PASSEPARTOUT
Software e servizi gestionali

Corso Sprix Mobile Base

Gianluca Suzzi

OBBIETTIVO DEL CORSO

Obbiettivo del corso è quello di apprendere le conoscenze per l'utilizzo di PassBuilder quale strumento di produzione di App Mobile per dispositivi mobile Android e iOS:

- programmazione App Mobile
- definizione base dati e collegamento con gli archivi dell'installazione
- utilizzo sul dispositivo mobile

Quanto esposto in questo corso è presente in maniera molto più dettagliata nel manuale SprixMobile scaricabile dalla sezione Area Sviluppatori della vostra Area Riservata sul portale di Passepartout.

Prerequisito necessario per il corso: conoscenza di base del linguaggio SPRIX

PROGRAMMA CORSO

- Introduzione a Pass Mobile
  - Come sono strutturate le App di tipo mobile
- Fondamenti di programmazione Sprix-Mobile
  - Gestione degli archivi
  - Definizione Form
  - Gestione delle Liste
  - Inserimento di pulsanti
  - Inserimento di caselle di input
  - Panoramica sulle funzioni più importanti

3

PASSBUILDER

Una app di tipo Mobile si può comporre dei seguenti elementi:

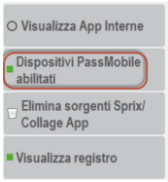
- **Sprix mobile (obbligatorio)**
- archivi mobile
- libreria sprix
- collage server remoto
- cartella dati mobile
- estensione collage mobile
- estensione sprix mobile

App di esempio disponibile sulla seguente area download:
https://download.passepartout.cloud/CondivisaPartner/PassMobile/

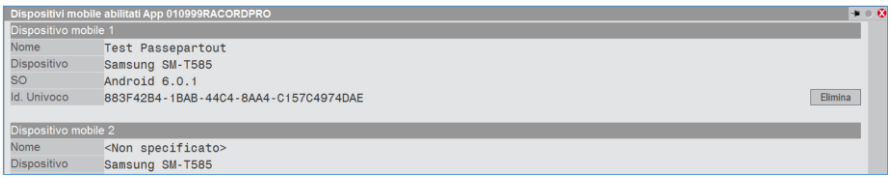
4

ATTIVAZIONE DELLE APP PRESSO I CLIENTI

Tutte le App di tipo mobile prevedono la gestione di dispositivi tanti quanti sono gli iDesk mobile attivati da YouPass: all'interno del gestionale è possibile gestire i dispositivi mobile che possono collegarsi utilizzando il pulsante "Dispositivi PassMobile abilitati".



Verrà mostrata una lista popolata con politiche FIFO, in cui saranno presenti tutti e soli di dispositivi mobile che potranno utilizzare l'App.



5

PASSEPARTOUT Software e servizi gestionali

PASSMOBILE

Esecutore delle App di tipo mobile da installare sul dispositivo mobile quale smartphone o tablet sia su sistemi Android che IoS

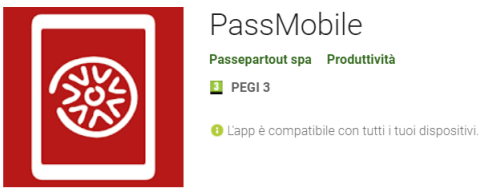
PassMobile gestisce:

- Immissione e revisione dei dati con interfaccia nativa
- Scrittura, modifica e cancellazione dei dati sulla propria base dati
- Sincronizzazione della propria base dati con quella di Mexal e PassCom

Completamente realizzato con linguaggio di sviluppo per Android e iOS

Dispositivi supportati:

- Android versione 4.4.4 o superiori
- iOS 11 e superiori



6

PASSEPARTOUT Software e servizi gestionali

PASSMOBILE

Installare PassMobile sul dispositivo mobile:

- Accedere agli store Apple/Android
- Scaricare e installare Passmobile sul dispositivo mobile
- Configurare l'installazione da cui prelevare l'App Mobile

Accedere all'installazione Mexal o PassCom e creare l'App mobile nello stesso modo di una qualsiasi App con PassBuilder.

