
EMS API Reference Guide

API Version 1.1.35

Dean Evans & Associates, Inc.



Copyright

Copyright© 2014 Dean Evans & Associates, Inc. All rights reserved.

No part of this document may be reproduced, transmitted or stored in a retrieval system in any form, electronic or mechanical, for purposes other than the personal use of the original licensee of the EMS Software except by written permission of Dean Evans & Associates, Inc.

Dean Evans & Associates, Inc.
6465 Greenwood Plaza Boulevard
Suite 600
Centennial, CO 80111

Information in this document is subject to change without notice.

For EMS Customers

The EMS API is software and documentation that is covered by the terms and conditions of your Software License Agreement with Dean Evans & Associates. This document may be copied for use by individuals employed by the EMS licensee. If you plan to use the EMS API internally you may proceed. If you intend to provide access to the EMS API software and/or documentation to a third-party Vendor or Integrator, please see our requirements below.

For Vendors and Integrators

If you are a Vendor or an Integrator or any third-party looking to use the EMS API software and/or documentation, either at the request of a Customer or based on a perceived business opportunity, you must email partners@dea.com to express your interest. We will follow-up to evaluate your company's background and experience and the scope of the intended solution that will use the EMS API. If we approve we will require that you execute our Non-disclosure Agreement in order to receive the API Guide documentation and access to our API testing "sandbox". We will also have the right but not the obligation to list you as a Partner on our website and the option to impose an annual Partner Fee.

EMS, the EMS logo and EMS Regics are registered trademarks of Dean Evans & Associates, Inc. Event Management Systems, EMS Enterprise, EMS Professional, EMS Lite, EMS Campus, EMS Workplace, EMS Legal, EMS District, EMS Master Calendar and Virtual EMS are trademarks of Dean Evans & Associates, Inc. Microsoft, Windows and Outlook are registered trademarks and SQL Server is a trademark of Microsoft Corporation. Lotus Notes and Domino are registered trademarks of IBM Corporation. Other products, brands and trademarks are property of their respective owners/companies.

EMS API Reference Guide

INTRODUCTION	6
CUSTOMER SUPPORT.....	7
EMS API REQUIREMENTS.....	7
EMS API LICENSING	7
HARDWARE AND SOFTWARE REQUIREMENTS	7
OBTAINING THE EMS API WEB SERVICE INSTALLATION FILES	7
INSTALLING/UPGRADING THE DATABASE COMPONENT IN EMS WORKPLACE / CAMPUS / ENTERPRISE / LEGAL / DISTRICT	8
INSTALLING/UPGRADING THE DATABASE COMPONENT IN EMS PROFESSIONAL	8
INSTALLING/UPGRADING THE EMS API WEB SERVICE.....	8
CONFIGURING EMS FOR EMS API WEB SERVICE USE	9
VERIFYING THE EMS API	9
BEST PRACTICES	10
USING FILTERS IS SMART	10
MORE ABOUT EVENTTYPES AND STATUSTYPES	10
UNDERSTANDING DISPLAYONWEB.....	11
ADDBOOKING (REQUIRES EMS API UPDATE LICENSE)	11
ADDBOOKING2 (REQUIRES EMS API UPDATE LICENSE)	12
ADDCONTACT (REQUIRES EMS API UPDATE LICENSE).....	12
ADDGROUP (REQUIRES EMS API UPDATE LICENSE).....	13
ADDPAYMENT (REQUIRES EMS API UPDATE LICENSE).....	14
ADDRESERVATION (REQUIRES EMS API UPDATE LICENSE).....	14
ADDRESERVATION2 (REQUIRES EMS API UPDATE LICENSE).....	15
ADDRESERVATION3 (REQUIRES EMS API UPDATE LICENSE).....	16
ADDRESERVATION4 (REQUIRES EMS API UPDATE LICENSE).....	16
ADDUDF (REQUIRES EMS API UPDATE LICENSE).....	17
ADDWEBREQUEST(REQUIRES EMS API UPDATE LICENSE).....	18
ADDWEBUSER (REQUIRES EMS API UPDATE LICENSE)	18

GETALLBOOKINGS.....	19
GETALLBOOKINGS2.....	21
GETALLCOMBOROOMCOMPONENTS.....	23
GETALLROOMBOOKINGS.....	24
GETALLROOMS	26
GETAPIVERSION.....	27
GETAREAS	27
GETBOOKING	28
GETBOOKINGCHECKIN	30
GETBOOKINGS.....	30
GETBOOKINGS2.....	33
GETBUILDINGHOLIDAYS	36
GETBUILDINGHOURS	37
GETBUILDINGS	38
GETCHANGEDBOOKINGS.....	38
GETCHECKINSTATUS	41
GETCOMBOROOMCOMPONENTS.....	42
GETCOMMENTTYPES	43
GETCONTACTDETAILS	43
GETCONTACTS.....	44
GETCOURSEBOOKINGS	45
GETEVENTTYPES	47
GETFEATURES	48
GETGROUPBOOKINGS.....	49
GETGROUPDETAILS.....	51
GETGROUPS.....	52
GETGROUPTYPES	52
GETHVACBOOKINGS	53

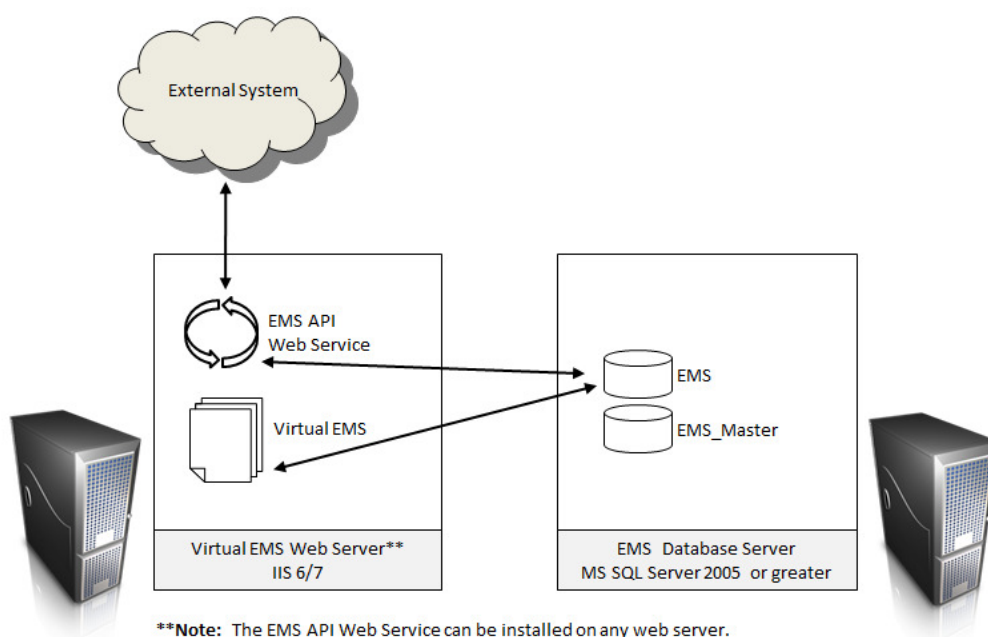
GETRESERVATIONCOMMENTS.....	56
GETRESERVATIONSOURCES.....	56
GETROOMAVAILABILITY	57
GETROOMBOOKINGS	58
GETROOMDETAILS	60
GETROOMS.....	61
GETROOMSAVAILABLE	62
GETROOMSAVAILABLE2	63
GETROOMSAVAILABLE3	65
GETROOMSAVAILABLE4	67
GETROOMSBYSETUPTYPE.....	69
GETROOMTYPESBYWPT	70
GETSERVICEORDERDETAILS	71
GETSERVICEORDERDETAILS2	72
GETSETUPTYPES	74
GETSTATUSES	74
GETUDFDEFINITIONS	75
GETUDFLISTITEMS.....	76
GETWEBPROCESSTEMPLATECATEGORIES	76
GETWEBSECURITYTEMPLATEROLES	77
GETWEBPROCESSTEMPLATESSETTINGS	78
GETWEBTEMPLATES.....	79
GETWEBUSERBOOKINGS	80
GETWEBUSERDETAILS	82
GETWEBUSEROPTIONS.....	83
GETWEBUSERS.....	84
GETWEBUSERWEBPROCESSTEMPLATES.....	84
UPDATEBOOKING (REQUIRES EMS API UPDATE LICENSE).....	85

UPDATEBOOKING2 (REQUIRES EMS API UPDATE LICENSE).....	86
UPDATEBOOKINGCHECKIN (REQUIRES EMS API UPDATE LICENSE).....	86
UPDATECONTACT (REQUIRES EMS API UPDATE LICENSE)	87
UPDATECHECKINSTATUS (REQUIRES EMS API UPDATE LICENSE).....	88
UPDATEGROUP (REQUIRES EMS API UPDATE LICENSE).....	88
UPDATERESERVATION (REQUIRES EMS API UPDATE LICENSE)	89
UPDATEUDF (REQUIRES EMS API UPDATE LICENSE)	90
UPDATEWEBUSER (REQUIRES EMS API UPDATE LICENSE)	90
UPDATEWEBUSEROPTIONS (REQUIRES EMS API UPDATE LICENSE)	91
RELEASE NOTES	92

Introduction

This is the EMS API Reference Guide. This guide is intended for programmers that need detailed information about our web service based API. Use of this guide assumes you are familiar with the following:

- XML
- Web services
- Query or SOAP
- One or more programming languages



Note: If an offsite external third party integrator (i.e. outside of your network) requires access to the EMS API, the EMS API Web Service will need to be installed as a public facing website.

Customer Support

Unlimited toll-free customer support is available to EMS users who have a current Annual Service Agreement (ASA). Please contact Customer Support if you have any questions or encounter any issues installing or configuring the EMS API.

Email:	support@dea.com
Web:	www.dea.com
Phone:	(800) 288-4565
Fax:	(303) 796-7429

EMS API Requirements

The latest version of EMS must be installed:

- EMS Workplace 5.2.1 or greater
- EMS Campus 2.2.1 or greater
- EMS Enterprise 5.2.1 or greater
- EMS Legal 5.2.1 or greater
- EMS District 6.0 or greater
- EMS Professional 11.1 or greater

EMS API Licensing

The Read Only API and the Updating API are both optional system features that may be licensed by EMS customers. Note that the Updating API includes the Read Only API.

The Read Only API provides all of the calls to pull data (i.e. “get” calls) from the EMS database. Calls to update or add reservations/bookings (i.e. “update” and “add” calls) also require the Updating API. If you are unsure whether or not your organization is licensed for the API, or if you would like to learn more, please contact your Account Executive.

Hardware and Software Requirements

The system requirements for the EMS API are similar to the Virtual EMS requirements. System requirements can be found on our [website](#).

Obtaining the EMS API Web Service Installation Files

Please contact your Professional Services Consultant or Customer Support to obtain the following components:

- **EMS API (EMSAPI.msi)** – Installation file
- **EMSAPIScript_WebService.sql** - SQL file required to create the necessary API objects within your EMS database.

Installing/Upgrading the Database Component in EMS Workplace / Campus / Enterprise / Legal / District

1. Obtain the **EMSAPIScript_WebService.sql** file.
2. Open *Microsoft SQL Server Management Studio*.
3. Within *Microsoft SQL Server Management Studio*, select **File > Open > File...** and locate **EMSAPIScript_WebService.sql**.
4. Select your EMS database and execute the query (**Query > Execute**). When query execution has completed, a 'Command(s) completed successfully' message will display in the Results section.

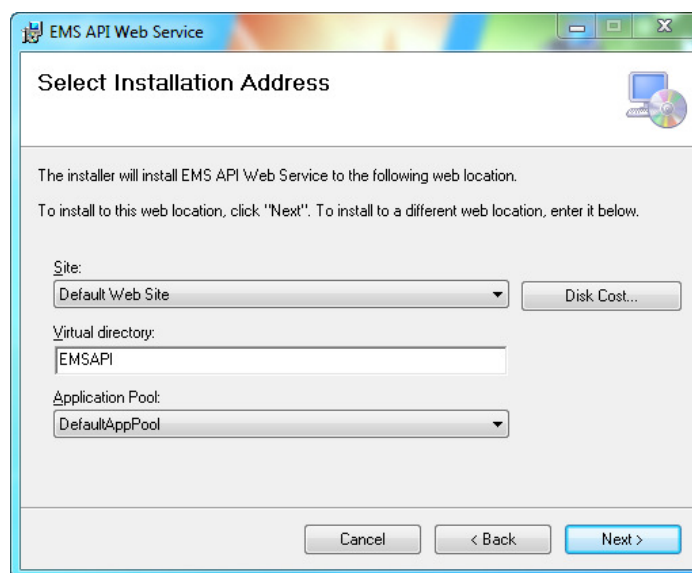
Installing/Upgrading the Database Component in EMS Professional

Please contact your Professional Services Consultant or Customer Support.

Installing/Upgrading the EMS API Web Service

Important: Before beginning the installation process, please do the following:

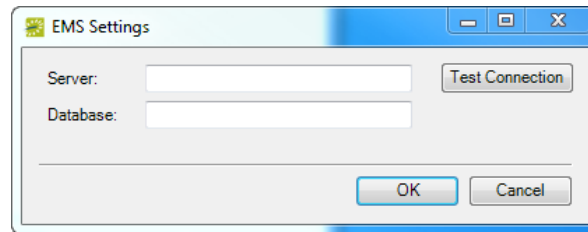
- Install or upgrade your EMS databases as outlined in the *EMS Installation Instructions*.
 - Manually **uninstall** any previous versions of the EMS API Web Service on your web server.
1. Verify that the prerequisite software is installed.
 2. Download **EMSAPI.msi** onto the web server.
 3. Run **EMSAPI.msi**.
 4. The first screen welcomes you to the EMS API Web Service Setup Wizard. Click **Next >** to begin the installation process. The *Select Installation Address* screen will appear.



5. Select the Site, Virtual Directory and Application Pool. It is recommended that you keep the default settings. The installation process will create a Physical and Virtual directory on your web server named "EMSAPI" (or whatever value you entered in the Virtual Directory field.) Click **Next >** to begin the installation process.

Note: The EMS API should not be installed in the same Virtual Directory as other EMS web-based products.

- The *Confirm Installation* screen will appear. Click **Next >**. The *EMS Settings* screen will appear.



- Enter your EMS SQL Server Name.
- Enter your EMS Database:
 - EMS Professional customers – enter “EMSDData”
 - EMS Workplace, EMS Campus, EMS Enterprise, EMS Legal and EMS District customers - typically named “EMS”
- Click **OK**.
- The *Installation Complete* screen will appear. Click **Close**.

Configuring EMS for EMS API Web Service Use

External applications require a pre-defined EMS User account to authenticate to the EMS API Web Service. The EMS User account should not be granted any rights within the EMS application itself.

- Launch the EMS client application and login with a user that has EMS administrator-level access.
- Go to **System Administration > Security > Users**.
- Click the **New** button. The *User* screen will appear.
- Input the following information:
 - Name
 - User ID
 - Password
- On the *Processes* screen, select the *Special – EMS API Web Service Access* process.

Note: If the *Special – EMS API Web Service Access* process does not exist, please contact Customer Support.

- Click the **OK** button.

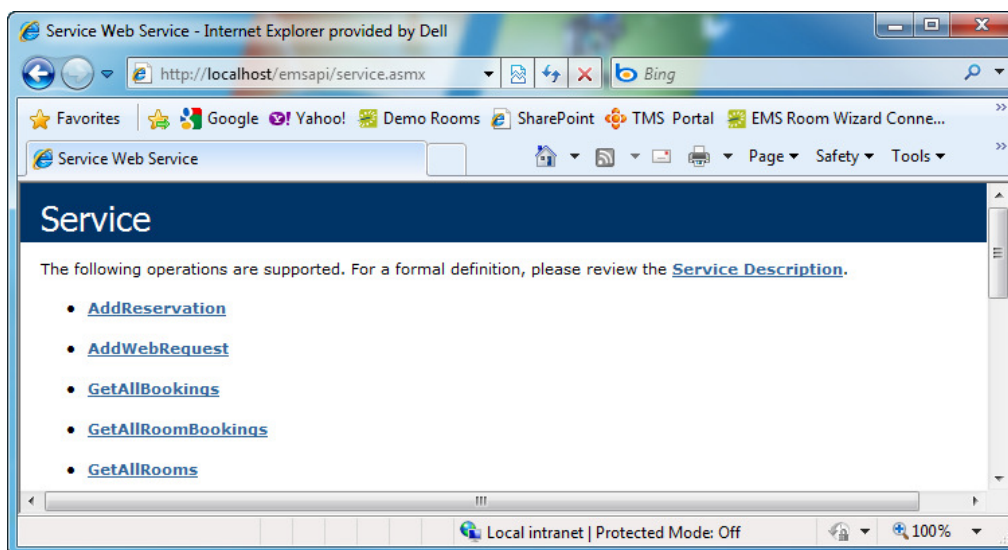
Note: In the *Sample Code* examples below, use the EMS account created in this section for the **UserID** and **Password**.

