

UiPathを使用した RPA業務自動化デモと考察

2021年3月8日

WGメンバー 岡田誠司

(株式会社オーシス代表取締役 ITコーディネータ)

UiPathでの自動化例

1. 決められた操作で、頻度、量が高い作業

■作成例：メール送受信

2. 単純で面倒な作業

■作成例：売上集計

3. 定期的に発生する業務

■作成例：月末の経費精算(交通費)

デモ内容作成パソコンスペック表

- スペック表
- 【最小要件】 CPU：1.4GHz 32-bit (x86) / RAM：4GB
- 【推奨要件】 CPU：デュアルコア 1.8 GHz、64 ビット / RAM：4GB
- 【実際の開発環境】 Core i7-10710U (1.1GHz)
RAM：16GB

UiPath開発環境について

- 推奨環境相当のマシンで開発し開発することは可能だった。
- しかしデバッグ機能の動作が遅すぎ快適な環境では開発できなかった。
そのため上記環境で開発した。
- 実際のスペック消費状況を監視している限りCPUではなく、メモリの容量のウェイトが高い
- 安定した開発環境を動かすことに多くのメモリが必要であるという結果が出た

メール送受信1

目的：営業支援の一環で週ごとに変わる在庫価格・在庫有無を確実に代理店へ送付する

◆デモ前提条件

- 送付先については固定とする
- 送付先が一覧になっているエクセルファイルはすでにあり、その内容の整合性は必ず正しいものとする
- 所定のエクセルファイルに商品名/在庫価格/在庫有無は記載されているとする
- メール送信についてはバウンスメールの判定は行わない

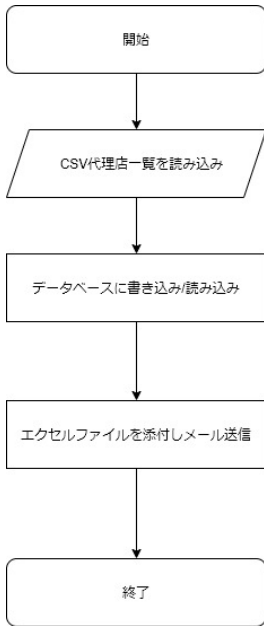
メール送受信2

CSVの代理店一覧を読み込み



会社の特定のメールアドレスからファイルを添付し
上記の代理店一覧に送信する

メール送受信 3



メール受送信 4

- デモ

メール受送信実際のソフトとしては必要な処理 1

- デモでは行わなかったが実際のソフトとしては必要な処理

1. CSV読み込み時のエラーチェック

メールアドレスの全角半角判定やメールアドレスの形式をチェックする処理が必要である

この中でもメールアドレスの形式の判定については、非常に奥が深くRFC5322で定義されているメールアドレスの書式を完全にサポートすることは簡単ではありません

後述するバウンスメールの判定を強化するか、一度以上営業がメールをやり取りした実績のあるメールアドレスの未登録するなど運用の工夫で逃げるなどもありだと思います

メール受送信実際のソフトとしては必要な処理 2

- デモでは行わなかったが実際のソフトとしては必要な処理

2. メール送信時のバウンスメールの判定が必要である

バウンスメールとはメールが送信できなかった場合の原因が書かれています

大きくはソフトバウンス2020/03/08（一時的な問題でのエラー）

ハードバウンス（時間がたっても解決しないエラー）の二つがあります

メール受送信考察

- メールは届くか届かないかの判定が非常に難しい
- 自社のメールアドレスが相手側のファイアウォールにはじかれるなどバウンスメールだけでは内容が判定できないことも多い
- UiPathで作ることは可能だが、あくまでBCCに一括登録や手で一件一件送付することよりはマシということであり、確実に届けたい場合は何か違う対応を考慮する必要もある

売上集計1

目的：本社/支社がある会社で、所定のフォーマットで入力された毎日のデータを集計し、月初の役員会議で使う

◆前提条件

- 本支店数は固定
- 集計エクセルファイルはすでにあり、その内容の整合性は必ず正しいものとする（入力内容の正誤判定は行わない）
- 所定のフォーマットで入力されたファイルは所定の位置にあるものとする
- 出力はCSVとし、あくまで集計のフォロー以上の機能は搭載しない