L'App mobile si distingue dalle altre perché presenta almeno un elemento di tipo Sprix mobile.

7

PASSMOBILE

- **Definizione database mobile**. Aggiungere un elemento archivio mobile per ogni tabella e archivio necessario alla nostra personalizzazione. PassBuilder fornisce un comodo wizard per questa gestione.

- **Menù funzioni utente**. Aggiungere un elemento sprix mobile per ogni menù tramite il quale l'utente accede alla funzionalità richiesta. PassBuilder utilizza l'ambiente di sviluppo di sprix con le classiche istruzioni più quelle specifiche per il mobile. La programmazione è a eventi come collage.

8

GESTIONE BASE DATI archivi SU MOBILE

Gestione Lettura tabelle

- La lettura di una tabella o di un singolo dato di un archivio utilizza il dizionario Sprix con la stessa nomenclatura

Gestione Lettura archivio

- La lettura di un singolo dato di un record di un archivio utilizza il dizionario Sprix con la stessa nomenclatura
- La lettura dei record degli archivi utilizzano le stesse istruzioni Sprix come GETPC, GETMM, GETAR, ecc, con le seguenti particolarità;
  - La GETAR preleva i dati di anagrafica e non quelli dei progressivi che non sono gestiti sul mobile
  - La GETMM per i movimenti di magazzino utilizza la sintassi **"**<sigla><serie>/<aa>-<numero>**"**, per ordini preventivi e matrici utilizza la classica sintassi **"**<sigla>[<serie>/]<numero>**"**

GESTIONE BASE DATI ARCHIVI SU MOBILE

Gestione scrittura archivio

- La scrittura degli archivi è possibile solo per i clienti fornitori e i documenti di magazzino. Le istruzioni sono le stesse di Sprix, ossia PUTPC e PUTMM. Queste funzioni non effettuano i controlli di validità dei dati che vanno realizzate nel codice sprix relativamente ai campi gestiti sul mobile.

Gestione cancellazione archivio

- La cancellazione degli archivi è possibile solo per i clienti fornitore e i documenti di magazzino. Le istruzioni sono le stesse di Sprix, ossia DELPC e DELMM, questa usa la sintassi della GETMM

GESTIONE SINCRONIZZAZIONE ARCHIVI

Sincronizzazione base dati mobile e base dati Mexal o PassCom

- La sincronizzazione degli archivi mobile si basa sulla data di ultima modifica di Mexal e non dipende da quella con cui sono memorizzati i record sul dispositivo mobile
- I controlli di validità sui dati vengono effettuati in fase di sincronizzazione archivi e gli eventuali errori sono visualizzati sul mobile

11

PROGRAMMAZIONE APP MOBILE – SPRIX MOBILE

Ogni Sprix mobile deve avere obbligatoriamente l'etichetta che individua l'inizio della funzione

- **ON_START_SPRIX**

Il form rappresenta la videata che è composta da input, pulsanti e liste le istruzioni e le variabili per la gestione sono:

- **_WF –** Variabili di definizione del form. Categoria variabili 38
- **WCREATEFORM –** Istruzione per la creazione del form in base all'impostazione delle variabili della famiglia _WF
- **WSHOWFORM –** Visualizzazione del form e di tutti gli elementi che sono stati creati e che lo costituiscono tra i quali:
  - **Campo di Input**
  - **Pulsante**
  - **Lista di dati**

12

PROGRAMMAZIONE APP MOBILE – SPRIX MOBILE

Istruzioni e variabili per creare gli elementi di un form

- Campo di input - Pulsante
  - **_WI –** Variabili di definizione di un input o bottone. Categoria variabili 41
  - **WCREATEINPUT –** Istruzione per la creazione di un input o di un bottone in base all'impostazione delle variabili della famiglia _WI
- Lista dati
  - **_WL –** Variabili di definizione di una lista. Categoria variabili 42
  - **WCREATELIST –** Istruzione per la creazione di una lista in base all'impostazione delle variabili della famiglia _WL

La lista può essere contenuta in un form oppure associata ad un campo di input.

