

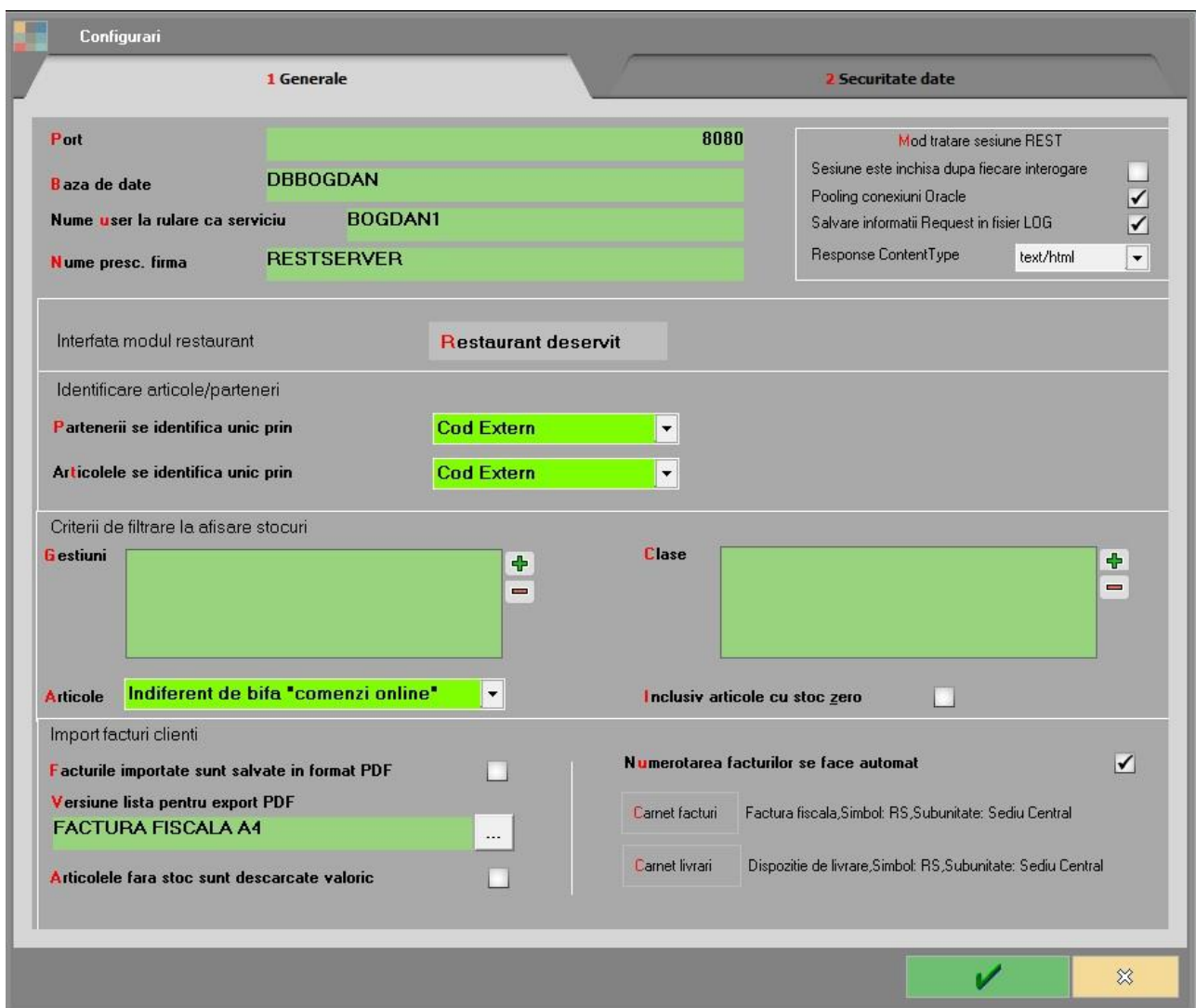
WMERestServer – FUNCTII RESTAURANT

ATENTIE!

