2025/12/05 05:17 1/2 Superclasse

## **Superclasse**

• cwbBpaWizard: Finestra di gestione wizard da estendere

## Interfaccia

• wizardable: Interfaccia da implementare

Per creare un wizard va creata una form 'container' con all'interno:

- \* la buttonBar (tasti gestiti: Avanti, Indietro, Annulla e Concludi).
- \* un div segnaposto chiamato 'divGestione' su cui verrannò inserite le form dei vari step.

Questa form container dovrà estendere la classe cwbBpaWizard mentre le form dei singoli step dovranno implementare l'interfaccia wizardable. Sulla form container nel metodo initVars vanno valorizzate le seguenti proprietà:

- \* nameForm: il nome della form container
- \* firstFormName: Il nome della form utilizzata come primo step
- \* lastFormName: Il nome della form utilizzata come ultimo step

Il primo e l'ultimo step sono quindi obbligatoriamente fissi, mentre tutte le form intermedie sono dinamiche.

Nella superclasse è presente una variabile 'listaGuida' che è una linkedlist che gestisce il flusso. La variabile listaGuida ha un metodo current (\$this→listaGuida→current()) che specifica la form corrente in cui ci troviamo.

Nel metodo preAvanti va gestito il comportamento del wizard al click del tasto avanti (va impostato qual'è lo step successivo e le azioni/parametri da passare) Per aggiungere uno step alla listaGuida va chiamato il metodo addFormToList passandogli come parametro il nome della form successiva (\$this→addFormToList('nomeDelloStepSuccessivo')).

es implementazione preAvanti:

```
protected function preAvanti() {
    switch ($this->listaGuida->current()) {
        case 'cwdForm1':
        $x = $_POST['cwdForm1_x'];
        switch ($x) {
        case "1":
```

```
$this->addFormToList('cwdForm2'); // se x=1 aggiungo
cwdForm2 come step successivo
                      // codice....
                      break;
                  case 2:
                      $this->addFormToList('cwdForm3'); // se x=2 aggiungo
cwdForm3 come step successivo
                      // codice....
                      break;
              }
              break;
          case 'cwdForm2':
              // se non aggiungo uno step successivo, vado allo step finale
              // codice....
              break;
      }
  }
```

Implementando l'interfaccia wizardable, verrà richiesto di implementare il metodo 'validaWizardStep(\$formData, &\$msg)' in cui va gestita la validazione della form al click del tasto 'avanti'. Il metodo deve tornare true o false in base al risultato e concatenare i messaggi di errore su \$msg.

From:

https://wiki.nuvolaitalsoft.it/ - wiki

Permanent link:

https://wiki.nuvolaitalsoft.it/doku.php?id=sviluppo:cityware wizard&rev=1467030597