13

PASSEPARTOUT

PROGRAMMAZIONE APP MOBILE – SPRIX MOBILE

Oggetto	Evento	Descrizione
Form	ON_PRECLOSE_<codiceoggetto> ON_CLOSE_<codiceoggetto>	Eventi chiamati alla chiusura del form
Campo di Input	ON_IN_<codiceoggetto> ON_OUT_<codiceoggetto> ON_ARCODSTRU_<codiceggetto> ON_ARQTA_<codiceoggetto> ON_ARQTAOK_<codiceoggetto>	Evento chiamato in ingresso dell'input Evento chiamato in uscita dall'input Evento chiamato all'uscita della maschera per inserimento articolo strutturato Evento chiamato in caso di input a taglie Evento chiamato alla conferma di input a taglie
Pulsante	ON_PRESS_<codiceoggetto>	Evento chiamato premendo il pulsante

14

PASSEPARTOUT

PROGRAMMAZIONE APP MOBILE – SPRIX MOBILE

Oggetto	Evento	Descrizione
Lista collegata ad archivio	ON_PRESS_<codiceoggetto> ON_ROWFILTER_<codiceoggetto> ON_ROW_<codiceoggetto>	Evento chiamato selezionando l'elemento Evento chiamato in fase di riempimento della lista per filtrare, abbinato alla variabile _WLROWOK$ _WLROWSTOP$ Evento chiamato per riempire la lista
Lista dinamica	ON_ROW_<codiceoggetto> ON_PRESS_<codiceoggetto>	Evento chiamato per riempire la lista Evento chiamato selezionando l'elemento
Estensione sprix mobile	ON_SHOW_<codiceoggetto>	Evento chiamato all'apertura del form identificato dal codice oggetto
Carrello	ON_CART_<codiceoggetto> ON_CARTQTA_<codiceoggetto> ON_CARTOK_<codiceoggetto>	Evento chiamato alla visualizzazione del carrello Evento chiamato per ogni elemento del carrello Evento chiamato alla conferma del carrello
Iteratore	ON_ITER_<codiceoggetto> ON_ITERFILTER_<codiceoggetto> ON_ITEREND_<codiceoggetto>	Evento chiamato per ogni record dell'archivio Evento chiamato per filtrare i record Evento chiamato alla fine dell'iterazione

15

PASSEPARTOUT

PROGRAMMAZIONE APP MOBILE – SPRIX MOBILE

Ogni istruzione di creazione ha come parametro il <codiceoggetto> che il programmatore deve assegnare ed è quello da usare per le etichette degli eventi dell'oggetto. La funzione ritorna un ID univoco utilizzato per referenziare l'oggetto in altri punti del codice.

Gli id degli oggetti creati vengono restituiti nelle seguenti variabili di struttura:

- ✓ _WFOID per il form
- ✓ _WIOID per gli input
- ✓ _WLOID per le liste
- ✓ _WDOID per le finestre di dialogo

16

PASSEPARTOUT

DEFINIZIONE DI UN FORM

Il form è l'elemento contenitore di tutti gli oggetti grafici che compongono la videata da mostrare sul dispositivo mobile.

