

問題点No.	CLID	問題点	発生日	解決日	原因	解決策
1	001004	JavaScriptファイルのコメントと仕様書の文言が一致してない部分がある	2019/8/15	2019/8/15	開発工程前後のため。また、仕様書にも一部わかりにくい文言があった。	コードレビューして、コメントと仕様書を修正。
2	107002	kg, gボタンクリック時の画像に違和感がある(横にずれる感じ)	2019/8/15	2019/8/15	画像の文字が1ピクセルずれていた	ボタンの画像ファイルを再作成
3	108006	×ボタンクリック時の画像に違和感がある(上にずれる感じ)	2019/8/15	2019/8/15	画像の文字が1ピクセルずれていた	ボタンの画像ファイルを再作成
4	108005,108009,108013	×ボタン、÷ボタン、＝ボタンのマウスオーバー時に画像が横にずれる感じがする	2019/8/15	2019/8/15	画像の文字が1ピクセルずれていた	ボタンの画像ファイルを再作成
5	314006	順計算で用量計算し、小児用量 0.480g の状態で乗除算画面に遷移した直後、乗算→等号→除算→等号ボタンの順に押下すると、計算結果が1.002になる	2019/8/16	2019/8/16	計算の入力値を得るときに、メモリに高精度の値があっても、その有無を判定せずに入出力テキストの値を読み込んで入力としている。	メモリに高精度の値があるかどうかの判定を追加し、あれば計算の入力値に高精度の値を用いるようにする。
6	315001	乗除算画面で、3を2回→乗算→kg,g→×,÷→2の順にボタン押下したとき、テキストが 2 にならず、33.002 になる	2019/8/16	2019/8/16	乗算または除算ボタンの初回押下時、メモリ(gCalcMem)に入力値に入れても glsMemDetail を true にしていない	gCalcMemに入力値を入れたら glsMemDetail を true にする。
7	316001	乗除算画面で、3を2回→等号ボタンを押すと、テキスト表示が 33.00 になるが、この後、桁数ドロップダウンリストを3桁に変えても 33.00 のまま(1桁に変えても同じ)	2019/8/16	2019/8/16	乗除算をしなかったら、等号を押しても glsMemDetail を true にしていない(メモリ上に計算データなしの扱い)。	計算しない場合 (gCalcOperator == OPERATOR_NONE)の場合でも、gCalcMem に入力値を入れて、「メモリに計算データあり」とする(No.5の修正により、この解決策が可能となった)。本件とNo.6はまとめて同時に修正する。
8	CL外	乗除算画面では、ACボタンを押しても、体重テキストと成人量テキストはクリアされない仕様であるが、そのことが画面上で分かりにくい	2019/8/18	2019/8/18	乗除算画面に遷移したとき、体重テキストと成人量/小児用テキストの見た目に何も変化が無い。	乗除算画面では、体重テキストと成人量/小児用テキストの下に「HOLD」の文字が出るようにする。 仕様書を修正し、CLにも当該項目を追加する。
9	401005	強制再読み込み後、用量単位ラジオボタンのg、mg、μgの文字の部分をクリックすると、性別ラジオボタンが、それぞれ無指定、男性、女性に変わる。ただし、ラジオボタンの○の部分をクリックした場合は本現象は発生しない	2019/8/20	2019/8/20	強制再読み込みに関係なく、初期状態であっても、用量単位ラジオボタンの文字部分をクリックすれば、当該現象が発生する。 HTMLソースにおいて、用量単位ラジオボタンのlabelタグのforの値の設定が誤っていたため。	用量単位ラジオボタンのlabelタグのforの値の設定を修正する。
10	107013	μgボタンにマウスオーバーすると画像が上にずれる感がある	2019/8/21	2019/8/21	マウスオーバー時の画像の「μg」の文字がわずかにずれていた。	ボタンの画像ファイルを再作成
11	401002	Microsoft Edgeで別ページ閲覧後に戻ってくると、ラジオボタンが未選択の状態になる	2019/8/23	2019/8/24	デフォルトのラジオボタンをチェックする処理において、配列の添字が間違っていた。 誤: radios[defaultIndex].checked = (i == defaultIndex); 正: radios[i].checked = (i == defaultIndex);	配列の添字を修正
12	001003	戻り値があるのに @return コメントを書いてない関数がある ・isCalcOperating() ・isOperationPushed()	2019/8/25	2019/8/25	コメントの記述漏れ。	@returnコメントを書く。 他に同様の書き漏れがないか、return で検索して調べる。
13	CL外	ソースに使ってない関数が存在する ・isCalcOperating()	2019/8/25	2019/8/26	No.6の修正時に、乗算ボタンと除算ボタンのボタン二重押しの判定処理を修正したが、当該関数は、修正前の判定処理で使っていた関数。この修正により、当該関数は不要になっていた。	関数 isCalcOperating() を削除する。