Verifying the EMS API

- Open a browser and enter the following URL:

[http://\[ServerName\]/EMSAPI/Service.asmx](http://[ServerName]/EMSAPI/Service.asmx) (replace [ServerName] with the name of your web server)

Your browser should return a list of API calls.



The EMS API version can be verified by going to the About page:

[http://\[ServerName\]/EMSAPI/About.aspx](http://[ServerName]/EMSAPI/About.aspx) (replace [ServerName] with the name of your web server)

Best Practices

Using filters is smart

A leading management consulting firm, with over 250,000 employees, has used EMS for more than 10 years to schedule and manage Classrooms, Lobby Areas, Meeting and Conference Rooms, Hot Desks, Parking Spaces and Video Conference Rooms. EMS Workplace manages 330 customer buildings around the world, comprising 33,095 rooms and hot-desks. In 2011, the number of bookings scheduled was 2,234,086. That's a lot of data!

We don't put a limit on the amount of data you can pull through the API, but we do provide numerous API call filters so you can request an efficient and targeted set of data. This helps reduce the work-load on the EMS system. It also improves the performance of your solution – why pull in a lot of extraneous data that you don't need or use?

More about EventTypes and StatusTypes

One of the reasons that EMS is so successful is because it's so flexible. That flexibility gives the user a lot of freedom in customizing the system. For example, EventTypes are user-defined - there is no single definitive list, they will be different at each EMS customer. You can get all the EventTypes using the GetEventTypes call.

The same is true for Statuses – these are user-defined, no single definitive list. You can get all the Statuses using the GetStatuses call.

Each Status has to be one of four StatusTypes:

Booked Space—the room is taken out of inventory and cannot be reserved by another booking. (The system prevents the space from being double-booked.)

Cancel—booking information is retained, but the room is released back into inventory.

Info Only—Booking information is retained, but the space is not taken out of inventory.

Wait—booking information is retained in case the space becomes available prior to the event date.

StatusTypeIDs are as follows: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel

Understanding DisplayOnWeb

The `GetEventTypes` and `GetStatuses` calls above both return a field called `DisplayOnWeb`. That field is set by the customer when defining a new `EventType` or a new `Status` – it's either Yes or No.

`DisplayOnWeb` is the field we use in our various web interfaces to EMS, including Virtual EMS and our Kiosk, to determine whether to display an event or not. Partners can use `DisplayOnWeb` to make a similar determination ... and then you don't need to worry about mapping `EventTypes` and `Statuses`. Plus this makes your solution consistent with the behavior that customers already understand about `DisplayOnWeb`.

Taken one step further, you can also determine which `EventTypes` and `Statuses` *may* be displayed (`DisplayOnWeb=Yes`). You can use those `EventTypes` and `Statuses` as filters on some of the `GetBooking` calls – the result is you get returned just the booking data needed rather than pulling in everything and then discarding a bunch of bookings.

API Functions

AddBooking (Requires EMS API Update License)

Description

Adds a Booking to a Reservation in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ReservationID	Reservation ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
EventName	Event Name	String
StatusID	Status ID	Integer

Response Elements

Name	Description	Type
BookingID	Booking ID that was created in EMS	Integer

Sample Code

```
Private Sub AddBooking()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddBooking("UserID", "Password", 1234, 3199, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"), "Meeting", 1))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Booking>
  <Data>
    <BookingID>4567</BookingID>
  </Data>
```

</Booking>

AddBooking2 (Requires EMS API Update License)

Description

Adds a Booking to a Reservation in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ReservationID	Reservation ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
EventName	Event Name	String
StatusID	Status ID	Integer
EventTypeID	Event Type ID	Integer

Response Elements

Name	Description	Type
BookingID	Booking ID that was created in EMS	Integer

Sample Code

```
Private Sub AddBooking()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddBooking2("UserID", "Password", 1234, 3199, Date.Today,
    CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"), "Meeting", 1, 1))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Booking>
  <Data>
    <BookingID>4567</BookingID>
  </Data>
</Booking>
```

AddContact (Requires EMS API Update License)

Description

Creates a Contact in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer
ContactName	Contact Name	String
Title	Title	String
Address1	Address Line 1	String
Address2	Address Line 2	String
City	City	String
State	State	String

ZipCode	Zip Code	String
Country	Country	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String

Response Elements

Name	Description	Type
ContactID	Contact ID that was created in EMS	Integer

Sample Code

```
Private Sub AddContact()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddContact("UserID", "Password", 1234,"Test Contact", "", "123
Main St.", "", "Denver", "CO", "80111", "", "x123", "", "Test@test.com", ""))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Contact>
  <Data>
    <ContactID>12345</ContactID>
  </Data>
</Contact>
```

AddGroup (Requires EMS API Update License)

Description

Creates a Group in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupName	Group Name	String
GroupTypeID	Group Type ID	Integer
Address1	Address Line 1	String
Address2	Address Line 2	String
City	City	String
State	State	String
ZipCode	Zip Code	String
Country	Country	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String

Response Elements

Name	Description	Type
GroupID	Group ID that was created in EMS	Integer

Sample Code

```
Private Sub AddGroup()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddGroup("UserID", "Password","Test Group", 0, "123 Main St.", "",
"Denver", "CO", "80111", "", "x123", "", "Test@test.com", ""))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```

<Group>
  <Data>
    <GroupID>1234</GroupID>
  </Data>
</Group>

```

AddPayment (Requires EMS API Update License)**Description**

Adds a payment to an Invoice in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
TransactionDate	Transaction Date	Date
PaymentTypeID	Payment Type ID	Integer
CheckNo	Check Number	String
PaymentAmount	Payment Amount	Decimal
InvoiceNo	Invoice Number	String
Notes	Notes	String

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Invoice Number is invalid."	String

Sample Code

```

Private Sub AddPayment()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddPayment("UserID", "Password", #12/15/2011#, 0,
"1234", "INV100", ""))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>

```

AddReservation (Requires EMS API Update License)**Description**

Creates a Reservation in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
EventName	Event Name	String

StatusID	Status ID	Integer
----------	-----------	---------

Response Elements

Name	Description	Type
ReservationID	Reservation ID that was created in EMS	Integer
BookingID	Booking ID that was created in EMS	Integer

Sample Code

```
Private Sub AddReservation()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddReservation("UserID", "Password", 11, 3199, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"), "Meeting", 1))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Reservation>
  <Data>
    <ReservationID>1234</ReservationID>
    <BookingID>4567</BookingID>
  </Data>
</Reservation>
```

AddReservation2 (Requires EMS API Update License)**Description**

Creates a Reservation in EMS tied to a Web User

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
EventName	Event Name	String
StatusID	Status ID	Integer
EventTypeID	Event Type ID	Integer
WebUserID	Web User ID	Integer
WebTemplateID	Web Template ID	Integer

Response Elements

Name	Description	Type
ReservationID	Reservation ID that was created in EMS	Integer
BookingID	Booking ID that was created in EMS	Integer

Sample Code

```
Private Sub AddReservation2()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddReservation2("UserID", "Password", 11, 3199, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"), "Meeting", 1, 2, 468, 2))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Reservation>
  <Data>
```

```

        <ReservationID>1234</ReservationID>
        <BookingID>4567</BookingID>
    </Data>
</Reservation>

```

AddReservation3 (Requires EMS API Update License)

Description

Creates a Reservation in EMS tied to a Web User

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
EventName	Event Name	String
StatusID	Status ID	Integer
EventTypeID	Event Type ID	Integer
WebUserID	Web User ID	Integer
WebTemplateID	Web Template ID	Integer
ReservationSourceID	Reservation Source ID	Integer

Response Elements

Name	Description	Type
ReservationID	Reservation ID that was created in EMS	Integer
BookingID	Booking ID that was created in EMS	Integer

Sample Code

```

Private Sub AddReservation3()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddReservation3("UserID", "Password", 11, 3199, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"), "Meeting", 1, 2, 468, 2, 1))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Reservation>
  <Data>
    <ReservationID>1234</ReservationID>
    <BookingID>4567</BookingID>
  </Data>
</Reservation>

```

AddReservation4 (Requires EMS API Update License)

Description

Creates a Reservation in EMS tied to a Web User and includes a Billing Reference

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer
RoomID	Room ID	Integer

BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
EventName	Event Name	String
StatusID	Status ID	Integer
EventTypeID	Event Type ID	Integer
WebUserID	Web User ID	Integer
WebTemplateID	Web Template ID	Integer
ReservationSourceID	Reservation Source ID	Integer
BillingReference	Billing Reference	String

Response Elements

Name	Description	Type
ReservationID	Reservation ID that was created in EMS	Integer
BookingID	Booking ID that was created in EMS	Integer

Sample Code

```
Private Sub AddReservation4()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddReservation4("UserID", "Password", 11, 3199, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"),"Meeting",1, 2, 468, 2, 1, "abc"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Reservation>
  <Data>
    <ReservationID>1234</ReservationID>
    <BookingID>4567</BookingID>
  </Data>
</Reservation>
```

AddUDF (Requires EMS API Update License)

Description

Creates a UDF in EMS (*Reservation or Booking level UDF only*)

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ParentLevelID	EMS Reservation Level: Reservation = 0, Booking = 1	Integer
ParentID	Reservation ID or Booking ID	Integer
UDFDefID	UDF Definition ID	Integer
UDFValue	UDF Value	String

Response Elements

Name	Description	Type
UDFID	UDF ID that was created in EMS	Integer

Sample Code

```
Private Sub AddUDF()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddUDF("UserID", "Password", 1, 2, 1, "udf text"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```

<UDF>
  <Data>
    <UDFID>12345</UDFID>
  </Data>
</UDF>

```

AddWebRequest(Requires EMS API Update License)

Description

Creates a Web Request in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
EventName	Event Name	String
EventTypeID	Event Type ID	Integer
GroupName	Group Name	String
Contact	Contact	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
WebUserID	Web User ID	Integer
BuildingID	Building ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
SetupTypeID	Setup Type ID	Integer
SetupCount	Setup Count	Integer
Notes	Notes	String

Response Elements

Name	Description	Type
WebRequestID	Web Request ID that was created in EMS	Integer

Sample Code

```

Private Sub AddWebRequest()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddWebRequest("UserID", "Password", "Meeting", 5, "ABC
Company", "John Smith", "555-1212", "555-1213", "JohnSmith@abc.com", 25, 1, 3199, Date.Today, CDate("1/1/1900 8:00 AM"),
CDate("1/1/1900 9:00 AM"), 3,10, "Notes..."))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<WebRequest>
  <Data>
    <WebRequestID>1234</WebRequestID>
  </Data>
</WebRequest>

```

AddWebUser (Requires EMS API Update License)

Description

Creates a Web User in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebUserName	Web User Name	String
WebUserPassword	Web User Password	String
EmailAddress	Email Address	String
Phone	Phone	String
Fax	Fax	String
ExternalReference	External Reference	String
NetworkID	Network ID	String
TimeZoneID	Time Zone ID	Integer
StatusID	Status ID (0=Active, 1=Pending, 2=Inactive, 3=Activate On Login)	Integer
WebSecurityTemplateID	Web Security Template ID	Integer
WebProcessTemplates	Web Process Template List	Array Of Integers
Groups	Group List	Array Of Integers

Response Elements

Name	Description	Type
WebUserID	Web User ID that was created in EMS	Integer

Sample Code

```
Private Sub AddWebUser()
    Dim gintWebProcessTemplates As New Generic.List(Of Int32)
    Dim gintGroups As New Generic.List(Of Int32)

    gintWebProcessTemplates.Add(1)
    gintWebProcessTemplates.Add(2)
    gintWebProcessTemplates.Add(3)

    gintGroups.Add(1234)

    Dim strWebTemplates As New EMSAPI.ArrayOfInt
    Dim strGroups As New EMSAPI.ArrayOfInt

    strWebTemplates.AddRange(gintWebProcessTemplates.ToArray)
    strGroups.AddRange(gintGroups.ToArray)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.AddWebUser("UserID", "Password", "New Web User", "password",
"newwebuser@test.com", "x123", "", "", "", 68, 0, 1, strWebTemplates, strGroups))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<WebUser>
  <Data>
    <WebUserID>2314</WebUserID>
  </Data>
</WebUser>
```

GetAllBookings

Description

Returns a list of bookings for a specific building

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
BuildingID	Building ID (-1= All Buildings)	Integer
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
HVACZone	Room HVAC Zone	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```
Private Sub GetBookings()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetAllBookings("UserID", "Password", Date.Today, Date.Today, 1, False))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```

<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>false</VIPEvent>
    <ClosedAllDay>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</RoomCode>
    <RoomID>3271</RoomID>
    <BuildingID>1</BuildingID>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <HVACZone/>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <DateAdded>2009-10-04T16:00:00</DateAdded>
    <AddedBy>User 1</AddedBy>
    <DateChanged>2009-10-04T16:00:00</DateChanged>
    <ChangedBy>User 1</ChangedBy>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetAllBookings2**Description**

Returns a list of bookings for a specific building for a specific UDF value (*Reservation or Booking level UDF only*)

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

StartDate	Start Date	DateTime
EndDate	End Date	DateTime
BuildingID	Building ID (-1 = All Buildings)	Integer
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean
UDFDefID	ID of UDF Definition	Integer
UDFValue	UDF Value	String

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
HVACZone	Room HVAC Zone	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```
Private Sub GetBookings2()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetAllBookings2("UserID", "Password", Date.Today, Date.Today, 1, False, 1,
"udf text"))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```

<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>false</VIPEvent>
    <ClosedAllDay>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</RoomCode>
    <RoomID>3271</RoomID>
    <BuildingID>1</BuildingID>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <HVACZone/>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <DateAdded>2009-10-04T16:00:00</DateAdded>
    <AddedBy>User 1</AddedBy>
    <DateChanged>2009-10-04T16:00:00</DateChanged>
    <ChangedBy>User 1</ChangedBy>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetAllComboRoomComponents**Description**

Returns component room details for all combo rooms

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
ParentRoom	Parent Room Code	String
ParentDescription	Parent Room Description	String
ParentBuilding	Parent Building	String
ParentID	Parent Room ID	Integer
ComponentRoom	Component Room Code	String
ComponentDescription	Component Room Description	String
ComponentBuilding	Component Building	String
ComponentID	Component Room ID	Integer

Sample Code

```
Private Sub GetAllComboRoomComponents()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetAllComboRoomComponents("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<RoomComponents>
  <Data>
    <ParentRoom>RM 101</Room>
    <ParentDescription>Room 101</Description>
    <ParentBuilding>Denver</Building>
    <ParentID>3271</ParentID>
    <ComponentRoom>Room 102 </ComponentRoom>
    <ComponentDescription>Room 102</Description>
    <ComponentBuilding>Denver</Building>
    <ComponentID>3272</ComponentID>
    <ParentRoom>RM 101</Room>
    <ParentDescription>Room 101</Description>
    <ParentBuilding>Denver</Building>
    <ParentID>3273</ParentID>
    <ComponentRoom>Room 103 </ComponentRoom>
    <ComponentDescription>Room 103</Description>
    <ComponentBuilding>Denver</Building>
    <ComponentID>3273</ComponentID>
  </Data>
</ RoomComponents >
```

GetAllRoomBookings

Description

Returns a list of bookings for a specific room

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
RoomID	Room ID	Integer
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime

GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
BuildingID	Building ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```
Private Sub GetBookings()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetAllRoomBookings("UserID", "Password", Date.Today, Date.Today, 3271,
False))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
```

```

    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>false</VIPEvent>
    <ClosedAllDay>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</Room>
    <RoomID>3271</RoomID>
    <BuildingID>1</BuildingID>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <DateAdded>2009-10-04T16:00:00</DateAdded>
    <AddedBy>User 1</AddedBy>
    <DateChanged>2009-10-04T16:00:00</DateChanged>
    <ChangedBy>User 1</ChangedBy>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetAllRooms

Description

Returns list of rooms

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	Building ID (-1= All Buildings)	Integer

Response Elements

Name	Description	Type
Room	Room Code	String
Description	Room Description	String
Building	Building	String
Floor	Floor Description	String
Type	Room Type	String
Classification	Room Classification	String
Phone	Room Phone	String
Notes	Room Notes	String
RoomSize	Room Size	Integer
SetupHours	Room Setup Hours	Decimal
TeardownHours	Room Teardown Hours	Decimal
HVACZone	Room HVAC Zone	String
URL	Room URL	String
Active	Room Active	Boolean
BuildingID	Building ID	Integer

ID	Room ID	Integer
ExternalReference	Room External Reference	String
DisplayOnWeb	Room Display On Web	Boolean

