

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS GRADUAÇÃO LATO SENSU
ESPECIALIZAÇÃO EM DESENVOLVIMENTO WEB

LUCAS EDUARDO CARNEIRO ALMEIDA

ICOMP UPDATE:
REESTRUTURAÇÃO COMPLETA DO CMS

MONOGRAFIA DE ESPECIALIZAÇÃO

Londrina
2015

LUCAS EDUARDO CARNEIRO ALMEIDA

ICOMP UPDATE:
REESTRUTURAÇÃO COMPLETA DO CMS

Via: @ Á Á Ô } & • ë [Á Á Ô ~ | • [Á] ! ^ • ^ } ca [Á e
 Ö ä ^ ç | ä Á Á Ú ^ • ~ ã ä Á Á 5 • E Ö | ä ~ ä e [Á ä
 W } ç ^ ! • ä ä Á ^ & } | 5 * ä á ^ á | ä [Á ä ä |
 ä [Ö é {] ~ • Á } ä | ä ä { [Á ^ ~ ä ä | Á ä ä ä
 [à ^ } 8 è [Á [Á | ä Á Á •] ^ ä ä ä { Á
 Ö ^ • ^ } ç [| ç ^ } ç Á ^ à È

Orientador: Prof. Me. Thiago Prado de Campos.

Londrina

2015



TERMO DE APROVAÇÃO

Título da Monografia

ICOMP UPDATE: REESTRUTURAÇÃO COMPLETA DO CMS

por

Lucas Eduardo Carneiro Almeida

Esta monografia foi apresentada às 11h00 do dia **13 de novembro de 2015** como requisito parcial para a obtenção do título de ESPECIALISTA EM DESENVOLVIMENTO WEB. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho APROVADO COM RESTRIÇÕES.

Prof. Me. Thiago Prado de Campos
(UTFPR)

Prof. Esp. Frederico de Figueiredo Siena
(UTFPR)

Prof. Me. Rodrigo Freese Gonzatto
(PUC-PR)

Visto da coordenação:

Prof. Me. Thiago Prado de Campos
Coordenador da esp. em Desenvolvimento Web

Prof. Me. José Luis Dalto
Coordenador de Pós-Graduação Lato Sensu

AGRADECIMENTOS

Agradeço aos amigos que me ajudaram no decorrer do projeto, que influenciaram direta ou indiretamente. Agradeço a Laís, minha esposa que aguentou o stress e as noites mal dormidas.

Agradeço ao programador back-end que concluiu com maestria e aceitou implementar o sistema em todos os sites desenvolvidos pela agência.

Agradeço a agência ICOMP e ao Prof. Dhiego Bicudo por me liberar durante horas e por, em algumas ocasiões, emprestar os equipamentos necessários para por o projeto em prática.

Também gostaria de agradecer meus pais por terem me dado apoio e suporte para alcançar o máximo possível na minha área de atuação profissional.

Por último, gostaria de agradecer ao Prof. Thiago Campos por contribuir diretamente com esse projeto.

RESUMO

O projeto visa documentar os atuais problemas do sistema de gerenciamento de conteúdo da Agência Digital ICOMP e suas respectivas resoluções, descrevendo os passos de todas as áreas envolvidas no projeto. Todas as páginas serão recriadas a partir de uma nova estrutura padrão, aplicando novos estilos e revisando todas as informações do painel anterior. A parte visual do sistema também será reformulada para se adequar às tendências atuais de design e melhorar a usabilidade para o usuário final. Esse novo sistema será desenvolvido de forma responsiva para que possa ser executado em qualquer tipo de aparelho que contenha um navegador e internet. Ao final do processo, serão realizados alguns testes para comprovar a eficiência do novo sistema.

Palavras-chave: painel administrativo, admin, sistema

ABSTRACT

This project aims to document the current problems of Agência ICOMP's content management system and its resolutions, describing the steps in all of the areas of the Project. All the webpages will be recreated based on a new standard structure, applying new styles and reviewing all the informations of the old system. The system's visuals are going to be recreated so it can fit the new design tendencies and to increase the usability for the final user. This new system will be developed responsively so it can be executed in any kind of device that have a browser and internet. At the end of the process, tests are going to be run to prove the new system's efficiency.

Keywords: administrative panel, admin, system

LISTA DE ILUSTRAÇÕES

FIGURA 1: Recorte do topo do sistema antigo	10
FIGURA 2: Representações de kits de interface do usuário	13
FIGURA 3: Exemplo de projeto estrutural	14
FIGURA 4: Wireframe e projeto gráfico completo	15
FIGURA 5: Exemplo de renderização pré-definida	16
FIGURA 6: Exemplo de aplicação básica de CSS	17
FIGURA 7: Exemplo de integração entre HTML e CSS	17
FIGURA 8: Exemplo de personalização de elementos básicos	18
FIGURA 9: Exemplo de compilação de arquivo SASS para CSS	19
FIGURA 10: Exemplo de compactação de código com o mixin	19
FIGURA 11: Exemplo de aplicação de atributo CSS com SASS + Compass	20
FIGURA 12: Versões do mesmo site para tamanhos de tela diferentes	21
FIGURA 13: Estrutura padrão por relevância	23
FIGURA 14: Divisão de áreas do sistema anterior	24
FIGURA 15: Nova divisão de áreas	25
FIGURA 16: Nova identidade visual do Update	25
FIGURA 17: Paleta de cores	26
FIGURA 18: Família tipográfica	27
FIGURA 19: Lâmina com os elementos do sistema anterior	28
FIGURA 20: Antigo e novo campo de texto	28
FIGURA 21: Antigo e novo campo de seleção	29
FIGURA 22: Antigos e novos botões	29
FIGURA 23: Comparativo entre títulos novos e antigos	30
FIGURA 24: Nova paginação	30
FIGURA 25: Elementos atualizados	31
FIGURA 26: Novo topo e navegação	31
FIGURA 27: Novo rodapé	32
FIGURA 28: Antes e depois da tabulação de dados	32
FIGURA 29: Visão geral da lista de informações	33
FIGURA 30: Melhora na visualização das galerias	33

FIGURA 31: Itens da lista de vídeos com ou sem destaque	34
FIGURA 32: Visão geral do cadastro de galerias de fotos	35
FIGURA 33: Visão geral da tela de login	36
FIGURA 34: Texto de recepção	37
FIGURA 35: Estrutura de classes para organização visual dos elementos	38
FIGURA 36: Estrutura de classes para botões	39
FIGURA 37: Representação da interação do usuário com o item da lista	39
FIGURA 38: Personalização de títulos de músicas	40
FIGURA 39: Visão geral do sistema em diferentes tamanhos de tela	41
FIGURA 40: Exemplo de resposta negativa para responsividade	42
FIGURA 41: Resposta do Pagespeed da página de login	43

SUMÁRIO

1 INTRODUÇÃO	9
2 RECURSOS DISPONÍVEIS	13
3 PLANEJAMENTO	23
4 DESIGN	28
4.1 Elementos de Formulário	28
4.2 Títulos	30
4.3 Paginação	30
4.4 Elementos Estruturais	31
4.5 Exibição de Dados	32
5 DESENVOLVIMENTO FRONT-END	38
6 TESTES	42
7 CONSIDERAÇÕES FINAIS	45
REFERÊNCIAS	47

1 INTRODUÇÃO

Com a tecnologia caminhando a passos largos, as empresas tem sofrido para se manter ativas no mercado digital. A maioria das acaba recorrendo para serviços gratuitos (como o Wordpress) para evitar custos e economizar tempo. As poucas que decidem desenvolver um sistema próprio recorrem a kits de interface prontos para facilitar o desenvolvimento.

