

Типичные ошибки

21.12.11

1. Неверное отображение при отключенном js:

а) Кастомных формы - при отключении js все элементы формы должны сохранять свое место положение, а форма в целом соответствовать дизайну. Поэтому сначала стоит сверстать форму с дефолтными элементами, по максимуму подогнать ее к дизайну, а уже потом подключать кастомные формы и прописывать окончательные стили.

б) Cufon - при отключенном js размер, цвет и положение текстового элемента должно максимально соответствовать дизайну. Если для web safe аналога нужно задать другой размер шрифта (или иные параметры), это можно сделать через класс `.cufon-active`:

Для обычного шрифта:

```
#content h2{
    font-size:20px;
}
```

Для куфона:

```
.cufon-active #content h2{
    font-size:26px;
}
```

в) Автоскейлинг меню - при отключенном js, на ссылках должны быть отступы между элементами, чтобы навигация не сливалась в одну строку. Это легко реализуется через класс `.scaling-active`:

Без js:

```
#nav a{
    padding:3px 5px;
}
```

С js:

```
#nav.scaling-active a{
    padding:3px 0;
}
```

2. Отступы реализованы с помощью line-height.

В вертикальных списках (side-nav) и дропдаунах расстояние между элементами списка нужно задавать маргином или падингом, а не большим значение line-height! Потому что, если одна из ссылок в таком списке будет занимать 2 строки, визуально не возможно будет отличить ее от 2ух разных элементов списка.

3. Горизонтальный скролл при 1024*768.

а) Если дизайн страницы шире 1003px, но контентная значимая часть влезает в 1003px нужно воспользоваться методами описанными в статье «Центрирование 3-мя блоками» на тулзах. Обязательно проверяйте результат. При сжатии окна по горизонтали тени, декор, пустые поля и прочие не значимые части должны обрезаться окном браузера, без появления горизонтального скrolла. Скролл должен появляться только когда границы окна браузера достигнут значимой части страницы, что регулируется значением min-width на `<body>`.

б) Если контентная значимая часть страницы шире 1003px, обязательно предупредите менеджера в CRM о неизбежности горизонтального скrolла при 1024*768.

4. Изображения вырезаны вместе с отступами:



Нельзя вырезать картинки с отступами, это может вызывать недопонимание и проблемы при замене изображений. Картинки должны иметь правильные размеры, а отступы реализовываться маргинами.

20.07.11

5. В *pricing* активный элемент не ссылка:

Активный элемент показывает страницу на которой мы находимся в данный момент, поэтому ссылаться некуда.

6. При *Optimized for Load speed* для изначально скрытых элементов необходим *preLoad(ep)*.

Для предварительной загрузки изображений у скрытых элементов, таких как табы, попапы, выпадающие меню, добавляем в конец нашего html пустой блок с классом *bg-preload*. Для него прописываем загрузку таких же фоновых изображений через свойство *multiple backgrounds CSS3* и ставим *background-position: -9999px*:

```
.bg-preload{
  height:0;
  font-size:0;
  line-height:0;
  background:url(images/bg-tab.gif) no-repeat -9999px,
  url(images/bg-pop-up.gif) no-repeat -9999px,
  url(images/bg-drop.jpg) no-repeat -9999px;
}
```

7. Обнуленный *Outline* на инпутах и других элементах

Нельзя обнулять *outline*, если только об этом не попросил заказчик.

8. Не весь текст в логотипе



```
<div class="header-holder">  
  <strong class="logo">  
    <a href="#">Berea</a>  
  </strong>  
</div class="header-info">
```

Текст в ссылке должен дублировать ВСЁ! Что написано в логотипе.

9. При ресайзе шрифта обрезаются иконки на строчных элементах



Это критично при Resizable Fonts, но лучше следить всегда, т.к. некоторые браузеры умеют ресайзить шрифты и без нашей помощи.

Такие элементы лучше делать блочными\строчно-блочными и задавать минимальную высоту.

10. Дефолтный список в контенте прописывается глобально (относительно #content) без какого либо класса.

HOW DO I KNOW IF I AM BURNING CORRECTLY IN MY WOODSTOVE OR FIREPLACE?

Your woodstove, fireplace, or barbecue must be used in accordance with the manufacturer's instructions. Also, the amount of smoke produced by these devices must be minimized. The smoke coming out of your chimney should be almost colorless and thin. Thick, white or black smoke indicates that your fire is not receiving enough air. Woodstoves, fireplaces, and barbecues should be used in such a way as to minimize the impact on neighbors. Here are some additional tips:

- Burn only dry fuel. Moist wood gives off more smoke and may produce up to 44% less heat.
- Ideally, burn wood that has been split and dried for one year.
- Never burn painted, stained or treated wood, colored newsprint, plastic, cardboard, garbage, diapers or magazines.

One myth, which is perhaps the most damaging to air quality and potentially damaging to health and safety, is that there are benefits to starving a wood stove for adequate combustion of air.

Неверно:

```
#content ul.list{..}
```

Верно:

```
#content ul{..}
```

Это критично для имплементации и желательно во всех остальных случаях, т.к. большинство проектов в конечном итоге все равно имплементируются.

