



HeartCore



Heartcore, Inc.
www.heartcore.co.jp
support@heartcore.co.jp

Heartcore, Inc.

HeartCore

Web コンテンツ管理システム

・ ・ ・ ・ ・

プログラミング **API** ガイド

簡単・安価に Web サイトを作成・公開・管理

著作権および機密情報

著作権 ハートコア株式会社 1999–2020。当ユーザガイドはハートコア株式会社の機密情報により構成されています。本書の無断複製、複写、転写、情報検索方式での格納、又は、いかなる言語への翻訳においてもハートコア株式会社の許可無しでは固く禁じられています。

注意

ハートコア株式会社は、本書の記載内容において、何時でも予告なく変更することがあります。また、本書において明示または黙示を問わず、一切保証致しません。ハートコア株式会社はいかなる場合においても、購入した際および本書の使用時、またはこれに含まれる情報によって生じた間接的、特別かつ偶発的損害に関しての責任を負いません。

ライセンスおよび商標

HeartCore Web コンテンツ管理や HeartCore ロゴは商標であり、ハートコア株式会社独自の登録商標です。また、その他の企業、製品、トレードネームは商標であり、各所有者の登録商標です。

HeartCore Web コンテンツ管理は wz_dragdrop.js ライブラリを使用しています。Copyright (c) 2002-2003 Walter Zorn (www.walterzorn.com) ライセンスは GNU Lesser General Public License (LGPL) (<http://www.gnu.org/copyleft/lesser.html>) の条件に準じます。

HeartCore Web コンテンツ管理は wz_jsgraphics.js ライブラリを使用しています。Copyright (c) 2002-2004 Walter Zorn (www.walterzorn.com) ライセンスは GNU Lesser General Public License (LGPL) (<http://www.gnu.org/copyleft/lesser.html>) の条件に準じます。

HeartCore Web コンテンツ管理は Dynarch DHTML Calendar ライブラリを使用しています。Copyright (c) 2002-2005 Mihai Bazon (<http://www.bazon.net/mishoo> - <http://www.dynarch.com/projects/calendar>) ライセンスは GNU Lesser General Public License (LGPL) (<http://www.gnu.org/licenses/lgpl.html>) の条件に準じます。

HeartCore Web コンテンツ管理は Kryogenix sortable ライブラリを使用しています。Copyright (c) 1997-2005 Stuart Langridge (<http://www.kryogenix.org/code/browser/sortable/>) ライセンスは MIT License (<http://www.kryogenix.org/code/browser/license.html>) の条件に準じます。

HeartCore Web コンテンツ管理は SWFupload コンポーネントとライブラリを使用しています。Copyright (c) 2006-2007 Lars Huring, Olov Nilzén, Mammon Media Copyright (c) 2007-2008 Jake Roberts (<http://www.swfupload.org/>) ライセンスは MIT License (<http://www.opensource.org/licenses/mit-license.php>) の条件に準じます。

HeartCore Web コンテンツ管理は Prototype ライブラリを使用しています。Copyright (c) 2005 Sam Stephenson (<http://prototype.conio.net/>) ライセンスは MIT License (<http://www.kryogenix.org/code/browser/license.html>) の条件に準じます。

HeartCore Web コンテンツ管理は Scriptaculous ライブラリを使用しています。Copyright (c) 2005 Thomas Fuchs (<http://script.aculo.us/>) このライブラリはフリーソフトウェアとして使用が許可されています。

HeartCore Web コンテンツ管理は Rico ライブラリの一部を使用しています。Copyright (c) 2005 Sabre Airline Solutions (<http://openrico.org/>) ライセンスは Apache License Version 2.0 の条件に準じます。

HeartCore Web コンテンツ管理は Rico Livegrid Plus ライブラリを使用しています。Copyright (c) 2006 Matt Brown (<http://dowdybrown.com/>) ライセンスは Apache License Version 2.0 の条件に準じます。

HeartCore Web コンテンツ管理は Lightbox library ライブラリを使用しています。Copyright (c) 2006 Lokesh Dhakar (<http://www.huddletogether.com/>) ライセンスは Creative Commons Attribution 2.5 License (<http://creativecommons.org/licenses/by/2.5/>) の条件に準じます。

HeartCore Web コンテンツ管理は TableKit ライブラリを使用しています。Copyright (c) 2007 Andrew Tetlaw & Millstream Web Software (<http://www.millstream.com.au/view/code/tablekit/>) このライブラリはフリーソフトウェアとして使用が許可されています。

HeartCore Web コンテンツ管理は jQuery ライブラリを使用しています。Copyright (c) 2011 John Resig (<http://jquery.org/>) ライセンスは MIT License (<http://jquery.org/license/>) の条件に準じます。

HeartCore Web コンテンツ管理は jsTree ライブラリを使用しています。Copyright (c) 2010 Ivan Bozhanov (<http://jstree.com/>) ライセンスは MIT License (<http://www.opensource.org/licenses/mit-license.php>) の条件に準じます。

HeartCore Web コンテンツ管理は JavaBeans Activation Framework ライブラリを使用しています。Copyright (c) Sun Microsystems (<http://www.sun.com/>) ライセンスは Sun Microsystems, Inc. Binary Code License Agreement の条件に準じます。

HeartCore Web コンテンツ管理は JavaMail ライブラリを使用しています。Copyright (c) 2009 Sun Microsystems (<http://www.sun.com/>) ライセンスは Sun Microsystems, Inc. Binary Code License Agreement の条件に準じます。

HeartCore Web コンテンツ管理は Apache Jakarta JCS ライブラリを使用しています。Copyright (c) 2001-2007 The Apache Software Foundation (<http://www.apache.org/>) ライセンスは Apache License Version 2.0 の条件に準じます。

HeartCore Web コンテンツ管理は Apache Commons Logging ライブラリを使用しています。Copyright (c) 2003-2007 The Apache Software Foundation (<http://www.apache.org/>) ライセンスは Apache License Version 2.0 の条件に準じます。

HeartCore Web コンテンツ管理は Apache log4j ライブラリを使用しています。Copyright (c) 2010 The Apache Software Foundation (<http://www.apache.org/>) ライセンスは Apache License Version 2.0 の条件に準じます。

HeartCore Web コンテンツ管理は concurrent ライブラリを使用しています。Copyright (c) Doug Lea (<http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html>) このライブラリはフリーソフトウェアとして使用が許可されています。

HeartCore Web コンテンツ管理は slidetabs ライブラリを使用しています。Copyright (c) WebStack (<http://www.slidetabs.com/>)

HeartCore Web コンテンツ管理は ContentBuilder.js ライブラリを使用しています。Copyright (c) InnovaStudio (<http://www.innovastudio.com/>)

HeartCore Web コンテンツ管理は CodeMirror ライブラリを使用しています。Copyright (c) 2017 Marijn Havebeke (marijnh@gmail.com) (<http://www.codemirror.net/>) ライセンスは MIT License (<https://codemirror.net/LICENSE>) の条件に準じます。

HeartCore Web コンテンツ管理は Tiny Colorpicker ライブラリを使用しています。Copyright (c) 2013 Maarten Baijs (<http://www.baijs.com/>) ライセンスは MIT License の条件に準じます。

HeartCore Web コンテンツ管理は HTML5 FormData Polyfill を使用しています。Copyright (c) 2016 Jimmy Karl Roland Wärtig (<https://github.com/jimmywarting/FormData>) ライセンスは MIT License の条件に準じます。

HeartCore

Web コンテンツ管理システム

簡単・安価に Web サイトを作成・公開・管理

はじめに

本書は、HeartCoreWeb コンテンツ管理システム用のプログラミング API ガイドです。Web サイトを作成、公開、管理するにあたり、HeartCoreWeb コンテンツ管理システムにて使用する独自のプログラムスクリプトをインストール、設定、使用方法をご説明します。

本書は、5つのパートに大別されます。

パート1では、追加の Web サイト管理セクションや、メニュー項目、機能を使った、独自追加モジュールの開発、インストール、設定方法をご説明します。

パート2では、プログラムされた追加 Web サイトコンテンツを使った、独自拡張プログラムスクリプトの開発、インストール、設定方法をご説明します。

パート3では、E コマース追加モジュールに関し、独自の自動商品在庫確認・商品配信プログラムスクリプトを開発、インストール、設定する方法をご説明します。

パート4では、Web サイトコンテンツ管理のワークフローに関し、独自のワークフローアクションプログラムスクリプトを開発、インストール、設定する方法をご説明します。

パート5では、Web サイトコンテンツ管理の編集に関し、独自の Web コンテンツ編集プログラムスクリプトを開発、インストール、設定する方法をご説明します。

パート6では、Web サイト管理機能をカスタマイズ／拡張するための、独自プログラムスクリプトの開発、インストール、設定方法をご説明します。コンテンツ項目の公開・非公開、アップロード済イメージファイルの処理、コンテンツ内容の検証、ユーザアカウント詳細の認証、などに有効です。

パート7では、独自 Web サイトコンテンツ配信スクリプト／テンプレートや、サードパーティ製 Web アプリケーションで、Web コンテンツ管理システムのコンテンツを使用する方法をご説明します。

パート8では、追加のメディアファイルクラウドストレージサービス用の独自のプログラムスクリプトを開発、インストール、構成する方法について説明します。

パート9では、独自のプログラムスクリプトを開発してインストールし、Web コンテンツ管理システムのワンタイムパスワードログイン機能をカスタマイズ/拡張する方法について説明します。

パート 10 では、Web コンテンツ管理システムの“ヘッドレス” REST API を介して Web サイトコンテンツと Web コンテンツ管理機能にアクセスする方法について説明します。

目次

はじめに	4
目次.....	6
1 カスタム / サードパーティ製追加モジュール	14
1.1 インストール及び設定	14
1.2 開発	14
1.3 支払い処理	19
2 カスタム/サードパーティ拡張.....	20
2.1 インストールと設定	20
2.2 開発	20
2.3 使用	21
3 商品の在庫確認・配信 カスタム/サードパーティ拡張	22
3.1 商品在庫確認 カスタム/サードパーティ拡張	22
3.1.1 インストールと設定	22
3.1.2 開発	22
3.2 商品配信 カスタム/サードパーティ拡張	23
3.2.1 インストールと設定	23
3.2.2 開発	23
3.3 使用方法	24
4 ワークフローアクションカスタム/サードパーティ拡張.....	25
4.1 インストールと設定	25
4.2 開発	25
4.3 使用方法	25
5 WEB サイト管理機能拡張.....	26
5.1 インストールと設定	26
5.2 開発	26

5.3	使用方法	26
6	WEB サイト管理機能拡張.....	27
6.1	外部 Web サイト公開／アーカイブ プログラミング API.....	27
6.2	ファイルアップロード プログラミング API.....	27
6.3	コンテンツデータ検証 プログラミング API.....	28
6.4	ユーザデータ認証 プログラミング API.....	29
6.5	外部メディアデジタル資産管理 API.....	29
6.5.1	メディアカテゴリ	29
6.5.2	メディアファイル	30
6.5.3	メディア検索	30
7	WEB アプリケーション統合とテンプレート プログラミング API.....	31
8	クラウドデプロイメントとメディアストレージ	34
8.1	メディアクラウドストレージ API.....	34
8.2	クラウドデプロイメント API.....	34
9	ワンタイムパスワードログイン	36
10	ヘッドレス REST API	37
10.1	概略	37
10.1.1	REST API ログインと承認アクセストークン.....	37
10.1.1.1	HTTP Authorization Bearer ヘッダ	37
10.1.1.2	JSON データパラメータ	37
10.1.1.3	HTML FORM POST パラメータ	37
10.1.1.4	HTTP URL パラメータ	38
10.1.1.5	期限	38
10.1.2	REST API メソッド	38
10.1.3	REST API リクエストパラメータ	38
10.1.4	REST API クッキー	38
10.1.5	URL パラメータを使用した GET リクエスト	39
10.1.6	JSON データパラメータを使用した POST, PUT, DELETE リクエスト.....	39
10.1.7	HTML FORM POST パラメータを使用した POST, PUT, DELETE リクエスト.....	39
10.2	Web サイトコンテンツと機能.....	39
10.2.1	ログイン	39
10.2.2	ページ	41
10.2.3	商品ページ	41
10.2.4	データページ	42
10.2.5	コンテンツ項目	42



10.2.6	データ項目	42
10.2.7	スクリプト	43
10.2.8	スタイルシート	43
10.2.9	イメージ	44
10.2.10	ファイル	44
10.2.11	リンク	44
10.2.12	検索	45
10.2.12.1	Web サイトコンテンツ検索	45
10.2.12.2	コンテンツデータベース検索	45
10.2.13	コンタクト	45
10.2.14	ポスト	46
10.2.14.1	Web サイトコンテンツ	46
10.2.14.2	データベースコンテンツ	47
10.2.15	ユーザ登録	47
10.2.16	ユーザ登録解除	48
10.2.17	ユーザグループ／タイプ入会	48
10.2.18	ユーザグループ／タイプ退会	48
10.2.19	ショッピングカート	49
10.2.20	ヒートマップ	50
10.2.21	パスワード	50
10.2.21.1	パスワード取得	51
10.2.21.2	期限切れパスワード	51
10.2.22	パーソナルページ	52
10.2.22.1	パーソナルページ	52
10.2.22.2	パーソナルページと管理	52
10.2.23	ログアウト	53
10.3	Web サイト管理機能	54
10.3.1	ログイン	54
10.3.2	コンテンツ	54
10.3.2.1	リスト	55
10.3.2.2	検索	56
10.3.2.3	読み取り	57
10.3.2.4	作成	60
10.3.2.5	更新	61
10.3.2.6	削除	63
10.3.3	データ	64
10.3.3.1	リスト	64
10.3.3.2	検索	65
10.3.3.3	読み取り	65
10.3.3.4	作成	66
10.3.3.5	更新	67
10.3.3.6	削除	68
10.3.4	ユーザ	69
10.3.4.1	リスト	69
10.3.4.2	検索	70
10.3.4.3	読み取り	71
10.3.4.4	作成	74
10.3.4.5	更新	75
10.3.4.6	削除	75



10.3.5	マイクロ Web サイト	76
10.3.5.1	リスト	76
10.3.5.2	読み取り	76
10.3.5.3	作成	77
10.3.5.4	更新	77
10.3.5.5	削除	78
10.3.6	構成要素	78
10.3.6.1	リスト	78
10.3.6.2	読み取り	79
10.3.6.3	作成	79
10.3.6.4	更新	80
10.3.6.5	削除	81
10.3.7	コンテンツグループ	81
10.3.7.1	リスト	81
10.3.7.2	読み取り	82
10.3.7.3	作成	82
10.3.7.4	更新	83
10.3.7.5	削除	84
10.3.8	コンテンツタイプ	84
10.3.8.1	リスト	84
10.3.8.2	読み取り	85
10.3.8.3	作成	85
10.3.8.4	更新	86
10.3.8.5	削除	87
10.3.9	イメージフォーマット	87
10.3.9.1	リスト	87
10.3.9.2	読み取り	88
10.3.9.3	作成	88
10.3.9.4	更新	88
10.3.9.5	削除	89
10.3.10	イメージグループ	89
10.3.10.1	リスト	89
10.3.10.2	読み取り	90
10.3.10.3	作成	90
10.3.10.4	更新	91
10.3.10.5	削除	92
10.3.11	イメージタイプ	92
10.3.11.1	リスト	92
10.3.11.2	読み取り	92
10.3.11.3	作成	93
10.3.11.4	更新	94
10.3.11.5	削除	94
10.3.12	ファイルフォーマット	95
10.3.12.1	リスト	95
10.3.12.2	読み取り	95
10.3.12.3	作成	96
10.3.12.4	更新	96
10.3.12.5	削除	97
10.3.13	ファイルグループ	97
10.3.13.1	リスト	97



10.3.13.2	読み取り	97
10.3.13.3	作成	98
10.3.13.4	更新	99
10.3.13.5	削除	99
10.3.14	ファイルタイプ	100
10.3.14.1	リスト	100
10.3.14.2	読み取り	100
10.3.14.3	作成	101
10.3.14.4	更新	102
10.3.14.5	削除	102
10.3.15	リンクグループ	103
10.3.15.1	リスト	103
10.3.15.2	読み取り	103
10.3.15.3	作成	104
10.3.15.4	更新	105
10.3.15.5	削除	105
10.3.16	リンクタイプ	106
10.3.16.1	リスト	106
10.3.16.2	読み取り	106
10.3.16.3	作成	107
10.3.16.4	更新	107
10.3.16.5	削除	108
10.3.17	コンテンツバージョン	109
10.3.17.1	リスト	109
10.3.17.2	読み取り	109
10.3.17.3	作成	109
10.3.17.4	更新	110
10.3.17.5	削除	110
10.3.18	コンテンツバンドル	111
10.3.18.1	リスト	111
10.3.18.2	読み取り	111
10.3.18.3	更新	111
10.3.18.4	削除	112
10.3.19	コンテンツパッケージ	112
10.3.19.1	リスト	112
10.3.19.2	読み取り	112
10.3.19.3	更新	113
10.3.19.4	削除	113
10.3.20	ユーザグループ	113
10.3.20.1	リスト	113
10.3.20.2	読み取り	114
10.3.20.3	作成	114
10.3.20.4	更新	115
10.3.20.5	削除	116
10.3.21	ユーザタイプ	116
10.3.21.1	リスト	116
10.3.21.2	読み取り	117
10.3.21.3	作成	117
10.3.21.4	更新	118
10.3.21.5	削除	118



10.3.22	ユーザセグメント	119
10.3.22.1	リスト	119
10.3.22.2	読み取り	119
10.3.22.3	作成	121
10.3.22.4	更新	122
10.3.22.5	削除	122
10.3.23	ユーザテスト	123
10.3.23.1	リスト	123
10.3.23.2	読み取り	123
10.3.23.3	作成	124
10.3.23.4	更新	124
10.3.23.5	削除	125
10.3.24	ワークフロー	125
10.3.24.1	リスト	125
10.3.24.2	読み取り	125
10.3.24.3	作成	126
10.3.24.4	更新	126
10.3.24.5	削除	127
10.3.25	コメント	127
10.3.25.1	リスト	127
10.3.25.2	読み取り	128
10.3.25.3	作成	128
10.3.25.4	更新	129
10.3.25.5	削除	129
10.3.26	プロジェクト	130
10.3.26.1	リスト	130
10.3.26.2	読み取り	130
10.3.26.3	作成	132
10.3.26.4	更新	132
10.3.26.5	削除	133
10.3.27	プロジェクトタスク	133
10.3.27.1	リスト	134
10.3.27.2	読み取り	134
10.3.27.3	作成	135
10.3.27.4	更新	136
10.3.27.5	削除	137
10.3.28	通貨	137
10.3.28.1	リスト	137
10.3.28.2	読み取り	138
10.3.28.3	作成	138
10.3.28.4	更新	138
10.3.28.5	削除	139
10.3.29	商品グループ	139
10.3.29.1	リスト	139
10.3.29.2	読み取り	140
10.3.29.3	作成	140
10.3.29.4	更新	141
10.3.29.5	削除	142
10.3.30	商品タイプ	142
10.3.30.1	リスト	142



10.3.30.2	読み取り	143
10.3.30.3	作成	143
10.3.30.4	更新	144
10.3.30.5	削除	145
10.3.31	割引	145
10.3.31.1	リスト	145
10.3.31.2	読み取り	146
10.3.31.3	作成	146
10.3.31.4	更新	147
10.3.31.5	削除	147
10.3.32	送料	148
10.3.32.1	リスト	148
10.3.32.2	読み取り	148
10.3.32.3	作成	149
10.3.32.4	更新	149
10.3.32.5	削除	150
10.3.33	税率	150
10.3.33.1	リスト	150
10.3.33.2	読み取り	151
10.3.33.3	作成	151
10.3.33.4	更新	152
10.3.33.5	削除	152
10.3.34	注文	153
10.3.34.1	リスト	153
10.3.34.2	読み取り	153
10.3.34.3	作成	154
10.3.34.4	更新	155
10.3.34.5	削除	155
10.3.35	注文アイテム	156
10.3.35.1	リスト	156
10.3.35.2	読み取り	156
10.3.35.3	作成	157
10.3.35.4	更新	158
10.3.35.5	削除	158
10.3.36	売上	159
10.3.36.1	売上レポート	159
10.3.36.2	リクエストパラメータ	160
10.3.36.3	レポートデータ	160
10.3.37	データベース	161
10.3.37.1	リスト	161
10.3.37.2	読み取り	161
10.3.37.3	作成	162
10.3.37.4	更新	162
10.3.37.5	削除	163
10.3.38	アクセス統計	164
10.3.38.1	アクセス統計値レポート	164
10.3.38.2	リクエストパラメータ	166
10.3.38.3	レポートデータ	167
10.3.39	ホスティングクライアント	168
10.3.39.1	リスト	168



10.3.39.2	読み取り	168
10.3.39.3	作成	169
10.3.39.4	更新	169
10.3.39.5	削除	170
10.3.40	ホスティンググループ	170
10.3.40.1	リスト	170
10.3.40.2	読み取り	171
10.3.40.3	作成	171
10.3.40.4	更新	172
10.3.40.5	削除	173
10.3.41	ホスティングタイプ	173
10.3.41.1	リスト	173
10.3.41.2	読み取り	173
10.3.41.3	作成	174
10.3.41.4	更新	175
10.3.41.5	削除	175
10.3.42	設定	176
10.3.42.1	読み取り	176
10.3.43	Web サイト管理者設定	176
10.3.43.1	読み取り	176
10.3.43.2	更新	178
10.3.44	全体管理者パスワード	179
10.3.44.1	読み取り	179
10.3.45	ログアウト	179

1 カスタム／サードパーティ製追加モジュール

HeartCore Web コンテンツ管理システムでは、独自のカスタム追加モジュールを作成したり、サードパーティ製の追加モジュールを使用したりすることができます。

カスタム／サードパーティ製追加モジュールを HeartCore Web コンテンツ管理システム管理ページと統合し、独自の管理セクションやメニュー項目を既存の管理セクションに追加することができます。

カスタム／サードパーティ製追加モジュールは、E コマース追加モジュールと併せて使用する決済サービスプロバイダモジュールでも可能です。

1.1 インストール及び設定

追加モジュールを追加でインストールするには、Web サーバにモジュールファイルをコピーします。（デフォルトでは、“/webadmin/module/” フォルダ下の新規フォルダです。）

HeartCore Web コンテンツ管理システムは、追加の追加モジュールをロードするよう設定されている必要があります。“/webadmin/module/config.xxx”ファイル（“xxx”は、使用する HeartCore Web コンテンツ管理システムのバージョンにより、“asp”、“jsp”または“php”）を編集し、ファイルの一番下に 1 行追加します。例えば、“example”というモジュールを有効にするには、次の行を追加します。

- .NET:
`<!-- #include file="example/config.aspx" -->`
- JSP:
`<%@ include file=" example /config.jsp" %>`
- PHP:
`<?php include " example /config.php"; ?>`

複数の追加モジュールが設定されている場合、ここでの設定順により Web コンテンツ管理システムのこれらの管理セクションやメニュー項目の表示順が決まります。

1.2 開発

カスタム追加モジュールは、ほぼどのようなものでも構いません。

- HeartCore Web コンテンツ管理システムと密接に統合された追加機能。
- 単一の管理システムから容易にアクセスできるよう、HeartCore Web コンテンツ管理システムと緩く統合された管理と、独自の機能を持った個別アプリケーション。
- 単一の管理システムから容易にアクセスできるよう、HeartCore Web コンテンツ管理システムと緩く統合された管理と、独自の機能を持った外部 Web サービス。

HeartCore Web コンテンツ管理システムは、Web コンテンツ管理システム管理ページに追加セクションやメニュー項目、またコンテンツ管理タブを追加するための、シンプルなインタフェースを定義します。

カスタム追加モジュールには、追加セクションやメニュー項目およびコンテンツ管理タブが Web コンテンツ管理システム管理ページと統合されるかどうか、される場合はどこに、どのように統合されるかを定義するための、モジュール設定ファイルが必要です。

モジュール設定ファイルの詳細については、"/webadmin/module/example/" モジュール例をご参照ください。モジュール設定ファイルの定義は、次の通りです。

- モジュール
モジュール名とバージョン番号
- ツールバーセクション
 - **moduleToolbarTitle**
Web コンテンツ管理システム管理ページのツールバーに表示されるモジュール名。
 - **moduleToolbarImage**
Web コンテンツ管理システム管理ページのツールバーに表示されるモジュールイメージ。
 - **moduleToolbarLink**
そのモジュールの Web コンテンツ管理システム管理ページのツールバー名とイメージのリンク先となる Web アドレス。
- ホーム
 - **moduleHomeIntroTitle**
Web コンテンツ管理システムホームページに表示される、モジュール紹介名。
 - **moduleHomeIntroText**
Web コンテンツ管理システムホームページに表示される、モジュール紹介テキスト。
 - **moduleHomeIntroImage**
Web コンテンツ管理システムホームページに表示される、モジュール紹介イメージ。
 - **moduleHomeIntroLink**
Web コンテンツ管理システムホームページの紹介名、テキスト、画像からのリンク先となる Web アドレス。
 - **moduleHomeMenuTitle**
Web コンテンツ管理システムホームページに表示される、モジュールメニュー項目名。
 - **moduleHomeMenuLink**
Web コンテンツ管理システムホームページのモジュールメニュー項目からリンクされる Web アドレス。
- コンテンツ管理 – Web サイトコンテンツ
 - **moduleContentIntroTitle**
Web コンテンツ管理システムコンテンツ管理ページに表示される、モジュール紹介名。
 - **moduleContentIntroText**
Web コンテンツ管理システムコンテンツ管理ページに表示される、モジュール紹介テキスト。

- **moduleContentMenuTitle**
Web コンテンツ管理システムコンテンツ管理ページに表示される、モジュールメニュー項目名。
- **moduleContentMenuLink**
モジュールのコンテンツ管理メニュー項目からリンクされる Web アドレス。
- **コンテンツ管理 – Web サイトコンテンツ – 管理タブ**
 - **moduleContentTab**
Web コンテンツ管理システムコンテンツ管理ページに使用される、モジュール HTML/Javascript ID。
 - **moduleContentTabTitle**
Web コンテンツ管理システムコンテンツ管理タブに表示される、モジュールタイトル。
 - **moduleContentTabContent**
Web コンテンツ管理システムコンテンツ管理ページタブに初期表示される、モジュール HTML/テキスト コンテンツ。
 - **moduleContentTabURL**
タブが選択された際に、読み込まれて Web コンテンツ管理システムコンテンツ管理ページタブに表示される、HTML/テキスト コンテンツの Web アドレス。
 - **moduleContentTabScript**
タブが選択された際に、Web コンテンツ管理システムコンテンツ管理ページタブで実行される、Javascript コード。
- **ライブラリ管理 – メディアライブラリ**
 - **moduleLibraryIntroTitle**
Web コンテンツ管理システムライブラリ管理ページに表示されるモジュール紹介名。
 - **moduleLibraryIntroText**
Web コンテンツ管理システムライブラリ管理ページに表示されるモジュール紹介テキスト。
 - **moduleLibraryMenuTitle**
Web コンテンツ管理システムライブラリ管理ページに表示されるメニュー項目名。
 - **moduleLibraryMenuLink**
モジュールのライブラリ管理メニュー項目からリンクされる Web アドレス。
- **E コマース管理 – 商品 & 注文 (E コマース追加モジュールのみ)**
 - **moduleEcommerceIntroTitle**
Web コンテンツ管理システム E コマース管理ページに表示されるモジュール紹介名。

- **moduleEcommerceIntroText**
Web コンテンツ管理システム E コマース管理ページに表示されるモジュール紹介テキスト。
- **moduleEcommerceMenuTitle**
Web コンテンツ管理システム E コマース管理ページに表示されるモジュールメニュー項目名。
- **moduleEcommerceMenuLink**
モジュールの E コマース管理メニュー項目からリンクされる Web アドレス。
- **コミュニティ管理（コミュニティ追加モジュールのみ）（将来使用するため）**
 - **moduleCommunityIntroTitle**
Web コンテンツ管理システムコミュニティ管理ページに表示されるモジュール紹介名。
 - **moduleCommunityIntroText**
Web コンテンツ管理システムコミュニティ管理ページに表示されるモジュール紹介テキスト。
 - **moduleCommunityMenuTitle**
Web コンテンツ管理システムコミュニティ管理ページに表示されるモジュールメニュー項目名。
 - **moduleCommunityMenuLink**
モジュールのコミュニティ管理メニュー項目からリンクされる Web アドレス。
- **データベース管理（データベース追加モジュールのみ）（将来使用するため）**
 - **moduleDatabasesIntroTitle**
Web コンテンツ管理システムデータベース管理ページに表示されるモジュール紹介名。
 - **moduleDatabasesIntroText**
Web コンテンツ管理システムデータベース管理ページに表示されるモジュール紹介テキスト。
 - **moduleDatabasesMenuTitle**
Web コンテンツ管理システムデータベース管理ページに表示されるモジュールメニュー項目名。
 - **moduleDatabasesMenuLink**
モジュールのデータベース管理メニュー項目からリンクされる Web アドレス。
- **統計管理（統計追加モジュールのみ）（将来使用するため）**
 - **moduleStatisticsIntroTitle**
Web コンテンツ管理システム統計管理ページに表示されるモジュール紹介名。
 - **moduleStatisticsIntroText**
Web コンテンツ管理システム統計管理ページに表示されるモジュール紹介テキスト。

- **moduleStatisticsMenuTitle**
Web コンテンツ管理システム統計管理ページに表示されるモジュールメニュー項目名。
- **moduleStatisticsMenuLink**
モジュールの統計管理メニュー項目からリンクされる Web アドレス。
- ユーザ管理 – ユーザデータベース
 - **moduleUsersIntroTitle**
Web コンテンツ管理システムユーザ管理ページに表示されるモジュール紹介名。
 - **moduleUsersIntroText**
Web コンテンツ管理システムユーザ管理ページに表示されるモジュール紹介テキスト。
 - **moduleUsersMenuTitle**
Web コンテンツ管理システムユーザ管理ページに表示されるモジュールメニュー項目名。
 - **moduleUsersMenuLink**
モジュールのユーザ管理メニュー項目からリンクされる Web アドレス。
- 設定
 - **moduleConfigIntroTitle**
Web コンテンツ管理システム設定ページに表示されるモジュール紹介名。
 - **moduleConfigIntroText**
Web コンテンツ管理システム設定ページに表示されるモジュール紹介テキスト。
 - **moduleConfigMenuTitle**
Web コンテンツ管理システム設定ページに表示されるモジュールメニュー項目名。
 - **moduleConfigMenuLink**
モジュールの設定メニュー項目からリンクされる Web アドレス。
- ホスティング管理 – ホスティングクライアント（ホスティングエディションのみ）
 - **moduleHostingIntroTitle**
Web コンテンツ管理システムホスティング管理ページに表示されるモジュール紹介名。
 - **moduleHostingIntroText**
Web コンテンツ管理システムホスティング管理ページに表示されるモジュール紹介テキスト。
 - **moduleHostingMenuTitle**
Web コンテンツ管理システムホスティング管理ページに表示されるモジュールメニュー項目名。

