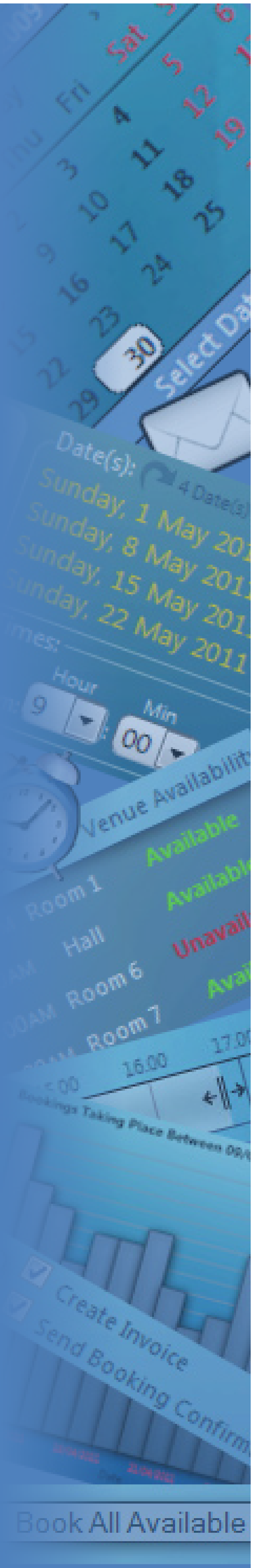
MIDAS

API Documentation

http://mid.as/api

v1.03

API



...making your facilities work for you!

...making your facilities work for you!

Table of Contents

API Usage Guide

Overview

The MIDAS API (Application Programming Interface) allows developers to interface directly with MIDAS from their own applications.

Installing the API

The MIDAS API is an optional add-on for MIDAS v4.03 (or later), that can be purchased along with MIDAS, or added at a later stage.

- To purchase MIDAS and the API addon, please visit: http://mid.as/purchase
- To add the API addon to an existing MIDAS installation, please go to MIDAS Admin Options → Manage Addons → Available Addons → API Access

Once purchased, the API will become available for one-click installation via MIDAS Admin Options → Manage Addons → Addons Ready To Install → API Access

Once installed, various API settings are available via MIDAS Admin Options → Manage Addons → Installed Addons → API Access

Making API calls

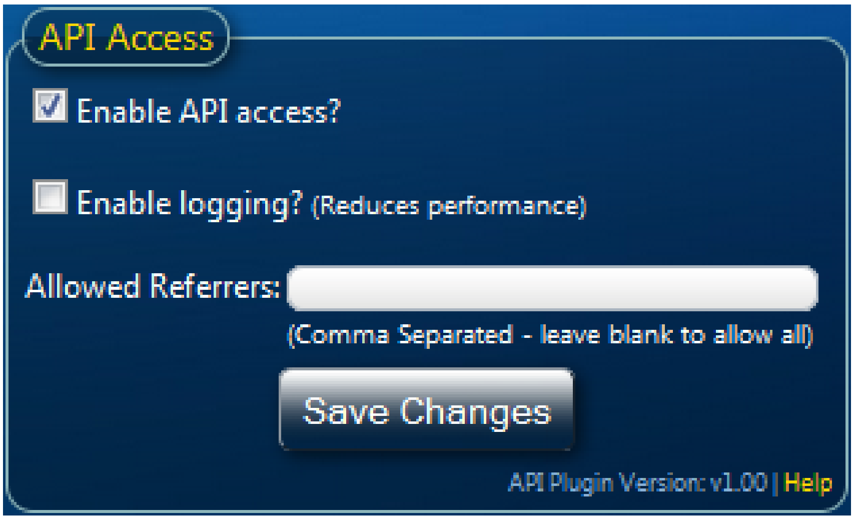
To make an API call, your application will need to perform an HTTP POST request to http://your_midas_url/api.pl. Each request must include as a minimum your unique API key, an API command and all associated required parameters for the API command issued. Please refer to the API Command Reference for details of available API commands and associated parameters.

API responses

The MIDAS API returns JSON (JavaScript Object Notation) formatted data. Example responses are shown throughout the API Command Reference. Error responses are also returned in JSON format and denoted by the term “error”.

API Settings

A number of API settings are available via MIDAS Admin Options → Manage Add-ons → API Access.



API Access

☑ Enable API access?

□ Enable logging? (Reduces performance)

Allowed Referrers: (Comma Separated - leave blank to allow all)

Setting	Description
Enable API access?	Allows enabling/disabling of API access. If disabled, any API calls will return: `{"error":"not enabled"}`
Enable logging?	*(Only available to self-hosted editions of MIDAS)* When enabled, API calls will be logged to api_log.dat (located in your MIDAS directory). The format of each line of this file is as follows: `timestamp   IP_address  api_command` This can be used to log/monitor calls to your API. It should only be enabled for debugging purposes as it will reduce performance of the API, and may lead to a large api_log.dat file if left enabled for a long of time.
Allowed Referrers	Allows calls to your API to be restricted from certain domains/IP addresses. If left blank, API calls will be allowed from any referrer. For calls from banned referrers, the API will return: `{"error":"referrer not allowed"}`

Global Parameters

The following parameters are required with each API call

Required Parameters	Possible Values	Description
key	*<API key>*	Your unique API key. For security, the API will only respond to your unique key. Calls made to the API without a key, or with an invalid key, will not be processed. You should not make your API key visible/available to anyone.
action	*<API command>*	The action you wish the API to perform. See the API Command Reference for a full list of supported commands.