11. Не предусмотрен нижний отступ на <p>, в превью (список) постов.

Довольно часто первый параграф может быть настолько коротким что в превью их поместится 2. Поэтому если позволяет дизайн следует избегать записи:

```
.post p{margin:0;}
```

12. Это «список определений»

Forget the Whales. Save the Plankton: 500 Points

Save Our History: 1,000 Points

Rain Forest Conservation: 200 Point

a Diagnostic Imaging 2011 All Rights Reserved
website designed: [Vienna Creative](#)



Ghana:
[Antipodeans](#)
[Abroad](#)

PH: 505.298.0301
FAX: 505.298.5203
EMAIL: warren.keipper@gmail.com

Почему-то игнорируется тег <dl>, а ведь во многих случаях он не только семантичен но и удобен. Кроме того им очень удобно, оформлять свойства продукта и страницу FAQ.

13. Рамка\тень картинки поста тянется только по-вертикали



WELCOME TO Zia Diagnostic Imaging (ZDI)

We are a group of board-certified radiologists with a wide range of specialty training and expertise in diagnostic and interventional radiology. Providing professional radiology services to Health Care facilities in New Mexico, ZDI radiologists are committed to the needs of our patients, referring physicians and healthcare delivery partners. Our focus is on maintaining patient and physician responsive clinical services with accuracy, efficiency, compassion and integrity.

[Learn more about ZDI >>](#)

Тянуть рамку или тень только по-вертикали это практически все равно что не делать ничего — картинки либо будут все одинакового размера либо все разного. Вариант же когда когда картинки поста имеют одинаковую ширину но разную высоту крайне мало вероятен (исключение: требование клиента или явный пример в дизайне).

Итого: либо тянем во все стороны, либо не тянем вообще, ориентируемся на дизайн и требования заказчика.

15.06.11

1. Текст всегда должен растягиваться при удалении картинки:

Если нет других требований клиента, никогда не нужно харкодить ширину текстовой



части (описания), указать `height:1% + overflow:hidden` достаточно. Всегда предусматривайте вариант с отсутствием картинки, не зависимо от того находится она справа или слева.

Второй момент: если вы используете для позиционирования картинок один из классов `alignright` или `alignleft`, не забывайте прописать второй. Классы `alignleft`, и `alignright` - довешиваются, цмс"кой (в основном непосредственно на картинку), в зависимости от выбранного радиобатона и они прописаны по дефолту, поэтому записывать и перебивать оба нужно.

Третий момент: без необходимости не стоит заворачивать картинку в теле поста в `<div>`, вешайте классы `alignright` или `alignleft` и указывайте соответствующие свойства непосредственно на ``

2. Отсутствует корпоративный CSS Reset в проектах под имплементацию:

Недавно был внедрен новый Имплементационный CSS Reset (см. тулзы). Его обязательно нужно использовать.

Исключение составляют Magento и проекты в которых уже выбран сторонний CSS Reset (требование заказчика приоритетней). Однако стоит предупредить менеджера, что после скажем Eric Meyer Css Reset, список в посте будет неотличим от параграфа.

3. В сайдбаре отсутствуют логические блоки

Даже если нет явных границ информация на странице все равно разбивается на логические блоки т.н. «секции». Например:

Неверно:

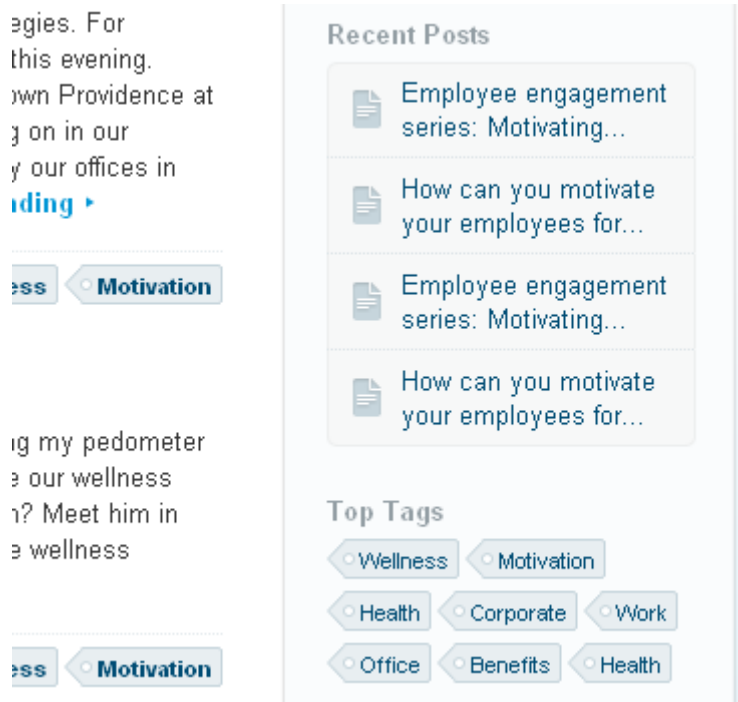
```
<div id="sidebar">
  <h3>..</h3>
  <ul>..</ul>
  <h3>..</h3>
  <ul>..</ul>
</div>
```

Верно:

```
<div id="sidebar">
  <div class="widget">
    <h3>..</h3>
    <ul>..</ul>
  </div>
  <div class="widget">
    <h3>..</h3>
    <ul>..</ul>
  </div>
</div>
```

```
</div>  
</div>
```

Логические блоки необходимы.