Sample Code

```
Private Sub GetRooms()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetAllRooms("UserID", "Password", 1))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Rooms>
  <Data>
    <Room>RM 101</Room>
    <Description>Room 101</Description>
    <Building>Denver</Building>
    <Floor>1st Floor</Floor>
    <Type>Meeting Room</Type>
    <Classification>Standard</Classification>
    <Phone>x123</Phone>
    <Notes />
    <RoomSize>20</RoomSize>
    <SetupHours>0</SetupHours>
    <TeardownHours>0</TeardownHours>
    <HVACZone />
    <URL />
    <Active>1</Active>
    <BuildingID>1</BuildingID>
    <ID>3271</ID>
    <ExternalReference />
    <DisplayOnWeb>1</DisplayOnWeb>
  </Data>
</Rooms>
```

GetAPIVersion

Description

Returns API version

Request Parameters – (none)

Example Response

```
<API>
  <APIVersion>
    <Version>1.1.19.0</Version>
    <License>Advanced</License>
  </APIVersion>
</API>
```

Note: the Read Only API returns a value of Basic, the Updating API returns a value of Advanced

GetAreas

Description

Returns a list of areas and associated buildings

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
AreaID	Area ID (-1 = All Areas)	Integer

Response Elements

Name	Description	Type
ID	AreaID	Integer
AreaDescription	Area Description	String
AreaNotes	Area Notes	String
BuildingID	Building ID	Integer

Sample Code

```
Private Sub GetAreas()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetAreas("UserID", "Password", 12))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Areas>
  <Data>
    <ID>12</ID>
    <AreaDescription>Midwest</AreaDescription>
    <AreaNotes></AreaNotes>
    <BuildingID>1</BuildingID>
  </Data>
  <Data>
    <ID>12</ID>
    <AreaDescription>Midwest</AreaDescription>
    <AreaNotes></AreaNotes>
    <BuildingID>2</BuildingID>
  </Data>
</Areas>
```

GetBooking**Description**

Returns booking information for a specific booking

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BookingID	Booking ID	Integer

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime

CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
BuildingID	Building ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
RoomTypeID	Room Type ID	Integer
BuildingID	Building ID	Integer
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
WebProcessTemplateID	Web Process Template ID	Integer

Sample Code

```
Private Sub GetBooking()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBooking("UserID", "Password", 222008))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Booking>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>false</VIPEvent>
    <ClosedAllDay>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
    <BuildingCode>DEN</BuildingCode>
```

```

    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</RoomCode>
    <RoomID>3271</RoomID>
    <RoomTypeID>0</RoomTypeID>
    <BuildingID>1</BuildingID>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <WebProcessTemplateID>0</WebProcessTemplateID>
  </Data>
</Booking>

```

GetBookingCheckIn

Description

Returns check in properties for a booking

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BookingID	Booking ID	Integer

Response Elements

Name	Description	Type
BookingID	Booking ID	Integer
CheckedIn	Checked In	Boolean
CheckInDate	Date/Time Checked In	DateTime

Sample Code

```

Private Sub GetBookingCheckIn()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetBookingCheckIn("UserID", "Password", 12345))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Bookings>
  <Data>
    <BookingID>12345</BookingID>
    <CheckedIn>true</CheckedIn>
    <CheckInDate>2015-01-07T11:17:22.833</CheckInDate>
  </Data>
</Bookings>

```

GetBookings

Description

Returns a list of bookings based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime

EndDate	End Date	DateTime
Buildings	List of Building IDs	Array Of Integers
Statuses	List of Status IDs	Array Of Integers
EventTypes	List of Event Type IDs	Array Of Integers
GroupTypes	List of Group Type IDs	Array Of Integers
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
BuildingID	Building ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
HVACZone	Room HVAC Zone	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```
Private Sub GetBookings()
    Dim dwvBuildings, dwvStatuses, dwvEventTypes, dwvGroupTypes As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintEventTypes As New Generic.List(Of Int32)
    Dim gintGroupTypes As New Generic.List(Of Int32)
```

```

Dim svc As New EMSAPI.ServiceSoapClient
Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

dvwBuildings = Me.GetData(objReader)

objReader = New StringReader(svc.GetStatuses("UserID", "Password"))
dvwStatuses = Me.GetData(objReader)

objReader = New StringReader(svc.GetEventTypes("UserID", "Password"))
dvwEventTypes = Me.GetData(objReader)

objReader = New StringReader(svc.GetGroupTypes("UserID", "Password"))
dvwGroupTypes = Me.GetData(objReader)

For Each drv As DataRowView In dvwBuildings
    gintBuildings.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwStatuses
    gintStatuses.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwEventTypes
    gintEventTypes.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwGroupTypes
    gintGroupTypes.Add(drv("ID"))
Next

Dim strBuildings As New EMSAPI.ArrayOfInt
Dim strStatuses As New EMSAPI.ArrayOfInt
Dim strEventTypes As New EMSAPI.ArrayOfInt
Dim strGroupTypes As New EMSAPI.ArrayOfInt

strBuildings.AddRange(gintBuildings.ToArray)
strStatuses.AddRange(gintStatuses.ToArray)
strEventTypes.AddRange(gintEventTypes.ToArray)
strGroupTypes.AddRange(gintGroupTypes.ToArray)

objReader = New StringReader(svc.GetBookings("UserID", "Password", Date.Today, Date.Today, strBuildings,
strStatuses, strEventTypes, strGroupTypes, False))

Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>false</VIPEvent>
    <ClosedAllDay>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
  
```



```

    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</Room>
    <RoomID>3271</RoomID>
    <BuildingID>1</BuildingID>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <HVACZone/>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <DateAdded>2009-10-04T16:00:00</DateAdded>
    <AddedBy>User 1</AddedBy>
    <DateChanged>2009-10-04T16:00:00</DateChanged>
    <ChangedBy>User 1</ChangedBy>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetBookings2

Description

Returns a list of bookings based on filters provided and Reservation ID

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ReservationID	Reservation ID	Integer
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
Buildings	List of Building IDs	Array Of Integers
Statuses	List of Status IDs	Array Of Integers
EventTypes	List of Event Type IDs	Array Of Integers
GroupTypes	List of Group Type IDs	Array Of Integers
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer

VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
BuildingID	Building ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
HVACZone	Room HVAC Zone	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```

Private Sub GetBookings2()
    Dim dwvBuildings, dwvStatuses, dwvEventTypes, dwvGroupTypes As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintEventTypes As New Generic.List(Of Int32)
    Dim gintGroupTypes As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dwvBuildings = Me.GetData(objReader)

    objReader = New StringReader(svc.GetStatuses("UserID", "Password"))
    dwvStatuses = Me.GetData(objReader)

    objReader = New StringReader(svc.GetEventTypes("UserID", "Password"))
    dwvEventTypes = Me.GetData(objReader)

    objReader = New StringReader(svc.GetGroupTypes("UserID", "Password"))
    dwvGroupTypes = Me.GetData(objReader)

    For Each drv As DataRowView In dwvBuildings
        gintBuildings.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dwvStatuses
        gintStatuses.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dwvEventTypes
        gintEventTypes.Add(drv("ID"))
    Next

```

Next

```
For Each drv As DataRowView In dvwGroupTypes
    gintGroupTypes.Add(drv("ID"))
```

Next

```
Dim strBuildings As New EMSAPI.ArrayOfInt
Dim strStatuses As New EMSAPI.ArrayOfInt
Dim strEventTypes As New EMSAPI.ArrayOfInt
Dim strGroupTypes As New EMSAPI.ArrayOfInt
```

```
strBuildings.AddRange(gintBuildings.ToArray)
strStatuses.AddRange(gintStatuses.ToArray)
strEventTypes.AddRange(gintEventTypes.ToArray)
strGroupTypes.AddRange(gintGroupTypes.ToArray)
```

```
objReader = New StringReader(svc.GetBookings2("UserID", "Password", 6650, Date.Today, Date.Today, strBuildings,
strStatuses, strEventTypes, strGroupTypes, False))
```

```
Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
```

End Sub

Example Response

```
<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>false</VIPEvent>
    <ClosedAllDay>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</Room>
    <RoomID>3271</RoomID>
    <BuildingID>1</BuildingID>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <HVACZone/>
    <StatusID>1</StatusID>
    <StatusTypeID>14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <DateAdded>2009-10-04T16:00:00</DateAdded>
    <AddedBy>User 1</AddedBy>
```

```

    <DateChanged>2009-10-04T16:00:00</DateChanged>
    <ChangedBy>User 1</ChangedBy>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetBuildingHolidays

Description

Returns a list of holidays for a specific date

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
Buildings	List of Building IDs	Array of Integers
HolidayDate	Date	DateTime

Response Elements

Name	Description	Type
BuildingID	BuildingID (-1 = All Buildings)	Integer
HolidayDescription	Holiday Description	String
HolidayNotes	Holiday Notes	String
DisplayOnWeb	Display on Web	Boolean

Sample Code

```

Private Sub GetBuildingHolidays()
    Dim dvwBuildings As DataView
    Dim gintBuildings As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dvwBuildings = Me.GetData(objReader)

    For Each drv As DataRowView In dvwBuildings
        gintBuildings.Add(drv("ID"))
    Next

    Dim strBuildings As New EMSAPI.ArrayOfInt

    strBuildings.AddRange(gintBuildings.ToArray)
    objReader = New System.IO.StringReader(svc.GetBuildingHolidays("UserID", "Password", strBuildings, Date.Today))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Holidays>
  <Data>
    <BuildingID>141</BuildingID>
    <HolidayDescription>Independence Day</HolidayDescription>
    <DisplayOnWeb>true</DisplayOnWeb>
  </Data>
  <Data>
    <BuildingID>142</BuildingID>
    <HolidayDescription>Independence Day</HolidayDescription>
    <DisplayOnWeb>true</DisplayOnWeb>
  </Data>
  <Data>
    <BuildingID>143</BuildingID>
    <HolidayDescription>Independence Day</HolidayDescription>
    <DisplayOnWeb>true</DisplayOnWeb>
  </Data>
</Holidays>

```

```
</Data>
</Holidays>
```

GetBuildingHours

Description

Returns a list of building hours for a specific date

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
Buildings	List of Building IDs	Array of Integers
BuildingHoursDate	Date	DateTime

Response Elements

Name	Description	Type
BuildingID	BuildingID	Integer
OpenTime	Building Open Time	DateTime
CloseTime	Building Close Time	DateTime
ClosedAllDay	Closed All Day	Boolean
HolidayBuilding	Building Holiday	Boolean

Sample Code

```
Private Sub GetBuildingHours()
    Dim dvwBuildings As DataView
    Dim gintBuildings As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dvwBuildings = Me.GetData(objReader)

    For Each drv As DataRowView In dvwBuildings
        gintBuildings.Add(drv("ID"))
    Next

    Dim strBuildings As New EMSAPI.ArrayOfInt

    strBuildings.AddRange(gintBuildings.ToArray)
    objReader = New System.IO.StringReader(svc.GetBuildingHours ("UserID", "Password", strBuildings, Date.Today))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<BuildingHours>
  <Data>
    <BuildingID>141</BuildingID>
    <OpenTime>1900-01-01T05:00:00</OpenTime>
    <CloseTime>1900-01-01T19:00:00</CloseTime>
    <ClosedAllDay>false</ClosedAllDay>
    <HolidayBuilding>false</HolidayBuilding>
  </Data>
  <Data>
    <BuildingID>142</BuildingID>
    <OpenTime>1900-01-01T05:00:00</OpenTime>
    <CloseTime>1900-01-01T19:00:00</CloseTime>
    <ClosedAllDay>false</ClosedAllDay>
    <HolidayBuilding>false</HolidayBuilding>
  </Data>
  <Data>
    <BuildingID>143</BuildingID>
    <OpenTime>1900-01-01T05:00:00</OpenTime>
    <CloseTime>1900-01-01T19:00:00</CloseTime>
```

```

        <ClosedAllDay>false</ClosedAllDay>
        <HolidayBuilding>false</HolidayBuilding>
    </Data>
</BuildingHours>

```

GetBuildings

Description

Returns a list of buildings

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
Description	Building Description	String
BuildingCode	Building Code	String
ID	Building ID	Integer
TimeZoneDescription	Time Zone Description	String
TimeZoneAbbreviation	Time Zone Abbreviation	String
CurrentLocalTime	Current Local Time	DateTime

Sample Code

```

Private Sub GetBuildings()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetBuildings("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Buildings>
  <Data>
    <Description>Denver</Description>
    <ID>1</ID>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
    <CurrentLocalTime>2012-04-13T07:47:12.503</CurrentLocalTime>
  </Data>
  <Data>
    <Description>Mountain</Description>
    <ID>2</ID>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
    <CurrentLocalTime>2012-04-13T07:47:12.503</CurrentLocalTime>
  </Data>
</Buildings>

```

GetChangedBookings

Description

Returns a list of bookings based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

StartDate	Start Date (DateChanged on tblBooking)	DateTime
EndDate	End Date (DateChanged on tblBooking)	DateTime
Rooms	List of Room IDs	Array Of Integers
Statuses	List of Status IDs	Array Of Integers
EventTypes	List of Event Type IDs	Array Of Integers
GroupTypes	List of Group Type IDs	Array Of Integers
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Booking Date based on reserved time	DateTime
StartBookingDate	Booking Date based on event time	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
BuildingID	Building ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
HVACZone	Room HVAC Zone	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String

Sample Code

```
Private Sub GetBookings()
    Dim dwvStatuses, dwvEventTypes, dwvGroupTypes As DataView
    Dim gintRooms As New Generic.List(Of Int32)
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintEventTypes As New Generic.List(Of Int32)
    Dim gintGroupTypes As New Generic.List(Of Int32)
```

```

Dim svc As New EMSAPI.ServiceSoapClient
Dim objReader As New StringReader(svc.GetStatuses("UserID", "Password"))

gintRooms.Add(3271)

dvwStatuses = Me.GetData(objReader)

objReader = New StringReader(svc.GetEventTypes("UserID", "Password"))
dvwEventTypes = Me.GetData(objReader)

objReader = New StringReader(svc.GetGroupTypes("UserID", "Password"))
dvwGroupTypes = Me.GetData(objReader)

For Each drv As DataRowView In dvwStatuses
    gintStatuses.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwEventTypes
    gintEventTypes.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwGroupTypes
    gintGroupTypes.Add(drv("ID"))
Next

Dim strRooms As New EMSAPI.ArrayOfInt
Dim strStatuses As New EMSAPI.ArrayOfInt
Dim strEventTypes As New EMSAPI.ArrayOfInt
Dim strGroupTypes As New EMSAPI.ArrayOfInt

strRooms.AddRange(gintRooms.ToArray)
strStatuses.AddRange(gintStatuses.ToArray)
strEventTypes.AddRange(gintEventTypes.ToArray)
strGroupTypes.AddRange(gintGroupTypes.ToArray)

objReader = New StringReader(svc.GetChangedBookings("UserID", "Password", Date.Today, Date.Today, strRooms,
strStatuses, strEventTypes, strGroupTypes, False))

Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>false</VIPEvent>
    <ClosedAllDay>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>

```



```

        <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
        <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
        <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
        <TimeZone>MT</TimeZone>
        <BuildingCode>DEN</BuildingCode>
        <Building>Denver</Building>
        <RoomCode>RM101</RoomCode>
        <Room>Room 101</Room>
        <RoomID>3271</RoomID>
        <BuildingID>1</BuildingID>
        <RoomTypeID>1</RoomTypeID>
        <RoomType>Classroom</RoomType>
        <HVACZone/>
        <StatusID>1</StatusID>
        <StatusTypeID>-14</StatusTypeID>
        <EventTypeID>4</EventTypeID>
        <GroupTypeID>3</GroupTypeID>
        <DateAdded>2009-10-04T16:00:00</DateAdded>
        <AddedBy>User 1</AddedBy>
        <DateChanged>2009-10-04T16:00:00</DateChanged>
        <ChangedBy>User 1</ChangedBy>
    </Data>
</Bookings>

```

GetCheckInStatus

Description

Returns a list of groups with their check in status

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
CheckInDate	Date	DateTime
Groups	Group List	Array Of Integers

Response Elements

Name	Description	Type
GroupName	Group Name	String
BuildingID	Building ID	Integer
Building	Building	String
CheckInStatus	Check In Status ("Not Checked In", "Checked In", "Checked Out")	String
CheckInDateTime	Check In Date/Time	DateTime
CheckInSource	Check In Source	String

