



EMS API Guide

V44.1

Last Updated: March 14, 2018

Table of Contents

CHAPTER 1: EMS API	1
CHAPTER 2: EMS API Process Overview	3
CHAPTER 3: Technical Benefits	4
EventTypes and StatusTypes	4
DisplayOnWeb	5
CHAPTER 4: Hardware and Software Requirements	6
EMS Web App	7
EMS Web App (Mobile)	8
EMS Mobile App	9
EMS for Outlook	10
EMS Platform Services	10
	11
EMS Integration for Exchange	11

EMS Desktop Client	12
EMS Kiosk App	12
EMS Room Sign App (formerly EMS Glance)	13
EMS Campus Web Service	14
EMS Services	14
Master Calendar V8.0 Services	15
Master Calendar V7.0 Services	16
Regics	16
CHAPTER 5: EMS API Requirements and Licensing	17
EMS API Requirements	17
EMS API Licensing	17
CHAPTER 6: Obtain EMS API Web Service Installation Files	19
Contact Customer Support	19
CHAPTER 7: Install or Upgrade Database Components	20
Install or Upgrade the Database Component in EMS Workplace / Campus / Enterprise / Legal / Dis-	20

trict	
Install or Upgrade the Database Component in EMS Professional	20
CHAPTER 8: Install or Upgrade EMS API Web Service	22
CHAPTER 9: Configure and Verify EMS for EMS API Web Service Use	25
Connect EMS Desktop Client to the EMS API Web Service	25
Enable JSON to Query Data	26
Verify the EMS API	27
CHAPTER 10: EMS API Functions—ADD Operations	29
AddBooking	29
Description	29
Request Parameters	29
Response Elements	30
Sample Code	30
Example Response	31
AddBooking2	31

Description	31
Request Parameters	31
Response Elements	32
Sample Code	32
Example Response	33
AddContact	33
Description	34
Request Parameters	34
Response Elements	35
Sample Code	35
Example Response	36
AddGroup	36
Description	36
Request Parameters	37
Response Elements	38

Sample Code	38
Example Response	39
AddPayment	39
Description	39
Request Parameters	39
Response Elements	40
Sample Code	40
Example Response	41
AddReservation	41
Description	41
Request Parameters	41
Response Elements	42
Sample Code	42
Example Response	43
AddReservation2	43

Description	44
Request Parameters	44
Response Elements	45
Sample Code	45
Example Response	46
AddReservation3	46
Description	46
Request Parameters	46
Response Elements	48
Sample Code	48
Example Response	48
AddReservation4	49
Description	49
Request Parameters	49
Response Elements	50

Sample Code	51
Example Response	51
AddReservation5	52
Description	52
Request Parameters	52
Response Elements	53
Sample Code	54
Example Response	54
AddServiceOrder	55
Description	55
Request Parameters	55
Response Elements	56
Sample Code	56
Example Response	57
AddServiceOrderDetail	57

Description	57
Request Parameters	57
Response Elements	59
Sample Code	59
Example Response	59
AddUDF	60
Description	60
Request Parameters	60
Response Elements	61
Sample Code	61
Example Response	61
AddWebRequest	62
Description	62
Request Parameters	62
Response Elements	63

Sample Code	64
Example Response	64
AddWebUser	65
Description	65
Request Parameters	65
Response Elements	66
Sample Code	66
Example Response	68
CHAPTER 11: EMS API Functions—ADD Operations	69
AddBooking	69
Description	69
Request Parameters	69
Response Elements	70
Sample Code	70
Example Response	71

AddBooking2	71
Description	71
Request Parameters	71
Response Elements	72
Sample Code	72
Example Response	73
AddContact	73
Description	74
Request Parameters	74
Response Elements	75
Sample Code	75
Example Response	76
AddGroup	76
Description	76
Request Parameters	77

Response Elements	78
Sample Code	78
Example Response	79
AddPayment	79
Description	79
Request Parameters	79
Response Elements	80
Sample Code	80
Example Response	81
AddReservation	81
Description	81
Request Parameters	81
Response Elements	82
Sample Code	82
Example Response	83

AddReservation2	83
Description	84
Request Parameters	84
Response Elements	85
Sample Code	85
Example Response	86
AddReservation3	86
Description	86
Request Parameters	86
Response Elements	88
Sample Code	88
Example Response	88
AddReservation4	89
Description	89
Request Parameters	89

Response Elements	90
Sample Code	91
Example Response	91
AddReservation5	92
Description	92
Request Parameters	92
Response Elements	93
Sample Code	94
Example Response	94
AddServiceOrder	95
Description	95
Request Parameters	95
Response Elements	96
Sample Code	96
Example Response	97

AddServiceOrderDetail	97
Description	97
Request Parameters	97
Response Elements	99
Sample Code	99
Example Response	99
AddUDF	100
Description	100
Request Parameters	100
Response Elements	101
Sample Code	101
Example Response	101
AddWebRequest	102
Description	102
Request Parameters	102

Response Elements	103
Sample Code	104
Example Response	104
AddWebUser	105
Description	105
Request Parameters	105
Response Elements	106
Sample Code	106
Example Response	108
CHAPTER 12: EMS API Functions—Auto Check-in Operation	109
AutoCheckin	109
Description	109
Request Parameters	109
Response Elements	110
Sample Code	110

Example Response	111
CHAPTER 13: EMS API Functions—Delete Operations	112
DeleteServiceOrder	112
Description	112
Request Parameters	112
Response Elements	113
Sample Code	113
Example Response	113
DeleteServiceOrderDetail	114
Description	114
Request Parameters	114
Response Elements	114
Sample Code	115
Example Response	115
CHAPTER 14: EMS API Functions—GET Operations	116

GetAllBookings	116
Description	116
Request Parameters	116
Response Elements	117
Sample Code	120
Example Response	121
GetAllBookings2	124
Description	124
Request Parameters	124
Response Elements	125
Sample Code	128
Example Response	129
GetAllComboRoomComponents	132
Description	132
Request Parameters	132

Response Elements	132
Sample Code	133
Example Response	133
GetAllRoomBookings	135
Description	135
Request Parameters	135
Response Elements	135
Sample Code	139
Example Response	140
GetAllRooms	143
Description	143
Request Parameters	143
Response Elements	143
Sample Code	145
Example Response	145

GetAPIVersion	146
Description	146
Request Parameters	147
Example Response	147
GetAreas	147
Description	147
Request Parameters	148
Response Elements	148
Sample Code	148
Example Response	149
GetBooking	150
Description	150
Request Parameters	150
Response Elements	150
Sample Code	154

Example Response	154
GetBookingCheckIn	157
Description	157
Request Parameters	157
Response Elements	157
Sample Code	158
Example Response	158
GetBookingHistory	159
Description	159
Request Parameters	159
Response Elements	160
Sample Code	164
Example Response	167
GetBookings	171
Description	171

Request Parameters	171
Response Elements	172
Sample Code	176
Example Response	179
GetBookings2	182
Description	182
Request Parameters	182
Response Elements	183
Sample Code	187
Example Response	190
GetBookings3	193
Description	193
Request Parameters	194
Response Elements	195
Sample Code	198

Example Response	202
GetBuildingHolidays	205
Description	205
Request Parameters	205
Response Elements	205
Sample Code	206
GetBuildingHours	208
Description	208
Request Parameters	208
Response Elements	209
Sample Code	209
Example Response	210
GetBuildings	212
Description	212
Request Parameters	212

Response Elements	212
Sample Code	213
Example Response	213
GetCategories	214
Description	214
Request Parameters	215
Response Elements	215
Sample Code	216
Example Response	217
GetCategoryStates	218
Description	218
Request Parameters	218
Response Elements	218
Sample Code	219
Example Response	219

GetChangedBookings	220
Description	220
Request Parameters	220
Response Elements	221
Sample Code	224
Example Response	227
GetComboRoomComponents	230
Description	230
Request Parameters	231
Response Elements	231
Sample Code	232
Example Response	233
GetCommentTypes	234
Description	234
Request Parameters	234

