

# SmartBrowser Update

for iOS / Version 1.3.1 build30

2017年8月

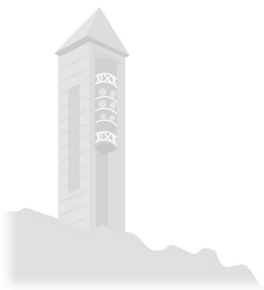


株式会社 ブルーテック  
Bluetech Corporation

## 更新内容-概要

### iOS Version 1.3.1 build28 の更新内容について

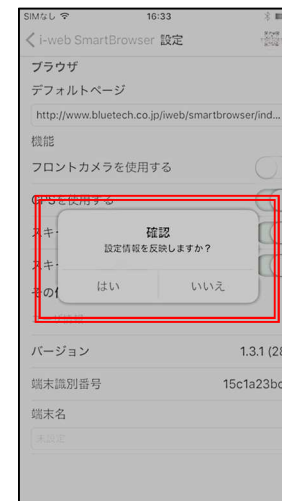
1. 設定をQRから読み込み更新する機能
2. URLをQRから読み込み画面遷移する機能
3. WEBページのローカルファイル保存と外部インテントからの起動
4. JQuery-LoadImageライブラリの組み込み
5. バイブレート及び音声ファイルの再生する機能
6. SmartBrowserAPI呼び出し方法の変更
7. GPS座標の取得と利用方法



# 更新内容 (1)

## 1. 設定をQRから読み込み更新する機能

メニュー → 設定 → 設定画面の右上QRボタンでカメラ起動 → 設定QRを読み込 → 確認



# 更新内容（1）

## 1. 設定をQRから読み込み更新する機能

設定書式 - 設定はJsonで記述します

```
{
  "default_page_url": string型, //デフォルトページ
  "use_front_camera": bool型, //フロントカメラを使用する
  "gps_enable": bool型, //GPSを使用する
  "vibrate_when_scanned": bool型, //スキャン時にバイブレーション
  "ring_when_scanned": bool型, //スキャン時に効果音を鳴らす
  "device_name": string型, //端末名
  "bookmark": [ //ブックマーク(json array )
    {
      "title": string型, //タイトル
      "url": string型, //URL
    },
    {
      "title": string型,
      "url": string型,
    }, //...
  ]
}
```

### ※仕様についての注意事項

- ・設定した項目のみ上書きします
- ・変更しない項目については、項目名を含めないようにしてください
- ・ブックマークについては、登録済みの内容をいったん削除した後に設定値を登録します
- ・bool型は“true” / “false” で記述してください(ダブルクォートで囲います)

# 更新内容（1）

## 1. 設定をQRから読み込み更新する機能

サンプル

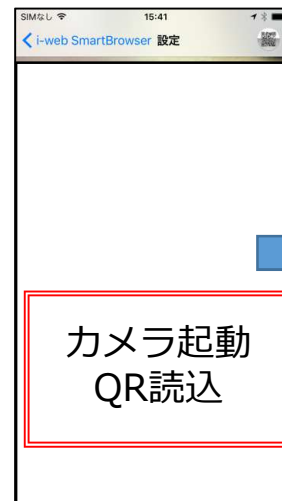
```
{  
  "default_page_url": "http://www.bluetech.co.jp/iweb/smartbrowser/",  
  "use_front_camera": "false",  
  "gps_enable": "true",  
  "vibrate_when_scanned": "true",  
  "ring_when_scanned": "true",  
  "device_name": "テスト端末1",  
  "bookmark": [  
    {  
      "title": "画像縮小",  
      "url": "http://resizer.myct.jp/",  
    },  
    {  
      "title": "LoadImageTest",  
      "url": "http://www.bluetech.co.jp/iweb/smartbrowser/test/loadImage.html",  
    },  
    {  
      "title": "LocalDownloadTest",  
      "url": "http://www.bluetech.co.jp/iweb/smartbrowser/test/localdownload.html",  
    },  
  ],  
}
```



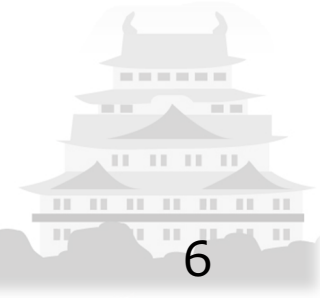
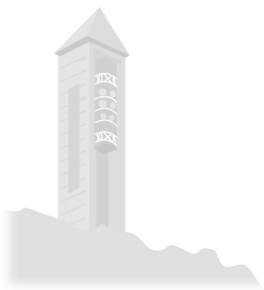
## 更新内容（2）

