

株式会社サイブリッジ
ウェブ制作ガイドライン



CYBRIDGE



目次

目次

制作ガイドラインの利用方法

クライアント環境

ディレクトリ・ファイル設計

ディレクトリ・ファイルの種類1

ディレクトリ・ファイルの種類2

ディレクトリ・ファイル命名規則1

ディレクトリ・ファイル命名規則2

ディレクトリ・ファイル命名規則3

ディレクトリ・ファイル命名規則4

ディレクトリ・ファイル命名規則5

ディレクトリ・ファイル命名規則6

文字・記号について1

文字・記号について2

文字・記号について3

ページプロパティ

画像について

JavaScriptの取り扱い1

JavaScriptの取り扱い2

表示速度の改善1

表示速度の改善2

表示速度の改善3



制作ガイドラインの利用方法

見方

必須 . . . 必ず守っていただきたいルールです

推薦 . . . 適用していただくことをオススメするルールです。

任意 . . . 任意で適用していただくルールです

ご質問・お問い合わせ

ご不明な点・疑問点、お困りになった事などがございましたら、ご連絡ください。

株式会社サイブリッジ
担当者 平山 恵子 TEL 03-6381-5900[代表] 03-6381-5904[直通]



エンドユーザーのインターネット利用環境を以下のように想定しています（※2013年1月現在。フィーチャーフォンを除く）。あくまで基準であり、詳細は案件によって都度変更されます。

OS環境

Microsoft Windows XP/Vista/7/8
Macintosh OS X

iOS 5.0以降 ※iPhone、iPad、iPod touch（iOS 4.xも場合によって考慮）
Android 2.3以降（1.6～2.2も場合によって考慮）

対応ブラウザ

Windows :
Internet Explorer 7以降、Firefox 最新版、Chrome 最新版、Opera 最新版
Opera 11.x以降
Macintosh :
Safari 5.x以降、Firefox 最新版、Chrome 最新版
iOS :
Safari
Android :
標準ブラウザ

※基準として想定しないが、表示に関して配慮する。
IE6、Safari 4.x以降、Firefox 3.x以降、
Opera Mini、Opera Mobile

OS・ブラウザはどのバージョンから対応するか、
制作の前に必ず明確にしておく。（特にIE）

スクリーン

デスクトップ・タブレット：画面解像度1024×768px以上
・ A 4サイズのノートブックに対応する
・ B 5サイズのノートブックに対応する

スマートフォン：画面解像度320×480px以上
・ 規格はiPhone4Sの画面解像度640×960pxを基準とし、
またはiPhone5の640px×1136pxを基準とし、
原則Retinaディスプレイに対応する。

色深度

16bit以上
・ 液晶モニタの視認性に対応&モノクロプリンタでの再現性に対応



ディレクトリ・ファイル設計ルール

- └ css/ . . . css格納フォルダ (パフォーマンス上、CSSのインポートは基本的に行わない)
 - └ common.css . . . CSSリセット&デフォルトCSS (OOCSSやレスポンシブ用のデフォルトスタイルを管理)
 - └ style.css . . . 全ページ共通のスタイルシートファイル
 - └ xxx.css . . . 各ページの固有スタイルシートファイル (例: home.css)

- └ js/ . . . JavaScriptファイル格納フォルダ
 - └ script.js . . . 共通JS(jQuery)ファイル

- └ img/ . . . 画像格納フォルダ (直下にファビコンファイルを格納)
 - └ common . . . 共通画像格納フォルダ
 - └ home . . . トップページ (index.html) の画像格納フォルダ
 - └ xxx . . . 各ページごとの画像格納フォルダ
 - └ bnr . . . バナーの画像格納フォルダ

- └ htc/ . . . HTCファイル格納フォルダ
- └ font/ . . . Webフォントファイル格納フォルダ
- └ media/ . . . リッチコンテンツ格納フォルダ
- └ docs/ . . . ドキュメント格納フォルダ
- └ index.html . . . トップページファイル

- └ company . . . 会社概要htmlファイル格納フォルダ
 - └ index.html . . . 会社概要下層ページファイル

- └ xxx . . . 各ページhtml格納フォルダ
 - └ index.html . . . 下層ページファイル
 - └ xxx.html . . . 下層ページファイル