The following parameters are optional, but may also be included with any API call

Optional Parameters	Possible Values	Description
l	*<language>*	By default, the API will use the "en-US" language pack where applicable. To optionally use a different language pack, its corresponding language code can be specified.
id	*<database>*	MIDAS allows you to optionally run multiple, independent, databases from the same interface. If multiple databases have been setup in your MIDAS, API calls will be made to whichever database is currently set as the "default". To make API calls to one of the other defined databases, its corresponding database code can be passed in the "id" attribute. To locate the correct database code to pass with the "id" parameter, open up your midasglobal.dat settings file in a standard text editor and locate the <database> section within this file. This section contains details of all defined databases for your MIDAS. The database "code" to use is the current value of the "r" parameter for the relevant database.
epoch	1 | 0	For API calls that require start/end times, setting "epoch" to "1" will accept start/end values in epoch seconds (instead of the default YYYYMMDDHHMM format) For API calls that return a date/time, setting "epoch" to "1" will return date/time values in epoch seconds (instead of following the current MIDAS date/time format settings, for instance, "DD/MM/YYYY @ HH:MM"

What are epoch seconds?

Epoch seconds are the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC) on 1 January 1970

Scope

Version 1.xx of the MIDAS API is a read-only API - which should be sufficient for the majority of applications, allowing you to use existing data from MIDAS in your own applications. It is envisaged that Version 2 of the API will go on to provide a bi-directional interface to MIDAS.

API Command Reference

get_activity

Returns all MIDAS user activity between two times. The optional “user” parameter can be used to limit the activity returned to a specific user, and the optional “filter” parameter can be used to limit the results to a specific activity.

Required Parameters	Possible Values	Description
start	Valid date and time (Format: YYYYMMDDHHMM)	Start time
end	Valid date and time (Format: YYYYMMDDHHMM)	End time

Optional Parameters	Possible Values	Description
user	*<User’s Full Name>* | SYSTEM	Limit the returned activity to a specific user identified by *<User’s Full Name>* or the keyword “SYSTEM”. “SYSTEM” will return activity not assigned to a specific user (for example database backups, new booking requests received, etc). If this parameter is omitted, activity for all users (including “SYSTEM”) will be returned
filter	*<activity code>*	Limit the returned activity to a specific action. See Appendix A for a list of activity codes

Example Response

```
{
      {
      "time":"5/4/2013 @ 07:53",
      "user":"SYSTEM",
      "action":"BKUP"
      },
      {
      "time":"5/4/2013 @ 07:26",
      "user":"Joe Bloggs",
      "action":"ADD",
      "data1":"12/5/2013 @ 22:00"
      }
}
```

Variables Returned	
time	Date/Time activity occurred
user	User who generated activity
action	Code of action performed. See Appendix A for a list of activity codes
data1	Returns additional data specific to action. See Appendix A for more information
data2	Returns additional data specific to action. See Appendix A for more information
data3	Returns additional data specific to action. See Appendix A for more information
data4	Returns additional data specific to action. See Appendix A for more information

get_availability

Checks whether a venue is available (free) for booking between two times

Required Parameters	Possible Values	Description
start	Valid date and time (Format: YYYYMMDDHHMM)	Start time
end	Valid date and time (Format: YYYYMMDDHHMM)	End time
venue	*<Venue name>*	Venue

Optional Parameters	Possible Values	Description
attendees	*<number>*	The number of desired attendees. If specified, availability will also be dependent upon whether the number of attendees if less than the venue's capacity and/or maximum global occupancy levels.

Example Response

```
{
      "availability":"1"
}
```

Variables Returned	
availability	Current Venue Availability A value of "1" means the venue is "available" on the dates/times specified A value of "*Unavailable*" means the venue is not available on the dates/times specified (i.e. clashes with an existing booking) Any other values also mean "Unavailable" but will specify the reason for the unavailability i.e. "*This will exceed your maximum occupancy level*", or "*Venue unavailable from 6/5/2103*"

get_bookings

Returns all bookings between two dates/times

Required Parameters	Possible Values	Description
start	Valid date and time (Format: YYYYMMDDHHMM)	Start time
end	Valid date and time (Format: YYYYMMDDHHMM)	End time

Optional Parameters	Possible Values	Description
venue	*<Venue name>*	Limits returned bookings to a specific venue
type	*<Booking Type>*	Limits returned bookings to a specific booking type
status	all | request | deleted	Controls which bookings are returned. “request” will only return booking requests. “deleted” will only return deleted bookings. “all” will return all bookings (including deleted bookings and booking requests). If this parameter is omitted only confirmed bookings will be returned