### 2. URLをQRから読み込み画面遷移する機能

メニュー → URL → QRコード → 設定QRを読み込 → URL先へ遷移



URL先へ  
画面遷移



## 更新内容（3）

### 3. WEBページのローカルファイル保存と外部インテントからの起動

#### 手順説明

##### <ローカルファイル準備>

1. 端末へローカル保存したいWEBページを作成し（複数ページ、階層構造可能）ZIP保存する
2. 保存したZIPをSmartBrowserでアクセス可能な場所に配置する
3. JavaScriptで上記ZIPダウンロード処理を記述する
4. 上記JavaScriptで端末内のローカルフォルダに保存される（3の処理の度に上書き）

##### <外部システムから起動する（メール、ブラウザ等）>

5. 外部から起動するリンクを記述し、SmartBrowserを起動する

※サンプルコードは以下のURLを参照ください

<http://www.bluetech.co.jp/iweb/smartbrowser/test/localdownload.html>

## 更新内容（3）

### 3. WEBページのローカルファイル保存と外部インテントからの起動

<ローカルファイル準備>

JavaScriptで上記ZIPダウンロード処理を記述する

```

<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <script src="http://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <script>

    function extractToLocal() {

      smartbrowser().callAPI({path:'extract_to_local', params:{
        name: 'test1', //ローカル保存先フォルダ名
        url: 'http://www.bluetech.co.jp/iweb/smartbrowser/test/local.zip', //ZIPファイルURL
      }});
      //ZIPファイルをローカルにダウンロードし、ローカル保存名で展開します。
    }

  </script>
</head>
<body>
  <input type="button" onclick="extractToLocal()" value="download">
</body>
</html>

```

#### ※解説

- 端末ローカルファイル内に「test1」フォルダを作成し  
その中にZIP内のWEBページを保存します
- 展開後にページへアクセスする場合は、  
指定したフォルダ名+ページ名で  
アクセスします





## 更新内容（3）

### 3. WEBページのローカルファイル保存と外部インテントからの起動

<アプリ起動用URLの作成>

smartbrowser://open?url=ページURL

ページURLは

local://test1/index.html?param1=abcde

ローカル保存先（上記のtest1）フォルダとZIP展開後の  
ファイル名を指定してパスを作成します

**ページURL部分はURLエンコードを行う必要がありますので**

**下記ようになります**

smartbrowser://open?url=local%3A%2F%2Ftest1%2Findex.html%3Fparam1%3Dabcde

#### ※解説

- ・ 端末ローカルファイル内に「test1」フォルダを作成し  
その中にZIP内のWEBページを保存します
- ・ 展開後にページへアクセスする場合は、  
指定したフォルダ名+ページ名で  
アクセスします

## 更新内容（3）

### 3. WEBページのローカルファイル保存と外部インテントからの起動

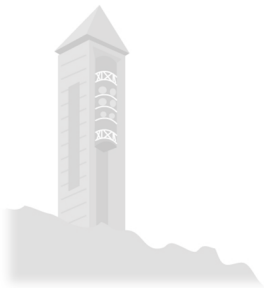
<SmartBrowser内で使用するURLの記述>

local://test1/index.html?param1=abcde

※画像等のパス指定についても同様です

#### ※解説

- ・ 端末ローカルファイル内に「test1」フォルダを作成し  
その中にZIP内のWEBページを保存します
- ・ 展開後にページへアクセスする場合は、  
指定したフォルダ名+ページ名で  
アクセスします



## 更新内容（3）

### 3. WEBページのローカルファイル保存と外部インテントからの起動

<ローカル保存ファイルの一覧表示>

JavaScriptで指定フォルダのファイル一覧を表示する

```
function getLocalFilesList() {  
  //指定した保存名のローカルファイルのリストを取得します。  
  smartbrowser().callAPI({path:'local/list', params:{  
    name: 'test1', //ローカル保存先フォルダ名  
  }});  
}
```

<ローカル保存ファイル一覧表示後のコールバック>

onIWebSmartBrowserLocalFilesList(data)

```
function onIWebSmartBrowserLocalFilesList(data) {  
  var $localFileList = $('#localfilelist').empty();  
  data['result']['files'].forEach(function(file) {  
    $localFileList.append($('- </li>').text(file));  
  });  
}

```

#### ※解説

- ・ 端末ローカルファイル内に「 test1 」フォルダを作成し  
その中にZIP内のWEBページを保存します
- ・ 展開後にページへアクセスする場合は、  
指定したフォルダ名+ページ名で  
アクセスします

## 更新内容（3）

### 3. WEBページのローカルファイル保存と外部インテントからの起動

#### <ローカル保存ファイルの削除>

JavaScriptで指定フォルダのファイル一覧を表示する

```
function removeLocalFiles() {  
  //指定した保存名のローカルファイルを削除します。  
  smartbrowser().callAPI({path:'local/remove', params:{  
    name: 'test1', //ローカル保存名  
  }});  
}
```