- └ content/ . . . Ajaxで読み込む外部HTMLファイルなどを格納



HTML

文書型：XHTML 1.0 Strict または XHTML 1.0 Transitional または HTML5

※文書型の選択は案件による。タブレット・スマートフォン最適化に関しては今後HTML5の使用を必須とする。

ファイルの拡張子は「.html」のみ。「.htm」は許容しない。

(SSI設定は、拡張し「.html」を有効とする)

画像ディレクトリ

[/img]

HTMLファイルから参照される画像ファイル（GIF画像やJPEG画像）を格納する。

格納されるファイルの拡張子は「.gif」「.jpg」「.png」「.ico」。

CSSディレクトリ

[/css]

HTMLファイルから参照されるCSSファイルを格納する。

格納されるファイルの拡張子は「.css」。

JavaScriptディレクトリ

[/js]

HTMLファイルから参照されるJavaScriptファイルを格納する。

格納されるファイルの拡張子は「.js」。



HTC (HTML Component)

[/htc]

IEのビヘイビア機能を利用するためのHTCファイルを格納する。
格納されるファイルの拡張子は「.htc」。(汎用化のためのPHPファイルも許容する)

※ビヘイビアとは、HTML/Dynamic HTML/JavaScript等によって構成されるHTC (HTML Component) ファイルをHTML上で使用する機能。HTCファイルを使用することで関数の再利用/カプセル化を図ることができる。

Webフォント

[/font]

Webフォントなどで使用するフォントファイルを格納する。
格納されるファイルの拡張子は「.eot」「.svg」「.ttf」「.woff」等

リッチコンテンツ

[/media]

動画ファイルや音声ファイルなどを格納する。
格納されるファイルの拡張子は「.swf」「.mp3」等各種メディアファイル。

ドキュメント

[/doc]

各種ドキュメントファイル。
格納されるファイルの拡張子は「.doc」「.xls」「.pdf」等



ファイル名・ディレクトリ名はできる限り短くし、ひと目で意味のわかるように、ファイルの内容が想像できるように命名する。

使用可能文字・記号

ディレクトリ名・ファイル名には半角(1バイト) 英数小文字・数字・若干の記号のみ使用可とする。

```
abcdefghijklmnopqrstuvwxyz  
0123456789  
_.
```

文字数

さまざまな環境下でファイルを扱えるようにするため、半角(1バイト)で最大31文字とする。

禁止事項

- ・空白（スペース）はファイル名では利用できない。
- ・「_」（アンダーバー）や「-」（ハイフン）をそれぞれ2回続けて使用しない。
（ファイル名・ディレクトリ名の中の属性または単語を2つ続ける場合は許容する。）
例) ○ sn_contact_price.gif
× sn__contact--price.gif
- ・基本は英数字とアンダーバーで構成。ハイフンは原則ファイル名に使わない。
例) ○ photo_01.gif
× photo-01.gif
- ・以下のファイル名の使用を禁止する。

```
con. aux. nul. prn.
```

※Windows環境下でファイルを操作できなくなるため



表記

ディレクトリ・ファイル名の表記を以下のように決定する。

- ・日本語…ヘボン式ローマ字表記
- ・外来語、欧文…欧文表記

※原則として、ファイルまたはディレクトリの意味を3単語以内で記述し、出来る限り1単語で表現する。

※ファイル名ディレクトリ名が2単語以上になる場合は以下のようにする。

例：家庭での事例・・・case_home.html

※その名前が英語を元になっている場合は、その英単語を使用する。また1単語が長い場合は短縮形を使用するか一部を省略する。

※その名前が日本語を元になっている場合は、ローマ字(ヘボン式)を使用する。さらに1単語が長い場合は短縮形を使用するか、一部を省略する。

※ファイル名・ディレクトリ名がカタカナないし日本語・英語の組み合わせを元になっている場合、無理に英語表記に統一しない。
ただし、システム上の制約(最大文字数など)によって、数字や省略形しか許容されない場合はこの限りではない。



ヘボン式ローマ字表記一覧

あA	いI	うU	えE	おO
かKA	きKI	くKU	けKE	こKO
さSA	しSHI	すSU	せSE	そSO
たTA	ちCHI	つTSU	てTE	とTO
なNA	にNI	ぬNU	ねNE	のNO
はHA	ひHI	ふFU	へHE	ほHO
まMA	みMI	むMU	めME	もMO
やYA	いI	ゆYU	えE	よYO
らRA	りRI	るRU	れRE	ろRO
わWA	いI	うU	えE	おO
んN (M)				