- `moduleHostingMenuLink`
モジュールのホスティング管理メニュー項目からリンクされる Web アドレス。
- 支払い処理設定 (E コマース追加モジュールのみ)
 - `modulePaymentTitle`
E コマース追加モジュールの支払い処理設定ページに表示されるモジュール紹介名。
 - `modulePaymentOptions`
E コマース追加モジュールの支払い処理設定ページに表示される支払いサービスプロバイダの紹介テキスト。あらゆる支払い処理オプションをいくつでも設定することができます。含まれている PayPal モジュールを例として参照してください。

全てのモジュール設定ファイル定義は、使用されているかどうかに関わらず、指定する必要がありますのでご注意ください。使用されていないモジュール設定ファイル定義には、""をセットします。

1.3 支払い処理

支払い処理のカスタム／サードパーティ追加モジュールは、追加のプログラムファイルが必要となります。“PayPal”というタイトルの支払い処理用カスタム／サードパーティ追加モジュールには、“/webadmin/module/PayPal/payment.asp”, “/webadmin/module/PayPal/payment.jsp” または “/webadmin/module/PayPal/payment.php” (Web コンテンツ管理システムで使用しているプログラミング言語バージョンに依存します) というプログラムファイルが存在し、これにより実際の支払い処理が行われる必要があります。

完了した注文に関する注文詳細が、支払い処理モジュールでは使用可能です。支払い処理モジュールは、この注文情報を使用して注文の支払い準備や支払い処理を行い、支払い方法や支払い確認を注文完了ショッピングカートやチェックアウトページに表示するため E コマース追加モジュールに送信します。例としては、付属の PayPal モジュールを参照してください。

2 カスタム／サードパーティ拡張

HeartCore Web コンテンツ管理システムでは、独自のカスタム拡張を作成したり、サードパーティ製の拡張を使用したりすることができます。

カスタム／サードパーティ製拡張を HeartCore Web コンテンツ管理システムに統合し、Web コンテンツ管理システム以外の他のアプリケーションやソースから、コンテンツを取り込むことができます。

2.1 インストールと設定

T 追加の拡張をインストールするには、拡張ファイルを Web サーバ（デフォルトでは、"/webadmin/extension/"配下の新規フォルダ）にコピーします。設定は必要ありません。

2.2 開発

カスタム拡張は、ほぼどんなものでも可能です。

- シンプルテキストや HTML ファイル
- コンテンツをプログラマ的に生成する、.NET、JSP、PHP プログラムスクリプト
- ファイルやデータベースからコンテンツを読み込む.NET、JSP、PHP プログラムスクリプト
- 外部 Web サービスからコンテンツを読み込む .NET、JSP、PHP プログラムスクリプト

カスタム拡張は、Web 管理システムの Web コンテンツ内でその特殊コードを使用し、簡単に追加／実行されます。カスタム拡張の出力は、Web 管理システムのコンテンツに表示されます。

シンプルな拡張ファイルの詳細については、"/webadmin/extension/" の拡張例“hello”をご参照ください。拡張ファイルは、単に使用言語にあわせた.NET、JSP あるいは PHP ファイルで、何らかの出力を生成します。パラメータ（もしある場合）は、Web コンテンツ管理システムから拡張へ、“extension”という名前のシンプルなセッション文字列変数を使って渡されます。

- .NET - hello.aspx:
<%@ Page Language="C#" Debug="true" ValidateRequest="false" %>
<%= "Hello " + Session["extension"] %>
- JSP - hello.jsp:
<%= "Hello " + session.getValue("extension") %>
- PHP - hello.php:
<?php echo "Hello " . \$_SESSION["extension"] ?>

“extension”セッションパラメータのほか、標準的な Web サーバリクエスト変数等を拡張で使うことができます。（例えば.NET では“Request”、JSP では“request”、PHP では“\$HTTP_GET_VARS”など。）

通常、カスタム拡張は、入力パラメータを処理し、閲覧ページの一部として表示するためのコンテンツを出力します。

任意で、Web サイトユーザ用に、サーバサイドのクッキーセッション変数をセットすることも可能です。これを行うには、カスタム拡張は、「WCM:SESSION:NAME=VALUE」（「NAME」をセッション変数名と置き換え、「VALUE」をセッション変数値を置き換え）だ

けを出力するよう、特別にフォーマットされたセッションコマンドを出力する／返す必要があります。

任意で、Web サイトユーザを他の Web ページにリダイレクト (Web ブラウザ) もしくはフォワード (Web サーバ) することも可能です。これを行うには、カスタム拡張は、「WCM:REDIRECT:URL」もしくは「WCM:FORWARD:URL」 (「URL」を「http://www.heartcore.co.jp」または「/index.html」などの Web アドレスと置換え) だけを出力するよう、特別にフォーマットされたリダイレクト／フォワードコマンドを出力する／返す必要があります。

2.3 使用

カスタム拡張は、テンプレートなどに使用されるその他の特殊コードに似た、シンプルな特殊コードを使用して、Web コンテンツ管理システムのコンテンツに埋め込まれます。

@@@extension:hello@@@	“hello”という名前の拡張によって生成された出力が挿入される位置を定義します。拡張 “hello” にパラメータは渡されていません。
@@@extension:hello(World)@@@	“hello”という名前の拡張によって生成された出力が挿入される位置を定義します。“World”がパラメータとして、拡張 “hello” に渡されています。

・独自 Extension ファイル作成時の注意点について

Extension ファイルの作成の際は、Extension 内で DB アクセス系の処理を行った場合、末尾に DB.close 処理が必要となります。

DB.close 処理が無い場合、接続されたままの Connection が残り、Connection 数が足りなくなる現象が発生します。

特に「webadmin.jsp」を include する際は必要となりますのでご注意ください。

■ 末尾 DB.close 処理の記述参考例

```
<% if (db != null) db.close(); %><% if (logdb != null) logdb.close(); %>
```

3 商品の在庫確認・配信 カスタム／サードパーティ拡張

HeartCore Web コンテンツ管理システムのEコマース追加モジュールを使用すると、任意の商品配信カスタム拡張を作成したり、サードパーティ製拡張を使用することができます。

商品配信／サードパーティ拡張を HeartCore Web コンテンツ管理システムのEコマース追加モジュールと統合すると、Web コンテンツ管理システム以外のアプリケーションや情報元からのコンテンツを使用することができます。

3.1 商品在庫確認 カスタム／サードパーティ拡張

商品在庫確認カスタム／サードパーティ拡張は、商品が購入可能かどうかを調べるために、商品、ショッピングカート、チェックアウトページにおいて使用することができます。

例えば、自身の外部在庫確認や棚卸システムをチェックし、商品の在庫があるかどうかを調べたり、あるいは業者側の外部在庫確認や棚卸システムをチェックし、商品の在庫があるか、また注文できるかどうかを調べます。また、ユーザ名、メールアドレス、インターネットドメイン名などのような現物のない商品の有効性や、それらがすでに登録済みかどうかを調べます。

3.1.1 インストールと設定

追加の商品配信拡張をインストールするには、Web サーバに拡張ファイルをコピーする必要があります (デフォルトでは、"/webadmin/productavailability/" フォルダ内の新規ファイルです)。設定は必要ありません。

3.1.2 開発

商品在庫確認カスタム拡張は、ほぼ何でも構いません。

- シンプルテキストもしくは HTML ファイル
- ファイルやデータベースからコンテンツを読み込む、.NET、JSP、PHP プログラムスクリプト
- 外部 Web サービスからコンテンツを読み込む、.NET、JSP、PHP プログラムスクリプト

商品在庫確認カスタム拡張は、商品配信カスタム拡張が選択された商品の商品ページ、またはショッピングカート／チェックアウトページが選択された際に実行されます。商品在庫確認カスタム拡張からの出力は、顧客に表示されるページに含むことができます。

サンプル拡張ファイル "/webadmin/productavailability/" の "hello" をご参照ください。

- .NET - hello.aspx:

```
<% Session("productavailability") = "+1 123 In Stock" %>
<% Session("productavailability") = "0 Unavailable" %>
<% Session("productavailability") = "-1 345 Available From Supplier" %>
```
- JSP - hello.jsp:

```
<% session.setValue("productavailability ", "+1 123 In Stock "); %>
<% session.setValue("productavailability ", "0 Unavailable "); %>
<% session.setValue("productavailability ", "-1 345 Available From Supplier "); %>
```
- PHP - hello.php:

```
<?php $_SESSION["productavailability "] = "+1 123 In Stock "; ?>
<?php $_SESSION["productavailability "] = "0 Unavailable "; ?>
<?php $_SESSION["productavailability "] = "-1 345 Available From Supplier "; ?>
```

拡張は"extension"セッションパラメータのほか、一般的な web サーバリクエスト変数などを使用することができます。例えば .NET では "Request.QueryString"、JSP では "request"、PHP の場合は "\$_GET" や "\$_POST" などです。

商品在庫確認カスタム拡張からの出力は、指定のフォーマットである必要があります。数字のあとに空白、そしていくつかの文字が続きます。Web コンテンツ管理システムでは数字は表示されませんが、商品が入手可能かどうかを示します。

- 正数は、商品の在庫があることを表します。商品は注文可能な状況であり、顧客が注文すると即座に料金が課されます。
- 負の数は、商品の在庫が無いことを表します。しかし、入荷待ち注文や先行予約注文などは可能です。注文が完了しても顧客はすぐに料金を課されるわけではありません。（例：後々、注文処理を行う際に、手動や外部システムを通して請求します。）。
- 数字の 0（ゼロ）は、商品が購入不可であることを意味します。その商品を注文することはできず、たとえ注文されたとしても除外され、顧客に料金が課されることはありません。

その他のテキスト出力は、商品、ショッピングカート、チェックアウトページにおいて @@@availability@@@ 特別コードが使用されている場所に表示されます。商品の在庫数を表示する場合は、テキストに数字が繰り返し含まれている必要があります。最初の数字は表示されず、購入可能かどうかを示すためだけに使用されます。

3.2 商品配信 カスタム／サードパーティ拡張

商品配信カスタム／サードパーティ拡張は、自動的にデジタル商品を生成し、商品が注文された際に外部システムを更新するために使用されます。例えば、自身の外部在庫や棚卸システムを更新したり、供給業者へ発注したり、あるいはユーザ名、メールアドレス、インターネットドメイン名などのような現物のない商品を生成／配信する目的で使用されます。

3.2.1 インストールと設定

追加の商品配信拡張をインストールするには、拡張ファイルは Web サーバへ（デフォルトでは、"/webadmin/productdelivery/" フォルダ下に新規ファイルとして）コピーされる必要があります。特に設定などは必要ありません。

3.2.2 開発

商品配信カスタム拡張は、ほぼ何でも構いません。

- シンプルテキストまたは HTML ファイル
- コンテンツをプログラマ的に生成する .NET、JSP、PHP プログラムスクリプト
- コンテンツをファイルやデータベースから読み込む、.NET、JSP、PHP プログラムスクリプト
- 外部 Web サービスからコンテンツを読み込む、.NET、JSP、PHP プログラムスクリプト

商品配信拡張は、この商品配信拡張が選択されている商品の注文と支払いが正常に完了した際に、単純に組み込まれ実行されます。商品配信拡張の出力は商品配信ページに組み込まれ顧客に表示されるか、顧客に送信される E メールに組み込まれます。

簡単な商品配信拡張ファイルについては、"hello"という拡張例"/webadmin/productdelivery/"を参照してください。この拡張ファイルはお使いの環境が使用するプログラミング言語に応じた.NET、JSP、あるいはPHPファイルで、なんらかの出力結果を生成します。

- .NET - hello.aspx:
`<% Session("productdelivery") = "Hello " %>`
- JSP - hello.jsp:
`<% session.setValue("productdelivery", "Hello"); %>`
- PHP - hello.php:
`<?php $_SESSION["productdelivery"] = "Hello"; ?>`

拡張は"extension"セッションパラメータのほか、一般的なWebサーバリクエスト変数などを使用することができます。例えば.NETでは"Request.QueryString"、JSPでは"request"、PHPの場合は"\$_GET"や"\$_POST"などです。

3.3 使用方法

Webコンテンツ管理システムの「商品&注文」セクションにて、商品の追加や更新を行う際に、「商品の出荷」タブにある「カスタムエクステンションプログラム」入力フィールドより、商品在庫確認／配信カスタム拡張を選択します。

4 ワークフローアクションカスタム／サードパーティ拡張

HeartCore Web コンテンツ管理システムでは、独自のワークフローアクションカスタム拡張を作成したり、サードパーティ製の拡張を使用することができます。

ワークフローアクションカスタム／サードパーティ拡張を、HeartCore Web コンテンツ管理システムと連携すると、Web コンテンツ管理システム以外のアプリケーションを使って、コンテンツの管理アクションを作成および記録することが可能になります。

4.1 インストールと設定

追加のワークフローアクション拡張をインストールするには、拡張ファイルを Web サーバーへ（デフォルトでは、"/webadmin/workflowaction/" フォルダ内の新規ファイルとして）コピーする必要があります。特に設定などは必要ありません。

4.2 開発

ワークフローアクションカスタム拡張は、ほぼ何でも構いません。

- ファイルやデータベースからコンテンツを読み込む、またはファイルやデータベースへコンテンツを書き込む、.NET、JSP、PHP プログラムスクリプト
- 外部 Web サービスからコンテンツを読み込む、または外部 Web サービスへコンテンツを書き込む、.NET、JSP、PHP プログラムスクリプト

ワークフローアクションカスタム拡張は、このワークフローアクションカスタム拡張が設定されているワークフローアクションが、コンテンツ項目や E コマースモジュールの注文に適用された際に、実行されます。

"/webadmin/workflowaction/" サンプル拡張の "Workflow Action Module Template" と "Test Workflow Action" をご覧いただくと、シンプルなワークフローアクション拡張ファイルについての詳細をご参照いただけます。"Workflow Action Module Template" は、Web コンテンツ管理システムからどのように関連データが読み込まれるかを示すものです。"Test Workflow Action" は、実行されたワークフローアクションを、コンテンツ項目の「変更履歴」へ記録します。

このファイルは、お使いのプログラミング言語に応じた、.NET、JSP、PHP のシンプルなファイルで、サンプル拡張でご覧いただける通り、Web コンテンツ管理システムから関連ワークフローおよびコンテンツや注文を読み込み、更新します。

4.3 使用方法

Web コンテンツ管理システムの「設定」管理セクションにて、ワークフローアクションの追加や更新を行う際に、「コンテンツの更新」にある「カスタムエクステンションプログラム」入力フィールドより、ワークフローアクションカスタム拡張を選択します。

5 Web サイト管理機能拡張

HeartCore Web コンテンツ管理システムでは、独自の Web コンテンツエディタカスタム拡張を作成したり、サードパーティ製の拡張を使用することができます。

Web コンテンツエディタカスタム／サードパーティ拡張を、HeartCore Web コンテンツ管理システムと連携すると、Web コンテンツ管理システムに含まれている HeartCore Web コンテンツエディタ以外の Web コンテンツエディタを使用することが可能になります。

5.1 インストールと設定

追加の Web コンテンツエディタ拡張をインストールするには、拡張ファイルを Web サーバーへ（デフォルトでは、"/webadmin/webeditors/EXTENSION NAME/"フォルダ内の新規ファイルとして）コピーする必要があります。特に設定などは必要ありません。

5.2 開発

Web コンテンツエディタカスタム拡張は、.NET、JSP、PHP プログラムスクリプトであり、Web コンテンツ管理システムのコンテンツ管理に必要な Javascript 機能を定義しています。

この Web コンテンツエディタカスタム拡張は、Web コンテンツ管理システムの各管理ページからコンテンツ項目や E コマースモジュールの商品を、追加あるいは更新する際に、組み込まれます。

Web コンテンツエディタ拡張ファイルについての詳細は、"/webadmin/webeditors/"にある"WebEditorAPI" サンプル拡張をご参照ください（デフォルトの HeartCore Web コンテンツエディタ拡張は、"HardCore"、"HardCore1"、"HardCore2"、"textarea" です）。

このファイルは、お使いのプログラミング言語に応じた、.NET、JSP、PHP のシンプルなファイルで、Web コンテンツ管理システムのコンテンツ管理に必要な Javascript 機能を定義し、またカスタム／サードパーティ Web コンテンツエディタに必要な他のプログラムスクリプトを読み込みます。

5.3 使用方法

Web コンテンツ管理システムの「設定」>「機能設定」>「コンテンツ編集」>「コンテンツエディタ」管理ページにて、Web コンテンツエディタカスタム拡張を選択します。そうすると、コンテンツ項目や E コマースモジュールの注文を追加あるいは更新する際に、その Web コンテンツエディタカスタム拡張が自動的に使用されます。

6 Web サイト管理機能拡張

HeartCoreWeb コンテンツ管理システムは、Web サイト上もしくは Web コンテンツ管理ページ上で特定の機能が使用される際、独自カスタム“/webadmin/api/”プログラムスクリプトを通じた Web サイト管理機能のカスタマイズ／拡張をサポートしています。

6.1 外部 Web サイト公開／アーカイブ プログラミング API

デフォルトでは、HeartCoreWeb コンテンツ管理システムは実際の Web サイト上で動作し、動的に Web サイトコンテンツを配信します。しかし、Web コンテンツ管理システム内で、例えば、他の Web サーバやアーカイブ／バックアップサーバにファイルをコピーするなど、コンテンツ項目が公開される際、または非公開される際に独自のプログラムスクリプトを実行させるようにプログラムしたい場合があります。

“静的ファイル名”を持つ新規もしくは更新済コンテンツ項目が公開されると、Web コンテンツ管理システムは、“/webadmin/api/published”、“/webadmin/api/published.bat”、

“/webadmin/api/published.sh” ファイルが存在するかをチェックし、公開されたコンテンツ項目の静的ファイル名をパラメータとして実行します。

“静的ファイル名”を持つコンテンツ項目が非公開にされると、Web コンテンツ管理システムは“/webadmin/api/unpublished”、“/webadmin/api/unpublished.bat”、

“/webadmin/api/unpublished.sh” ファイルが存在するかをチェックし、公開されたコンテンツ項目の静的ファイル名をパラメータとして実行します。

6.2 ファイルアップロード プログラミング API

デフォルトでは、HeartCoreWeb コンテンツ管理システムはアップロードされたイメージとその他のファイルを、コンテンツ項目として Web コンテンツ管理システムに追加します。しかし、イメージまたはファイルが Web コンテンツ管理システムにアップロードされる際に、ファイルのウィルスチェックをしたり、他のフォーマットやサイズにファイルを変換するための独自のプログラムを実行したい場合があります。

“イメージ” または “ファイル” がアップロードされる際、Web コンテンツ管理システムは、“/webadmin/api/image”、“/webadmin/api/image.bat”、“/webadmin/api/image.sh”、

“/webadmin/api/file”、“/webadmin/api/file.bat”、“/webadmin/api/file.sh” のファイルが存在するかをチェックし、アップロードされたファイルのファイル名をパラメータとして実行します。プログラムスクリプトは、プログラムスクリプトが何をするか、また Web コンテンツ管理システムが何をなすかによって、以下を出力／返します。

- パラメータとしてプログラムスクリプトに渡されたファイル名
プログラムスクリプトが、アップロード済ファイルをリネーム、移動、または削除していない場合
- アップロードされたファイルの新しいファイル名
プログラムスクリプトがアップロード済ファイルをリネーム、または移動していた場合。
Web コンテンツ管理システムは、コンテンツ項目を新しいファイル名で更新します。
- 何もしない
プログラムスクリプトがアップロード済ファイルを削除していた場合。Web コンテンツ管理システムもまた、コンテンツ項目を削除します。

“イメージ” または “ファイル” がアップロードされると、Web コンテンツ管理システムは以下のファイルが存在するかチェックします。

“/webadmin/api/image1”, “/webadmin/api/image1.bat”, “/webadmin/api/image1.sh”,
“/webadmin/api/image2”, “/webadmin/api/image2.bat”, “/webadmin/api/image2.sh”,
“/webadmin/api/image3”, “/webadmin/api/image3.bat”, “/webadmin/api/image3.sh”,
“/webadmin/api/file1”, “/webadmin/api/file1.bat”, “/webadmin/api/file1.sh”, “/webadmin/api/file2”,
“/webadmin/api/file2.bat”, “/webadmin/api/file2.sh”, “/webadmin/api/file3”, “/webadmin/api/file3.bat” or
“/webadmin/api/file3.sh”

そして、アップロードされたファイルのファイル名をパラメータとして実行します。プログラムスクリプトは、プログラムスクリプトが何をするか、また Web コンテンツ管理システムが何をなすかによって、以下を出力／返します。

- アップロード済ファイルの新しい、代替コピーのファイル名
例えば、イメージの低い解像度のバージョン、または、Microsoft Word ドキュメントの PDF バージョン、または、プログラムファイルの圧縮バージョンなど。Web コンテンツ管理システムは、新しいファイルに対して追加のコンテンツ項目を作成します。元のアップロードされたファイルのコンテンツ項目の対応する追加コンテンツ/イメージ 1/イメージ 2/イメージ 3/ファイル 1/ファイル 2/ファイル 3 の属性は、新しい、代替ファイルのコンテンツ項目を指します。新しい、代替ファイルのコンテンツ項目の関連コンテンツ/ページアップ属性は、元のアップロードされたファイルのコンテンツ項目を指します。
- 何もしない
プログラムスクリプトが、アップロード済ファイルの新しい、代替コピーを作成していない場合。Web コンテンツ管理システムは、追加のコンテンツ項目を作成しません。

6.3 コンテンツデータ検証 プログラミング API

Web サイトコンテンツに対し特別な要件がある場合、コンテンツが Web コンテンツ管理システムに保存される時／前に検証されるよう、独自のプログラムスクリプトをプログラムすることが可能です。プログラムスクリプト“/webadmin/api/validatecontent.xxx” (“xxx” を該当のプログラミング言語拡張子に置き換える: “aspx”, “jsp” “php”) が存在する場合、コンテンツが追加または更新されて保存される際に、実行されます。コンテンツのデータは、標準 HTML POST フォームデータとしてプログラムスクリプトへポストされます。

そうすると、ポストされたコンテンツ項目データは検証され、プログラムスクリプトが構造化レスポンスを Web コンテンツ管理システムへ返します。

- “OK”
コンテンツ項目データは OK、コンテンツは保存されます。
- “OK:ALERT:MESSAGE”
コンテンツ項目データは OK、コンテンツは保存されます。“MESSAGE”が Web サイト管理者へ表示されます (“MESSAGE”を独自の文字列に置き換える)。
- “ERROR:CONFIRM:MESSAGE”
コンテンツ項目データに問題がある可能性があり、Web サイト管理者はコンテンツを保存するか、キャンセルするか、再編集するか確認するよう“MESSAGE”で促されます (“MESSAGE”を独自の文字列に置き換える)。
- “ERROR:ALERT:MESSAGE”
コンテンツ項目データに問題があり、コンテンツは保存されません。“MESSAGE”が Web サイト管理者へ表示され、コンテンツを保存するには編集し直す必要があります (“MESSAGE”を独自の文字列に置き換える)。



6.4 ユーザデータ認証 プログラミング API

Web サイトユーザアカウントに対し特別な要件がある場合、ユーザデータが Web コンテンツ管理システムに保存される時／前に検証されるよう、独自のプログラムスクリプトをプログラムすることが可能です。プログラムスクリプト“/webadmin/api/validateuser.xxx” (“xxx” を該当のプログラミング言語拡張子に置き換える: “aspx”, “jsp” “php”) が存在する場合、ユーザアカウントが追加または更新されて保存される際、また Web サイト上でのユーザ登録際に実行されます。ユーザアカウントのデータは、標準 HTML POST フォームデータとしてプログラムスクリプトへポストされます。

そうすると、ポストされたユーザアカウントデータは検証され、プログラムスクリプトが構造化レスポンスを Web コンテンツ管理システムへ返します。

- “OK”
ユーザアカウントデータは OK、ユーザアカウントは保存されます。
- “OK:ALERT:MESSAGE”
ユーザアカウントデータは OK、ユーザアカウントは保存されます。“MESSAGE”が Web サイト管理者へ表示されます (“MESSAGE”を独自の文字列に置き換える)。Web サイトユーザ登録では、“MESSAGE”が Web サイトユーザへ表示されます。
- “ERROR:CONFIRM:MESSAGE”
ユーザアカウントデータに問題がある可能性があり、Web サイト管理者はユーザアカウントを保存するか、キャンセルするか、再編集するか確認するよう“MESSAGE”で促されます (“MESSAGE”を独自の文字列に置き換える)。Web サイトユーザ登録では、“MESSAGE”が Web サイトユーザへ表示されます。
- “ERROR:ALERT:MESSAGE”
ユーザアカウントデータに問題があり、ユーザアカウントは保存されません。“MESSAGE”が Web サイト管理者へ表示され、ユーザアカウントを保存するには編集し直す必要があります (“MESSAGE”を独自の文字列に置き換える)。Web サイトユーザ登録では、“MESSAGE”が Web サイトユーザへ表示されます。

6.5 外部メディアデジタル資産管理 API

別のデジタル資産管理システムを使用して、Web コンテンツ管理システムの内外で使用したいメディアを管理している場合は、独自のプログラムスクリプトをプログラムすることで、外部メディアストレージ、およびサービスからメディアカテゴリとファイルにアクセス、リストすることができます。

“/webadmin/api/media.xxx” (“xxx” をプログラミング言語の拡張機能 “aspx”、“jsp”、または “php” に置き換えます) プログラムスクリプトが存在する場合、Web コンテンツエディタがメディアを挿入したときに実行されます。ハイパーリンクの挿入ツールバー機能が使用され、プログラムスクリプトによってリストされたメディアカテゴリ (フォルダ) とファイルは、メディアの挿入およびハイパーリンクの挿入ポップアップウィンドウで使用できます。

6.5.1 メディアカテゴリ

プログラムスクリプトにパラメータが渡されない場合、メディアカテゴリ (フォルダ) 名のリストがリストされます (改行区切り)。

- Aaa
- Bbb
- Ccc
-
- Xxx
- Yyy
- Zzzz

リストされたメディアカテゴリ（/フォルダ）名は、[メディアの挿入]ポップアップウィンドウの左側のパネルから利用できます。

6.5.2 メディアファイル

カテゴリ（/フォルダ）名がパラメータとしてプログラムスクリプトに渡されると、そのメディアカテゴリ（/フォルダ）名のリストが、メディアファイル名と URL のリストとしてリストされます（改行区切り、メディアファイル名と URL はタブ区切り）。

- Aaaaa http://somedialibrary.net/abc/aaaaa.gif
- Bbbbb http://somedialibrary.net/abc/logo.png
- Ccccc http:// somedialibrary.net/abc/zzzzz.png
-
- Xxxxx http://somedialibrary.net/abc/xxx123.png
- Yyyyy http://somedialibrary.net/abc/xyz/xxx123.png
- Zzzzz http://somedialibrary.net/xyz/z789.png

リストされたメディアファイル名は、[メディアの挿入]および[ハイパーリンクの挿入]ポップアップウィンドウの中央パネルから利用できます。

6.5.3 メディア検索

検索パラメータ（任意でカテゴリ（/フォルダ）名パラメータ）がプログラムスクリプトに渡されると、それは Web サイト管理者が入力した 1 つ以上の検索語となり、一致するメディアファイル名と URL のリストがリストされます（改行区切り、メディアファイル名と URL はタブ区切り）。

- Aaaaa http://somedialibrary.net/abc/aaaaa.gif
- Bbbbb http://somedialibrary.net/abc/logo.png
- Ccccc http:// somedialibrary.net/abc/zzzzz.png
-
- Xxxxx http://somedialibrary.net/abc/xxx123.png
- Yyyyy http://somedialibrary.net/abc/xyz/xxx123.png
- Zzzzz http://somedialibrary.net/xyz/z789.png

リストされたメディアファイル名は、[メディアの挿入]および[ハイパーリンクの挿入]ポップアップウィンドウの中央パネルから利用できます。

7 Web アプリケーション統合とテンプレートプログラミング API

デフォルトでは、すべてのコンテンツやテンプレートなどは Web コンテンツ管理システムで作成・提供されていて、プログラミングは必要ありません。しかし、特別な要件によっては独自のテンプレートをプログラムしたり、任意のサードパーティアプリケーションと Web コンテンツ管理システムとのコンテンツを統合したい場合があります。簡単に使用できる Web コンテンツ管理システムプログラミング API を使用して、Web コンテンツ管理システムのコンテンツにアクセスすることができます。

Web コンテンツ管理システムプログラミング API 機能は、"/webadmin.xxx" ファイルにあり、これを Web アプリケーションで使用します。Web コンテンツ管理システムプログラミング API の機能は次の通りです。

- **ReadContent(id)**
指定した id のコンテンツ項目を読み込みます。
- **ReadPage(id)**
指定した id のページを、そのページのテンプレートや追加コンテンツ項目も含めて読み込みます。
- **ReadProduct(id)**
指定した id の商品を、その商品のテンプレートや追加コンテンツ項目も含めて読み込みます。
- **ReadData(database, id)**
指定したデータベース名のカスタムコンテンツデータベースから、指定された id を持つデータを読み込みます。
- **ContentHeader(id, attribute)**
指定した id のコンテンツ項目の指定された属性値を返します。
- **PageDOCTYPE(id)**
指定した id ページの DOCTYPE HTML コードを、その HTML ページのトップへ出力します。
- **PageHeader(id, attribute, "")**
HTML HEAD 出力に対し、指定した id を持つページの指定属性名の値を返します。
- **PageHeader(id, element, attribute)**
HTML HEAD 出力に対し、指定した id ページの指定コンテンツエレメントの指定属性名の値を返します。
- **PageStyleSheet(id)**
HTML HEAD 出力に対し、指定した id を持つページのスタイルシート HTML コードを返します。
- **ProductDOCTYPE(id)**
指定した id 商品の DOCTYPE HTML コードを、その HTML ページのトップへ出力します。
- **ProductHeader(id, attribute, "")**
HTML HEAD 出力に対し、指定した id を持つ商品の指定属性名の値を返します。