Le variabili da impostare sono:

- _WFTITLE$: definisce il titolo del form
- _WFHDRTXT$: definisce un testo da mostrare nell'header del form
- _WFFTRTXT$: definisce un testo da mostrare sul footer del form

L'oggetto viene creato con l'istruzione **WCREATEFORM <"codice_oggetto">** e sarà mandato a video solo mediante l'istruzione **WSHOWFORM <id_oggetto>**

Esempio

```
_WFTITLE$ = "Titolo del form"
_WFHDRTXT$ = "Testo HEADER"
_WFFTRTXT$ = "Testo FOOTER"
WCREATEFORM "FCLIFOR"          - FCLIFOR è il codice oggetto
ID_FCLIFOR = _WFOID            - ID_FCLIFOR è l'id univoco restituito dopo la creazione
 < codice in cui definire i vari oggetti grafici>
WSHOWFORM ID_FCLIFOR
```

17

PASSEPARTOUT Software e servizi gestionali

DEFINIZIONE DI UN FORM

Esempio di creazione di un form vuoto

18

PASSEPARTOUT Software e servizi gestionali

DEFINIZIONE CAMPO DI INPUT

Tipologia

- Campo di testo _WITYPE$ = "TEXT"
- Pulsante _WITYPE$ = "BUTTON" (pulsante)
- Campo web _WITYPE$ = "WEB" (icona web)
- Campo telefono _WITYPE$ = "PHONE" (icona telefono)
- Campo email _WITYPE$ = "MAIL" (icona e-mail)
- Campo data _WITYPE$ = "DATE" (icona date-time picker)
- Campo Art. Strutt. _WITYPE$ = "ARCOD" (articoli strutturati)
- Campo Numerico _WITYPE$ = "NUMBER" (icona input numerico)
- Campo a taglie _WITYPE$ = "ARQTA" (icona input gestione taglie)
- Separatore _WITYPE$ = "DIVIDER" (divisore grafico)
- BarCode _WITYPE$ = "BARCODE" (icona lettore codici a barre)
- Campo ora _WITYPE$ = "TIME" (icona time picker)
- TexArea _WITYPE$ = "TEXTAREA" (text area di massimo 4 righe)

19

PASSEPARTOUT

DEFINIZIONE CAMPO INPUT

Layout:

- Numero caratteri _WILENCAR = <numcaratteri>
- Allineamento in linea _WILINE$ = "INLINE"

NB: I campi verranno affiancati solo se c'è sufficiente spazio sul display del dispositivo

Form a cui agganciarlo:

- _WIPARENTID = ID del form a cui agganciarlo, creato precedentemente

Lista collegata al campo di input (combo box)

- _WILISTID = ID della lista creata precedentemente
- _WILISTNFLD = Numero della colonna della lista da cui prelevare il valore delle riga selezionata dall'utente
- Nel caso il campo sia di sola lettura l'input diventa un combo _WIOUTONLY$ = "S/T"

Eventi associati al campo

- ON_IN_<codiceoggetto> Evento che si scatena all'entrata
- ON_OUT_<codiceoggetto> Evento che si scatena all'uscita

20

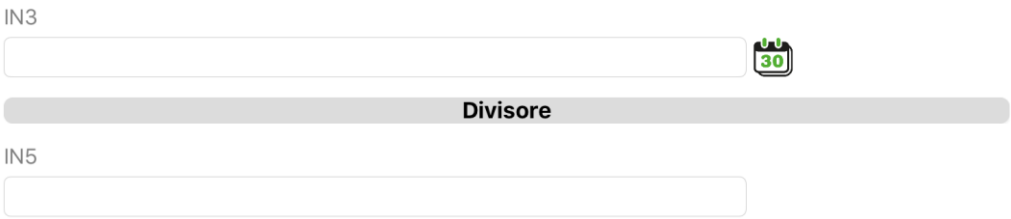
PASSEPARTOUT

DEFINIZIONE CAMPO DI INPUT separatore

Creando l'input con questi parametri

- _WTYPE = “DIVIDER”
- _WIVALUE$ = “<testo_separatore>”

verrà creato un elemento separatore con un testo personalizzato.