がGA	ぎGI	ぐGU	げGE	ごGO
ざZA	じJI	ずZU	ぜZE	ぞZO
だDA	ぢJI	づZU	でDE	どDO
ばBA	びBI	ぶBU	べBE	ぼBO
ぱPA	ぴPI	ぷPU	ぺPE	ぽPO
きゃKYA	-	きゅKYU	-	きょKYO
しゃSHA	-	しゅSHU	-	しょSHO
ちゃCHA	-	ちゅCHU	-	ちょCHO
にゃNYA	-	にゅNYU	-	にょNYO
ひゃHYA	-	ひゅHYU	-	ひょHYO
みゃMYA	-	みゅMYU	-	みょMYO
りゃRYA	-	りゅRYU	-	りょRYO
ぎゃGYA	-	ぎゅGYU	-	ぎょGYO
じゃJA	-	じゅJU	-	じょJO
びゃBYA	-	びゅBYU	-	びょBYO
ぴゃPYA	-	ぴゅPYU	-	ぴょPYO



注意する表記

「ヘボン式」については個人の認識のズレや下記のような改定があるので注意する。

長音表記

本サイトにおいては下記の「原則」の項目に従い、改定後の表記は使用しない。

氏名のフリガナ	原則	改定後可能となる表記	例	原則	改定後の表記
オオ	O	OH	大野(オオノ)	ONO	OHNO
オウ	O	OH	河野(コウノ)	KONO	KOHNO

撥音表記

ヘボン式では「b」・「m」・「p」の前に「n」の代わりに「m」をおく。ただし判断ができない発音の場合は「n」を許容する。

NAMBA 難波 (なんば)・HOMMA 本間 (ほんま)・SAMPEI 三瓶 (さんぺい)

促音表記

子音を重ねて示す。

HATTORI 服部 (はっとり)・KIKKAWA 吉川 (きっかわ)
但し、チ (CHI)、チャ (CHA)、チュ (CHU)、チョ (CHO) 音に限り、
その前に「T」を加える。
HOTCHI 発地 (ほっち)・HATCHO 八町 (はっちょう)



ディレクトリ・ファイル命名規則

項目	説明
company	会社、企業
corporate	法人の、企業の
about	〇〇とは？
register	会員登録
contact	お問い合わせ
terms	利用規約
privacy	プライバシーポリシー
philosophy	理念
legal	特定商取引法に関する表示
recruit	採用情報
fresh	新卒採用
career	中途採用
news	ニュース、最新情報
info	お知らせ
history	沿革、歴史
greeting, message	挨拶
profile	プロフィール、経歴、略歴

項目	説明
guide	はじめての方、ガイドライン、ご案内
business	事業内容
feature	特徴、特長、特色、特性
specialty	専門、特殊性、特徴、特色
results	実績、業績
works	作品
cases	事例、実例
member	会員、構成員、メンバー
access	交通、アクセス
map	地図、案内図
faq	FAQ、よくある質問
reservation	予約
policy	方針
sitemap	サイトマップ
introduce	紹介
flow	流れ、順序
rule	ルール、規則、規定、規約

参考：<http://blog.btmup.com/web-general/contents-directory-file-name-list.html>



画像ファイル命名規則

項目	説明
logo	ロゴ画像で使用。
	例: logo.gif logo_footer.gif
side_	サイドナビゲーション画像で使用。
	例: side_xxx.gif
keyv_	キービジュアル画像で使用。
	例: keyv_xxx.gif
icon_	アイコン画像で使用。
	例: icon_xxx.gif
btn_	ボタン画像で使用。
	例: btn_xxx.gif
_on	マウスオーバー画像で使用。(ただし、原則1枚の画像を使用し、CSSでポジション指定してマウスオーバーさせる。例外の時に使用)
	例: btn_xxx_on.gif
title_	タイトル画像で使用。
	例: title_xxx.gif

項目	説明
bg_	背景画像で使用。
	例: bg_xxx.jpg
line_	罫線画像で使用。
	例: line_xxx.jpg
pht_	写真画像で使用。
	例: pht_xxx.jpg
pic_	イラスト画像で使用。
	例: pic_xxx.gif
h_	見出し画像で使用。
	例: h_xxx.gif
txt_	テキスト画像で使用。
	例: txt_xxx.gif
bnr_	バナー画像で使用。
	例: bnr_xxx.gif