Sample Code

```

Private Sub GetCheckInStatus()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim gintGroups As New Generic.List(Of Int32)

    gintGroups.Add(1234)
    gintGroups.Add(456)

    Dim strGroups As New EMSAPI.ArrayOfInt

    strGroups.AddRange(gintGroups.ToArray)

    Dim objReader As New System.IO.StringReader(svc.GetCheckInStatus("UserID", "Password", Date.Today, strGroups))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Groups>
  <Data>
    <GroupName>ABC Company</GroupName>
    <BuildingID>4</BuildingID>
    <Building>Building 2</Building>
    <CheckInStatus>Not Checked In</CheckInStatus>
    <CheckInDateTime/>
    <CheckInSource/>
  </Data>
  <Data>
    <GroupName>XYZ Company</GroupName>
    <BuildingID>4</BuildingID>
    <Building>Building 2</Building>
    <CheckInStatus>Checked In</CheckInStatus>
    <CheckInDateTime>2012-06-25T09:00:00</CheckInDateTime>
    <CheckInSource>Kiosk</CheckInSource>
  </Data>
</Groups>

```

GetComboRoomComponents

Description

Returns component room details for a specific combo room

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
RoomID	Room ID	Integer

Response Elements

Name	Description	Type
Room	Room Code	String
Description	Room Description	String
Building	Building	String
Floor	Floor Description	String
Type	Room Type	String
Classification	Room Classification	String
Phone	Room Phone	String
Notes	Room Notes	String
RoomSize	Room Size	Integer
SetupHours	Room Setup Hours	Decimal
TeardownHours	Room Teardown Hours	Decimal
HVACZone	Room HVAC Zone	String
URL	Room URL	String

Sample Code

```

Private Sub GetComboRoomComponents()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetComboRoomComponents("UserID", "Password", 3272))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<RoomComponents>
  <Data>
    <Room>RM 102</Room>
    <Description>Room 102</Description>
    <Building>Denver</Building>
    <Floor>1st Floor</Floor>
    <Type>Meeting Room</Type>
    <Classification>Standard</Classification>
  </Data>
</RoomComponents>

```

```

        <Phone>x456</Phone>
        <Notes />
        <RoomSize>10</RoomSize>
        <SetupHours>0</SetupHours>
        <TeardownHours>0</TeardownHours>
        <HVACZone />
        <URL />
    </Data>
</RoomComponents>

```

GetCommentTypes

Description

Returns a list of comment types

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
Description	Comment Type Description	String
ID	Comment Type ID	Integer
Comments	Default Comments	String

Sample Code

```

Private Sub GetCommentTypes()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetCommentTypes("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<CommentTypes>
  <Data>
    <Description>(user specified comment)</Description>
    <ID>-38</ID>
    <Comments/>
  </Data>
  <Data>
    <Description>Res. Comments</Description>
    <ID>2</ID>
    <Comments>test comments</Comments>
  </Data>
</CommentTypes>

```

GetContactDetails

Description

Returns details for a specific Contact

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ContactID	Contact ID	Integer

Response Elements

Name	Description	Type
Contact	Contact Name	String
GroupName	Group Name	String
Address1	Address Line 1	String
Address2	Address Line 2	String
City	City	String
State	State	String
ZipCode	Zip Code	String
Country	Country	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String
GroupID	Group ID	Integer

Sample Code

```
Private Sub GetContactDetails()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetContactDetails("UserID", "Password", 12345))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<ContactDetails>
  <Data>
    <Contact>Test Contact</Contact>
    <GroupName>Text Group</GroupName>
    <Address1>123 Main St.</Address1>
    <Address2/>
    <City>Denver</City>
    <State>CO</State>
    <ZipCode>80111</ZipCode>
    <Country />
    <Phone>x123</Phone>
    <Fax />
    <EmailAddress>testcontact@test.com</EmailAddress>
    <ExternalReference />
    <GroupID>1234</GroupID>
  </Data>
</ContactDetails>
```

GetContacts**Description**

Returns Contact list for a Group

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer

Response Elements

Name	Description	Type
ContactID	Contact ID	Integer
Contact	Contact Name	String
GroupName	Group Name	String
Address1	Address Line 1	String

Address2	Address Line 2	String
City	City	String
State	State	String
ZipCode	Zip Code	String
Country	Country	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String

Sample Code

```
Private Sub GetContacts()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetContacts("UserID", "Password", 1234))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Contacts>
  <Data>
    <ContactID>12345</ContactID>
    <Contact>Test Contact</Contact>
    <GroupName>Text Group</GroupName>
    <Address1>123 Main St.</Address1>
    <Address2/>
    <City>Denver</City>
    <State>CO</State>
    <ZipCode>80111</ZipCode>
    <Country />
    <Phone>x123</Phone>
    <Fax />
    <EmailAddress>testcontact@test.com</EmailAddress>
    <ExternalReference />
  </Data>
</Contacts>
```

GetCourseBookings

Description

Returns a list of bookings tied to courses based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
Buildings	List of Building IDs	Array Of Integers
Statuses	List of Status IDs	Array Of Integers
EventTypes	List of Event Type IDs	Array Of Integers
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
ReservationID	Reservation ID	Integer

GroupID	Group/Customer ID	Integer
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
DateAdded	Date/Time Booking Added	DateTime
TermCode	Term Code	String
CRN	Course CRN	String
SubjectCode	Subject Code	String
Course	Course	String
Section	Section	String
CourseTitle	Course Title	String
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```

Private Sub GetBookings()
    Dim dwvBuildings, dwvStatuses, dwvEventTypes As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintEventTypes As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dwvBuildings = Me.GetData(objReader)

    objReader = New StringReader(svc.GetStatuses("UserID", "Password"))
    dwvStatuses = Me.GetData(objReader)

    objReader = New StringReader(svc.GetEventTypes("UserID", "Password"))
    dwvEventTypes = Me.GetData(objReader)

    For Each drv As DataRowView In dwvBuildings
        gintBuildings.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dwvStatuses
        gintStatuses.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dwvEventTypes
        gintEventTypes.Add(drv("ID"))
    Next

    Dim strBuildings As New EMSAPI.ArrayOfInt

```

```

Dim strStatuses As New EMSAPI.ArrayOfInt
Dim strEventTypes As New EMSAPI.ArrayOfInt

strBuildings.AddRange(gintBuildings.ToArray)
strStatuses.AddRange(gintStatuses.ToArray)
strEventTypes.AddRange(gintEventTypes.ToArray)

objReader = New StringReader(svc.GetCourseBookings("UserID", "Password", Date.Today, Date.Today, strBuildings,
strStatuses, strEventTypes, False))

Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
End Sub

```

Example Response

```

<CourseBookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <ReservationID>6650</ReservationID>
    <GroupID>499</GroupID>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</Room>
    <RoomID>3271</RoomID>
    <BuildingID>1</BuildingID>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <TermCode>Fall 2012</TermCode>
    <CRN>10101</CRN>
    <SubjectCode>ACCT</SubjectCode>
    <Course>101</Course>
    <Section>01</Section>
    <CourseTitle>Accounting Course</CourseTitle>
    <DateAdded>2009-10-04T16:00:00</DateAdded>
    <AddedBy>User 1</AddedBy>
    <DateChanged>2009-10-04T16:00:00</DateChanged>
    <ChangedBy>User 1</ChangedBy>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</CourseBookings>

```

GetEventTypes

Description

Returns a list of event types

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
Description	Event Type Description	String
ID	Event Type ID	Integer
DisplayOnWeb	Display On Web Flag	Boolean

Sample Code

```
Private Sub GetEventTypes()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetEventTypes("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<EventTypes>
  <Data>
    <Description>(none)</Description>
    <ID>0</ID>
    <DisplayOnWeb>true</DisplayOnWeb>
  </Data>
  <Data>
    <Description>Meeting</Description>
    <ID>2</ID>
    <DisplayOnWeb>true</DisplayOnWeb>
  </Data>
  <Data>
    <Description>Inactive</Description>
    <ID>3</ID>
    <DisplayOnWeb>true</DisplayOnWeb>
  </Data>
</EventTypes>
```

GetFeatures**Description**

Returns a list of Features

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	Building ID (-1 = All Buildings)	Integer

Response Elements

Name	Description	Type
Description	Description	String
ID	ID	Integer

Sample Code

```
Private Sub GetFeatures()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetFeatures("UserID", "Password", 1))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```


End Sub

Example Response

```
<Features>
  <Data>
    <Description>Whiteboard</Description>
    <ID>1</ID>
  </Data>
  <Data>
    <Description>Projector</Description>
    <ID>2</ID>
  </Data>
</Features>
```

GetGroupBookings

Description

Returns a list of bookings for a group based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
GroupID	Group ID	Integer
Statuses	List of Status IDs	Array Of Integers
StatusTypes	List of Status Types	Array Of Integers
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
ReservationID	Reservation ID	Integer
BookingID	Booking ID	Integer
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GMTEventStart	Booking GMT Event Start Time	DateTime
GMTEventEnd	Booking GMT Event End Time	DateTime
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTBookingStart	Booking GMT Reserved Start Time	DateTime
GMTBookingEnd	Booking GMT Reserved End Time	DateTime
EventName	Event Name	String
BuildingID	Building ID	Integer
BuildingCode	Building Code	String
Building	Building Description	String
TimeZoneID	Time Zone ID	Integer
TimeZone	Time Zone abbreviation	String
RoomID	Room ID	Integer
RoomCode	Room Code	String
RoomDescription	Location Description	String
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
StatusDescription	Status Description	String
EventTypeID	Event Type ID	Integer
EventTypeDescription	Event Type Description	String

SetupTypeID	Setup Type ID	Integer
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
GroupID	Group/Customer ID	Integer
GroupName	Group/Customer name	String
GroupTypeID	Group Type ID	Integer
GroupTypeDescription	Group Type Description	String
Contact1ID	First Contact ID	Integer
Contact1Name	First Contact Name	String
Contact2ID	Second Contact ID	Integer
Contact2Name	Second Contact Name	String
WebProcessTemplateID	Web Process Template ID (0 is none)	Integer
ContactEmailAddress	Contact Email Address	String

Sample Code

```
Private Sub GetGroupBookings()
    Dim dwvStatuses As DataView
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintStatusTypes As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetStatuses("UserID", "Password"))

    dwvStatuses = Me.GetData(objReader)

    For Each drv As DataRowView In dwvStatuses
        gintStatuses.Add(drv("ID"))
    Next

    gintStatusTypes.Add(-14) 'Booked Space

    Dim strStatuses As New EMSAPI.ArrayOfInt
    Dim strStatusTypes as New EMSAPI.ArrayOfInt

    strStatuses.AddRange(gintStatuses.ToArray)
    strStatusTypes.AddRange(gintStatusTypes.ToArray)

    objReader = New StringReader(svc.GetGroupBookings("UserID", "Password", Date.Today, Date.Today, 499, strStatuses,
    strStatusTypes, False))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Bookings>
  <Data>
    <ReservationID>6650</ReservationID>
    <BookingID>222008</BookingID>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GMTEventStart>2009-10-04T22:00:00</GMTEventStart>
    <GMTEventEnd>2009-10-04T23:00:00</GMTEventEnd>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTBookingStart>2009-10-04T22:00:00</GMTBookingStart>
    <GMTBookingEnd>2009-10-04T23:00:00</GMTBookingEnd>
    <EventName>Spring Conference</EventName>
    <BuildingID>1</BuildingID>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <TimeZoneID>68</TimeZoneID>
    <TimeZone>MT</TimeZone>
    <RoomID>3271</RoomID>
    <RoomCode>RM101</RoomCode>
    <RoomDescription>Room 101</RoomDescription>
```

```

    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <StatusDescription>Confirmed</StatusDescription>
    <EventTypeID>4</EventTypeID>
    <EventTypeDescription>Conference</EventTypeDescription>
    <SetupTypeID>1</SetupTypeID>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <GroupID>499</GroupID>
    <GroupName>ABC Company </GroupName>
    <GroupTypeID>3</GroupTypeID>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <Contact1ID>53</Contact1ID>
    <Contact1Name>Jim Smith</Contact1Name>
    <Contact2ID>-1</Contact2ID>
    <Contact2Name/>
    <WebProcessTemplateID>2</WebProcessTemplateID>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetGroupDetails

Description

Returns details for a specific Group

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer

Response Elements

Name	Description	Type
GroupName	Group Name	String
GroupType	Group Type	String
Address1	Address Line 1	String
Address2	Address Line 2	String
City	City	String
State	State	String
ZipCode	Zip Code	String
Country	Country	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String
GroupTypeID	Group Type ID	Integer

Sample Code

```

Private Sub GetGroupDetails()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetGroupDetails("UserID", "Password", 1234))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```
<GroupDetails>
```

```

<Data>
  <GroupName>Test Group</GroupName>
  <GroupType>(none)</GroupType>
  <Address1>123 Main St.</Address1>
  <Address2/>
  <City>Denver</City>
  <State>CO</State>
  <ZipCode>80111</ZipCode>
  <Country />
  <Phone>x123</Phone>
  <Fax />
  <EmailAddress>testgroup@test.com</EmailAddress>
  <ExternalReference />
  <GroupTypeID>0</GroupTypeID>
</Data>
</GroupDetails>

```

GetGroups

Description

Returns a list of groups

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupName	Group Name Filter	String
EmailAddress	Email Address Filter	String

Response Elements

Name	Description	Type
GroupName	GroupName	String
EmailAddress	Email Address	String
ID	Group ID	Integer

Sample Code

```

Private Sub GetGroups()
  Dim svc As New EMSAPI.ServiceSoapClient
  Dim objReader As New System.IO.StringReader(svc.GetGroups("UserID", "Password", "XYZ Co", "xyz@xyz.com"))
  Dim dst As New DataSet("Data")
  dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Groups>
  <Data>
    <GroupName>XYZ Company</GroupName>
    <EmailAddress>xyz@xyz.com</EmailAddress>
    <ID>58</ID>
  </Data>
</Groups>

```

GetGroupTypes

Description

Returns a list of group types

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
Description	Group Type Description	String
ID	Group Type ID	Integer
AvailableOnWeb	Available On Web Flag	Boolean

Sample Code

```
Private Sub GetGroupTypes()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetGroupTypes("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<GroupTypes>
  <Data>
    <Description>(none)</Description>
    <ID>0</ID>
    <AvailableOnWeb>true</AvailableOnWeb>
  </Data>
  <Data>
    <Description>Big Wigs</Description>
    <ID>1</ID>
    <AvailableOnWeb>false</AvailableOnWeb>
  </Data>
  <Data>
    <Description>Managers</Description>
    <ID>2</ID>
    <AvailableOnWeb>true</AvailableOnWeb>
  </Data>
</GroupTypes>
```

GetHVACBookings**Description**

Returns a list of bookings based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
Buildings	List of Building IDs	Array Of Integers
Statuses	List of Status IDs	Array Of Integers
EventTypes	List of Event Type IDs	Array Of Integers
GroupTypes	List of Group Type IDs	Array Of Integers
HVACZones	List of HVAC Zones	Array Of Strings
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String

TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
BuildingID	Building ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
HVACZone	Room HVAC Zone	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
DateAdded	Date/Time Booking Added	DateTime
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```

Private Sub GetBookings()
    Dim dwvBuildings, dwvStatuses, dwvEventTypes, dwvGroupTypes As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintEventTypes As New Generic.List(Of Int32)
    Dim gintGroupTypes As New Generic.List(Of Int32)
    Dim gintHVACZones As New Generic.List(Of String)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dwvBuildings = Me.GetData(objReader)

    objReader = New StringReader(svc.GetStatuses("UserID", "Password"))
    dwvStatuses = Me.GetData(objReader)

    objReader = New StringReader(svc.GetEventTypes("UserID", "Password"))
    dwvEventTypes = Me.GetData(objReader)

    objReader = New StringReader(svc.GetGroupTypes("UserID", "Password"))
    dwvGroupTypes = Me.GetData(objReader)

```

```

For Each drv As DataRowView In dvwBuildings
    gintBuildings.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwStatuses
    gintStatuses.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwEventTypes
    gintEventTypes.Add(drv("ID"))
Next

For Each drv As DataRowView In dvwGroupTypes
    gintGroupTypes.Add(drv("ID"))
Next

gintHVACZones.Add(1)

Dim strBuildings As New EMSAPI.ArrayOfInt
Dim strStatuses As New EMSAPI.ArrayOfInt
Dim strEventTypes As New EMSAPI.ArrayOfInt
Dim strGroupTypes As New EMSAPI.ArrayOfInt
Dim strHVACZones As New EMSAPI.ArrayOfString

strBuildings.AddRange(gintBuildings.ToArray)
strStatuses.AddRange(gintStatuses.ToArray)
strEventTypes.AddRange(gintEventTypes.ToArray)
strGroupTypes.AddRange(gintGroupTypes.ToArray)
strHVACZones.AddRange(gintHVACZones.ToArray)

objReader = New StringReader(svc.GetBookings("UserID", "Password", Date.Today, Date.Today, strBuildings,
strStatuses, strEventTypes, strGroupTypes, strHVACZones, False))

Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
    <EventCoordinator />
    <GroupID>499</GroupID>
    <VIP xml:space="preserve">
    </VIP>
    <VIPEvent>>false</VIPEvent>
    <ClosedAllDay>>false</ClosedAllDay>
    <OpenTime>1900-01-01T00:00:00</OpenTime>
    <CloseTime>1900-01-01T00:00:00</CloseTime>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <EventTypeDescription>Conference</EventTypeDescription>
    <Contact>Jim Smith</Contact>
    <AltContact />
    <BookingID>222008</BookingID>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
    <GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
    <TimeZone>MT</TimeZone>
  </Data>
</Bookings>

