

内製システムのためのシステムモジュール開発

前田 悠作[†] 山田 哲[‡] 浅木森 浩樹[‡] 末廣 紀史^{††} 武田 啓之^{††} 八重樫 理人[†]

香川大学創造工学部[†] 株式会社リコー[‡] 香川大学情報メディアセンター^{††}

1. はじめに

ローコード・ノーコードプラットフォームは、システム開発における「ニーズの曖昧さ」、「短期間で変化する要求」に対応するための手段として、DX 推進での活用が広がっている[1]。ローコード・ノーコードプラットフォームによる開発は、開発工数を大幅に削減し、リーン・スタートアップ[2]で提唱される実用最小限の製品である MVP (Minimum Viable Product) [3]の「検証」までの時間を大幅に短縮する効果が見込めることから、要件と開発されるシステムの粒度的ギャップが生じにくく、MVP を特定したシステム開発に有効であるとの報告[4]も存在する。

香川大学は、DX を推進すべく 2021 年 5 月に DX 推進戦略「デジタル ONE 戦略」を策定するとともに、香川大学情報メディアセンターは、DX 推進戦略「デジタル ONE 戦略」の実現に向けて「DX 化推進部門」を組織した。また同メディアセンターは、同じタイミングで「DX ラボ」も組織した[5]。「DX ラボ」は、情報技術を学ぶ香川大学の学生を中心に構成され、ローコード・ノーコードプラットフォームとして注目されている Microsoft Power Platform[6]を導入し、香川大学の DX 推進に資する業務システムの内製開発に取り組んでいる。香川大学では、2021 年 5 月から 2023 年 1 月の約 1 年半で 50 を超える業務システムが内製開発され、それらは実際に香川大学で実運用されている。

香川大学で開発された業務システムでは、多くの共通の機構（フロー）が使われているが、その都度それらフローを 1 から開発しなくてはならず、それが開発の生産性低下を招いていた。またそれら共通の機構で一部改修や不具合が発生しても、それぞれのシステム毎に個別に改修をおこなわなければならない、運用や保守性の低下も招いていた。

本研究では香川大学における内製システムの生産性・保守性向上に向けたシステムモジュール

開発をおこなう。本論文では、香川大学における内製システムのシステムモジュール開発事例について述べる。

2. 内製システムのためのシステムモジュール開発

本章では、内製システムのためのシステムモジュール開発の事例について述べる。

本研究では、Microsoft 社の提供する Microsoft Power Platform を用いて開発する。Microsoft Power Platform は、Microsoft Power Apps, Microsoft Power Automate, Microsoft Power BI, Microsoft Power Pages, Microsoft Power Virtual Agents の 5 種類のサービスから構成される。本研究では、Microsoft Power Automate, Microsoft Power Apps, Microsoft Dataverse, Microsoft Power BI を連携させることにより、システムのモジュール開発をおこなう。本研究では Power Automate で開発された香川大学の内製開発された業務システムを取り扱う。

図 1 は実際の内製システムで使われているファイルアップロードのフローを示している。図 1 のフローは、Microsoft Forms からアップロードされたファイルを Microsoft SharePoint で構築されたデータベースに保存することを意味している。このフローは、はじめに Forms から送信するファイルの拡張子を取得、送信後のファイル名を指定し、Microsoft SharePoint で構築されたデータベースにファイルを格納している。Microsoft Forms からアップロードされたファイルを Microsoft SharePoint で構築されたデータベースに格納するフローについては、香川大学で開発された多くの業務システムで実装されており、その仕組みを開発するには上記で述べたフローをゼロから開発する必要があった。本研究では、Microsoft Forms からアップロードされたファイルを SharePoint のデータベースに格納するフローをモジュール化した「Forms ファイルアップロードモジュール」を開発した。図 2 は、開発した「Forms ファイルアップロードモジュール」を示している。「Forms ファイルアップロードモジュール」は、Microsoft Forms からアップロードされたファイルをデータベースに保存する一連のフローを一つのシステムモジュールとしている。

Development of System Module for In-House Business Systems

[†]Yusaku Maeda [‡]Satoru Yamada [‡]Hiroki Asakimori

^{††}Norifumi Suehiro ^{††}Hiroyuki Takeda [†]Rihito Yaegashi

[†]Faculty of Creative Engineering, Kagawa University

[‡]Ricoh Co., Ltd.