注) 可能な限り、種別ごとにまとめて1枚で書き出し、CSS Spriteで表示する。

文字実体参照

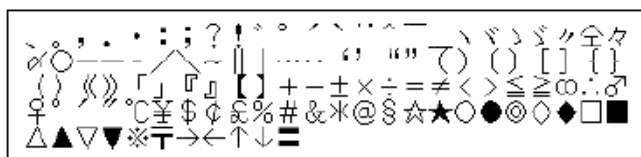
HTML内の通常テキストに存在する半角の「>」や「”」は、文字実体参照で記述する。

表示される文字	意味	マークアップ記号
	半角スペース	
&	半角アンパサンド	&
TM	トレードマーク	™
"	ダブルクォーテーション	"
>	山鏡括弧 (パンくずナビで使用)	>

※alt=""で上記記号を使用すると、ブラウザによっては文字化けする場合がありますので使用せず、通常通り記述する。

記号

下記の記号はすべての機種で問題なく表示されるので、そのまま使用してよい。



使用に注意が必要な文字・記号

半角円マーク (¥) はMacintoshだとバックスラッシュに文字化けする。下記、実体参照を使用すること。

¥

文字コード・改行コード

HTMLファイルの文字コード（「文字符号化形式」または「エンコーディング」）に関しては、必ずMETA要素（<meta>タグ）で指定する。
HTMLファイル・CSSファイル・JavaScriptファイルの改行コードはサーバーで直接ファイル編集する際に改行されないため「LF」（Unix）とする。

Unicode正規化フォーム

Unicodeはすべての言語をカバーしているために、同じ文字が複数存在している。

これらの文字の並びや機能を同じと見なすための措置として、正規化という手段を用いる。
Unicode正規化を行うことで、テキストの比較をしやすくなり検索や整列を行うときに役立つ。
ドキュメントの文字コードをUTF-8にしている場合のみ選択可能。

初期設定の**C（標準分解の後で標準合成を適用）**のままにしておくこと。

Unicode署名（BOM）

BOMとはByte Order Markの略で、16ビットのUnicodeのエンコーディング方式において、エンディアンを指定したもの。
UTF-8においては、BOMがあるとUnicodeであることを判別することができるため、UTF-8で書かれた文書であっても先頭にBOMをつけられることがある。

基本的にはBOMを含めないで使うので、**[Unicode署名を含める]**のチェックは外しておく。

画像基本仕様

使用カラーはCIマニュアルなどを参照のうえ、基本カラー、指定カラーを確認する。
画像ファイルはGIFファイルとJPEGファイル、PNGファイルのみ許容する。

■ PNG

PNG-8、PNG-24、PNG-32の3種類がある。

PNG-24とPNG-32はフルカラーに対応。可逆圧縮式。拡張子は「.png」。ファイル単体でのアニメーションの作成は不可。

PNG-8は8ビットカラー256色まで書き出し。**(※IE6でも透過指定可能)** アイコンやロゴなどを同条件で書きだした場合、GIFよりも圧縮率が高い場合が多い。

PNG-24は24ビットカラー（フルカラー）で書き出し。透過指定はできない。

PNG-32は24ビットカラー（フルカラー）で書き出し。さらにアルファチャンネル情報（8ビット）を加えたもの。
透過指定できるが、画像の容量が大きい。透過はJSなどを使用しない場合、IE7から対応。

※Photoshopの「PNG-24」表記は、透明部分にチェックを入れれば実質「PNG-32」という扱いになる。

■ GIF

汎用的に使用できる画像形式。拡張子は「.gif」。透過やアニメーションなどで使用。

旧式のガラケーなどに対応する場合はPNGが使えないため、GIFを使用する。

Photoshopで書き出しの際は、カラー設定を「特定」にする。

Fireworksで書き出しの際は、カラー設定を「すべての色を割り付け」にする。

■ JPEG

写真画像は原則的にJPEG画像で扱う。拡張子は「.jpg」。

書き出しの際は、劣化を最小限にとどめつつも、ファイルサイズを軽量化する。



JavaScriptの基本仕様

ほとんどのものはjQueryによる実装が可能。
簡単なものは基本jQueryを使用すること。

バージョン：JavaScript 1.6 jQuery 1.8.3
JavaScriptは原則外部ファイルで管理する。読み込みは</body>の直前が望ましい。

外部JavaScriptファイル一覧

ファイル名	内容	格納フォルダ
common.js	ページトップへもどるスムーズなスクロールなど、全ページ共通のjQueryに関して記述する。	/js/

命名ルール

変数名は必ず var に続けて変数名を宣言する。
原則関連する英語もしくは英語を略した単語を用い、小文字で記述する。
命名の際、関連する単語が2語以上になる場合は、2語目以降の先頭文字を大文字で記述する（キャメルケース）。

