

Kreditoren-Management mit ArchivistaDMS

Egg, 3. Dezember 2018: Die letzten Monate waren von satter Arbeit bei Kunden geprägt. Viel Zeit, um die ArchivistaBox zu erweitern, blieb dabei nicht. Seit einigen Tagen steht das Release 2018/XI zur Verfügung. An erster Linie steht optional ein neues Kreditoren-Management-Modul bei ArchivistaDMS zur Verfügung. Weiter werden neu die Fujitsu fi-7140 und fi-7240 unterstützt. Nicht zuletzt wurden aber einige Bugs bei ArchivistaERP und ArchivistaDMS gelöst, auch dies gehört zur Produktpflege. Nachfolgend wird das Kreditoren-Management vorgestellt.



Schritt 1: Skript aktivieren

Im Rahmen eines neuen Projektes ging es darum, dass alle eingehenden Kreditoren-Rechnungen elektronisch verarbeitet werden können. Dabei gab es mehrere Rechtestufen zu implementieren. Das Sekretariat erfasst die Rechnungen (Scannen oder Datei-Upload) und weist diese den Abteilungen zu. Dort können diese begutachten und elektronisch visiert werden. Die Geschäftsleitung sieht beim Anmelden alle visierten Posten und autorisiert diese für die Buchhaltung, womit die Rechnungen zur Zahlung freigegeben sind. Tönt einfach, und ist es auch. Mit dem nachfolgenden Code wird ArchivistaDMS um ein komplettes Kreditoren-Management erweitert:

```
#!/usr/bin/perl
```

```
use lib qw(/home/cvs/archivista/jobs);
use AVJobs;
my $host = shift;
my $db = shift;
my $user = shift;
my $pw = shift;
my $doc = shift;
my $dbh=MySQLOpen($host,$db,$user,$pw);
checkChange($dbh,$doc,$user);
$dbh->disconnect();

sub checkChange {
    my ($dbh,$doc,$user) = @_;
    my $docit = "where Laufnummer=$doc " .
        "and DokTyp2Nr=4 and (Gesperrt='' or Gesperrt is null)";
    my $sql = "select Laufnummer from archiv $docit";
    my ($lnr)=$dbh->selectrow_array($sql);
```

```

if ($lnr>0) {
  if ($user eq "buha") {
    $sql = "select Sachbearbeiter,SachbearbeiterHidden,".
      "GeschaeftsleitungVisum from archiv $docit";
    my ($sach,$hidden,$visum) =
$dbh->selectrow_array($sql);
    if ($visum==0) {
      if ($sach ne $hidden) {
        $sql = "update $dbh.archiv set
Sachbearbeiter=".$dbh->quote($sach).
        ",SachbearbeiterHidden=".$dbh->quote($sach).
        ",SachbearbeiterVisum=0,SachbearbeiterDatum=NULL
        ".
          "where Laufnummer=$doc";
        $dbh->do($sql);
      }
    } else {
      if ($sach ne $hidden) {
        $sql="update archiv set
Sachbearbeiter=".$dbh->quote($hidden)." $docit";
        $dbh->do($sql);
      }
    }
    $sql = "select BuchhaltungVisum,BuchhaltungDatum,".
      "GeschaeftsleitungVisum from archiv $docit";
    my ($vis,$dat,$gl) = $dbh->selectrow_array($sql);
    if ($gl==1) {
      if ($vis==1 && $dat==0) {
        $sql = "update archiv set
BuchhaltungDatum='".SQLStamp1().
        "' where Laufnummer=$doc";
        $dbh->do($sql);
      } elsif ($vis==0 && $dat>0) {
        $sql = "update archiv set BuchhaltungVisum=1 where
Laufnummer=$doc";
        $dbh->do($sql);
      }
    } else {
      if ($vis==1) {
        $sql = "update archiv set BuchhaltungVisum=0 where
Laufnummer=$doc";
        $dbh->do($sql);
      }
    }
  } elsif ($user eq "gl") {
    my $sql = "select GeschaeftsleitungDatum from archiv
$docit".
      " and GeschaeftsleitungVisum=1 and
BuchhaltungVisum=0";
    my @row = $dbh->selectrow_array($sql);
    if (int($row[0])==0) {
      $sql = "update archiv set
GeschaeftsleitungDatum='".SQLStamp1().
      "' where Laufnummer=$doc";
      $dbh->do($sql);
    }
  } else { # Sachbearbeiter