Response Elements	234
Sample Code	234
Example Response	235
GetContactDetails	236
Description	236
Request Parameters	236
Response Elements	236
Sample Code	237
Example Response	238
GetContacts	239
Description	239
Request Parameters	239
Response Elements	239
Sample Code	240
Example Response	241

GetCourseBookings	242
Description	242
Request Parameters	242
Response Elements	243
Sample Code	246
Example Response	249
GetEventTypes	252
Description	252
Request Parameters	252
Response Elements	252
Sample Code	253
Example Response	253
GetFeatures	254
Description	254
Request Parameters	254

Response Elements	255
Sample Code	255
Example Response	255
GetGroupBookings	256
Description	256
Request Parameters	256
Response Elements	257
Sample Code	261
Example Response	262
GetGroupDetails	265
Description	265
Request Parameters	265
Response Elements	266
Sample Code	267
Example Response	267

GetGroups	268
Description	268
Request Parameters	268
Response Elements	269
Sample Code	269
Example Response	270
GetGroupTypes	270
Description	270
Request Parameters	270
Response Elements	271
Sample Code	271
Example Response	271
GetHVACBookings	272
Description	272
Request Parameters	273

Response Elements	274
Sample Code	278
Example Response	281
GetPricingMethods	284
Description	284
Request Parameters	284
Response Elements	285
Sample Code	285
Example Response	285
GetReservationComments	286
Description	286
Request Parameters	286
Response Elements	286
Sample Code	287
Example Response	287

GetReservationSources	288
Description	288
Request Parameters	288
Response Elements	288
Sample Code	288
Example Response	289
GetResources	290
Description	290
Request Parameters	290
Response Elements	290
Sample Code	291
Example Response	291
GetRoomAvailability	292
Description	292
Request Parameters	292

Response Elements	293
Sample Code	293
Example Response	294
GetRoomBookings	294
Description	294
Request Parameters	294
Response Elements	295
Sample Code	299
Example Response	302
GetRoomDetails	305
Description	305
Request Parameters	305
Response Elements	305
Sample Code	306
Example Response	307

GetRooms	308
Description	308
Request Parameters	308
Response Elements	309
Sample Code	310
Example Response	311
GetRoomsAvailable	313
Description	313
Request Parameters	313
Response Elements	314
Sample Code	314
Example Response	315
GetRoomsAvailable2	316
Description	316
Request Parameters	316

Response Elements	317
Sample Code	318
Example Response	320
GetRoomsAvailable3	323
Description	323
Request Parameters	323
Response Elements	324
Sample Code	325
Example Response	328
GetRoomsAvailable4	330
Description	330
Request Parameters	331
Response Elements	332
Sample Code	333
Example Response	335

GetRoomsBySetupType	338
Description	338
Request Parameters	338
Response Elements	339
Sample Code	339
Example Response	340
GetRoomTypesByWPT	342
Description	342
Request Parameters	342
Response Elements	342
Sample Code	343
Example Response	345
GetServiceOrderDetails	346
Description	346
Request Parameters	346

Response Elements	347
Sample Code	348
Example Response	350
GetServiceOrderDetails2	351
Description	351
Request Parameters	351
Response Elements	352
Sample Code	353
Example Response	355
GetSetupTypes	357
Description	357
Request Parameters	357
Response Elements	358
Sample Code	358
Example Response	358

GetStatuses	359
Description	359
Request Parameters	359
Response Elements	360
Sample Code	360
Example Response	361
GetUDFDefinitions	362
Description	362
Request Parameters	362
Response Elements	362
Sample Code	362
Example Response	363
GetUDFListItems	364
Description	364
Request Parameters	364

Response Elements	364
Sample Code	365
Example Response	365
GetWebProcessTemplateCategories	366
Description	366
Request Parameters	366
Response Elements	367
Sample Code	367
Example Response	368
GetWebSecurityTemplateRoles	369
Description	369
Request Parameters	369
Response Elements	369
Sample Code	369
Example Response	370

GetWebProcessTemplateSettings	371
Description	371
Request Parameters	371
Response Elements	371
Sample Code	374
Example Response	374
GetWebTemplates	376
Description	376
Request Parameters	376
Response Elements	377
Sample Code	377
Example Response	378
GetWebUserBookings	379
Description	379
Request Parameters	379

Response Elements	380
Sample Code	383
Example Response	385
GetWebUserDetails	388
Description	388
Request Parameters	388
Response Elements	388
Sample Code	389
Example Response	389
GetWebUserOptions	390
Description	390
Request Parameters	391
Response Elements	391
Sample Code	391
Example Response	392

GetWebUsers	395
Description	395
Request Parameters	395
Response Elements	396
Sample Code	396
Example Response	397
GetWebUserWebProcessTemplates	397
Description	397
Request Parameters	397
Response Elements	398
Sample Code	398
Example Response	399
CHAPTER 15: EMS API Functions—Update Operations	400
UpdateBooking	400
Description	400

Request Parameters	400
Response Elements	401
Sample Code	401
Example Response	402
UpdateBooking2	402
Description	402
Request Parameters	402
Response Elements	403
Sample Code	403
Example Response	404
UpdateBookingCheckIn	404
Description	404
Request Parameters	405
Response Elements	405
Sample Code	405

Example Response	406
UpdateContact	406
Description	406
Request Parameters	406
Response Elements	408
Sample Code	408
Example Response	408
UpdateGroup	409
Description	409
Request Parameters	409
Response Elements	410
Sample Code	411
Example Response	411
UpdateReservation	411
Description	412

Request Parameters	412
Response Elements	413
Sample Code	413
Example Response	413
UpdateServiceOrder	414
Description	414
Request Parameters	414
Response Elements	415
Sample Code	415
Example Response	416
UpdateServiceOrderDetail	416
Description	416
Request Parameters	416
Response Elements	418
Sample Code	418

Example Response	418
UpdateUDF	419
Description	419
Request Parameters	419
Response Elements	419
Sample Code	420
Example Response	420
UpdateWebUser	420
Description	421
Request Parameters	421
Response Elements	422
Sample Code	422
Example Response	424
UpdateWebUserOptions	424
Description	424

Request Parameters	424
Response Elements	425
Sample Code	425
Example Response	426

CHAPTER 16: EMS API Functions—Validate Operation427

ValidateBilling	427
Description	427
Request Parameters	427
Response Elements	428
Sample Code	428
Example Response	428

CHAPTER 1: EMS API

This EMS API Guide provides information regarding special EMS customizations available to System Administration and IT users once EMS has been installed and set up.

Important: Update 9 introduced the new EMS Platform Services API. See

Also: [Install EMS Platform Services](#).

The following topics are covered in the EMS API Guide:

- » [Introduction to EMS APIs](#)
 - » [EMS API Process Overview](#)
 - » [Technical Benefits](#)
- » [Hardware and Software Requirements](#)
- » [EMS API Requirements and Licensing](#)
- » [Obtain EMS API Web Service Installation Files](#)
- » [Install or Upgrade Database Components](#)
- » [Install or Upgrade EMS API Web Service](#)
- » [Configure and Verify EMS for EMS API Web Service Use](#)

» EMS API Functions—ADD Operations

» ADD Operations

» AUTOCHECKIN Operations

» DELETE Operations

» GET Operations

» UPDATE Operations

» VALIDATE Operations

CHAPTER 2: EMS API Process Overview

To use EMS APIs, you will need to do the following:

- » [Obtain and Install Database Components](#)
- » [Install the EMS API Web Service](#)
- » [Configure and Verify EMS for EMS API Web Service Use](#)
- » [Work with the API Functions](#)

CHAPTER 3: Technical Benefits

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. Their EMS implementation 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.

There is no limit on the amount of data you can pull through the API. The EMS API provides numerous call filters so you can request an efficient and targeted set of data. This helps reduce the workload on the EMS system. It also improves the performance of your solution, so no extraneous data is processed.

EMS APIs provide infinite configurability by enabling you to completely customize your API calls and returns.