```

```

    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <RoomCode>RM101</RoomCode>
    <Room>Room 101</RoomCode>
    <RoomID>3271</RoomID>
    <BuildingID>1</BuildingID>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <HVACZone>1</HVACZone>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <EventTypeID>4</EventTypeID>
    <GroupTypeID>3</GroupTypeID>
    <DateAdded>2009-10-04T16:00:00</DateAdded>
    <AddedBy>User 1</AddedBy>
    <DateChanged>2009-10-04T16:00:00</DateChanged>
    <ChangedBy>User 1</ChangedBy>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetReservationComments

Description

Returns Comments list for a Reservation

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ReservationID	Reservation ID	Integer

Response Elements

Name	Description	Type
CommentID	Comment ID	Integer
CommentTypeID	Comment Type ID	Integer
Comments	Comments	String

Sample Code

```

Private Sub GetReservationComments()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetReservationComments("UserID", "Password", 1234))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<ReservationComments>
  <Data>
    <CommentID>1</CommentID>
    <CommentTypeID>2</CommentTypeID>
    <Comments>test Reservation comments</Comments>
  </Data>
</ReservationComments>

```

GetReservationSources

Description

Returns a list of Reservation Sources

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
ReservationSourceID	Reservation Source ID	Integer
ReservationSourceDescription	Reservation Source Description	String

Sample Code

```
Private Sub GetReservationSources()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetReservationSources("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<ReservationSources>
  <Data>
    <ReservationSourceID>1</ReservationSourceID>
    <ReservationSourceDescription>Phone</ReservationSourceDescription>
  </Data>
  <Data>
    <ReservationSourceID>2</ReservationSourceID>
    <ReservationSourceDescription>Email</ReservationSourceDescription>
  </Data>
</ReservationSources>
```

GetRoomAvailability**Description**

Returns whether a room is available or not

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
RoomID	Room ID	Integer
BookingDate	Booking Date to check availability on	DateTime
StartTime	Start Time to check availability	DateTime
EndTime	EndTime to check availability	DateTime

Response Elements

Name	Description	Type
Available	Room Available	Boolean

Sample Code

```
Private Sub GetRoomsAvailable()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetRoomsAvailable("UserID", "Password", 3199, Date.Today,
    CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM")))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<RoomAvailability>
  <Data>
```

```

    <Available>true</Available>
  </Data>
</RoomAvailability>

```

GetRoomBookings

Description

Returns a list of bookings for a specific room based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
RoomID	Room ID	Integer
Statuses	List of Status IDs	Array Of Integers
EventTypes	List of Event Type IDs	Array Of Integers
GroupTypes	List of Group Type IDs	Array Of Integers
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
RoomDescription	Location Description	String
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GroupName	Group/Customer name	String
EventName	Event Name	String
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
ReservationID	Reservation ID	Integer
EventCoordinator	Initials of Event Coordinator	String
GroupID	Group/Customer ID	Integer
VIP	'VIP' or empty string to signify a VIP event	String
VIPEvent	0 or 1 to signify a VIP event	Boolean
ClosedAllDay	0 or 1 to signify if the Building is closed	Boolean
OpenTime	Time Building Opens	DateTime
CloseTime	Time Building Closes	DateTime
GroupTypeDescription	Group Type Description	String
EventTypeDescription	Event Type Description	String
Contact	Contact Name	String
AltContact	Alternate Contact Name	String
BookingID	Booking ID	Integer
BuildingID	Building ID	Integer
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTStartTime	Booking Reserved GMT Start Time	DateTime
GMTEndTime	Booking Reserved GMT End Time	DateTime
TimeZone	Time Zone abbreviation	String
BuildingCode	Building Code	String
Building	Building Description	String
RoomCode	Room Code	String
Room	Room	String
RoomID	Room ID	Integer
BuildingID	Building ID	Integer
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer

EventTypeID	Event Type ID	Integer
GroupTypeID	Group Type ID	Integer
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
DateAdded	Date/Time Booking Added	Date/Time
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	Date/Time
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String

Sample Code

```
Private Sub GetBookings()
    Dim dvwStatuses, dvwEventTypes, dvwGroupTypes As DataView
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintEventTypes As New Generic.List(Of Int32)
    Dim gintGroupTypes As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetStatuses("UserID", "Password"))

    dvwStatuses = Me.GetData(objReader)

    objReader = New StringReader(svc.GetEventTypes("UserID", "Password"))
    dvwEventTypes = Me.GetData(objReader)

    objReader = New StringReader(svc.GetGroupTypes("UserID", "Password"))
    dvwGroupTypes = Me.GetData(objReader)

    For Each drv As DataRowView In dvwStatuses
        gintStatuses.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dvwEventTypes
        gintEventTypes.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dvwGroupTypes
        gintGroupTypes.Add(drv("ID"))
    Next

    Dim strStatuses As New EMSAPI.ArrayOfInt
    Dim strEventTypes As New EMSAPI.ArrayOfInt
    Dim strGroupTypes As New EMSAPI.ArrayOfInt

    strStatuses.AddRange(gintStatuses.ToArray)
    strEventTypes.AddRange(gintEventTypes.ToArray)
    strGroupTypes.AddRange(gintGroupTypes.ToArray)

    objReader = New StringReader(svc.GetRoomBookings("UserID", "Password", Date.Today, Date.Today, 3271,
        strStatuses, strEventTypes, strGroupTypes, False))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Bookings>
  <Data>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <RoomDescription>DEN Room 101</RoomDescription>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GroupName>ABC Company </GroupName>
    <EventName>Spring Conference</EventName>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <ReservationID>6650</ReservationID>
```

```

<EventCoordinator />
<GroupID>499</GroupID>
<VIP xml:space="preserve">
</VIP>
<VIPEvent>false</VIPEvent>
<ClosedAllDay>false</ClosedAllDay>
<OpenTime>1900-01-01T00:00:00</OpenTime>
<CloseTime>1900-01-01T00:00:00</CloseTime>
<GroupTypeDescription>Corporate</GroupTypeDescription>
<EventTypeDescription>Conference</EventTypeDescription>
<Contact>Jim Smith</Contact>
<AltContact />
<BookingID>222008</BookingID>
<TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
<TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
<GMTStartTime>2009-10-04T22:00:00</GMTStartTime>
<GMTEndTime>2009-10-04T23:00:00</GMTEndTime>
<TimeZone>MT</TimeZone>
<BuildingCode>DEN</BuildingCode>
<Building>Denver</Building>
<RoomCode>RM101</RoomCode>
<Room>Room 101</RoomCode>
<RoomID>3271</RoomID>
<BuildingID>1</BuildingID>
<StatusID>1</StatusID>
<StatusTypeID>-14</StatusTypeID>
<EventTypeID>4</EventTypeID>
<GroupTypeID>3</GroupTypeID>
<RoomTypeID>1</RoomTypeID>
<RoomType>Classroom</RoomType>
<DateAdded>2009-10-04T16:00:00</DateAdded>
<AddedBy>User 1</AddedBy>
<DateChanged>2009-10-04T16:00:00</DateChanged>
<ChangedBy>User 1</ChangedBy>
<ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
</Data>
</Bookings>

```

GetRoomDetails

Description

Returns details for a specific room

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
RoomID	Room ID	Integer

Response Elements

Name	Description	Type
Room	Room Code	String
Description	Room Description	String
Building	Building	String
Floor	Floor Description	String
Type	Room Type	String
Classification	Room Classification	String
Phone	Room Phone	String
Notes	Room Notes	String
RoomSize	Room Size	Integer
SetupHours	Room Setup Hours	Decimal

TeardownHours	Room Teardown Hours	Decimal
HVACZone	Room HVAC Zone	String
URL	Room URL	String
DisplayOnWeb	Display On Web	Boolean
FloorSequence	Floor Sequence	Integer

Sample Code

```
Private Sub GetRoomDetails()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetRoomDetails("UserID", "Password", 3271))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<RoomDetails>
  <Data>
    <Room>RM 101</Room>
    <Description>Room 101</Description>
    <Building>Denver</Building>
    <Floor>1st Floor</Floor>
    <Type>Meeting Room</Type>
    <Classification>Standard</Classification>
    <Phone>x123</Phone>
    <Notes />
    <RoomSize>20</RoomSize>
    <SetupHours>0</SetupHours>
    <TeardownHours>0</TeardownHours>
    <HVACZone />
    <URL />
    <DisplayOnWeb>1</DisplayOnWeb>
    <FloorSequence>1</FloorSequence>
  </Data>
</RoomDetails>
```

GetRooms

Description

Returns list of rooms for multiple buildings

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
Buildings	List of Building IDs	Array Of Integers

Response Elements

Name	Description	Type
Room	Room Code	String
Description	Room Description	String
Building	Building	String
Floor	Floor Description	String
Type	Room Type	String
Classification	Room Classification	String
Phone	Room Phone	String
Notes	Room Notes	String
RoomSize	Room Size	Integer
SetupHours	Room Setup Hours	Decimal
TeardownHours	Room Teardown Hours	Decimal
HVACZone	Room HVAC Zone	String
URL	Room URL	String
Active	Room Active	Boolean

BuildingID	Building ID	Integer
ID	Room ID	Integer
ExternalReference	Room External Reference	String
DisplayOnWeb	Room Display On Web	Boolean

Sample Code

```
Private Sub GetRooms()
    Dim dvwBuildings As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dvwBuildings = Me.GetData(objReader)

    For Each drv As DataRowView In dvwBuildings
        gintBuildings.Add(drv("ID"))
    Next

    Dim strBuildings As New EMSAPI.ArrayOfInt

    strBuildings.AddRange(gintBuildings.ToArray)

    objReader = New System.IO.StringReader(svc.GetRooms("UserID", "Password", strBuildings))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Rooms>
  <Data>
    <Room>RM 101</Room>
    <Description>Room 101</Description>
    <Building>Denver</Building>
    <Floor>1st Floor</Floor>
    <Type>Meeting Room</Type>
    <Classification>Standard</Type>
    <Phone>x123</Phone>
    <Notes />
    <RoomSize>20</RoomSize>
    <SetupHours>0</SetupHours>
    <TeardownHours>0</TeardownHours>
    <HVACZone />
    <URL />
    <Active>1</Active>
    <BuildingID>1</BuildingID>
    <ID>3271</ID>
    <ExternalReference />
    <DisplayOnWeb>1</DisplayOnWeb>
  </Data>
</Rooms>
```

GetRoomsAvailable

Description

Returns a list of available rooms

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	Building ID (-1 = All)	Integer
BookingDate	Booking Date to check availability on	DateTime
StartTime	Start Time to check availability	DateTime
EndTime	EndTime to check availability	DateTime

Response Elements

Name	Description	Type
Room	Room Code	String
Description	Room Description	String
Building	Building	String
RoomID	Room ID	Integer

Sample Code

```
Private Sub GetRoomAvailability()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetRoomAvailability("UserID", "Password", 1, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM")))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<RoomAvailability>
  <Data>
    <Room>RM 101</Room>
    <Description>Room 101</Description>
    <Building>Denver</Building>
    <RoomID>3271</RoomID>
  </Data>
  <Data>
    <Room>RM 102</Room>
    <Description>Room 102</Description>
    <Building>Denver</Building>
    <RoomID>3272</RoomID>
  </Data>
</RoomAvailability>
```

GetRoomsAvailable2**Description**

Returns a list of available rooms

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	Building ID (-1 = All)	Integer
RoomTypeID	Room Type ID (-1 = All)	Integer
FloorID	Floor ID (-1 = All)	Integer
WebProcessTemplates	List of Web Process Template IDs (-1 = All)	Array Of Integers
StartDateTime	Start Date/Time to check availability	DateTime
EndDateTime	End Date/Time to check availability	DateTime

Response Elements

Name	Description	Type
RoomID	Room ID	Integer
RoomCode	Room Code	String
RoomDescription	Room Description	String
BuildingID	Building ID	Integer
BuildingDescription	Building Description	String
TimeZoneID	Time Zone ID	Integer
RoomTypeID	Room Type ID	Integer
RoomTypeDescription	Room Type Description	String
TimeZoneDescription	Time Zone Description	String
TimeZoneAbbreviation	Time Zone Abbreviation	String
RecordTypeID	Record Type: 1=Reserve, 2=Request, 3=Both	Integer
FloorID	Floor ID	Integer

FloorDescription	Floor Description	String
FloorSequence	Floor Sequence	Integer
DisplayOnWeb	Display On Web	Boolean
RoomSetupHours	Room Setup Hours	Decimal
RoomTeardownHours	Room TeardownHours	Decimal
SetupTypeID	Room Default Setup Type ID	Integer
SetupTypeDescription	Default Setup Type Description	String
MinimumCapacity	Minimum Capacity for Default Setup Type	Integer
MaximumCapacity	Maximum Capacity for Default Setup Type	Integer

Sample Code

```
Private Sub GetRoomAvailability2()
    Dim dvwTemplates As DataView
    Dim gintTemplates As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetWebTemplates("UserID", "Password"))

    dvwTemplates= Me.GetData(objReader)

    For Each drv As DataRowView In dvwTemplates
        gintTemplates.Add(drv("ID"))
    Next

    Dim strTemplates As New EMSAPI.ArrayOfInt

    strTemplates.AddRange(gintTemplates.ToArray)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetRoomAvailability2("UserID", "Password", 1, -1, -1, strTemplates,
CDate("9/1/2014 8:00 AM"), CDate("9/1/2014 9:00 AM"))))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<RoomAvailability>
  <Data>
    <RoomID>3271</RoomID>
    <RoomCode>RM 101</RoomCode>
    <RoomDescription>Room 101</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <TimeZonelD>68</TimeZonelD>
    <RoomTypeID>1</RoomTypeID>
    <RoomTypeDescription>Meeting</RoomTypeDescription>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
    <RecordTypeID>1</RecordTypeID>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <FloorSequence>0</FloorSequence>
    <DisplayOnWeb>true</DisplayOnWeb>
    <RoomSetupHours>1.5</RoomSetupHours>
    <RoomTeardownHours>1.5</RoomTeardownHours>
    <SetupTypeID>0</SetupTypeID>
    <SetupTypeDescription>(none)</ SetupTypeDescription>
    <MinimumCapacity>0</MinimumCapacity>
    <MaximumCapacity>0</MaximumCapacity>
  </Data>
  <Data>
    <RoomID>3272</RoomID>
    <RoomCode>RM 102</RoomCode>
    <RoomDescription>Room 102</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <TimeZonelD>68</TimeZonelD>
    <RoomTypeID>1</RoomTypeID>
```



```

    <RoomTypeDescription>Meeting</RoomTypeDescription>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
    <RecordTypeID>3</RecordTypeID>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <FloorSequence>0</FloorSequence>
    <DisplayOnWeb>true</DisplayOnWeb>
    <RoomSetupHours>1.5</RoomSetupHours>
    <RoomTeardownHours>1.5</RoomTeardownHours>
    <SetupTypeID>0</SetupTypeID>
    <SetupTypeDescription>(none)</ SetupTypeDescription>
    <MinimumCapacity>0</MinimumCapacity>
    <MaximumCapacity>0</MaximumCapacity>
  </Data>
</RoomAvailability>

```

GetRoomsAvailable3

Description

Returns a list of available rooms

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	Building ID (-1 = All)	Integer
RoomTypeID	Room Type ID (-1 = All)	Integer
FloorID	Floor ID (-1 = All)	Integer
WebProcessTemplates	List of Web Process Template IDs (-1 = All)	Array Of Integers
Features	List of Feature IDs	Array Of Integers
StartDateTime	Start Date/Time to check availability	DateTime
EndDateTime	End Date/Time to check availability	DateTime

Response Elements

Name	Description	Type
RoomID	Room ID	Integer
RoomCode	Room Code	String
RoomDescription	Room Description	String
BuildingID	Building ID	Integer
BuildingDescription	Building Description	String
TimeZoneID	Time Zone ID	Integer
RoomTypeID	Room Type ID	Integer
RoomTypeDescription	Room Type Description	String
TimeZoneDescription	Time Zone Description	String
TimeZoneAbbreviation	Time Zone Abbreviation	String
RecordTypeID	Record Type: 1=Reserve, 2=Request, 3=Both	Integer
FloorID	Floor ID	Integer
FloorDescription	Floor Description	String
FloorSequence	Floor Sequence	Integer
DisplayOnWeb	Display On Web	Boolean
RoomSetupHours	Room Setup Hours	Decimal
RoomTeardownHours	Room TeardownHours	Decimal
SetupTypeID	Room Default Setup Type ID	Integer
SetupTypeDescription	Default Setup Type Description	String
MinimumCapacity	Minimum Capacity for Default Setup Type	Integer
MaximumCapacity	Maximum Capacity for Default Setup Type	Integer

Sample Code