記述例

```
var countTime;
```

JavaScriptのコメント

どの動作に対してのスキプトかをコメントで記述する。



jQueryの読み込み

CDNを利用する。Googleを利用しているサイトが多い。基本的に圧縮されている軽量のminファイルを使用すること。
※jQuery1系の最新版を参照する方法は非推奨とする。※最新版に何らかの不具合があった場合、その影響を直接受けてしまうため。

記述例

```
<script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/#version#/jquery.min.js"></script>
```

script.jsの内容

script.jsには汎用的な標準スクリプトが記述されているので参照。不要な機能は削除してから新規追加すること。



表示速度の影響

WebサイトのパフォーマンスはコンバージョンやSEOに影響するので、表示速度に関して十分に配慮する。
デザイナー・マークアップエンジニアは、HTML・CSS・画像・JavaScriptなどのフロントエンドの部分を工夫することで、表示速度を大きく改善することができる。

W3Cに準拠したマークアップ

- HTMLの文書構造を理解して、的確なマークアップをする
- ソースコードはシンプルかつ論理的に

HTTPリクエストの数を減らす

画像や外部ファイルのリクエストを減らすことで表示速度を改善できる

外部ファイル

CSSとJavaScriptをインラインで記述している場合、HTMLドキュメントがリクエストされる際に毎回ダウンロードされることになるので、基本は外部ファイルで扱う。

CSSの読み込みは@importを使用しない。

head要素に記述する場合も、外部CSS→外部JSの順に読み込む。
(※インラインの記述が必要な場合は、外部ファイルを読み込んだ後に記述する)



画像の書き出し

- 実寸で書き出す（レスポンシブデザインやRetinaディスプレイ対応時は、この限りではない）
- 画像は劣化しない程度に最小化して使用。

圧縮用ツール

--<http://www.jpegmini.com/>

--<http://www.punypng.com/>

--<http://www.gigafree.net/tool/graphiccomp/caesium.html>

- JPEG・GIF・PNGを適宜使い分ける。（GIFよりもPNG8の方が高圧縮率なことが多い）

「Smush.it」

<http://www.smushit.com/ysmush.it/>

8bit-PNGの画像ファイルにはチャンクと呼ばれるメタデータも含まれているものの、すべてが表示に必要なわけではない。「Smush.it」というツールを使うことでムダな部分を省いた画像を生成することができる。

- 背景画像などは、一枚の大きな画像を使うのではなく、小さなサイズの画像を繰り返して使う。
- 必要に応じて画像の先読み（プリロード）を行う。※JavaScriptによるロールオーバーなどで画像のチラつきを防ぐ。

CSSの記述

- セレクタの指定について、子孫セレクタをできる限り減らす

※OOCSS(Object-Oriented CSS)の採用

目的別にclassを複数指定するオブジェクト指向の考え方、**OOCSS**(Object-Oriented CSS)でコーディングする。ページの読み込みスピード対策になる他、再利用可能なモジュールで構成すればコーディングにかかる工数が減らせる。

- 画像は劣化しない程度に最小化して使用
- 角丸やグラデーションなどのエフェクトは可能であれば画像ではなくCSS3を利用する
- 二重指定を極力避ける



- ユニバーサルセレクター（『*』）はレンダリングを遅くしてしまうので避ける
- PC・スマートフォンサイトのコーディングであれば色の指定は省略して記述（#FFFFFF→#FFF）
※ガラケーサイトでは認識されないことがあるので、きちんと6桁で記述する
- 固定レイアウトの場合であれば、tableタグにtable-layout:fixedを使用する

CSSの圧縮・軽量化

■CSS Optimize（CSS最適化ツール）

<http://css.webcreativepark.net/csstidy/>

（CSSの記述例）

margin:0px → margin:0

color:#FFFFFF → color:#FFF

font:1.0em → font:1em

padding:5px 10px 5px 10px → margin:5px 10px

CSS Sprite

画像を1枚の同じ画像として書き出し、CSSで必要な箇所だけを表示させる。

ナビゲーションなどによく使用する。

1枚の画像を読み込むだけでよいのでHTTPリクエストを減らすことができる。

可能な場合は積極的に利用すること。