^{††}Information Media Center, Kagawa University

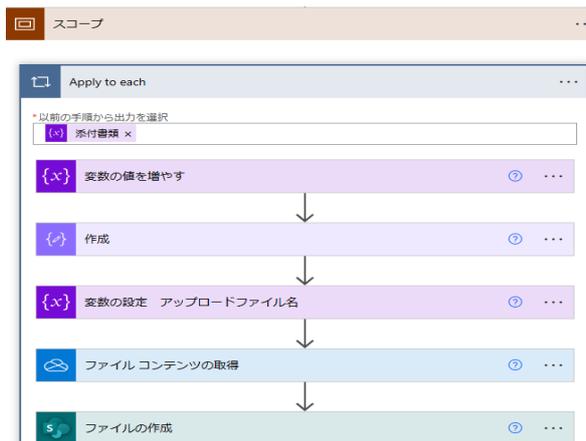


図1 内製システムでのファイルアップロードのフロー

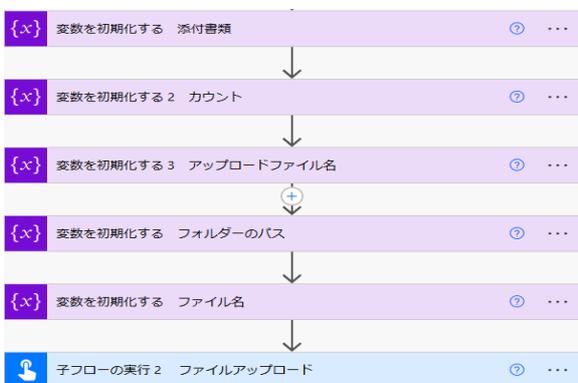


図2 Formファイルアップロードモジュール

香川大学で開発した「発明相談/事前評価書承認システム」では、本研究で開発した「Forms ファイルアップロードモジュール」が用いられている。「Forms ファイルアップロードモジュール」を用いることでフロー数が17%削減された。香川大学では、「Forms ファイルアップロードモジュール」が開発工数の削減に貢献していることが明らかとなった。

3. まとめ

本研究では、内製システムのためのシステムモジュールの開発について述べた。「Forms ファイルアップロードモジュール」などシステムモジュールを用いることで開発工数を削減する効果があることがわかった。香川大学では「Forms ファイルアップロードモジュール」だけでなく、生産性や保守性にむけて他のシステムモジュール開発についても検討している。

システムモジュールなどの部品を用いることで、不具合発生時の対応など保守工数削減の効果も期待できる。今後システムモジュール開発の効

果についても測定する予定である。

参考文献

- [1] 情報処理推進機構 (IPA), DX 白書 2021, <https://www.ipa/go.jp/files/000093706.pdf> (2023年1月12日参照)
- [2] エリック・リース, 井口 耕二, 伊藤 穰一: リーンスタートアップ ムダのない起業プロセスでイノベーションを生み出す, 日経 BP, 2012.
- [3] 坂口 和敏, 小林 延至, 白坂 成功: 外部制約と内部制約の観点に基づく階層型サービスデザインモデル, サービスロジー論文誌, pp. 1-13, 2021.
- [4] Microsoft Corporation: Power Apps vs 従来のアプリ開発のアプローチ -Power Apps. <https://docs.microsoft.com/ja-jp/powerapps/guidance/planning/app-development-approaches> (2023年1月12日参照)
- [5] 石川 颯馬, 山田 哲, 末廣 紀史, 武田 啓之, 國枝 孝之, 米谷 雄介, 後藤田 中, 浅木森 浩樹, 八重樫 理人: 香川大学の DX 推進環境の整備と DX 推進の取り組みについて—業務システムの内製開発による DX 推進—, 情報処理学会論文誌教育とコンピュータ (TCE), vol. 8, No. 1, pp. 88-99, 2022.
- [6] Microsoft: Microsoft Power Platform, Microsoft. <https://www.microsoft.com/ja-jp/biz/dynamics/power-platform.aspx> (2023年1月12日参照)