14	320005	乗除算画面で、「1.005＝」と入力して小数点以下桁数を2桁にすると、1.01ではなく1.00と表示される	2019/8/25	2019/8/26	小数計算誤差を意識して、数値を文字列化して端数切り捨て処理をしていたが、文字列化する前に、数値として桁繰り上げ→Math.round()→数値として桁繰り下げの処理を行っていたため、誤差に対処できないケースがあった。	文字列化とMath.round()の処理順を修正し、文字列化→文字列として桁繰り上げ→数値化→Math.round()→文字列化→文字列として桁繰り下げ→端数切り捨ての順で処理する。
15	321001	乗除算画面で、小数点以下桁数を3桁にして、1÷4000000(四百万)の計算をすると、結果が「NaN」になる	2019/8/28	2019/8/30	この計算をすると、formatDecimal()関数に渡る数値が「2.5e-7」になる。このとき、この文字列の小数点を3文字分動かすと「25e-7」となって、数値として成立しなくなる。	formatDecimal()関数で、小数点をずらす前に Number.toFixed() を使って、数値を小数表現の文字列に変換してから処理する。 →デグレードのため、No.17のように再修正。
16	321005	1を10000000で3回割って10000000を3回掛けると1.0001になるはずが、0.1001になる	2019/8/28	2019/8/30	この計算をするとメモリの値が0.9999999999999998になるが、整形で四捨五入後したとき、0.9999から1.000に桁上がりするケースを考えていなかった。	整形の四捨五入で桁上りするケースに対応する。
17	320005	No.15の修正により、「1.005＝」と入力して小数点以下桁数を2桁にすると、1.01ではなく1.00と表示されるように戻った	2019/8/30	2019/8/30	Number(1.005).toFixed(18) を実行すると、1.004999999999999893 になる。	数値を小数表現に変換するときには String() を使う。それで指数表現になったときは自前の処理で小数表現に変換する。
18	CL外	画面の各項目のタイトルが茶色地に茶色文字なので、バリエーションになっていない	2019/8/30	2019/9/1	画面色の検討不足。	タイトルの地の色と文字色のコントラストを強調する。
19	CL外	液晶ディスプレイの画面をやや下方向から見ると、ボタンの色が少し暗くて、文字が見えにくいことがある	2019/9/3	2019/9/4	ボタンの色の青成分が多めだった。	ボタンの色を全体的に黄色っぽくして明るくする。
20	501001	画面幅750前後のとき、画面高さ764より大きくしても縦(小)画面にならない	2019/9/13	2019/9/13	min-height の高さ条件に「px」が書かれていなかった。	「px」の記述を追加。

21	501001	画面幅600前後で画面高さ756〜764の範囲において、仕様では縦(兼小)画面でタイトルは標準表示だが、実際は縦コ1の表示になる	2019/9/13	2019/9/22	縦(小)画面における遷移が縦(兼小)画面にも残って影響していた。	仕様とする。別紙1の記述を修正する。高さ690〜720も同様。 理由:ソースを従来の仕様に合わせるように修正すると、ステップ数が増える割に、見た目が改善する範囲が微々たるものであり、得られる利益が少ない。 →No.22の対応・機能強化により、見た目の改善度が増したため、縦(兼小)画面でも、縦コ1、縦コ2の表示区分を他の画面と同様にする
22	501002	Google Chromeでズームを80%または125%にしたとき、縦組のときは体重・成人量のエリア、横組のときは計算機ボタンのエリアがわずかにずれる(1ピクセルの隙間が空いたり、端が飛び出たりする)	2019/9/15	2019/9/22	・計算機エリアのブロックだけDIVではなくTABLEにしている、枠のサイズは自動計算させていたので、厳密なサイズ調整がしづらい。そのため、テーブルのサイズがピクセル単位で小数になっていた。 ・blockよりもtableの高さを1ピクセル小さくしているので、ズームすると隙間が開く。	・計算機エリアも他のブロックと同様DIVにして、サイズをすべてピクセル単位で整数にする。 ・blockとtableの高さを同じにする。 →ズームを100%以外にすると、どのようにサイズを設定しても、画面上のサイズが小数になることがある。そのために多少枠がずれることがある。DIVではなくTABLEの方がずれにくく、ソースの可読性も向上するので、DIVをやめてTABLEにする。 ただし、ズームしたときに枠がずれることがあるのは止むなしとして、CLにもその旨を記述する。
23	501005	IE11でズームを80%にしたとき、入出力エリアの余白が狭くなって見た目に違和感がある。操作には支障ない	2019/9/15	2019/9/16	入出力エリアの高さが十分でない。	入出力エリアの高さを56pxにする。iPhone用画面では、フォントが小さいため、46pxとする。
24	001003	外部仕様書に、小児体重が成人体重を超えている場合、小児薬用量の計算値は成人量を超えないことが書かれてない	2019/9/15	2019/9/16	仕様書の記述漏れ。	仕様書の5.1節に記述する。
25	501001	画面幅680前後で画面の高さを640前後から徐々に小さくすると、高さ633で一瞬、横(極小)・縦コ2になってから横(極小)・標準になる	2019/9/16	2019/9/16	CSSと別紙資料の修正漏れ。 縦(兼用小)と横(極小)の境界は画面高さ634が正しいが、632になっていた。	CSSと別紙資料の境界値の記述を修正する。