EVENTTYPES AND STATUSTYPES

You define your organization's EventTypes and Statuses; you can retrieve a list of these calls using the GetEventTypes and GetStatusTypes calls. Status types, for example, are as follows:

- » **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

DISPLAYONWEB

The `GetEventTypes` and `GetStatuses` calls above both return a field called `DisplayOnWeb`. You set this field when defining a new `EventType` or a new `Status`, to either Yes or No. `DisplayOnWeb` is the field used in EMS web interfaces, including EMS Web App and EMS Kiosk, to determine whether to display an event. 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 `GetBooking` calls; the result is you see just the booking data needed, rather than pulling in all data and then discarding a long list of bookings.

CHAPTER 4: Hardware and Software Requirements

DATABASE SERVER

Microsoft SQL Server 2012 SP2

Microsoft SQL Server 2012 SP3

Microsoft SQL Server 2014 SP1, Compatibility Level 110

Microsoft SQL Server 2016 SP1, Compatibility Level 110

NOTE: To view best practices for setting up your SQL server, refer to [EMS Best Practices for SQL Server](#).

WEB SERVER

Operating System

IIS App Pool

Windows Server 2008 R2*

7/7.5

Windows Server 2012

8

WEB SERVER

Windows Server 2012 R2

8.5

Windows Server 2016

Prerequisites

Application Pool Running 4.0**

.NET Framework 4.6.1**

Minimum System Requirements

Processor: 2.0 GHz and 4 cores or faster

Memory: 8 GB or more**

Hard-Disk Space: 1 GB or more

*For up to 100 concurrent users. Increased specs required for 100+ concurrent users.

*Requires an update to Windows Management Framework to version 3.

**= varies per EMS Software Application

EMS WEB APP

DESKTOP BROWSER

Internet Explorer 11 (please see Tip below)

DESKTOP BROWSER

Microsoft Edge (latest)

Firefox (latest)

Chrome (latest)

Safari (Mac) (latest)

*= varies per application

TIP: EMS Web App V44.1 has been optimized for Internet Explorer 11 and does not require compatibility with previous versions of Internet Explorer. EMS recommends *disabling compatibility mode* when using EMS Web App V44.1.

EMS WEB APP (MOBILE)

MOBILE BROWSER

PLATFORM

Internet Explorer for Mobile 8.1

Windows

Internet Explorer for Mobile 10

Windows

MOBILE BROWSER	PLATFORM
Chrome	Android 4.4, 6.0, 7.0, 7.1
	iOS 9.x, iOS 10.x
Safari	iOS 9.x, iOS 10.x

EMS MOBILE APP

SUPPORTED PLATFORMS

Android 4.4, 5.0, 6.0, 7.0, 7.1

iOS 9.x, 10.x

Prerequisites

To host and install EMS Mobile App, you will need the following:

- » EMS database server, web server and EMS Platform Services (see [Requirements](#))
- » mobile phone(s)

EMS FOR OUTLOOK

Microsoft® Office	365
Outlook (32- and 64-bit)	2010, 2013, 2016
.NET Framework	4.6.1
Microsoft® Visual Studio 2010 Tools for Office Runtime	VSTOR 2010
Prerequisites	
EMS Web App	Latest
On User Workstations	Desktop requirements for Microsoft® Outlook Windows 7, 8, or 10

EMS PLATFORM SERVICES

OPERATING SYSTEM	IIS
Windows Server 2008 R2	7/7.5
Windows Server 2012	8
Windows Server 2012 R2	8.5

OPERATING SYSTEM	IIS
.NET Framework	4.6.1
Application Pool	4.0
Prerequisites	
HTTPPlatformHandler IIS Module	Download Version 1.2 here OR download the installer here .
PowerShell	5+ Version
ASP.NET Version 4.6	Under Web Server (IIS)->Web Server->Application Development: <ul style="list-style-type: none"> » ISAPI Extensions » ISAPI Filters » .NET Extensibility 4.6

EMS INTEGRATION FOR EXCHANGE

Microsoft® Exchange	2010 SP3, 2013, 2016
Microsoft® Office	365

[Exchange Web Services \(EWS\) impersonation](#)

EMS DESKTOP CLIENT

OPERATING SYSTEMS

Windows 7 (32-bit and 64-bit)

Windows 8 (32-bit and 64-bit)

Windows 8.1 (32-bit and 64-bit)

Windows 10 (32-bit and 64-bit)

.NET Framework 4.6.1

EMS KIOSK APP

SUPPORTED BROWSERS

Internet Explorer 11

Google Chrome (latest)

.NET Framework 4.6.2

EMS ROOM SIGN APP (FORMERLY EMS GLANCE)

TABLET OPERATING SYSTEMS

SUPPORTED BROWSERS

iOS 9, 10

Safari

Android 5, 6

Chrome, KioWare Lite (see [Best Practice: Device Configuration Steps](#))

Windows 7, 8, 10

Internet Explorer 11, Chrome, Firefox

.NET Framework: 4.6.2

Tablet Require-
ments

Supported Resolutions

1024 x 768

1280 x 1024

1660 x 900

1920 x 1080

2048 x 1536

Note: EMS Room Sign App will run on touch-enabled tablets (with above resolutions) set in landscape mode. Integrating with ArmorActive enclosures require a browser that can play sound. **See Also:** [Best Practice: Device Configuration Steps](#).

EMS tests on and recommends the [iPad Mini](#) and [8" Samsung Tab A](#).

Prerequisites

EMS Desktop Client, running on database and web servers that meet our [EMS System Requirements](#).

EMS CAMPUS WEB SERVICE

OPERATING SYSTEM

Windows Server 2008/2008 R2*

When installing the EMS Campus Web Service on a Windows Server 2008/2008 R2 operating system, please ensure the following Web Server (IIS) Role Services are installed:

- » Common HTTP Features
- » ASP.NET
- » IIS 6 Management Compatibility

Note: The EMS Campus Planning Interface also requires Microsoft .NET Framework 4.6.1 on the web server. In Windows Server 2008/2008 R2, this is a feature that can be enabled in Windows Server Manager.

EMS SERVICES

SERVICE	.NET VERSION	DISK SPACE
EMS Email Notification Service	4.6.1	100 mb
EMS No Show Notification Service	4.6.1	100 mb

SERVICE	.NET VERSION	DISK SPACE
EMS Automated Reports Service	4.6.1	100 mb
EMS Campus Auto Sync Service	4.6.1	100 mb
EMS Check-In Service	4.6.1	100 mb
EMS CISCO TMS Service	4.6.1	10 mb
EMS Floor Plans Web Service	4.6.1	2 mb
EMS Polycom Service	4.6.1	10 mb

MASTER CALENDAR V8.0 SERVICES

SERVICE	.NET VERSION	DISK SPACE
Email Notification Service	4.6.1	10 mb
Data Connector Service	4.6.1	10 mb
Data Import Service	4.6.1	10 mb

MASTER CALENDAR V7.0 SERVICES

SERVICE	.NET VERSION	DISK SPACE
Email Notification Service	4.6.1	10 mb
Data Connector Service	4.6.1	10 mb
Data Import Service	4.6.1	10 mb

REGICS

SERVICE	.NET VERSION	DISK SPACE
Email Notification Service	4.6.1	10 mb

CHAPTER 5: EMS API Requirements and Licensing

The EMS API allows other platforms (or applications) a conduit for access to the data contained in EMS. This may include reading data *from* or sending data to the EMS database. The EMS API is commonly connected to another API or database via integration point, or ‘hook’, usually a third party application or web service. EMS Software does not provide this additional component (or offer support) to create the connection from the external application to the EMS API, as we are an open framework to it.

EMS API REQUIREMENTS

The latest version of EMS must be installed, plus additional requirements and prerequisites for each EMS Component you wish to interact with via API.

See [EMS V44.1 System Requirements](#).

EMS API LICENSING

The Basic API and the Advanced API are both optional system features that may be licensed by EMS customers. Note that the Advanced API includes the Basic API.

The Advanced 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 Advanced 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.

CHAPTER 6: Obtain 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.