Example Response

```
{
      {
      "id":"222",
      "start":"5/4/2013 @ 13:00",
      "end":"5/4/2013 @ 14:00",
      "venue":"Room 3",
      "client":"Some User (Some Org)",
      "type":"Community",
      "attendees":"30",
      "notes":"",
      "resources":[
            {
            "resource":"Laptop",
            "qty":"2"
            },
            {
            "resource":"Projector",
            "qty":"1"
            }
            ],
      "invoice":"MIDAS00001",
      "history":[
            {
            "action":"Added",
            "date":"2/4/2013 @ 08:23",
            "user":"Joe Bloggs"
            },
            {
            "action":"Modified",
            "date":"2/4/2013 @ 16:52",
            "user":"Joe Bloggs"
            }
            ]
      },
      {
      "id":"223",
      "start":"5/4/2013 @ 14:15",
      "end":"5/4/2013 @ 16:30",
      "venue":"Room 4",
      "client":"Some User (Some Org)",
      "type":"Community",
      "attendees":"40",
      "notes":"",
      "resources":[],
      "invoice":"MIDAS00001",
      "history":[
            {
            "action":"Added",
            "date":"2/4/2013 @ 08:25",
            "user":"Joe Bloggs"
            }
            ]
      }
}
```

Variables Returned	
id	Internal Booking ID
start	Start date/time of booking
end	End date/time of booking
venue	Venue
client	Client the booking is for
type	Booking Type
attendees	Estimated number of people attending booking
notes	Booking Notes
resources	Resources assigned to booking
resource	The name of the resource
qty	The quantity of the resource
invoice	Invoice in which the booking appears
history	Booking history
action	The action performed. This will be one of the following: “Booking Request Received”, “Added”, “Modified”, “Deleted”, “Restored”, or “Invoice Created”
date	The date/time the action occurred
user	The user who performed the action
custom fields	Any custom booking fields setup in your MIDAS will also be returned

get_client

Returns client records

Required Parameters *(at least 1 required)*	Possible Values	Description
client	*<Client name>*	Client name
org	*<Organization name>*	Organization name
email	*<email address>*	Email address

Optional Parameters	Possible Values	Description
match	*exact | loose*	Allows specifying the closeness of the match. If match is set “exact” and “client” is set to “Joe Bloggs”, only clients with the name “Joe Bloggs” will be returned. If match is set to “loose” and “email” is set to “@mid.as”, any client with an “@mid.as” email address will be returned. If the match parameter is omitted only exact matches are returned

Example Response

```
{
        {
        "name":"Joe Bloggs",
        "organization":"MIDAS",
        "email":"joe@mid.as",
        "address":"PO Box 224,Cheadle,Cheshire.SK8 4AF",
        "phone":"01234 567 890",
        "fax":"01234 567 891",
        "mobile":"07123456789",
        "notes":"",
        "added":"7/8/2012 @ 16:49"
        }
}
```

Variables Returned	
name	Client’s name
organization	Client’s organization
email	Client’s email address
address	Client’s postal address
phone	Client’s telephone number
fax	Client’s fax number
mobile	Client’s mobile (cell) number
notes	Notes about the client
added	Date/time when client was added to MIDAS

get_consumable_levels

Returns the current stock level of all consumable items

Required Parameters
This API command has no additional required parameters

Example Response

```
{
        {
        "consumable":"Flipchart Paper",
        "qty_remaining":"23",
        },
        {
        "consumable":"Permanent Marker Pens",
        "qty_remaining":"156",
        }
}
```

Variables Returned	
consumable	The name of the consumable
qty_remaining	The current stock level. If no stock quantity has been defined, **qty_remaining** will return “Unlimited”

get_invoice

Retrieves a specific invoice

Required Parameters	Possible Values	Description
invoice	*<Invoice Reference>*	Invoice Reference

Example Response

```
{
        "client":"MIDAS (MIDAS2)",
        "date":"5/11/2012 @ 10:00",
        "items":[
                {
                "description":"Room 1 (6/11/2012 @ 19:30 - 21:30)",
                "qty":"2.00",
                "rate":"23.50"
                },
                {
                "description":"Room 2 (13/11/2012 @ 19:30 - 21:30)",
                "qty":"2.00",
                "rate":"23.50"
                }
                ],
        "total":"94.00",
        "tax_rate":"20.00",
        "tax_amount":"18.80",
        "paid":"0.00",
        "history":[
                {
                "action":"Created",
                "date":"5/11/2012 @ 08:45",
                "user":"MIDAS Administrator"
                },
                {
                "action":"Modified",
                "date":"5/11/2012 @ 09:42",
                "user":"MIDAS Administrator"
                },
                {
                "action":"Emailed",
                "date":"5/11/2012 @ 10:00",
                "user":"MIDAS Administrator"
                }
                ]
}
```