Na agência ICOMP, os desenvolvedores optaram por desenvolver um sistema próprio sem nenhum kit ou serviço externo. O problema é que a última atualização desse sistema foi feita em 2011. Atualmente o sistema está defasado tecnicamente, apresentando inúmeros erros decorrentes de diversas adaptações sem supervisão ou planejamento. Essas adaptações geralmente se originam em novas necessidades, novos módulos requisitados pelo cliente e até mesmo devido à evolução da parte visual do site. A implementação do sistema funciona de acordo com o projeto, sob demanda. Não existe um padrão de campos ou funcionalidades bem definidas.

Existem vários problemas como a interface é antiga, a falta de uma divisão coerente de informações mais e menos relevantes e um dos principais problemas é a falta de um padrão visual eficiente para campos de formulário, títulos, listagem e tabulação de dados. Muitas vezes, os campos não seguem uma ordem lógica e coerente.

Alguns formulários precisam de botões para envio de vários tipos de arquivos como músicas, fotos, arquivos comprimidos, etc. Uma biblioteca foi utilizada para facilitar o processo de envio desses arquivos. O sistema ficou mais rápido, porém os botões, que eram padrão e não foram estilizados, destoaram totalmente dos botões restantes da interface.

Em 2008, quando o sistema foi desenvolvido, era comum desenhar as interfaces com, em média, 1000 pixels de largura. Essa medida atendia ao tamanho de tela mais comum até então, que era 1024 por 768 pixels. Os mil pixels eram o resultado de uma conta simples: subtraia-se a largura da barra de rolagem lateral do navegador, que era de aproximadamente 20 pixels dos 1024 pixels, resultando em

1004 pixels. Os 4 pixels restantes serviam para garantir que, em navegadores diferentes, a largura da barra de rolagem não interferisse na interface do sistema.

Em alguns casos, os 4 pixels da margem não eram considerados suficientes pelo fato de a interface ficar colada nas laterais da janela do navegador. Provavelmente por esse motivo, o gerenciador da ICOMP foi criado com um respiro de aproximadamente 20 px* dentro da área dos 1000 pixels. Nessa época, o espaço preenchia muito bem a tela.

Um problema visual menos técnico do sistema é que o nome dele, Update, aparece apenas escrito no topo, no canto direito. Com isso, o nome não é reconhecido por quase nenhum cliente, que geralmente o chama de admin. Na lateral esquerda do topo fica o logo da agência dona do sistema, como é possível observar na imagem abaixo:

Figura 1: Recorte do topo do sistema antigo



Fonte: Acervo pessoal

As dificuldades técnicas são ainda maiores pelo longo tempo sem manutenção. As bibliotecas de terceiros utilizadas no sistema inicial não foram atualizadas. Algumas já não tem mais documentação, outras foram descontinuadas pelo desenvolvedor, outras ainda não são mais necessárias e ocupam um espaço precioso.

Uma das primeiras dificuldades encontradas no decorrer dessa reestruturação é que não havia nenhum tipo de versionamento. Todas as novas ferramentas eram instaladas nos arquivos aproveitados do site anterior. Com o tempo, algumas páginas foram ficando sem uso e não já figuravam mais entre as versões mais recentes. Um exemplo disso é a página de gerenciamento de vídeos que existia a algum tempo atrás, mas já não faz mais parte das novas "versões".

Alem das páginas perdidas, a falta de versionamento acabou por deixar sistema vários arquivos inúteis para o projeto. Em uma das versões anteriores, uma biblioteca de edição de texto foi usada para o administrador cadastrar notícias. Esses arquivos ainda estão em todos os painéis mesmo não sendo úteis.

Algumas dessas páginas incluem bibliotecas que já não estão mais presentes ou já não são mais úteis. Todas as vezes em que o sistema é implementado em um novo site, essas chamadas tem de ser removidas manualmente de cada página, o que acarreta um desperdício de tempo relativamente grande durante o desenvolvimento.

Na versão atual, as páginas tem um esqueleto pronto e os conteúdos são inseridos de maneira assíncrona. Na parte estética, não é de todo ruim pois o sistema mostra que está carregando a página, dando a impressão de que o carregamento da página é rápido. Na parte técnica, já não é tão bom pois todas as informações são adicionadas na página por outro arquivo, dificultando a edição das informações.

Todas as alterações básicas no html do sistema tem de ser feitas nos arquivos javascript, trazendo diversos problemas de manutenção. Como o html tem que ser escrito fora de seu ambiente nativo, muitos recursos dos editores de código não funcionam, dificultando a inspeção do código e tomando um tempo extra nas manutenções.

Levando todos esses problemas em consideração, seja pela falta de apelo visual ou pela defasagem técnica, é claro que o sistema precisa de uma atualização. Esta poderia ser parcial se alguns cuidados básicos já citados aqui como o versionamento e a manutenção periódica fossem tomados. Como esse não é o caso, a atualização completa do sistema é a mais indicada.

O principal objetivo dessa atualização é facilitar a implementação do sistema, diminuindo o tempo gasto e aumentando a qualidade do painel. Outros objetivos são torná-lo mais atrativo para o responsável pela gestão do conteúdo do site, agregar valor ao serviço prestado pela agência, melhorar o desempenho e, por fim,

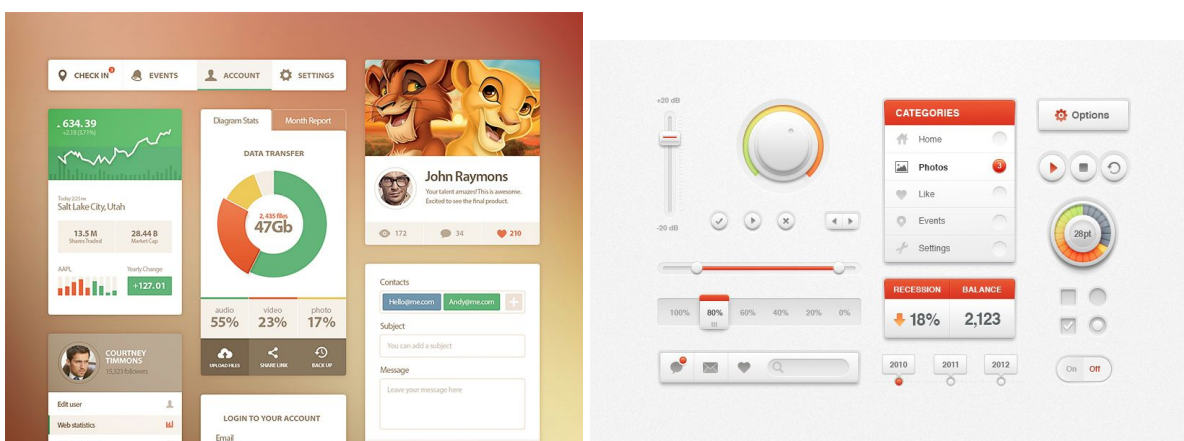
disponibilizá-lo de forma responsiva e acessível.

Com as mudanças, o desenvolvedor ganha tempo na hora do desenvolvimento por não precisar fazer tantas alterações no código fonte a cada implementação. Também não precisará mais duplicar páginas por falta de módulos antigos, já que o sistema será arquitetado de acordo com as necessidades atuais da agência e será atualizado frequentemente.

2 RECURSOS DISPONÍVES

A interface do usuário, por definição, é o espaço onde a interação entre humano e máquina ocorre. No campo do desenho para web, a interface do usuário é pensada de acordo com as principais funcionalidades do sistema. Assim, é possível identificar quais serão os elementos mais utilizados e que merecem uma atenção maior que a comum.