8.06.11

1. указано свойство **font-weight:bold** на **:hover** (меню прыгает)

Если в горизонтальной менюшке указать **font-weight:bold**; для хOVERного состояния, то при наведении мышки, вся менюшка будет прыгать в ширину.

Это происходит потому, что **болдовый** вариант шрифта шире нормального, и если такой эффект указан в дизайне, лучше его оставлять только для состояния:

```
#nav li.active a{}
```

*всегда сообщайте менеджеру что хOVER эффект будет изменен и почему

2. Пустые ссылки

Пустые ссылки недопустимы, текстовая альтернатива нужна всегда.

3. Цвет указан в верхнем регистре **#F3AE2C**

Даже если вы копируете цвет из какого-то инструмента, который выдает результаты в верхнем регистре, все равно нужно все переводить в нижний (**Ctrl+Shift+Y** в аптане для выделенного текста)

4. Звездочка **required (Name*)** задана фоном на **form label{.}**

Получается что все поля, какие не добавь, становятся обязательными. Лучше звездочку делать **спаном**: `<label>Name: *</label>` так ее будет просто добавлять и убирать.

5. Верхний отступ **#sidebar'a** задан **margin'ом** на первом блоке в нем.

Опять же, а перестановка блоков местами, если первым будет идти другой блок, то отступа не будет. Поэтому верхний отступ для сайдбара задаем по возможности

падингом на нем.

6. *Facebook плагины вставляются картинкой в html а не фоном.*

В данном случае картинка выполняет роль плейсхолдера, для будущей имплементации. Поэтому важно еще позиционировать не саму картинку, а какой-то контейнер в котором она лежит, т.к. в дальнейшем она будет заменена на fb код.



```
<div class="fb-like">  
    
</div>
```

Сюда же относится кнопка твитера и share this (в любых вариантах)

7. *Незаполненный value="" для кнопок*

Value="" кнопок заполнять нужно всегда, это и альтернативный текст, и программеры часто ориентируются на него.

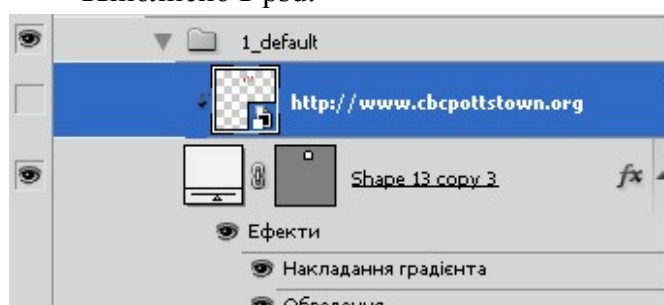
8. *Список продуктов/портфолио выполнен фоном/спрайтом, картинки не отделены от маски*

Если список продуктов/портфолио/картинки в постах выполнен в виде картинок с закруглениями, возможно отбрасывающих тень, или тень появляется на ховер (неважно, с любым эффектом) не стоит из-за эффекта делать их фоном/спрайтом.

Картинки продуктов должны быть легко заменяемыми и выполнены в html. Все эффекты(скругления, тени) должны быть по возможности отделены от картинки — в идеале заказчик должен иметь возможность вставить прямоугольную картинку без теней и получить приемлемый результат. Рассмотрим пример:



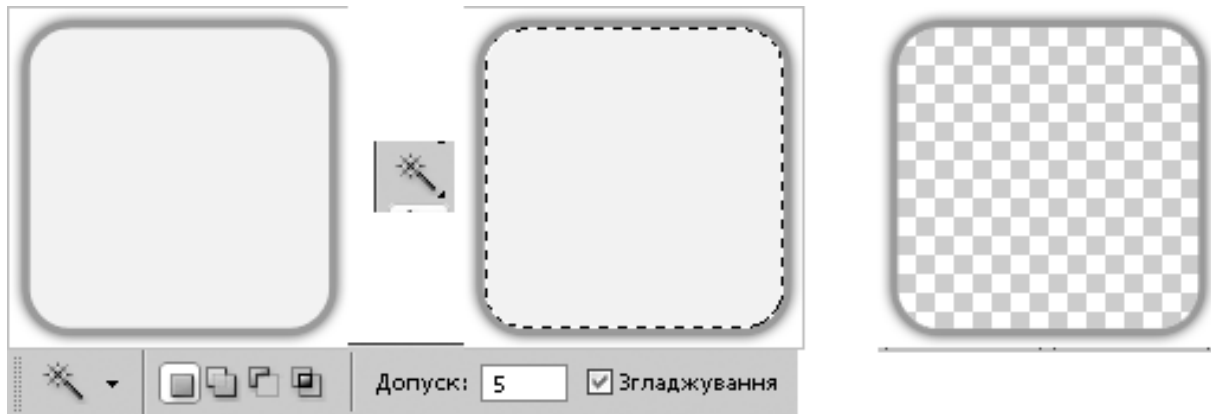
Это портфолио в виде картинок с закруглениями и на ховер появляется тень. Как это выполнено в psd:



У нас есть прямоугольная картинка(выделенный слой), и отсекающая маска(слой под выделенным) на наличие которой указывает стрелочка вниз перед иконкой картинки на выделенном слое. Если отключить слой с картинкой(справа) увидим непрозрачный слой маски который находится под картинкой, а не поверх как должно бы быть.

В идеале такие вещи хотелось бы делать CSS3 но тут нужно согласие заказчика.

Но поскольку все портфолио находится на однородном фоне, обойдемся и png маской.



копируем слой маски, выделяем ее внутреннюю часть с помощью волшебной палочки (обязательно с галочкой сглаживание — чтоб края были размытые) и удаляем (см рисунок выше).

Чтобы скопировать прямоугольную картинку нужно отключить отсекающую маску через контекстное меню слоя или Shift+LeftMouseButton на темно серой иконке Shape под выделенным слоем.

Теперь в маркаде мы сможем выполнить правильное наложение маски на любую прямоугольную картинку, повесив спрай с указанным фоном маски на абсолюте поверх картинки.



Для обеспечения какого-либо ховерного эффекта, маску выполняем в виде спрайта.

1.06.11

1. Неверный базовый цвет ссылки

Давайте, так. Порядок заполнения корпоративного CSS Reset:

- открываем дизайн, выбираем инструмент “Т”, кликаем на параграф в контенте — заполняем CSS reset для **body**: font-size, line-height*, font-family, color.
- указываем background для **body**: цвет или цвет + картинку
- находим ссылку в контентной части (если таковой нет берем свойства ссылок преобладающих на странице) — заполняем CSS reset для **a**: color, text-decoration
- находим формы — заполняем CSS reset для **input, textarea, select**: color, font-size, font-family

*кого как учили, лично я рекомендую

2. Необоснованное многократное применение !important

Использовать !important без крайней необходимости* запрещено, свойства указанные с этим параметром сложно перебивать, что может привести к трудностям при продолжении кода или апдейтах. В большинстве случаев можно обойтись простым увеличением веса селектора.

* под крайнюю необходимость в частности попадает: задание размеров для textarea, перебивание дефолтных свойств cms изначально заданных с параметром !important, и т.п.

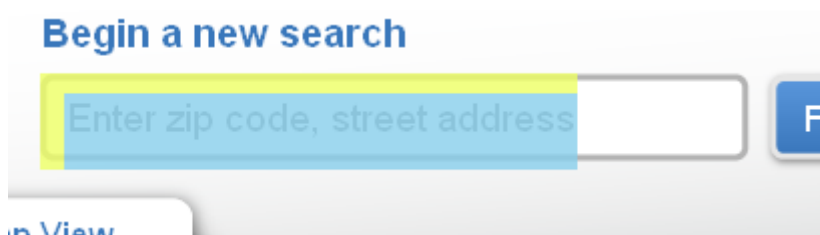
3. Неверное название CSS файла, подключаемого для старых версий IE

Согласно стандартам он должен называться только **ie.css**, а не lt7.css, ltie9.css...

4. Имена тегов в названиях картинок

Не стоит указывать имена тегов в названиях картинок (bg-body.gif, bg-ul.gif, и др.)
Лучше использовать осмысленные названия: bg-page-t.gif, bg-navigation.gif и т.д.

5. Неверный размер текстового инпута



Текстовый инпут должен занимать все отведенное ему дизайном место

Наиболее распространенные причины, этой ошибки: забыли поставить ширину, или забыли вычесть отступы.

6. На текстовый инпут задана высота

Задавать высоту на текстовый инпут НЕЛЬЗЯ (за исключением случаев когда по другому не получается), - это приводит к проблемам с кроссбраузерным выравниванием содержимого value="" по вертикали. Высоту инпута регулируем вертикальными падингами

7. Форматирование CSS (пожелание)

Селекторы с одним правилом CSS записывайте, плиз, в одну строку, так и читается лучше и места меньше занимает:

не круто:

```
#wrapper{
  padding:20px 0 0;
}
```

круто:

```
#wrapper{ padding:20px 0 0; }
```

8. Заголовки постов выполнены не ссылками

Заголовки постов всегда выполняются ссылками:

```
<h2><a href="#">Post title</a></h2>
```

это особенность функционала блогов, ссылки ведут на страницу содержащую полную версию поста . Исключение заголовков single post'a, т.е. его полной версии — он не ссылка.