21

PASSEPARTOUT

DEFINIZIONE CAMPO DI INPUT

Esempio di inserimenti campi di input all’interno di un form

22

PASSEPARTOUT

DEFINIZIONE DI UN PULSANTE

- Pulsante gestito in fondo al form nella barra dei pulsanti
  - _WITYPE$ = "BUTTON"
  - _WIPARENTZN$ = ""
- Pulsante azione gestito nella barra del titolo
  - _WITYPE$ = "BUTTON"
  - _WIPARENTZN$ = "ACTION "

_WIICO$ = "<codice-icona>" o percorso immagine personalizzata importata con l'elemento Cartella dati mobile
_WIOUTONLY$ = "S/N": se impostata ad "S" rende il pulsante inattivo e non cliccabile. Visivamente il pulsante cambierà colorazione (più chiara) per evidenziare lo stato di disabilitazione.

E' possibile impostare delle icone invece del testo valorizzando la variabile _WIICO$ con il codice corrispondente, oppure utilizzare il percorso della "**Cartella dati mobile**" nel formato "<nome_cartella$>\<nomefile>" (vedi elemento Cartella Dati Mobile di Passbuilder)

Con un long press sul pulsante di tipo icona verrà mostrato un *tooltip* con il valore impostato in _WIVALUE$

23

PASSEPARTOUT

DEFINIZIONE DI UN PULSANTE

Codici icone Passepartout:

Codici	1	2	3	4	5	6	7	8	9	10	11	12	13
Icone													
Significato	Ok	Annulla	Modifica	Salva	Aggiungi	Elimina	Stampa	Copia	Cerca doc.	Cerca contatto	Cerca articolo	Carrello	Immagine

Form a cui agganciare il pulsante
- _WIPARENTID = ID del form a cui agganciarlo, creato precedentemente
- Può essere specificato come *parentid* l'ID di un altro pulsante: in questo caso il risultato è la realizzazione di un menu contestuale

Evento di pressione del pulsante
- ON_PRESS_<codiceoggetto>

24

PASSEPARTOUT

DEFINIZIONE DI UN PULSANTE

Esempio di inserimenti di un pulsante all'interno di un form

DEFINIZIONE GENERALE DELLE LISTE

Liste Dinamiche – Liste collegate ad archivio – Estensione di lista collegata ad archivio

Dimensione lista

- _WLTOTROW = nr — Numero Righe
- _WLTOTFLD = nc — Numero Colonne

Definizione colonne

- _WLDESFLD$(nc) — Descrizione delle colonne
- _WLLENCARFLD(nc) = <numcaratterricampo> — Dimensione campo espressa in caratteri
- _WLLINEFLD$(nc) = "INLINE" — Indica se il campo deve essere affiancato
- _WLROWICO$= "S" o "ico_archivio" — Indica se usare una icona (S per liste din.)
- _WLVALICO$ — Icona da mostrare (liste dinamiche)
- _WLFMTFLD$()= ".,n<>stringa" — Numero con il separatore delle migliaia, numero di decimali e prefisso</suffisso>

es. ".,2>€" produce un numero formattato in questo modo 1.250,33€

DEFINIZIONE GENERALI DELLE LISTE

Lista agganciata al Form

- _WLPARENTID = ID del form a cui agganciarlo, creato precedentemente
- _WLPARENTZN$ = ""

Lista agganciata al campo di input

- _WLPARENTID = ID del form relativo al campo di input, creato precedentemente
- _WLPARENTZN$ = "INPUT"
- Nel relativo input occorre impostare le variabili _WILISTID e _WILISTNFLD

Evento di scelta di una riga

- ON_PRESS_<codiceoggetto>

Le immagini inserite in lista potranno essere ingrandite facendo un "tap" sull'immagine senza scatenare eventi personalizzati