```

Private Sub GetRoomAvailability3()
    Dim dwvTemplates As DataView
    Dim dwvFeatures As DataView
    Dim gintTemplates As New Generic.List(Of Int32)
    Dim gintFeatures As New Generic.List(Of Int32)

```

```

Dim svc As New EMSAPI.ServiceSoapClient
Dim objReader As New StringReader(svc.GetWebTemplates("UserID", "Password"))

dvwTemplates= Me.GetData(objReader)

For Each drv As DataRowView In dvwTemplates
    gintTemplates.Add(drv("ID"))
Next

Dim strTemplates As New EMSAPI.ArrayOfInt
strTemplates.AddRange(gintTemplates.ToArray)

objReader = New StringReader(svc.GetFeatures("UserID", "Password", 1))

dvwFeatures = Me.GetData(objReader)

For Each drv As DataRowView In dvwFeatures
    gintFeatures.Add(drv("ID"))
Next

Dim strFeatures As New EMSAPI.ArrayOfInt
strFeatures.AddRange(gintFeatures.ToArray)

Dim svc As New EMSAPI.ServiceSoapClient
Dim objReader As New System.IO.StringReader(svc.GetRoomAvailability3("UserID", "Password", 1, -1, -1, strTemplates,
strFeatures, CDate("9/1/2014 8:00 AM"), CDate("9/1/2014 9:00 AM")))
Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
End Sub

```

Example Response

```

<RoomAvailability>
  <Data>
    <RoomID>3271</RoomID>
    <RoomCode>RM 101</RoomCode>
    <RoomDescription>Room 101</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <TimeZoneID>68</TimeZoneID>
    <RoomTypeID>1</RoomTypeID>
    <RoomTypeDescription>Meeting</RoomTypeDescription>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
    <RecordTypeID>1</RecordTypeID>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <FloorSequence>0</FloorSequence>
    <DisplayOnWeb>true</DisplayOnWeb>
    <RoomSetupHours>1.5</RoomSetupHours>
    <RoomTeardownHours>1.5</RoomTeardownHours>
    <SetupTypeID>0</SetupTypeID>
    <SetupTypeDescription>(none)</SetupTypeDescription>
    <MinimumCapacity>0</MinimumCapacity>
    <MaximumCapacity>0</MaximumCapacity>
  </Data>
  <Data>
    <RoomID>3272</RoomID>
    <RoomCode>RM 102</RoomCode>
    <RoomDescription>Room 102</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <TimeZoneID>68</TimeZoneID>
    <RoomTypeID>1</RoomTypeID>
    <RoomTypeDescription>Meeting</RoomTypeDescription>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
  </Data>
</RoomAvailability>

```

```

    <RecordTypeID>3</RecordTypeID>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <FloorSequence>0</FloorSequence>
    <DisplayOnWeb>true</DisplayOnWeb>
    <RoomSetupHours>1.5</RoomSetupHours>
    <RoomTeardownHours>1.5</RoomTeardownHours>
    <SetupTypeID>0</SetupTypeID>
    <SetupTypeDescription>(none)</SetupTypeDescription>
    <MinimumCapacity>0</MinimumCapacity>
    <MaximumCapacity>0</MaximumCapacity>
  </Data>
</RoomAvailability>

```

GetRoomsAvailable4

Description

Returns a list of available rooms based on requested setup and capacity

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	Building ID (-1 = All)	Integer
RoomTypeID	Room Type ID (-1 = All)	Integer
FloorID	Floor ID (-1 = All)	Integer
WebProcessTemplates	List of Web Process Template IDs (-1 = All)	Array Of Integers
SetupTypeID	Setup Type ID (-1 = Use Room Default Setup)	Integer
RequestedCapacity	Requested Capacity for Setup Type	Integer
StartDateTime	Start Date/Time to check availability	DateTime
EndDateTime	End Date/Time to check availability	DateTime

Response Elements

Name	Description	Type
RoomID	Room ID	Integer
RoomCode	Room Code	String
RoomDescription	Room Description	String
BuildingID	Building ID	Integer
BuildingDescription	Building Description	String
TimeZoneID	Time Zone ID	Integer
RoomTypeID	Room Type ID	Integer
RoomTypeDescription	Room Type Description	String
TimeZoneDescription	Time Zone Description	String
TimeZoneAbbreviation	Time Zone Abbreviation	String
RecordTypeID	Record Type: 1=Reserve, 2=Request, 3=Both	Integer
FloorID	Floor ID	Integer
FloorDescription	Floor Description	String
FloorSequence	Floor Sequence	Integer
DisplayOnWeb	Display On Web	Boolean
RoomSetupHours	Room Setup Hours	Decimal
RoomTeardownHours	Room TeardownHours	Decimal
SetupTypeID	Setup Type ID specified or Room Default Setup Type ID	Integer
SetupTypeDescription	Setup Type Description	String
MinimumCapacity	Minimum Capacity for Setup Type	Integer
MaximumCapacity	Maximum Capacity for Setup Type	Integer

Sample Code

```

Private Sub GetRoomAvailability4()
    Dim dwvTemplates As DataView
    Dim gintTemplates As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetWebTemplates("UserID", "Password"))

```

```

dvwTemplates= Me.GetData(objReader)

For Each drv As DataRowView In dvwTemplates
    gintTemplates.Add(drv("ID"))
Next

Dim strTemplates As New EMSAPI.ArrayOfInt

strTemplates.AddRange(gintTemplates.ToArray)

Dim svc As New EMSAPI.ServiceSoapClient
Dim objReader As New System.IO.StringReader(svc.GetRoomAvailability4("UserID", "Password", 1, -1, -1, strTemplates,
1, 10, CDate("9/1/2014 8:00 AM"), CDate("9/1/2014 9:00 AM")))
Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
End Sub

```

Example Response

```

<RoomAvailability>
  <Data>
    <RoomID>3271</RoomID>
    <RoomCode>RM 101</RoomCode>
    <RoomDescription>Room 101</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <TimeZoneID>68</TimeZoneID>
    <RoomTypeID>1</RoomTypeID>
    <RoomTypeDescription>Meeting</RoomTypeDescription>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
    <RecordTypeID>1</RecordTypeID>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <FloorSequence>0</FloorSequence>
    <DisplayOnWeb>true</DisplayOnWeb>
    <RoomSetupHours>1.5</RoomSetupHours>
    <RoomTeardownHours>1.5</RoomTeardownHours>
    <SetupTypeID>1</SetupTypeID>
    <SetupTypeDescription>Circle</SetupTypeDescription>
    <MinimumCapacity>0</MinimumCapacity>
    <MaximumCapacity>20</MaximumCapacity>
  </Data>
  <Data>
    <RoomID>3272</RoomID>
    <RoomCode>RM 102</RoomCode>
    <RoomDescription>Room 102</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <TimeZoneID>68</TimeZoneID>
    <RoomTypeID>1</RoomTypeID>
    <RoomTypeDescription>Meeting</RoomTypeDescription>
    <TimeZoneDescription>Mountain Time</TimeZoneDescription>
    <TimeZoneAbbreviation>MT</TimeZoneAbbreviation>
    <RecordTypeID>3</RecordTypeID>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <FloorSequence>0</FloorSequence>
    <DisplayOnWeb>true</DisplayOnWeb>
    <RoomSetupHours>1.5</RoomSetupHours>
    <RoomTeardownHours>1.5</RoomTeardownHours>
    <SetupTypeID>1</SetupTypeID>
    <SetupTypeDescription>Circle</SetupTypeDescription>
    <MinimumCapacity>5</MinimumCapacity>
    <MaximumCapacity>30</MaximumCapacity>
  </Data>
</RoomAvailability>

```

GetRoomsBySetupType

Description

Returns a list of rooms by setup type and capacity

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	Building ID (-1 = All Buildings)	Integer
RoomTypeID	Room Type ID (-1 = All Room Types)	Integer
FloorID	Floor ID (-1 = All Floors)	Integer
SetupTypeID	Setup Type ID	Integer

Response Elements

Name	Description	Type
RoomID	Room ID	Integer
RoomCode	Room Code	String
RoomDescription	Room Description	String
BuildingID	Building ID	Integer
BuildingDescription	Building Description	String
RoomTypeID	Room Type ID	Integer
RoomTypeDescription	Room Type Description	String
FloorID	Floor ID	Integer
FloorDescription	Floor Description	String
SetupCount	Setup Count	Integer

Sample Code

```
Private Sub GetRoomsBySetupType()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetRoomsBySetupType("UserID", "Password", 1, -1, -1, 10))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Rooms>
  <Data>
    <RoomID>3271</RoomID>
    <RoomCode>RM 101</RoomCode>
    <RoomDescription>Room 101</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <RoomTypeID>12</RoomTypeID>
    <RoomTypeDescription>Meeting Room</RoomTypeDescription>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <SetupCount>20</SetupCount>
  </Data>
  <Data>
    <RoomID>3272</RoomID>
    <RoomCode>RM 102</RoomCode>
    <RoomDescription>Room 102</RoomDescription>
    <BuildingID>1</BuildingID>
    <BuildingDescription>Denver</BuildingDescription>
    <RoomTypeID>12</RoomTypeID>
    <RoomTypeDescription>Meeting Room</RoomTypeDescription>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <SetupCount>20</SetupCount>
  </Data>
</Rooms>
```

GetRoomTypesByWPT

Description

Returns list of rooms for multiple web process templates and multiple buildings

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebProcessTemplates	List of Web Process Template IDs (-1 = All)	Array Of Integers
Buildings	List of Building IDs (-1 = All)	Array Of Integers

Response Elements

Name	Description	Type
Type	Room Type Description	String
RoomTypeID	Room Type ID	Integer
FloorID	Floor ID	Integer
FloorDescription	Floor Description	String
BuildingID	Building ID	Integer

Sample Code

```
Private Sub GetRoomTypesByWPT()
    Dim dvwBuildings As DataView
    Dim dvwTemplates As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim gintTemplates As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dvwBuildings = Me.GetData(objReader)

    objReader = New StringReader(svc.GetWebTemplates("UserID", "Password"))
    dvwTemplates = Me.GetData(objReader)

    For Each drv As DataRowView In dvwBuildings
        gintBuildings.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dvwTemplates
        gintTemplates.Add(drv("ID"))
    Next

    Dim strBuildings As New EMSAPI.ArrayOfInt
    Dim strTemplates As New EMSAPI.ArrayOfInt

    strBuildings.AddRange(gintBuildings.ToArray)
    strTemplates.AddRange(gintTemplates.ToArray)

    objReader = New System.IO.StringReader(svc.GetRoomTypesByWPT("UserID", "Password", strTemplates, strBuildings))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<RoomTypes>
  <Data>
    <Type>Meeting</Type>
    <RoomTypeID>1</RoomTypeID>
    <FloorID>1</FloorID>
    <FloorDescription>1st Floor</FloorDescription>
    <BuildingID>1</BuildingID>
```

```

</Data>
<Data>
  <Type>Conference</Type>
  <RoomTypeID>2</RoomTypeID>
  <FloorID>1</FloorID>
  <FloorDescription>1st Floor</FloorDescription>
  <BuildingID>1</BuildingID>
</Data>
</RoomTypes>

```

GetServiceOrderDetails

Description

Returns a list of service order details based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
Buildings	List of Building IDs	Array Of Integers
Statuses	List of Status IDs	Array Of Integers

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
ServiceOrderStartTime	Service Order Start Time	DateTime
ServiceOrderEndTime	Service Order End Time	DateTime
ResourceDescription	Resource Description	String
ResourceExternalReference	Resource External Reference	String
ResourceQuantity	Resource Quantity In Use	Decimal
ServiceOrderID	Service Order ID	Integer
BookingID	Booking ID	Integer

Sample Code

```

Private Sub GetServiceOrderDetails()
    Dim dwvBuildings, dwvStatuses As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim gintStatuses As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dwvBuildings = Me.GetData(objReader)

    objReader = New StringReader(svc.GetStatuses("UserID", "Password"))
    dwvStatuses = Me.GetData(objReader)

    For Each drv As DataRowView In dwvBuildings
        gintBuildings.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dwvStatuses
        gintStatuses.Add(drv("ID"))
    Next

    Dim strBuildings As New EMSAPI.ArrayOfInt
    Dim strStatuses As New EMSAPI.ArrayOfInt

    strBuildings.AddRange(gintBuildings.ToArray)

```

```

strStatuses.AddRange(gintStatuses.ToArray)

objReader = New StringReader(svc.GetServiceOrderDetails("UserID", "Password", Date.Today, Date.Today, strBuildings,
strStatuses))

Dim dst As New DataSet("Data")
dst.ReadXml(objReader)
End Sub

```

Example Response

```

<ServiceOrderDetails>
  <Data>
    <BookingDate>2012-07-05T00:00:00</BookingDate>
    <ServiceOrderStartTime>1900-01-01T08:00:00</ServiceOrderStartTime>
    <ServiceOrderEndTime>1900-01-01T09:00:00</ServiceOrderEndTime>
    <ResourceDescription>Projector</ ResourceDescription >
    <ResourceExternalReference>ABC123</ ResourceExternalReference >
    <ResourceQuantity>1</ResourceQuantity>
    <ServiceOrderID>2</ServiceOrderID>
    <BookingID>1</BookingID>
  </Data>
  <Data>
    <BookingDate>2012-07-05T00:00:00</BookingDate>
    <ServiceOrderStartTime>1900-01-01T08:00:00</ServiceOrderStartTime>
    <ServiceOrderEndTime>1900-01-01T09:00:00</ServiceOrderEndTime>
    <ResourceDescription>Whiteboard</ ResourceDescription >
    <ResourceExternalReference>XYZ456</ ResourceExternalReference >
    <ResourceQuantity>1</ResourceQuantity>
    <ServiceOrderID>2</ServiceOrderID>
    <BookingID>1</BookingID>
  </Data>
</ServiceOrderDetails>

```

GetServiceOrderDetails2

Description

Returns a list of service order details with production item information based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
Buildings	List of Building IDs	Array Of Integers
Statuses	List of Status IDs	Array Of Integers

Response Elements

Name	Description	Type
BookingDate	Date that booking is reserved to start on	DateTime
ServiceOrderStartTime	Service Order Start Time	DateTime
ServiceOrderEndTime	Service Order End Time	DateTime
ResourceDescription	Resource Description	String
ResourceExternalReference	Resource External Reference	String
ResourceQuantity	Resource Quantity In Use	Decimal
ResourceCategoryName	Category Description	String
ProductionItemQuantity	Production Item Quantity	Decimal
ProductionItemExternalReference	Production Item External Reference	String
ProductionItemDescription	Production Item Description	String
ServiceOrderID	Service Order ID	Integer
BookingID	Booking ID	Integer

Sample Code


```

Private Sub GetServiceOrderDetails2()
    Dim dwvBuildings, dwvStatuses As DataView
    Dim gintBuildings As New Generic.List(Of Int32)
    Dim gintStatuses As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetBuildings("UserID", "Password"))

    dwvBuildings = Me.GetData(objReader)

    objReader = New StringReader(svc.GetStatuses("UserID", "Password"))
    dwvStatuses = Me.GetData(objReader)

    For Each drv As DataRowView In dwvBuildings
        gintBuildings.Add(drv("ID"))
    Next

    For Each drv As DataRowView In dwvStatuses
        gintStatuses.Add(drv("ID"))
    Next

    Dim strBuildings As New EMSAPI.ArrayOfInt
    Dim strStatuses As New EMSAPI.ArrayOfInt

    strBuildings.AddRange(gintBuildings.ToArray)
    strStatuses.AddRange(gintStatuses.ToArray)

    objReader = New StringReader(svc.GetServiceOrderDetails2("UserID", "Password", Date.Today, Date.Today,
    strBuildings, strStatuses))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<ServiceOrderDetails>
  <Data>
    <BookingDate>2012-07-05T00:00:00</BookingDate>
    <ServiceOrderStartTime>1900-01-01T08:00:00</ServiceOrderStartTime>
    <ServiceOrderEndTime>1900-01-01T09:00:00</ServiceOrderEndTime>
    <ResourceDescription>Continental Breakfast</ ResourceDescription >
    <ResourceExternalReference>ABC123</ ResourceExternalReference >
    <ResourceQuantity>1</ResourceQuantity>
    <ResourceCategoryName>Catering</ResourceCategoryName>
    <ProductionItemQuantity>1</ProductionItemQuantity>
    <ProductionItemExternalReference>OJ1</ProductionItemExternalReference>
    <ProductionItemDescription>Orange Juice</ProductionItemDescription>
    <ServiceOrderID>2</ServiceOrderID>
    <BookingID>1</BookingID>
  </Data>
  <Data>
    <BookingDate>2012-07-05T00:00:00</BookingDate>
    <ServiceOrderStartTime>1900-01-01T08:00:00</ServiceOrderStartTime>
    <ServiceOrderEndTime>1900-01-01T09:00:00</ServiceOrderEndTime>
    <ResourceDescription>Continental Breakfast</ ResourceDescription >
    <ResourceExternalReference>ABC123</ ResourceExternalReference >
    <ResourceQuantity>1</ResourceQuantity>
    <ResourceCategoryName>Catering</ResourceCategoryName>
    <ProductionItemQuantity>0.5</ProductionItemQuantity>
    <ProductionItemExternalReference>DA1</ProductionItemExternalReference>
    <ProductionItemDescription>Danish - Assorted</ProductionItemDescription>
    <ServiceOrderID>2</ServiceOrderID>
    <BookingID>1</BookingID>
  </Data>
</ServiceOrderDetails>

```

GetSetupTypes

Description

Returns a list of event types

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
Description	Setup Type Description	String
ID	Setup Type ID	Integer

Sample Code