- **ProductHeader(id, element, attribute)**
HTML HEAD 出力に対し、指定した id を持つ商品の指定エレメントクラス名の追加コンテンツエレメントの指定属性名の値を返します。
- **ProductStyleSheet(id)**
HTML HEAD 出力に対し、指定された id の商品のスタイルシート HTML コードを返します。
- **StyleSheetHeader()**
HTML スタイルシートのヘッダを出力します("Content-Type: text/css")。
- **DisplayContent(id, attributename)**
HTML BODY 出力に対し、指定された id のページの指定された属性名の値を返します。
- **DisplayPage(id, attribute, "")**
HTML BODY 出力に対し、指定した id ページの指定属性名の値を返します。
- **DisplayPage(id, element, attribute)**
HTML BODY 出力に対し、指定した id ページの指定エレメントクラス名の追加コンテンツエレメントの指定属性名の値を返します。
- **DisplayProduct(id, attribute, "")**
HTML BODY 出力に対し、指定した id を持つ商品の指定属性名の値を返します。
- **DisplayProduct(id, element, attribute)**
HTML BODY 出力に対し、指定した id 商品の指定エレメントクラス名の追加コンテンツエレメントの指定属性名の値を返します。
- **OutputContent(content)**
コンテンツで特殊コード "@@@extension:...@@" を実行し、プログラム拡張の出力結果とコンテンツを返します。
- **CMSHeader(id)**
指定した id ページの閲覧&編集モードの HTML ヘッダコードを返します。
- **CMSTDisplay(id)**
指定した id ページの閲覧&編集モードの Web ページヘッダコードを返します。
- **CMSSStyleSheet(id)**
指定した id ページの閲覧&編集モードのスタイルシート Web ページヘッダコードを返します。
- **CMSTemplate(id)**
指定した id ページの閲覧&編集モードのテンプレート Web ページヘッダコードを返します。
- **CMSLog(id, class, "")**
指定した id とクラス名のコンテンツ項目へのリクエストを記録します。
- **CMSLog(id, "data", database)**
指定したデータベース名のカスタムコンテンツデータベース内の、指定された id とクラス名を持つコンテンツ項目へのリクエストを記録します。



Web コンテンツ管理システムのプログラミング API 機能がどのように使用されているかの例としては、Web コンテンツ管理システムプログラムファイル"/page.xxx"、"/product.xxx"、"/data.xxx" を参照してください。

8 クラウドデプロイメントとメディアストレージ

8.1 メディアクラウドストレージ API

デフォルトでは、HeartCore Web コンテンツ管理システムは、イメージやファイルを Web サイトのサーバーへ保管します。Web サイトと HeartCore Web コンテンツ管理システムが、複数サーバーのクラスタ上で動作している場合は、すべてのサーバーでイメージやファイルが使用できるよう、共有ファイルストレージやミラーリング／複製ファイルストレージにて保管するようセットアップが必要です。代わりに、クラウドストレージサービス（もしくはその他の共有ストレージサービス）を使用することも可能です。

クラウドストレージの設定は、「設定」>「システム」>「ウェブサイト」>「メディアストレージ」>「クラウドストレージ」にて行います。クラウドストレージを使用するよう設定されている場合、イメージやファイルをアップロード、コピー、移動／リネーム、削除、ダウンロードする際に、様々な“/webadmin/api/” プログラムスクリプトが実行されます。

- /webadmin/api/exists.xxx” (“xxx” は、“aspx”、“jsp”、“php”のいずれか)
指定したイメージ／ファイルがクラウドストレージに存在するか識別します。
- /webadmin/api/upload.xxx” (“xxx” は、“aspx”、“jsp”、“php”のいずれか)
Web サイトへアップロードされた新しいイメージ／ファイルを、クラウドストレージへアップロードします。
- /webadmin/api/copy.xxx” (“xxx” は、“aspx”、“jsp”、“php”のいずれか)
コピーされたイメージ／ファイルを、クラウドストレージでコピーします。
- /webadmin/api/move.xxx” (“xxx” は、“aspx”、“jsp”、“php”のいずれか)
移動／リネームされたイメージ／ファイルを、クラウドストレージで移動／リネームします。
- /webadmin/api/delete.xxx” (“xxx” は、“aspx”、“jsp”、“php”のいずれか)
削除されたイメージ／ファイルを、クラウドストレージから削除します。
- /webadmin/api/download.xxx” (“xxx” は、“aspx”、“jsp”、“php”のいずれか)
変更されたもしくは存在しないイメージ／ファイルがアクセスされると、クラウドストレージからダウンロードします。

これらの“/webadmin/api/” プログラムスクリプトは、Web コンテンツ管理システムに含まれており、様々なクラウドストレージサービスプロバイダに対応しています。また、プログラムコードにはプレースホルダーコメントが記載されており、独自もしくは他のクラウドストレージサービスプロバイダ向けに、カスタムプログラミングを追加することも可能です。詳細は、各“/webadmin/api/” プログラムスクリプトをご確認ください。

メディアクラウドストレージ API は、単独で使用するほか、クラウドデプロイメント API と併用することもできます（クラウドデプロイメント API の使用には、メディアクラウドストレージ API が必須です）。詳しくは、8.2 クラウドデプロイメント API をご参照ください。

8.2 クラウドデプロイメント API

HeartCore Web コンテンツ管理システムは、自社サーバーにインストールするほか、ホスティングサービスプロバイダやクラウドホスティングサービスを使用することもできます。

Web コンテンツ管理システムをクラウド上にインストールし、ウェブサーバーの追加による動的な拡張性を持たせる場合、新しいウェブサーバーが追加された際に、設定したデータベースサーバーおよびメディアクラウドストレージを自動で検出し接続させる必要があるでしょう。

HeartCore Web コンテンツ管理システムでは、様々なクラウドホスティングサービスおよびデータベースサーバーが標準でサポートされており、データベースサーバーの自動検出や自動接続に対応しています。標準でサポートされていない他のクラウドホスティングサービスやデータベースを使用するには、設定したデータベースサーバーを自動で検出し、接続文字列を自動生成するためのプログラムコードを、独自で追加する必要があります。Web コンテンツ管理システムは、生成された接続文字列を使って、データベースサーバーおよびメディアクラウドストレージ設定へ接続します。

クラウドデプロイメント検出、データベース接続、メディアクラウドストレージは、`"/config.cloud.xxx"`（`xxx` は、`"aspx"`、`"jsp"`、`"php"`のいずれか）という特別な設定プログラムスクリプトによって処理されます。他のクラウドホスティングサービスプロバイダやデータベースサーバーに対応するには、このプログラムスクリプトを開いて、下記が可能となるよう、独自のプログラムコードを追加します。

- サーバー環境変数からクラウドデプロイメント設定を検出できるようにします。もしくは、他の何らかの方法によって、クラウドホスティングサービスがクラウドデプロイメント設定を検出できるようにします。
- `"database"` を、設定したデータベースサーバーの接続文字列にセットします。管理画面の「設定」>「システム」>「データベース」>「データベース接続」での入力と同じ要領です。
- `"database_init"` 設定を、SQL コマンドにセットします。データベースサーバーの作成や初期化に必要となります。
- `"csservice"` 設定を、サポートされているメディアクラウドストレージサービスプロバイダのいずれかにセットします。もしくは、他のクラウドストレージサービスプロバイダに固有の ID や名前にセットします。メディアクラウドストレージ API のプログラムスクリプトで使用しているプロバイダと同じです。
- `"csusername"`、`"cspassword"`、`"csrootpath"`、`"csURLrootpath"` 設定を、それぞれ適切な値へセットします。管理画面の「設定」>「システム」>「ウェブサイト」>「メディアストレージ」での入力と同じ要領です。

プログラミング API 機能の使用例については、`"/config.cloud.xxx"`（`xxx` は、`"aspx"`、`"jsp"`、`"php"`のいずれか）プログラムファイルをご確認ください。

クラウドデプロイメント API は、メディアファイルを保管するメディアクラウドストレージ API と併用する必要があります。詳しくは、8.1 メディアクラウドストレージ API をご参照ください。

9 ワンタイムパスワードログイン

HeartCore Web コンテンツ管理システムは、独自の“/webadmin/api”カスタムプログラムスクリプトを使った、ワンタイムパスワードログイン機能のカスタマイズ／拡張に対応しています。

デフォルトでは、Web コンテンツ管理システムが生成したランダムなワンタイムパスワードコードの使用とウェブサイトユーザ／管理者への E メール送信、サードパーティアプリやデバイスを使った標準 RFC 6238 Time-Based One-Time Password に準拠したコード、に対応しています。

カスタムログインスクリプトを使用すると、Web コンテンツ管理システムが生成したワンタイムパスワードコードを、E メール以外の方法でウェブサイトユーザ／管理者へ送信することができます。

また、カスタムログインスクリプトを使用すると、独自のワンタイムパスワードコードを生成して、E メールや他の方法でウェブサイトユーザ／管理者へ送信することができます。

ワンタイムパスワードコードが生成されウェブサイトユーザ／管理者に送信されると、Web コンテンツ管理システムは、“/webadmin/api/login”、“/webadmin/api/login.bat”、“/webadmin/api/login.sh”、“/webadmin/api/login.xxx”（“xxx”は、“aspx”、“jsp”、“php”のいずれか）のいずれかのファイルが存在するかチェックし、ウェブサイトユーザ／管理者の情報と生成されたワンタイムパスワードコードで実行します。

プログラムスクリプトおよび Web コンテンツ管理システムが何を行うかに応じて、プログラムスクリプトの返却／出力は以下のようになります。

- プログラムスクリプトにパラメータとして渡されたコードと同じワンタイムパスワードコード：
プログラムスクリプトがそのワンタイムパスワードコードをウェブサイトユーザ／管理者に送信している場合。
- 新しいワンタイムパスワードコード：
プログラムスクリプトが新しいワンタイムパスワードコードを生成して、ウェブサイトユーザ／管理者に送信している場合。
- なし：
プログラムスクリプトが何も行っていない場合。Web コンテンツ管理システムは独自に生成したワンタイムパスワードコードを使用し、ウェブサイトユーザ／管理者に送信します。

10 ヘッドレス REST API

HeartCoreWeb コンテンツ管理システムは、REST API 関数を介したサードパーティ、カスタムコンテンツ配信、および管理システムからの“ヘッドレス”アクセスをサポートします。REST API は、公開された Web サイトのコンテンツと機能、および Web サイト管理機能の両方で使用できます。

10.1 概略

10.1.1 REST API ログインと承認アクセストークン

Web サイト管理機能の REST API と、公開された Web サイトコンテンツの REST API にアクセスするには、Web コンテンツ管理システムへログインし、承認アクセストークンを取得する必要があります。

10.1.1.1 HTTP Authorization Bearer ヘッダ

発行された承認アクセストークンは、すべてのタイプのリクエスト（GET、POST、PUT、DELETE）の HTTP Authorization Bearer ヘッダとして REST API リクエストで提供されることをお勧めします。

HTTP Authorization Bearer ヘッダ
GET /webadmin/rest/content/?id=123 Authorization: Bearer XXXXX.YYYYY.ZZZZZ

10.1.1.2 JSON データパラメータ

あるいは、発行された承認アクセストークンは JSON データパラメータとして提供される必要があります。

JSON データパラメータ
POST /webadmin/rest/content/ Content-Type: application/json { “access_token”: “XXXXX.YYYYY.ZZZZZ”, “id”: “123”, }

10.1.1.3 HTML FORM POST パラメータ

あるいは、発行された承認アクセストークンは HTML FORM POST パラメータとして提供される必要があります。

HTML FORM POST パラメータ
<form action="/webadmin/rest/content/" method="POST"> <input type="hidden" name="access_token" value="XXXXX.YYYYY.ZZZZZ"> <input type="hidden" name="id" value="123"> </form>

10.1.1.4 HTTP URL パラメータ

発行された承認アクセストークンは HTML URL パラメータとしても提供できますが、このようなリクエストはキャッシュやログとして記録され、許可されていないユーザが発行された承認アクセストークンと REST API にアクセスできてしまう可能性があるため、セキュリティ上の理由からお勧めしません。

HTTP URL パラメータ
GET /webadmin/rest/content/?id=ID&access_token=XXXXX.YYYYY.ZZZZZ

10.1.1.5 期限

発行された承認アクセストークンは、Web コンテンツ管理システムにて設定された一定の期間が経過すると期限切れになる場合があります。

期限切れの承認アクセストークンを使用した REST API リクエストは、“HTTP 401 Unauthorized” エラーによって失敗します。新しい REST API ログインリクエストにて、新しい承認アクセストークンが必要となります。

10.1.2 REST API メソッド

REST API リクエストには、GET、POST、UPDATE、DELETE メソッドが提供されています。すべてのメソッドがすべての REST API 関数でサポートされているわけではないことに注意してください。

GET メソッドは、Web コンテンツ管理システムからコンテンツ／データを読み取るために使用されます。

POST メソッドは、Web コンテンツ管理システムでの新しいコンテンツ／データの作成や、ログインなどに使用されます。

UPDATE メソッドは、Web コンテンツ管理システムの既存のコンテンツ／データを更新するために使用されます。

DELETE メソッドは、Web コンテンツ管理システムから既存のコンテンツ／データを削除するために使用されます。

10.1.3 REST API リクエストパラメータ

REST API リクエストは、要求されたコンテンツ／データを識別、フィルタリングするために、また他のコンテンツ／データ属性のために、さまざまなパラメータを必要とする場合があります。パラメータは、REST API 関数と使用／必須メソッドに応じて、さまざまな方法で指定できます。

10.1.4 REST API クッキー

一部の REST API 関数は、一部のデータを保存、および転送するために Cookie を使用場合があります。たとえば、コンテンツ配信機能は、エクスペリエンス管理機能の “usersegments”、“usertests”、“usertesttags” の Cookie を読み取り／設定します。また、ショッピングカート機能は、ショッピングカートコンテンツの “shopcart”、“discount” の Cookie を読み取り／設定します。

10.1.5 URL パラメータを使用した GET リクエスト

REST API GET リクエストは、コンテンツ／データを識別し、コンテンツ／データ属性を読み取るためのさまざまなパラメータが必要な場合があります。GET リクエストのパラメータは、標準の Web ブラウザの URL アドレスと同様に、URL パラメータとして指定する必要があります。

GET リクエスト URL パラメータ
GET /rest/page/?id=ID&version=VERSION

10.1.6 JSON データパラメータを使用した POST, PUT, DELETE リクエスト

REST API POST、PUT、DELETE リクエストは、コンテンツ／データを識別し、コンテンツ／データ属性を作成／更新／削除するためのさまざまなパラメータが必要な場合があります。POST、PUT、および DELETE のリクエストパラメータは、JSON データとして指定できます。

JSON データパラメータ
<pre>POST /webadmin/rest/content/ Content-Type: application/json { "title": "abc", "content": "xyz", }</pre>

10.1.7 HTML FORM POST パラメータを使用した POST, PUT, DELETE リクエスト

あるいは、JSON パラメータの代わりに、POST、PUT、DELETE リクエストのパラメータを HTML FORM データとして指定できます。ただし、Web ブラウザは HTML FORM GET、POST リクエストのみをサポートし、HTML FORM PUT、DELETE リクエストはサポートしない場合があることに注意してください。

HTML FORM POST パラメータ
<pre><form action="/webadmin/rest/content/" method="POST"> <input type="text" name="title" value="abc"> <input type="text" name="content" value="xyz"> </form></pre>

10.2 Web サイトコンテンツと機能

公開された Web サイトコンテンツと機能は、“http://www.yourwebsite.com/rest/xxxxx/” REST API 関数を介して利用できます。基本的に、これらの REST API 関数は、さまざまな “http://www.yourwebsite.com/xxxxx.aspx|jsp|php” プログラムスクリプトと同じコンテンツ、および機能へのアクセスを提供します。

10.2.1 ログイン

通常、REST API Web サイトコンテンツ機能は、ログインを要求するように設定する必要があります。必要に応じて、REST API Web サイトコンテンツ機能を公開に設定できます。この場合、ログインは不要で、REST API 承認アクセストークンは必要ありません。

REST API Web サイト管理者のログインが必要な場合は、Web コンテンツシステムのユーザデータベースで Web サイト管理者の有効なユーザ名とパスワードを提供する必要があります。REST API Web サイトユーザのログインが必要な場合は、Web コンテンツシステムのユーザデータベースに登録されている Web サイトユーザ、または Web サイト管理者の有効なユーザ名とパスワードを提供する必要があります。

任意で、API キーも必要になる場合があります。

有効な REST API ログインへのレスポンスは、JSON データ形式の承認アクセストークンになります。承認アクセストークンは、他の REST API のリクエストで提供する必要があります。

また、Web コンテンツ管理システムが Web サイトユーザにメール送信（または送信）されたランダムに生成されたワンタイムパスワードを使用するように構成されている場合、REST API 関数はログインコード検証用の “login_code” Cookie を読み取り/設定します。このような場合：最初に、ユーザ名とパスワードのパラメータを使用してログインします。失敗すると、“login_code” Cookie が返され、ログインコードが Web サイトユーザにメール送信（または送信）されます。次に、ユーザ名とパスワードのパラメータ、およびメール送信されたワンタイムパスワードコード（および最初の失敗したログインで返された “login_code” Cookie）を含むコードパラメータを使用して、再度ログインします。

POST /rest/login/	
username	Web コンテンツ管理システムの設定に応じた登録済み Web サイトユーザ、または Web サイト管理者のユーザ名
password	Web コンテンツ管理システムの設定に応じた登録済み Web サイトユーザ、または Web サイト管理者のパスワード
code	Web コンテンツ管理システムの設定に応じた登録済み Web サイトユーザ、または Web サイト管理者のパスコード
apikey	Web コンテンツ管理システムに設定された API キー（設定されている場合のみ）
Response	
<pre>{ "refresh_token": "XXXXXXXXXX.YYYYYYYYYY.ZZZZZZZZZZ", "access_token": "XXXXXXXXXX.YYYYYYYYYY.ZZZZZZZZZZ" }</pre>	

ログイン用にアクセストークンに加えてリフレッシュトークンが提供されている場合、ユーザ名/パスワード/コードの代わりにリフレッシュトークンを使用して、アクセストークンの有効期限が切れたときに新しいアクセストークンをリクエストできます。通常、リフレッシュトークンは、アクセストークン（分/時間）よりもかなり長い時間（時間/日）の間有効です。

POST /rest/login/	
refresh_token	ログイン時に提供された有効なリフレッシュトークン
apikey	Web コンテンツ管理システムにて設定された API キー（設定されている場合）
Response	
<pre>{ "access_token": "XXXXXXXXXX.YYYYYYYYYY.ZZZZZZZZZZ" }</pre>	

10.2.2 ページ

“/rest/page/” REST API 関数は、テンプレートなどを含む完全に解析された Web ページを JSON データとして個々の HTML ページパーツに分割して返します。

GET /rest/page/?id=ID GET /rest/page/?id=ID&version=VERSION	
id	ページの ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "doctype": "<!DOCTYPE html>", "html": "", "head": "", "metainfo": "<meta name=\"language\" content=\"EN\">", "author": "", "description": "What we do.", "keywords": "The homepage for My Business.", "title": "My Business", "stylesheets": "<link rel=\"stylesheet\" type=\"text/css\" href=\"/stylesheet.jsp?id=1\">\r\n\r\n", "scripts": "", "htmlheader": "", "contentschema": "", "htmlbodyonload": "", "body": "<div id=\"wrapper_outer\">\r\n <div id=\"wrapper_inner\">\r\n </div>\r\n </div>" }</pre>	

10.2.3 商品ページ

“/rest/product/” REST API 関数は、テンプレートなどを含む完全に解析された商品の Web ページを JSON データとして個々の HTML ページパーツに分割して返します。

GET /rest/product/?id=ID GET /rest/product/?id=ID&version=VERSION	
id	商品の ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "doctype": "<!DOCTYPE html>", "html": "", "head": "", "metainfo": "<meta name=\"language\" content=\"EN\">", "author": "", "description": "What we do.", "keywords": "The homepage for My Business.", "title": "My Business", "stylesheets": "<link rel=\"stylesheet\" type=\"text/css\" href=\"/stylesheet.jsp?id=1\">\r\n\r\n", "scripts": "", "htmlheader": "", "contentschema": "", "htmlbodyonload": "", "body": "<div id=\"wrapper_outer\">\r\n <div id=\"wrapper_inner\">\r\n </div>\r\n </div>" }</pre>	

10.2.4 データページ

“/rest/data/” REST API 関数は、テンプレートなどを含む完全に解析されたデータの Web ページを JSON データとして個々の HTML ページパーツに分割して返します。

GET /rest/data/?database=DATABASE&id=ID GET /rest/data/?database=DATABASE&id=ID&version=VERSION	
database	コンテンツデータベース名
id	コンテンツデータベース項目の ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "doctype": "<!DOCTYPE html>", "html": "", "head": "", "metainfo": "<meta name=\"language\" content=\"EN\">", "author": "", "description": "What we do.", "keywords": "The homepage for My Business.", "title": "My Business", "stylesheets": "<link rel=\"stylesheet\" type=\"text/css\" href=\"/stylesheet.jsp?id=1\">\r\n\r\n", "scripts": "", "htmlheader": "", "contentschema": "", "htmlbodyonload": "", "body": "<div id=\"wrapper_outer\">\r\n <div id=\"wrapper_inner\">\r\n </div>\r\n </div>" }</pre>	

10.2.5 コンテンツ項目

“/rest/contentitem/” REST API 関数は、テンプレートなどを除く、コンテンツ項目の解析されたプライマリコンテンツのみを返します。

GET /rest/contentitem/?id=ID GET /rest/contentitem/?id=ID&version=VERSION	
id	コンテンツ項目の ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "content": "<div id=\"wrapper_outer\">\r\n <div id=\"wrapper_inner\">\r\n </div>\r\n </div>" }</pre>	

10.2.6 データ項目

“/rest/dataitem/” REST API 関数は、テンプレートなどを除く、コンテンツデータベース項目の解析されたプライマリコンテンツのみを返します。

GET /rest/dataitem/?database=DATABASE&id=ID GET /rest/dataitem/?database=DATABASE&id=ID&version=VERSION	
database	コンテンツデータベース名
id	コンテンツ項目の ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	

```
{
  "content": "<div id=\"wrapper_outer\">\r\n  <div
id=\"wrapper_inner\">\r\n    .....    </div>\r\n  </div>"
}
```

10.2.7 スクリプト

“/rest/script/” REST API 関数は、スクリプトコンテンツ項目の解析されたプライマリコンテンツのみを返します。

GET /rest/script/?id=ID GET /rest/script/?id=ID&version=VERSION	
id	スクリプトの ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "content": "function show(click_id,showchar,hidechar){\r\n document.getElementById(click_id).style.display = '';\r\n document.getElementById(showchar).style.display = 'none';\r\n document.getElementById(hidechar).style.display = '';\r\n\r\n }\r\n\r\n function hide(click_id,showchar,hidechar){\r\n document.getElementById(click_id).style.display = 'none';\r\n document.getElementById(hidechar).style.display = 'none';\r\n document.getElementById(showchar).style.display = '';\r\n }\r\n" }</pre>	

10.2.8 スタイルシート

“/rest/stylesheet/” REST API 関数は、スタイルシートコンテンツ項目の解析されたプライマリコンテンツのみを返します。

GET /rest/stylesheet/?id=ID GET /rest/stylesheet/?id=ID&version=VERSION	
id	スタイルシートの ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "content": "html, body, div, span, applet, object, iframe,\r\n\th1, h2, h3, h4, h5, h6, p, blockquote, pre,\r\n\ta, abbr, acronym, address, big, cite, code,\r\n\t del, dfn, em, font, img, ins, kbd, q, s, samp,\r\n\tsmall, strike, strong, tt, var,\r\n\tb, u, i, center,\r\n\t dl, dt, dd, ol,\r\n\t fieldset, form, label, legend,\r\n\t table, caption, tbody, tfoot, thead, tr, th, td{\r\n\t\tmargin:0;\r\n\t\tpadding:0;\r\n\t\tborder:0;\r\n\t\toutline:0;\r\n\t\tverti cal-align:baseline; \r\n}\r\n\r\n\r\n\ttable{\r\n\t\tfont- size:100%;\r\n}\r\n\r\n\r\n\tbody{\r\n\t\twidth:100%;\r\n\t\tfont-size:80%;\r\n\t\t/* Minimum width of layout - remove line if not required */\r\n\t\tmin-width:600px; \r\n\t\tline-height:1.2;\r\n\t\tfont-family:Arial, Helvetica, sans- serif;\r\n\t\tcolor:#0A0A0A;\r\n\t\tbackground:#FFFFFF; \r\n}\r\n\r\n\r\n\t/* NO LIST MARKERS. */\r\n\tol, ul{\r\n\t\tlist-style-position:inside;\r\n\t\tlist-style- type:disc;\r\n}\r\n\r\n\r\n\t/* FOCUS STYLES */\r\n\r\n\t:focus{\r\n\t\toutline:0;\r\n}\r\n\r\n\r\n\t/* tables still need 'cellspacing="0"' in the markup */\r\n\ttable{\r\n\t\tborder- collapse:collapse;\r\n\t\tborder-spacing:0;\r\n}\r\n\r\n\r\n\t/* Adjusts web content editor toolbar drop-down images. */\r\n\ttable.webeditor_toolbar img{\r\n\t\tvertical-align:bottom; \r\n}\r\n\r\n\r\n\t/* Creates frame around editor. */\r\n\t.editorframe{\r\n\t\tborder:1px solid #CCC;\r\n}\r\n\r\n\r\n\t.editorframe td{\r\n\t\tpadding:5px;\r\n}\r\n" }</pre>	

10.2.9 イメージ

“/rest/image/” REST API 関数は、指定されたイメージの ID の URL を返します。Web コンテンツ管理システムのメディアコンテンツ配信が実際のイメージファイルにリダイレクトするように設定されている場合、返される URL は実際のイメージファイルになります。Web コンテンツ管理システムのメディアコンテンツ配信がプログラムスクリプト “/image.xxx” を介してイメージデータを返すように構成されている場合、返される URL は、パラメータとしてイメージの ID を使用したコンテンツ配信プログラムスクリプトとなります。

GET /rest/image/?id=ID GET /rest/contentitem/?id=ID&version=VERSION	
id	イメージの ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "url": "http://www.yourwebsite.com/image/logo.gif" }</pre>	

10.2.10 ファイル

“/rest/file/” REST API 関数は、指定されたファイルの ID の URL を返します。Web コンテンツ管理システムのメディアコンテンツ配信が実際のファイルにリダイレクトするように設定されている場合、返される URL は実際のファイルになります。Web コンテンツ管理システムのメディアコンテンツ配信がプログラムスクリプト “/file.xxx” を介してファイルデータを返すように構成されている場合、返される URL は、パラメータとしてファイルの ID を使用したコンテンツ配信プログラムスクリプトとなります。

GET /rest/file/?id=ID GET /rest/file/?id=ID&version=VERSION	
id	ファイルの ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "url": "\file.jsp?id=123" }</pre>	

10.2.11 リンク

“/rest/link/” REST API 関数は、指定されたリンクの ID の URL を返します。

GET /rest/link/?id=ID GET /rest/link/?id=ID&version=VERSION	
id	リンクの ID
version	コンテンツバージョン名
device	コンテンツデバイス名
Response	
<pre>{ "url": "http://www.somewebsite.com/" }</pre>	

10.2.12 検索

“/rest/search/” REST API 関数は、Web サイトのコンテンツまたはコンテンツデータベースの検索結果を JSON データとして返します。

10.2.12.1 Web サイトコンテンツ検索

“database” パラメータが指定されていない場合、公開された Web サイトのコンテンツが検索されます。一致する Web サイトコンテンツ項目のタイトル、作成者、ディスクリプション、キーワード、コンテンツクラス、および id 属性のデータが返されます。

GET /rest/search/?search=SEARCH	
search	検索ワード（必須）
Response	
<pre>[{ "title": "ABC", "author": "", "description": ".....", "keywords": "", "contentclass": "page", "id": "123" }, { "title": "XYZ", "author": "", "description": ".....", "keywords": "", "contentclass": "page", "id": "789" }]</pre>	

10.2.12.2 コンテンツデータベース検索

“database” パラメータが指定されている場合、そのコンテンツデータベースが検索されます。コンテンツデータベース項目のタイトルと id 属性のデータが返されます。

GET /rest/search/?database=DATABASE&search=SEARCH	
search	検索ワード（必須）
database	コンテンツデータベース名（必須）
Response	
<pre>[{ "title": "ABC", "id": "123" }, { "title": "XYZ", "id": "789" }]</pre>	

10.2.13 コンタクト

“/rest/contact/” REST API 関数は、コンタクトフォームデータを Web サイト管理者にメールで送信します。

この REST API 関数は、“/contact.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「4.1 コンタクトフォーム」セクションをご覧ください。

POST リクエストは、空のレスポンスを返します。

POST /rest/contact/	
from	E メールアドレス
to	E メールアドレス
cc	E メールアドレス
bcc	E メールアドレス
subject	E メールの件名
.....	コンタクトフォームデータ
Request	
<pre>{ "from": "johndoe@example.com", "to": "help@yourwebsite.com", "cc": "archive@yourwebsite.com", "subject": ".....", "message": "....." }</pre>	
Response	
<pre>{ }</pre>	

10.2.14 ポスト

“/rest/post/” REST API 関数は、コンテンツを Web サイトまたはコンテンツデータベースに投稿します。

10.2.14.1 Web サイトコンテンツ

この REST API 関数は、“/post.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「7.5 ユーザ投稿コンテンツ」セクションをご覧ください。

POST リクエストは、作成されたコンテンツを返します。

POST /rest/post/	
id	コンテンツ項目の ID (必須)
.....	コンテンツデータ属性
publish	“yes” (任意)
ready_to_publish	“yes” (任意)
email_template	コンテンツ項目の ID (任意)
Request	
<pre>{ "id": "123", , "publish": "yes", "email_template": "456" }</pre>	
Response	
<pre>{ "content": "....." }</pre>	

10.2.14.2 データベースコンテンツ

この REST API 関数は、“/post.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「8.1.2 データベースコンテンツのポスト」セクションをご覧ください。

POST リクエストは、作成されたコンテンツデータベースのデータを返します。

POST /rest/post/	
database	コンテンツデータベース名（必須）
id	コンテンツデータベースレコードの ID（任意）
.....	コンテンツデータベースのデータ属性
email_template	コンテンツ項目の ID（任意）
email_notification	“yes”（任意）
Request	
<pre>{ "database": "XYZ", "id": "123", "email_template": "456", "email_notification": "yes" }</pre>	
Response	
<pre>{ "database": "XYZ", "id": "789", }</pre>	

10.2.15 ユーザ登録

“/rest/register/” REST API 関数は、Web サイト訪問者を新しいユーザアカウントとして登録します。

この REST API 関数は、“/register.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「7.2 ユーザ登録」をご覧ください。