WMERestServer se poate porni fie ca serviciu (WMERestService.exe) fie ca executabil (WMERestServer.exe). Pentru schimbarea versiunii executabilului WMERestService.exe, trebuie inlocuit executabilul, dezinstalat serviciul din Task Manager si apoi reinstalat. Pasii detalati sunt disponibili in documentatia Upgrade WMERestService.doc.

Setari generale

Parametrii de functionare se seteaza in Configurari > Setari > Generale.



The screenshot shows the 'Configurari' window with the '1 Generale' tab selected. The configuration is as follows:

- Port:** 8080
- Baza de date:** DBBOGDAN
- Nume user la rulare ca serviciu:** BOGDAN1
- Nume presc. firma:** RESTSERVER
- Mod tratare sesiune REST:**
 - Sesiune este inchisa dupa fiecare interogare:
 - Pooling conexiuni Oracle:
 - Salvare informatii Request in fisier LOG:
 - Response ContentType: text/html
- Interfata modul restaurant:** Restaurant deservit
- Identificare articole/parteneri:**
 - Partenerii se identifica unic prin: Cod Extern
 - Articolele se identifica unic prin: Cod Extern
- Criterii de filtrare la afisare stocuri:**
 - Gestioni: (empty list)
 - Clase: (empty list)
- Articole:** Indiferent de bifa "comenzi online"
- Inclusiv articole cu stoc zero:**
- Import facturi clienti:**
 - Facturile importate sunt salvate in format PDF:
 - Versiune lista pentru export PDF: FACTURA FISCALA A4
 - Articolele fara stoc sunt descarcate valoric:
- Numerotarea facturilor se face automat:**
 - Carnet facturi: Factura fiscala, Simbol: RS, Subunitate: Sediul Central
 - Carnet livrari: Dispozitie de livrare, Simbol: RS, Subunitate: Sediul Central

Pentru functionarea in modul Restaurant este necesara setarea restaurantului pentru care vor fi furnizate informatiile.

Nume user la rulare ca serviciu	BOGDAN1	Salvare informatii Request in fisier LOG	<input checked="" type="checkbox"/>
Nume presc. firma	REST1	Response ContentType	application/json
Interfata modul restaurant	Restaurant deservit		
Identificare articole/parteneri			
Partenerii se identifica unic prin	Cod Extern		
Articolele se identifica unic prin	Cod Extern		

Funcții disponibile:

1. Citirea nomenclatorului de sali - GetListaSali
2. Citirea meselor dintr-o sala - GetListaMese
3. Citirea nomenclatorului de ospatari - GetListaOspatari
4. Citirea ospatarului dupa parola - GetOspatarByParola
5. Citirea nomenclatorului de gestiuni - GetListaGestiuni
6. Citirea articolelor din meniul restaurantului - GetListaArticole
7. Blocarea mesei – BlocheazaMasa
8. Deblocarea mesei – DeblocheazaMasa
9. Postarea comenzii restaurant - UpdateComandaMasa
10. Postarea starii mesei - UpdateStareMasa
11. Citirea comenzilor deschise la restaurant – GetComenziDeschise
12. Citirea liniilor unei comenzi restaurant - GetLiniiCmdRest

Descriere FUNCTII

Citirea nomenclatorului de Sali - GetListaSali

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/GetListaSali**

Function **GetListaSali**: TJsonArray;

Exemplu returnat:

```
{
  "Error": "ok"
}, {
  "Data": [{
    "CodSala": "103",
    "Denumire": "Bar"
  }, {
    "CodSala": "105",
    "Denumire": "Bistroteca"
  }, {
    "CodSala": "104",
    "Denumire": "DISCO"
  }, {
    "CodSala": "102",
    "Denumire": "TERASA"
  }
]}
```

Mergi la [Top document](#)