```
Private Sub GetSetupTypes()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetSetupTypes("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<SetupTypes>
  <Data>
    <Description>(none)</Description>
    <ID>0</ID>
  </Data>
  <Data>
    <Description>Circle</Description>
    <ID>2</ID>
  </Data>
  <Data>
    <Description>Banquet</Description>
    <ID>3</ID>
  </Data>
</SetupTypes>
```

GetStatuses

Description

Returns a list of statuses

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
Description	Status Description	String
ID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
DisplayOnWeb	Display On Web Flag	Boolean

Sample Code

```
Private Sub GetStatuses()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetStatuses("UserID", "Password"))
```

```

        Dim dst As New DataSet("Data")
        dst.ReadXml(objReader)
    End Sub

```

Example Response

```

<Statuses>
  <Data>
    <Description>Confirmed</Description>
    <ID>1</ID>
    <StatusTypeID>-14</StatusTypeID>
    <DisplayOnWeb>true</DisplayOnWeb>
  </Data>
  <Data>
    <Description>Info Only</Description>
    <ID>2</ID>
    <StatusTypeID>-11</StatusTypeID>
    <DisplayOnWeb>false</DisplayOnWeb>
  </Data>
</Statuses>

```

GetUDFDefinitions

Description

Returns a list of User Defined Field Definitions

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
UDFDefID	ID of UDF Definition	Integer
Description	Description	String
FieldType	Field Type, i.e.: "Text", "Date", "Number", or "List"	String
FieldAppliesTo	Field Applies To, i.e. "Reservation", "Booking", "Service Order", etc.	String

Sample Code

```

Private Sub GetUDFDefinitions()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetUDFDefinitions("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<UDFDefinitions>
  <Data>
    <UDFDefID>1</UDFDefID>
    <Description>Text UDF</Description>
    <FieldType>Text</FieldType>
    <FieldAppliesTo>Reservation</FieldAppliesTo>
  </Data>
  <Data>
    <UDFDefID>2</UDFDefID>
    <Description>Number UDF</Description>
    <FieldType>Number</FieldType>
    <FieldAppliesTo>Booking</FieldAppliesTo>
  </Data>
</UDFDefinitions>

```

GetUDFListItems

Description

Returns a list of User Defined Field Definition List Items

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
UDFDefID	ID of UDF Definition	Integer
UDFDefDescription	Description of UDF Definition	String
ListUDFID	ID of List Item	Integer
ListUDFDefinition	Description of List UDF Item	String

Sample Code

```
Private Sub GetUDFListItems()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetUDFListItems("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<UDFListItems>
  <Data>
    <UDFDefID>3</UDFDefID>
    <UDFDefDescription>List UDF</UDFDefDescription>
    <ListUDFID>5</ListUDFID>
    <ListUDFDescription>Option 1</ListUDFDescription>
  </Data>
  <Data>
    <UDFDefID>3</UDFDefID>
    <UDFDefDescription>List UDF</UDFDefDescription>
    <ListUDFID>6</ListUDFID>
    <ListUDFDescription>Option 2</ListUDFDescription>
  </Data>
</UDFListItems>
```

GetWebProcessTemplateCategories

Description

Returns a list of categories for a specific web process template

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebProcessTemplateID	Web Process Template ID	Integer

Response Elements

Name	Description	Type
CategoryID	Category ID	Integer
CategoryDescription	Category Description	String
CutOffTime	Cutoff Time	DateTime
CutOffDays	Cutoff Days	Integer

CutOffHours	Cutoff Hours	Integer
-------------	--------------	---------

Sample Code

```
Private Sub GetWebProcessTemplateCategories()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc. GetWebProcessTemplateCategories ("UserID", "Password", 10))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Categories>
  <Data>
    <CategoryID>2</CategoryID>
    <CategoryDescription>Catering</CategoryDescription>
    <MinCutOffTime>1900-01-01T00:00:00</MinCutOffTime>
    <MinCutOffDays>1</MinCutOffDays>
    <MinCutOffHours>0</MinCutOffHours>
  </Data>
  <Data>
    <CategoryID>3</CategoryID>
    <CategoryDescription>Audio Visual</CategoryDescription>
    <MinCutOffTime>1900-01-01T00:00:00</MinCutOffTime>
    <MinCutOffDays>1</MinCutOffDays>
    <MinCutOffHours>0</MinCutOffHours>
  </Data>
</Categories>
```

GetWebSecurityTemplateRoles**Description**

Returns a list of roles for a web security template

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebSecurityTemplateID	Web Security Template ID	Integer

Response Elements

Name	Description	Type
WebRoleID	Web Role ID	Integer
WebRoleCode	Web Role Code	String
WebRoleDescription	Web Role Description	String

Sample Code

```
Private Sub GetWebSecurityTemplateRoles()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc. GetWebSecurityTemplateRoles ("UserID", "Password", 4))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Roles>
  <Data>
    <WebRoleID>1</WebRoleID>
    <WebRoleCode>eventbrowser</WebRoleCode>
    <WebRoleDescription>Browse Events</WebRoleDescription>
  </Data>
  <Data>
    <WebRoleID>12</WebRoleID>
    <WebRoleCode>locategroup</WebRoleCode>
```

```

    <WebRoleDescription>Locate Group</WebRoleDescription>
  </Data>
</Roles>

```

GetWebProcessTemplateSettings

Description

Returns settings for a web process template

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebProcessTemplateID	Web Process Template ID	Integer

Response Elements

Name	Description	Type
WebProcessTemplateDescription	Web Process Template Description	String
WebProcessTemplateMenuText	Web Process Template Menu Text	String
RequestStatusID	Request Status ID	Integer
RequestStatusDescription	Request Status Description	String
ReserveStatusID	Reserve Status ID	Integer
ReserveStatusDescription	Reserve Status Description	String
ConflictStatusID	Conflict Status ID	Integer
ConflictStatusDescription	Conflict Status Description	String
CancelStatusID	Cancel Status ID	Integer
CancelStatusDescription	Cancel Status Description	String
MaxBookings	Max No. Bookings Allowed per Reservation	Integer
MaxMinutes	Max No. Minutes Allowed	DateTime
CancelCutOffTime	Cancellation Cutoff Time	DateTime
CancelCutOffDays	Cancellation Number of Days	Integer
CancelCutOffHours	Cancellation Cutoff Hours	Integer
AllowCancelBookingInProgress	Cancel Booking In Progress	Boolean
CancelBookingInProgressStatusID	Cancel Booking In Progress Status ID	Integer
CancelBookingInProgressStatusDescription	Cancel Booking In Progress Status Description	String
AllowEndNow	End Booking In Progress	Boolean
MaxDays	Only Allow New Bookings Within Range of This Many Days	Integer
MaxDate	Only Allow New Bookings Prior To This Date	DateTime
MinDate	Only Allow New Bookings On/After This Date	DateTime
MinCutOffTime	New Booking Cutoff Time	DateTime
MinCutOffDays	New Booking Number of Days	Integer
MinCutOffHours	New Booking Cutoff Hours	Integer

Sample Code

```

Private Sub GetWebProcessTemplateSettings()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc. GetWebProcessTemplateSettings ("UserID", "Password", 10))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Settings>
  <Data>
    <WebProcessTemplateDescription>Executive Template</WebProcessTemplateDescription>
    <WebProcessTemplateMenuText>Executive Conference Room Template</WebProcessTemplateMenuText>
    <RequestStatusID>11</RequestStatusID>
    <RequestStatusDescription>Web Request</RequestStatusDescription>
    <ReserveStatusID>12</ReserveStatusID>
    <ReserveStatusDescription>Web Reservation</ReserveStatusDescription>
    <ConflictStatusID>2</ConflictStatusID>
    <ConflictStatusDescription>Info Only</ConflictStatusDescription>
  </Data>
</Settings>

```

```

    <CancelStatusID>3</CancelStatusID>
    <CancelStatusDescription>Cancelled</CancelStatusDescription>
    <MaxBookings>100</MaxBookings>
    <MaxMinutes>0</MaxMinutes>
    <CancelCutOffTime>1900-01-01T00:00:00</CancelCutOffTime>
    <CancelCutOffDays>1</CancelCutOffDays>
    <CancelCutOffHours>0</CancelCutOffHours>
    <AllowCancelBookingInProgress>true</AllowCancelBookingInProgress>
    <CancelBookingInProgressStatusID>3</CancelBookingInProgressStatusID>
    <CancelBookingInProgressStatusDescription>Cancelled</CancelBookingInProgressStatusDescription>
    <AllowEndNow>true</AllowEndNow>
    <MaxDays>350</MaxDays>
    <MaxDate>1900-01-01T00:00:00</MaxDate>
    <MinDate>1900-01-01T00:00:00</MinDate>
    <MinCutOffTime>1900-01-01T00:00:00</MinCutOffTime>
    <MinCutOffDays>1</MinCutOffDays>
    <MinCutOffHours>0</MinCutOffHours>
  </Data>
</Settings>

```

GetWebTemplates

Description

Returns a list of web templates

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String

Response Elements

Name	Description	Type
Description	Web Template Description	String
ID	Web Template ID	Integer
MaxBookingsPerDay	Web Template Max. No. of Bookings Per Day	Integer
MaxBookingsPerDateRange	Web Template Max. No. of Bookings Per Date Range	Integer
NewBookingDays	Web Template 'Only Allow New Bookings Within Range of This Many Days'	Integer

Sample Code

```

Private Sub GetWebTemplates()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetWebTemplates("UserID", "Password"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<WebTemplates>
  <Data>
    <Description>Executive Template</Description>
    <ID>1</ID>
    <MaxBookingsPerDay>5</MaxBookingsPerDay>
    <MaxBookingsPerDateRange>30</MaxBookingsPerDateRange>
    <NewBookingDays>1</NewBookingDays>
  </Data>
  <Data>
    <Description>Standard Template</Description>
    <ID>2</ID>
    <MaxBookingsPerDay>1</MaxBookingsPerDay>
    <MaxBookingsPerDateRange>10</MaxBookingsPerDateRange>
    <NewBookingDays>2</NewBookingDays>
  </Data>
</WebTemplates>

```

```
</Data>
</WebTemplates>
```

GetWebUserBookings

Description

Returns a list of bookings for a web user based on filters provided

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
StartDate	Start Date	DateTime
EndDate	End Date	DateTime
WebUserID	Web User ID	Integer
Statuses	List of Status IDs	Array Of Integers
StatusTypes	List of Status Types	Array Of Integers
ViewComboRoomComponents	Option to see bookings in components of combo rooms if bookings are in combo rooms	Boolean

Response Elements

Name	Description	Type
ReservationID	Reservation ID	Integer
BookingID	Booking ID	Integer
BookingDate	Date that booking is reserved to start on	DateTime
StartBookingDate	Date that booking event starts on	DateTime
TimeEventStart	Booking Event Start Time	DateTime
TimeEventEnd	Booking Event End Time	DateTime
GMTEventStart	Booking GMT Event Start Time	DateTime
GMTEventEnd	Booking GMT Event End Time	DateTime
TimeBookingStart	Booking Reserved Start Time	DateTime
TimeBookingEnd	Booking Reserved End Time	DateTime
GMTBookingStart	Booking GMT Reserved Start Time	DateTime
GMTBookingEnd	Booking GMT Reserved End Time	DateTime
EventName	Event Name	String
BuildingID	Building ID	Integer
BuildingCode	Building Code	String
Building	Building Description	String
TimeZoneID	Time Zone ID	Integer
TimeZone	Time Zone abbreviation	String
RoomID	Room ID	Integer
RoomCode	Room Code	String
RoomDescription	Location Description	String
RoomTypeID	Room Type ID	Integer
RoomType	Room Type	String
StatusID	Status ID	Integer
StatusTypeID	Status Type: -14 = Booked Space, -11 = Info Only, -13 = Wait, -12 = Cancel	Integer
StatusDescription	Status Description	String
EventTypeID	Event Type ID	Integer
EventTypeDescription	Event Type Description	String
SetupTypeID	Setup Type ID	Integer
SetupTypeDescription	Setup Type Description	String
SetupCount	Setup Count	Integer
GroupID	Group/Customer ID	Integer
GroupName	Group/Customer name	String
GroupTypeID	Group Type ID	Integer
GroupTypeDescription	Group Type Description	String
Contact1ID	First Contact ID	Integer
Contact1Name	First Contact Name	String
Contact2ID	Second Contact ID	Integer
Contact2Name	Second Contact Name	String

WebProcessTemplateID	Web Process Template ID (0 is none)	Integer
ContactEmailAddress	Contact Email Address	String
BillingReference	Billing Reference	String

Sample Code

```
Private Sub GetWebUserBookings()
    Dim dwvStatuses As DataView
    Dim gintStatuses As New Generic.List(Of Int32)
    Dim gintStatusTypes As New Generic.List(Of Int32)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New StringReader(svc.GetStatuses("UserID", "Password"))

    dwvStatuses = Me.GetData(objReader)

    For Each drv As DataRowView In dwvStatuses
        gintStatuses.Add(drv("ID"))
    Next

    gintStatusTypes.Add(-14) 'Booked Space

    Dim strStatuses As New EMSAPI.ArrayOfInt
    Dim strStatusTypes As New EMSAPI.ArrayOfInt

    strStatuses.AddRange(gintStatuses.ToArray)
    strStatusTypes.AddRange(gintStatusTypes.ToArray)

    objReader = New StringReader(svc.GetWebUserBookings("UserID", "Password", Date.Today, Date.Today, 12,
    strStatuses, strStatusTypes, False))

    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Bookings>
  <Data>
    <ReservationID>6650</ReservationID>
    <BookingID>222008</BookingID>
    <BookingDate>2009-10-04T00:00:00</BookingDate>
    <StartBookingDate>2009-10-04T00:00:00</StartBookingDate>
    <TimeEventStart>2009-10-04T16:00:00</TimeEventStart>
    <TimeEventEnd>2009-10-04T17:00:00</TimeEventEnd>
    <GMTEventStart>2009-10-04T22:00:00</GMTEventStart>
    <GMTEventEnd>2009-10-04T23:00:00</GMTEventEnd>
    <TimeBookingStart>2009-10-04T16:00:00</TimeBookingStart>
    <TimeBookingEnd>2009-10-04T17:00:00</TimeBookingEnd>
    <GMTBookingStart>2009-10-04T22:00:00</GMTBookingStart>
    <GMTBookingEnd>2009-10-04T23:00:00</GMTBookingEnd>
    <EventName>Spring Conference</EventName>
    <BuildingID>1</BuildingID>
    <BuildingCode>DEN</BuildingCode>
    <Building>Denver</Building>
    <TimeZoneID>68</TimeZoneID>
    <TimeZone>MT</TimeZone>
    <RoomID>3271</RoomID>
    <RoomCode>RM101</RoomCode>
    <RoomDescription>Room 101</RoomDescription>
    <RoomTypeID>1</RoomTypeID>
    <RoomType>Classroom</RoomType>
    <StatusID>1</StatusID>
    <StatusTypeID>-14</StatusTypeID>
    <StatusDescription>Confirmed</StatusDescription>
    <EventTypeID>4</EventTypeID>
    <EventTypeDescription>Conference</EventTypeDescription>
    <SetupTypeID>1</SetupTypeID>
    <SetupTypeDescription>U - Shape</SetupTypeDescription>
    <SetupCount>50</SetupCount>
    <GroupID>499</GroupID>
```