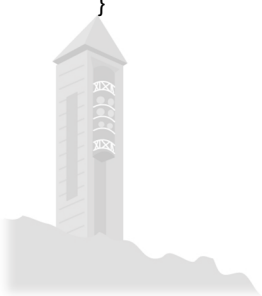
#### <ローカル保存ファイル削除後のコールバック>

onIWebSmartBrowserLocalFilesRemoved(data)

```
function onIWebSmartBrowserLocalFilesRemoved(data) {  
  alert(data['result']['success']?'成功':'失敗');  
}
```

#### ※解説

- ・ 端末ローカルファイル内に「 test1 」フォルダを作成し  
その中にZIP内のWEBページを保存します
- ・ 展開後にページへアクセスする場合は、  
指定したフォルダ名+ページ名で  
アクセスします



## 更新内容（3）

### 3. WEBページのローカルファイル保存と外部インテントからの起動

<ローカルファイル展開後のコールバック>

onIWebSmartBrowserExtractedToLocal(data)

Return value	Json label	Json value
data	error	処理時のエラーがあれば内容出力、なければfalse
	+ result	
	- extracted_path	ローカルファイル展開先パス
	+ request	
	- name	ローカルフォルダ名
	- url	ローカルファイルダウンロード元パス

```

<script>
function onIWebSmartBrowserExtractedToLocal(data) {
//展開コールバック
//data["request"]からAPIコール時のparamsが取得できます。
$('#result').text(JSON.stringify(data));

//ローカルファイル起動用URLの作成と画面遷移
// local://ローカル保存名/パス
var localLinkHref = 'local://' + data['request']['name']+'/index.html?aaa=hhhh';
location.href= localLinkHref ;
}
</script>

```



## 更新内容（４）

### ４． JQuery-LoadImageライブラリの組み込み

SmartBrowserからJquery-LoadImageをコールし、  
SmartBrowserに組み込まれているJquery-LoadImageを使用します

<使い方 例（Image loading）>

```
document.getElementById('file-input').onchange = function (e) {  
    smartbrowser().loadImage(  
        e.target.files[0],  
        function (img) {  
            document.body.appendChild(img);  
        },  
        {maxWidth: 600} // Options  
    );  
};
```

※LoadImageについて、詳しくは以下のURLを参照ください

<https://github.com/blueimp/JavaScript-Load-Image>

※サンプルコードは以下のURLを参照ください

<http://www.bluetech.co.jp/iweb/smartbrowser/test/loadImage.html>

## 更新内容（5）

### 5. バイブレート及び音声ファイルの再生

SmartBrowserからバイブレート及び音声ファイルの再生を行います

<使い方 バイブレート>

```
function vib() {  
    smartbrowser().callAPI({path:"vibrate", params: {  
        //ループ回数を指定  
        "count": 3  
    }});  
}
```

<使い方 音声ファイル>

```
function playSound() {  
    smartbrowser().callAPI({path:"sound/play", params: {  
        //URLまたはURI (Base64String) を指定  
        "uri": "http://www.kurage-kosho.info/mp3/button62.mp3" //URL  
        // "uri": "data:audio/mp3;base64,///uQZAAAAAAAAAAAAA....." //URI (Base64String)  
    }});  
}
```

※サンプルコードは以下のURLを参照ください

<http://www.bluetech.co.jp/iweb/smartbrowser/test/sound.html>

## 更新内容（6）

### 6. SmartBrowserAPI呼び出し方法の変更

Version 1.3.1 build30 からの変更 SmartBrowserAPI呼び出し方法の変更

旧（V1.3.0まで）                      新（V1.3.1以降）

**iwebsmartbrowser.xxxx** → **smartbrowser().xxxx**

※ iwebsmartbrowser.xxxxの記述方法は非推奨となりますが、当面利用は可能です

#### <利用方法例>

```
document.getElementById('itemid106').innerHTML = smartbrowser().location.latitude;  
document.getElementById('itemid107').innerHTML = smartbrowser().location.longitude;  
document.getElementById('itemid108').innerHTML = smartbrowser().location.accuracy;
```

```
function vib() {  
  smartbrowser().callAPI({path:"vibrate", params: {  
    //ループ回数を指定  
    "count": 3  
  }});  
}
```



## 更新内容（7）

### 7. GPS座標の取得と利用方法

Version 1.3.1 build30 からの変更 GPS API呼び出し方の変更

<使い方 GPS情報の取得API呼び出し>

```
function locationRequest() {  
    smartbrowser().callAPI({ path:'location/request', params:{} });  
}
```

<使い方 APIで取得した値の利用方法>

```
document.getElementById('itemid106').innerHTML = smartbrowser().location.latitude;  
document.getElementById('itemid107').innerHTML = smartbrowser().location.longitude;  
document.getElementById('itemid108').innerHTML = smartbrowser().location.accuracy;  
document.getElementById('itemid109').innerHTML = new  
Date(smartbrowser().location.timestamp).toLocaleString();
```