CONTACT CUSTOMER SUPPORT

- » Option 1 (Recommended): Search the [Knowledge Base](#) available in the [EMS Support Portal](#).
- » Option 2: Submit a [Case](#) directly via the [EMS Support Portal](#).
- » Option 3: Email support@emssoftware.com.
- » Option 4 (Recommended for critical issues only): Phone (800) 288-4565

IMPORTANT: If you do not have a customer login, register [here](#).

CHAPTER 7: Install or Upgrade Database Components

INSTALL OR UPGRADE 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, the message "Command(s) completed successfully" will display in the Results section.

INSTALL OR UPGRADE THE DATABASE COMPONENT IN EMS PROFESSIONAL

Please contact your Professional Services Consultant or Customer Support.

- » Option 1 (Recommended): Search the [Knowledge Base](#) available in the [EMS Support Portal](#).
- » Option 2: Submit a [Case](#) directly via the [EMS Support Portal](#).
- » Option 3: Email support@emssoftware.com.
- » Option 4 (Recommended for critical issues only): Phone (800) 288-4565

IMPORTANT: If you do not have a customer login, register [here](#).

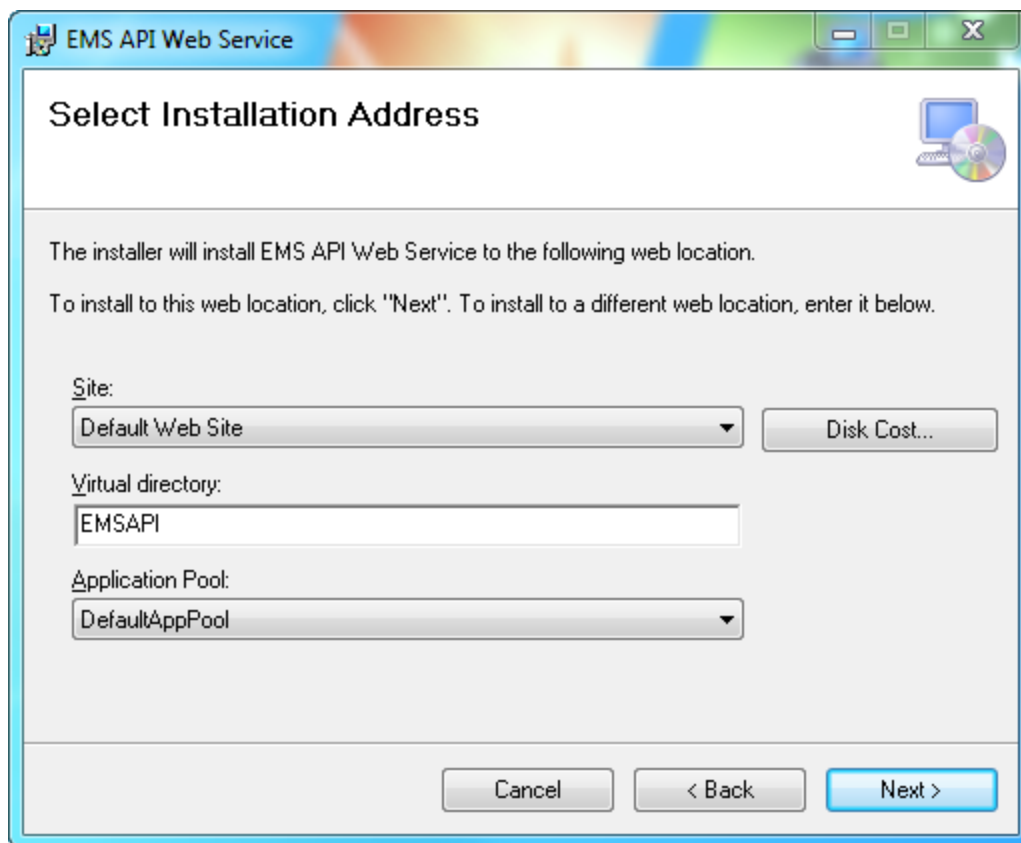
CHAPTER 8: Install or Upgrade EMS API Web Service

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

- » Install or upgrade your EMS databases as outlined in the [EMS Desktop Client Installation Instructions](#).
- » Manually **uninstall** any previous versions of the EMS API Web Service on your web server.

To install or upgrade EMS API Web Service:

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.

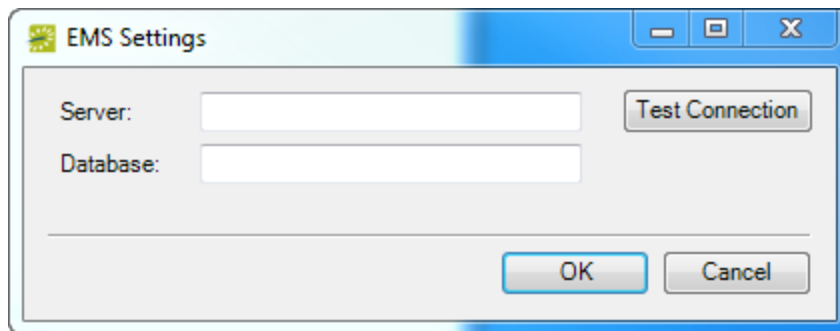


5. Select the **Site**, **Virtual Directory**, and **Application Pool**. It is recommended that you keep the default settings. The installation process will create a Physical and Virtual directory on your web server named “EMSAPI” (or whatever value you entered in the Virtual Directory field.) Click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

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



7. Enter your EMS SQL Server Name.
8. Enter your EMS Database:
- » EMS Professional customers - enter "EMSData"
 - » EMS Workplace, EMS Campus, EMS Enterprise, EMS Legal, and EMS District customers—typically named "EMS"
9. Click **OK**.
10. The Installation Complete screen will appear. Click **Close**.

CHAPTER 9: Configure and Verify EMS for EMS API Web Service Use

This topic provides information about the following:

- » [Connect EMS Desktop Client to the EMS API Web Service](#)
- » [Enable JSON to Query Data](#)
- » [Verify the EMS API](#)

CONNECT EMS DESKTOP CLIENT TO THE EMS API WEB SERVICE

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.

1. Launch the EMS client application and login with a user that has EMS administrator-level access.
2. Go to **System Administration > Security > Users**.
3. Click the **New** button. The User screen will appear.

4. Enter the following information:

- » Name
- » User ID
- » Password

5. 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.

6. Click the **OK** button.

ENABLE JSON TO QUERY DATA

If using JSON to interface with the API, it must be enabled. (It is disabled by default).

Important: Installation must be complete before enabling JSON.