POST リクエストは、ユーザ登録のステータスを含む「エラー」メッセージを返します（成功した場合は“[[register.created]]”）。

POST /rest/register/	
user_id	ユーザアカウントの ID
content_id	コンテンツ項目の ID
from	E メールアドレス
cc	E メールアドレス
bcc	E メールアドレス
email	E メールアドレス
username	ユーザ名
password	パスワード
name	名前
.....	ユーザデータ
Request	
<pre>{ "user_id": "1", "email": "johndoe@example.com", }</pre>	

<pre> "username": "johndoe", "password": "secret", "name": "John Doe", } </pre>
Response
<pre> { "error": "[[register.created]]" } </pre>

10.2.16 ユーザ登録解除

“/rest/unregister/” REST API 関数は、ログイン中の Web サイトユーザのユーザアカウントを非アクティブ化/削除します。

POST リクエストは空のレスポンスを返します。

POST /rest/unregister/	
Request	
{	
}	
Response	
{	
}	

10.2.17 ユーザグループ/タイプ入会

“/rest/subscribe/” REST API 関数は、Web サイトユーザのユーザアカウントを特定のユーザグループまたはユーザタイプに割り当てます。

この REST API 関数は、“/subscribe.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「7.4 ユーザグループ/タイプ入会」セクションをご覧ください。

POST リクエストは空のレスポンスを返します。

POST /rest/subscribe/	
usergroup	ユーザグループ名
usertype	ユーザタイプ名
Request	
{	
"usergroup": "ABC"	
}	
Request	
{	
"usertype": "XYZ"	
}	
Response	
{	
}	

10.2.18 ユーザグループ/タイプ退会

“/rest/unsubscribe/” REST API 関数は、Web サイトユーザのユーザアカウントを特定のユーザグループまたはユーザタイプの割り当てを解除します。

この REST API 関数は、“/unsubscribe.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「7.4 ユーザグループ/タイプ入会」セクションをご覧ください。

POST リクエストは、空のレスポンスを返します。

POST /rest/unsubscribe/	
usergroup	ユーザグループ名
usertype	ユーザタイプ名
Request	
{ "usergroup": "ABC" }	
Request	
{ "usertype": "XYZ" }	
Response	
{ }	

10.2.19 ショッピングカート

“/rest/shopcart/” REST API 関数は、ショッピングカート機能进行处理します。

この REST API 関数は、“/shopcart.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「9.2 特殊リンク」セクションをご覧ください。

POST リクエストは、テンプレートなどを含む完全に解析されたショッピングカート Web ページを JSON データとして個々の HTML ページパーツに分割して返します。

REST API 関数は、ショッピングカートコンテンツの “shopcart” と “discount” の Cookie も読み取り/設定します。

POST /rest/shopcart/	
add	商品の ID
ID	商品の ID の数量
checkout	“checkout”
confirm	“confirm”
complete	“complete”
Request – 商品 id=123 をショッピングカートに追加する	
{ "add": "123" }	
Request – 商品 id=123 の数量を「2」に設定する	
{ "123": "2" }	
Request – 注文する	
{ "complete": "complete" }	
Response	
{ "doctype": "<!DOCTYPE html>",&br/> "html": "", "head": "",	

```

    "metainfo": "",
    "author": "",
    "description": "",
    "keywords": "",
    "title": "Shopping Cart",
    "stylesheets": "<link rel=\"stylesheet\" type=\"text/css\"
href=\"\"/stylesheet.jsp?id=1\">\r\n\r\n",
    "scripts": "",
    "htmlheader": "",
    "contentschema": "",
    "htmlbodyonload": "",
    "body": "<div id=\"wrapper_outer\">\r\n    ....    </div>"
}

```

10.2.20 ヒートマップ

“/rest/heatmap/” REST API 関数はヒートマップアクションをログに記録します。

この REST API 関数は、“/heatmap.xxx” プログラムスクリプト機能を反映しています。

POST リクエストは、ログに記録されたデータを返します。

POST /rest/heatmap/	
id	ヒートマップの ID
action	ヒートマップアクション (“click”, “mouse”, “fold”, “reach”)
x	ヒートマップアクション x 位置
y	ヒートマップアクション y 位置
w	ヒートマップアクション 幅
h	ヒートマップアクション 高さ
usersegments	ユーザセグメントデータ
usertests	ユーザテストデータ
Request	
<pre> { "id": "123", "action": "click", "x": "535", "y": "63", "w": "800", "h": "600", "usersegments": "", "usertests": "" } </pre>	
Response	
<pre> { "id": "123", "action": "click", "logged": "2020-12-31 23:59:59", "x": "535", "y": "63", "w": "800", "h": "600", "usersegments": "", "usertests": "" } </pre>	

10.2.21 パスワード

“/rest/password/” REST API 関数は、Web サイトユーザのパスワードを取得して更新します。

10.2.21.1 パスワード取得

この REST API 関数は、“/password/index.xxx” プリグラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「5.8 パスワード取得ページ」セクションをご覧ください。

POST リクエストは、テンプレートなどを含む完全に解析されたパスワード取得ページを JSON データとして個々の HTML ページパーツに分割して返します。

POST /rest/password/	
email	E メールアドレス (email か username が必須)
username	Web サイトユーザアカウントのユーザ名 (email か username が必須)
from	E メールアドレス (任意)
cc	E メールアドレス (任意)
bcc	E メールアドレス (任意)
Request	
<pre>{ "username": "johndoe", "email": "johndoe@example.com", "from": "password@yourwebsite.com", "cc": "support@yourwebsite.com", "bcc": "archive@yourwebsite.com" }</pre>	
Response	
<pre>{ "doctype": "<!DOCTYPE html>", "html": "", "head": "", "metainfo": "", "author": "", "description": "", "keywords": "", "title": "Forgot your login details?", "stylesheets": "<link rel=\"stylesheet\" type=\"text/css\" href=\"/stylesheet.jsp?id=1\">\r\n\r\n", "scripts": "", "htmlheader": "", "contentschema": "", "htmlbodyonload": "", "body": "<div id=\"wrapper_outer\">\r\n </div>" }</pre>	

10.2.21.2 期限切れパスワード

この REST API 関数は、“/password/expired.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「5.12 パスワード期限切れページ」セクションをご覧ください。

PUT リクエストは、空のレスポンスまたはエラーメッセージを返します。

PUT /rest/password/	
old_password	ログイン中の Web サイトユーザの既存のパスワード (必須)
new_password	ログイン中の Web サイトユーザの新しいパスワード (必須)
new_password2	ログイン中の Web サイトユーザの新しいパスワード (必須)

Request
<pre>{ "old_password": "secret", "new_password": "alsosecret", "new_password2": "alsosecret" }</pre>
Response
<pre>{ }</pre>

10.2.22 パーソナルページ

“/rest/personal/” REST API 関数は、Web サイトユーザのパーソナルページを取得して更新します。

10.2.22.1 パーソナルページ

この REST API 関数は、“/personal/?USERNAME” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「7.3 パーソナルページと管理」セクションをご覧ください。

GET リクエストは、テンプレートなどを含む完全に解析されたパーソナルページを JSON データとして個々の HTML ページパーツに分割して返します。

GET /rest/personal/?USERNAME GET /rest/personal/?username=USERNAME GET /rest/personal/?id=USERNAME	
username	Web サイトユーザアカウントのユーザ名（必須）
Response	
<pre>{ "doctype": "<!DOCTYPE html>", "html": "", "head": "", "metainfo": "", "author": "", "description": "", "keywords": "", "title": "John Smith", "stylesheets": "<link rel=\"stylesheet\" type=\"text/css\" href=\"/stylesheet.jsp?id=1\">\r\n\r\n", "scripts": "", "htmlheader": "", "contentschema": "", "htmlbodyonload": "", "body": "<div id=\"wrapper_outer\">\r\n </div>" }</pre>	

10.2.22.2 パーソナルページと管理

この REST API 関数は、“/personal/admin.xxx” プログラムスクリプト機能を反映しています。詳細については、ウェブサイト開発者ガイドの「7.3 パーソナルページと管理」セクションをご覧ください。

PUT リクエストは、ログイン中の Web サイトユーザのパーソナルコンテンツと設定、個人情報を更新します。

PUT リクエストは、テンプレートなどを含む完全に解析されたパーソナルページを JSON データとして個々の HTML ページパーツに分割して返します。

PUT /rest/personal/	
title	パーソナルページのタイトル
content	パーソナルページのコンテンツ
template	テンプレートのコンテンツ項目の ID
stylesheet	スタイルシートのコンテンツ項目の ID
personal_ELEMENT_content	構成要素のコンテンツ項目の ID
publish	“publish”
contentclass	“page”
.....	個人情報
Request	
<pre>{ "title": "ABC" "content": "XYZ" "template": "123" "stylesheet": "456" "personal_news_content": "789" "email": "johndoe@example.com", "name": "John Doe", "password": "secret", "title": "ABC", "publish": "publish" }</pre>	
Response	
<pre>{ "doctype": "<!DOCTYPE html>", "html": "", "head": "", "metainfo": "", "author": "", "description": "", "keywords": "", "title": "John Smith", "stylesheets": "<link rel=\"stylesheet\" type=\"text/css\" href=\"/stylesheet.jsp?id=1\">\r\n\r\n", "scripts": "", "htmlheader": "", "contentschema": "", "htmlbodyonload": "", "body": "<div id=\"wrapper_outer\">\r\n </div>" }</pre>	

10.2.23 ログアウト

“/rest/logout/” REST API 関数は、ログイン中の Web サイトユーザアカウントをログアウトします。

POST リクエストは、空のレスポンスを返します。

POST /rest/logout/	
Request	
<pre>{ }</pre>	
Response	
<pre>{ }</pre>	

10.3 Web サイト管理機能

全ての Web サイト管理機能は、“<http://www.yourwebsite.com/webadmin/rest/xxxxx/>” REST API 関数を介して利用できます。基本的に、これらの REST API 関数は、さまざまな “<http://www.yourwebsite.com/webadmin/xxxxx/>” Web コンテンツシステム管理ページと同じコンテンツ、および機能へのアクセスを提供します。

10.3.1 ログイン

REST API Web サイト管理機能はログインを必要とします。

Web コンテンツシステムのユーザデータベースで、Web サイト管理者の有効なユーザ名とパスワードを入力する必要があります。

任意で、API キーも必要になる場合があります。

有効な REST API ログインへのレスポンスは、JSON データ形式の承認アクセストークンになります。承認アクセストークンは、他の REST API のリクエストで提供する必要があります。

任意で、リフレッシュトークンが提供されている場合、リフレッシュトークンを使用して、新しい承認アクセストークンをリクエストできます。

POST /webadmin/rest/login/	
username	Web サイト管理者のユーザ名
password	Web サイト管理者のパスワード
code	Web コンテンツ管理システムの設定に応じて、Web サイト管理者のワンタイムパスコード
apikey	Web コンテンツ管理システムに設定された API キー（設定されている場合のみ）
Response	
<pre>{ "refresh_token": "XXXXXXXXXX.YYYYYYYYYY.ZZZZZZZZZZ", "access_token": "XXXXXXXXXX.YYYYYYYYYY.ZZZZZZZZZZ" }</pre>	

ログイン用にアクセストークンに加えてリフレッシュトークンが提供されている場合、ユーザ名/パスワード/コードの代わりにリフレッシュトークンを使用して、アクセストークンの有効期限が切れたときに新しいアクセストークンをリクエストできます。通常、リフレッシュトークンは、アクセストークン（分/時間）よりもかなり長い時間（時間/日）の間有効です。

POST /webadmin/rest/login/	
refresh_token	ログイン時に提供された有効なリフレッシュトークン
apikey	Web コンテンツ管理システムにて設定された API キー（設定されている場合）
Response	
<pre>{ "access_token": "XXXXXXXXXX.YYYYYYYYYY.ZZZZZZZZZZ" }</pre>	

10.3.2 コンテンツ

“<http://www.yourwebsite.com/webadmin/rest/content/>” REST API 関数は、1 つ以上のコンテンツ項目のすべてのデータを返し、コンテンツ項目を作成、更新、削除します。

10.3.2.1 リスト

“id”、“archive”、および“search”パラメータが指定されていない場合、他の指定されたパラメータと一致するコンテンツ項目のリストが返されます。

GET /webadmin/rest/content/?contentclass=CONTENTCLASS&contentgroup=CONTENTGROUP&contenttype=CONTENTTYPE&contentbundle=CONTENTBUNDLE&contentpackage=CONTENTPACKAGE&version=VERSION&status=STATUS&stock=STOCK&pagesize=PAGESIZE&offset=OFFSET&sort_col=ATTRIBUTE&sort_dir=DESC	
contentclass	コンテンツクラス名 (“page”, “template”, “stylesheet”, “script”, “image”, “file”, “link”, “product”、または設定されたコンテンツ構成要素クラス名)
contentgroup	コンテンツグループ名、または“-” (なし)
contenttype	コンテンツタイプ名、または“-” (なし)
contentbundle	コンテンツバンドル名
contentpackage	コンテンツパッケージ名
version	コンテンツバージョン名、または“-” (なし)
status	コンテンツステータスコード (“new”, “updated”, “scheduled”, “published”, “unpublished”, “expiring”, “expired”, “checkedout”)
stock	商品在庫状況コード (“unlimited”, “in”, “low”, “no”, “orderable”, “ordered”)
pagesize	リストするコンテンツ項目最大数 デフォルトでは、一致するすべてのコンテンツ項目がリストされます。
offset	リストされるコンテンツ項目の最初のコンテンツ項目番号 デフォルトでは、一致するすべてのコンテンツ項目がリストされます。 例： 最初の 10 個のコンテンツ項目 (1-10) “offset=0&pagesize=10” 次の 10 個のコンテンツ項目 (11-20) “offset=10&pagesize=10” 次の 10 個のコンテンツ項目 (21-30) “offset=20&pagesize=10”
sort_col	リストされたコンテンツ項目をソートするためのコンテンツ項目属性名 (“id”, “title”, “created”, “updated”, “published” など)
sort_dir	“ASC”、または“DESC” リストされたコンテンツ項目を昇順、または降順でソートします。デフォルトでは、コンテンツ項目は昇順でソートされます。
Response	
<pre>[{ "id": "123", "title": "ABC" } , , {</pre>	

```

    "id": "789",
    .....
    "title": "XYZ"
  }
]
```

10.3.2.2 検索

“search” パラメータが指定されている場合、“search” パラメータと指定した他のすべてのパラメータに一致するコンテンツ項目の検索結果リストが返されます。

GET /webadmin/rest/content/?search=SEARCH&id=ID,ID,ID§ion=SECTION &class=CONTENTCLASS&group=CONTENTGROUP&type=CONTENTTYPE&bundle=CONTENT BUNDLE&package=CONTENTPACKAGE&version=VERSION&device=DEVICE&usersegment=U SERSEGMENT&usertest=USERTEST&status=STATUS&filename=FILENAME&attribute=ATTRIB UTE&search_datetime=DATETIME&search_datetime_attribute=DATETIMEATTRIBUTE&pagesize =PAGESIZE&offset=OFFSET&sort_col=ATTRIBUTE&sort_dir=DESC	
search	検索ワード
id	コンテンツ項目の ID（複数の場合はカンマ区切り）
section	コンテンツセクション名（“content”, “page”, “template”, “stylesheet”, “script”, “library”, “image”, “file”, “link”, “ecommerce”, “product”、または設定されたコンテンツ構成要素クラス名）
class	コンテンツクラス名（“page”, “template”, “stylesheet”, “script”, “image”, “file”, “link”, “product”、または設定されたコンテンツ構成要素クラス名）
group	コンテンツグループ名、または“-”（なし）
type	コンテンツタイプ名、または“-”（なし）
bundle	コンテンツバンドル名
package	コンテンツパッケージ名
version	コンテンツバージョン名、または“-”（なし）
device	コンテンツデバイス名
usersegment	コンテンツユーザセグメント名
usertest	コンテンツユーザテスト名
status	コンテンツステータスコード（“new”, “updated”, “scheduled”, “published”, “unpublished”, “expiring”, “expired”, “checkedout”, “workflow”（空白以外のワークフローステータス）、または設定されたワークフローステータス名）
filename	コンテンツファイル名、またはコンテンツファイル名の一部
attribute	検索するコンテンツ属性名（“title”, “content”, “summary”, “description”, “keywords”, “metainfo”, “htmlheader”, “product_info”, “product_code”, “product_price”, “product_cost”, “product_weight”, “product_width”, “product_height”, “product_depth”, “product_volume”, “product_brand”, “product_colour”, “product_size”） カンマ区切り、指定しない場合は全属性が対象
search_datetime	日付/時刻フォーマット文字列、または部分文字列

search_datetime_attribute	“created”, “updated”, “published”, “unpublished”, “scheduled”, “expiring”, “deleted”、指定しない場合は全属性が対象
pagesize	リストするコンテンツ項目最大数 デフォルトでは、一致するすべてのコンテンツ項目がリストされます。
offset	リストされるコンテンツ項目の最初のコンテンツ項目番号 デフォルトでは、一致するすべてのコンテンツ項目がリストされます。 例： 最初の 10 個のコンテンツ項目 (1-10) “offset=0&pagesize=10” 次の 10 個のコンテンツ項目 (11-20) “offset=10&pagesize=10” 次の 10 個のコンテンツ項目 (21-30) “offset=20&pagesize=10”
sort_col	リストされたコンテンツ項目をソートするためのコンテンツ項目属性名 (“id”, “title”, “created”, “updated”, “published” など)
sort_dir	“ASC”、または “DESC” リストされたコンテンツ項目を昇順、または降順でソートします。デフォルトでは、コンテンツ項目は昇順でソートされます。
Response	
<pre>[{ "id": "123", "title": "ABC" }, { "id": "789", "title": "XYZ" }]</pre>	

10.3.2.3 読み取り

“id”、または “archive” パラメータが指定されており、“search” パラメータが指定されていない場合、GET リクエストはそのコンテンツ項目のデータを返します。返されるデータには、すべてのコンテンツ項目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/content/?id=ID	
GET /webadmin/rest/content/?archive=ID	
id	コンテンツ項目の ID
archive	アーカイブされたコンテンツ項目のアーカイブ ID
Response	
<pre>{ "id": "32",</pre>	



```
"created": "2009-11-30 09:20:27",
"updated": "2013-08-08 13:42:37",
"published": "2013-08-08 13:42:37",
"unpublished": "",
"deleted": "",
"created_by": "admin",
"updated_by": "admin",
"published_by": "admin",
"unpublished_by": "",
"deleted_by": "",

"revision": "",

"title": "My Business",
"content": "<div class=\"featureimage\">\r\n .... </div>\r\n<br
class=\"clearalign\">",
"summary": "",
"contentformat": "",
"searchable": "",
"menuitem": "",
"upload_filename": "",
"server_filename": "",
"contentdelivery": "",
"url": "",

"template": "34",
"stylesheet": "",

"element": {
  "banner": "161",
  "breadcrumbs": "-1",
  "featurebox1": "-1",
  "featurebox2": "-1",
  "footer": "-1",
  "logo": "-1",
  "menu": "-1",
  "personal": "-1",
  "toolbar": "-1",
  "utilities": "-1"
},

"image1": "",
"image2": "",
"image3": "",
"file1": "",
"file2": "",
"file3": "",
"link1": "",
"link2": "",
"link3": "",

"author": "",
"keywords": "The homepage for My Business.",
"description": "What we do.",
"metainfo": "<language>EN</language>",
"contentschema": "",
"segmentation": "",

"scripts": "",
"doctype": "",
"htmlattributes": "",
"headattributes": "",
"htmlheader": "",
"htmlbodyonload": "",

"contentpackage": "",
"contentbundle": "",
"contentclass": "page",
"contentgroup": "Home",
"contenttype": "",
```



```
"version_master": "0",
"version": "",
"device": "",
"usersegment": "",
"usertest": "",

"users_users": "",
"users_type": "",
"users_group": "",

"creators_users": "",
"creators_type": "",
"creators_group": "",
"editors_users": "",
"editors_type": "",
"editors_group": "",
"publishers_users": "",
"publishers_type": "",
"publishers_group": "",
"developers_users": "",
"developers_type": "",
"developers_group": "",
"administrators_users": "",
"administrators_type": "",
"administrators_group": "",

"page_top": "",
"page_up": "",
"page_previous": "0",
"page_next": "531",
"page_first": "32",
"page_last": "570",
"related": "",

"scheduled_publish": "",
"requested_publish": "",
"scheduled_unpublish": "",
"requested_unpublish": "",

"checkedout": "",
"status": "",
"status_by": "",
"locked": 0,
"locked_user": 0,
"locked_creator": 0,
"locked_developer": 0,
"locked_editor": 0,
"locked_publisher": 0,
"locked_administrator": 0,
"locked_schedule": 0,
"locked_unschedule": 0,

"product_code": "",
"product_currency": "",
"product_cost": "",
"product_price": "",
"product_period": "",
"product_weight": "",
"product_volume": "",
"product_width": "",
"product_height": "",
"product_depth": "",
"product_email": "",
"product_url": "",
"product_brand": "",
"product_colour": "",
"product_size": "",
"product_info": "",
"product_options": "",
"product_delivery": "",
"product_stock_amount": "0",
"product_stock_text": "",
```

```

"product_restocked_amount": "0",
"product_lowstock_amount": "",
"product_lowstock_text": "",
"product_nostock_order": "",
"product_nostock_text": "",
"product_location": "",
"product_user": "",
"product_page": "",
"product_program": "",
"product_hosting": "",
"product_content": "",
"product_contentgroup": "",
"product_contenttype": "",
"product_imagegroup": "",
"product_imagetype": "",
"product_filegroup": "",
"product_filetype": "",
"product_linkgroup": "",
"product_linktype": "",
"product_productgroup": "",
"product_producttype": "",
"product_usergroup": "",
"product_usertype": "",

"heatmap": "",
"heatmapalign": "",
"usagelog": "",

"user": true,
"creator": true,
"editor": true,
"publisher": true,
"developer": true,
"administrator": true
}

```

10.3.2.4 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいコンテンツ項目を作成します。

“id” パラメータが指定されている場合、その ID のコンテンツ項目がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたコンテンツ項目のデータを返します。返されるデータには、すべてのコンテンツ項目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/content/	
id	コピーするコンテンツ項目の ID（任意）
.....	Web コンテンツ管理システムで使用するすべてのコンテンツ項目の属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
file_data	Base64 でエンコードされた “data:” URL 形式の画像、またはその他のファイルコンテンツ “file_data” パラメータを指定する場合は “upload_filename” および “server_filename” パラメータも指定する必要があります。
publish	“publish”



10.3.2.5 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツ項目を更新します。

PUT リクエストは、更新されたコンテンツ項目のデータを返します。返されるデータには、すべてのコンテンツ項目の属性と、現在ログインしている **Web サイト管理者**の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の **true/false** フラグが含まれます。

Page 61 of 179

Page 62 of 179

10.3.2.6 削除

DELETE リクエストは、指定された ID の既存のコンテンツ項目を削除、または非公開にします。

DELETE リクエストは、削除/非公開されたコンテンツ項目のデータを返します。返されるデータには、すべてのコンテンツ項目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/content/	
archive	コンテンツ項目のアーカイブ ID コンテンツ項目のアーカイブ ID が指定されている場合、そのアーカイブされたコンテンツ項目は削除されます。公開済み、更新中、およびその他のアーカイブされたコンテンツ項目は削除されません。
id	コンテンツ項目の ID コンテンツ項目の ID が指定されている場合、そのコンテンツ項目は削除されます。デフォルトでは、公開済みと更新中の両方、およびアーカイブされたすべてのコンテンツ項目が削除されます。
unpublish	“unpublish” “unpublish” パラメータが指定された場合、指定されたコンテンツ項目は非公開となります。更新中、およびアーカイブされたコンテンツ項目は削除されません。
Request	
<pre>{ "id": "123" }</pre>	
Request	
<pre>{ "id": "123", "unpublish": "unpublish" }</pre>	
Request	
<pre>{ "archive": "456" }</pre>	
Response	
<pre>{ "id": "123", "created": "2020-01-01 00:00:00", "updated": "2020-01-31 23:59:59", "unpublished": "2020-12-31 23:59:59", "....", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": false }</pre>	

10.3.3 データ

“/webadmin/rest/data/” REST API 関数は、1 つ以上のコンテンツデータベースのデータ項目のすべてのデータを返し、コンテンツデータベースのデータ項目を作成、更新、削除します。

10.3.3.1 リスト

“id” および “search” パラメータが指定されていない場合、他の指定されたパラメータと一致するコンテンツデータベースのデータ項目のリストが返されます。

GET /webadmin/rest/data/?database=DATABASE&pagesize=PAGESIZE&offset=OFFSET&sort_col=ATTRIBUTE&sort_dir=DESC	
database	コンテンツデータベースの ID または名前（必須）
columns	返されるコンテンツデータベースの属性名 デフォルトでは、すべてのコンテンツデータベース属性が返されます。
pagesize	リストするデータ項目最大数 デフォルトでは、一致するすべてのデータ項目がリストされます。
offset	リストされるデータ項目の最初のデータ項目番号 デフォルトでは、一致するすべてのデータ項目がリストされます。 例： 最初の 10 個のデータ項目（1-10） “offset=0&pagesize=10” 次の 10 個のデータ項目（11-20） “offset=10&pagesize=10” 次の 10 個のデータ項目（21-30） “offset=20&pagesize=10”
sort_col	リストされたデータ項目をソートするためのコンテンツデータベース属性名（“id”、“title”、または設定されたコンテンツデータベース属性名）
sort_dir	“ASC”、または “DESC” リストされたデータ項目を昇順、または降順でソートします。デフォルトでは、データ項目は昇順でソートされます。
Response	
<pre>[{ "database": "Events", "id": "123", "Title": "ABC", } , , { "database": "Events", "id": "789", "Title": "XYZ", }]</pre>	

10.3.3.2 検索

“search” パラメータが指定されている場合、“search” パラメータと他のすべての指定されたパラメータに一致するコンテンツデータベースのデータ項目の検索結果リストが返されます。

GET /webadmin/rest/data/?database=DATABASE&search=SEARCH&pagesize=PAGESIZE&offset=OFFSET&sort_col=ATTRIBUTE&sort_dir=DESC	
database	コンテンツデータベースの ID または名前 (必須)
search	検索ワード
attribute	検索するコンテンツデータベース属性名 カンマ区切り、指定しない場合は全属性が対象
columns	返されるコンテンツデータベース属性名 デフォルトでは、すべてのコンテンツデータベース属性が返されます。
pagesize	リストするデータ項目最大数 デフォルトでは、一致するすべてのデータ項目がリストされます。
offset	リストされるデータ項目の最初のデータ項目番号 デフォルトでは、一致するすべてのデータ項目がリストされます。 例： 最初の 10 個のデータ項目 (1-10) “offset=0&pagesize=10” 次の 10 個のデータ項目 (11-20) “offset=10&pagesize=10” 次の 10 個のデータ項目 (21-30) “offset=20&pagesize=10”
sort_col	リストされたデータ項目をソートするためのコンテンツデータベース属性名 (“id”, “title”, または設定されたコンテンツデータベース属性名)
sort_dir	“ASC”、または “DESC” リストされたデータ項目を昇順、または降順でソートします。デフォルトでは、データ項目は昇順でソートされます。
Response	
<pre>[{ "database": "Events", "id": "123", "Title": "ABC", }, { "database": "Events", "id": "789", "Title": "XYZ", }]</pre>	

10.3.3.3 読み取り

“id” パラメータが指定されており、“search” パラメータが指定されていない場合、GET リクエストはそのコンテンツデータベースのデータ項目のデータを返します。返されるデータには、す

すべてのコンテンツデータベースのデータ項目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/data/?database=DATABASE&id=ID	
database	コンテンツデータベースの ID または名前（必須）
id	コンテンツデータベースのデータ項目の ID
Response	
<pre>{ "database": "Events", "id": "1", "Title": "First Quarter Results", "Category": "Announcement", "Location": "", "From": "2021-03-17", "To": "2021-03-17", "Description": "First quarter results are announced.", "Details": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam imperdiet erat pulvinar ante mollis non sodales orci rutrum. Nulla ac quam ut arcu rhoncus molestie. Donec in lectus eros. Sed sit amet sapien vitae purus consectetur interdum. Nulla ultrices arcu id magna ultrices cursus. Ut sed ante odio, et ullamcorper augue.", "Signup": "Not Required", "Signup_Expiry": "", "Created_By": "admin", "Created_Date": "2010-03-16 12:34:46", "Type": "Shareholder", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.3.4 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいコンテンツデータベースのデータ項目を作成します。

“id” パラメータが指定されている場合、その ID のコンテンツデータベースのデータ項目がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたコンテンツデータベースのデータ項目のデータを返します。返されるデータには、すべてのデータ項目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/data/	
database	コンテンツデータベースの ID または名前（必須）
id	コンテンツデータベースのデータ項目の ID（任意）
.....	コンテンツデータベースに設定されたすべてのコンテンツデータベースの属性、または任意の数と組み合わせ
Request	
<pre>{ "database": "Events",</pre>	



```
"id": "123",
"Title": "First Quarter Results",
"Category": "Announcement",
"Location": "",
"From": "2021-03-17",
"To": "2021-03-17",
"Description": "First quarter results are announced.",
"Details": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam
imperdiet erat pulvinar ante mollis non sodales orci rutrum. Nulla ac quam ut
arcu rhoncus molestie. Donec in lectus eros. Sed sit amet sapien vitae purus
consectetur interdum. Nulla ultrices arcu id magna ultrices cursus. Ut sed
ante odio, et ullamcorper augue.",
"Signup": "Not Required",
"Signup_Expiry": "",
"Created_By": "admin",
"Created_Date": "2010-03-16 12:34:46",
"Type": "Shareholder",
"user": true,
"creator": true,
"editor": true,
"publisher": true,
"administrator": true
}
```

Response

```
{
  "database": "Events",
  "id": "789",
  "Title": "First Quarter Results",
  "Category": "Announcement",
  "Location": "",
  "From": "2021-03-17",
  "To": "2021-03-17",
  "Description": "First quarter results are announced.",
  "Details": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam
imperdiet erat pulvinar ante mollis non sodales orci rutrum. Nulla ac quam ut
arcu rhoncus molestie. Donec in lectus eros. Sed sit amet sapien vitae purus
consectetur interdum. Nulla ultrices arcu id magna ultrices cursus. Ut sed
ante odio, et ullamcorper augue.",
  "Signup": "Not Required",
  "Signup_Expiry": "",
  "Created_By": "admin",
  "Created_Date": "2010-03-16 12:34:46",
  "Type": "Shareholder",
  "user": true,
  "creator": true,
  "editor": true,
  "publisher": true,
  "administrator": true
}
```

