Introduzione

Il modulo serve per esporre dei web services utilizzando la modalità REST. E' presente nella cartella 'wsrest' all'interno di '/lib'.

Definizione routes

Le routes vanno censite su un apposito file di configurazione: '/config/routes.ini'. Per ogni route devono essere specificate le seguenti informazioni:

- **VERB**: indica il 'verb http' (GET o POST)
- **CONTROLLER**: è il nome della classe controller (presente in '/controller') esclusa l'estensione ('.php')
- **ACTION**: è il nome del metodo (public) del controller da invocare
- **OUTPUT**: indica il formato dell'output (xml/json)
- **CHECKAUTH**: indica se la chiamata deve essere autenticata (se 1 = controlla autenticazione, altrimenti 0)

Esempio

```
[dummyActionGET]
VERB = GET
CONTROLLER = dummyController
ACTION = dummyActionGET
OUTPUT = json
CHECKAUTH = 1
```

Controller

Il controller ha il compito di gestire la chiamata. Per ogni controller possono essere presenti più 'actions'. Ogni 'action' corrisponde ad un metodo 'public', che riceve in ingresso un array di parametri. Ogni controller estende la classe padre 'RestController'. Ogni 'action' deve restituire un risultato (sarà poi compito del server REST formattare adeguatamente l'output).

Esempio di una action

```
public function dummyActionGET($params) {
    $toReturn = array(
        'chiave1' => 'valore1',
        'chiave2' => 'valore2',
```

```
        'params' => array()
    );
    if ($params != null) {
        $toReturn['params'] = $params;
    }
    return $toReturn;
}
```

Autenticazione

Nel file 'routes.ini' sono presenti i seguenti metodi per la gestione dell'autenticazione:

- **GetItaEngineContextToken**: date in input le credenziali per il login, restituisce un token di autenticazione che dovrà essere utilizzato per tutte le chiamate successive autenticate.
- **CheckItaEngineContextToken**: controlla la validità del token in input.
- **DestroyItaEngineContextToken**: cancella il token indicato in input dalla sessione.

Esempio di chiamata per ottenere un token di autenticazione

http://localhost:4000/itaTest/wsrest/service.php/login/GetItaEngineContextToken?UserName=italsoft&Password=DemoPal1&DomainCode=X999

Chiamate autenticate

Per le chiamate autenticate deve essere passato nell'header HTTP della request il parametro **X-ITA-TOKEN**, che deve contenere il token ottenuto dalla chiamata '**GetItaEngineContextToken**' (il token deve essere valido e non scaduto).

Multipart Request

Per le richieste multipart, l'unica differenza rispetto alle altre chiamate è che nell'array dei parametri in input della action viene passata la chiave 'FILES' di tipo array, che contiene l'elenco dei file (corrisponde alla variabile superglobal '$_FILES')

REST Model

Per le classiche operazioni CRUD su un model attraverso l'utilizzo di servizi REST (ad esempio, se si vuole utilizzare itaEngine come back-end completo di una app implementata con un'altra tecnologia -

come AngularJS o Ionic), occorre definire delle classi model specifiche che estendono '**wsModelRest**'. Le operazioni messe a disposizione sono le seguenti:

- query
- readRecord
- insertRecord
- updateRecord
- deleteRecord

Per un funzionamento di default, le varie sottoclassi dovranno solamente specializzare i metodi:

- initVars
- getAuditInfo

Esempio

```
class cwbBtaGrunazRest extends wsModelRest{
    public function initVars() {
       $this->setDbName('CITYWARE');
       $this->setDbSuffix('');
       $this->setTableName('BTA_GRUNAZ');
       $this->setPk('CODGRNAZ');
       $this->setKeyAutoincrement(false);
       $this->setUseInsertAudit(false);
       $this->setUseUpdateAudit(true);
    }

    protected function getAuditInfo($data) {
        return 'Oggetto: ' . $data['CODGRNAZ'] . " " . $data['DESGRNAZ'];
    }

}
```

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

Permanent link:
https://wiki.nuvolaitalsoft.it/doku.php?id=sviluppo:rest_server&rev=1456737775

Last update: **2018/03/19 10:45**