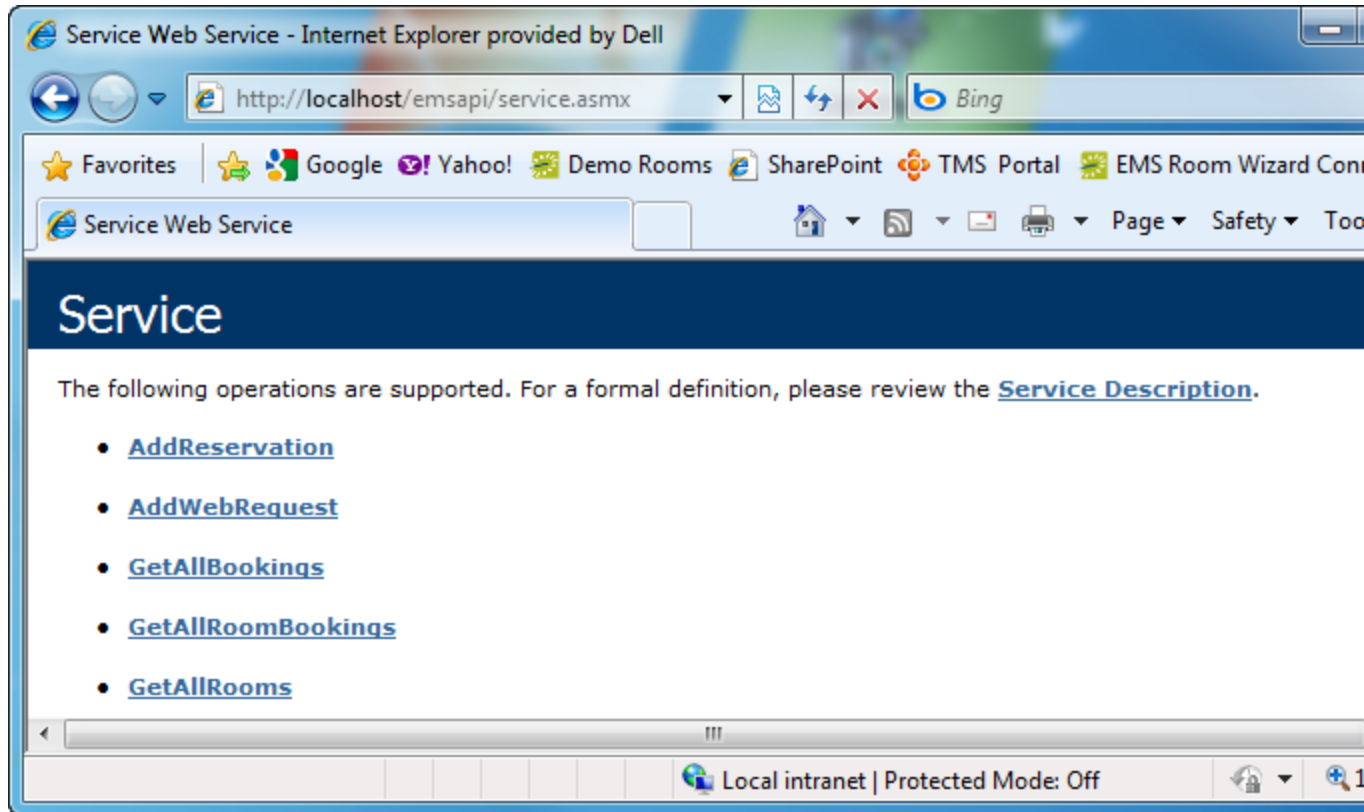
1. Edit the web.config file and enter `<add key="UseJSON" value="false"/>`.
2. Set the value to "true".

VERIFY THE EMS API

1. Open a browser and enter the following URL:

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` (replace [ServerName] with the name of your web server)

CHAPTER 10: EMS API Functions—ADD Operations

ADDBOOKING

(Requires EMS API Advanced 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

NAME	DESCRIPTION	TYPE
StartTime	Start Time	DateTime
EndTime	EndTime	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 Advanced License)

DESCRIPTION

Adds a Booking to a Reservation in EMS.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced License)

DESCRIPTION

Creates a reservation in EMS.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String

NAME	DESCRIPTION	TYPE
Password	EMS User Password	String
GroupID	Group ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	EndTime	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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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>
```

ADDRESERVATION5

(Requires EMS API Advanced 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

NAME	DESCRIPTION	TYPE
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
PONumber	PO Number for reservation	String
SetupCount	Setup count for booking	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 AddReservation5()  
  
    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", '000011asdf', '2'))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)  
  
End Sub
```

EXAMPLE RESPONSE

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

ADDSERVICEORDER

(Requires EMS API Advanced License)

DESCRIPTION

This API allows new service orders to be added to bookings.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
CategoryID	EMS category ID for which to add service order	Integer
BookingID	EMS Booking ID to attach service order	Integer
StateID	EMS State	Integer
ServeTypeID	EMS Service Type	Integer
TimeStart	Service Order Start Time	Datetime
TimeEnd	Service Order End Time	Datetime

NAME	DESCRIPTION	TYPE
EstimatedCount	Estimated Count	Integer
GuaranteedCount	Guaranteed Count	Integer
ActualCount	Actual Count	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
ServiceOrderID	Service Order ID that was created in EMS	Integer

SAMPLE CODE

```
Private Sub AddServiceOrder()  
  
    Dim svc As New EMSAPI.ServiceSoapClient  
  
    Dim objReader As New System.IO.StringReader  
    (svc.AddUDFt("UserID", "Password", 1, 2, 1, 2,  
        '1/1/2012 11:00:00.000 AM', '2/1/2012 12:00:00.000  
        PM', 1, 2, 1))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)  
  
End Sub
```

EXAMPLE RESPONSE

```
<ServiceOrder>

<Data>

    <ServiceOrderID >12345</ServiceOrderID>

</Data>

</ServiceOrder>
```

ADDSERVICEORDERDETAIL

(Requires EMS API Advanced License)

DESCRIPTION

This API allows new service order details (resources) to be added to existing service orders.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String

NAME	DESCRIPTION	TYPE
ServiceOrderID	Service Order ID for which to add details	Integer
ResourceID	Resource ID to be added	Integer
Quantity	Quantity of item being ordered	Decimal
PricingMethodID	1: 'Unit' 2: 'Hourly (Reserved Time)' 3: 'Hourly (Event Time)' 4: 'Hourly (Service Time)' 5: 'Hourly (Specific)' 6: 'Half Day/Full Day (Reserved Time)' 7: 'Half Day/Full Day (Event Time)' 8: 'Half Day/Full Day (Service Time)'	Integer
UnitPrice	Price per unit	Decimal
Notes	Notes	String
SpecialInstructions	Special Instructions	String

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
ServiceOrderDetailID	Service Order Detail ID that was created in EMS	Integer

SAMPLE CODE

```
Private Sub AddServiceOrderDetail()  
  
    Dim svc As New EMSAPI.ServiceSoapClient  
  
    Dim objReader As New System.IO.StringReader  
    (svc.AddServiceOrderDetail("UserID", "Password",  
    1, 2, 1, 2, 1, 1.00, 'notest', "special instructions"))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)  
  
End Sub
```

EXAMPLE RESPONSE

```
<ServiceOrderDetail>  
  
    <Data>  
  
        <ServiceOrderDetailID>12345</ServiceOrderDetailID>  
  
    </Data>
```

```
</ServiceOrderDetail>
```

ADDUDF

(Requires EMS API Advanced 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.AddUDFt("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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
NetworkID	Network ID	String
TimeZoneID	Time Zone ID	Integer
StatusID	Status ID (0=Active, 1=Pending, 2=Inactive, 3=Active On Login)	Integer
WebSecurityTemplateID	Web Security Template ID	Integer
WebProcessTemplates	Web Process Template List	Array Of Integers
Groups	Group List	Array Of Integers
Validated	Validated	Boolean

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, 0))
```

```
Dim dst As New DataSet("Data")

dst.ReadXml(objReader)

End Sub
```

EXAMPLE RESPONSE

```
<WebUser>
```

```
<Data>
```

```
<WebUserID>2314</WebUserID>
```

```
</Data>
```

```
</WebUser>
```

CHAPTER 11: EMS API Functions—ADD Operations

ADDBOOKING

(Requires EMS API Advanced 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

NAME	DESCRIPTION	TYPE
StartTime	Start Time	DateTime
EndTime	EndTime	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 Advanced License)

DESCRIPTION

Adds a Booking to a Reservation in EMS.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced License)

DESCRIPTION

Creates a reservation in EMS.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String

NAME	DESCRIPTION	TYPE
Password	EMS User Password	String
GroupID	Group ID	Integer
RoomID	Room ID	Integer
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	EndTime	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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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

NAME	DESCRIPTION	TYPE
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>
```

ADDRESERVATION5

(Requires EMS API Advanced 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

NAME	DESCRIPTION	TYPE
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
PONumber	PO Number for reservation	String
SetupCount	Setup count for booking	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 AddReservation5()  
  
    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", '00001lasdf', '2'))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)  
  
End Sub
```

EXAMPLE RESPONSE

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

ADDSERVICEORDER

(Requires EMS API Advanced License)

DESCRIPTION

This API allows new service orders to be added to bookings.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
CategoryID	EMS category ID for which to add service order	Integer
BookingID	EMS Booking ID to attach service order	Integer
StateID	EMS State	Integer
ServeTypeID	EMS Service Type	Integer
TimeStart	Service Order Start Time	Datetime
TimeEnd	Service Order End Time	Datetime

NAME	DESCRIPTION	TYPE
EstimatedCount	Estimated Count	Integer
GuaranteedCount	Guaranteed Count	Integer
ActualCount	Actual Count	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
ServiceOrderID	Service Order ID that was created in EMS	Integer

SAMPLE CODE