Citirea meselor dintr-o sala - GetListaMese

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/GetListaMese/CodSala**

Function **GetListaMese(CodSala: Integer): TJsonArray;**

Exemplu returnat:

```
{
  "Error": "ok"
}, {
  "Data": [{
    "CodMasa": "127",
    "Denumire": "Masa 1",
    "Stare": "2"
  }, {
    "CodMasa": "128",
    "Denumire": "Masa 2",
    "Stare": "0"
  }, {
    "CodMasa": "129",
    "Denumire": "Masa 3",
    "Stare": "0"
  }, {
    "CodMasa": "130",
    "Denumire": "Masa 4",
    "Stare": "0"
  }, {
    "CodMasa": "131",
    "Denumire": "C.Flori",
    "Stare": "2"
  }, {
    "CodMasa": "132",
    "Denumire": "Masa 6",
    "Stare": "0"
  }
]}
}
```

Mergi la [Top document](#)

Citirea nomenclatorului de ospatari - GetListaOspatari

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/GetListaOspatari**

Function **GetListaOspatari**: TJsonArray;

Exemplu returnat:

```
[{
  "Cod": "117",
  "Nume": "CHISU",
  "Prenume": "ADRIAN"
}, {
  "Cod": "118",
  "Nume": "CINDEA",
  "Prenume": "FLORIN"
}, {
  "Cod": "102",
  "Nume": "PASCU",
  "Prenume": "COSMIN"
}, {
  "Cod": "103",
  "Nume": "CALBOREAN",
  "Prenume": "ALIN"
}, {
  "Cod": "120",
  "Nume": "COLDEA",
  "Prenume": "CLAUDIU REMUS"
}, {
  "Cod": "105",
  "Nume": "PARCALAB",
  "Prenume": "ARNOLD NORBERT"
}, {
  "Cod": "122",
  "Nume": "MATEIU",
  "Prenume": "FLORIN-OVIDIU"
}, {
  "Cod": "119",
  "Nume": "CRISTEA",
  "Prenume": "IOAN"
}, {
  "Cod": "123",
  "Nume": "ALEXANDRU",
  "Prenume": "ROBERT SEBASTIAN"
}]
```

Mergi la [Top document](#)

Citirea ospatarului dupa parola - GetOspatarByParola

Mod invocare:

`http://host:port/datasnap/rest/TServerMethods/GetOspatarByParola/DataIn`

Function **GetOspatarByParola(DataIn: TJSonValue): TJSONArray;**

Exemplu cu structura in care este asteptata comanda (parametrul dataIn):

```
["Parola": "123"]
```

Exemplu returnat:

```
[{  
  "CodOspatar": "102",  
  "NumeOspatar": "Danielevici Veronica",  
  "isSupervizor": "1"  
}]
```

Mergi la [Top document](#)

Citirea nomenclatorului de gestiuni - GetListaGestiuni

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/GetListaGestiuni**

Function **GetListaGestiuni**: TJsonArray;

Exemplu returnat:

```
[{
  "Cod": "128",
  "Denumire": "DISCO BAUM",
  "Simbol": "DISCO"
}, {
  "Cod": "129",
  "Denumire": "TERASA BAUM",
  "Simbol": ""
}, {
  "Cod": "136",
  "Denumire": "BISTROTECA",
  "Simbol": "BISTRO"
}, {
  "Cod": "274",
  "Denumire": "BISTRO",
  "Simbol": "BISTROO"
}, {
  "Cod": "374",
  "Denumire": "PIZERIE",
  "Simbol": "PIZZA"
}, {
  "Cod": "454",
  "Denumire": "PRODUCTIE SERA",
  "Simbol": "SERA"
}]
```

Mergi la [Top document](#)

Citirea articolelor din meniul restaurantului - GetListaArticole

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/GetListaArticole**

Function **GetListaArticole**: TJsonArray;

