優秀賞

外出先からメールでリアルタイム在庫を問い合せ

島根 英行 様
シルフ


シルフ

業務内容は IBM i を中心としたシステム開発。販売管理をメインに、オフコンや PC サーバーからのリプレース提案を得意としている。

1.業務課題

営業担当者が外出先から商品在庫を確認したい、という要望はよくあるが、Web サーバーなどの追加投資なしで、営業担当者が既に持ち歩いている携帯電話だけで実現できないか、という課題があった。そこで、IBM i 基幹システムに保有するリアルタイムの商品在庫を、携帯メールで手軽に照会できる仕組みを検討することとした。

2.技術課題

Web システムを構築することなく、メールだけで在庫照会を実現することが技術的な課題であった。その解決のための基本的な仕組みは、以下の通りである。

[1]ユーザーは、照会したい商品の商品コードを記載したメールを所定のアドレス宛に送信

[2]Delphi/400 プログラムは以下の処理を行う

(1) メールの待ち受け:一定時間 (5 分) 間隔でメールサーバーをチェック
(2) 受信メールの解析：在庫照会メールが存在した場合、メール本文から商品コードを抽出
(3) 在庫データの取得：IBM i にログインし、指定された商品コードの在庫 DB 照会などを行う
(4) 結果の自動返信：結果をメールに記載（または添付）して差出人のユーザー宛に返信。また、なんらかのエラー発生時は、システム管理者にメールで通知

[3]ユーザーは返信メールにより、在庫などを確認

本稿では、特に（a）メール送受信機能の実現、(b) 本来不定型のメールをシステムへの入力データとして扱う方法を技術課題として紹介する。

3.技術課題の解決策

(a) メール送受信機能について

(1) メール受信：コンポーネント（Indy コンポーネント POP3）を使用して実現。【ソース 1】
考慮点：照会メールの差出人が正当なユーザーであるかチェックを行う
(2) メール送信：コンポーネント（Indy コンポーネント SMTP）を使用して実現。【ソース 2】

(b) 本来不定型のメールをシステムへの入力データとして扱う方法

在庫問い合わせメールの本文はシンプルに「No.（通番）＋商品コード」のみで送信することとした。

入力テキスト例　0：4520179484556
（コロンの前半が No.、後半が商品コード）

No. と商品コードの間は「:」（コロン）

【ソース1】

ソース１　　受信メールの確認

```
procedure TfrmSendMail.chkmaile;
var
  //中略
begin

  try
  //中略

    //各パラメータを設定し、メールサーバーからメールを取得
    IdPOP31.Host          := Edit1.Text;
    IdPOP31.Port          := 110;
    IdPOP31.Username    := Edit2.Text;
    IdPOP31.Password    := Edit3.Text;
    IdPOP31.Connect;
    TStest:= TStringList.Create;
    Msg :=    TIdMessage.Create(Self);
    if IdPOP31.CheckMessages>0 then begin
      i := 1;
      IdPOP31.Retrieve(i, Msg);          //受信メールを取得
      Edit4.Text := Msg.Subject;         //件名取得
      Edit5.Text := Msg.From.Address; //差出人
      //差出人の正当性を確認する（多くの人を管理する時は、データベース化する）
      if Edit5.Text = 'XXXXX@AAAA.ne.jp' then chkA := True;
      TStest.AddStrings(Msg.Body);       //メール本文
      Memo1.Lines.Text:= (jconvert.ConvertJCode(TStest.Text, SJIS_OUT));
      IdPOP31.Delete(i);     //サーバーの受信メールを削除
    end;
  finally
    IdPOP31.Disconnect;
    Msg.Free;
    TStest.Free;

    //以下省略       IBM i へのログイン、在庫等データの取得へ続く
```

で区切るルールとし、プログラムで読み取っている。【ソース 3】

4.業務課題解決と効果

実際には、在庫数に加えて、出荷データ、入庫データ、棚卸しデータなどの問い合わせが可能な仕組みとし、問い合わせ件数が多い場合は、結果を csv ファイルにして添付ファイルで返信するなどの工夫を行っている。これにより、当初の目的通り、外出先からリアルタイム在庫残高などを照会する仕組みを、追加投資なしで実現することができた。

M

【ソース2】

ソース 2　　ユーザーへのメールの送信

```
function TfrmSendMail.chkZaiko(chkZA002 :String): Boolean;
var
        //中略
begin
  Result := False;

  try
    //SQL で在庫ファイル取得　（中略）
      chkkazu := DM.Query1.FieldByName('ZA014').AsInteger;      //在庫数セット
      chkMei := DM.Query1.FieldByName('IM011').AsString;        //商品名セット
    end;
        //中略

  try
    SMTP := TIdSMTP.Create(nil);          // 以下、送信メールを定義
    msg := TIdMessage.Create(SMTP);
    msg.From.Name := edtFromName.Text;
    msg.ContentType := 'text/plain';
    msg.CharSet := 'ISO-2022-JP';
    msg.ContentTransferEncoding := '7bit';
    msg.Recipients.EMailAddresses := edtToMail.Text;  //宛先
    msg.From.Text :=  'aaaa@ccc.co.jp';       //差出人
    edtSubject.Text := 'ZAIKO';      //在庫
    memBody.Lines.Add(chkMei+'の在庫数は、'+IntToStr(chkkazu)+'  です'+#13);

        //中略
    // メッセージを送信
    try
      SMTP.Host       := edtHost.Text;        //SMTP サーバー名
      SMTP.Port       := 0;                   //ポート番号
      SMTP.Username := edtUserName.Text;  //SMTP ユーザー名
      SMTP.Password := '';                    //SMTP パスワード
      SMTP.Connect;
      SMTP.Authenticate;
      SMTP.Send(Msg);
      Memo2.Lines.add(edtToMail.Text + ' 送信しました。');
      //以下略
```

【ソース3】

ソース 3　　メール本文から「商品コード」と「項目 No.」を取得

```
{************************************************************
  関数名　　：ComBoGet_Text
  機能　　　：コンボボックス内の項目テキストを取得
  引数[I/O]：コンボボックスのテキスト
  戻り値　　：コンボボックスのテキスト内の「商品コード」部分
  備考　　　：入力文字列　0:4520179484556（No.＋商品コード）
************************************************************}
Function ComBoGet_Text(Comb_Text:String):String;
Var
  i:Integer;
begin
  i := Pos(':', Comb_Text)+1;    //コロン(:)より後の文字列を抽出
  result:=Copy(Comb_Text,i, Length(Comb_Text));  //長さを多くとる
end;

{************************************************************
  関数名　　：ComBoGet_No
  機能　　　：コンボボックス内の項目№を取得
  引数[I/O]：コンボボックスのテキスト
  戻り値　　：コンボボックスのテキスト内の「項目 No.」部分
  備考　　　：入力文字列　0:4520179484556　（No.＋商品コード）
************************************************************}
Function ComBoGet_No(Comb_Text:String):String;
Var
  i:Integer;
begin
  i := Pos(':', Comb_Text)-1;     //コロン(:)より前の文字列を抽出
  result:=Copy(Comb_Text,1,i)
end;
```