```
Private Sub AddServiceOrder()  
  
    Dim svc As New EMSAPI.ServiceSoapClient  
  
    Dim objReader As New System.IO.StringReader  
    (svc.AddUDFt("UserID", "Password", 1, 2, 1, 2,  
        '1/1/2012 11:00:00.000 AM', '2/1/2012 12:00:00.000  
        PM', 1, 2, 1))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)  
  
End Sub
```

EXAMPLE RESPONSE

```
<ServiceOrder>

<Data>

    <ServiceOrderID >12345</ServiceOrderID>

</Data>

</ServiceOrder>
```

ADDSERVICEORDERDETAIL

(Requires EMS API Advanced License)

DESCRIPTION

This API allows new service order details (resources) to be added to existing service orders.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String

NAME	DESCRIPTION	TYPE
ServiceOrderID	Service Order ID for which to add details	Integer
ResourceID	Resource ID to be added	Integer
Quantity	Quantity of item being ordered	Decimal
PricingMethodID	1: 'Unit' 2: 'Hourly (Reserved Time)' 3: 'Hourly (Event Time)' 4: 'Hourly (Service Time)' 5: 'Hourly (Specific)' 6: 'Half Day/Full Day (Reserved Time)' 7: 'Half Day/Full Day (Event Time)' 8: 'Half Day/Full Day (Service Time)'	Integer
UnitPrice	Price per unit	Decimal
Notes	Notes	String
SpecialInstructions	Special Instructions	String

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
ServiceOrderDetailID	Service Order Detail ID that was created in EMS	Integer

SAMPLE CODE

```
Private Sub AddServiceOrderDetail()  
  
    Dim svc As New EMSAPI.ServiceSoapClient  
  
    Dim objReader As New System.IO.StringReader  
    (svc.AddServiceOrderDetail("UserID", "Password",  
    1, 2, 1, 2, 1, 1.00, 'notest', "special instructions"))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)  
  
End Sub
```

EXAMPLE RESPONSE

```
<ServiceOrderDetail>  
  
    <Data>  
  
        <ServiceOrderDetailID>12345</ServiceOrderDetailID>  
  
    </Data>
```

```
</ServiceOrderDetail>
```

ADDUDF

(Requires EMS API Advanced 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.AddUDFt("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 Advanced 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

NAME	DESCRIPTION	TYPE
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	EndTime	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 Advanced 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

NAME	DESCRIPTION	TYPE
NetworkID	Network ID	String
TimeZoneID	Time Zone ID	Integer
StatusID	Status ID (0=Active, 1=Pending, 2=Inactive, 3=Active On Login)	Integer
WebSecurityTemplateID	Web Security Template ID	Integer
WebProcessTemplates	Web Process Template List	Array Of Integers
Groups	Group List	Array Of Integers
Validated	Validated	Boolean

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, 0))
```

```
Dim dst As New DataSet("Data")

dst.ReadXml(objReader)

End Sub
```

EXAMPLE RESPONSE

```
<WebUser>

<Data>

  <WebUserID>2314</WebUserID>

</Data>

</WebUser>
```

CHAPTER 12: EMS API Functions—Auto Check-in Operation

AUTOCHECKIN

(Requires EMS API Advanced License)

DESCRIPTION

Automatically checks-in to a Booking in EMS.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
EmailAddress	Email Address	String
WebUserID	Web User ID	Integer
UDFID	User Defined Field ID	Integer

NAME	DESCRIPTION	TYPE
UDFValue	Value of User Defined Field	String
BuildingID	Building ID	Integer
RoomID	Room ID	Integer
FloorID	Floor ID	Integer
BookingDate	Date of Event	DateTime
EventType	Event Type (To remove this filter use '-1'.)	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
Message	"Success!" or error, i.e. "The information provide did not return any records."	String
BookingID	Booking ID	Integer
RoomID	Room ID	Integer

SAMPLE CODE

```
Private Sub AutoCheckin()
```

```
Dim svc As New EMSAPI.ServiceSoapClient

Dim objReader As New System.IO.StringReader
(svc.AutoCheckin("UserID", "Password",
"test@test.com", 5, 6, "UDF Value",1, 2, 3,
Date.Today, 3))

Dim dst As New DataSet("Data")

dst.ReadXml(objReader)

End Sub
```

EXAMPLE RESPONSE

```
<AutoCheckinDetails>

<Data>

    <Message>Success</>

    <BookingID>1</BookingID>

    <RoomID>3</RoomID>

</Data>

</ AutoCheckinDetails
```

CHAPTER 13: EMS API Functions—Delete Operations

DELETESERVICEORDER

(Requires EMS API Advanced License)

DESCRIPTION

This API allows deletion of service orders.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
ServiceOrderID	Service Order ID	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
Message	“Success!” or error, i.e. “Service Order is not available.”	String

SAMPLE CODE

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

EXAMPLE RESPONSE

<Messages>

<Message>

<Message>Success!</Message>

</Message>

</Messages>

DELETESERVICEORDERDETAIL

(Requires EMS API Advanced License)

DESCRIPTION

This API allows deletion of service order details (resources).

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
ServiceOrderDetailID	Service Order Detail ID	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
Message	"Success!" or error, i.e. "Service Order Detail is not available."	String

SAMPLE CODE

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

EXAMPLE RESPONSE

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

CHAPTER 14: EMS API Functions—GET Operations

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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

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>

<CheckedIn>>false</CheckedIn>

</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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

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>

<CheckedIn>>false</CheckedIn>

</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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

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</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>

<CheckedIn>>false</CheckedIn>

</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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

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>

<CheckedIn>>false</CheckedIn>

</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>
```

GETBOOKINGHISTORY

DESCRIPTION

For each booking that has changed or has been created within a specified time period, this API returns all of the changes, including the baseline record.

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

NAME	DESCRIPTION	TYPE
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

Baseline (latest data for a booking that has changed during the time period).

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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

History (one for each change for the above booking for the time period).

NAME	DESCRIPTION	TYPE
Date	Date and time of change	DateTime
ChangedBy	Booking changed by	String
Field	Field that changed	String
OldValue	Old value	String
NewValue	New value	String
ID	Booking ID	Integer

SAMPLE CODE

```
Private Sub GetBookings()  
  
    Dim dvwStatuses, dvwEventTypes, dvwGroupTypes 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", "Pass-
word"))

dvwEventTypes = Me.GetData(objReader)

objReader = New StringReader(svc.GetGroupTypes("UserID", "Pass-
word"))

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

```
<BookingHistory>

<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>test@test.com</ContactEmailAddress>

<CheckedIn>false</CheckedIn>

</Data>

<Data1>

    <Date>2016-10-27T09:26:23.183</Date>

    <ChangedBy>User 1</ChangedBy>

    <Field>Room</Field>

    <OldValue>Room 1</OldValue>

    <NewValue>Room 2</NewValue>

    <ID>222008</ID>

</Data1>

</BookingHistory>
```


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

NAME	DESCRIPTION	TYPE
		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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

SAMPLE CODE

```
Private Sub GetBookings()  
  
    Dim dvwBuildings, dvwStatuses, dvwEventTypes, dvwGroupTypes 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", "Pass-  
word"))  
  
    dvwStatuses = Me.GetData(objReader)
```

```
objReader = New StringReader(svc.GetEventTypes("UserID", "Pass-  
word"))  
  
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>

<CheckedIn>>false</CheckedIn>

</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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
AddedBy	Booking Added By	String
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String
CheckedIn	0 or 1 to signify Checked in or not	Boolean

SAMPLE CODE

```
Private Sub GetBookings2()  
  
    Dim dvwBuildings, dvwStatuses, dvwEventTypes,  
        dvwGroupTypes 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.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>

<CheckedIn>>false</CheckedIn>

</Data>

</Bookings>
```

GETBOOKINGS3

DESCRIPTION

Returns a list of bookings that occurs (starts and/or ends) within a date range based on filters provided.

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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

SAMPLE CODE