Figura 2: Representações de kits de interface do usuário



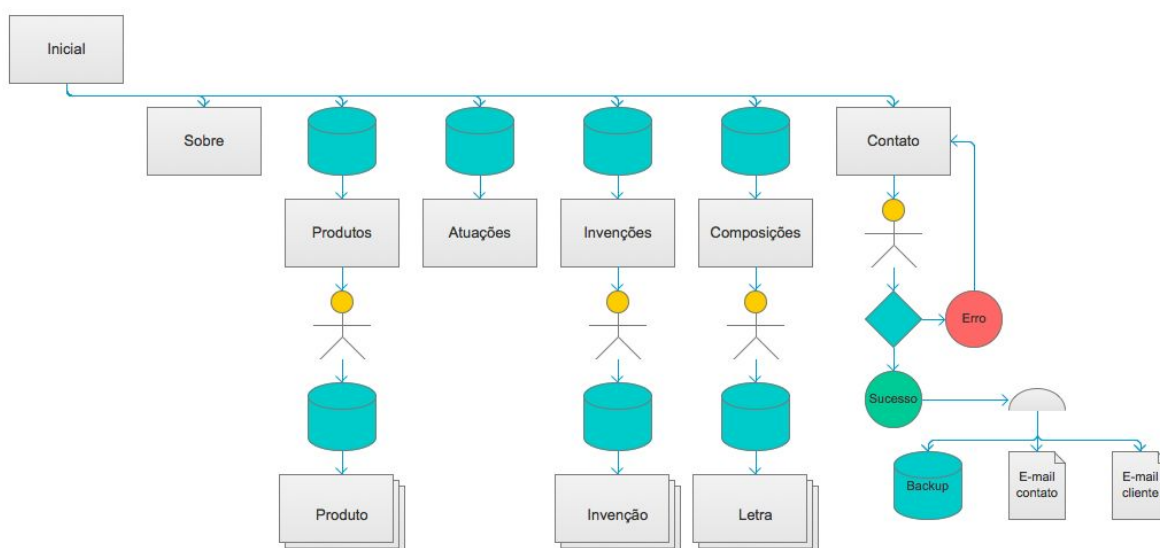
Fonte: portfolio de Mike da Creative Mints

Ambos foram criados por Mike, designer na Creative Mints de Praga, República Checa. Entre os dois, é fácil notar as diferenças e semelhanças. À esquerda, uma interface criada para um aplicativo de gerenciamento de dados onde se destacam três elementos: gráficos, campos de formulário, itens de menu entre outros. À direita, uma interface criada para um sistema com players de música. Nesse caso, se destacam os controles de volume, os botões de ação e a barra de notificações.

Durante o processo de criação de uma interface, a usabilidade é um conceito que deve ser explorado do início ao fim. Segundo Nielsen, 2007, p. 21, "no que diz respeito aos usuários, cada página deve justificar sua importância quando chamada. Se uma página não fizer isso imediatamente e de maneira clara, eles vão para outro site.". Levando isso em consideração, um sistema deve ser extremamente útil para o cliente, já que é a seção do site onde o mesmo passará a maior parte do tempo.

Junto com a usabilidade, uma boa arquitetura da informação contida no painel administrativo é necessária. São três as motivações básicas da arquitetura de informação: estruturar a informação, localizar e disponibilizar a informação e conectar fontes de informação. De acordo com M. C. Siqueira, 2005. p. 72, "essas motivações são originadas na necessidade de vencer a dispersão das informações contidas em uma organização.". Criar um mapa do sistema ajuda nessa batalha contra a dispersão das informações. Abaixo, um exemplo de projeto estrutural para um site simples:

Figura 3: Exemplo de projeto estrutural



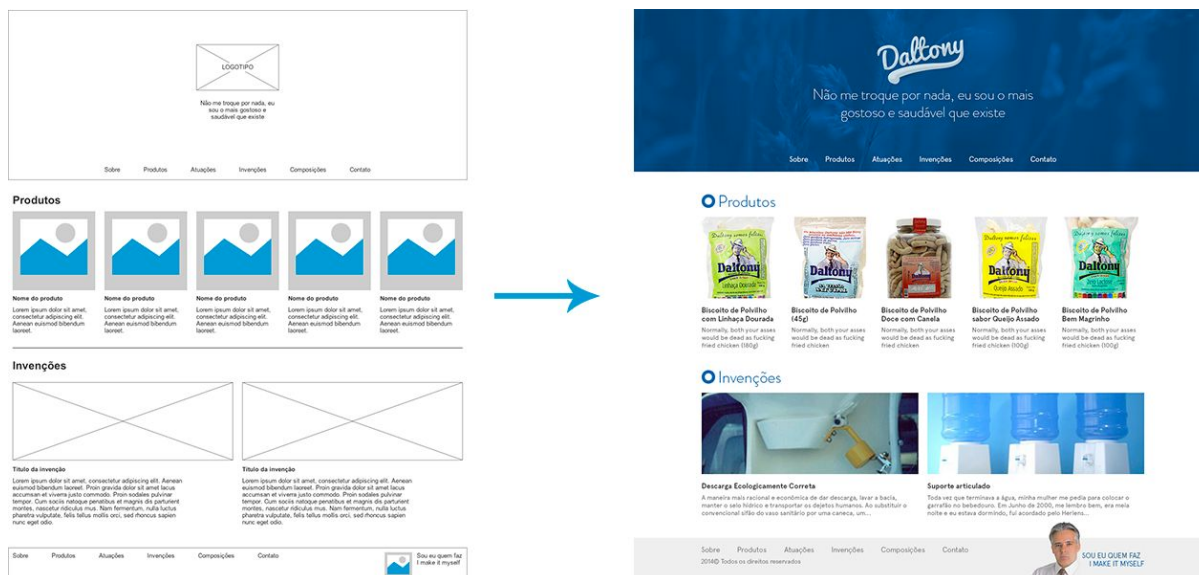
Fonte: Acervo pessoal

O modelo acima foi criado para um site institucional. Todas as interações do sistema, desde as páginas até o envio de e-mails de confirmação foram representadas com ícones que facilitam a compreensão à todos os envolvidos no processo de desenvolvimento.

Após o processo de criação dos elementos de interface e do mapa do site, começa o processo de design das páginas. Nessa fase, são elaborados os wireframes, desenhos básicos que demonstram de forma direta a arquitetura de como a interface global será. Dessa forma, é possível executar testes de interação com o usuário final, reposicionando a navegação do site, alterando a maneira como

a informação é disposta, etc. Com os wireframes prontos, o designer passa a criar, de fato, a interface aplicando seus conhecimentos em cores, tipografia e diagramação.

Figura 4: Wireframe e projeto gráfico completo



Fonte: Acervo pessoal

Acima, um exemplo do resultado obtido após a conversão do wireframe em interface finalizada. O caminho para o designer fica mais curto, pois a informação já vem organizada da forma mais eficiente possível.

Na sequência, inicia-se o desenvolvimento front-end. Na programação, existe uma classificação diferente para os desenvolvedores. O desenvolvedor front-end utiliza linguagens de marcação de texto e folhas de estilo para transformar as interfaces gráficas em sites estáticos. As interações básicas como a abertura de menus ocultos também são responsabilidade do front-end, que tem esse nome por escrever códigos que são executados pelo processador do usuário, e não do servidor.

