



GPLⁱ



Copyright © 2011 – 2015 JetLab S.r.l. Tutti i diritti riservati

 è un marchio di JetLab S.r.l.

Microsoft, Windows, Windows NT, Windows XP, Windows Vista, Windows 7 e il logo di Windows sono marchi o marchi registrati di Microsoft Corporation in U.S. e/o altri paesi.

Java e tutti i marchi base di Java e loghi sono marchi o marchi registrati di SunMicrosystems, Inc. in the U.S. e/o altri paesi.

UNIX è un marchio registrato in U.S. e in altri paesi rilasciati esclusivamente Attraverso X/Open Company Limited.

IBM i5 iSeries 400, OS/400, IBM i, , i5/OS per Power Systems, AS/400 sonomarchi di IBM Corporation, U.S.

Altri nomi di aziende, prodotti e servizi che devono essere rappresentati da un doppio asterisco (**), possono essere marchi o marchi di servizio o altro.

Sommario

Descrizione generale	3
Requisiti	3
Prova l'esercizio	3
Analisi delle componenti	5
Modello PDF	5
Sorgenti	6
Compilazione	8

Descrizione generale

Questa White Paper mostra come creare un programma RPG che mediante il prodotto **GPL4i** generi un documento Pdf contenente tra l'altro un'Immagine e un Barcode.

Il documento Pdf sarà generato su IBM i.

Requisiti

Questi i requisiti che devono essere rispettati per poter completare con esito positivo le attività proposte nel documento

- Il prodotto **GPL4i** deve essere già installato e configurato sul sistema IBM i.
http://www.jetlab.com/downloads/gpl4i/GPL4i-V2R0M0-WP001_Installazione.pdf
- La libreria **GPL4i** User Tool deve essere installata sul sistema IBM i.
http://www.jetlab.com/downloads/gpl4i/GPL4i-V2R0M0-WP002_Esempi.pdf

Il documento è rivolto a programmatori IBM i con conoscenze del linguaggio RPG e ambiente ILE.

Prova l'esercizio

Questi i passi per attivare il programma RPG che genera il documento Pdf e verificarne il risultato

___ Passo 1

Collegarsi con l'utente R4SP020000 e attivare il runtime e la Device di default.

```
STRSBS JTJT040000  
STRR4DEV JTJT040000
```