```
Private Sub GetBookings2()  
  
    Dim dvwBuildings, dvwStatuses, dvwEventTypes,  
        dvwGroupTypes 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.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>

<CheckedIn>>false</CheckedIn>

</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>
```

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. GetBuidlingHours

("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

NAME	DESCRIPTION	TYPE
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>
```

GETCATEGORIES

DESCRIPTION

This API retrieves EMS category information. By passing a building parameter, it will only return categories valid for the building ID entered.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
BuildingID	EMS category ID for which to return active resources	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
CategoryID	Category ID	Integer
CategoryDescription	Category Friendly Name	String
UseCutOff	Indicates if there a cut-off when this category can no longer be added	Boolean
CutOffTime	Indicates the time of day (along with number of days specified in CutOffDays) this category can be added to bookings. (local time to the facility being booked) - NOTE: This is always used in conjunction with CutOffDays	Datetime
CutOffDays	Indicates after how many days prior to booking date this category can no longer be added to bookings. (local to the facility being booked) - NOTE: This is always used in conjunction with CutOffTime - Date & Time	Datetime

NAME	DESCRIPTION	TYPE
CutOffHours	Indicates how many hours prior to the booking start time this category can no longer be added	Integer
PONumberRequired	Indicates whether a PO number is required to order items from this category	Boolean
BillingReferenceRequired	Indicates whether a billing reference number is required to order items from this category	Boolean
CurrencySymbol	Indicates which currency symbol is applicable for resources in this category	String
UseStates	Indicates whether this category uses states to define the current state of a service order	Boolean
DefaultStateID	Indicates the default state ID to be assigned to new service orders that have not specified another state.	Integer

SAMPLE CODE

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

End Sub

EXAMPLE RESPONSE

```
<Categories>
```

```
<Data>
```

```
    <CategoryID>1</CategoryID>
```

```
    <CategoryDescription>Mountain Time</CategoryDescription>
```

```
    <UseCutOff>true</UseCutOff>
```

```
    <CutOffTime>2012-04-13T07:47:12.503</CutOffTime>
```

```
    <CutOffDays>2012-04-13T07:47:12.503</CutOffDays>
```

```
    <CutOffHours>1</CutOffHours>
```

```
    <PONumberRequired>1</PONumberRequired>
```

```
    <BillingReferenceRequired>1</BillingReferenceRequired>
```

```
    <CurrencySymbol>Dollar</CurrencySymbol>
```

```
    <UseStates>false</UseStates>
```

```
    <DefaultStateID>1</DefaultStateID>
```

```
</Data>
```

```
</Categories>
```

GETCATEGORYSTATES

DESCRIPTION

This API retrieves EMS category state information. By passing a category parameter, it will only return categories valid for the building ID entered.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
CategoryID	EMS category ID for which to return active resources	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
ID	ID	Integer
Description	Description of Category State	String
HighlightColor	RGB Color Value for State	String

SAMPLE CODE

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

EXAMPLE RESPONSE

```
<CategoryState>  
  
    <Data>  
  
        <ID>1</ID>  
  
        <Description>Mountain Time</Description>  
  
        <HighlightColor>true</HighlightColor>  
  
    </Data>  
  
</CategoryState>
```

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

NAME	DESCRIPTION	TYPE
		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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

SAMPLE CODE

```
Private Sub GetBookings()
```

```
Dim dvwStatuses, dvwEventTypes, dvwGroupTypes 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>

<CheckedIn>>false</CheckedIn>

</Data>

</Bookings>
```

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

NAME	DESCRIPTION	TYPE
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>
```

```
    <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

NAME	DESCRIPTION	TYPE
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 a 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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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 dvwBuildings, dvwStatuses, dvwEventTypes  
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"))  
  
  
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)

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

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</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>

<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

NAME	DESCRIPTION	TYPE
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
```

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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

SAMPLE CODE

```
Private Sub GetGroupBookings()  
  
    Dim dvwStatuses 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"))  
  
  
    dvwStatuses = Me.GetData(objReader)  
  
  
    For Each drv As DataRowView In dvwStatuses  
  
        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>

<CheckedIn>>false</CheckedIn>

</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

NAME	DESCRIPTION	TYPE
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	Boolean

NAME	DESCRIPTION	TYPE
	combo rooms if bookings are in combo rooms	

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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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
CheckedIn	0 or 1 to signify Checked in or not	Boolean

SAMPLE CODE

```
Private Sub GetBookings()  
  
    Dim dvwBuildings, dvwStatuses,  
        dvwEventTypes, dvwGroupTypes 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"))  
  
  
    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
```

```
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>

<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>

<CheckedIn>false</CheckedIn>

</Data>

</Bookings>
```

GETPRICINGMETHODS

DESCRIPTION

This API returns active pricing methods to use with Service Order Detail API calls.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
PricingMethodID	PricingMethodID	Integer
Description	Description of pricing method	String

SAMPLE CODE

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

EXAMPLE RESPONSE

```
<PricingMethods>  
  
<PricingMethods>  
  
<PricingMethodID>1</PricingMethodID>  
  
<Description>wqer asdfpa ahdfppwe oijf hiwer a</Description>
```

```
</PricingMethods>
```

```
</PricingMethods>
```

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>
```

GETRESOURCES

DESCRIPTION

This API returns active resource information for the category entered.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
CategoryID	EMS category ID for which to return active resources	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
GroupingID	Grouping ID for resource	Integer
ResourceID	Resource ID	Integer
GroupingDescription	Resource Grouping Description	String
ResourceDescription	Resource Description	String

NAME	DESCRIPTION	TYPE
Notes	Resource Notes	String
QuantityAvailable	Resource quantity available	Integer
ResourceCategoryTypeCode	Category Type	String

SAMPLE CODE

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

EXAMPLE RESPONSE

```
<Resources>  
  
<Data>  
  
<GroupingID>1</GroupingID>  
  
<ResourceID>2</ResourceID>
```

```
<GroupingDescription>wqer asdfpa ahdppwe oijf hiwer a</GroupingDescription>

<ResourceDescription>'asld fasdfjas; fa;sdkf asdf;aksd fas;dfk as;d-
faksjdf</ResourceDescription>

<Notes>as faasfda asd asdf</Notes>

<QuantityAvailable>1</QuantityAvailable>

<ResourceCategoryTypeCode>asd</ResourceCategoryTypeCode>

</Data>

</Resources>
```

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

NAME	DESCRIPTION	TYPE
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 the 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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
DateChanged	Date/Time Booking Changed	DateTime
ChangedBy	Booking Changed By	String
ContactEmailAddress	Contact Email Address	String
CheckedIn	0 or 1 to signify Checked in or not	Boolean

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", "Pass-  
word"))  
  
dvwEventTypes = Me.GetData(objReader)  
  
objReader = New StringReader(svc.GetGroupTypes("UserID", "Pass-  
word"))  
  
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>

<CheckedIn>>false</CheckedIn>

</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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
FloorDescription	Floor Description	String
FloorSequence	Floor Sequence	Integer
DisplayOnWeb	Display On Web	Boolean
RoomSetupHours	Room Setup Hours	Decimal
RoomTeardownHours	RoomTeardownHours	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>  
  
    <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>

<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

NAME	DESCRIPTION	TYPE
FloorDescription	Floor Description	String
FloorSequence	Floor Sequence	Integer
DisplayOnWeb	Display On Web	Boolean
RoomSetupHours	Room Setup Hours	Decimal
RoomTeardownHours	RoomTeardownHours	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 dvwTemplates As DataView  
  
    Dim dvwFeatures 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>
```

```
<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

NAME	DESCRIPTION	TYPE
FloorDescription	Floor Description	String
FloorSequence	Floor Sequence	Integer
DisplayOnWeb	Display On Web	Boolean
RoomSetupHours	Room Setup Hours	Decimal
RoomTeardownHours	RoomTeardownHours	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
PromptForBillingReference	Prompt for Billing Reference	Boolean

SAMPLE CODE

```
Private Sub GetRoomAvailability4()  
  
    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.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>