Variables Returned	
client	Client
date	Invoice Date (or “Invoice Not Sent if invoice hasn’t been emailed/printed)
items	Invoice items
description	The item’s description
qty	The item’s quantity
rate	The item’s rate (charge)
total	Invoice total (ex tax)

tax_rate	Tax rate (percentage)
tax_amount	Tax amount
paid	Amount paid
history	Invoice history
action	The action performed. This will be one of the following: “Created”, “Modified”, “Printed”, “Emailed”, “Payment Received”, “Payment Overdue”, “Paid In Full”
date	The date/time the action occurred
user	The user who performed the action

get_invoices

Retrieves a list of invoices for a specific client

Required Parameters *(at least 1 required)*	**Possible Values**	**Description**
client	*<Client name>*	Client name
org	*<Organization name>*	Organization name
email	*<email address>*	Email address

Example Response

```
{
       "client":"Joe Bloggs (Bloggs Inc)",
       "email":"joe@bloggsinc.com",
       "invoices":"MIDAS0001,MIDAS0003,MIDAS0004"
}
```

Variables Returned	
client	Client/Organization
email	Client’s email address
invoices	A comma separated list of all invoices associated with the client

get_messages

Retrieves all current internal messages for a specific user

Required Parameters *(only 1 required)*	**Possible Values**	**Description**
email	*<user's email>*	User's email address
user	*<user's name>*	User's name

Example Response

```
{
        {
        "message":"Hello World!",
        "author":"Joe Bloggs",
        "created","24/4/2013 @ 00:03",
        "expires","26/4/2013 @ 00:00"
        },
        {
        "message":"This is a test message",
        "author":"Jane Doe",
        "created","25/4/2013 @ 03:42",
        "expires","29/4/2013 @ 12:30"
        }
}
```

Variables Returned	
message	Details of the watch
author	The user who created the message
created	Date/Time the message was created
expires	Date/Time at which the message expires

get_reminders

Retrieves all current reminders for a specific user

Required Parameters *(only 1 required)*	Possible Values	Description
email	*<user's email>*	User's email address
user	*<user's name>*	User's name

Example Response

```
{
        {
        "reminder":"Reminder: Room 3: 25/4/2013 @ 13:00 - 15:00 Booking for Joe Bloggs",
        "expires":"25/4/2013 @ 15:00"
        },
        {
        "reminder":"Reminder: Room 3: 26/4/2013 @ 14:00 - 16:30 Booking for Jane Doe",
        "expires":"26/4/2013 @ 16:30"
        }
}
```

Variables Returned	
reminder	Details of the reminder
expires	Date/Time at which the reminder expires

get_resource_availability

Returns the quantity available of a specified resource between two dates/times

Required Parameters	Possible Values	Description
start	Valid date and time (Format: YYYYMMDDHHMM)	Start time
end	Valid date and time (Format: YYYYMMDDHHMM)	End time
resource	*<Resource name>*	Resource name
qty	*Number*	Quantity to check

Example Response

```
{
        "available":"2"
}
```

Variables Returned	
available	Quantity of specified resources available during times specified

get_setting

Returns a current MIDAS setting

Required Parameters	Possible Values	Description
setting	*<setting name>*	Setting name. See Appendix B for available setting names

Example Response

```
{
      "email_sendfrom":"midas@yourorganization.com"
}
```

Variables Returned	
<setting>	The current value of the specified <setting>

get_user

Returns user information

Required Parameters *(at least 1 required)*	Possible Values	Description
user	*<name>*	User’s full name
email	*<email address>*	User’s email address

Optional Parameters	Possible Values	Description
match	*exact | loose*	Allows specifying the closeness of the match. If match is set “exact” and “user” is set to “Joe Bloggs”, only users with the name “Joe Bloggs” will be returned. If match is set to “loose” and “user” is set to “Joe”, any user named “Joe” will be returned. If the match parameter is omitted only exact matches are returned

Example Response

```
{
      {
      "name":"Joe Bloggs",
      "email":"joe@bloggs.com",
      "added":"25/2/2013 @ 16:49",
      "last_login":"23/4/2013 @ 20:51",
      "last_password_change":"29/3/2013 @ 18:34",
      "last_modified":"25/2/2013 @ 23:42",
      "account_locked":"0",
      "failed_login_attempts":"0",
      "day_starts_at":"00",
      "day_runs_for":"24",
      "do_not_log_activity":"0"
      "email_calendar_events":"1",
      "email_messages":"0",
      "email_pending_notifications":"1",
      "email_watch_notifications":"0",
      "force_pw_change_at_login":"0",
      "suppress_messages_popup":"0",
      "venue_group_access":"Group A,Group B",
      "permissions":[
            {
            "can_add_bookings":"1",
            "can_add_clients":"1",
            "can_add_day_notes":"1",
            "can_add_historical_bookings":"0",
            "can_add_out_of_hours_bookings":"0",
            "can_change_password":"1",
            "can_delete_bookings":"2",
            "can_delete_clients":"1",
            "can_email_clients":"1",
            "can_evac":"1",
            "can_invoice":"1",
            "can_manage_booking_types":"1",
            "can_manage_midas":"1",
            "can_manage_resources":"1",
            "can_manage_users":"1",
            "can_manage_venues":"1",
            "can_modify_bookings":"2",
            "can_modify_clients":"",
            "can_print":"1",
            "can_process_requests":"1",
            "can_restore_bookings":"1",
            "can_use_mymessages":"1",
            "can_view_activity_log":"1",
            "can_view_clients":"2",
            "can_view_statistics":"1"
            }
            ]
      }
}
```

