 2: Database Design

On the Dashboard Screen, users can view and manage requests by checking pending, approved, and completed requests, and they can also view detailed information about specific requests. If the user needs to submit a new request, they navigate to the Request Screen, where they fill out a request form. Submitting this form triggers [4] a Power Automate workflow to handle the request. The user can also view their request history from this screen.

For managing documents, the user can go to the Attachment Screen, where they can attach files by filling out and submitting an attachment form, which also triggers a workflow upon submission. At any point, the user can log out, ending their session. This flowchart

Accepted Date: 25.10.2024
www.ucsh.edu.mm

provides a clear and structured pathway for users to interact with the application, ensuring that each action is linked to a corresponding automated process, making the overall experience efficient and organized.

RESULTS

After the system is developed, it must undergo thorough testing to ensure it is free of bugs. If any issues or errors are detected during this phase, the developer will need to promptly address and resolve them. Testing is the process of evaluating the software to detect discrepancies between the given input and the expected output. This step is essential for assessing the system's quality, serving as both a validation and verification process. System testing is not solely the responsibility of the development team; it also involves collaboration with external stakeholders to validate system acceptance. Figure 3 illustrates the implementation of the payment approval system.

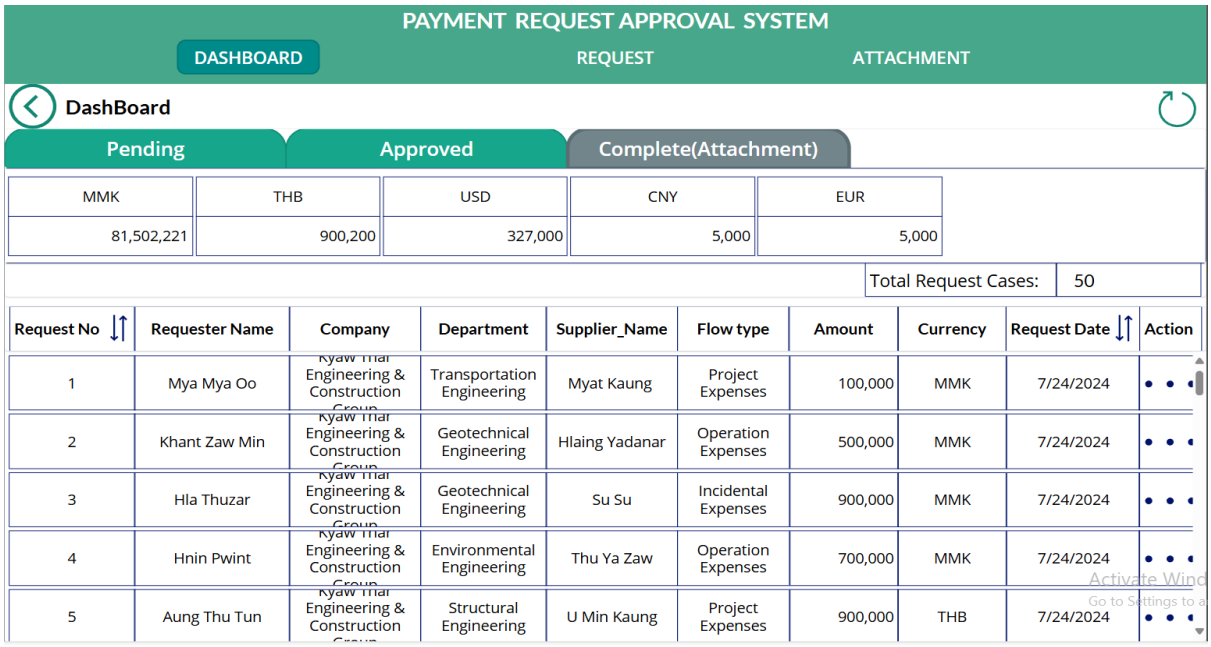

Figure 3: Implementation of this system

CONCLUSION

In today's world, technological advancements have transformed our lives, making tasks more convenient and comfortable. People are increasingly shifting from manual processes to digital solutions. Instead of recording information in books, they use software for data storage and retrieval. These technologies provide security, user-friendly interfaces, low costs and maintenance, and reduced traffic and queuing times. This project aimed to streamline the approval process for financial transactions. Many organizations face challenges with manual approval processes, which can be time-consuming and prone to errors. The system will provide employees with an intuitive and user-friendly interface, eliminating the need for lengthy waits and manual data entry. It will enable faster data entry, efficient searching, and comprehensive income summaries. The system will enhance the

Accepted Date: 25.10.2024
www.ucsh.edu.mm

efficiency and accuracy of payment request approvals, ensuring a smoother and more reliable process for handling financial transactions.

REFERENCES

[1] Microsoft Coporation,James Phillips and Charles Lamanna Microsoft Learn, https://learn. microsoft. com/en- us/power-apps/, November 30,2015.

[2] Microsoft Coporation,Outlook help & learning //microsoft.com, January 16, 1997.

[3] Microsoft Coporation,Jeff Teper,SharePointdocumentation|Microsoft Learn,March 28,2001.

[4] James Phillips,Official Microsoft Power Automate documentation - PowerAutomate | Microsoft Learn, October 31,2016

[5] Reza Dorrani, Powerapps full training course, youtube.com/ rezadorrani? sub_ confirmation=1, September 2,2023

Accepted Date: 25.10.2024
www.ucsh.edu.mm