<PromptForBillingReference>true</PromptForBillingReference>

</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>
```

```
<PromptForBillingReference>true</PromptForBillingReference>
```

```
</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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
ServiceOrderID	Service Order ID	Integer
BookingID	Booking ID	Integer

SAMPLE CODE

```
Private Sub GetServiceOrderDetails()  
  
    Dim dvwBuildings, dvwStatuses 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"))  
  
  
    dvwBuildings = Me.GetData(objReader)  
  
  
    objReader = New StringReader(svc.GetStatuses  
    ("UserID", "Password"))
```

```
dvwStatuses = 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

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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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 dvwBuildings, dvwStatuses 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"))

dvwBuildings = Me.GetData(objReader)

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

dvwStatuses = 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

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 setup 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

NAME	DESCRIPTION	TYPE
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 (UDF) 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 (UDF) 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>

<WebRoleDescxription>Browse Events</WebRoleDescription>

    </Data>

<Data>

<WebRoleID>12</WebRoleID>

<WebRoleCode>locategroup</WebRoleCode>

<WebRoleDescxription>Locate Group</WebRoleDescription>
```



```
</Data>
```

```
</Roles>
```

GETWEBPROCESSTEMPLATESSETTINGS

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	DEFINITION	TYPE
WebProcessTemplateDescription	Web Process Template Description	String
WebProcessTemplateMenuText	Web Process Template Menu Text	String

NAME	DEFINITION	TYPE
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

NAME	DEFINITION	TYPE
AllowCancelBookingInProgress	Cancel Booking In Progress	Boolean
CancelBookingInProgress StatusID	Cancel Booking In Progress Status ID	Integer
CancelBookingInProgress StatusDescription	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>

<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>
```


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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
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

NAME	DESCRIPTION	TYPE
Contact2Name	Second Contact Name	String
WebProcessTemplateID	Web Process Template ID (0 is none)	Integer
ContactEmailAddress	Contact Email Address	String
BillingReference	Billing Reference	String
CheckedIn	0 or 1 to signify Checked in or not	Boolean

SAMPLE CODE

```
Private Sub GetWebUserBookings()  
  
Dim dvwStatuses 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”))
```

```
dvwStatuses = Me.GetData(objReader)

For Each drv As DataRowView In dvwStatuses

    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>

<CheckedIn>>false</CheckedIn>

</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

NAME	DESCRIPTION	TYPE
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	OptionName	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 Tem-  
plate</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 </Web-  
bProcessTemplateDescription>  
  
<OptionName>HomePage</OptionName>  
  
<OptionValue>default.aspx</OptionValue>  
  
</Data>  
  
<Data>  
  
<WebProcessTemplateID>10</WebProcessTemplateID>  
  
<WebProcessTemplateDescription> Executive Template </Web-  
bProcessTemplateDescription>  
  
<OptionName>StartTime</OptionName>  
  
<OptionValue>Jan  1 1900 12:00AM</OptionValue>  
  
</Data>  
  
<Data>  
  
<WebProcessTemplateID>10</WebProcessTemplateID>  
  
<WebProcessTemplateDescription> Executive Template </Web-  
bProcessTemplateDescription>  
  
<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

NAME	DESCRIPTION	TYPE
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>
```

CHAPTER 15: EMS API Functions—Update Operations

UPDATEBOOKING

(Requires EMS API Advanced 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

NAME	DESCRIPTION	TYPE
EndTime	EndTime	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 Advanced 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

NAME	DESCRIPTION	TYPE
BookingDate	Booking Date	DateTime
StartTime	Start Time	DateTime
EndTime	EndTime	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 Advanced 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 Advanced License)

DESCRIPTION

Updates a single Contact in EMS.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String

NAME	DESCRIPTION	TYPE
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>
```

UPDATEGROUP

(Requires EMS API Advanced 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

NAME	DESCRIPTION	TYPE
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 Advanced 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>

UPDATESERVICEORDER

(Requires EMS API Advanced License)

DESCRIPTION

This API allows new service orders to be added to bookings.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
ServiceOrderID	EMS Service Order ID for which to edit	Integer
StateID	EMS State	Integer
ServeTypeID	EMS Service Type	Integer
TimeStart	Service Order Start Time	Datetime
TimeEnd	Service Order End Time	Datetime

NAME	DESCRIPTION	TYPE
EstimatedCount	Estimated Count	Integer
GuaranteedCount	Guaranteed Count	Integer
ActualCount	Actual Count	Integer

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
Message	"Success!" or error, i.e. "Service Order ID does not exist."	String

SAMPLE CODE

```
Private Sub UpdateServiceOrder()  
  
    Dim svc As New EMSAPI.ServiceSoapClient  
  
    Dim objReader As New System.IO.StringReader  
    (svc.AddUDFt("UserID", "Password", 1, 2, 1, 2,  
        '1/1/2012 11:00:00.000 AM', '2/1/2012 12:00:00.000 PM',  
        1, 2, 1))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)
```

```
End Sub
```

EXAMPLE RESPONSE

```
<Messages>
```

```
<Message>
```

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

```
</Message>
```

```
</Messages>
```

UPDATESERVICEORDERDETAIL

(Requires EMS API Advanced License)

DESCRIPTION

This API allows for edits to service order details (resources).

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String

NAME	DESCRIPTION	TYPE
ServiceOrderDetailID	Service Order Details ID for which to edit	Integer
ResourceID	Resource ID to be added	Integer
Quantity	Quantity of item being ordered	Decimal
PricingMethodID	1: 'Unit' 2: 'Hourly (Reserved Time)' 3: 'Hourly (Event Time)' 4: 'Hourly (Service Time)' 5: 'Hourly (Specific)' 6: 'Half Day/Full Day (Reserved Time)' 7: 'Half Day/Full Day (Event Time)' 8: 'Half Day/Full Day (Service Time)'	Integer
UnitPrice	Price per unit	Decimal
Notes	Notes	String
SpecialInstructions	Special Instructions	String

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
Message	“Success!” or error, i.e. “Service Order Detail ID does not exist.”	String

SAMPLE CODE

```
Private Sub UpdateServiceOrderDetail()  
  
    Dim svc As New EMSAPI.ServiceSoapClient  
  
    Dim objReader As New System.IO.StringReader  
    (svc.UpdateServiceOrderDetail("UserID", "Password",  
    1, 2, 1, 2, 1, 1.00, 'notest', "special instructions"))  
  
    Dim dst As New DataSet("Data")  
  
    dst.ReadXml(objReader)  
  
End Sub
```

EXAMPLE RESPONSE

```
<Messages>
```

```
<Message>
```

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

```
</Message>
```



```
</Messages>
```

UPDATEUDF

(Requires EMS API Advanced 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 Advanced 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

NAME	DESCRIPTION	TYPE
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
Validated	Validated	Boolean

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, 0)

Dim dst As New DataSet("Data")

dst.ReadXml(objReader)
```

```
End Sub
```

EXAMPLE RESPONSE

```
<Messages>
```

```
<Message>
```

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

```
</Message>
```

```
</Messages>
```

UPDATEWEBUSEROPTIONS

(Requires EMS API Advanced 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

NAME	DESCRIPTION	TYPE
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>
```


CHAPTER 16: EMS API Functions—Validate Operation

VALIDATEBILLING

(Requires EMS API Advanced License)

DESCRIPTION

This API validates either Billing Reference/PO Number or both are valid in EMS.

REQUEST PARAMETERS

NAME	DESCRIPTION	TYPE
UserName	EMS User Name	String
Password	EMS User Password	String
BillingReference	Billing Reference	String
PONumber	PO Number	String

RESPONSE ELEMENTS

NAME	DESCRIPTION	TYPE
BillingReferenceValid	Is Billing Reference number valid	Boolean
PONumberValid	Is the PO Number valid	Boolean

SAMPLE CODE

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

EXAMPLE RESPONSE

```
<ValdidateBilling>  
  
<Data>  
  
    <BillingReferenceValid>false</BillingReferenceValid>  
  
    <PONumberValid>true</PONumberValid>
```

</Data>

</ValidateBilling>