9. Лишние картинки

В результате метания между форматами файлов или техниками исполнения тех или иных блоков остается много копий одного и того же содержимого в разных форматах, который не используются в конечном варианте проекта и являются мусором. А мусор нужно выбрасывать перед постановкой на QA. В помощь у нас есть софтина, которая в частности помогает искать и такой мусор (автор. akozyr)

10. Объединенный first-child

Expression для реализации свойства first-child прописан несколько раз, это **не верно**:

```
#nav li{
  z-index: expression( runtimeStyle.zIndex = 1, this ==
parentNode.firstChild ? (className += ' first-child') : 0 );
}
#footer ul li{
  z-index: expression( runtimeStyle.zIndex = 1, this ==
parentNode.firstChild ? (className += ' first-child') : 0 );
}
```

Верно:

```
#nav li,
#footer ul li{
  z-index: expression( runtimeStyle.zIndex = 1, this ==
parentNode.firstChild ? (className += ' first-child') : 0 );
}
```

Также обратите внимание, на то какой expression вы используете. Очень важно использовать тот, что содержит команду runtime, он срабатывает только один раз, при загрузке, и не тормозит работу IE6.

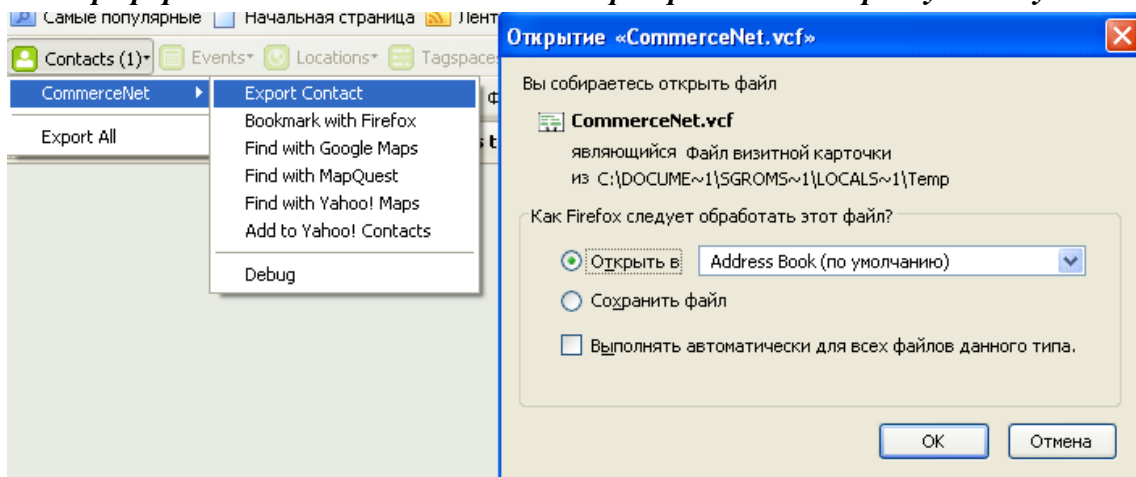
11. Использование классов для <p>

Использование классов для тега <p> крайне не желательно, как и построение на его основе какой-либо сложной многоэлементной\мультивложенной структуры, - для этого больше подходит <div>

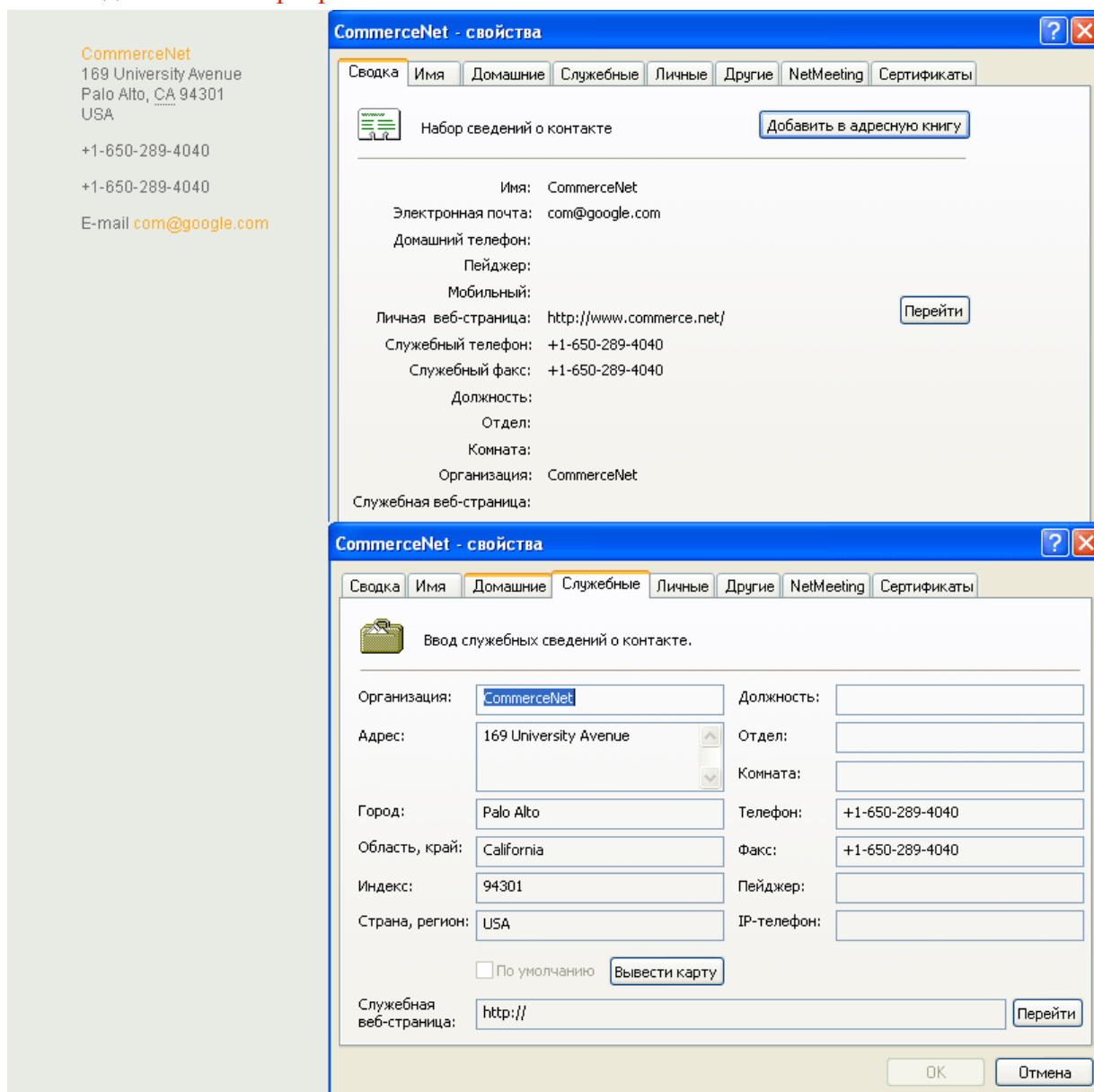
Допустимыми исключения являются:

- p.soru – копирайт в футере (и то если необходимо);
- p.note – сноска в форме (* все что со звездочкой должно быть заполнено)

12. Микроформат контактов должен экспортироваться в адресную книгу



все элементы заявленные в html должны занять свои поля, если какой-то элемент не экспортируется например телефон или адрес эл.почты это **BUG**, так же обратите внимание чтобы данные экспортировались без подписей, т.е. **сами слова phone и email не должны экспортироваться!**



Основная причина ошибок — неуказанный параметр type

13. Лишние свойства HasLayout, при отсутствии IE6 compatibility (height:1%;

Если в проекте нет поддержки IE6, то overflow:hidden становится достаточным свойством для кроссбраузерного задания HasLayout.

Напомню немного теории:

Если у вас в коде есть блок(и) со свойствами **float:left;** или **float:right;** то для их *первого общего родителя* необходимо прописать **свойства задающие hasLayout**.

Такие свойства как правило записываются парами: одно для Современных браузеров и одно для IE6. А именно:

CSS	Современные браузеры	IE7	IE6
.parent:after{ content:""; display:block; clear:both; }	+		
width:100%; или width:px; height:px; height:1%;		+	+
overflow:hidden;	+	+	
zoom:1;		+	+
display:table; + (width:px; или width:100%;)	+	+	+
display:inline-block; + (width:px; или width:100%;)	+	+	+
float:left; или float:right;	+	+	+
position:absolute;	+	+	+

Исключение: свойства задающие hasLayout прописывать не нужно если у их первого общего родителя уже есть необходимые свойства. Например: float:left;.

25.05.11

1. *грамматические ошибки в именах классов*

Если вы не уверены в правильности написания, пользуйтесь хотя бы google translate. Так же напоминаю что задавать имена классов нужно на английском языке, вне зависимости от того на каком языке выполнен проект.

Обратите внимание: **address**, **carousel**, **col**, **column**, **two-columns**, **switcher**

2. *центрирование менюх и блоков с помощью паддингов.*

Блочные элементы центрируем, с помощью `margin:0 auto; + width`.

Если ширину задать нельзя, например для основной навигации с флотнутыми li, завернем ul в nav-holder со свойством `text-align:center`; а на ul поставим `display:inline-block; + развилка для IE6,7 display:inline + zoom:1;`

3. *полная запись background: для состояния :hover*

неверно:

```
ul a{background:url(..images/bg-link.gif) no-repeat;}
ul a:hover{background:url(..images/bg-link.gif) no-repeat 0 -20px;}
```

верно:

```
ul a{background:url(..images/bg-link.gif) no-repeat;}
ul a:hover{background-position:0 -20px;}
```

4. *запись background: прописана только для состояния :hover*

неверно:

```
ul a{display:block;}
ul a:hover{background:url(..images/bg-link.gif) repeat-x;}
```

верно:

```
ul a{
    display:block;
    background:url(..images/bg-link.gif) repeat-x 0 20px;
}
ul a:hover{background-position:0 0;}
```

5. *не верный код для форм поиска и логина под Drupal*

для этой цмски сложно изменить дефолтный код для указанных форм, поэтому обязательно используйте шаблоны указанные на тулзах