Variables Returned	
name	User's name
organization	User's email address
added	Date/Time user was added to MIDAS
last_login	Date/Time user last successfully logged in
last_password_change	Date/Time user last changed their password
last_modified	Date/Time user information was last modified
account_locked	Indicates if the user has been suspended / locked out of MIDAS
failed_login_attempts	The number of unsuccessful login attempt on this account since last successful login
day_starts_at	The hour (in 24-hour clock mode) the user's booking grid is displayed from
day_runs_for	The number of hours the user's booking grid displays
do_not_log_activity	If "0" all user activity within MIDAS will be recorded in the Recent Activity Log If "1" user activity within MIDAS will appear in the Recent Activity Log, except for logins/logouts If "2" no user activity will be recorded in the Recent Activity Log

email_calendar_events	Indicates whether reminders should also be sent to user's email as calendar events
email_messages	Indicates whether messages are forwarded to user's email
email_pending_notifications	Indicates whether new booking request notifications are forwarded to user's email
email_watch_notifications	Indicates whether watch notifications are forwarded to user's email
force_pw_change_at_login	Indicates whether user is required to change their password upon next login
suppress_messages_popup	Indicates whether the "My Messages" pop-up is suppressed (not shown) after login
venue_group_access	Indicates which Venue Groups the user has access to in the Booking Grid. A value of "*" indicates user can view all Venue Groups
permissions	The various permissions associated with the user account. Most take a value of either "1" meaning user has been granted a permission, "0" meaning the user does not have a permission. Permissions with additional possible values are indicated.
can_add_bookings	User can add bookings. "0" indicates user may not make bookings "1" indicates user may make bookings "2" indicates user may only make booking requests
can_add_clients	User can add clients
can_add_day_notes	Use can add notes to calendar dates
can_add_historical_bookings	User can add bookings for dates occurring in the past
can_add_out_of_hours_bookings	User can add bookings outside of a venue's operating hours
can_change_password	User can change their password
can_delete_bookings	User can delete bookings "0" indicates user cannot delete any bookings "1" indicates user may delete only those bookings originally added by them "2" indicates user may delete any booking
can_delete_clients	User can delete clients
can_email_clients	User can email clients directly from MIDAS
can_evac	User can print Emergency Evacuation data
can_invoice	User can use invoicing
can_manage_booking_types	User can manage Booking Types
can_manage_midas	User can manage MIDAS
can_manage_resources	User can manage Resources
can_manage_users	User can manage Users & Permissions
can_manage_venues	User can manage Venues
can_modify_bookings	User can modify bookings "0" indicates user cannot modify any bookings "1" indicates user may modify only those bookings originally added by them "2" indicates user may modify any booking
can_modify_clients	User can modify clients
can_print	User can use print functions
can_process_requests	User can approve/reject pending booking requests

can_restore_bookings	User can restore previously deleted bookings
can_use_mymessages	Use can use My Messages (Messages, Reminders, and Watches)
can_view_activity_log	Use can access the Recent Activity log
can_view_clients	User can view client information “0” indicates user cannot view any client data “1” indicates user can view client and organization names only “2” indicates user can view full client info for any client “3” indicates user can only view full client into for clients they’ve added bookings for
can_view_statistics	Use can access Statistics

get_venue

Returns venue information

Required Parameters	Possible Values	Description
venue	*<Venue name>*	Venue name

Optional Parameters	Possible Values	Description
match	*exact | loose*	Allows specifying the closeness of the match. If match is set “exact” and “venue” is set to “Meeting Room”, only the venue with the name “Meeting Room” will be returned. If match is set to “loose” and “venue” is set to “Meeting Room”, any venue containing the name “Meeting Room”, such as “Meeting Room A”, “Meeting Room B”, etc will be returned. If the match parameter is omitted only exact matches are returned

Example Response

```
{
      {
      "venue":"Room 1",
      "capacity":"20",
      "description":"A small meeting room",
      "requestable":"1",
      "rates":[
            {
            "Mon":[
                  {
                  "rate":"100.00",
                  "rate_length":"1",
                  "rate_type":"Hourly"
                  },
                  {
                  "rate":"80.00",
                  "rate_length":"4",
                  "rate_type":"Hourly"
                  },
                  {
```

```
                  "rate":"450.00",
                  "rate_length":"6",
                  "rate_type":"Booking"
                  }
                  ],
            "Tue":[
                  {
                  "rate":"20.00",
                  "rate_type":"Hourly"
                  }
                  ],
            "Wed":[
                  {
                  "rate":"20.00",
                  "rate_type":"Hourly"
                  }
                  ],
            "Thu":[
                  {
                  "rate":"20.00",
                  "rate_type":"Hourly"
                  }
                  ],
            "Fri":[
                  {
                  "rate":"20.00",
                  "rate_type":"Hourly"
                  }
                  ],
            "Sat":[
                  {
                  "rate":"20.00",
                  "rate_type":"Hourly"
                  }
                  ],
            "Sun":[
                  {
                  "rate":"20.00",
                  "rate_type":"Hourly"
                  }
                  ]
            }
            ],
      "operating_hours":[
            {
            "day":"Mon",
            "closed":"0000-0800,2100-2400",
            },
            {
            "day":"Tue",
            "closed":"0000-0800,2100-2400",
            },
            {
            "day":"Wed",
            "closed":"0000-0800,2100-2400",
            },
            {
            "day":"Thu",
            "closed":"0000-0800,2100-2400",
            },
            {
            "day":"Fri",
            "closed":"0000-0800,2100-2400",
            },
            {
            "day":"Sat",
            "closed":"0000-0930,2000-2400",
            },
            {
            "day":"Sun",
            "closed":"0000-1000,1600-2400"
            }
            ],
      "closed_from":"",
      "closed_until":"1/4/2013",
      "managers":"Jane Doe,Joe Bloggs",
      "member_of":"Group A,Group B"
      }
}
```