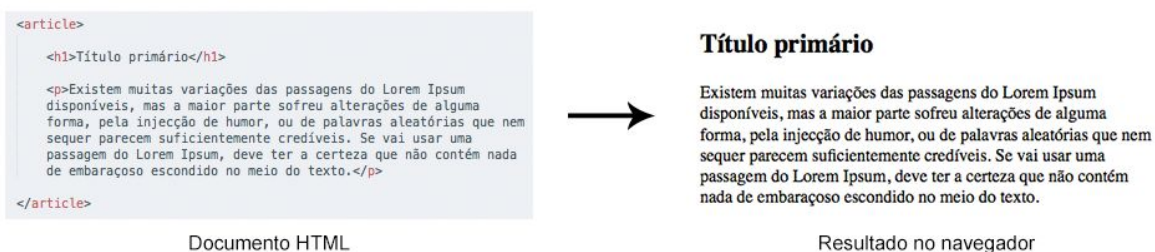
Já o desenvolvedor back-end escreve códigos que são executados pelo processador. São códigos que geralmente consultam um banco de dados, preenchem informações no HTML escrito pelo front-end e são exibidos ao usuário.

A HTML é uma linguagem de marcação de hipertextos. Através dela, o navegador é capaz de entender quais são as partes mais importantes, criando uma estrutura que mostra quais partes do documento são títulos, parágrafos, quais fazem parte de uma tabela e por aí vai.

Escrever um documento HTML é um processo parecido com criar um documento em um programa de edição de texto. No editor, para criar títulos, o usuário muda o tamanho da fonte e deixa o texto em negrito. No documento HTML, o desenvolvedor marca essa parte do texto como um título, fazendo com que o navegador consiga distinguir do resto do conteúdo.

O exemplo a seguir mostra o um trecho simples de um código HTML e o seu resultado renderizado pelo navegador. É possível observar que o conteúdo marcado com título de primeiro nível (h1) é destacado pelo navegador e o parágrafo (p) mantém a formatação padrão.

Figura 5: Exemplo de renderização pré-definida



Documento HTML

Resultado no navegador

Fonte: Acervo pessoal

Na maioria das vezes, apenas exibir o conteúdo na formatação padrão não é o suficiente. Geralmente, na execução de um projeto digital, é necessário aplicar estilos específicos de acordo com a interface do usuário criada pelo designer.

Para isso, o desenvolvedor usa o CSS, Cascading Style Sheet ou folha de estilos em cascata, em português.

Em cascata pois as declarações são lidas de cima para baixo e, caso duas declarações para o mesmo elemento sejam diferentes, vale a última presente no código.

Figura 6: Exemplo de aplicação básica de CSS



Fonte: Acervo pessoal

Com o auxílio do CSS, fica fácil transformar a renderização padrão do HTML em algo mais esteticamente agradável para o usuário. O CSS possibilita ao desenvolvedor mudar fontes, cores, formatos e posicionamento dos elementos. Aplicar estilos a um arquivo é relativamente fácil. Existem três formas, a primeira é adicionar os estilos diretamente no elemento, a segunda é adicionar os estilos dentro de um elemento *style* no topo do documento e o terceiro é criar um arquivo CSS e incluí-lo no documento HTML por meio de um *link*.

Figura 7: Exemplo de integração entre HTML e CSS



Fonte: Acervo pessoal

Com o CSS, bastam algumas declarações e o que era um arquivo simples se transforma em uma página diagramada, bem organizada e com estética agradável. No exemplo a seguir, alguns estilos básicos como fontes, cores e tamanhos foram alterados para tornar o formulário mais atrativo.

Figura 8: Exemplo de personalização de elementos básicos



Fonte: Acervo pessoal

Mesmo com toda a facilidade que o CSS oferece, são necessários alguns cuidados. Cada navegador tem a sua própria forma de interpretar e por isso, existem algumas diferenças. Para minimizar essas diferenças, alguns desenvolvedores optam por utilizar o SASS (*Sintatically Awesome Style Sheet* ou Folha de Estilos Sintaticamente Impressionante). O SASS também é muito utilizado por aumentar a produtividade pois economiza e compacta o código.

A sintaxe do SASS é diferente da do CSS. A diferença é que ao invés de colocar chaves para isolar as declarações para cada elemento, o SASS interpreta a identificação do arquivo. Sendo assim, chaves para isolar e ponto e vírgula para fechar cada declaração não são necessários.

Também é possível aninhar os elementos para economizar espaço e deixar o arquivo mais curto e organizado. Em pequenos sites, talvez não seja absolutamente

necessário, mas quando se trata de grandes sistemas com inúmeras páginas e muitos elementos diferentes a serem estilizados, o uso do SASS pode reduzir drasticamente a quantidade de linhas por arquivo.

Figura 9: Exemplo de compilação de arquivo SASS para CSS

```

1  .conteudo
2  .lista
3      margin: 0
4      padding: 0
5      list-style: none
6
7      li
8          display: inline-block
9
10         a
11             display: block
12             padding: 6px 12px
13             text-decoration: none

```

```

1  .conteudo .lista {
2      margin: 0;
3      padding: 0;
4      list-style: none;
5  }
6
7  .conteudo .lista li {
8      display: inline-block;
9  }
10
11 .conteudo .lista li a {
12     display: block;
13     padding: 6px 12px;
14     text-decoration: none;
15 }

```

Fonte: Acervo pessoal

No exemplo acima, uma pequena amostra do aninhamento e do resultado do arquivo após ser compilado. Mesmo com pouquíssimas declarações, o SASS ainda possibilitou a diminuição de uma linha no código. O SASS também possibilita ao desenvolvedor utilizar variáveis, facilitando as alterações e a reprodução do código e ainda criar algumas funções para reproduzir padrões chamados de *mixins*. Com eles, não é necessário escrever vários trechos de códigos parecidos, e sim utilizar variáveis para alterar os trechos que diferem de uma aplicação para outra.

Figura 10: Exemplo de compactação de código com o mixin

```

1  @mixin botao($cor,$tamanho){
2      border:1px solid $cor;
3      color:$cor;
4      @if $tamanho == pequeno{
5          padding:5px 10px;
6          font-size:16px;
7      }
8      @else{
9          padding: 10px 40px;
10         font-size:22px;
11     }
12 }

```

```

@include botao(#F00,pequeno);

```

Fonte: Acervo pessoal

Na figura 10, ao invés de escrever praticamente as mesmas propriedades para um botão diversas vezes ao longo do arquivo, foi criado um *mixin* onde o desenvolvedor pode escolher a cor predominante e o tamanho do botão.

A terceira e atual versão do CSS trouxe ainda diversos recursos para os desenvolvedores. Transições mais suaves, novos controles de estilo e até animações elevaram o nível da interação com o usuário. Para aproveitar ao máximo essas novidades, foram criadas algumas bibliotecas que, quando combinadas com o SASS, aumentam drasticamente a escalabilidade. Fica mais fácil para reproduzir e reutilizar códigos.

Uma dessas bibliotecas é o Compass. Ele auxilia na adequação do CSS para os vários navegadores. Algumas propriedades do CSS precisam ser escritas de forma diferente para cada navegador. O Compass agrupa essas formas diferentes de escrita em uma só chamada, como no exemplo a seguir.