10.3.3.5 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツデータベースのデータ項目を更新します。

PUT リクエストは、更新されたデータ項目のデータを返します。返されるデータには、すべてのデータ項目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/data/	
database	コンテンツデータベースの ID または名前 (必須)
id	コンテンツデータベースのデータ項目の ID (必須)

.....	コンテンツデータベースに設定されたすべてのコンテンツデータベースの属性、または任意の数と組み合わせ
Request	
<pre>{ "database": "Events", "id": "789", "Title": "First Quarter Results", "Category": "Announcement", "Location": "", "From": "2021-03-17", "To": "2021-03-17", "Description": "First quarter results are announced.", "Details": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam imperdiet erat pulvinar ante mollis non sodales orci rutrum. Nulla ac quam ut arcu rhoncus molestie. Donec in lectus eros. Sed sit amet sapien vitae purus consectetur interdum. Nulla ultrices arcu id magna ultrices cursus. Ut sed ante odio, et ullamcorper augue.", "Signup": "Not Required", "Signup_Expiry": "", "Created_By": "admin", "Created_Date": "2010-03-16 12:34:46", "Type": "Shareholder", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	
Response	
<pre>{ "database": "Events", "id": "789", "Title": "First Quarter Results", "Category": "Announcement", "Location": "", "From": "2021-03-17", "To": "2021-03-17", "Description": "First quarter results are announced.", "Details": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam imperdiet erat pulvinar ante mollis non sodales orci rutrum. Nulla ac quam ut arcu rhoncus molestie. Donec in lectus eros. Sed sit amet sapien vitae purus consectetur interdum. Nulla ultrices arcu id magna ultrices cursus. Ut sed ante odio, et ullamcorper augue.", "Signup": "Not Required", "Signup_Expiry": "", "Created_By": "admin", "Created_Date": "2010-03-16 12:34:46", "Type": "Shareholder", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.3.6 削除

DELETE リクエストは、指定された ID の既存のコンテンツデータベースのデータ項目を削除します。

DELETE リクエストは、削除されたデータ項目のデータを返します。返されるデータには、すべてのデータ項目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/data/	
database	コンテンツデータベースの ID または名前 (必須)
id	コンテンツデータベースのデータ項目の ID (必須)
Request	
<pre>{ "database": "Events", "id": "123" }</pre>	
Response	
<pre>{ "database": "Events", "id": "123", "Title": "First Quarter Results", "Category": "Announcement", "Location": "", "From": "2021-03-17", "To": "2021-03-17", "Description": "First quarter results are announced.", "Details": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam imperdiet erat pulvinar ante mollis non sodales orci rutrum. Nulla ac quam ut arcu rhoncus molestie. Donec in lectus eros. Sed sit amet sapien vitae purus consectetur interdum. Nulla ultrices arcu id magna ultrices cursus. Ut sed ante odio, et ullamcorper augue.", "Signup": "Not Required", "Signup_Expiry": "", "Created_By": "admin", "Created_Date": "2010-03-16 12:34:46", "Type": "Shareholder", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.4 ユーザ

“/webadmin/rest/users/” REST API 関数は、1 つ以上のユーザアカウントのすべてのデータを返し、ユーザアカウントを作成、更新、削除します。

10.3.4.1 リスト

“id”、および“search”パラメータが指定されていない場合、他の指定されたパラメータと一致するユーザアカウントのリストが返されます。

GET /webadmin/rest/users/?userclass=USERCLASS&usergroup=USERGROUP&usertype=USERTYPE&status=STATUS&pagesize=PAGESIZE&offset=OFFSET&sort_col=ATTRIBUTE&sort_dir=DESC	
userclass	ユーザクラス名 (“administrator”, “user”、指定しない場合は全ユーザクラスが対象)
usergroup	ユーザグループ名、または“-” (なし)
usertype	ユーザタイプ名、または“-” (なし)
status	ユーザステータスコード (“pending”, “active”, “expiring”, “expired”)
pagesize	リストするユーザアカウント最大数 デフォルトでは、一致するすべてのユーザアカウントがリストされます。

offset	リストされるユーザアカウントの最初のユーザアカウント番号 デフォルトでは、一致するすべてのユーザアカウントがリストされます。 例： 最初の 10 個のユーザアカウント (1-10) “offset=0&pagesize=10” 次の 10 個のユーザアカウント (11-20) “offset=10&pagesize=10” 次の 10 個のユーザアカウント (21-30) “offset=20&pagesize=10”
sort_col	リストされたユーザアカウントをソートするためのユーザアカウント属性名 (“id”, “name”, “username”, “userclass”, “usergroup”, “usertype” など)
sort_dir	“ASC”、または “DESC” リストされたユーザアカウントを昇順、または降順でソートします。デフォルトでは、ユーザアカウントは昇順でソートされます。
Response	
<pre>[{ "id": "123", "username": "ABC" }, { "id": "789", "username": "XYZ" }]</pre>	

10.3.4.2 検索

“search” パラメータが指定されている場合、“search” パラメータと指定した他のすべてのパラメータに一致するユーザアカウントの検索結果リストが返されます。

GET /webadmin/rest/users/?search=SEARCH& attribute=ATTRIBUTE &pagesize=PAGESIZE&offset=OFFSET&sort_col=ATTRIBUTE&sort_dir=DESC	
search	検索ワード
attribute	検索するユーザアカウント属性名 (“name”, “organisation”, “email”, “username”, “delivery_name”, “delivery_organisation”, “delivery_email”, “delivery_website”, “invoice_name”, “invoice_organisation”, “invoice_email”, “invoice_website”, “notes”, “userinfo”) カンマ区切り、指定しない場合は全属性が対象
pagesize	リストするユーザアカウント最大数 デフォルトでは、一致するすべてのユーザアカウントがリストされます。
offset	リストされるユーザアカウントの最初のユーザアカウント番号

	デフォルトでは、一致するすべてのユーザアカウントがリストされます。 例： 最初の 10 個のユーザアカウント (1-10) “offset=0&pagesize=10” 次の 10 個のユーザアカウント (11-20) “offset=10&pagesize=10” 次の 10 個のユーザアカウント (21-30) “offset=20&pagesize=10”
sort_col	リストされたユーザアカウントをソートするためのユーザアカウント属性名 (“id”, “name”, “username”, “userclass”, “usergroup”, “usertype” など)
sort_dir	“ASC”、または “DESC” リストされたユーザアカウントを昇順、または降順でソートします。デフォルトでは、ユーザアカウントは昇順でソートされます。
Response	
<pre>[{ "id": "123", "username": "ABC" }, { "id": "789", "username": "XYZ" }]</pre>	

10.3.4.3 読み取り

“id” パラメータが指定されており、“search” パラメータが指定されていない場合、GET リクエストはそのユーザアカウントのデータを返します。返されるデータには、すべてのユーザアカウントの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/users/?id=ID	
id	ユーザアカウントの ID (必須)
Response	
<pre>{ "id": "6", "created": "2010-03-24 13:59:08", "updated": "2020-02-28 11:04:32", "created_by": "admin", "updated_by": "admin", "password_updated": "2010-04-01 14:13:53", "username": "paulgreen", "password": "XYABQ0EB1LG7MCCUXTUJXRXBK1FJ0LXL4H86HEGVY8VK5QHRHGA71KB9AXGLVBYN", "onetimepassword": "", "onetimepassword_updated": "2020-02-28 11:04:32", "secret": "", </pre>	



```
"secret_updated": "2020-02-28 11:04:32",

"title": "",
"name": "Paul Green",
"organisation": "",
"email": "",
"gender": "",
"photo": "",
"birthdate": "",
"birthyear": "",
"birthmonth": "",
"birthday": "",
"notes": "",
"userinfo": "",
"keywords": "",
"description": "",

"userclass": "administrator",
"usergroup": "Human Resources",
"usertype": "",
"usergroups": [
    "Website Administrators"
],
"usertypes": [
],

"users_group": "User Managers",
"users_type": "",
"users_users": "",
"creators_group": "User Managers",
"creators_type": "",
"creators_users": "",
"editors_group": "User Managers",
"editors_type": "",
"editors_users": "",
"publishers_group": "User Managers",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "Website Administrators",
"administrators_type": "",
"administrators_users": "",

"scheduled_publish": "",
"scheduled_publish_email": "",
"scheduled_notify": "",
"scheduled_notify_email": "",
"scheduled_unpublish": "",
"scheduled_unpublish_email": "",
"scheduled_last": "",

"invoice_name": "",
"invoice_organisation": "",
"invoice_address": "",
"invoice_postalcode": "",
"invoice_city": "",
"invoice_state": "",
"invoice_country": "",
"invoice_phone": "",
"invoice_fax": "",
"invoice_email": "",
"invoice_website": "",

"delivery_name": "",
"delivery_organisation": "",
"delivery_address": "",
"delivery_postalcode": "",
"delivery_city": "",
"delivery_state": "",
"delivery_country": "",
"delivery_phone": "",
"delivery_fax": "",
"delivery_email": "",
```



```
"delivery_website": "",

"card_type": "",
"card_number": "",
"card_issuedmonth": "",
"card_issuedyear": "",
"card_expiryyear": "",
"card_expiryyear": "",
"card_name": "",
"card_cvc": "",
"card_issue": "",
"card_postalcode": "",

"content_editor": "",
"hardcore_upload": "",
"hardcore_format": "",
"hardcore_width": "",
"hardcore_height": "",
"hardcore_onenter": "",
"hardcore_onctrlenter": "",
"hardcore_onshiftenter": "",
"hardcore_onaltenter": "",
"hardcore_skin": "",
"hardcore_toolbar": "",
"hardcore_toolbar1": "",
"hardcore_toolbar2": "",
"hardcore_toolbar3": "",
"hardcore_toolbar4": "",
"hardcore_toolbar5": "",
"hardcore_toolbar6": "",
"hardcore_toolbar7": "",
"hardcore_toolbar8": "",
"hardcore_toolbar9": "",
"hardcore_toolbar10": "",
"hardcore_formatblock": "",
"hardcore_fontname": "",
"hardcore_fontsize": "",
"hardcore_customscript": "",
"hardcore_encoding": "",
"hardcore_language": "",

"startpage": "",

"webadmin_library_packages": "forbid",
"webadmin_ecommerce": "forbid",
"webadmin_user": "forbid",
"webadmin_ecommerce": "forbid",
"webadmin_user": "forbid",
"webadmin_library_packages": "forbid",
"webadmin_user": "forbid",
"webadmin_ecommerce": "forbid",
"webadmin_library_packages": "forbid",
"webadmin_user": "forbid",
"webadmin_ecommerce": "forbid",
"webadmin_library_packages": "forbid",

"workspace_sections": "",
"index_workspace": "",
"index_workspace_projects": "",
"index_workspace_comments": "",
"index_content": "",
"index_library": "",
"index_search": "",
"index_searchadvanced": "",
"index_searchreplace": "",
"index_product": "",
"index_stock": "",
"index_order": "",
"index_segments": "",
"index_usertests": "",
"index_heatmaps": "",
"index_user": "",
```

```

    "index_websites": "",
    "index_sales": "",
    "index_databases": "",
    "index_hosting": "",
    "index_projects": "",

    "menu_selection": "",
    "statistics_reports": "",
    "sales_reports": "",

    "shopcart": "",
    "wishlist": "",

    "failed": 0,
    "locked": 0,

    "user": true,
    "creator": true,
    "editor": true,
    "publisher": true,
    "administrator": true
}

```

10.3.4.4 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいユーザアカウントを作成します。

“id” パラメータが指定されている場合、その ID のユーザアカウントがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたユーザアカウントのデータを返します。返されるデータには、すべてのユーザアカウントの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/users/	
id	コピーするユーザアカウントの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべてのユーザアカウントの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre> { "id": "123", "userclass": "administrator", "usergroup": "Home", "usertype": "", "username": "johndoe", "password": "SECRET" } </pre>	
Response	
<pre> { "id": "789", "created": "2020-01-31 23:59:59", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true } </pre>	

```
}

```

10.3.4.5 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のユーザアカウントを更新します。

PUT リクエストは、更新されたユーザアカウントのデータを返します。返されるデータには、すべてのユーザアカウントの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/users/	
id	更新するユーザアカウントの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのユーザアカウントの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "789", "userclass": "administrator", "usergroup": "Home", "usertype": "", "username": "johndoe", "password": "SECRET" }</pre>	
Response	
<pre>{ "id": "789", "created": "2020-01-01 00:00:00", "updated": "2020-01-31 23:59:59", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": false }</pre>	

10.3.4.6 削除

DELETE リクエストは、指定された ID の既存のユーザアカウントを削除します。

DELETE リクエストは、削除されたユーザアカウントのデータを返します。返されるデータには、すべてのユーザアカウントの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/users/	
id	ユーザアカウントの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{</pre>	

```

    "id": "123",

    "created": "2020-01-01 00:00:00",
    "updated": "2020-01-31 23:59:59",
    .....
    "user": true,
    "creator": true,
    "editor": true,
    "publisher": true,
    "developer": true,
    "administrator": true
  }

```

10.3.5 マイクロ Web サイト

“/webadmin/rest/websites/” REST API 関数は、1 つ以上のマイクロ Web サイトのすべてのデータを返し、マイクロ Web サイトを作成、更新、削除します。

10.3.5.1 リスト

“id” パラメータが指定されていない場合、マイクロ Web サイトのリストが返されます。

GET /webadmin/rest/websites/	
Response	
<pre> [{ "id": "123", "domain": "ABC", }, { "id": "789", "domain": "XYZ", }] </pre>	

10.3.5.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのマイクロ Web サイトのデータを返します。返されるデータには、すべてのマイクロ Web サイトの属性が含まれます。

GET /webadmin/rest/websites/?id=ID	
id	マイクロ Web サイトの ID (必須)
Response	
<pre> { "id": "1", "domain": "amp.yourwebsite.com", "remote": "", "useragent": "", "language": "", "referrer": "", "keywords": "", "default_version": "", "default_template": "", "default_stylesheet": "", "default_doctype": "", "default_html": "", "default_head": "", </pre>	

```

    "default_page": "123",
    "default_page_nonexisting": "",
    "default_page_unpublished": "",
    "default_page_expired": "",
    "default_login": "",
    "default_login_code": "",
    "default_login_code_email": "",
    "default_searchresults_page": "",
    "default_searchresults_entry": "",
    "default_register_confirmation_page": "",
    "default_register_notification_page": "",
    "retrieve_password_page": "",
    "retrieve_password_confirmation": "",
    "retrieve_password_email": "",
    "retrieve_password_error": "",

    "default_price": "",
    "default_country": "",
    "default_state": "",

    "default_list_entry": "",
    "default_publish_ready": "",
    "default_personal_admin_page": ""
  }

```

10.3.5.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいマイクロ Web サイトを作成します。

“id” パラメータが指定されている場合、その ID のマイクロ Web サイトがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたマイクロ Web サイトのデータを返します。返されるデータには、すべてのマイクロ Web サイトの属性が含まれます。

POST /webadmin/rest/websites/	
id	コピーするマイクロ Web サイトの ID（任意）
.....	Web コンテンツ管理システムで使用するすべてのマイクロ Web サイトの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "domain": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "domain": "XYZ", }</pre>	

10.3.5.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のマイクロ Web サイトを更新します。

PUT リクエストは、更新されたマイクロ Web サイトのデータを返します。返されるデータには、すべてのマイクロ Web サイトの属性が含まれます。

PUT /webadmin/rest/websites/	
id	更新するマイクロ Web サイトの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのマイクロ Web サイトの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "default_page": "789" }</pre>	
Response	
<pre>{ "id": "123", "domain": "ABC", "default_page": "789", }</pre>	

10.3.5.5 削除

DELETE リクエストは、指定された ID の既存のマイクロ Web サイトを削除します。

DELETE リクエストは、削除されたマイクロ Web サイトのデータを返します。返されるデータには、すべてのマイクロ Web サイトの属性が含まれます。

DELETE /webadmin/rest/websites/	
id	マイクロ Web サイトの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "domain": "ABC", }</pre>	

10.3.6 構成要素

“/webadmin/rest/elements/” REST API 関数は、1 つ以上の構成要素のすべてのデータを返し、構成要素を作成、更新、削除します。

10.3.6.1 リスト

“id” パラメータが指定されていない場合、構成要素のリストが返されます。

GET /webadmin/rest/elements/	
Response	
<pre>[{ "id": "123", "element": "ABC", }</pre>	

```

}
,
.....
,
{
  "id": "789",
  "element": "XYZ",
  .....
}
]

```

10.3.6.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその構成要素のデータを返します。返されるデータには、すべての構成要素の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/elements/?id=ID	
id	構成要素の ID（必須）
Response	
<pre> { "id": "1", "element": "banner", "users_group": "", "users_type": "", "users_users": "", "creators_group": "", "creators_type": "", "creators_users": "", "developers_group": "", "developers_type": "", "developers_users": "", "editors_group": "", "editors_type": "", "editors_users": "", "publishers_group": "", "publishers_type": "", "publishers_users": "", "administrators_group": "", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true } </pre>	

10.3.6.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい構成要素を作成します。

“id” パラメータが指定されている場合、その ID の構成要素がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された構成要素のデータを返します。返されるデータには、すべての構成要素の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更

新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/elements/	
id	コピーする構成要素の ID (任意)
.....	Web コンテンツ管理システムで使用されるすべての構成要素の属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "element": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "element": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.6.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の構成要素を更新します。

PUT リクエストは、更新された構成要素のデータを返します。返されるデータには、すべての構成要素の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/elements/	
id	更新する構成要素の ID (必須)
.....	Web コンテンツ管理システムで使用されるすべての構成要素の属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "administrators_group": "Website Administrators" }</pre>	
Response	
<pre>{ "id": "123", "element": "ABC", "administrators_group": "Website Administrators", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, }</pre>	

```
"administrator": true
}
```

10.3.6.5 削除

DELETE リクエストは、指定された ID の既存の構成要素を削除します。

DELETE リクエストは、削除された構成要素のデータを返します。返されるデータには、すべての構成要素の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/elements/	
id	構成要素の ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "element": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.7 コンテンツグループ

“/webadmin/rest/contentgroups/” REST API 関数は、1 つ以上のコンテンツグループのすべてのデータを返し、コンテンツグループを作成、更新、削除します。

10.3.7.1 リスト

“id” パラメータが指定されていない場合、コンテンツグループのリストが返されます。

GET /webadmin/rest/contentgroups/	
Response	
<pre>[{ "id": "123", "contentgroup": "ABC" }, { "id": "789", "contentgroup": "XYZ" }]</pre>	

10.3.7.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのコンテンツグループのデータを返します。返されるデータには、すべてのコンテンツグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/contentgroups/?id=ID	
id	コンテンツグループの ID（必須）
Response	
<pre>{ "id": "1", "contentgroup": "About Us", "parentgroup": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Press Officers", "creators_type": "", "creators_users": "", "developers_group": "Website Developers", "developers_type": "", "developers_users": "", "editors_group": "Press Officers", "editors_type": "", "editors_users": "", "publishers_group": "Press Officers", "publishers_type": "", "publishers_users": "", "administrators_group": "Website Administrators", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.7.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいコンテンツグループを作成します。

“id” パラメータが指定されている場合、その ID のコンテンツグループがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたコンテンツグループのデータを返します。返されるデータには、すべてのコンテンツグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/contentgroups/

id	コピーするコンテンツグループの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのコンテンツグループの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "contentgroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "contentgroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.7.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツグループを更新します。

PUT リクエストは、更新されたコンテンツグループのデータを返します。返されるデータには、すべてのコンテンツグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/contentgroups/	
id	更新するコンテンツグループの ID (必須)
.....	Web コンテンツ管理システムで使用されるすべてのコンテンツグループの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "template": "456", "stylesheet": "789" }</pre>	
Response	
<pre>{ "id": "123", "contentgroup": "ABC", "parentgroup": "", "template": "456", "stylesheet": "789" "user": true, "creator": true, "editor": true, "publisher": true, "developer": true,</pre>	

```
"administrator": true
}
```

10.3.7.5 削除

DELETE リクエストは、指定された ID の既存のコンテンツグループを削除します。

DELETE リクエストは、削除されたコンテンツグループのデータを返します。返されるデータには、すべてのコンテンツグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/contentgroups/	
id	コンテンツグループの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "contentgroup": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.8 コンテンツタイプ

“/webadmin/rest/contenttypes/” REST API 関数は、1 つ以上のコンテンツタイプのすべてのデータを返し、コンテンツタイプを作成、更新、削除します。

10.3.8.1 リスト

“id” パラメータが指定されていない場合、コンテンツタイプのリストが返されます。

GET /webadmin/rest/contenttypes/	
Response	
<pre>[{ "id": "123", "contenttype": "ABC" }, { "id": "789", "contenttype": "XYZ" }]</pre>	

10.3.8.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのコンテンツタイプのデータを返します。返されるデータには、すべてのコンテンツタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/contenttypes/?id=ID	
id	コンテンツタイプの ID (必須)
Response	
<pre>{ "id": "1", "contenttype": "News", "parenttype": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Press Officers", "creators_type": "", "creators_users": "", "developers_group": "Website Developers", "developers_type": "", "developers_users": "", "editors_group": "Press Officers", "editors_type": "", "editors_users": "", "publishers_group": "Press Officers", "publishers_type": "", "publishers_users": "", "administrators_group": "Website Administrators", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.8.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいコンテンツタイプを作成します。

“id” パラメータが指定されている場合、その ID のコンテンツタイプがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたコンテンツタイプのデータを返します。返されるデータには、すべてのコンテンツタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/contenttypes/	
id	コピーするコンテンツタイプの ID (任意)
.....	Web コンテンツ管理システムで使用するすべてのコンテンツタイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "contenttype": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "contenttype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.8.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツタイプを更新します。

PUT リクエストは、更新されたコンテンツタイプのデータを返します。返されるデータには、すべてのコンテンツタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/contenttypes/	
id	更新するコンテンツタイプの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのコンテンツタイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "template": "456", "stylesheet": "789" }</pre>	
Response	
<pre>{ "id": "123", "contenttype": "ABC", "parenttype": "", "template": "456", "stylesheet": "789" "user": true, "creator": true, "editor": true, "publisher": true, </pre>	

```
{
  "developer": true,
  "administrator": true
}
```

10.3.8.5 削除

DELETE リクエストは、指定された ID の既存のコンテンツタイプを削除します。

DELETE リクエストは、削除されたコンテンツタイプのデータを返します。返されるデータには、すべてのコンテンツタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/contenttypes/	
id	コンテンツタイプの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "contenttype": "ABC", , "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.9 イメージフォーマット

“/webadmin/rest/imageformats/” REST API 関数は、1 つ以上のイメージフォーマットのすべてのデータを返し、イメージフォーマットを作成、更新、削除します。

10.3.9.1 リスト

“id” パラメータが指定されていない場合、イメージフォーマットのリストが返されます。

GET /webadmin/rest/imageformats/	
Response	
<pre>[{ "id": "123", "filenameextension": "ABC" }, { "id": "789", "filenameextension": "XYZ" }]</pre>	

10.3.9.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのイメージフォーマットのデータを返します。返されるデータには、すべてのイメージフォーマットの属性が含まれます。

GET /webadmin/rest/imageformats/?id=ID	
id	イメージフォーマットの ID (必須)
Response	
<pre>{ "id": "1", "filenameextension": "gif" }</pre>	

10.3.9.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいイメージフォーマットを作成します。

“id” パラメータが指定されている場合、その ID のイメージフォーマットがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたイメージフォーマットのデータを返します。返されるデータには、すべてのイメージフォーマットの属性が含まれます。

POST /webadmin/rest/imageformats/	
id	コピーするイメージフォーマットの ID (任意)
.....	Web コンテンツ管理システムで使用するすべてのイメージフォーマットの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "filenameextension": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "filenameextension": "XYZ" }</pre>	

10.3.9.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のイメージフォーマットを更新します。

PUT リクエストは、更新されたイメージフォーマットのデータを返します。返されるデータには、すべてのイメージフォーマットの属性が含まれます。

PUT /webadmin/rest/imageformats/	
id	更新するイメージフォーマットの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのイメージフォーマットの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）

Request
<pre>{ "id": "123", "filenameextension": "XYZ" }</pre>
Response
<pre>{ "id": "123", "filenameextension": "XYZ" }</pre>

10.3.9.5 削除

DELETE リクエストは、指定された ID の既存のイメージフォーマットを削除します。

DELETE リクエストは、削除されたイメージフォーマットのデータを返します。返されるデータには、すべてのイメージフォーマットの属性が含まれます。

DELETE /webadmin/rest/imageformats/	
id	イメージフォーマットの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "filenameextension": "ABC" }</pre>	

10.3.10 イメージグループ

“/webadmin/rest/imagegroups/” REST API 関数は、1 つ以上のイメージグループのすべてのデータを返し、イメージグループを作成、更新、削除します。

10.3.10.1 リスト

“id” パラメータが指定されていない場合、イメージグループのリストが返されます。

GET /webadmin/rest/imagegroups/
Response
<pre>[{ "id": "123", "imagegroup": "ABC" }, { "id": "789", "imagegroup": "XYZ" }]</pre>

10.3.10.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのイメージグループのデータを返します。返されるデータには、すべてのイメージグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/imagegroups/?id=ID	
id	イメージグループの ID (必須)
Response	
<pre>{ "id": "1", "imagegroup": "Template", "parentgroup": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Website Developers", "creators_type": "", "creators_users": "", "developers_group": "Website Developers", "developers_type": "", "developers_users": "", "editors_group": "Website Developers", "editors_type": "", "editors_users": "", "publishers_group": "Website Developers", "publishers_type": "", "publishers_users": "", "administrators_group": "Website Administrators", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.10.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいイメージグループを作成します。

“id” パラメータが指定されている場合、その ID のイメージグループがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたイメージグループのデータを返します。返されるデータには、すべてのイメージグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/imagegroups/

id	コピーするイメージグループの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのイメージグループの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "imagegroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "imagegroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.10.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のイメージグループを更新します。

PUT リクエストは、更新されたイメージグループのデータを返します。返されるデータには、すべてのイメージグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/imagegroups/	
id	更新するイメージグループの ID (必須)
.....	Web コンテンツ管理システムで使用されるすべてのイメージグループの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "parentgroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "imagegroup": "ABC", "parentgroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.10.5 削除

DELETE リクエストは、指定された ID の既存のイメージグループを削除します。

DELETE リクエストは、削除されたイメージグループのデータを返します。返されるデータには、すべてのイメージグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/imagegroups/	
id	イメージグループの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "imagegroup": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.11 イメージタイプ

“/webadmin/rest/imagetypes/” REST API 関数は、1 つ以上のイメージタイプのすべてのデータを返し、イメージタイプを作成、更新、削除します。

10.3.11.1 リスト

“id” パラメータが指定されていない場合、イメージタイプのリストが返されます。

GET /webadmin/rest/imagetypes/	
Response	
<pre>[{ "id": "123", "imagetype": "ABC" }, { "id": "789", "imagetype": "XYZ" }]</pre>	

10.3.11.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのイメージタイプのデータを返します。返されるデータには、すべてのイメージタイプの属性と、現在ログインしている Web サ

イト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/imagetypes/?id=ID	
id	イメージタイプの ID (必須)
Response	
<pre>{ "id": "1", "imagetype": "Thumbnails", "parenttype": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Website Editors", "creators_type": "", "creators_users": "", "developers_group": "Website Developers", "developers_type": "", "developers_users": "", "editors_group": "Website Editors", "editors_type": "", "editors_users": "", "publishers_group": "Website Editors", "publishers_type": "", "publishers_users": "", "administrators_group": "Website Administrators", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.11.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいイメージタイプを作成します。

“id” パラメータが指定されている場合、その ID のイメージタイプがコピーされ、その属性が (部分的に) POST されたデータによって上書きされます。

POST リクエストは、作成されたイメージタイプのデータを返します。返されるデータには、すべてのイメージタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/imagetypes/	
id	コピーするイメージタイプの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのイメージタイプの属性、または任意の



	数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
{ "id": "123", "imagetype": "XYZ" }	
Response	
{ "id": "789", "imagetype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }	

10.3.11.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のイメージタイプを更新します。

PUT リクエストは、更新されたイメージタイプのデータを返します。返されるデータには、すべてのイメージタイプの属性と、現在ログインしている **Web** サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/imagetypes/	
id	更新するイメージタイプの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのイメージタイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "parenttype": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "imagetype": "ABC", "parenttype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.11.5 削除

DELETE リクエストは、指定された ID の既存のイメージタイプを削除します。

DELETE リクエストは、削除されたイメージタイプのデータを返します。返されるデータには、すべてのイメージタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/imagetypes/	
id	イメージタイプの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "imagetype": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.12 ファイルフォーマット

“/webadmin/rest/fileformats/” REST API 関数は、1 つ以上のファイルフォーマットのすべてのデータを返し、ファイルフォーマットを作成、更新、削除します。

10.3.12.1 リスト

“id” パラメータが指定されていない場合、ファイルフォーマットのリストが返されます。

GET /webadmin/rest/fileformats/	
Response	
<pre>[{ "id": "123", "filenameextension": "ABC" }, { "id": "789", "filenameextension": "XYZ" }]</pre>	

10.3.12.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのファイルフォーマットのデータを返します。返されるデータには、すべてのファイルフォーマットの属性が含まれます。

GET /webadmin/rest/fileformats/?id=ID	
id	ファイルフォーマットの ID (必須)
Response	
<pre>{ "id": "1", "filenameextension": "pdf" }</pre>	

```
}

```

10.3.12.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいファイルフォーマットを作成します。

“id” パラメータが指定されている場合、その ID のファイルフォーマットがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたファイルフォーマットのデータを返します。返されるデータには、すべてのファイルフォーマットの属性が含まれます。

POST /webadmin/rest/fileformats/	
id	コピーするファイルフォーマットの ID（任意）
.....	Web コンテンツ管理システムで使用するすべてのファイルフォーマットの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "filenameextension": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "filenameextension": "XYZ" }</pre>	

10.3.12.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のファイルフォーマットを更新します。

PUT リクエストは、更新されたファイルフォーマットのデータを返します。返されるデータには、すべてのファイルフォーマットの属性が含まれます。

PUT /webadmin/rest/fileformats/	
id	更新するファイルフォーマットの ID（必須）
.....	Web コンテンツ管理システムで使用するすべてのファイルフォーマットの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "filenameextension": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "filenameextension": "XYZ" }</pre>	

10.3.12.5 削除

DELETE リクエストは、指定された ID の既存のファイルフォーマットを削除します。

DELETE リクエストは、削除されたファイルフォーマットのデータを返します。返されるデータには、すべてのファイルフォーマットの属性が含まれます。

DELETE /webadmin/rest/fileformats/	
id	ファイルフォーマットの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "filenameextension": "ABC" }</pre>	