___ Passo 2

Aggiungere la libreria GPL4IUT in lista librerie

```
ADDLIBLE GPL4IUT
```

___ Passo 3

Lanciare il programma PRDDTL

```
CALL PGM(PRDDTL) PARM('JTJT040000')
```

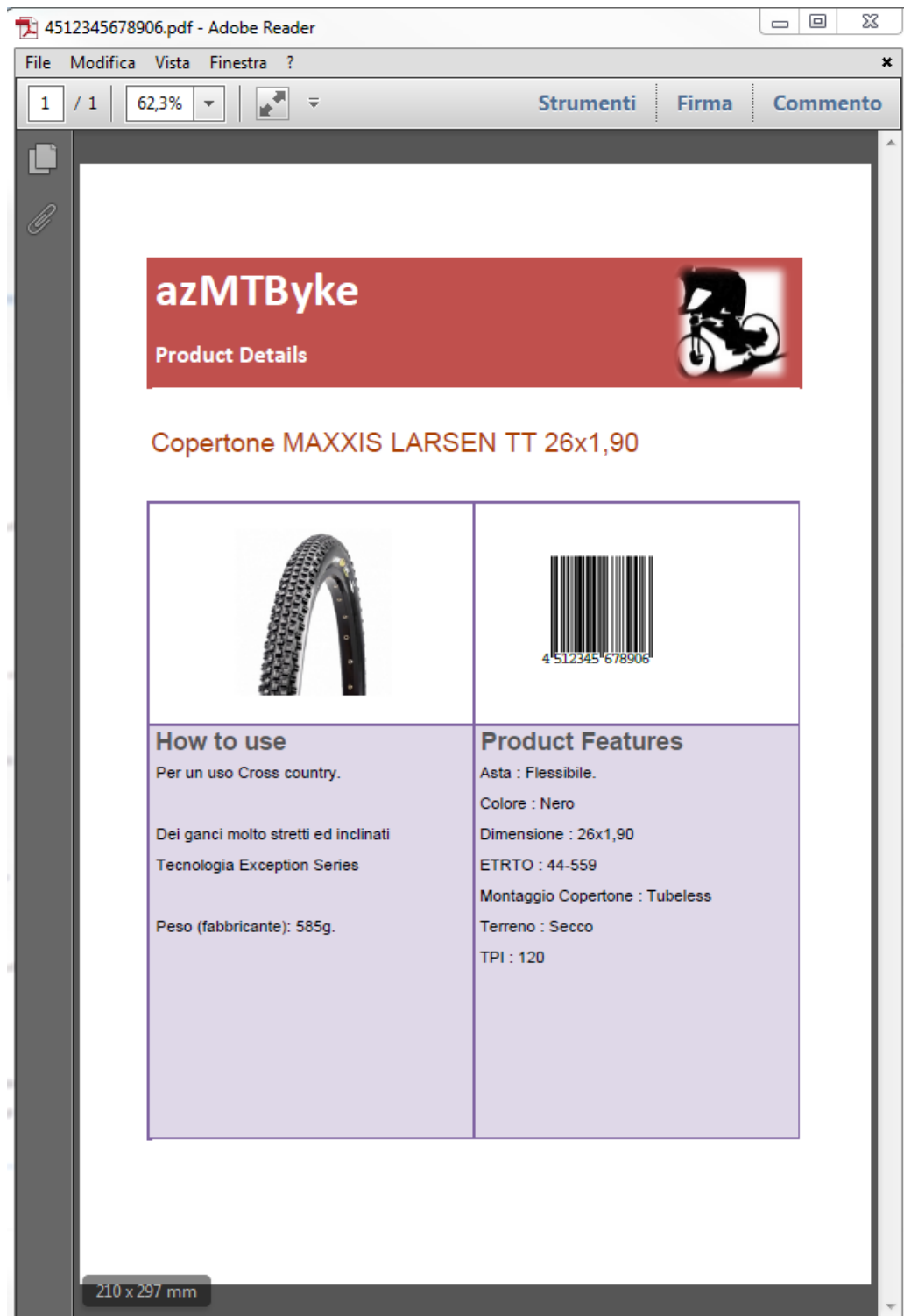
Passo 4

Verificare che il documento

/gp14iut/temp/4512345678906.pdf

sia stato generato. Aprire il documento con un Pdf Reader.

Questo il risultato



Analisi delle componenti

Per ottenere il documento PDF sono state utilizzate le seguenti risorse.