26	501001	別紙1の番号17のあたりで画面幅を少しずつ小さくしたとき、縦(小)になった後に一瞬、縦コ1の表示になってから、また標準の画面に戻ることもある	2019/9/22	2019/9/22	CSSで、モバイル用中画面のmin-widthとモバイル用小画面のmax-widthが同じ値になっていた。	・モバイル用中画面の設定からmin-widthを除外する。 ・縦方向コンパクト化一旦解除の設定にモバイル用小画面用を追加する(中画面→小画面に遷移するときのため)。
27	501003	Microsoft Edgeで別紙1の番号14のあたりで画面の高さを少しずつ小さくすると、境界ギリギリのところまで画面がわずかにみ出る	2019/9/23	2019/9/23	Edgeでは、リストボックスの高さがGoogle Chromeより2px大きいので、その分、モバイル用中画面・縦方向コンパクト1の画面高さがChromeより大きくなる。	すべての画面の高さの下限を4pxずつ大きくする。 モバイル用中画面・縦コ1は996から1000に、iPhone用画面・縦コ1は660から664に修正する。
28	501001	現象としては発生していないが、min-width: 756pxとmax-width: 756pxの記述が重複しているため、別紙1の番号18あたりで画面幅を少しずつ小さくすると、一瞬、縦(小)・縦コ2の表示になってから、また縦(小)・標準の画面に戻る可能性がある	2019/9/23	2019/9/23	No.26の対策漏れ。	・min-width: 756pxの設定を削除する。 ・同様の修正漏れが無いか、再度確認する。→確認した結果、修正漏れはここだけだった。 ※max-width: 780pxとmin-width: 780pxの記述が重複しているが、ここはmax-heightの別条件で分けられており、2つの条件が同時に発生することはないので、問題ない。
29	CL外	iPadでモバイル用大画面を表示させると、入出力テキストが上のブロックにはみ出て表示される(viewportのメタタグを削除したHTMLで発生)	2019/9/24	2019/9/28	体重・成人量のセルの高さがフォントサイズの割に小さい。 また、入出力のセルの高さも、フォントサイズに比べてギリギリの大きさ。	体重・成人量のセルの高さを8ピクセル、入出力のセルを10ピクセル大きくする。 ただし、外枠の大きさは変えないようにする。

30	CL外	Firefoxでズームを80%にすると、横組の場合に計算機エリアが画面からはみ出る	2019/9/24	2019/9/28	Firefoxでは、border-collapse: collapse; box-sizing: border-box; の指定の場合、枠線半分の大きさがセルに追加されるので、その分セルが大きくなり、枠線がずれたり、セルがはみ出たりすることがある。	Firefoxの場合だけ、横組ではtable-C、縦組ではtd-A3をそれぞれ幅を1ピクセル小さくする。 ※論理的には、0.5ピクセル小さくすれば良い筈だが、それでもズーム67%のときに画面からはみ出るので、1ピクセル小さくすることにする。 →No.31と併せて根本解決を図る。詳細はNo.31参照。
31	501007	iPadでiPhone用画面以外のサイズの画面を表示させると、一番上の段の真ん中の縦線と2番目の段の真ん中の縦線がわずかにずれて見える	2019/9/26	2019/9/29	No.30と同様、セルの大きさが枠線分ずれている。	他のブラウザでも同様の問題が出ないように、以下のようにして、No.30と併せて根本的解決を図る。 ・外枠outer-1の右端と下端をborderではなくbackgroundにする。枠のサイズは変えない。 ・左右のセル間の境界線は、左側のセルでborder-rightで描画していたが、右側のセルでborder-leftで描画するように改める。 ・table-Aでは、右側のセル(td-A3)の幅を指定せず、左側のセル(td-A1)の幅を指定する。また、td-A1の幅を0.5px小さくする。 ・calculator-outerframeの右側と下側のpaddingを0にする。
32	CL外	Google Chromeでズーム80%、縦(小)画面の状態で×,÷ボタンを押下すると、ACボタンとCボタンの位置が少し変わる	2019/9/28	2019/9/29	9/24以前のバージョンでも同様の現象が発生する。ズーム100%、75%では発生しない。Microsoft Edge、IE、Firefoxでは発生しない。 Google Chromeのズーム80%では、以下のボタンの位置が変わる。 ・縦(大): 数字ボタン ・標準、やや小さめ: 数字ボタン ・縦(小): AC、Cボタン ・兼用小: 数字ボタン 原因は、ズームにより計算機部分のボタンと余白のサイズが変わるためと考えられる。	以下の理由により、この件は対処保留とする。 ・他のブラウザでは現象が発生しない。 ・ズーム率を変えると発生しなくなる。 ・操作に大きな影響はない。 なお、一番下のボタン群にmargin-bottom: 0を適用させれば、縦(小)画面ではボタンが動かなくなるが、兼用小画面では、それでもボタンが動く。 →右下端のボタンのみにmargin-bottom: 0 !importantを設定することによって回避できたので、これを以て対処とする。 また、marginとpaddingの設定を全体的に見直した。