27

PASSEPARTOUT

GESTIONE DEL LAYOUT NELLE LISTE

Gestione delle liste in modalità default, elenco, griglia

- La proprietà viene impostata direttamente sul dispositivo mobile a discrezione dell'utente
- Nell'istruzione WCREATELIST sono disponibili 2 variabili di struttura per impostare la dimensione del campo e quali colonne vanno affiancate:

  _WLLENCARFLD(1) = <numcaratterecampo>

  _WLLINEFLD$(1) = "INLINE"

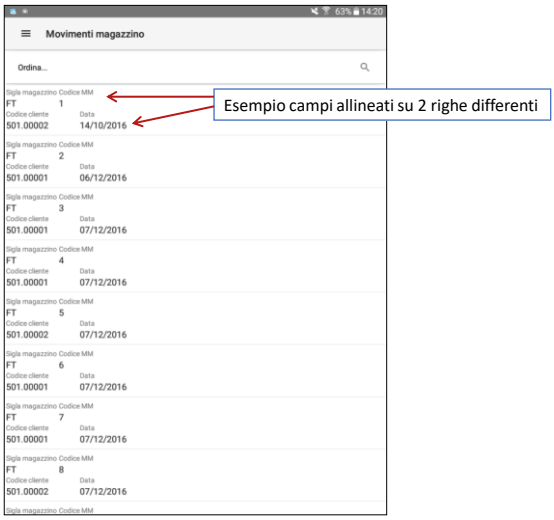
N.B. Le colonne verranno affiancate solo se c'è sufficiente spazio sul display del dispositivo

28

PASSEPARTOUT

GESTIONE DEL LAYOUT

Scelta layout da utente



29

PASSEPARTOUT

DEFINIZIONE DELLA LISTA DINAMICA

Impostazione variabili _WL

- Archivio di riferimento vuoto _WLARCNAME$ = ""

Impostazione del valore delle righe della lista dinamica

- L'esecutore, prima della visualizzazione della lista, richiama l'etichetta ON_ROW_<codiceoggetto> per impostare il contenuto dell'i-esima riga
- L'esecutore valorizza ad ogni chiamata dell'etichetta la variabile _WLNUMROW che identifica la riga da processare
- Il programmatore valorizzando le variabili _WLVALFLD$(nc) una per ogni colonna fornisce il valore della colonna per la riga iesima.

Aggiornamento della visualizzazione della lista

- Istruzione WCALL "REFRESHLIST", <ID_oggetto>

30

PASSEPARTOUT

DEFINIZIONE DELLA LISTA DINAMICA

Ordinamento e filtri

- _WLTPFLD$(n) permette di definire il tipo di dato del campo n-esimo:
  - Numerico = "NUMERO,n" dove n è il numero di decimali dopo la virgola
  - Data= "DATA"
  - Testo="STRINGA"
- _WLUSERORD$ e _WLUSERFLT$ se impostate a "S" abiliteranno rispettivamente l'ordinamento ed il filtro sulla lista dinamica

31

DEFINIZIONE DELLA LISTA DINAMICA

Esempio di creazione di una lista dinamica

32

DEFINIZIONE DELLA LISTA ARCHIVIO

Impostazione variabili _WL

- Archivio di riferimento _WLARCNAME$ = "codicearchivio" archivio mobile
- Campo dell'archivio da associare alla colonna _WLARCFLD$(nc) = "codicecampoarchivio" dell'archivio mobile
- Ordinamento della lista
  - _WLORDFLD(1) = numero di colonna per cui ordinare la lista
  - _WLORDDR$(1) = tipo di ordinamento "A" ascendente "D" discendente

Impostazione del valore delle righe della lista

- L'esecutore legge in automatico l'archivio e imposta le righe con i dati delle colonne richieste.
- L'esecutore fornisce anche un filtro per limitare le righe da visualizzare
  - Ad ogni riga invoca l'etichetta ON_ROWFILTER_ "codiceoggetto"
  - Il programmatore imposta _WLROWOK$ = "S" riga da visualizzare o "N" riga da scartare
  - Con _WLROWSTOP$ = "S" si interrompe il ciclo di lettura