6. *невыполнение пунктов Optimized for load speed*

- объединяем файлы css и js, так например print.css, form.css, grid.css (если валиден), reset.css (если валиден) должны быть включены в all.css

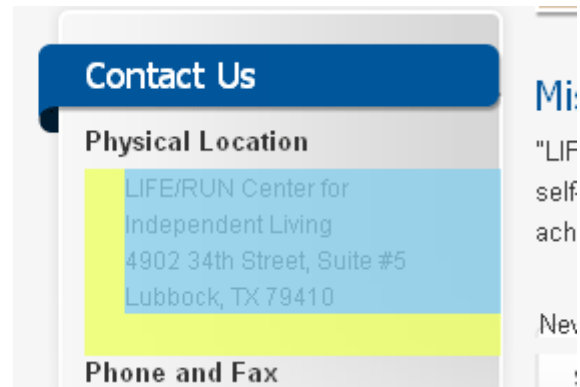
- объединяем картинки в спрайты

- оптимизируем спрайты и наиболее весомые картинки.

7. *Resizable fonts внутри блоков с фиксированной высотой*

Если элемент по какой-либо причине имеет фиксированную высоту, то ресайзить в нем текст не нужно

8. *отсутствует padding справа*



padding справа нужно указывать всегда, если нельзя определить точный размер ставим хотя бы минимальный padding(на свое усмотрение)

например:

```
#footer{padding:10px 0 10px 15px;}
```

а если текст таки достанет до правого конца, например при ресайзе фонта.

9. padding+margin

есть несколько моментов:

- не стоит часть одного и того же отступа делать padding'ом а часть margin'ом;
- если вы уже используете padding для отступа слева, и хотите добавить еще отступ снизу, то оптимальней будет скорректировать запись padding'a, чем вводить запись margin.

Не оптимально:

```
ul li{
    padding:0 0 0 10px;
    margin:0 0 5px;
}
```

Оптимально:

```
ul li{ padding:0 0 5px 10px; }
```

10. если нет IE6, png-box можно выполнять вложенными друг в друга блоками

```
<div class="box">
    <div class="holder">
        <div class="frame">
            ...
        </div>
    </div>
</div>
```

```
.box{
    width:300px;
    background:url(../images/bg-box-t.png) no-repeat;
    padding:15px 0 0;
}
.box .holder{
    background:url(../images/bg-box-b.png) no-repeat 0 100%;
    padding:0 0 15px;
}
.box .frame{
    background:url(../images/bg-box-c.png) repeat-y;
    padding:0 10px;
}
```

преимущество в отсутствии пустых блоков

11. a{display:inline-block;} * html a{display:inline;}

тут несколько моментов,
во-первых, строчные элементы и так поддаются вертикальному выравниванию;
во-вторых, даже если `display:inline-block`; вам необходим он просто не сработает в IE6,7 и ссылка останется строчной как и была, - развилка не нужна для строчных элементов.

12. Развилки под IE с валидными свойствами оставляйте в `all.css`, в отдельный `ie.css` выносим только невалидную часть (`filter`, `expression` etc.)

p.s. конечно если нет особых пожеланий заказчика.

13. лишние обнуления для `ul` при использовании `reset.css`

```
ul{  
    padding:0;  
    margin:0;  
    list-style:none;  
}
```

эти свойства в `all.css` не нужны, если подключен `reset.css`

14. отсутствие `min-height` при реализации `gif-box`

Если картинки для верхней и нижней части содержат градиенты или просто большие по высоте, то существует вероятность наложения верхней части на нижнюю при малом количестве контента.

Поэтому, необходимо указывать `min-height` для `.box .frame {}`, и не забываем развилку для ie6

15. Отсутствует маргин снизу на последних элементах:

Отступы между блоками/виджетами/боксами в контенте или сайдбаре реализуются посредством указания нижнего марджина. Но на последних блоках о нем почему-то забывают. А ведь порядок вывода блоков например в sidebar может измениться, и тогда получится что у одного из блоков в середине sidebar не будет отступа снизу.

Heading 1

Phasellus eleifend metus ac diam fermentum gravida. Praesent et velit lacus. Nam egestas, leo hendrerit venenatis vulputate, augue orci dictum nisi, a vehicula massa lacus sed odio. Phasellus nulla ligula, volutpat eget posuere eget, sodales sed arcu. Maecenas varius bibendum nulla, nec vulputate urna mollis sed. Donec laoreet vestibulum urna, et egestas metus bibendum varius. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed nulla metus, vestibulum sed dignissim vel, tristique vel urna. Nam accumsan egestas lacus sectetur a.