Figura 11: Exemplo de aplicação de atributo CSS com SASS + Compass

```
@include background-clip(padding-box);
```

```
-moz-background-clip: padding;
-o-background-clip: padding-box;
-webkit-background-clip: padding;
background-clip: padding-box;
```

Fonte: Acervo pessoal

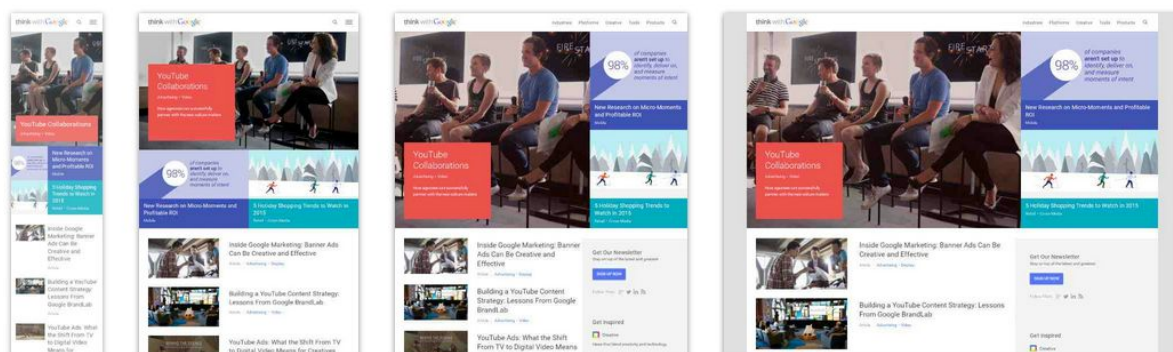
À esquerda, a chamada de uma propriedade CSS utilizando Compass e à direita, o CSS compilado. Nesse caso, o Compass reescreveu a propriedade CSS para que a mesma resultasse no mesmo estilo em todos os navegadores mais utilizados.

Após a parte gráfica do desenvolvimento, começa o desenvolvimento das interações com o sistema. Para abrir e fechar menus, mostrar informações ocultas ao passar o mouse sobre uma determinada área, o desenvolvedor utiliza o Javascript. De acordo com Duckett, o "JavaScript é escrito em texto puro, tal qual o HTML e o CSS, para que você não precise de nenhuma outra ferramenta para escrever um código.". Para facilitar o uso do Javascript, uma biblioteca chamada jQuery foi criada. Ela ajuda a manipular e animar o conteúdo HTML e ainda permite ao desenvolvedor utilizar *plugins*, que são bibliotecas com animações ou interações

prontas. Essas bibliotecas podem, geralmente, ser personalizadas de acordo com os requisitos de cada projeto.

Alguns conceitos que cabem ao desenvolvedor front-end são a responsividade e a acessibilidade presentes no sistema. Um site responsivo se adapta a todos os tamanhos de tela, permitindo ao usuário acessá-lo sem perder informações utilizando desde um monitor de 27 polegadas até um celular com uma tela de pouco mais de 3 polegadas.

Figura 12: Versões do mesmo site para tamanhos de tela diferentes



Fonte: <http://www.mediaqueri.es>

Na figura acima, a estrutura HTML é a mesma para todos os exemplos. O que faz com que o site seja visualizado de maneira diferente e se adeque a cada tamanho de tela são as *media queries*. Elas permitem que o desenvolvedor adicione propriedades específicas à um elemento em determinados tamanhos de tela.

Já a acessibilidade tem estado cada vez mais presentes em sistemas de qualidade elevada. São cuidados tomados pelo desenvolvedor para que uma pessoa que tenha alguma problema de visão, por exemplo, consiga navegar pelo sistema com o auxílio de leitores de tela.

Desenvolver um painel do zero pode ser muito trabalhoso. São muitos elementos que devem ser desenvolvidos de forma fácil e eficiente tanto na parte visual quanto na parte técnica. Por isso, muitas empresas optam por utilizar frameworks front-end que já possuem interfaces gráficas bem definidas e com uma implementação facilitada.

De acordo com Pagani, "*framework* é um conjunto de componentes que provêm uma estrutura básica de elementos reutilizáveis, tendo uma arquitetura consistente de funcionalidade genérica sob a qual a aplicação será construída". Os frameworks fornecem, geralmente, a formatação de elementos básicos utilizados pelos desenvolvedores que vão desde títulos e parágrafos até ícones.

Ainda segundo Pagani, "... os frameworks constituem bibliotecas que visam permitir a codificação do CSS de modo mais fácil e compatível com padrões de estilos, agregando uma série de opções já prontas para projetar uma página web, como se fosse um esqueleto, reduzindo o tempo gasto com o desenvolvimento."

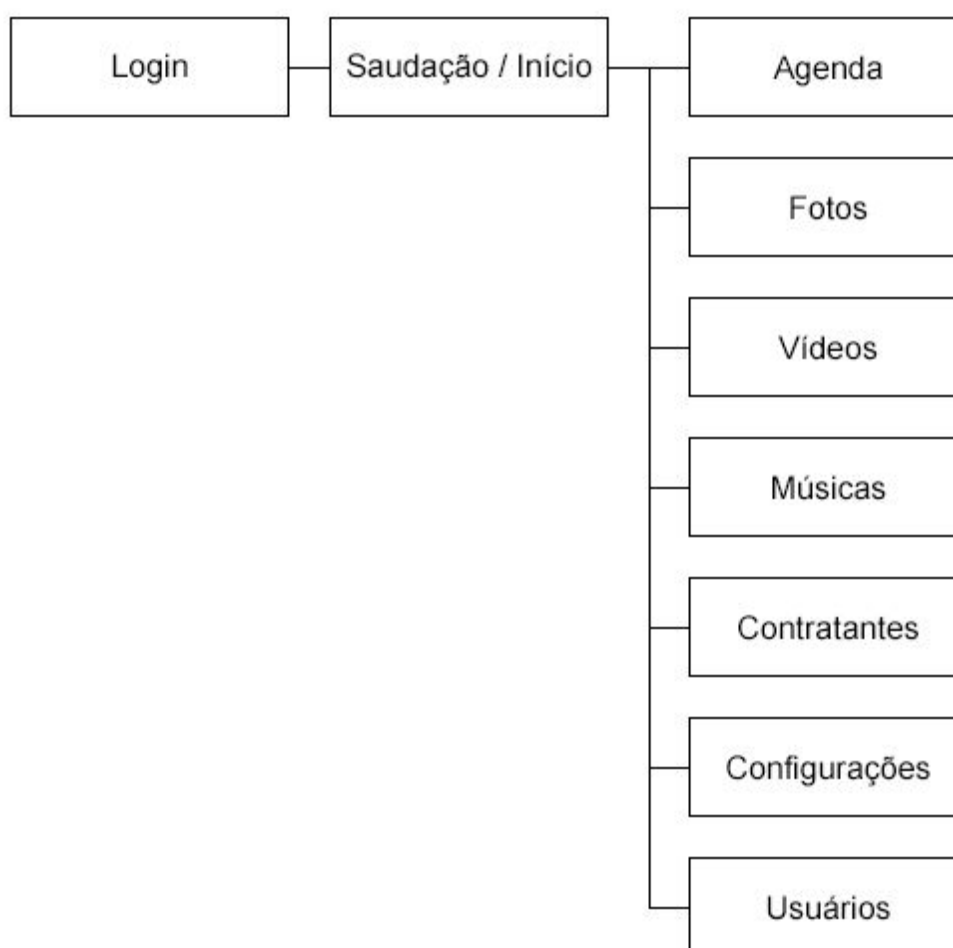
São muitos os frameworks populares entre os desenvolvedores. Entre eles, destacam-se o Bootstrap e o Foundation.

O Bootstrap foi criado por um designer e um desenvolvedor como um guia de estilos interno do Twitter. Com o crescimento interno e a abertura do projeto para o público geral, acabou se tornando um sucesso e foi muito difundido. Algumas agências cobram-no nos currículos de profissionais por sua facilidade de implementação e a qualidade do produto final, já que é responsivo e cobre todos os elementos do HTML, além de contar com algumas bibliotecas javascript próprias e até mesmo ícones. O concorrente à altura é o Foundation, da Zurb. Ele também possui padrões de design e bibliotecas javascript para turbinar o desenvolvimento.