33

PASSEPARTOUT

DEFINIZIONE DELLA LISTA ARCHIVIO

Esempio di lista collegata ad archivio

34

PASSEPARTOUT

PERSONALIZZAZIONE LISTA ARCHIVIO

Definizione delle colonne personalizzate

- Numero di colonne personalizzate _WLTOTFLDEXT = ncp
- Descrizione _WLNMFLDEXT$(ncp)
- Tipologia _WLTPFLDEXT$(ncp) "STRINGA" alfanumerico, "NUMERO,ndec" numerico con eventuali decimali

Utilizzo delle colonne personalizzate nelle colonne della lista

- Impostare le variabili _WLARCFLD$(nc) = "_WLNMFLDEXT$(ncp)"

PERSONALIZZAZIONE LISTA ARCHIVIO

Impostazione delle colonne personalizzate nelle righe della lista

- L'esecutore, prima della visualizzazione della lista, richiama l'etichetta ON_ROW_<codiceoggetto>
- Istruzione di lettura di un valore del record caricato dalla ON_ROW_<codiceoggetto>.
  - GETROWVALFLD "codicecampo", <variabilesprix>
- Valorizzazione delle variabili _WLVLFLDEXT$(nc) relative alle colonne personalizzate

Impostazione dell'ordinamento personalizzato della lettura archivio

- L'ordinamento delle righe presentate all'evento ON_ROW_<codiceoggetto> avviene tramite le variabili _WLROWORDNM$(1) = "codicecampoarchivio" e _WLROWORDDR$(1) = "A" ascendente o "D" discendente

Interruzione del ciclo di lettura dell'archivio delle righe

- Negli eventi ON_ROWFILTER_<codiceoggetto> e ON_ROW_<codiceoggetto> impostando la variabile _WLROWSTOP$ = "S" si interrompe il ciclo di lettura

ESEMPI LISTE

- Esempio di estensione di lista collegata ad archivio
  - Esempio di lista visualizzata come Combo Box

37

GESTIONE CHIAMATE REMOTE

Le chiamate remote permettono di interagire in tempo reale con il gestionale interrogando opportuni collage remoti mediante l'utilizzo della seguente istruzione sprix-mobile :

SRVCALL <nome_coll_remoto$>,<etichetta_server$>,<etichetta_client$>

- <nome_coll_remoto$>: è il nome logico del file contenente il codice del collage server remoto. Il collage server remoto è un elemento che deve essere inserito nella app scegliendo "Collage Server Remoto"
- <etichetta_server$>: è l'etichetta dichiarata dentro il collage server remoto e che sarà usata da questa chiamata remota
- <etichetta_client$>: è l'etichetta di call back che deve essere definita all'interno dello Sprix-mobile necessaria per processare i dati ritornati dalla chiamata remota

La funzione valorizza le variabili di errore _ERRSRV e _ERRSRV$ che devono essere testate dentro l'etichetta di callback

38

GESTIONE CHIAMATE REMOTE

Sia nelle etichette lato client che lato server dovranno essere utilizzate opportune istruzioni per la ricezione o invio dei dati.

Per ricevere dati:

- **GETREM_NUM** <tag$>,<numero>
- **GETREM_STR** <tag$>,<stringa$>
- **GETREM_ARRAY** <tag$>,<array_di_stringhe$ / array_di_numeri>
- **GETREM_FILE** <tag$>,<percorso_file$>

Per inviare dati:

- **PUTREM_NUM** <tag$>,<numero>
- **PUTREM_STR** <tag$>,<stringa$>
- **PUTREM_ARRAY** <tag$>,<array_di_stringhe$ / array_di_numeri>
- **PUREM_FILE** <tag$>,<percorso_file$>

Dove <tag$> è l'identificativo dell'elemento da inviare/ricevere.