↓ `margin:0 0 20px;`

Heading 2



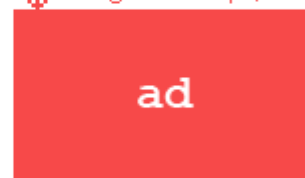
↓ `margin:?????`



↓ `margin:0 0 20px;`



↓ `margin:0 0 20px;`

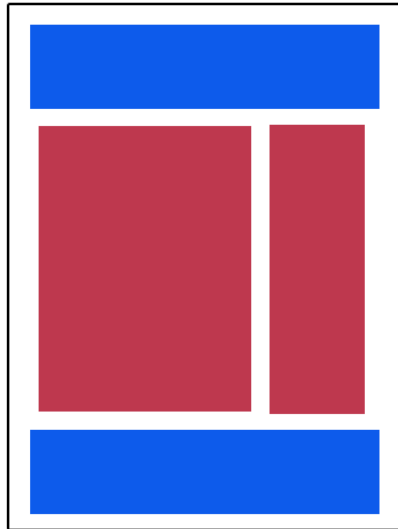


↓ `margin:?????`

18.05.11

1. *нет hasLayout на родителях флотнутых элементов.*

Некоторые из вас считают, что если не развалилось, то зачем он нужен, хотя это лишь результат стечения обстоятельств. hasLayout на родителях флотнутых элементов **НУЖЕН ВСЕГДА**.



Некоторые считают, что для красных блоков hasLayout будет обеспечивать блок с рамкой, а на нижний синий достаточно поставить `clear:both`; но это не так. Для красных блоков необходим свой холдер с заданным hasLayout.

2. *не заданы паддинги на родителя, вместо этого ставятся маржины на все внутренние элементы*

Внутренние отступы, всегда должны быть прописаны на родителе - один раз. Ставить марджины или паддинги на все внутренние элементы неверно, т.к. Невозможно предусмотреть все типы элементов.

3. *спрайты не выполняются ???*

4. *пустые альты в картинках с явным описанием*

если на картинке есть текст, или она имеет вполне определенное название, alt должен быть заполнен осмысленно.

5. *Неверный выбор формата изображений, все jpg.*

Jpg используется лишь для фото и сложных (содержащих большое кол-во цветов) градиентов. Выбор уровня качества сохранения должен быть результатом компромиса между качеством изображения и весом файла. Для однопиксельного градиента можно и 100%, а для фотографии подойдет и 60%, для фотографий содержащих надписи — может понадобиться 80% качество.

Все простые изображения (содержащие мало цветов): простые градиенты, иконки, кнопки, фоны блоков со скруглениями — должны быть gifом (за исключение случаев PNG24)

6. *SEO*

Теперь, только:

- content в коде перед sidebar;

- заголовки в правильном порядке;
- теги по прямому назначению;

7. Свойства шрифта и цвет, заданы для *ul li, p, ul a*.

Неверно:

```
.box p{color:#ccc;}
.side-nav a{font-size:15px;}
```

но верно:

`#nav a{color:#f3c}` – потому что цвет ссылки будет задан в ресетах.

Эти свойства наследуются, поэтому нужно задавать на родительских элементах

8. Не оптимальные записи CSS `#nav li ul li a`

все что находится в списке в любом случае находится в `li`, поэтому в записи селекторов этот тег можно пропускать

`#nav a`

`#nav ul a`

`#nav ul ul a`

9. Забываете про ресет фонта для инпутов

Если по всех инпутам на странице используется шрифт Trebuchet MS, то логичней его указать в корпоративном ресете, чем задавать для каждого инпута.

10. Запрещено юзать без класса, *div* и строчные элементы (*a, span, em, strong*) используемые не по прямому назначению

Исключение для строчных элементов являются только очень конкретные записи, или элементы в которых нереально использования дополнительных тегов:

`.button span{}` - для реализации тянущегося фона, - вряд ли будут дополнительные теги внутри готовой кнопки выполненной по дизайну;

`#nav a em{}` - для той же цели, - вряд ли будут дополнительные теги внутри навигации выполненной по дизайну;

`.coru span{}` - вряд ли эта запись когданибудь изменится.

неверно:

```
.post strong{font-size:16px;}
```

```
.news strong{display:block;}
```

но верно:

```
.post .heading h2 strong{font-size:16px;}
```

```
.news strong.title{display:block;}
```

11. *padding* на `<a>` в инлайновом меню причем с одной стороны, а с другой *margin*

Меню вида:

Text | Text text | Text | Long text text | Text

- на `li` вешаем `display:inline` + сепаратор + падинги слева-справа

12. Необоснованное использование двойных, тройных классов *.box-holder .box-frame, info-box-holder*

Неправильное решение:

```
.info-box {}  
.info-box-holder {}  
.info-box-frame {}
```

Правильное решение:

```
.info-box {}  
.info-box .holder {}  
.info-box .frame {}
```

Но в случае если на родителе стоит id:

```
#main {}  
.main-holder {}  
.main-frame {}
```

13. Использование универсального div.holder

```
.holder{clear:both; width:100%;}
```

для обеспечения hasLayout не приветствуется, т.к зачастую приводит к излишним оборачивающим конструкциям или появлению на многих дивах вторых классов

14. отсутствует закрашивание бг-цветов в случае использования непрозрачных бг-имаджей

Необходимо по возможности задавать альтернативный цвет фона, на случай если картинка не загрузиться или будет долго грузиться

15. фон на боду 0 50% уезжает при уменьшении окна браузера

min-width на html зачастую решает проблему