Modello PDF

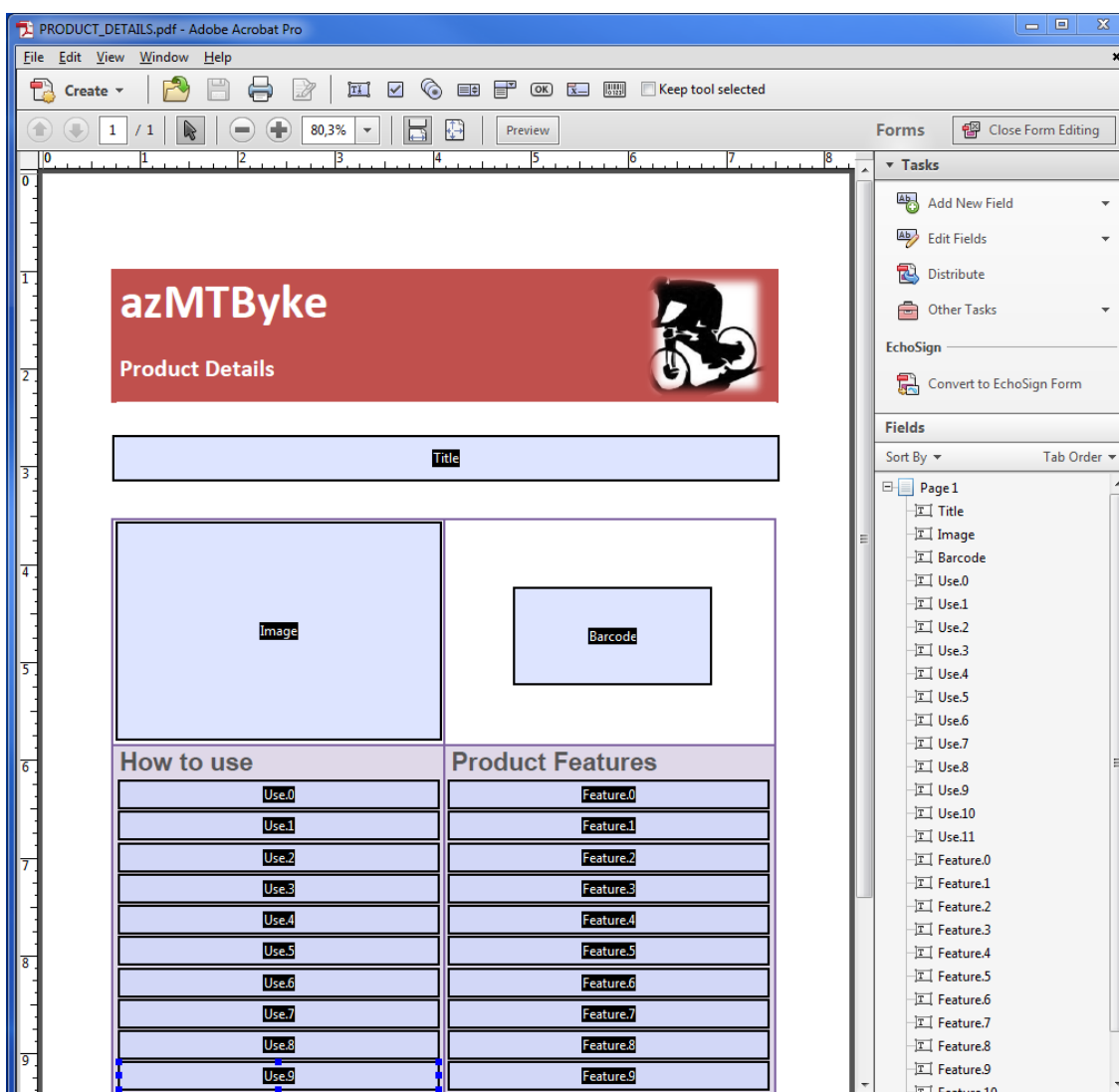
Il Form utilizzato è contenuto nel file

/gp14iut/PRODUCT_DETAILS.zip

e si chiama

PRODUCT_DETAILS.pdf

Come mostra l'immagine seguente è un Pdf che contiene i seguenti campi di tipo testo:



Il sorgente RPG

GPL4IUT/QRPGLESRC (PRDDTL)

farà riferimento a questi nomi per compilare il Form con i dati del programma.

Sorgenti

Il sorgente RPG

GPL4IUT/QRPGLESRC (PRDDTL)

svolge le seguenti operazioni per generare il documento:

- Include delle definizioni delle API GPL4i

```
D/copy *LIBL/QRPGLECPY,GPL4I
```

- Stabilisce il modello da generare

```
c          callp      GPLSetDataModel(  
c          'PRODUCT_DETAILS'+x'00'  
c          :'/gpl4iut/PRODUCT_DETAILS.zip'+x'00'  
c          )
```

- Apre un nuovo documento

```
c          callp      GPLOpenDoc
```

- Assegna un dato utente al documento (Product_code = 4512345678906)

```
c          eval       val = code  
c          callp      GPLAddDMInfo(  
c          'Product_code'+x'00'  
c          :%addr(val)  
c          :%len(%trim(val))  
c          :handle  
c          )  
c          callp      GPLSetFolder('Products'+x'00')
```

- Assegna un nome al file .pdf da generare e attiva l'attributo replace=true per sovrascrivere il documento se già presente

```
c          eval       val = 'true'  
c          callp      GPLAddSrvInfo(  
c          'save'+x'00'  
c          :'replace'+x'00'  
c          :%addr(val)  
c          :%len(%trim(val))  
c          )  
c          callp      GPLSetPdfName(  
c          '/GPL4IUT/temp/'  
c          +%trim(code)+'.pdf'+x'00'  
c          )
```

- Stabilisce quale Form deve compilare tra i Form disponibili nel modello

```
c          callp      GPLNewForm(  
c          'PRODUCT_DETAILS.pdf'+x'00'  
c          )
```

- Inizializza i campi del Form (ad esempio il campo Title)

```

c          eval      val='Copertone MAXXIS LARSEN TT 26x1,90'
c          callp     GPLSetTextFld(
c                  'Title'+x'00'
c                  :%addr(Val)
c                  :%len(%trim(Val))
c                  )

```

- Inizializza il campo Barcode di tipo EAN

```

c          eval      val = code
c          callp     GPLSetBarcode(
c                  'Barcode'+x'00'
c                  : 'EAN'+x'00'
c                  :%addr(val)
c                  :%len(%trim(val))
c                  )

```

- Inizializza il campo Image

Viene anche applicata una riduzione del 60% delle dimensione reale dell'immagine

```

c          eval      val = 'percente'+x'00'
c          callp     GPLSetFldAttr(
c                  'scale'+x'00'
c                  :%addr(val)
c                  )
c          eval      scale100 = 60.0
c          callp     GPLSetFldAttr(
c                  'scale100'+x'00'
c                  :%addr(scale100)
c                  )
c          eval      val
c                  = 'http://www.jetlab.com'
c                  + '/gpl4i/gpl4iut/copertone.png'
c          callp     GPLSetImage(
c                  'Image'+x'00'
c                  :%addr(Val)
c                  :%len(%trim(Val))
c                  )

```

- Termina la compilazione del Form

```

c          callp     GPLSetBreak

```

- Chiude il documento

```

c          callp     GPLCloseDoc

```

- Attiva la generazione del Pdf affidandola alla Device JTJT040000

Nel caso di errore la variabile gplErr conterrà le indicazioni sull'errore incontrato.

```

D gplErr      ds          likeds(GPLerr_T) based(gplErr_ptr)
D gplErr_ptr  s          *
...

```



```
c          if          GPLRun(device) = 0
c          eval          gplErr_ptr = GPLGetCurErr
c      gplErr.MSG      DSPLY
c          endif
```

Compilazione

Il sorgente PRDDTL viene compilato in Bind con i seguenti Service Programs

R4SP020000/GPL4I

ADDLIBLE JTJT040000

ADDLIBLE R4SP020000

ADDLIBLE GPL4IUT

CRTRPGMOD MODULE(GPL4IUT/PRDDTL) DBGVIEW(*ALL)

CRTPGM PGM(GPL4IUT/PRDDTL) BNDSRVPGM((GPL4I))