Variables Returned	
venue	Venue name
capacity	The maximum occupancy of the venue
description	The venue's description
requestable	Whether the venue is available for public requesting
rates	The room rate/hire charge for each day of the week
<day>	Can be one of the following: "Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun" to denote subsequent rate information for each day of the week
rate	The room/rate hire charge amount
rate_length	The minimum length of the booking the rate applies to (in hours)
rate_type	If "Hourly", the rate is per-hour, if "Daily", the rate is per-day, if "Booking", the rate is per-booking
operating_hours	The times between which the venue is closed during each day of the week
day	The day of the week
closed	A comma separated list of the times (24 hour format) the venue is closed
closed_from	The date after which the venue is considered closed
closed_until	The date until which the venue is considered closed
managers	A comma separated list of user who are "Managers" of the venue i.e. can approve Booking Requests for the venue
member_of	A comma separated list of Venue Groups to which the venue belongs to

get_venues_in_group

Returns a list of venues in a venue group

Required Parameters	Possible Values	Description
group	*<venue group>*	Name of a venue group

Example Response

```
{
	"Group A":"Room 1,Room 2,Room3"
}
```

Variables Returned	
<group name>	Comma separated list of current venues in group

get_venues_in_use

Returns a list of all venues in use at the point in time when the API call is made. If optional start and end parameters are supplied, returns a list of all venues in use between start and end parameters.

Optional Parameters	Possible Values	Description
start	Valid date and time (Format: YYYYMMDDHHMM)	Start time
end	Valid date and time (Format: YYYYMMDDHHMM)	End time

Example Response

```
{
	"venues":"Room 1,Room 2,Room3"
}
```

Variables Returned	
venues	Comma separated list of venues in use

get_watches

Retrieves all current watches for a specific user

Required Parameters *(only 1 required)*	Possible Values	Description
email	*<user's email>*	User's email address
user	*<user's name>*	User's name

Example Response

```
{
	{
	"watch":"Booking Matching [Venue: Room 1] on 25/4/2013 @ 13:00 - 15:00 For Joe Bloggs Added",
	"expires":"25/4/2013 @ 15:00"
	},
	{
	"watch":"Booking Matching [Venue: Room 4] on 25/4/2013 @ 14:00 - 16:30 For Jane Doe Deleted",
	"expires":"26/4/2013 @ 16:30"
	}
}
```

Variables Returned	
watch	Details of the watch
expires	Date/Time at which the watch expires

util_from_epoch

Converts epoch seconds to a standard date/time format

Required Parameters	Possible Values	Description
data	Epoch seconds	Epoch seconds to convert

Optional Parameters	Possible Values	Description
format	1 | 0	If omitted (or “0”), the returned date/time string will be in the format “YYYYMMDDHHMM”. If “1”, the returned date/time string will reflect the current time/date format settings in MIDAS, for instance, “DD/MM/YYYY @ HH:MM”

Example Response

```
{
      "response":"201305121008"
}
```

Variables Returned	
response	The converted epoch time, returned in a date/time format specified by the “format” parameter. If no “format” parameter specified, response is returned as “YYYYMMDDHHMM”

util_to_epoch

Converts a date/time to epoch seconds

Required Parameters	Possible Values	Description
data	Valid date and time (Format: YYYYMMDDHHMM)	Date/Time string to convert to epoch seconds