```

```

my $sql = "select Sachbearbeiter from archiv $docit";
my @row = $dbh->selectrow_array($sql);
if ($row[0] eq "$user") {
    my $sql = "select
SachbearbeiterVisum,SachbearbeiterDatum ".
    "from archiv $docit and GeschaeftsleitungVisum=0
and ".
    "BuchhaltungVisum=0";
    my ($vis,$dat) = $dbh->selectrow_array($sql);
    if ($vis==1 && $dat==0) {
        $sql = "update archiv set
SachbearbeiterDatum='".SQLStamp1().
        "' where Laufnummer=$doc";
        $dbh->do($sql);
    } elsif ($vis==0 && $dat>0) {
        $sql="update archiv set SachbearbeiterVisum=1
where Laufnummer=$doc";
        $dbh->do($sql);
    }
    } else {
        $sql="update archiv set SachbearbeiterVisum=0 where
Laufnummer=$doc";
        $dbh->do($sql);
    }
}
}
}

sub SQLStamp1 {
    my @t = localtime( time() );
    my ( $stamp, $y, $m, $d, $h, $mi, $s );
    $y = $t[5] + 1900;
    $m = $t[4] + 1;
    $m = sprintf( "%02d", $m );
    $d = sprintf( "%02d", $t[3] );
    $stamp = $y . "-" . $m . "-" . $d . " 00:00:00";
    return $stamp;
}

```

Das Skript ist unter dem Namen der Datenbank (z.B. archivista.pl) in WebAdmin unter „Jobs administrieren“ zu aktivieren. Damit das Skript arbeitet, ist plus/minus die folgende Maske zu erstellen:

Schritt 2: Maske anpassen

The screenshot shows the 'Bearbeiten' (Edit) tab in the ArchivistaBox WebAdmin interface. The form is organized into two columns of input fields. The left column includes fields for 'Akte', 'Freitextabfrage', 'Eigentümer', 'AdrName', 'AdrNr', 'DoctNr/RgNr', 'Sachbearbeiter', 'Sachb. Visum', 'Sachb. Datum', 'GL Visum', and 'Notiz'. The right column includes fields for 'Archiviert', 'GL Datum', 'Buha Visum', 'Buha Datum', 'Projektlr', 'Projekt-Sachb.', 'Dateiname', and 'Dok-Typ'. Each field is represented by a text input box. The interface has a light blue header with tabs for 'Ansicht', 'Suchen', 'Ersetzen', and 'Bearbeiten'.

Schritt 3: SQL-Definitionen erstellen und zuweisen

Damit die einzelnen Benutzer/innen beim Einstieg in ArchivistaDMS die korrekten Dokumente vorfinden, müssen in WebAdmin entsprechende SQL-Definitionen erfasst werden:

Sekretariat/Buchhaltung: DokTyp2Nr=4 and BuchhaltungVisum=0 and (((Sachbearbeiter is Null or Sachbearbeiter=») or (GeschaeftsleitungVisum=1 and BuchhaltungVisum=0))) order by Datum desc, Laufnummer desc

Sachbearbeiter: DokTyp2Nr=4 and SachbearbeiterVisum=0 and GeschaeftsleitungVisum=0 and Sachbearbeiter='[user]\' order by Datum asc

Geschäftsleitung: DokTyp2Nr=4 and ((Sachbearbeiter<>»» and GeschaeftsleitungVisum=0)) order by Datum asc

Die entsprechenden SQL-Definitionen sind den Benutzer/innen zuzuweisen — und schon ist ArchivistaDMS um ein vollwertiges Kreditoren-Management erweitert.

Fazit: Automatisierung leicht gemacht

ArchivistaDMS lässt sich mit ‚Jobs administrieren‘ einfach über WebAdmin automatisieren. Im obenstehendem Beispiel wurde die Programmiersprache Perl verwendet. Ebenfalls verwendet werden können die Skriptsprachen PHP und Python. Weitere Sprachen können optional ebenfalls eingebunden werden.

Wem die obenstehenden Zeilen ‚spanisch‘ vorkommen, darf sich gerne an uns wenden, entsprechende Module werden für Kunden gerne auf jeder ArchivistaBox implementiert.