売上集計2

所定のエクセルファイルで送受信されたデータを読み込む



データベースに書き込み/読み込み

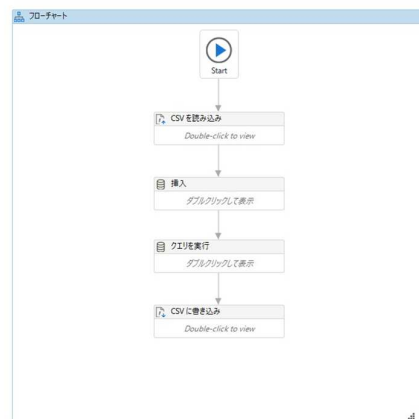


各支店ごとに集計を行い



CSVファイルに支店名/日付/売上としてデータを書き込む

売上集計3



売上集計4

- デモ

売上集計実際のソフトとしては必要な処理1

1. 入力元であるエクセルデータの整合性を判定する必要がある

これについては

- エクセル側でエラー判定を行い精度を高める
- UiPath側でエラー判定を行い精度を高める

の二つの対処方法があると思います

理想は両方行うことになるとは思いますが、あくまで入力データがエクセルなので、セルへの入力規制、入力規則などを強化し、出来る限り入力データの精度を保つことが安定運用につながると思います

売上集計実際のソフトとしては必要な処理2

2. 運用上の問題

エクセルを入力元にする場合、ファイル名の命名規約などを正しく入力してもらう必要がある

そのため入力する人間への教育が必要になる可能性が高い

売上集計考察

エクセルを入力元にする場合、ファイル名の命名規約などを正しく入力してもらう必要がある

そのため入力する人間への教育が必要になる可能性が高い

月末の経費精算(交通費) 1

目的：月末に社員へ立替え処理してもらっている電車代を計算する

◆前提条件

- 集計エクセルファイルはすでにあり、その内容の整合性は必ず正しいものとする（入力内容の入力チェック/正誤判定は行わない）
- 所定のエクセルファイルに社員別でどこからどこまでを移動したかは記載されている
- その移動方法が最短かどうかの判定はおこなわない

月末の経費精算(交通費) 2

所定のエクセルファイルを読み込む



データベースに書き込み/読み込み



既存のAPIを使って集計



データベースに書き込み/読み込み



CSVファイルに支店名/日付/売上としてデータを書き込む

月末の経費精算(交通費) 3



月末の経費精算(交通費) 4

- デモ

該当処理運用上の問題

- 使用するAPI そのものが有償か無償かなどもUipathで作成可否や難易度もかなり変わってくる

月末の経費精算(交通費) 考察

- 非常にAPIへの依存度が高い
- そのためAPIの仕様変更によっては完全に作り直しをするなどのリスクがあげられる
- ただしそのリスクを飲めるのであれば、毎月月末に必ず発生する交通費の清算という業務を自動化できることはRPA本来の目的と合致するものであると考えられる

考察1

- まずUiPathでもWinActorでもRPAを使用するために必要なことは、業務をフロー化できるかどうかだと考えている
- そして一般的なシステム開発と同じで業務が煩雑であれば煩雑であるほど作成難易度はあがる
- 現状でユーザーが業務フローを紙に書けるレベルに達していない場合、ツールではなくその部分からの人材育成が必要になると予想される
- こういった部分に関しては一般的なスクラッチ開発であろうが、RPAであろうが同じであり、開発リスクもまったく同じと考えられる。

考察2

- さらにそれとは別にツールに関しては教育コスト（費用・時間）が発生する
- そのため一般的な総務部/人事部などの内勤ではなく、システム部があるような企業が導入に適しているのではないか
- ないしはフロー図作成から開発までITC（ないしはフロー図作成をITCが行い、実際の開発は協力会社）が外部委託で受注作成するかなどではないかと考える
- ただし一般的なシステム開発であれば図示されず実際のコードが変更された際に変更されない設計書などの問題があるが、RPAソフトではその部分に関しては図示されるため比較的作成者退任による引継ぎなどは平易であると考える