Example Response

```
{
      "response":" 1357066800"
}
```

Variables Returned	
response	The converted date/time, returned as epoch seconds

Appendix A – Activity Codes

The following codes may be used in conjunction with the “get_activity” call

CODE	Description	Data1	Data2	Data3	Data4
ADD	User added booking	Date/Time			
ADD2VG	User added venue to group	Venue	Venue Group		
ADDBT	User added booking type	Type	Color		
ADDC	User added client	Client	Organization		
ADDI	User created invoice	Invoice			
ADDRES	User added resource	Resource			
ADDU	User added User	Name	Email		
ADDV	User added venue	Venue			
ADDVG	User added venue group	Venue Group			
BKUP	MIDAS backup (Automatic)				
CANR	Booking request cancelled	Client	Organization		
DELB	User deleted booking	Date/Time			
DELBT	User deleted booking type	Type			
DELC	User deleted client	Client	Organization		
DELI	User deleted invoice	Invoice			
DELRES	User Deleted Resource	Resource			
DELV	User deleted venue	Venue			
DELVG	User deleted venue group	Venue Group			
EMS	User sent email	Client	Subject		
EMSI	User sent email invoice	Client	Invoice		
EXPB	User exported bookings				
EXPC	User exported clients				
EXPIS	User exported invoice summaries				
EXPR	User exported resources				
GENI	User generated invoice	Invoice			
IN	User logged in				
MBKUP	MIDAS backup (Manual)				
MOD	User modified booking	Date/Time			
MODBT	User modified booking type	Previous Type	Previous Color	New Type	New Color
MODC	User modified client	Client	Organization		
MODI	User modified invoice	Invoice			
MODRES	User Modified Resource	Resource			
MODU	User modified User	Name	Email		
MODV	User modified venue	Venue			
MREST	User Restored a MIDAS backup	Date/Time			
NEWR	Booking request received	Client	Organization		
OUT	User logged out				
PEED	User printed Emergency Evacuation Data				
PENA	User approved booking	Date/Time	Client		
PENR	User rejected booking	Date/Time	Client	Reason	
PRNT	User printed bookings	Print Title			
PRNTI	User printed invoice	Client	Invoice		
PWC	User changed password				
REMFVG	User removed venue from group	Venue	Venue Group		
REMU	User deleted User	Name	Email		
RENVG	User renamed venue group	Previous Venue Group	New Venue Group		
REST	User restored booking	Date/Time			
UPSI	User updated invoice status	Invoice			

Appendix B – Setting Names

The following codes may be used in conjunction with the “get_setting” call

Setting	Description	Example value
api_version	Current version of the MIDAS API	1.02
backup_email	The email address that automated database backups are sent to	backup@yourdomain.com
backup_last	The time of the last backup (epoch seconds)	1365609454
backup_persist	The number of days to keep backups on server	7
build_date	Current MIDAS build date	1365608000
datetime_amsymbol	The symbol denoting the first 12 hours of the day (when running in 12 hour clock mode)	AM
datetime_datefirst	If "1" dates are shown before times, otherwise times are shown before dates	1
datetime_dateformat	The date format	DD/MM/YYYY
datetime_dtlink	The link symbol between date and time	@
datetime_gmtoffset	The timezone's GMT offset	0
datetime_minterval	The granularity of minutes	5
datetime_pmsymbol	The symbol denoting the first 12 hours of the day (when running in 12 hour clock mode)	PM
datetime_startofweek	The day that should be considered the start of the week (0 = Sunday, 1 = Monday, etc)	1
datetime_timeformat	The time format	HH:mm
datetime_timenow	The current time (in epoch seconds)	1365696187
datetime_timezone	The timezone	Europe/London
datetime_ttlink	The link symbol between two times	-
email_bcc	The email address that outgoing email should be bcc'd to	bcc@yourdomain.com
email_sendfrom	The email address that outgoing email should appear sent from	midas@yourdomain.com
invoicing_counter	The next invoice number to be generated	4
invoicing_createbydefault	If "1" the "Create Invoice" option is checked when adding bookings	0
invoicing_createifzero	If "1" invoices will be created even if the calculated invoice total is zero. If "0" invoices will only be created if their values are non-zero	1
invoicing_currencycode	The currency code	USD
invoicing_currencysymbol	The currency symbol	$
invoicing_decimalsep	The decimal separator	.
invoicing_google_account	Your Google Wallet (formally Google Checkout) ID	0000000000000000
invoicing_google_enabled	If "1" online invoice payments are allowed via Google Wallet	1
invoicing_includezero	If "1" items will be included on invoices even if their total value is zero. If "0" invoices will only include items that have an associated cost	1
invoicing_paypal_account	Your PayPal email address	paypal@yourdomain.com

invoicing_paypal_enabled	If "1" online invoice payments are allowed via PayPal	1
invoicing_pay_require_email	If "1" clients are required to enter their email address in order to view their invoice online	1
invoicing_prefix	The prefix that is added to all your invoice numbers	MIDAS
invoicing_rounding	If "1" invoice values are rounded up to the nearest whole number 0	1
invoicing_silentdisc	If "1" booking type discounts are applied to invoices without any indication. If "0" invoices will denote that a discount has been applied	0
invoicing_tax	The default tax percentage to be added to invoices	0
invoicing_thousandsep	The thousand separator	,
occupancy_max	The maximum number of people allowed on your site at any one time	510
occupancy_show	Show the occupancy level when adding/modifying bookings	1
occupancy_warn	Warn if the number of people on site exceeds this value	300
print_evactolerance	When printing Emergency Evacuation Data, this number reflects how many minutes either side of the current time the calculation of the number of people on site should account for	15
print_sortby	The booking field to sort booking print outs by	start
print_sortdir	The sort order of print outs (0 = ascending, 1 = descending)	0
search_maxresults	The maximum number of search results to return per page	50
version	Current MIDAS version	4.03
view_autoclose_alerts	The number of seconds the "My Messages" pop-up should display for, if enabled by users	10
view_autoclose_notes	The number of seconds after which any "day notes" pop-ups should automatically close	15
view_changemonthcell	Allow users to choose what's displayed in the monthly overview	1
view_cutoffdays	The number of days to keep temporary logs for	30
view_default	The default view in the Booking Grid	All
view_gridrowheight	The height (in pixels) of each row in the booking grid	30
view_gridvenuewidth	The width (in pixels) of the first column in the booking grid	100
view_showonblocks	The booking field(s) to show on booking "blocks" in the booking grid	organisation
view_showonmonthcell	The booking field to show on dates in the monthly overview	client
view_showontools	The booking field(s) to show on tooltips when hovering over booking "blocks" in the booking grid	organisation
view_updatefrequency	The number of seconds between successive background data refreshes	30
view_viewsize	The number of days to show in the booking grid by default	1
webrequest_alloweddomains	A comma separated list of email domains from which	@yourdomain.com,