Exemplu returnat:

```
{
  "Error": "ok"
}, {
  "Data": [{
    "Cod": "19602",
    "Denumire": "ALUNE MINIBAR",
    "UM": "Buc",
    "Clasa": "BAUTURI",
    "Pret": "0",
    "FaraListare": "0",
    "GestImplicita": "128",
    "CodUM": "121",
    "CodExtern": " "
  }, {
    "Cod": "19604",
    "Denumire": "APA MINERALA 330/ML MINIBAR",
    "UM": "Sticle",
    "Clasa": "BAUTURI",
    "Pret": "0",
    "FaraListare": "0",
    "GestImplicita": "128",
    "CodUM": "126",
    "CodExtern": " "
  } ...
]
}
```

Pretul este adus de la constanta de tip user "Categorie implicita pret vanzare la case de marcat/POS-uri". Daca acesta nu este setata cu o anumita categorie de pret, atunci se ia in considerare constanta cu aceeasi denumire dar de tip general.

Mergi la [Top document](#)

Blocarea mesei - BlocheazaMasa

Mod invocare: **`http://host:port/datasnap/rest/TServerMethods/BlocheazaMasa/DataIn`**

Function **BlocheazaMasa(DataIn: TJsonValue): TJSONArray;**

Exemplu cu structura in care este asteptata comanda (parametrul dataIn):

```
[{"CodMasa": "150"},  
 {"CodOspatar": "2"}  
]
```

Daca rezultatul este [{"Error": "Masa este acum blocata de un alt utilizator"}], atunci masa este blocata de alt utilizator si trebuie asteptata eliberarea ei.

Doar cand rezultatul este [{"Error": "ok"}] se poate trimite o comanda pe respectiva masa.

Mergi la [Top document](#)

Deblocarea mesei - DeblocheazaMasa

Mod invocare:

http://host:port/datasnap/rest/TServerMethods/DeblocheazaMasa/DataIn

Function **DeblocheazaMasa(DataIn: TJSonValue): TJSONArray;**

Exemplu cu structura in care este asteptata comanda (parametrul dataIn):

```
[{"CodMasa": "150"}]
```

Mergi la [Top document](#)

Postarea comenzii restaurant - UpdateComandaMasa

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/ /DataIn**

Function **UpdateComandaMasa(DataIn: TJsonValue): TJSONArray;**

Exemplu cu structura in care este asteptata comanda (parametrul dataIn):

```
{
  "CodComanda": "0",           //0=comanda noua; AAA= cod comanda pe care o
                               //va modifica, completa; identificarea codului AAA se
                               //face folosind functia GetComenziDeschiseMasa
                               //descrisa mai jos
  "CodMasa": "150",
  "Obs": "Observatii comanda",
  "CodOspatar": "350",
  "AdresaLivrare": "Sos.Nationala",
  "ListareAutomata": "N"
}, {
  "Linii": [{
    "Cod": "106",
    "Pret": "5.5",
    "FaraListare": "0",       //0=cu listare; 1=fara listare
    "CodGest": "128",
    "CodUM": "121",
    "Cant": "1",
    "ListareGest": "0",      //0=nelistat la gestiune; 1=listat la gestiune
    "Obs": "aaa"
  }, {
    "Cod": "111",
    "Pret": "6",
    "FaraListare": "0",
    "CodGest": "128",
    "CodUM": "126",
    "Cant": "2",
    "ListareGest": "0",
    "Obs": "bbb"
  }, {
    "Cod": "112",
    "Pret": "6",
    "FaraListare": "0",
    "CodGest": "128",
    "CodUM": "126",
    "Cant": "3",
    "ListareGest": "0",
    "Obs": "ccc"
  }
  ]
}
```

Este deci un array din doua obiecte: primul cu informatii generale comanda (CodComanda = 0 inseamna comanda noua), al doilea obiect (numit "Linii" contine un array cu liniile comenzii.

Informatiile FaraListare, CodGest, CodUM vin catre tablet/aparat din citirea nomenclatorului de articole, trebuie gestionate acolo, si trebuie apoi sa se intoarca odata cu comanda. Este solutia cea mai simpla pentru a degreva serverul de operatii suplimentare de

interogari complicate asupra bazei de date pentru a regasi aceste informatii.