10.3.13 ファイルグループ

“/webadmin/rest/filegroups/” REST API 関数は、1 つ以上のファイルグループのすべてのデータを返し、ファイルグループを作成、更新、削除します。

10.3.13.1 リスト

“id” パラメータが指定されていない場合、ファイルグループのリストが返されます。

GET /webadmin/rest/filegroups/	
Response	
<pre>[{ "id": "123", "filegroup": "ABC" }, { "id": "789", "filegroup": "XYZ" }]</pre>	

10.3.13.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのファイルグループのデータを返します。返されるデータには、すべてのファイルグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/filegroups/?id=ID	
id	ファイルグループの ID (必須)
Response	
<pre>{ "id": "1", "filegroup": "Careers", ... }</pre>	

```

"parentgroup": "",

"template": "",
"stylesheet": "",
"title_prefix": "",
"title_suffix": "",
"doctype": "",

"users_group": "",
"users_type": "",
"users_users": "",
"creators_group": "",
"creators_type": "",
"creators_users": "",
"developers_group": "Human Resources",
"developers_type": "",
"developers_users": "",
"editors_group": "Human Resources",
"editors_type": "",
"editors_users": "",
"publishers_group": "Human Resources",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "Website Administrators",
"administrators_type": "",
"administrators_users": "",

"user": true,
"creator": true,
"developer": true,
"editor": true,
"publisher": true,
"administrator": true
}

```

10.3.13.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいファイルグループを作成します。

“id” パラメータが指定されている場合、その ID のファイルグループがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたファイルグループのデータを返します。返されるデータには、すべてのファイルグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/filegroups/	
id	コピーするファイルグループの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべてのファイルグループの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "filegroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "789",</pre>	

```

    "filegroup": "XYZ",
    ....
    "user": true,
    "creator": true,
    "editor": true,
    "publisher": true,
    "developer": true,
    "administrator": true
  }

```

10.3.13.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のファイルグループを更新します。

PUT リクエストは、更新されたファイルグループのデータを返します。返されるデータには、すべてのファイルグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/filegroups/	
id	更新するファイルグループの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのファイルグループの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre> { "id": "123", "parentgroup": "XYZ" } </pre>	
Response	
<pre> { "id": "123", "filegroup": "ABC", "parentgroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true } </pre>	

10.3.13.5 削除

DELETE リクエストは、指定された ID の既存のファイルグループを削除します。

DELETE リクエストは、削除されたファイルグループのデータを返します。返されるデータには、すべてのファイルグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/filegroups/	
id	ファイルグループの ID (必須)
Request	
<pre> { </pre>	

<pre>"id": "123"</pre>
}
Response
<pre>{ "id": "123", "filegroup": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>

10.3.14 ファイルタイプ

“/webadmin/rest/filetypes/” REST API 関数は、1 つ以上のファイルタイプのすべてのデータを返し、ファイルタイプを作成、更新、削除します。

10.3.14.1 リスト

“id” パラメータが指定されていない場合、ファイルタイプのリストが返されます。

GET /webadmin/rest/filetypes/
Response
<pre>[{ "id": "123", "filetype": "ABC" }, { "id": "789", "filetype": "XYZ" }]</pre>

10.3.14.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのファイルタイプのデータを返します。返されるデータには、すべてのファイルタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/filetypes/?id=ID	
id	ファイルタイプの ID (必須)
Response	
<pre>{ "id": "1", "filetype": "Instruction Manuals", "parenttype": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": ""</pre>	

```

"doctype": "",

"users_group": "",
"users_type": "",
"users_users": "",
"creators_group": "Website Editors",
"creators_type": "",
"creators_users": "",
"developers_group": "Website Developers",
"developers_type": "",
"developers_users": "",
"editors_group": "Website Editors",
"editors_type": "",
"editors_users": "",
"publishers_group": "Website Editors",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "Website Administrators",
"administrators_type": "",
"administrators_users": "",

"user": true,
"creator": true,
"developer": true,
"editor": true,
"publisher": true,
"administrator": true
}

```

10.3.14.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいファイルタイプを作成します。

“id” パラメータが指定されている場合、その ID のファイルタイプがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたファイルタイプのデータを返します。返されるデータには、すべてのファイルタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/filetypes/	
id	コピーするファイルタイプの ID（任意）
.....	Web コンテンツ管理システムで使用するすべてのファイルタイプの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "filetype": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "filetype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true,</pre>	

```
{
  "developer": true,
  "administrator": true
}
```

10.3.14.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のファイルタイプを更新します。

PUT リクエストは、更新されたファイルタイプのデータを返します。返されるデータには、すべてのファイルタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/filetypes/	
id	更新するファイルタイプの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのファイルタイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "parenttype": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "filetype": "ABC", "parenttype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.14.5 削除

DELETE リクエストは、指定された ID の既存のファイルタイプを削除します。

DELETE リクエストは、削除されたファイルタイプのデータを返します。返されるデータには、すべてのファイルタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/filetypes/	
id	ファイルタイプの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123",</pre>	

```

    "filetype": "ABC",
    ....
    "user": true,
    "creator": true,
    "editor": true,
    "publisher": true,
    "developer": true,
    "administrator": true
  }

```

10.3.15 リンクグループ

“/webadmin/rest/linkgroups/” REST API 関数は、1 つ以上のリンクグループのすべてのデータを返し、リンクグループを作成、更新、削除します。

10.3.15.1 リスト

“id” パラメータが指定されていない場合、リンクグループのリストが返されます。

GET /webadmin/rest/linkgroups/	
Response	
<pre> [{ "id": "123", "linkgroup": "ABC" }, { "id": "789", "linkgroup": "XYZ" }] </pre>	

10.3.15.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのリンクグループのデータを返します。返されるデータには、すべてのリンクグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/linkgroups/?id=ID	
id	リンクグループの ID (必須)
Response	
<pre> { "id": "1", "linkgroup": "Partners", "parentgroup": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "", </pre>	

```

"creators_type": "",
"creators_users": "",
"developers_group": "Website Developers",
"developers_type": "",
"developers_users": "",
"editors_group": "Website Editors",
"editors_type": "",
"editors_users": "",
"publishers_group": "Website Publishers",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "Website Administrators",
"administrators_type": "",
"administrators_users": "",

"user": true,
"creator": true,
"developer": true,
"editor": true,
"publisher": true,
"administrator": true
}

```

10.3.15.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいリンクグループを作成します。

“id” パラメータが指定されている場合、その ID のリンクグループがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたリンクグループのデータを返します。返されるデータには、すべてのリンクグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/linkgroups/	
id	コピーするリンクグループの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべてのリンクグループの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "linkgroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "linkgroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.15.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のリンクグループを更新します。

PUT リクエストは、更新されたリンクグループのデータを返します。返されるデータには、すべてのリンクグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/linkgroups/	
id	更新するリンクグループの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのリンクグループの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "parentgroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "linkgroup": "ABC", "parentgroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.15.5 削除

DELETE リクエストは、指定された ID の既存のリンクグループを削除します。

DELETE リクエストは、削除されたリンクグループのデータを返します。返されるデータには、すべてのリンクグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/linkgroups/	
id	リンクグループの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "linkgroup": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, }</pre>	

```
{
  "developer": true,
  "administrator": true
}
```

10.3.16 リンクタイプ

“/webadmin/rest/linktypes/” REST API 関数は、1 つ以上のリンクタイプのすべてのデータを返し、リンクタイプを作成、更新、削除します。

10.3.16.1 リスト

“id” パラメータが指定されていない場合、リンクタイプのリストが返されます。

GET /webadmin/rest/linktypes/	
Response	
<pre>[{ "id": "123", "linktype": "ABC" }, { "id": "789", "linktype": "XYZ" }]</pre>	

10.3.16.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのリンクタイプのデータを返します。返されるデータには、すべてのリンクタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/linktypes/?id=ID	
id	リンクタイプの ID (必須)
Response	
<pre>{ "id": "1", "linktype": "Product Websites", "parenttype": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Website Editors", "creators_type": "", "creators_users": "", "developers_group": "Website Developers", "developers_type": "", "developers_users": "", "editors_group": "Website Editors",</pre>	

```

"editors_type": "",
"editors_users": "",
"publishers_group": "Website Editors",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "Website Administrators",
"administrators_type": "",
"administrators_users": "",

"user": true,
"creator": true,
"developer": true,
"editor": true,
"publisher": true,
"administrator": true
}

```

10.3.16.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいリンクタイプを作成します。

“id” パラメータが指定されている場合、その ID のリンクタイプがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたリンクタイプのデータを返します。返されるデータには、すべてのリンクタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/linktypes/	
id	コピーするリンクタイプの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべてのリンクタイプの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "linktype": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "linktype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.16.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のリンクタイプを更新します。

PUT リクエストは、更新されたリンクタイプのデータを返します。返されるデータには、すべてのリンクタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/linktypes/	
id	更新するリンクタイプの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのリンクタイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "parenttype": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "linktype": "ABC", "parenttype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.16.5 削除

DELETE リクエストは、指定された ID の既存のリンクタイプを削除します。

DELETE リクエストは、削除されたリンクタイプのデータを返します。返されるデータには、すべてのリンクタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/linktypes/	
id	リンクタイプの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "linktype": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.17 コンテンツバージョン

“/webadmin/rest/versions/” REST API 関数は、1 つ以上のコンテンツバージョンのすべてのデータを返し、コンテンツバージョンを作成、更新、削除します。

10.3.17.1 リスト

“id” パラメータが指定されていない場合、コンテンツバージョンのリストが返されます。

GET /webadmin/rest/versions/	
Response	
<pre>[{ "id": "123", "version": "ABC", }, { "id": "789", "version": "XYZ", }]</pre>	

10.3.17.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのコンテンツバージョンのデータを返します。返されるデータには、すべてのコンテンツバージョンの属性が含まれます。

GET /webadmin/rest/versions/?id=ID	
id	コンテンツバージョンの ID (必須)
Response	
<pre>{ "id": "1", "version": "Danish", "currencies": "2" }</pre>	

10.3.17.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいコンテンツバージョンを作成します。

“id” パラメータが指定されている場合、その ID のコンテンツバージョンがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたコンテンツバージョンのデータを返します。返されるデータには、すべてのコンテンツバージョンの属性が含まれます。

POST /webadmin/rest/versions/	
id	コピーするコンテンツバージョンの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのコンテンツバージョンの属性、または

	任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "version": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "version": "XYZ", }</pre>	

10.3.17.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツバージョンを更新します。

PUT リクエストは、更新されたコンテンツバージョンのデータを返します。返されるデータには、すべてのコンテンツバージョンの属性が含まれます。

PUT /webadmin/rest/versions/	
id	更新するコンテンツバージョンの ID（必須）
.....	Web コンテンツ管理システムで使用するすべてのコンテンツバージョンの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "currencies": "1" }</pre>	
Response	
<pre>{ "id": "123", "version": "ABC", "currencies": "1" }</pre>	

10.3.17.5 削除

DELETE リクエストは、指定された ID の既存のコンテンツバージョンを削除します。

DELETE リクエストは、削除されたコンテンツバージョンのデータを返します。返されるデータには、すべてのコンテンツバージョンの属性が含まれます。

DELETE /webadmin/rest/versions/	
id	コンテンツバージョンの ID（必須）
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "version": "ABC", }</pre>	

10.3.18 コンテンツバンドル

“/webadmin/rest/contentbundles/” REST API 関数は、1 つ以上のコンテンツバンドルのすべてのデータを返し、コンテンツバンドルを更新、削除します。

10.3.18.1 リスト

“id” パラメータが指定されていない場合、コンテンツバンドルのリストが返されます。

GET /webadmin/rest/contentbundles/	
Response	
<pre>[{ "contentbundle": "ABC" }, { "contentbundle": "XYZ", }]</pre>	

10.3.18.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのコンテンツバンドルのデータを返します。返されるデータには、すべてのコンテンツバンドルの属性が含まれます。

GET /webadmin/rest/contentbundles/?id=ID	
id	コンテンツバンドルの ID（必須）
Response	
<pre>{ "contentbundle": "ABC" }</pre>	

10.3.18.3 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツバンドルを更新します。

PUT リクエストは、更新されたコンテンツバンドルのデータを返します。返されるデータには、すべてのコンテンツバンドルの属性が含まれます。

PUT /webadmin/rest/contentbundles/	
id	更新するコンテンツバンドルの ID（必須）
.....	Web コンテンツ管理システムで使用するすべてのコンテンツバンドルの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "ABC", "contentbundle": "XYZ" }</pre>	
Response	
<pre>{ "contentbundle": "XYZ" }</pre>	

```
}

```

10.3.18.4 削除

DELETE リクエストは、指定された ID の既存のコンテンツバンドルを削除します。

DELETE リクエストは、削除されたコンテンツバンドルのデータを返します。返されるデータには、すべてのコンテンツバンドルの属性が含まれます。

DELETE /webadmin/rest/contentbundles/	
id	コンテンツバンドルの ID (必須)
Request	
<pre>{ "contentbundle": "ABC" }</pre>	
Response	
<pre>{ "contentbundle": "ABC" }</pre>	

10.3.19 コンテンツパッケージ

“/webadmin/rest/contentpackages/” REST API 関数は、1 つ以上のコンテンツパッケージのすべてのデータを返し、コンテンツパッケージを更新、削除します。

10.3.19.1 リスト

“id” パラメータが指定されていない場合、コンテンツパッケージのリストが返されます。

GET /webadmin/rest/contentpackages/	
Response	
<pre>[{ "contentpackage": "ABC" }, { "contentpackage": "XYZ", }]</pre>	

10.3.19.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのコンテンツパッケージのデータを返します。返されるデータには、すべてのコンテンツパッケージの属性が含まれます。

GET /webadmin/rest/contentpackages/?id=ID	
id	コンテンツパッケージの ID (必須)
Response	
<pre>{ "contentpackage": "ABC" }</pre>	

10.3.19.3 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツパッケージを更新します。

PUT リクエストは、更新されたコンテンツパッケージのデータを返します。返されるデータには、すべてのコンテンツパッケージの属性が含まれます。

PUT /webadmin/rest/contentpackages/	
id	更新するコンテンツパッケージの ID (必須)
.....	Web コンテンツ管理システムで使用されるすべてのコンテンツパッケージの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "ABC", "contentpackage": "XYZ" }</pre>	
Response	
<pre>{ "contentpackage": "XYZ" }</pre>	

10.3.19.4 削除

DELETE リクエストは、指定された ID の既存のコンテンツパッケージを削除します。

DELETE リクエストは、削除されたコンテンツパッケージのデータを返します。返されるデータには、すべてのコンテンツパッケージの属性が含まれます。

DELETE /webadmin/rest/contentpackages/	
id	コンテンツパッケージ ID (必須)
Request	
<pre>{ "contentpackage": "ABC" }</pre>	
Response	
<pre>{ "contentpackage": "ABC" }</pre>	

10.3.20 ユーザグループ

“/webadmin/rest/usergroups/” REST API 関数は、1 つ以上のユーザグループのすべてのデータを返し、ユーザグループを作成、更新、削除します。

10.3.20.1 リスト

“id” パラメータが指定されていない場合、ユーザグループのリストが返されます。

GET /webadmin/rest/usergroups/	
Response	
<pre>[{ "id": "123", "usergroup": "ABC" }</pre>	

```

}
,
.....
,
{
  "id": "789",
  .....
  "usergroup": "XYZ"
}
]

```

10.3.20.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのユーザグループのデータを返します。返されるデータには、すべてのユーザグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/usergroups/?id=ID	
id	ユーザグループの ID (必須)
Response	
<pre> { "id": "1", "usergroup": "Event Managers", "parentgroup": "", "login_page": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "", "creators_type": "", "creators_users": "", "editors_group": "", "editors_type": "", "editors_users": "", "publishers_group": "", "publishers_type": "", "publishers_users": "", "administrators_group": "", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true } </pre>	

10.3.20.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいユーザグループを作成します。

“id” パラメータが指定されている場合、その ID のユーザグループがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたユーザグループのデータを返します。返されるデータには、すべてのユーザグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作

成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/usergroups/	
id	コピーするユーザグループの ID (任意)
.....	Web コンテンツ管理システムで使用するすべてのユーザグループの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "usergroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "usergroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.20.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のユーザグループを更新します。

PUT リクエストは、更新されたユーザグループのデータを返します。返されるデータには、すべてのユーザグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/usergroups/	
id	更新するユーザグループの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのユーザグループの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "parentgroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "usergroup": "ABC", "parentgroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

```
}

```

10.3.20.5 削除

DELETE リクエストは、指定された ID の既存のユーザグループを削除します。

DELETE リクエストは、削除されたユーザグループのデータを返します。返されるデータには、すべてのユーザグループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/usergroups/	
id	ユーザグループの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "usergroup": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.21 ユーザタイプ

“/webadmin/rest/usertypes/” REST API 関数は、1 つ以上のユーザタイプのすべてのデータを返し、ユーザタイプを作成、更新、削除します。

10.3.21.1 リスト

“id” パラメータが指定されていない場合、ユーザタイプのリストが返されます。

GET /webadmin/rest/usertypes/	
Response	
<pre>[{ "id": "123", "usertype": "ABC" }, { "id": "789", "usertype": "XYZ" }]</pre>	

10.3.21.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのユーザタイプのデータを返します。返されるデータには、すべてのユーザタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/usertypes/?id=ID	
id	ユーザタイプの ID（必須）
Response	
<pre>{ "id": "1", "usertype": "Senior Management", "parenttype": "", "login_page": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "", "creators_type": "", "creators_users": "", "editors_group": "", "editors_type": "", "editors_users": "", "publishers_group": "", "publishers_type": "", "publishers_users": "", "administrators_group": "", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.21.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいユーザタイプを作成します。

“id” パラメータが指定されている場合、その ID のユーザタイプがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたユーザタイプのデータを返します。返されるデータには、すべてのユーザタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/usertypes/	
id	コピーするユーザタイプの ID（任意）
.....	Web コンテンツ管理システムで使用するすべてのユーザタイプの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	

<pre>{ "id": "123", "usertype": "XYZ" }</pre>
Response
<pre>{ "id": "789", "usertype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>

10.3.21.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のユーザタイプを更新します。

PUT リクエストは、更新されたユーザタイプのデータを返します。返されるデータには、すべてのユーザタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/usertypes/	
id	更新するユーザタイプの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのユーザタイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "parenttype": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "usertype": "ABC", "parenttype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.21.5 削除

DELETE リクエストは、指定された ID の既存のユーザタイプを削除します。

DELETE リクエストは、削除されたユーザタイプのデータを返します。返されるデータには、すべてのユーザタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/usertypes/	
id	コンテンツタイプの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "usertype": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.22 ユーザセグメント

“/webadmin/rest/segments/” REST API 関数は、1 つ以上のユーザセグメントのすべてのデータを返し、ユーザセグメントを作成、更新、削除します。

10.3.22.1 リスト

“id” パラメータが指定されていない場合、ユーザセグメントのリストが返されます。

GET /webadmin/rest/segments/	
Response	
<pre>[{ "id": "123", "segmentid": "ABC", }, { "id": "789", "segmentid": "XYZ", }]</pre>	

10.3.22.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのユーザセグメントのデータを返します。返されるデータには、すべてのユーザセグメントの属性が含まれます。

GET /webadmin/rest/segments/?id=ID	
id	ユーザセグメントの ID (必須)
Response	
<pre>{ "id": "1", "created": "", "updated": "", "created by": "", }</pre>	



```
"updated_by": "",

"segmentid": "Christmas",
"segment": "New Christmas period user",
"weight": "100",
"active": "1",
"scheduled": "",
"unscheduled": "",

"datetimefull": "",
"datetimetype": "",
"datetimetypemonth": "",
"datetimetypeday": "",
"datetimetypeweek": "",
"datetimetypeweekday": "",
"datetimetypehour": "",
"datetimetypemin": "",
"datetimetypesec": "",

"clientaddr": "",
"clienthost": "",
"clientuser": "",
"clientuseragent": "",
"clientbrowser": "",
"clientversion": "",
"clientes": "",
"clientesversion": "",
"clientdevice": "",
"clientdeviceid": "",
"clientdeviceversion": "",
"preferredlanguage": "",
"acceptedlanguages": "",

"referrerhost": "",
"referrerpath": "",
"referrerquery": "",
"referrersearchengine": "",
"referrersearchquery": "",
"referrerid": "",
"referrerclass": "",
"referrerdatabase": "",

"requestid": "",
"requestclass": "",
"requestdatabase": "",
"requesthost": "",
"requestpath": "",
"requestquery": "",
"requestmethod": "",
"requestport": "",
"requestprotocol": "",

"title": "",
"name": "",
"organisation": "",
"email": "",
"gender": "",
"age": "",
"birthdate": "",
"birthyear": "",
"birthmonth": "",
"birthday": "",
"notes": "",
"userinfo": "",

"username": "",
"password": "",

"userclass": "",
"usergroup": "",
"usertype": "",
```

```

"scheduled_publish": "",
"scheduled_notify": "",
"scheduled_unpublish": "",
"scheduled_publish_email": "",
"scheduled_notify_email": "",
"scheduled_unpublish_email": "",
"scheduled_last": "",

"invoice_name": "",
"invoice_organisation": "",
"invoice_address": "",
"invoice_postalcode": "",
"invoice_city": "",
"invoice_state": "",
"invoice_country": "",
"invoice_phone": "",
"invoice_fax": "",
"invoice_email": "",
"invoice_website": "",

"delivery_name": "",
"delivery_organisation": "",
"delivery_address": "",
"delivery_postalcode": "",
"delivery_city": "",
"delivery_state": "",
"delivery_country": "",
"delivery_phone": "",
"delivery_fax": "",
"delivery_email": "",
"delivery_website": "",

"card_type": "",
"card_number": "",
"card_issuedmonth": "",
"card_issuedyear": "",
"card_expirymonth": "",
"card_expiryyear": "",
"card_name": "",
"card_postalcode": "",

"geoipcountry": "",
"geoipregion": "",
"geoipcity": "",
"geoippostalcode": "",
"geoiplatitude": "",
"geoiplongitude": "",
"geoiptimezone": "",

"description": "",
"keywords": "",
"shopcart": "",
"wishlist": ""
}

```

10.3.22.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいユーザセグメントを作成します。

“id” パラメータが指定されている場合、その ID のユーザセグメントがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたユーザセグメントのデータを返します。返されるデータには、すべてのユーザセグメントの属性が含まれます。

POST /webadmin/rest/segments/

id	コピーするユーザセグメントの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのユーザセグメントの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "segmentid": "XYZ", "segment": "Segment XYZ", </pre>	
Response	
<pre>{ "id": "789", </pre>	

10.3.22.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のユーザセグメントを更新します。

PUT リクエストは、更新されたユーザセグメントのデータを返します。返されるデータには、すべてのユーザセグメントの属性が含まれます。

PUT /webadmin/rest/segments/	
id	更新するユーザセグメントの ID (必須)
.....	Web コンテンツ管理システムで使用されるすべてのユーザセグメントの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "weight": "100", "active": "1" }</pre>	
Response	
<pre>{ "id": "123", </pre>	

10.3.22.5 削除

DELETE リクエストは、指定された ID の既存のユーザセグメントを削除します。

DELETE リクエストは、削除されたユーザセグメントのデータを返します。返されるデータには、すべてのユーザセグメントの属性が含まれます。

DELETE /webadmin/rest/segments/	
id	ユーザセグメントの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "segmentid": "ABC", "segment": "Segment ABC", }</pre>	

10.3.23 ユーザテスト

“/webadmin/rest/usertests/” REST API 関数は、1 つ以上のユーザテストのすべてのデータを返し、ユーザテストを作成、更新、削除します。

10.3.23.1 リスト

“id” パラメータが指定されていない場合、ユーザテストのリストが返されます。

GET /webadmin/rest/usertests/	
Response	
<pre>[{ "id": "123", "usertest": "ABC", }, { "id": "789", "usertest": "XYZ", }]</pre>	

10.3.23.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのユーザテストのデータを返します。返されるデータには、すべてのユーザテストの属性が含まれます。

GET /webadmin/rest/usertests/?id=ID	
id	ユーザテストの ID (必須)
Response	
<pre>{ "id": "1", "usertest": "Logo", "description": "Test response to new logo.", "variants": "Old\r\nNew", "type": "ab", "confidence": "95", "probability": "100", "visitors": "all", "goals": "", "active": "1", "scheduled": "", }</pre>	

```
"unscheduled": ""
}
```

10.3.23.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいユーザテストを作成します。

“id” パラメータが指定されている場合、その ID のユーザテストがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたユーザテストのデータを返します。返されるデータには、すべてのユーザテストの属性が含まれます。

POST /webadmin/rest/usertests/	
id	コピーするユーザテストの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべてのユーザテストの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "usertest": "Featured", "description": "Home page feature story", }</pre>	
Response	
<pre>{ "id": "789", "usertest": "Featured", "description": "Home page feature story", }</pre>	

10.3.23.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のユーザテストを更新します。

PUT リクエストは、更新されたユーザテストのデータを返します。返されるデータには、すべてのユーザテストの属性が含まれます。

PUT /webadmin/rest/usertests/	
id	更新するユーザテストの ID（必須）
.....	Web コンテンツ管理システムで使用されるすべてのユーザテストの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "confidence": "95", "active": "1" }</pre>	
Response	
<pre>{</pre>	

```
{
  "id": "123",
  ....
  "confidence": "95",
  "active": "1",
  ....
}
```

10.3.23.5 削除

DELETE リクエストは、指定された ID の既存のユーザテストを削除します。

DELETE リクエストは、削除されたユーザテストのデータを返します。返されるデータには、すべてのユーザテストの属性が含まれます。

DELETE /webadmin/rest/usertests/	
id	ユーザテストの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "usertest": "Logo", "description": "Test response to new logo.", }</pre>	

10.3.24 ワークフロー

“/webadmin/rest/workflows/” REST API 関数は、1 つ以上のワークフローのすべてのデータを返し、ワークフローを作成、更新、削除します。

10.3.24.1 リスト

“id” パラメータが指定されていない場合、ワークフローのリストが返されます。

GET /webadmin/rest/workflows/	
Response	
<pre>[{ "id": "123", "title": "ABC", "action": "abc", }, { "id": "789", "title": "XYZ", "action": "xyz", }]</pre>	

10.3.24.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのワークフローアクションのデータを返します。返されるデータには、すべてのワークフローアクションの属性が含まれます。

GET /webadmin/rest/workflows/?id=ID	
id	ワークフローアクションの ID (必須)
Response	
<pre>{ "id": "9", "title": "Two-step approval", "action": "Approve publishing", "fromstate": "Approved", "tostate": "", "usertype": "null", "usergroup": "null", "contentclass": "null", "contenttype": "null", "contentgroup": "null", "version": "null", "notify_email": "null", "contentchanges": "null", "workflow_program": "null", "userrestriction": "null" }</pre>	

10.3.24.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいワークフローアクションを作成します。

“id” パラメータが指定されている場合、その ID のワークフローアクションがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたワークフローアクションのデータを返します。返されるデータには、すべてのワークフローアクションの属性が含まれます。

POST /webadmin/rest/workflows/	
id	コピーするワークフローアクションの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのワークフローアクションの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "title": "XYZ", "action": "xyz", }</pre>	
Response	
<pre>{ "id": "789", "title": "XYZ", "action": "xyz", }</pre>	

10.3.24.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のワークフローアクションを更新します。

PUT リクエストは、更新されたワークフローアクションのデータを返します。返されるデータには、すべてのワークフローアクションの属性が含まれます。

PUT /webadmin/rest/workflows/	
id	更新するワークフローアクションの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのワークフローアクションの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "fromstate": "Approved", "tostate": "" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", "action": "abc", "fromstate": "Approved", "tostate": "", }</pre>	

10.3.24.5 削除

DELETE リクエストは、指定された ID の既存のワークフローアクションを削除します。

DELETE リクエストは、削除されたワークフローアクションのデータを返します。返されるデータには、すべてのワークフローアクションの属性が含まれます。

DELETE /webadmin/rest/workflows/	
id	ワークフローアクションの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", "action": "abc", }</pre>	

10.3.25 コメント

“/webadmin/rest/comments/” REST API 関数は、1 つ以上のコメントのすべてのデータを返し、コメントを作成、更新、削除します。

10.3.25.1 リスト

“id” パラメータが指定されていない場合、コメントのリストが返されます。

GET /webadmin/rest/comments/	
Response	
<pre>[{ "id": "123", }]</pre>	

```

    "title": "ABC",
    ....
  }
  ,
  ....
  ,
  {
    "id": "789",
    ....
    "title": "XYZ",
    ....
  }
]
```

10.3.25.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのコメントのデータを返します。返されるデータには、すべてのコメントの属性が含まれます。

GET /webadmin/rest/comments/?id=ID	
id	コメントの ID（必須）
Response	
<pre> { "id": "15", "created": "2018-04-04 17:58:00", "created_by": "paulgreen", "title": "Job Applications", "content": "Job applications are automatically uploaded to this folder when users apply.", "section": "library", "cid": "", "class": "file", "group": "Careers", "type": "", "package": "", "bundle": "", "version": "", "status": "", "stock": "", "locked": "1" }</pre>	

10.3.25.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいコメントを作成します。

“id” パラメータが指定されている場合、その ID のコメントがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたコメントのデータを返します。返されるデータには、すべてのコメントの属性が含まれます。

POST /webadmin/rest/comments/	
id	コピーするコメントの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべてのコメントの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	

<pre>{ "id": "123", "title": "XYZ", "content": "xyz" }</pre>
Response
<pre>{ "id": "789", "title": "XYZ", "content": "xyz", }</pre>

10.3.25.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコメントを更新します。

PUT リクエストは、更新されたコメントのデータを返します。返されるデータには、すべてのコメントの属性が含まれます。

PUT /webadmin/rest/comments/	
id	更新するコメントの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのコメントの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "title": "XYZ", "content": "xyz" }</pre>	
Response	
<pre>{ "id": "123", "title": "XYZ", "content": "xyz", }</pre>	

10.3.25.5 削除

DELETE リクエストは、指定された ID の既存のコメントを削除します。

DELETE リクエストは、削除されたコメントのデータを返します。返されるデータには、すべてのコメントの属性が含まれます。

DELETE /webadmin/rest/comments/	
Id	コメントの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", }</pre>	

```

    "title": "ABC",
    .....
}

```

10.3.26 プロジェクト

“/webadmin/rest/projects/” REST API 関数は、1 つ以上のプロジェクトのすべてのデータを返し、プロジェクトを作成、更新、削除します。