```

    <GroupName>ABC Company </GroupName>
    <GroupTypeID>3</GroupTypeID>
    <GroupTypeDescription>Corporate</GroupTypeDescription>
    <Contact1ID>53</Contact1ID>
    <Contact1Name>Jim Smith</Contact1Name>
    <Contact2ID>-1</Contact2ID>
    <Contact2Name/>
    <WebProcessTemplateID>2</WebProcessTemplateID>
    <ContactEmailAddress>jsmith@test.com</ContactEmailAddress>
  </Data>
</Bookings>

```

GetWebUserDetails

Description

Returns details for a specific Web User

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebUserID	Web User ID	Integer

Response Elements

Name	Description	Type
UserName	Web User Name	String
EmailAddress	Email Address	String
Phone	Phone	String
Fax	Fax	String
ExternalReference	External Reference	String
NetworkID	Network ID	String
TimeZoneID	Time Zone ID	Integer
SecurityStatus	Status ID	Integer
TemplateID	Web Security Template ID	Integer

Sample Code

```

Private Sub GetWebUserDetails()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetWebUserDetails("UserID", "Password", 2312))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<WebUserDetails>
  <Data>
    <UserName>Text Web User</UserName>
    <EmailAddress>testcontact@test.com</EmailAddress>
    <Phone>x123</Phone>
    <Fax />
    <ExternalReference />
    <NetworkID />
    <TimeZoneID>68</TimeZoneID>
    <SecurityStatus>0</SecurityStatus>
    <TemplateID>1</TemplateID>
  </Data>
</WebUserDetails>

```

GetWebUserOptions

Description

Returns a list of options for a specific web user

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebUserID	Web User ID	Integer

Response Elements

Name	Description	Type
WebProcessTemplateID	Web Process Template ID	Integer
WebProcessTemplateDescription	Web Process Template Description	String
OptionName	Option Name	String
OptionValue	Option Value	String

Sample Code

```
Private Sub GetWebUserOptions()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc. GetWebUserOptions ("UserID", "Password", 2))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Options>
  <Data>
    <WebProcessTemplateID>1</WebProcessTemplateID>
    <WebProcessTemplateDescription>Hoteling Space</WebProcessTemplateDescription>
    <OptionName>AreaID</OptionName>
    <OptionValue>-1</OptionValue>
  </Data>
  <Data>
    <WebProcessTemplateID>10</WebProcessTemplateID>
    <WebProcessTemplateDescription>Executive Template</WebProcessTemplateDescription>
    <OptionName>BuildingID</OptionName>
    <OptionValue>-1</OptionValue>
  </Data>
  <Data>
    <WebProcessTemplateID>10</WebProcessTemplateID>
    <WebProcessTemplateDescription> Executive Template </WebProcessTemplateDescription>
    <OptionName>EndTime</OptionName>
    <OptionValue>Jan 1 1900 12:00AM</OptionValue>
  </Data>
  <Data>
    <WebProcessTemplateID>10</WebProcessTemplateID>
    <WebProcessTemplateDescription> Executive Template </WebProcessTemplateDescription>
    <OptionName>EventTypeID</OptionName>
    <OptionValue>0</OptionValue>
  </Data>
  <Data>
    <WebProcessTemplateID>10</WebProcessTemplateID>
    <WebProcessTemplateDescription> Executive Template </WebProcessTemplateDescription>
    <OptionName>HomePage</OptionName>
    <OptionValue>default.aspx</OptionValue>
  </Data>
  <Data>
    <WebProcessTemplateID>10</WebProcessTemplateID>
    <WebProcessTemplateDescription> Executive Template </WebProcessTemplateDescription>
    <OptionName>StartTime</OptionName>
    <OptionValue>Jan 1 1900 12:00AM</OptionValue>
  </Data>
  <Data>
    <WebProcessTemplateID>10</WebProcessTemplateID>
```

```

        <WebProcessTemplateDescription> Executive Template </WebProcessTemplateDescription>
        <OptionName>TimeZoneID</OptionName>
        <OptionValue>-1</OptionValue>
    </Data>
</Options>

```

GetWebUsers

Description

Returns a list of web users

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebUserName	Web User Name Filter	String
EmailAddress	Email Address Filter	String
ExternalReference	External Reference	String
NetworkID	Network ID	String

Response Elements

Name	Description	Type
UserName	UserName	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String
NetworkID	Network ID	String
ID	Web User ID	Integer

Sample Code

```

Private Sub GetWebUsers()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetWebUsers("UserID", "Password", "John", "john@xyz.com",
"john123", "john123"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<WebUsers>
  <Data>
    <UserName>John Smith</UserName>
    <EmailAddress>john@xyz.com</EmailAddress>
    <ExternalReference>john12345</ExternalReference>
    <NetworkID>john12345</NetworkID>
    <ID>58</ID>
  </Data>
</WebUsers>

```

GetWebUserWebProcessTemplates

Description

Returns a list of Web Process Templates for a specific Web User

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebUserID	Web User ID	Integer

Response Elements

Name	Description	Type
Description	Web Template Description	String
ID	Web Template ID	Integer
MenuText	Web Process Template Menu	String

Sample Code

```
Private Sub GetWebUserWebProcessTemplates()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.GetWebUserWebProcessTemplates("UserID", "Password", 2312))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<WebTemplates>
  <Data>
    <Description>Template 1</Description>
    <ID>2</ID>
    <MenuText>Template 1</MenuText>
  </Data>
  <Data>
    <Description>Template 2</Description>
    <ID>3</ID>
    <MenuText>Template 2</MenuText>
  </Data>
</WebTemplates>
```

UpdateBooking (Requires EMS API Update License)**Description**

Updates a single Booking in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BookingID	Booking ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
StatusID	Status ID	Integer
RoomID	Room ID	Integer

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Room is not available."	String

Sample Code

```
Private Sub UpdateBooking()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateBooking("UserID", "Password", 1234, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"), 1, 3199))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Messages>
  <Message>
```

```

        <Message>Success!</Message>
    </Message>
</Messages>

```

UpdateBooking2 (Requires EMS API Update License)

Description

Updates a single Booking in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BookingID	Booking ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	End Time	DateTime
StatusID	Status ID	Integer
RoomID	Room ID	Integer
EventName	Event Name	String

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Room is not available."	String

Sample Code

```

Private Sub UpdateBooking()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateBooking2("UserID", "Password", 1234, Date.Today,
CDate("1/1/1900 8:00 AM"), CDate("1/1/1900 9:00 AM"), 1, 3199, "Test Meeting"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Messages>
    <Message>
        <Message>Success!</Message>
    </Message>
</Messages>

```

UpdateBookingCheckIn (Requires EMS API Update License)

Description

Updates check in for a single booking

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
BookingID	Booking ID	Integer

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Booking does not exist."	String

Sample Code

```

Private Sub UpdateBookingCheckIn()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateBookingCheckIn("UserID", "Password", 12345))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>

```

UpdateContact (Requires EMS API Update License)

Description

Updates a single Contact in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ContactID	Contact ID	Integer
ContactName	Contact Name	String
Title	Title	String
Address1	Address Line 1	String
Address2	Address Line 2	String
City	City	String
State	State	String
ZipCode	Zip Code	String
Country	Country	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String
Active	Active	Boolean

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Contact Name is required."	String

Sample Code

```

Private Sub UpdateContact()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateContact("UserID", "Password", 12345, "Test Contact", "",
"123 Main St.", "", "Denver", "CO", "80111", "", "x123", "", "testcontact@test.com", "", True))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>

```

UpdateCheckInStatus (Requires EMS API Update License)

Description

Updates check in status for a single group

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
CheckIn	Check In/Out (true = Check In, false = Check Out)	Boolean
GroupID	Group ID	Integer
BuildingID	Building ID	Integer
CheckInOutDate	Check In/Out Date	DateTime

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Building is required."	String

Sample Code

```
Private Sub UpdateCheckInStatus()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateCheckInStatus("UserID", "Password", True, 456, 1,
Date.Today)
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>
```

UpdateGroup (Requires EMS API Update License)

Description

Updates a single Group in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
GroupID	Group ID	Integer
GroupName	Group Name	String
GroupTypeID	Group Type ID	Integer
Address1	Address Line 1	String
Address2	Address Line 2	String
City	City	String
State	State	String
ZipCode	Zip Code	String
Country	Country	String
Phone	Phone	String
Fax	Fax	String
EmailAddress	Email Address	String
ExternalReference	External Reference	String
Active	Active	Boolean

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Group Name is required."	String

Sample Code

```
Private Sub UpdateGroup()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateGroup("UserID", "Password", 1234, "Test Group", 0, "123
Main St.", "", "Denver", "CO", "80111", "", "x123", "", "testgroup@test.com", "", True)
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>
```

UpdateReservation (Requires EMS API Update License)**Description**

Updates a single Reservation in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
ReservationID	Reservation ID (use -1 to lookup by Booking ID)	Integer
BookingID	Booking ID (ignored unless Reservation ID = -1)	Integer
WebUserID	Web User ID (0 = none. Use -1 to use existing Reservation's Web User ID)	Integer
WebTemplateID	Web Template ID (0 = none. Use -1 to use existing Reservation's Web Template ID, otherwise Web Template ID must be accessible by the Web User)	Integer
ReservationSourceID	Reservation Source ID (Use -1 to use existing Reservation Source ID)	I
BillingReference	Billing Reference	String

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Reservation ID does not exist."	String

Sample Code

```
Private Sub UpdateReservation()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateReservation("UserID", "Password", 1234, 0, 1, 2, 1, "abc")
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>
```

UpdateUDF (Requires EMS API Update License)

Description

Updates a single UDF in EMS (*Reservation or Booking level UDF only*)

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
UDFID	User Defined Field ID	Integer
UDFValue	UDF Value	String

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "UDF ID is invalid."	String

Sample Code

```
Private Sub UpdateUDF()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateUDF("UserID", "Password", 12345, "udf text 2"))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub
```

Example Response

```
<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>
```

UpdateWebUser (Requires EMS API Update License)

Description

Updates a single Web User in EMS

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebUserID	Web User ID	Integer
WebUserName	Web User Name	String
EmailAddress	Email Address	String
Phone	Phone	String
Fax	Fax	String
ExternalReference	External Reference	String
NetworkID	Network ID	String
TimeZoneID	Time Zone ID	Integer
StatusID	Status ID (0=Active, 1=Pending, 2=Inactive, 3=Activate On Login)	Integer
WebSecurityTemplateID	Web Security Template ID	Integer
WebProcessTemplates	Web Process Template List	Array Of Integers
Groups	Group List	Array Of Integers

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Contact Name is required."	String

Sample Code

```

Private Sub UpdateWebUser()
    Dim gintWebProcessTemplates As New Generic.List(Of Int32)
    Dim gintGroups As New Generic.List(Of Int32)

    gintWebProcessTemplates.Add(1)
    gintWebProcessTemplates.Add(2)
    gintWebProcessTemplates.Add(3)

    gintGroups.Add(1234)

    Dim strWebTemplates As New EMSAPI.ArrayOfInt
    Dim strGroups As New EMSAPI.ArrayOfInt

    strWebTemplates.AddRange(gintWebProcessTemplates.ToArray)
    strGroups.AddRange(gintGroups.ToArray)

    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateWebUser("UserID", "Password", 2312, "Test Web User",
"testwebuser@test.com", "x123", "", "", "", 68, 0, 1, strWebTemplates, strGroups))
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Messages>
  <Message>
    <Message>Success!</Message>
  </Message>
</Messages>

```

UpdateWebUserOptions (Requires EMS API Update License)**Description**

Updates web user options for a single web user

Request Parameters

Name	Description	Type
UserName	EMS User Name	String
Password	EMS User Password	String
WebUserID	Web User ID	Integer
WebProcessTemplateID	Web Process Template ID	Integer
OptionName	OptionName	String
OptionValue	Option Value	String

Response Elements

Name	Description	Type
Message	"Success!" or error, i.e. "Web User does not exist."	String

Sample Code

```

Private Sub UpdateWebUserOptions()
    Dim svc As New EMSAPI.ServiceSoapClient
    Dim objReader As New System.IO.StringReader(svc.UpdateWebUserOptions ("UserID", "Password", 5, 10,
"EventTypeID", "1")
    Dim dst As New DataSet("Data")
    dst.ReadXml(objReader)
End Sub

```

Example Response

```

<Messages>
  <Message>

```

```

    <Message>Success!</Message>
  </Message>
</Messages>

```

Release Notes

Version 1.1.14
New call added: UpdateCheckInStatus
New call added: GetCheckInStatus
New call added: GetAPIVersion
New call added: GetCourseBookings
Modified call: GetRoomDetail – added DisplayOnWeb
Modified call: GetBookings – added RoomTypeID, RoomType
Modified call: GetAllBookings – added RoomTypeID, RoomType
Modified call: GetHVACBookings – added RoomTypeID, RoomType
Modified call: GetHVACBookings – data type for HVACZones changed to Array of Strings
Version 1.1.15
New call added: GetServiceOrderDetails
Version 1.1.17
New call added: GetWebUserBookings
New call added: GetWebSecurityTemplateRoles
New call added: GetWebProcessTemplateSettings
New call added: GetBuildingHours
New call added: GetBuildingHolidays
New call added: GetWebProcessTemplateCategories
New call added: GetWebUserOptions
New call added: UpdateWebUserOptions
New call added: AddReservation3
New call added: GetReservationSources
Modified call: GetRoomDetails – added Classification
Modified call: GetRoomsAvailable2 – added RecordTypeID, FloorID, FloorDescription, FloorSequence, DisplayOnWeb, WebProcessTemplateID, RoomSetupHours, RoomTeardownHours
Version 1.1.18
New call added: GetServiceOrderDetails2
New call added: GetChangedBookings
New call added: UpdateBooking2
New call added: GetContacts
New call added: GetGroupReservations
Modified call: AddReservation – added BookingID created in EMS
Modified call: AddReservation2 – added BookingID created in EMS
Modified call: AddReservation3 – added BookingID created in EMS
Modified call: GetRoomsAvailable2 – removed WebProcessTemplateID, changed RecordTypeID
Modified call: GetWebUserBookings – removed RecordTypeID, added Statuses and StatusTypes Request Parameters
Modified call: GetBooking – added RoomTypeID
Added app setting in web.config file to UseJSON
Version 1.1.19
New call added: GetGroupBookings
Removed call: GetGroupReservations
Modified call: GetAPIVersion – added License (Basic or Advanced)
Modified call: GetRoomsAvailable2 – changed request parameters to allow for date range search.
Version 1.1.20
Modified JSON return data and date format.
Version 1.1.21
Modified JSON return data.
Modified call: GetBooking – added WebProcessTemplateID
Modified call: GetRoomDetails – added FloorSequence
Modified call: GetBuildingHours – removed GMT Open and Close times
Version 1.1.22
Minor modification to calls if error is returned from SQL Server
Version 1.1.23
Minor modification to throw an error if AddReservation, AddReservation2, AddReservation3 has no Bookings returned after creating a Reservation.
Version 1.1.24

Minor modification to data access to improve functionality when API is under load.
Version 1.1.25
New call added: AddBooking
New call added: AddBooking2
Version 1.1.26
New call added: GetCommentTypes
New call added: GetReservationComments
Modified call: GetServiceOrderDetails – added ServiceOrderID and BookingID
Modified call: GetServiceOrderDetails2 – added ServiceOrderID and BookingID
Version 1.1.27
New call added: GetFeatures
New call added: GetRoomsAvailable3
Version 1.1.28
Modified call: GetAllBookings – added ContactEmailAddress
Modified call: GetBookings – added ContactEmailAddress
Modified call: GetHVACBookings – added ContactEmailAddress
Modified call: GetAllRoomBookings – added ContactEmailAddress
Modified call: GetRoomBookings – added ContactEmailAddress
Modified call: GetCourseBookings – added ContactEmailAddress
Modified call: GetWebUserBookings – added ContactEmailAddress
Modified call: GetGroupBookings – added ContactEmailAddress
New call added: GetUDFDefinitions
New call added: GetUDFListItems
New call added: AddUDF
New call added: UpdateUDF
New call added: GetAllBookings2
Version 1.1.29
Modified call: GetRoomsAvailable2 – added SetupTypeID, SetupTypeDescription, MinimumCapacity, MaximumCapacity
Modified call: GetRoomsAvailable3 – added SetupTypeID, SetupTypeDescription, MinimumCapacity, MaximumCapacity
Modified call: GetWebUserBookings – added BillingReference
Added call: GetRoomsAvailable4
Added call: AddReservation4
Added call: UpdateReservation
Version 1.1.30
Fixed UpdateWebUser call: was throwing '@WebUserName' parameter is required error.
Version 1.1.31
Fixed GetWebUserBookings call: GMTEventEnd had incorrect value.
Version 1.1.32
New call added: GetBookingCheckIn
New call added: UpdateBookingCheckIn
Version 1.1.33
Fixed GetRoomsAvailable2-3: without SetupTypeID specified, no results would be returned
Version 1.1.34
Modified call: GetWebTemplates – added MaxBookingsPerDay
Modified call: GetWebTemplates – added MaxBookingsPerDateRange
Modified call: GetWebTemplates – added NewBookingDays
Version 1.1.35
New call added: GetBookings2