Mergi la [Top document](#)

Postarea starii mesei - UpdateStareMasa

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/UpdateStareMasa/DataIn**

Function **UpdateStareMasa(DataIn: TJsonValue): TJSONArray;**

Exemplu cu structura in care este asteptata comanda (parametrul dataIn):

```
[{"CodMasa": "150"},  
 {"Stare": "2"},  
 {"CodMasa": "151"},  
 {"Stare": "2"}  
]
```

Stari posibile|:

0= Libera,

1= Rezervata,

2= ComandaInCurs

3= ComandaInchisa

Mergi la [Top document](#)

Citirea comenzilor deschise la restaurant - GetComenziDeschiseMasa

Mod invocare:

http://host:port/datasnap/rest/TServerMethods/GetComenziDeschiseMasa/DataIn

Funcția **GetComenziDeschiseMasa(CodMasa: Integer; Ospatar: Integer): TJsonArray;**

Exemplu returnat:

```
[{"Error":"ok"},
 {"Data":[{"CodRestCmd":"53612",
          "NrComanda":"74"}
 ]
 ]
```

Funcția returnează comenzile deschise la o masă (trimisă ca parametru) de către un ospatar (parametru).

ATENȚIE: trebuie să gestionați acel CodRestCmd atunci când intrați pe o comandă deja salvată și să-l returnați în momentul când lansați din nou updateComandaMasa. A se vedea câmpul CodComanda din exemplul pentru updateComandaMasa (CodComanda "0" înseamnă doar comandă nouă).

Mergi la [Top document](#)

Citirea liniilor unei comenzi restaurant - GetLiniiCmdRest

Mod invocare: **http://host:port/datasnap/rest/TServerMethods/ GetLiniiCmdRest/DataIn**

Funcția **GetLiniiCmdRest(Codcomanda: Integer): TJsonArray;**

Exemplu returnat:

```
{{"Error":"ok"},
 {"Data":[{"Cod":"8949",
 "Denumire":"BERE NEFILTRATA STAROPRAMEN",
 "Pret":"6",
 "Cant":"1",
 "CodGest":"128",
 "CodUM":"121",
 "DenUM":"Buc",
 "FaraListare":"0",
 "ListareGest":"1",
 "Obs": "aaa"
 },{
 "Cod":"1015",
 "Denumire":"BERE STAROPRAMEN ST/500ML",
 "Pret":"5",
 "Cant":"1",
 "CodGest":"128",
 "CodUM":"126",
 "DenUM":"Sticle",
 "FaraListare":"0",
 "ListareGest":"1",
 "Obs": "bbb"
 },{
 "Cod":"122",
 "Denumire":"BERE STELA ARTOIS FA ST/500 ML",
 "Pret":"7",
 "Cant":"2",
 "CodGest":"128",
 "CodUM":"126",
 "DenUM":"Sticle",
 "FaraListare":"0",
 "ListareGest":"1",
 "Obs": "ccc"
 }
 ]
 }
```

Funcția returnează liniile continute de o comandă (codcomanda este parametru).

Atenție la acel parametru ListareGest care trebuie să se întoarcă la salvarea comenzii.

Mergi la [Top document](#)

Interogare stoc la baza - GetStocArtInRestaurant

Mod invocare: ***http://host:port/datasnap/rest/TServerMethods/GetStocArtInRestaurant/DataIn***

Functia **GetStocArtInRestaurant(CodArticol, CodGest : integer)**

Exemplu returnat:

```
{ "result": "ok",  
  "InfoStoc":  
    { "Stoc": "98",  
      "StocNedescarcat": "90",  
      "StocReal": "8" },  
  "ErrorList": [  
  ] }
```

Functia returneaza stocul de la baza si vanzarile de la restaurant care inca nu au fost preluate pe baza.

Mergi la [Top document](#)