3 PLANEJAMENTO

A estrutura anterior do Update não era bem definida. As áreas eram incluídas sobre demanda e não existia nenhuma arquitetura básica. Para identificar as páginas mais utilizadas, os projetos com áreas administrativas mais recentes foram acessados e a repetição de algumas áreas ficaram evidentes. São elas:

Figura 13: Estrutura padrão por relevância



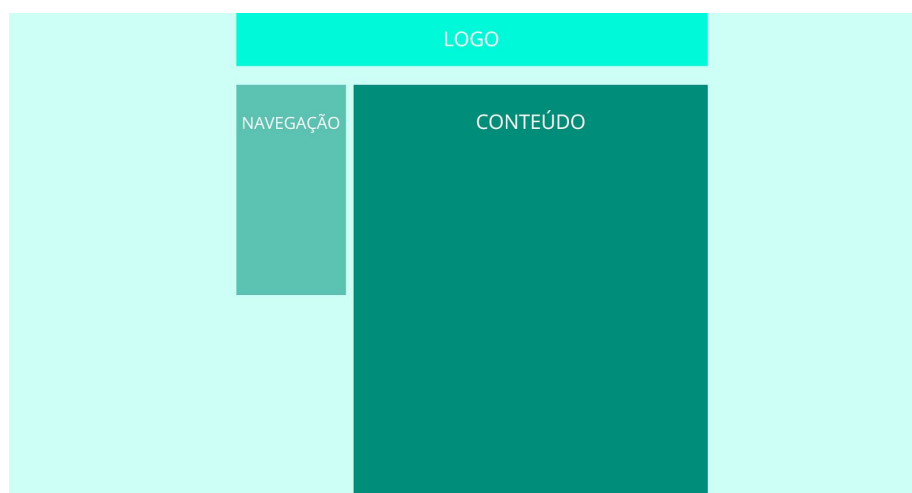
Fonte: Acervo pessoal

Outras páginas podem ser adicionadas de acordo com projetos específicos, mas essa estrutura contempla a maioria dos novos projetos iniciados pela agência.

Os novos campos de formulário também foram escolhidos de acordo com a análise dos últimos projetos. As informações mostradas na área pública variam de projeto a projeto e, para não precisar alterar a área administrativa todas as vezes, duas ações foram necessárias. A primeira foi instruir a equipe de criação da agência a tentar ao máximo padronizar as informações exibidas na área pública com as disponíveis na área administrativa. A segunda foi manter todos os campos recorrentes tornar obrigatórios apenas os que são básicos e necessários para cada área.

Com a nova estrutura em mãos, o objetivo era criar uma interface que fosse mais interessante no aspecto estético e que conseguisse se manter atual pelo máximo de tempo possível. Partindo daí, alguns estudos foram feitos para melhorar o uso dos espaços utilizados pelo admin. Na versão anterior, o conteúdo de cada página ocupava apenas 32% da área útil do navegador Firefox em um monitor de 22", como mostra a figura a seguir.

Figura 14: Divisão de áreas do sistema anterior



Fonte: Acervo pessoal

Na nova versão, esses 32% foram transformados em 53%. A solução foi ampliar a largura da página de 960 para 1240 pixels, retirar a navegação da lateral esquerda e estender o conteúdo por toda a área central resultando num foco no conteúdo muito maior e o acesso mais rápido à navegação, que foi posicionada ao lado do novo logotipo.

Figura 15: Nova divisão de áreas



Fonte: Acervo pessoal

Segundo Lidwell, "A regra do 80/20 afirma que aproximadamente 80% dos efeitos gerados em qualquer grande sistema são causados por 20% das variáveis daquele sistema". No caso do Update, 80% do conteúdo pode ser acessado por 20% da tela do usuário. O menu acessa quase todas as áreas do sistema.

Em paralelo, a nova identidade e um kit de interface do usuário estavam sendo criados. A nova identidade precisava ser simples e eficiente. Para isso, a nova marca da agência foi usada acompanhada do nome Update escrito em uma fonte espessa, forte, que tivesse poder de reconhecimento. A fonte escolhida foi a SansitaOne, fonte utilizada no site da agência que iria ao ar em breve. Assim, a unidade entre o site da agência e o novo painel administrativo estava garantida. A cor também não foi escolhida por acaso. O azul claro da marca é o mesmo da marca da agência, mais um ponto para a unidade.

Figura 16: Nova identidade visual do Update



Assim que a marca ficou pronta, os esforços se concentraram na nova interface do usuário. Algumas decisões precisavam ser tomadas no início, eram elas: a paleta de cores e a fonte padrão.

A paleta de cores foi criada a partir da identidade visual. O azul predominante ganhou variações mais claras, a cor complementar ao azul também foi adicionada com a suas variações de iluminação para alcançar um nível alto de contraste pois, segundo Lupton, "a decisão de utilizar cores análogas ou contrastantes afeta a energia visual e a atmosfera de qualquer composição.". Sendo assim, o sistema com altos contrastes ficaria mais moderno, passando a sensação de ser mais rápido e dinâmico que o anterior, que utilizava cores análogas.

Figura 17: Paleta de cores



Fonte: Acervo pessoal

A fonte escolhida foi a Open Sans, uma fonte sem serifa humanista criada pelo designer Steve Matteson que tem, de acordo com o próprio autor, uma aparência neutra, porém amigável. Ela foi escolhida por ser otimizada para o uso na web, de fácil leitura e distribuída gratuitamente para uso comercial. Para o projeto, apenas as versões light, normal e bold foram selecionadas por cobrirem todos os aspectos necessários.

Figura 18: Família tipográficaLight 300

Grumpy wizards make toxic brew for the evil Queen and Jack.

Normal 400

Grumpy wizards make toxic brew for the evil Queen and Jack.

Bold 700**Grumpy wizards make toxic brew for the evil Queen and Jack.**Fonte: <https://www.google.com/fonts/specimen/Open+Sans>

Com as cores e fontes definidas, era hora de criar o kit de interface do usuário que seria usado em todas as páginas. O anterior era confuso e fraco esteticamente, não tinha consistência nas formas e nem na tipografia e não passava a sensação de unidade.

4 DESIGN

Figura 19: Lâmina com os elementos do sistema anterior

The image shows a user interface for event management. At the top, there is a navigation bar with a left arrow, buttons for page numbers 1, 2, 3, 4, and 5, a right arrow, and a button labeled '355'. Below this is a blue header bar with the text 'Agenda » Eventos Futuros'. To the right of the header is a 'Novo' button with a green plus icon. The main form is titled 'Dados do Evento' and contains a text input field for 'Evento', a dropdown menu for 'UF' with 'AC' selected, and a checkbox labeled 'Ocultar Horário'. At the bottom of the form is a large orange button labeled 'OK'.

Fonte: Acervo pessoal

4.1 Elementos de formulário

Examinando cada elemento separadamente, mais problemas foram encontrados. Os campos de texto tinham o rótulo pequeno, não tinham espaçamento e as cores dos textos eram diferentes do restante do layout. Isso foi resolvido com a aplicação da nova fonte e das cores e com o aumento do respiro entre os campos.

Figura 20: Antigo e novo campo de texto

The image compares two versions of a text input field. On the left, labeled 'ANTES', the text 'Evento' is in a small, dark font above a text box containing 'Villa Mix Festival'. On the right, labeled 'DEPOIS', the text 'Evento' is in a larger, lighter font above a text box containing 'Villa Mix Festival'. The text box in the 'DEPOIS' version is also larger and has a more consistent color scheme with the rest of the interface.