39

PASSEPARTOUT

GESTIONE CHIAMATE REMOTE

Lato mobile

All'interno di qualsiasi Sprix Mobile

```
...
PUTREM_STR "test_str","Hello world!"
SRVCALL "col_rem","TEST_S","TEST_C"
...
TEST_C:
  GETREM_NUM "test_num",NUM
  <fa qualcosa con il risultato>
END
```

Lato gestionale

All'interno del collage server remoto chiamato <col_rem>

```
TEST_S:
 GETREM_STR "test_str",TEXT$
 IFF TEXT$ = "Hello world!"
   N = 0
 ELSEF
   N = 1
 ENDF
 PUTREM_NUM "test_num",N
END
```

40

PASSEPARTOUT

GESTIONE CHIAMATE REMOTE

Esempio di utilizzo del Collage Server Remoto

41

ISTRUZIONI WSET, WGET, WGETOID

Istruzioni e variabili per leggere e impostare i valori delle variabili di un oggetto runtime

- **_WOP –** Variabili per leggere e scrivere la definizione degli oggetti. Categoria variabili 43
- **WGET <ID_oggetto>** Istruzione per leggere
- **WSET <ID_oggetto>** Istruzione per impostare
- **WGETOID "codiceoggetto"** Istruzione che restituisce l'ID univoco dell'oggetto

In **_WOPNAME$** va impostata la proprietà su cui si vuole agire (ad esempio il valore di una casella di testo _WIVALUE$ oppure il titolo di un form _WFTITLE$)

In **_WOPVAL** (numerico) o **_WOPVAL$** (stringa) va impostato o letto il valore relativo alla proprietà indicata in _WOPNAME$

Il codice ed il valore di eventuali errori sono presenti nelle variabili **_ERRWOP** e **_ERRWOP$**

42

ISTRUZIONI WSET, WGET, WGETOID

Esempio: leggere un valore da una casella di testo

_WOPNAME$="_WIVALUE$"

WGET <id_casella_di_testo>

MIA_VAR$ = _WOPVAL$

Esempio: scrivere un valore in un casella esistente

_WOPNAME$="_WIVALUE$"

_WOPVAL$="<mio_valore>"

WSET <id_casella_di_testo>

GESTIONE FINESTRE DI DIALOGO

Gestione delle dialog

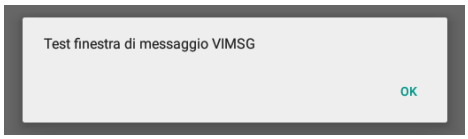
Oltre alle classiche VIMSG (che non bloccano il flusso del programma), sono presenti istruzioni per gestire la visualizzazione di messaggi con eventuali pulsanti di conferma oltre al solo pulsante di conferma. La dialog viene creata con lo stesso criterio con cui vengono gestiti i FORM.

```
_WDTITLE$="Titolo"
_WDMESSAGE$="Messaggio"
 WCREATEDIALOG "NOME_DIALOG"
    … definizione pulsanti..
 WSHOWDIALOG
```

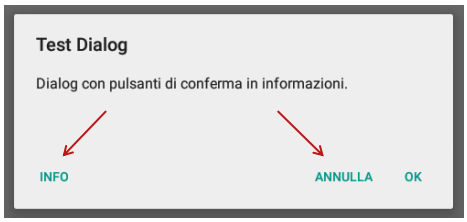
può prevedere fino a 3 tasti intercettabili con l'evento ON_PRESS_nomeTasto.

GESTIONE FINESTRE DI DIALOGO

Esempio di messaggio con l'istruzione VIMSG



Esempio di messaggio con le finestre di dialogo



PASSEPARTOUT
Software e servizi gestionali

PASSEPARTOUT®
Software e servizi gestionali

Corso Sprix Mobile Base

Gianluca Suzzi