10.3.26.1 リスト

“id” パラメータが指定されていない場合、プロジェクトのリストが返されます。

GET /webadmin/rest/projects/	
Response	
<pre> [{ "id": "123", "title": "ABC", }, , { "id": "789", "title": "XYZ", }] </pre>	

10.3.26.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのプロジェクトのデータを返します。返されるデータには、すべてのプロジェクトの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/projects/?id=ID	
id	プロジェクトの ID (必須)
Response	
<pre> { "id": "6", "created": "2018-04-04 16:59:39", "updated": "2018-04-04 17:11:16", "closed": "", "deleted": "", "created_by": "admin", "updated_by": "admin", "closed_by": "", "deleted_by": "", "title": "Refresh Product A Presentation", "description": "Review, suggest and implement a refresh of the the presentation of Product A.", "category": "", "status": "", "status_by": "admin", "priority": "", </pre>	



```
"severity": "",
"duration_amount": "2",
"duration_period": "604800",
"started": "",
"ended": "",

"assets_contents": "484,177,182,183,188",
"assets_contentgroups": "",
"assets_contenttypes": "",
"assets_imagegroups": "",
"assets_imagetypes": "",
"assets_filegroups": "",
"assets_filetypes": "",
"assets_linkgroups": "",
"assets_linktypes": "",
"assets_productgroups": "",
"assets_producttypes": "",
"assets_databases": "",
"assets_users": "",
"assets_usergroups": "",
"assets_usertypes": "",
"assets_orders": "",
"assets_sales": "",
"assets_segments": "",
"assets_usertests": "",
"assets_heatmaps": "",
"assets_usage": "",
"assets_clients": "",
"assets_hostinggroups": "",
"assets_hostingtypes": "",
"assets_websites": "",
"assets_packages": "",
"assets_bundles": "",
"assets_other": "",

"users_group": "",
"users_type": "",
"users_users": "",
"creators_group": "",
"creators_type": "",
"creators_users": "",
"editors_group": "",
"editors_type": "",
"editors_users": "",
"publishers_group": "",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "",
"administrators_type": "",
"administrators_users": "",

"categories":
":#EEEEEE:white\r\nBacklog:#E2445C:white\r\nReview:#FDAB3D:white\r\nImplement:#00C875:
white\r\nUpcoming:#0086C0:white\r\nSuggest:#579BFC:white\r\n:A25DDC:white\r\n:#037F4C
:white\r\n:#CAB641:white\r\n:#FFCB00:white\r\n:#333333:white\r\n:#BB3354:white\r\n:#FF
158A:white\r\n:#FF5AC4:white\r\n:#784BD1:white\r\n:#9CD326:white\r\n:#66CCFF:white\r\n
:#808080:white\r\n:#7F5347:white\r\n:#FF642E:white",
"statuses": "To Begin:#EEEEEE:white\r\nIn Progress:#00C875:white\r\nAwaiting
Review:#FDAB3D:white\r\nNot Started:#E2445C:white\r\nAwaiting
Suggestions:#0086C0:white\r\n:#579BFC:white\r\nAwaiting
Implementation:#A25DDC:white\r\nCompleted:#037F4C:white\r\n:#CAB641:white\r\n:#FFCB00:
white\r\n:#333333:white\r\n:#BB3354:white\r\n:#FF158A:white\r\n:#FF5AC4:white\r\n:#784
BD1:white\r\n:#9CD326:white\r\n:#66CCFF:white\r\n:#808080:white\r\n:#7F5347:white\r\n:
#FF642E:white",
"priorities":
":#EEEEEE:white\r\nLow:#00C875:white\r\nMedium:#FDAB3D:white\r\nHigh:#E2445C:white\r\n
:#0086C0:white\r\n:#579BFC:white\r\nCritical:#A25DDC:white\r\nNone:#037F4C:white\r\n:#
CAB641:white\r\n:#FFCB00:white\r\n:#333333:white\r\n:#BB3354:white\r\n:#FF158A:white\r
\r\n:#FF5AC4:white\r\n:#784BD1:white\r\n:#9CD326:white\r\n:#66CCFF:white\r\n:#808080:whi
te\r\n:#7F5347:white\r\n:#FF642E:white",
"severities":
":#EEEEEE:white\r\nMinor:#00C875:white\r\nModerate:#FDAB3D:white\r\nSignificant:#E2445
```

```
C:white\r\n:#0086C0:white\r\n:#579BFC:white\r\nExtensive:#A25DDC:white\r\nNone:#037F4C
:white\r\n:#CAB641:white\r\n:#FFCB00:white\r\n:#333333:white\r\n:#BB3354:white\r\n:#FF
158A:white\r\n:#FF5AC4:white\r\n:#784BD1:white\r\n:#9CD326:white\r\n:#66CCFF:white\r\n
:#808080:white\r\n:#7F5347:white\r\n:#FF642E:white",

    "user": true,
    "creator": true,
    "editor": true,
    "publisher": true,
    "administrator": true
}
```

10.3.26.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいプロジェクトを作成します。

“id” パラメータが指定されている場合、その ID のプロジェクトがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたプロジェクトのデータを返します。返されるデータには、すべてのプロジェクトの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/projects/	
id	コピーするプロジェクトの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべてのプロジェクトの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "title": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.26.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のプロジェクトを更新します。

PUT リクエストは、更新されたプロジェクトのデータを返します。返されるデータには、すべてのプロジェクトの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/projects/	
id	更新するプロジェクトの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのプロジェクトの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "title": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.26.5 削除

DELETE リクエストは、指定された ID の既存のプロジェクトを削除します。

DELETE リクエストは、削除されたプロジェクトのデータを返します。返されるデータには、すべてのプロジェクトの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/projects/	
id	プロジェクトの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.27 プロジェクトタスク

“/webadmin/rest/projecttasks/” REST API 関数は、1 つ以上のプロジェクトタスクのすべてのデータを返し、プロジェクトタスクを作成、更新、削除します。

10.3.27.1 リスト

“project_id” パラメータが指定されており、“id” パラメータが指定されていない場合、プロジェクトタスクのリストが返されます。

GET /webadmin/rest/projecttasks/	
project_id	プロジェクトタスクの ID (必須)
Response	
<pre>[{ "id": "123", "project_id": "456", "title": "ABC", }, { "id": "789", "project_id": "456", "title": "XYZ", }]</pre>	

10.3.27.2 読み取り

“id” が指定されている場合、GET リクエストはそのプロジェクトタスクのデータを返します。返されるデータには、すべてのプロジェクトタスクの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/projecttasks/?id=ID	
id	プロジェクトタスクの ID (必須)
Response	
<pre>{ "id": "15", "project_id": "6", "parent_id": "", "created": "2018-04-04 17:03:43", "updated": "2018-04-04 17:06:20", "closed": "", "deleted": "", "created_by": "admin", "updated_by": "admin", "closed_by": "", "deleted_by": "", "title": "Review Current Presentation", "description": "Review the current presentation and make notes about areas that need to be improved.", "category": "Review", "status": "In Progress", "status_by": "admin", "priority": "Medium", "severity": "Extensive", "duration_amount": "2", "duration_period": "86400", "started": "", "ended": "", "after_tasks": "",</pre>	



```
"before_tasks": "",

"assets_contents": "",
"assets_contentgroups": "",
"assets_contenttypes": "",
"assets_imagegroups": "",
"assets_imagetypes": "",
"assets_filegroups": "",
"assets_filetypes": "",
"assets_linkgroups": "",
"assets_linktypes": "",
"assets_productgroups": "",
"assets_producttypes": "",
"assets_databases": "",
"assets_users": "",
"assets_usergroups": "",
"assets_usertypes": "",
"assets_orders": "",
"assets_sales": "",
"assets_segments": "",
"assets_usertests": "",
"assets_heatmaps": "",
"assets_usage": "",
"assets_clients": "",
"assets_hostinggroups": "",
"assets_hostingtypes": "",
"assets_websites": "",
"assets_packages": "",
"assets_bundles": "",
"assets_other": "",

"users_group": "",
"users_type": "",
"users_users": "",
"creators_group": "",
"creators_type": "",
"creators_users": "",
"editors_group": "",
"editors_type": "",
"editors_users": "",
"publishers_group": "",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "",
"administrators_type": "",
"administrators_users": "",

"user": true,
"creator": true,
"editor": true,
"publisher": true,
"administrator": true
}
```

10.3.27.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいプロジェクトタスクを作成します。

“id” パラメータが指定されている場合、その ID のプロジェクトタスクがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたプロジェクトタスクのデータを返します。返されるデータには、すべてのプロジェクトタスクの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/projecttasks/	
id	コピーするプロジェクトタスク項目の ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのプロジェクトタスクの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "project_id": "456", "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "project_id": "456", "title": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.27.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のプロジェクトタスクを更新します。

PUT リクエストは、更新されたプロジェクトタスクのデータを返します。返されるデータには、すべてのプロジェクトタスクの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/projecttasks/	
id	更新するプロジェクトタスクの ID (必須)
.....	Web コンテンツ管理システムで使用されるすべてのプロジェクトタスクの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "project_id": "456", "title": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.27.5 削除

DELETE リクエストは、指定された ID の既存のプロジェクトタスクを削除、または非公開にします。

DELETE リクエストは、削除/非公開されたプロジェクトタスクのデータを返します。返されるデータには、すべてのプロジェクトタスク目の属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/projecttasks/	
id	プロジェクトタスクの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "project_id": "456", "title": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.28 通貨

“/webadmin/rest/currency/” REST API 関数は、1 つ以上の通貨のすべてのデータを返し、通貨を作成、更新、削除します。

10.3.28.1 リスト

“id” パラメータが指定されていない場合、通貨のリストが返されます。

GET /webadmin/rest/currency/	
Response	
<pre>[{ "id": "123", "title": "ABC", }, { "id": "789", "title": "XYZ", }]</pre>	

10.3.28.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその通貨のデータを返します。返されるデータには、すべての通貨の属性が含まれます。

GET /webadmin/rest/currency/?id=ID	
id	通貨の ID (必須)
Response	
<pre>{ "id": "1", "title": "GBP", "symbol": "&#163;", "rate": "100" }</pre>	

10.3.28.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい通貨を作成します。

“id” パラメータが指定されている場合、その ID の通貨がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された通貨のデータを返します。返されるデータには、すべての通貨の属性が含まれます。

POST /webadmin/rest/currency/	
id	コピーする通貨の ID (任意)
.....	Web コンテンツ管理システムで使用されるすべての通貨の属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "title": "XYZ", }</pre>	

10.3.28.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の通貨を更新します。

PUT リクエストは、更新された通貨のデータを返します。返されるデータには、すべての通貨の属性が含まれます。

PUT /webadmin/rest/currency/	
id	更新する通貨の ID (必須)
.....	Web コンテンツ管理システムで使用されるすべての通貨の属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）

Request
<pre>{ "id": "123", "title": "XYZ" }</pre>
Response
<pre>{ "id": "123", "title": "XYZ", }</pre>

10.3.28.5 削除

DELETE リクエストは、指定された ID の既存の通貨を削除します。

DELETE リクエストは、削除された通貨のデータを返します。返されるデータには、すべての通貨の属性が含まれます。

DELETE /webadmin/rest/currency/	
id	通貨の ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", }</pre>	

10.3.29 商品グループ

“/webadmin/rest/productgroups/” REST API 関数は、1 つ以上の商品グループのすべてのデータを返し、商品グループを作成、更新、削除します。

10.3.29.1 リスト

“id” パラメータが指定されていない場合、商品グループのリストが返されます。

GET /webadmin/rest/productgroups/
Response
<pre>[{ "id": "123", "productgroup": "ABC" }, { "id": "789", "productgroup": "XYZ" }]</pre>

10.3.29.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその商品グループのデータを返します。返されるデータには、すべての商品グループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/productgroups/?id=ID	
id	商品グループの ID（必須）
Response	
<pre>{ "id": "1", "productgroup": "Subscription Services", "parentgroup": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Shop Managers", "creators_type": "", "creators_users": "", "developers_group": "Website Developers", "developers_type": "", "developers_users": "", "editors_group": "Shop Managers", "editors_type": "", "editors_users": "", "publishers_group": "Shop Managers", "publishers_type": "", "publishers_users": "", "administrators_group": "Website Administrators", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.29.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい商品グループを作成します。

“id” パラメータが指定されている場合、その ID の商品グループがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された商品グループのデータを返します。返されるデータには、すべての商品グループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/productgroups/

id	コピーする商品グループの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべての商品グループの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "productgroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "productgroup": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.29.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の商品グループを更新します。

PUT リクエストは、更新された商品グループのデータを返します。返されるデータには、すべての商品グループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/productgroups/	
id	更新する商品グループの ID (必須)
.....	Web コンテンツ管理システムで使用されるすべての商品グループの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "template": "456", "stylesheet": "789" }</pre>	
Response	
<pre>{ "id": "123", "productgroup": "ABC", "parentgroup": "", "template": "456", "stylesheet": "789" "user": true, "creator": true, "editor": true, "publisher": true, "developer": true,</pre>	

```
"administrator": true
}
```

10.3.29.5 削除

DELETE リクエストは、指定された ID の既存の商品グループを削除します。

DELETE リクエストは、削除された商品グループのデータを返します。返されるデータには、すべての商品グループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/productgroups/	
id	商品グループの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "productgroup": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.30 商品タイプ

“/webadmin/rest/producttypes/” REST API 関数は、1 つ以上の商品タイプのすべてのデータを返し、商品タイプを作成、更新、削除します。

10.3.30.1 リスト

“id” パラメータが指定されていない場合、商品タイプのリストが返されます。

GET /webadmin/rest/producttypes/	
Response	
<pre>[{ "id": "123", "producttype": "ABC" }, { "id": "789", "producttype": "XYZ" }]</pre>	

10.3.30.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその商品タイプのデータを返します。返されるデータには、すべての商品タイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/producttypes/?id=ID	
id	商品タイプの ID（必須）
Response	
<pre>{ "id": "1", "producttype": "Digital Services", "parenttype": "", "template": "", "stylesheet": "", "title_prefix": "", "title_suffix": "", "doctype": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Shop Managers", "creators_type": "", "creators_users": "", "developers_group": "Website Developers", "developers_type": "", "developers_users": "", "editors_group": "Shop Managers", "editors_type": "", "editors_users": "", "publishers_group": "Shop Managers", "publishers_type": "", "publishers_users": "", "administrators_group": "Website Administrators", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.30.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい商品タイプを作成します。

“id” パラメータが指定されている場合、その ID の商品タイプがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された商品タイプのデータを返します。返されるデータには、すべての商品タイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/producttypes/

id	コピーする商品タイプの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべての商品タイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "producttype": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "producttype": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.30.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の商品タイプを更新します。

PUT リクエストは、更新された商品タイプのデータを返します。返されるデータには、すべての商品タイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/producttypes/	
id	更新する商品タイプの ID (必須)
.....	Web コンテンツ管理システムで使用されるすべての商品タイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "template": "456", "stylesheet": "789" }</pre>	
Response	
<pre>{ "id": "123", "producttype": "ABC", "parenttype": "", "template": "456", "stylesheet": "789" "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, }</pre>	

```
"administrator": true
}
```

10.3.30.5 削除

DELETE リクエストは、指定された ID の既存の商品タイプを削除します。

DELETE リクエストは、削除された商品タイプのデータを返します。返されるデータには、すべての商品タイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/producttypes/	
id	商品タイプの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "producttype": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "developer": true, "administrator": true }</pre>	

10.3.31 割引

“/webadmin/rest/discounts/” REST API 関数は、1 つ以上の割引のすべてのデータを返し、割引を作成、更新、削除します。

10.3.31.1 リスト

“id” パラメータが指定されていない場合、割引のリストが返されます。

GET /webadmin/rest/discounts/	
Response	
<pre>[{ "id": "123", "title": "ABC", }, { "id": "789", "title": "XYZ", }]</pre>	

10.3.31.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその割引のデータを返します。返されるデータには、すべての割引の属性が含まれます。

GET /webadmin/rest/discounts/?id=ID	
id	割引の ID (必須)
Response	
<pre>{ "id": "1", "title": "10% Off Everything", "country": "", "state": "", "product_id": "", "product_group": "", "product_type": "", "product_weight_from": "0.000", "product_weight_to": "0.000", "product_volume_from": "0.00", "product_volume_to": "0.00", "product_width_from": "0.00", "product_width_to": "0.00", "product_height_from": "0.00", "product_height_to": "0.00", "product_depth_from": "0.00", "product_depth_to": "0.00", "quantity_from": "0", "quantity_to": "0", "total_currency": "", "total_from": "0.00", "total_to": "0.00", "total_weight_from": "0.000", "total_weight_to": "0.000", "total_volume_from": "0.00", "total_volume_to": "0.00", "total_width_from": "0.00", "total_width_to": "0.00", "total_height_from": "0.00", "total_height_to": "0.00", "total_depth_from": "0.00", "total_depth_to": "0.00", "discount_description": "", "discount_type": "general", "discount_quantity": "0", "discount_quantity2": "0", "discount_products": "", "discount_currency": "%", "discount_amount": "10.00", "discount_orderitems": "total", "user_username": "", "user_group": "", "user_type": "", "user_code": "", "period_start": "", "period_end": "" }</pre>	

10.3.31.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい割引を作成します。

“id” パラメータが指定されている場合、その ID の割引がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された割引のデータを返します。返されるデータには、すべての割引の属性が含まれます。

POST /webadmin/rest/discounts/	
id	コピーする割引の ID (任意)
.....	Web コンテンツ管理システムで使用するすべての割引の属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "title": "XYZ", </pre>	

10.3.31.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の割引を更新します。

PUT リクエストは、更新された割引のデータを返します。返されるデータには、すべてのフィールドの属性が含まれます。

PUT /webadmin/rest/discounts/	
id	更新する割引の ID (必須)
.....	Web コンテンツ管理システムで使用するすべての割引の属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", </pre>	

10.3.31.5 削除

DELETE リクエストは、指定された ID の既存の割引を削除します。

DELETE リクエストは、削除された割引のデータを返します。返されるデータには、すべての割引の属性が含まれます。

DELETE /webadmin/rest/discounts/	
id	割引の ID (必須)
Request	
{	

<code>"id": "123"</code>
<code>}</code>
Response
<code>{</code>
<code> "id": "123",</code>
<code> "title": "ABC",</code>
<code> </code>
<code>}</code>

10.3.32 送料

“/webadmin/rest/shipping/” REST API 関数は、1 つ以上の送料のすべてのデータを返し、送料を作成、更新、削除します。

10.3.32.1 リスト

“id” パラメータが指定されていない場合、送料のリストが返されます。

GET /webadmin/rest/shipping/
Response
<code>[</code>
<code>{</code>
<code> "id": "123",</code>
<code> "title": "ABC",</code>
<code> </code>
<code>}</code>
<code>,</code>
<code>.....</code>
<code>,</code>
<code>{</code>
<code> "id": "789",</code>
<code> "title": "XYZ",</code>
<code> </code>
<code>}</code>
<code>]</code>

10.3.32.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその送料のデータを返します。返されるデータには、すべての送料の属性が含まれます。

GET /webadmin/rest/shipping/?id=ID	
id	送料の ID (必須)
Response	
<pre>{ "id": "1", "title": "Royal Mail 2nd Class Small Parcel", "country": "", "state": "", "product_id": "", "product_group": "Physical Products", "product_type": "", "product_weight_from": "0.000", "product_weight_to": "0.000", "product_volume_from": "0.00", "product_volume_to": "0.00", "product_width_from": "0.00", "product_width_to": "0.00", "product_height_from": "0.00", "product_height_to": "0.00", "product_depth_from": "0.00", "product_depth_to": "0.00", "quantity_from": "0", "quantity_to": "0"</pre>	

```
{
  "total_currency": "",
  "total_from": "0.00",
  "total_to": "0.00",
  "total_weight_from": "0.000",
  "total_weight_to": "2.000",
  "total_volume_from": "0.00",
  "total_volume_to": "0.00",
  "total_width_from": "0.00",
  "total_width_to": "0.00",
  "total_height_from": "0.00",
  "total_height_to": "0.00",
  "total_depth_from": "0.00",
  "total_depth_to": "0.00",
  "ship_description": "Basic Shipping",
  "ship_currency": "",
  "ship_order": "3.00",
  "ship_item": "0.00",
  "ship_percent": "0.00",
  "ship_total": "0.00"
}
```

10.3.32.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい送料を作成します。

“id” パラメータが指定されている場合、その ID の送料がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された送料のデータを返します。返されるデータには、すべての送料の属性が含まれます。

POST /webadmin/rest/shipping/	
id	コピーする送料の ID（任意）
.....	Web コンテンツ管理システムで使用されるすべての送料の属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
{ "id": "123", "title": "XYZ" }	
Response	
{ "id": "789", "title": "XYZ", }	

10.3.32.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の送料を更新します。

PUT リクエストは、更新された送料のデータを返します。返されるデータには、すべての送料の属性が含まれます。

PUT /webadmin/rest/shipping/	
id	更新する送料の ID（必須）

.....	Web コンテンツ管理システムで使用するすべての送料の属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", </pre>	

10.3.32.5 削除

DELETE リクエストは、指定された ID の既存の送料を削除します。

DELETE リクエストは、削除された送料のデータを返します。返されるデータには、すべての送料の属性が含まれます。

DELETE /webadmin/rest/shipping/	
id	送料の ID（必須）
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", </pre>	

10.3.33 税率

“/webadmin/rest/tax/” REST API 関数は、1 つ以上の税率のすべてのデータを返し、税率を作成、更新、削除します。

10.3.33.1 リスト

“id” パラメータが指定されていない場合、税率のリストが返されます。

GET /webadmin/rest/tax/	
Response	
<pre>[{ "id": "123", "title": "ABC", }, { "id": "789", "title": "XYZ", </pre>	

```
}
]
```

10.3.33.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその税率のデータを返します。返されるデータには、すべての税率の属性が含まれます。

GET /webadmin/rest/tax/?id=ID	
id	税率の ID (必須)
Response	
<pre>{ "id": "1", "title": "United Kingdom VAT", "country": "United Kingdom", "state": "", "product_id": "", "product_group": "", "product_type": "", "product_weight_from": "0.000", "product_weight_to": "0.000", "product_volume_from": "0.00", "product_volume_to": "0.00", "product_width_from": "0.00", "product_width_to": "0.00", "product_height_from": "0.00", "product_height_to": "0.00", "product_depth_from": "0.00", "product_depth_to": "0.00", "quantity_from": "0", "quantity_to": "0", "total_currency": "", "total_from": "0.00", "total_to": "0.00", "total_weight_from": "0.000", "total_weight_to": "0.000", "total_volume_from": "0.00", "total_volume_to": "0.00", "total_width_from": "0.00", "total_width_to": "0.00", "total_height_from": "0.00", "total_height_to": "0.00", "total_depth_from": "0.00", "total_depth_to": "0.00", "tax_description": "United Kingdom VAT 20%", "tax_currency": "", "tax_order": "0.00", "tax_item": "0.00", "tax_percent": "20.00", "tax_total": "0.00" }</pre>	

10.3.33.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい税率を作成します。

“id” パラメータが指定されている場合、その ID の税率がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された税率のデータを返します。返されるデータには、すべての税率の属性が含まれます。

```
POST /webadmin/rest/tax/
```

id	コピーする税率の ID (任意)
.....	Web コンテンツ管理システムで使用するすべての税率の属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
{ "id": "123", "title": "XYZ" }	
Response	
{ "id": "789", "title": "XYZ", }	

10.3.33.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の税率を更新します。

PUT リクエストは、更新された税率のデータを返します。返されるデータには、すべての税率の属性が含まれます。

PUT /webadmin/rest/tax/	
id	更新する税率の ID (必須)
.....	Web コンテンツ管理システムで使用するすべての税率の属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
{ "id": "123", "title": "XYZ" }	
Response	
{ "id": "123", "title": "ABC", }	

10.3.33.5 削除

DELETE リクエストは、指定された ID の既存の税率を削除します。

DELETE リクエストは、削除された税率のデータを返します。返されるデータには、すべての税率の属性が含まれます。

DELETE /webadmin/rest/tax/	
id	税率の ID (必須)
Request	
{ "id": "123" }	
Response	
{ "id": "123",	

```
"title": "ABC",
.....
}
```

10.3.34 注文

“/webadmin/rest/orders/” REST API 関数は、1 つ以上の注文のすべてのデータを返し、注文を作成、更新、削除します。

10.3.34.1 リスト

“id” パラメータが指定されていない場合、注文のリストが返されます。

GET /webadmin/rest/orders/	
Response	
<pre>[{ "id": "123", }, , { "id": "789", }]</pre>	

10.3.34.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその注文のデータを返します。返されるデータには、すべての注文の属性が含まれます。

GET /webadmin/rest/orders/?id=ID	
id	注文の ID (必須)
Response	
<pre>{ "id": "1", "user_id": "5", "created": "2013-08-20 12:08:48", "updated": "2013-08-20 12:08:48", "published": "", "created_by": "johnsmith", "updated_by": "johnsmith", "published_by": "", "paid": "", "status": "", "status_by": "", "checkedout": "", "revision": "", "card_type": "VISA", "card_number": "4213451248543248", "card_issuedmonth": "", "card_issuedyear": "", "card_expirymonth": "04", "card_expiryyear": "2014", "card_name": "Mr. John Smith", "card_cvc": "564", "card_issue": "", "card_postalcode": "",</pre>	

```
{
  "delivery_name": "John Smith",
  "delivery_organisation": "",
  "delivery_address": "10 Melbourne Street",
  "delivery_postalcode": "E17 2SW",
  "delivery_city": "London",
  "delivery_state": "",
  "delivery_country": "United Kingdom",
  "delivery_phone": "",
  "delivery_fax": "",
  "delivery_email": "admin@asbrusoft.com",
  "delivery_website": "",

  "invoice_name": "",
  "invoice_organisation": "",
  "invoice_address": "",
  "invoice_postalcode": "",
  "invoice_city": "",
  "invoice_state": "",
  "invoice_country": "",
  "invoice_phone": "",
  "invoice_fax": "",
  "invoice_email": "",
  "invoice_website": "",

  "order_quantity": "1",
  "order_currency": "&#163;",
  "order_currencytitle": "GBP",
  "discount_description": "",
  "discount_total": "0.00",
  "discount_total_base": "0.00",
  "order_subtotal": "100.00",
  "order_subtotal_base": "100.00",
  "tax_description": "<span class=\"tax\">United Kingdom VAT 17.5%</span>",
  "tax_total": "17.50",
  "tax_total_base": "17.50",
  "shipping_description": "<span class=\"shipping\">1 x Basic  
Shipping</span>",
  "shipping_total": "3.50",
  "shipping_total_base": "3.50",
  "order_total": "121.00",
  "order_total_base": "121.00",
  "order_availability": "",

  "usersegments": "",
  "usertests": ""
}
```

10.3.34.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい注文を作成します。

“id” パラメータが指定されている場合、その ID の注文がコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された注文のデータを返します。返されるデータには、すべての注文の属性が含まれます。

POST /webadmin/rest/orders/	
id	コピーする注文の ID（任意）
.....	Web コンテンツ管理システムで使用されるすべての注文の属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	

<pre>{ "id": "123", "user_id": "456" }</pre>
Response
<pre>{ "id": "789", "user_id": "456" </pre>

10.3.34.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の注文を更新します。

PUT リクエストは、更新された注文のデータを返します。返されるデータには、すべての注文の属性が含まれます。

PUT /webadmin/rest/orders/	
id	更新する注文の ID（必須）
.....	Web コンテンツ管理システムで使用するすべての注文の属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "paid": "2020-01-31 23:59:59" }</pre>	
Response	
<pre>{ "id": "123", "paid": "2020-01-31 23:59:59", </pre>	

10.3.34.5 削除

DELETE リクエストは、指定された ID の既存の注文を削除します。

DELETE リクエストは、削除された注文のデータを返します。返されるデータには、すべての注文の属性が含まれます。

DELETE /webadmin/rest/orders/	
id	注文の ID（必須）
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "user_id": "456" </pre>	

10.3.35 注文アイテム

“/webadmin/rest/orderitems/” REST API 関数は、1 つ以上の注文アイテムのすべてのデータを返し、注文を作成、更新、削除します。

10.3.35.1 リスト

“order_id” パラメータが指定されており、“id”パラメータが指定されていない場合、注文アイテムのリストが返されます。

GET /webadmin/rest/orderitems/	
order_id	注文の ID（必須）
Response	
<pre>[{ "id": "123", "order_id": "456", }, { "id": "789", "order_id": "456", }]</pre>	

10.3.35.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはその注文アイテムのデータを返します。返されるデータには、すべての注文アイテムの属性が含まれます。

GET /webadmin/rest/orderitems/?id=ID	
id	注文アイテムの ID（必須）
Response	
<pre>{ "id": "1", "order_id": "1", "item_quantity": "1", "discount_description": "", "discount_total": "0.00", "discount_total_base": "", "item_subtotal": "100.00", "item_subtotal_base": "", "tax_description": "United Kingdom VAT 17.5%", "tax_total": "17.50", "tax_total_base": "", "shipping_description": "1 x Basic Shipping", "shipping_total": "3.50", "shipping_total_base": "", "item_total": "100.00", "item_total_base": "", "product_id": "169", "product_code": "Item Id: #54PRD001", "product_currencytitle": "GBP", "product_currency": "&#163;", "product_price": "100", "product_period": "", "product_weight": "1", "product_volume": "3", "product_width": "100",</pre>	

```

"product_height": "200",
"product_depth": "300",
"product_brand": "",
"product_colour": "",
"product_size": "",
"product_stock": "400",
"product_comment": "In Stock",
"product_status": "",
"product_email": "",
"product_url": "",
"product_delivery": "",
"product_user": "",
"product_page": "",
"product_program": "",
"product_hosting": "",
"product_options": "",
"product_content": "",
"product_contentgroup": "",
"product_contenttype": "",
"product_imagegroup": "",
"product_imagetype": "",
"product_filegroup": "",
"product_filetype": "",
"product_linkgroup": "",
"product_linktype": "",
"product_productgroup": "",
"product_producttype": "",
"product_usergroup": "",
"product_usertype": "",
"version": "",
"device": "",
"usersegment": "",
"usertest": "",
"title": "Product 1",
"author": "",
"keywords": "",
"description": "Product 1 #54PRD001.",
"summary": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",
"server_filename": "",
"image1": "422",
"image2": "414",
"image3": "415",
"file1": "",
"file2": "",
"file3": "",
"link1": "",
"link2": "",
"link3": ""
}

```

10.3.35.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しい注文アイテムを作成します。

“id” パラメータが指定されている場合、その ID の注文アイテムがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成された注文アイテムのデータを返します。返されるデータには、すべての注文アイテムの属性が含まれます。