Fonte: Acervo pessoal

Os campos de seleção de valores apresentavam os mesmos problemas dos campos de texto e também foram atualizados.

Figura 21: Antigo e novo campo de seleção



Fonte: Acervo pessoal

A falta de padrão entre o botão para adicionar um novo item e o botão para confirmar a ação criava uma certa confusão no usuário. Em alguns casos, o usuário não soube identificar onde ficava o botão "novo" pois o mesmo era diferente dos demais botões do sistema. Com a atualização, os botões foram padronizados e a diferenciação entre os dois ficou por conta da cor. Os botões para adicionar novos itens ficaram em azul e os botões de confirmação ficaram com a cor laranja.

Figura 22: Antigos e novos botões



Fonte: Acervo pessoal

4.2 Títulos

Os títulos foram alterados. No sistema anterior, os títulos de primeiro nível eram pequenos e o pouco destaque que tinham do restante do conteúdo era ofuscado pela escolha ruim da tipografia, que não tinha leitura e foi usada de forma invertida, com uma cor clara aplicada sobre outra cor, mais escura. Para solucionar esses problemas, os novos títulos ficaram maiores e com uma cor mais vibrante. Para aumentar a visibilidade, o tamanho da fonte aumentou drasticamente e o fundo passou a ser branco, como no restante do layout. Os títulos secundários também foram aprimorados.

Figura 23: Comparativo entre títulos novos e antigos



Fonte: Acervo pessoal

4.3 Paginação

A paginação passou a ser bem menos agressiva, com informações mais simples e menos páginas por exibição, já que a intenção seria aumentar o número de itens por página. As cores e fontes também foram adequadas a nova identidade.

Figura 24: Nova paginação



No geral, o novo UI kit ficou mais conciso, mais atrativo e mais coerente. Os elementos passaram a se relacionar esteticamente por conta das cores, formas e fontes aplicadas.

Figura 25: Elementos atualizados

Fonte: Acervo pessoal

4.4 Elementos gerais

Com o UI kit pronto, era a hora de desenhar a nova página padrão com topo e rodapé. A diagramação da página já havia sido definida anteriormente, então só faltava posicionar os itens de acordo com o modelo. O topo ficou com a identidade ao lado esquerdo, seguida pela navegação que mostrava as páginas principais e deixava as secundárias em um menu oculto. Do lado direito, ficou a área de identificação do usuário com o link para sair.

Figura 26: Novo topo e navegação



Fonte: Acervo pessoal

O rodapé ficou extremamente simples. Contava apenas com as informações de copyright e a assinatura da agência.

Figura 27: Novo rodapé

ICOMP Update © 2015

Agência ICOMP 

Fonte: Acervo pessoal

4.5 Exibição de dados

Após criar as áreas estáticas do sistema, o foco foi para a área de conteúdos. No painel anterior, todas as listagens eram feitas em formato de tabela. Em alguns casos, como na agenda, era de certa forma eficiente pois nesta área não havia nenhuma mídia, apenas textos. Já em outros casos onde os itens continham mídias, a listagem ficava pobre visualmente.

O primeiro caso a ser solucionado foi a página "agenda", por ser a mais utilizada pelos clientes. Os eventos eram listados em forma de tabela e tinham pouco apelo visual. A solução foi transformar os itens em caixas ordenadas pelo show mais próximo e destacar o dia do show, que é a informação mais relevante. No anterior, as datas tinham a mesma formatação das outras informações da lista.

Figura 28: Antes e depois da tabulação de dados

Data	Cidade/UF	
04/09/2015	Itu - SP	 
06/09/2015	Goiânia - GO	 
25/08/2015	Campo Grande - MS	 
27/08/2015	Barretos - SP	 
29/08/2015	São Paulo - SP	 
28/08/2015	São Paulo - SP	 
03/09/2015	Guapiaçu - SP	 
05/09/2015	Viradouro - SP	 



Fonte: Acervo pessoal

Essas caixas melhoraram a visualização das informações por deixarem em evidência as informações mais importantes de cada evento. Assim, a busca pelo

item ficou mais rápida e objetiva.

Figura 29: Visão geral da lista de informações



Fonte: Acervo pessoal

Esse mesmo estilo de lista foi aplicado a outras páginas que não continham mídias em sua listagem, com era o caso dos materiais para contratantes e dos usuários do sistema. Além desses outros dois casos, o estilo ficou a disposição para ser utilizado a quaisquer outras páginas necessárias.

Agora, o maior problema eram as páginas que continham mídias. A primeira página a ser estudada foi a de fotos. No sistema anterior, as galerias cadastradas eram tabuladas da mesma forma que os eventos na agenda. A solução escolhida foi criar uma lista com itens na horizontal com a foto de capa de cada galeria, assim a lista ficaria mais visual, já que se tratam de fotos.

Figura 30: Melhora na visualização das galerias



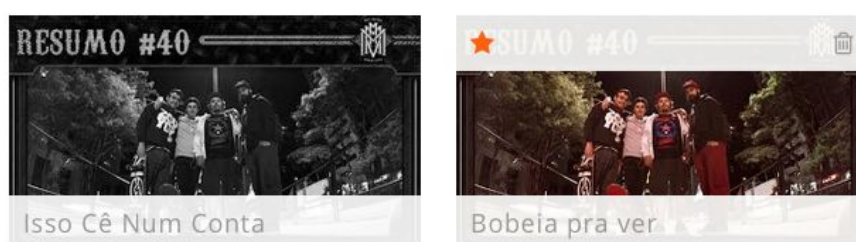
Fonte: Acervo pessoal

As listas ficaram definitivamente mais visuais com o novo layout. As capas ficaram em preto e branco e, assim que o usuário posicionasse o mouse sobre a galeria, as informações apareceriam. Sendo assim, o que passou a valer foi a capa da galeria, que é escolhida pelo próprio usuário, criando uma memória visual que facilita a pesquisa posterior.

As próximas duas páginas eram a de vídeos e a de músicas. Nenhuma delas existia até então e era insistentemente cobrada tanto pelos desenvolvedores quanto pelos clientes da agência. Até então, os vídeos eram retirados do canal do cliente no YouTube e as músicas eram atualizadas manualmente pelos desenvolvedores.

Pensando na página de vídeos, uma das reclamações mais recorrentes dos clientes era que, às vezes, precisavam colocar um vídeo em destaque por um determinado tempo, mas não existia meio de fazer isso sem gerar uma manutenção para o desenvolvedor. Outra reclamação era a de que os clientes gostariam de colocar vídeos que não haviam sido enviados ao YouTube por eles, mas por apresentadores de TV ou fãs, por exemplo. A solução foi criar uma página no gerenciador onde o usuário pudesse cadastrar um título, um link do YouTube e marcar como destaque ou não. Assim, teria total controle sobre os vídeos.