	booking requests are permitted	@hotmail.com
webrequest_blockcolor	The color of existing booking blocks on the public booking request screen	red
webrequest_disablepast	The date past which booking requests are not permitted	
webrequest_enabled	If "1" public booking requests are enabled, if "0" public booking requests are disabled	1
webrequest_leadintime	The number of days in advance public booking request must be made	7
webrequest_leadouttime	The number of days in advance public booking request are allowed to be made for	14
webrequest_showonblocks	What information should be shown on booking blocks in the public booking request screen	times

Appendix C – Error Handling

When making API calls, your applications should take into account possible errors that may be returned.

API errors will be returned in JSON format as follows:

Example Error Response
`{"error":"error response"}`

Error Response	Meaning	Applies To
"invalid api key"	You have not supplied your API key in your call, or the API key supplied is invalid	*All*

"no matches found"	No data was returned for the API call. Try broadening the search parameters	`get_activity`
		`get_bookings`
		`get_client`
		`get_consuable_levels`
		`get_invoice`
		`get_invoices`
		`get_messages`
		`get_reminders`
		`get_user`
		`get_venues_in_group`
		`get_watches`

"not enabled"	API access is not enabled at this time. API access can be enabled from within MIDAS via MIDAS Admin Options → Manage Addons → API Access → Enabled	*All*
"referrer not allowed"	The API call originates from a blocked domain/IP address. Allowed domains/IP's may be specified from within MIDAS via MIDAS Admin Options → Manage Addons → API Access → Allowed Referrers	*All*
"required parameter missing"	One or more parameters are required for the particular call you're making is missing	*All*
"unknown booking type"	Indicates that the specified booking type doesn't exist	`get_bookings` (when the type parameter is specified)
"unknown command"	The API command you've specified was unrecognized. Please check you're using a valid command	*All*
"unknown resource"	Indicates that the specified resource doesn't exist	`get_resource_availability`
"unknown setting"	Indicates that the specified setting isn't recognized	`get_setting`

"unknown user"	Indicates that the specified user doesn't exist	`get_activity`
		`get_messeges`
		`get_reminders`
		`get_watches`

"unknown venue group"	Indicates that the specified venue group doesn't exist	`get_venues_in_group`

"unknown venue"	Indicates that the specified venue doesn't exist	`get_availability`
		`get_bookings` (when the venue parameter is specified)
		`get_venue`

Code Samples

jQuery

```
$.post("http://your_midas_url/api.pl", { key: "your_api_key", action:
"get_client", client: "Joe Bloggs" })
.done(function(response) {
 alert("Response: " + response);
}, "JSON");
```

Perl

```
use LWP::UserAgent;
my$ua = LWP::UserAgent->new(env_proxy => 0,keep_alive => 0,timeout =>
30,agent =>'Mozilla/4.0 (compatible; MSIE 9.0; Windows NT 5.0)');
my$r = $ua-
>post("http://your_midas_url/api.pl",[key=>"your_api_key",action=>"get_c
lient",client=>"Joe Bloggs"]);
if ($r->is_success) {
     $response=$r->content;
}
```

PHP

```
$myvars = "key=your_api_key&action=get_client&client=Joe Bloggs";

$ch = curl_init("http://your_midas_url/api.pl");
curl_setopt( $ch, CURLOPT_POST, 1);
curl_setopt( $ch, CURLOPT_POSTFIELDS, $myvars);
curl_setopt( $ch, CURLOPT_FOLLOWLOCATION, 1);
curl_setopt( $ch, CURLOPT_HEADER, 0);
curl_setopt( $ch, CURLOPT_RETURNTRANSFER, 1);

$response = curl_exec( $ch );
```

Release Notes

v1.03	
20th March 2014	Added: "member_of" to the list of returned values for the "get_venue" call Change: Format of "rates" information returned by "get_venue" call updated for MIDAS v4.06 Fixed: "closed_from"/"closed_to" values from "get_venue" call may return 1/1/1970

v1.02	
18th February 2014	Added: Support for multiple databases

v1.01	
7th October 2013	Added: "get_venues_in_use" call

v1.00	
1st June 2013	Initial API release

For the most up-to-date API documentation, please visit http://mid.as/api