POST /webadmin/rest/orderitems/	
id	コピーする注文アイテムの ID（任意）
.....	Web コンテンツ管理システムで使用されるすべての注文アイテムの属性、または任意の数

	と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "order_id": "456", "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "order_id": "456", "title": "XYZ", }</pre>	

10.3.35.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存の注文アイテムを更新します。

PUT リクエストは、更新された注文アイテムのデータを返します。返されるデータには、すべての注文アイテムの属性が含まれます。

PUT /webadmin/rest/orderitems/	
id	更新する注文アイテムの ID（必須）
.....	Web コンテンツ管理システムで 사용되는すべての注文アイテムの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "order_id": "456", "title": "XYZ", }</pre>	

10.3.35.5 削除

DELETE リクエストは、指定された ID の既存の注文アイテムを削除します。

DELETE リクエストは、削除された注文アイテムのデータを返します。返されるデータには、すべての注文アイテムの属性が含まれます。

DELETE /webadmin/rest/orderitems/	
id	注文アイテムの ID（必須）
Request	
<pre>{ "id": "123" }</pre>	
Response	

```
{
  "id": "123",
  "order_id": "456",
  .....
}
```

10.3.36 売上

“/webadmin/rest/sales/REPORT/” REST API 関数は、さまざまな売上レポートを返します。

10.3.36.1 売上レポート

GET /webadmin/rest/sales/REPORT/	
summary	サマリー
websites	対象＞ウェブサイト/ドメイン
products	対象＞商品 ?id=ID（指定された id の商品） ?contentgroup=商品グループ名（指定された商品グループのすべての商品） ?contenttype=商品タイプ名（指定された商品タイプのすべての商品）
productgroups	対象＞商品＞グループ
producttypes	対象＞商品＞タイプ
daily	日時＞日毎
weekly	日時＞週間
monthly	日時＞月間
yearly	日時＞年間
hours	日時＞本日の時間別
weekdays	日時＞今週の日別
days	日時＞今月の日別
weeks	日時＞今年の週別
months	日時＞今年の月別
countries	人＞国
operatingsystems	人＞オペレーティングシステム
webbrowsers	人＞ブラウザ
devices	人＞デバイス
users	人＞ユーザ ?usergroup=ユーザグループ名（指定されたユーザグループのすべての注文） ?usertype=ユーザタイプ名（指定されたユーザタイプのすべての注文）
usergroups	人＞ユーザ＞グループ
usertypes	人＞ユーザ＞タイプ
affiliates	理由＞アフィリエイト
referers	理由＞リファラー
searchengines	理由＞検索エンジン
searchqueries	理由＞検索クエリ
searchwords	理由＞検索ワード
entry	方法＞入口

10.3.36.2 リクエストパラメータ

ほとんどの売上レポートは、同じリクエストパラメータをサポートしています。

GET /webadmin/rest/sales/REPORT/?period=PERIOD&limit=LIMIT¤cy=CURRENCY&details=DETAILS	
GET /webadmin/rest/sales/REPORT/?period_start=PERIODSTART&period_end=PERIODEND&limit=LIMIT¤cy=CURRENCY&details=DETAILS	
period	レポートの対象期間 “now”（デフォルト：30 分） / “today”（本日） / “last24hours”（過去 24 時間） / “yesterday”（昨日） / “thisweek”（今週） / “last7days”（過去 7 日間） / “lastweek”（先週） / “last14days”（過去 14 日間） / “thismonth”（今月） / “last30days”（過去 30 日間） / “lastmonth”（先月） / “thisquarter”（今四半期） / “last3months”（過去 3 ケ月間） / “lastquarter”（先四半期） / “thishalfyear”（今半期） / “last6months”（過去 6 ケ月間） / “lasthalfyear”（先半期） / “this year”（今年） / “last12months”（過去 12 ケ月間） / “last year”（昨年） / “all”（すべて）
period_start	レポートの開始日時 “YYYY-MM-DD hh:mm:ss” 形式
period_end	レポートの終了日時 “YYYY-MM-DD hh:mm:ss” 形式
limit	レポートする売上データの最大数 上限の売上データ数のみをレポートします。
currency	レポートする注文の通貨の ID すべての通貨の場合は、“*”
details	レポートの詳細追加 “+countusersegments” ユーザセグメント毎 “+countusertests” ユーザテスト毎 “+countusersegments+countusertests” ユーザセグメントとユーザテスト毎

10.3.36.3 レポートデータ

ほとんどの売上レポートは、リクエストされたデータについて同じタイプのデータをレポートします。

countcustomers	顧客数
countorders	注文数
countorderitems	注文アイテム数
sumitemquantity	注文された商品の数量
sumitemtotal	注文された商品の売上合計
maxcustomers	顧客の最大数
maxorders	注文の最大数
maxorderitems	注文アイテムの最大数
maxitemquantity	注文された商品の最大数量
maxitemtotal	注文された商品の最大売上合計

totalcustomers	顧客数の合計
totalorders	注文数の合計
totalorderitems	注文アイテム数の合計
totalitemquantity	注文された商品の合計数量
totalitemtotal	注文された商品の売上合計

10.3.37 データベース

“/webadmin/rest/databases/” REST API 関数は、1つ以上のコンテンツデータベースのすべてのデータ（設定）を返し、コンテンツデータベースを作成、更新、削除します。

10.3.37.1 リスト

“id” パラメータが指定されていない場合、コンテンツデータベースのリストが返されます。

GET /webadmin/rest/databases/	
Response	
<pre>[{ "id": "123", "title": "ABC", }, { "id": "789", "title": "XYZ", }]</pre>	

10.3.37.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのコンテンツデータベースのデータ（設定）を返します。返されるデータには、すべてのコンテンツデータベースの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/databases/?id=ID	
id	コンテンツデータベースの ID（必須）
Response	
<pre>{ "id": "1", "title": "Job Directory", "content": "1 1 Name text 80 1\r\n2 2 Summary html 800 600\r\n4 4 Parent datum Job Directory Name\r\n5 5 Created By createdby \r\n6 6 Created Date created \r\n8 8 Sub-Category text 80 1", "searchresults": "", "searchresult": "", "viewpage": "", "adminpage": "", "adminindex": "", "users_group": "", "users_type": "", "users_users": "", "creators_group": "Human Resources",</pre>	

```

"creators_type": "",
"creators_users": "",
"editors_group": "Human Resources",
"editors_type": "",
"editors_users": "",
"publishers_group": "Human Resources",
"publishers_type": "",
"publishers_users": "",
"administrators_group": "Website Administrators",
"administrators_type": "",
"administrators_users": "",

"user": true,
"creator": true,
"editor": true,
"publisher": true,
"administrator": true
}

```

10.3.37.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいコンテンツデータベースを作成します。

“id” パラメータが指定されている場合、その ID のコンテンツデータベースがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたコンテンツデータベースのデータ（設定）を返します。返されるデータには、すべてのコンテンツデータベースの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/databases/	
id	コピーするコンテンツデータベースの ID（任意）
.....	Web コンテンツ管理システムで使用するすべてのコンテンツデータベースの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre> { "id": "123", "title": "XYZ" } </pre>	
Response	
<pre> { "id": "789", "title": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true } </pre>	

10.3.37.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のコンテンツデータベースを更新します。

PUT リクエストは、更新されたコンテンツデータベースのデータ（設定）を返します。返されるデータには、すべてのコンテンツデータベースの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/databases/	
id	更新するコンテンツデータベースの ID（必須）
.....	Web コンテンツ管理システムで使用するすべてのコンテンツデータベースの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "title": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "title": "XYZ", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.37.5 削除

DELETE リクエストは、指定された ID の既存のコンテンツデータベースを削除します。

DELETE リクエストは、削除されたコンテンツデータベースのデータ（設定）を返します。返されるデータには、すべてのコンテンツデータベースの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/databases/	
id	コンテンツデータベースの ID（必須）
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "title": "ABC", "user": true, "creator": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.38 アクセス統計

“/webadmin/rest/usage/REPORT/” REST API 関数は、さまざまなアクセス統計値レポートを返します。

10.3.38.1 アクセス統計値レポート

GET /webadmin/rest/usage/REPORT/	
summary	サマリー
websites	目的>ウェブサイト/ドメイン
contents	目的>ウェブサイトコンテンツ
contentitem	目的>ウェブサイトコンテンツ>コンテンツ項目
pages	目的>ウェブサイトコンテンツ>ページ ?contentgroup=コンテンツグループ名 ?contenttype=コンテンツタイプ名
pagegroups	目的>ウェブサイトコンテンツ>ページ>グループ
pagetypes	目的>ウェブサイトコンテンツ>ページ>タイプ
contacts	目的>ウェブサイトコンテンツ>コンタクト
posts	目的>ウェブサイトコンテンツ>掲示
logins	目的>ウェブサイトコンテンツ>ログイン
logouts	目的>ウェブサイトコンテンツ>ログアウト
register	目的>ウェブサイトコンテンツ>登録
unregister	目的>ウェブサイトコンテンツ>登録解除
stylesheets	目的>ウェブサイトコンテンツ>スタイルシート
scripts	目的>ウェブサイトコンテンツ>スクリプト
library	目的>メディアライブラリ
images	目的>メディアライブラリ>イメージ ?contentgroup=イメージグループ名 ?contenttype=イメージタイプ名
imagegroups	目的>メディアライブラリ>イメージ>グループ
imagetypes	目的>メディアライブラリ>イメージ>タイプ
files	目的>メディアライブラリ>ファイル ?contentgroup=ファイルグループ名 ?contenttype=ファイルタイプ名
filegroups	目的>メディアライブラリ>ファイル>グループ
filetypes	目的>メディアライブラリ>ファイル>タイプ
links	目的>メディアライブラリ>リンク ?contentgroup=リンクグループ名 ?contenttype=リンクタイプ名
linkgroups	目的>メディアライブラリ>リンク>グループ
linktypes	目的>メディアライブラリ>リンク>タイプ
products	目的>E コマース>商品



	?contentgroup=商品グループ名 ?contenttype=商品タイプ名
productgroups	目的>E コマース>商品>グループ
producttypes	What / Ecommerce / Products / Types 目的>E コマース>商品>タイプ
shopcart	目的>E コマース>ショッピングカート追加 ?contentgroup=商品グループ名 ?contenttype=商品タイプ名
shopcartgroups	目的>E コマース>ショッピングカート追加>グループ
shopcarttypes	目的>E コマース>ショッピングカート追加>タイプ
shopcart.checkout	目的>E コマース>ショッピングカートチェックアウト ?contentgroup=商品グループ名 ?contenttype=商品タイプ名
shopcartgroups.checkout	目的>E コマース>ショッピングカートチェックアウト>グループ
shopcarttypes.checkout	目的>E コマース>ショッピングカートチェックアウト>タイプ
shopcart.confirm	目的>E コマース>ショッピングカート確認 ?contentgroup=商品グループ名 ?contenttype=商品タイプ名
shopcartgroups.confirm	目的>E コマース>ショッピングカート確認>グループ
shopcarttypes.confirm	目的>E コマース>ショッピングカート確認>タイプ
shopcart.complete	目的>E コマース>ショッピングカート完了 ?contentgroup=商品グループ名 ?contenttype=商品タイプ名
shopcartgroups.complete	目的>E コマース>ショッピングカート完了>グループ
shopcarttypes.complete	目的>E コマース>ショッピングカート完了>タイプ
shopcart.conversion	目的>E コマース>ショッピングカートコンバージョン ?contentgroup=商品グループ名 ?contenttype=商品タイプ名
databases	目的>コンテンツデータベース
database	目的>コンテンツデータベース>データベース ?database=データベース名
daily	日時>日毎
weekly	日時>週間
monthly	日時>月間
yearly	日時>年間
hours	日時>本日の時間別
weekdays	日時>今週の日別
days	日時>今月の日別

weeks	日時＞今年の週別
months	日時＞今年の月別
countries	人＞国
clienthosts	人＞ホスト
clienthost	人＞ホスト＞ホスト
visitors	人＞ビジター
robots	人＞ロボット
operatingsystems	人＞オペレーティングシステム
webbrowsers	人＞ブラウザ
devices	人＞デバイス
users	人＞ユーザ ?usergroup=ユーザグループ名 ?usertype=ユーザタイプ名
usergroups	人＞ユーザ＞グループ
usertypes	人＞ユーザ＞タイプ
usersegments	人＞ユーザ＞セグメント
usertests	人＞ユーザ＞テスト
referrers	理由＞リファラー
searchengines	理由＞検索エンジン
searchqueries	理由＞検索項目
searchwords	理由＞検索語
entry	方法＞エントリー（ページ）
paths	方法＞パス（ページ）
exit	方法＞終了（ページ）
duration	方法＞期間
visits	方法＞訪問
visit	方法＞訪問＞訪問
username	方法＞訪問＞ユーザ名
visitor	方法＞訪問＞ビジター

10.3.38.2 リクエストパラメータ

ほとんどのアクセス統計値レポートは、同じリクエストパラメータをサポートしています。

GET /webadmin/rest/usage/REPORT/?period=PERIOD&limit=LIMIT&details=DETAILS	
GET /webadmin/rest/usage/REPORT/?period_start=PERIODSTART&period_end=PERIODEND&limit=LIMIT&details=DETAILS	
period	レポートの対象期間 “now”（デフォルト：30 分） / “today”（本日） / “last24hours”（過去 24 時間） / “yesterday”（昨日） / “thisweek”（今週） / “last7days”（過去 7 日間） / “lastweek”（先週） / “last14days”（過去 14 日間） / “thismonth”（今月） / “last30days”（過去 30 日間） / “lastmonth”（先月） / “thisquarter”（今四半期） / “last3months”（過去 3 ケ月間） / “lastquarter”（先四 半期） / “thishalfyear”（今半期） / “last6months”（過 去 6 ケ月間） / “lasthalfyear”（先半期） / “this year” （今年） / “last12months”（過去 12 ケ月間） / “last year”（昨年） / “all”（すべて）

period_start	レポートの開始日時 “YYYY-MM-DD hh:mm:ss” 形式
period_end	レポートの終了日時 “YYYY-MM-DD hh:mm:ss” 形式
limit	レポートするアクセス統計値の最大数 上限のアクセス統計値データ数のみをレポートします。
details	レポートの詳細追加 "+countusersegments" ユーザセグメント毎 "+countusertests" ユーザテスト毎 "+countusersegments+countusertests" ユーザセグメントとユーザテスト毎

10.3.37.3 レポートデータ

ほとんどのアクセス統計値レポートは、リクエストされたデータについて同じタイプのデータをレポートします。

countclienthosts	ユニークな Web サイト訪問者のインターネットアドレスドメイン名、または IP (番号) の数
countvisitors	“cookies” を使用する Web コンテンツ管理システムによって識別される、ユニークな Web サイト訪問者の数
countvisits	“session ids” を使用する Web コンテンツ管理システムによって識別される個別の訪問の数
countpages	リクエストされた Web ページの数
counthits	リクエストされた Web ページ、イメージ、ファイル、スタイルシート、スクリプトの数
maxclienthosts	一意の Web サイト訪問者のインターネットアドレスドメイン名、または IP (番号) の最大数
maxvisitors	“cookies” を使用する Web コンテンツ管理システムによって識別される、ユニークな Web サイト訪問者の最大数
maxvisits	“session ids” を使用する Web コンテンツ管理システムによって識別される個別の訪問の最大数
maxpages	リクエストされた Web ページの最大数
maxhits	リクエストされた Web ページ、イメージ、ファイル、スタイルシート、スクリプトの最大数
totalclienthosts	一意の Web サイト訪問者のインターネットアドレスドメイン名、または IP (番号) の総数
totalvisitors	“cookies” を使用する Web コンテンツ管理システムによって識別される、ユニークな Web サイト訪問者の総数

totalvisits	“session ids” を使用する Web コンテンツ管理システムによって識別される個別の訪問の総数
totalpages	リクエストされた Web ページの総数
totalhits	リクエストされた Web ページ、イメージ、ファイル、スタイルシート、スクリプトの総数

10.3.39 ホスティングクライアント

“/webadmin/rest/hosting/” REST API 関数は、1 つ以上のホスティングクライアントのすべてのデータを返し、ホスティングクライアントを作成、更新、削除します。

10.3.39.1 リスト

“id” パラメータが指定されていない場合、ホスティングクライアントのリストが返されます。

GET /webadmin/rest/hosting/	
Response	
<pre>[{ "id": "123", "user_id": "0", "clientaddress": "www.abc.com", }, { "id": "789", "user_id": "0", "clientaddress": "www.xyz.com", }]</pre>	

10.3.39.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのホスティングクライアントのデータを返します。返されるデータには、すべてのホスティングクライアントの属性が含まれます。

GET /webadmin/rest/hosting/?id=ID	
id	ホスティングクライアントの ID (必須)
Response	
<pre>{ "id": "1", "user_id": "0", "client_address": "www.abc.com", "client_URLrootpath": "\/SITEPERS\/", "client_database": "mysql:mysql:\\\/\/username:password@localhost\/foocom", "personal_license": "HOSTINGTRIAL:personal:8T4EXQERBGX9A7C4", "professional_license": "", "enterprise_license": "", "hosting_license": "", "ecommerce_license": "", "community_license": "", "databases_license": "", "statistics_license": "", }</pre>	

```

    "experience_license": "",
    "collaboration_license": "",
    "superadmin": "admin",
    "superadmin_password": "",
    "superadmin_email": "",
    "hostinggroup": "",
    "hostingtype": "",
    "scheduled_publish": "",
    "scheduled_notify": "",
    "scheduled_unpublish": "",
    "scheduled_publish_email": "",
    "scheduled_notify_email": "",
    "scheduled_unpublish_email": "",
    "scheduled_last": ""
  }

```

10.3.39.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいホスティングクライアントを作成します。

“id” パラメータが指定されている場合、その ID のホスティングクライアントがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたホスティングクライアントのデータを返します。返されるデータには、すべてのホスティングクライアントの属性が含まれます。

POST /webadmin/rest/hosting/	
id	コピーするホスティングクライアントの ID (任意)
.....	Web コンテンツ管理システムで使用するすべてのホスティングクライアントの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre> { "id": "123", "clientaddress": "www.xyz.com" } </pre>	
Response	
<pre> { "id": "789", "clientaddress": "www.xyz.com", } </pre>	

10.3.39.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のホスティングクライアントを更新します。

PUT リクエストは、更新されたホスティングクライアントのデータを返します。返されるデータには、すべてのホスティングクライアントの属性が含まれます。

PUT /webadmin/rest/hosting/	
id	更新するホスティングクライアントの ID (必須)

.....	Web コンテンツ管理システムで使用するすべてのホスティングクライアントの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "personal_license": "www.xyz.com:personal:3TH6XBEUJGX4A9C2" }</pre>	
Response	
<pre>{ "id": "123", "clientaddress": "www.xyz.com", "personal_license": "www.xyz.com:personal:3TH6XBEUJGX4A9C2" }</pre>	

10.3.39.5 削除

DELETE リクエストは、指定された ID の既存のホスティングクライアントを削除します。

DELETE リクエストは、削除されたホスティングクライアントのデータを返します。返されるデータには、すべてのホスティングクライアントの属性が含まれます。

DELETE /webadmin/rest/hosting/	
id	ホスティングクライアントの ID（必須）
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "clientaddress": "www.abc.com", }</pre>	

10.3.40 ホスティンググループ

“/webadmin/rest/hostinggroups/” REST API 関数は、1 つ以上のホスティンググループのすべてのデータを返し、ホスティンググループを作成、更新、削除します。

10.3.40.1 リスト

“id” パラメータが指定されていない場合、ホスティンググループのリストが返されます。

GET /webadmin/rest/hostinggroups/	
Response	
<pre>[{ "id": "123", "hostinggroup": "ABC" }, { "id": "789",</pre>	

```
.....
    "hostinggroup": "XYZ"
  }
]
```

10.3.40.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのホスティンググループのデータを返します。返されるデータには、すべてのホスティンググループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/hostinggroups/?id=ID	
id	ホスティンググループの ID (必須)
Response	
<pre>{ "id": "1", "hostinggroup": "Enterprise", "users_group": "", "users_type": "", "users_users": "", "creators_group": "", "creators_type": "", "creators_users": "", "editors_group": "", "editors_type": "", "editors_users": "", "publishers_group": "", "publishers_type": "", "publishers_users": "", "administrators_group": "", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.40.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいホスティンググループを作成します。

“id” パラメータが指定されている場合、その ID のホスティンググループがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたホスティンググループのデータを返します。返されるデータには、すべてのホスティンググループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/hostinggroups/	
id	コピーするホスティンググループの ID (任意)

.....	Web コンテンツ管理システムで使用されるすべてのホスティンググループの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "hostinggroup": "XYZ" }</pre>	
Response	
<pre>{ "id": "789", "hostinggroup": "XYZ", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.40.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のホスティンググループを更新します。

PUT リクエストは、更新されたホスティンググループのデータを返します。返されるデータには、すべてのホスティンググループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/hostinggroups/	
id	更新するホスティンググループの ID（必須）
.....	Web コンテンツ管理システムで使用されるすべてのホスティンググループの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "administrators_group": "XYZ" }</pre>	
Response	
<pre>{ "id": "123", "hostinggroup": "ABC", "administrators_group": "Hosting Administrators", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.40.5 削除

DELETE リクエストは、指定された ID の既存のホスティンググループを削除します。

DELETE リクエストは、削除されたホスティンググループのデータを返します。返されるデータには、すべてのホスティンググループの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/hostinggroups/	
id	ホスティンググループの ID (必須)
Request	
<pre>{ "id": "123" }</pre>	
Response	
<pre>{ "id": "123", "hostinggroup": "ABC", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.41 ホスティングタイプ

“/webadmin/rest/hostingtypes/” REST API 関数は、1 つ以上のホスティングタイプのすべてのデータを返し、ホスティングタイプを作成、更新、削除します。

10.3.41.1 リスト

“id” パラメータが指定されていない場合、ホスティングタイプのリストが返されます。

GET /webadmin/rest/hostingtypes/	
Response	
<pre>[{ "id": "123", "hostingtype": "ABC" }, { "id": "789", "hostingtype": "XYZ" }]</pre>	

10.3.41.2 読み取り

“id” パラメータが指定されている場合、GET リクエストはそのホスティングタイプのデータを返します。返されるデータには、すべてのホスティングタイプの属性と、現在ログインしてい

る Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

GET /webadmin/rest/hostingtypes/?id=ID	
id	ホスティングタイプの ID (必須)
Response	
<pre>{ "id": "1", "hostingtype": "Trial Users", "users_group": "", "users_type": "", "users_users": "", "creators_group": "", "creators_type": "", "creators_users": "", "editors_group": "", "editors_type": "", "editors_users": "", "publishers_group": "", "publishers_type": "", "publishers_users": "", "administrators_group": "", "administrators_type": "", "administrators_users": "", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>	

10.3.41.3 作成

POST リクエストは、指定されたデータ属性と自動的に割り当てられた唯一の ID を持つ新しいホスティングタイプを作成します。

“id” パラメータが指定されている場合、その ID のホスティングタイプがコピーされ、その属性が（部分的に）POST されたデータによって上書きされます。

POST リクエストは、作成されたホスティングタイプのデータを返します。返されるデータには、すべてのホスティングタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

POST /webadmin/rest/hostingtypes/	
id	コピーするホスティングタイプの ID (任意)
.....	Web コンテンツ管理システムで使用されるすべてのホスティングタイプの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre>{ "id": "123", "hostingtype": "XYZ" }</pre>	
Response	
<pre>{</pre>	

```

    "id": "789",

    "hostingtype": "XYZ",
    .....
    "user": true,
    "creator": true,
    "developer": true,
    "editor": true,
    "publisher": true,
    "administrator": true
  }

```

10.3.41.4 更新

PUT リクエストは、指定された ID、および指定されたデータ属性で既存のホスティングタイプを更新します。

PUT リクエストは、更新されたホスティングタイプのデータを返します。返されるデータには、すべてのホスティングタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

PUT /webadmin/rest/hostingtypes/	
id	更新するホスティングタイプの ID (必須)
.....	Web コンテンツ管理システムで使用するすべてのホスティングタイプの属性、または任意の数と組み合わせ (上記の GET リクエストのレスポンス例を参照)
Request	
<pre> { "id": "123", "administrators_group": "XYZ" } </pre>	
Response	
<pre> { "id": "123", "hostingtype": "ABC", "administrators_group": "Hosting Administrators", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true } </pre>	

10.3.41.5 削除

DELETE リクエストは、指定された ID の既存のホスティングタイプを削除します。

DELETE リクエストは、削除されたホスティングタイプのデータを返します。返されるデータには、すべてのホスティングタイプの属性と、現在ログインしている Web サイト管理者の「閲覧者」、「作成者」、「更新者」、「公開者」、「開発者」、および「管理者」のアクセス権限の true/false フラグが含まれます。

DELETE /webadmin/rest/hostingtypes/	
id	ホスティングタイプの ID (必須)

Request
<pre>{ "id": "123" }</pre>
Response
<pre>{ "id": "123", "hostingtype": "ABC", "user": true, "creator": true, "developer": true, "editor": true, "publisher": true, "administrator": true }</pre>

10.3.42 設定

“/webadmin/rest/config/” REST API 関数は、1 つ以上の設定項目のデータを返します。

10.3.42.1 読み取り

“id” パラメータが指定されている場合、GET リクエストはその設定項目のデータを返します。

GET /webadmin/rest/config/?id=ID,ID,ID	
id	設定項目名（必須） 複数の場合はカンマ区切り 例： ?id=default_template,default_stylesheet,default_version
Response	
<pre>{ "default_template": "35", "default_stylesheet": "1", "default_version": "" }</pre>	

10.3.43 Web サイト管理者設定

“/webadmin/rest/user/” REST API 関数は、ログイン中の Web サイト管理者のすべてのユーザアカウントの設定を返し、更新します。

10.3.43.1 読み取り

ログイン中の Web サイト管理者のユーザアカウントの設定を返します。

GET /webadmin/rest/user/	
Response	
<pre>{ "id": "6", "username": "paulgreen", "password": "XYABQ0EB1LG7MCCUXTUJXRXBK1FJ0LXL4H86HEGVY8VK5QHRHGA71KB9AXGLVBYN", "onetimepassword": "", "onetimepassword_updated": "2018-04-04 17:42:29", "title": "", "name": "Paul Green", "organisation": "", "email": "" }</pre>	



```
"gender": "",
"photo": "",
"birthdate": "",
"birthyear": "",
"birthmonth": "",
"birthday": "",

"invoice_name": "",
"invoice_organisation": "",
"invoice_address": "",
"invoice_postalcode": "",
"invoice_city": "",
"invoice_state": "",
"invoice_country": "",
"invoice_phone": "",
"invoice_fax": "",
"invoice_email": "",
"invoice_website": "",

"delivery_name": "",
"delivery_organisation": "",
"delivery_address": "",
"delivery_postalcode": "",
"delivery_city": "",
"delivery_state": "",
"delivery_country": "",
"delivery_phone": "",
"delivery_fax": "",
"delivery_email": "",
"delivery_website": "",

"card_type": "",
"card_number": "",
"card_issuedmonth": "",
"card_issuedyear": "",
"card_expirymonth": "",
"card_expiryyear": "",
"card_name": "",
"card_cvc": "",
"card_issue": "",
"card_postalcode": "",

"content_editor": "",
"hardcore_upload": "",
"hardcore_format": "",
"hardcore_width": "",
"hardcore_height": "",
"hardcore_onenter": "",
"hardcore_onctrlenter": "",
"hardcore_onshiftenter": "",
"hardcore_onaltenter": "",
"hardcore_skin": "",
"hardcore_toolbar": "",
"hardcore_toolbar1": "",
"hardcore_toolbar2": "",
"hardcore_toolbar3": "",
"hardcore_toolbar4": "",
"hardcore_toolbar5": "",
"hardcore_toolbar6": "",
"hardcore_toolbar7": "",
"hardcore_toolbar8": "",
"hardcore_toolbar9": "",
"hardcore_toolbar10": "",
"hardcore_formatblock": "",
"hardcore_fontname": "",
"hardcore_fontsize": "",
"hardcore_customscript": "",
"hardcore_encoding": "",
"hardcore_language": "",

"startpage": "",
```

```

"webadmin_user": "forbid",
"webadmin_ecommerce": "forbid",
"webadmin_library_packages": "forbid",
"webadmin_user": "forbid",
"webadmin_ecommerce": "forbid",
"webadmin_library_packages": "forbid",
"webadmin_user": "forbid",
"webadmin_ecommerce": "forbid",
"webadmin_library_packages": "forbid",

"workspace_sections": "",
"index_workspace": "",
"index_workspace_projects": "",
"index_workspace_comments": "",
"index_content": "",
"index_library": "",
"index_search": "",
"index_searchadvanced": "",
"index_searchreplace": "",
"index_product": "",
"index_stock": "",
"index_order": "",
"index_segments": "",
"index_usertests": "",
"index_heatmaps": "",
"index_user": "",
"index_websites": "",
"index_sales": "",
"index_databases": "",
"index_hosting": "",
"index_projects": "",

"menu_selection": ""
}

```

10.3.43.2 更新

PUT リクエストは、ログイン中の Web サイト管理者のユーザアカウントの設定を、指定されたデータ属性で更新します。

PUT リクエストは、更新された Web サイト管理者のユーザアカウントの設定を返します。

PUT /webadmin/rest/user/	
.....	Web コンテンツ管理システムで使用するすべての Web サイト管理者のユーザアカウントの属性、または任意の数と組み合わせ（上記の GET リクエストのレスポンス例を参照）
Request	
<pre> { "password": "secret", "birthdate": "1999-12-31", "birthyear": "1999", "birthmonth": "12", "birthday": "31" } </pre>	
Response	
<pre> { "id": "6", "username": "paulgreen", "password": "XYABQ0EB1LG7MCCUXTUJXRXBK1FJ0LXL4H86HEGVY8VK5QHRHGA71KB9AXGLVBYN", "birthdate": "1999-12-31", "birthyear": "1999", "birthmonth": "12", </pre>	

```

    "birthday": "31",
    .....
}

```

10.3.44 全体管理者パスワード

“/webadmin/rest/password/” REST API 関数は、設定された全体管理者（スーパーユーザ）のメールアドレスにログインの詳細情報をメールで送信します。

10.3.44.1 読み取り

空のレスポンスが返されます。

GET /webadmin/rest/password/	
Response	
{ }	

10.3.45 ログアウト

“/webadmin/rest/logout/” REST API 関数は、ログイン中の Web サイト管理者のアカウントをログアウトします。

POST リクエストは空のレスポンスを返します。

POST /webadmin/rest/logout/	
Request	
{ }	
Response	
{ }	