Figura 31: Itens da lista de vídeos com ou sem destaque



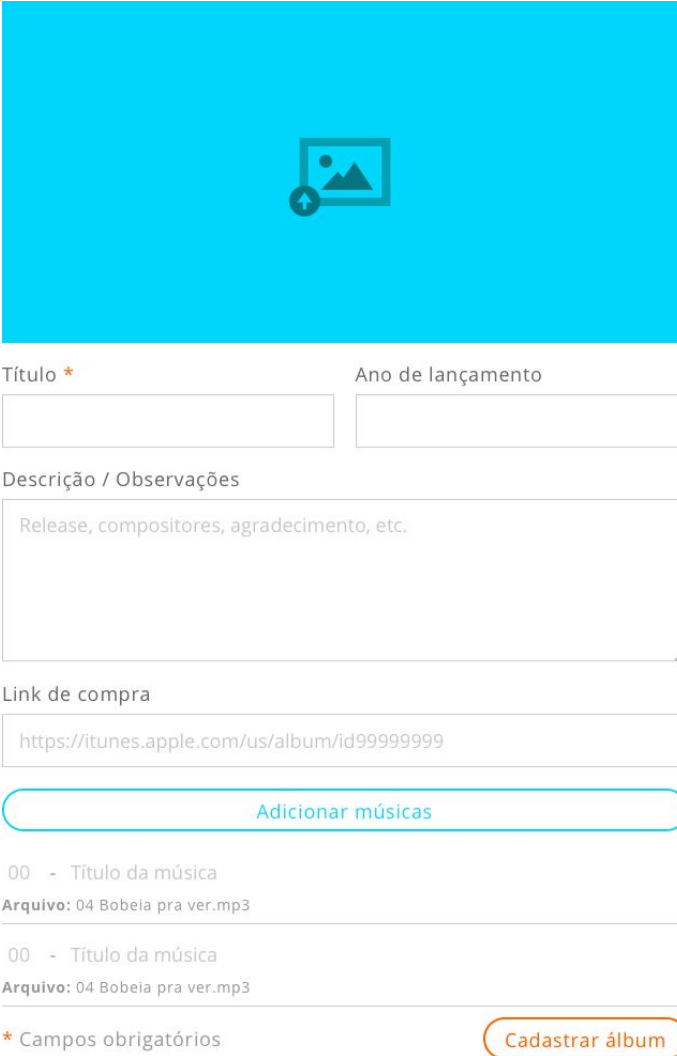
Fonte: Acervo pessoal

Na listagem, o vídeo em destaque seria identificado por uma estrela e os demais poderiam ser selecionados ali mesmo, apenas com um clique. Dessa forma, o usuário não precisaria sair da tela para salvar outro item como destaque.

Já a página de álbuns nunca havia sido se quer considerada para fazer parte do sistema por ter uma estrutura complexa. Para facilitar a inclusão de álbuns, a página de cadastro foi a que concentrou mais esforços no planejamento. Era

necessário criar um formulário que contemplasse os campos básicos de um álbum e que permitisse ao usuário enviar os arquivos de áudio que seriam utilizados quando o site do artista tivesse player de músicas, como era de costume.

Figura 32: Visão geral do cadastro de galerias de fotos



Título *

Ano de lançamento

Descrição / Observações

Release, compositores, agradecimento, etc.

Link de compra

https://itunes.apple.com/us/album/id999999999

Adicionar músicas

00 - Título da música

Arquivo: 04 Bobeia pra ver.mp3

00 - Título da música

Arquivo: 04 Bobeia pra ver.mp3

* Campos obrigatórios

Cadastrar álbum

Fonte: Acervo pessoal

Os campos obrigatórios eram apenas a capa e o título. Ainda haviam os campos de ano de lançamento, descrição, link de compra e o botão para adicionar as músicas. Com esse problema resolvido, a listagem ficou como a lista de galerias com a capa do álbum em destaque e as informações adicionais que apareciam com o mouse sobre o item.

A página de configurações havia sido recém construída para atender um requisito interno: a configuração de e-mail e SMTP. Toda vez que era solicitada,

essa configuração tinha que ser feita na mão pelos desenvolvedores. Então, foi criada a página de configurações para que a alteração pudesse ser feita no próprio sistema. Como a página havia sido criada a pouco tempo o visual não era tão ruim, mas como todo o sistema passava por uma readequação, foi atualizada juntamente com as demais. Aproveitando a atualização, foram adicionados campos para configurar as informações de SEO dos sites.

Algumas telas mais simples como a de login e a de saudação foram criadas em paralelo às demais. A página de login ficou extremamente simples, com os campos de usuário e senha e o botão para acessar.

Figura 33: Visão geral da tela de login

Fonte: Acervo pessoal

Na página de saudação foi mantido um texto parecido com o da página de login, porém com um adicional informando a função do admin. Esses textos já eram utilizados na versão anterior do administrador.

Figura 34: Texto de recepção

BEM-VINDO AO UPDATE

Com ele, o cliente tem total acesso aos módulos para gerenciar os conteúdo de seu site. Todas as áreas dinâmicas contratadas estão facilmente disponíveis para atualização pelo cliente.

Em caso de dificuldades para utilizar o sistema, entre em contato pelo e-mail suporte@icomp.com.br ou 43 3342-8331

Fonte: Acervo pessoal

Após finalizar as etapas relacionadas ao design, todos os esforços foram concentrados no desenvolvimento front-end.

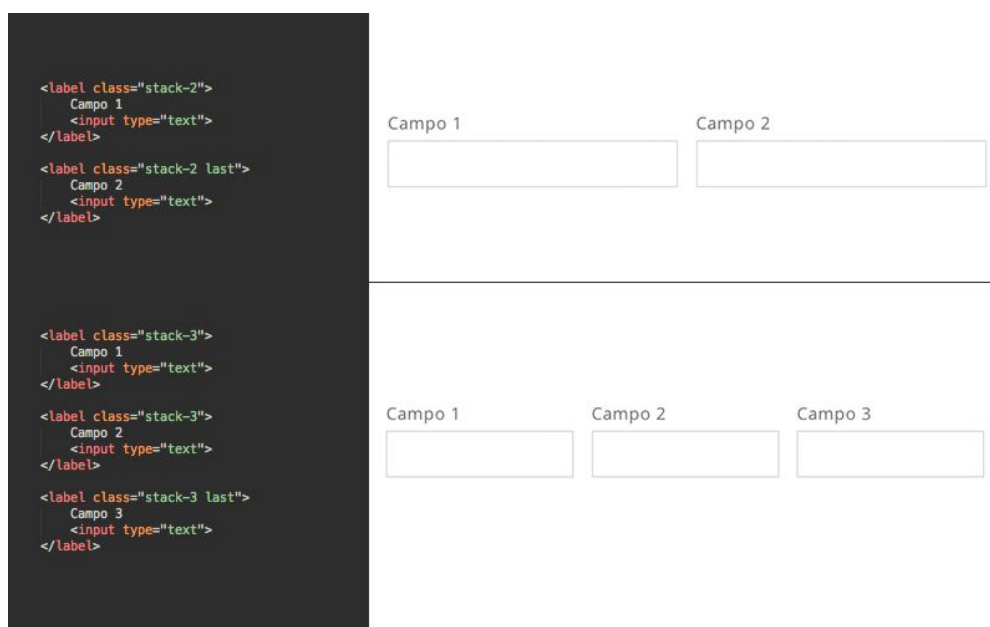
5 DESENVOLVIMENTO FRONT-END

Como o layout do painel foi todo criado em conjunto com o desenvolvedor responsável pelo projeto, a estrutura HTML foi facilmente criada. Para aumentar ainda mais o desempenho, duas páginas que continham a maior parte dos elementos do kit de interface foram desenvolvidas primeiro. Após apenas algumas horas de trabalho, o desenvolvedor já tinha estilizado a maioria dos elementos que usaria no decorrer do desenvolvimento.

Como modelo, foram usadas a página de agenda e a de adicionar novo evento. O layout de agenda foi escolhido pois sua estrutura seria reaproveitada em outras duas telas do sistema e o novo evento era a base de todos os formulários por conter os elementos mais comuns: campo de texto, de seleção, área de texto e botão.

As áreas que seriam padrão em todas as páginas (topo e rodapé) já tiveram suas versões responsivas todas desenvolvidas. Para os formulários, algumas classes especiais também foram criadas para agrupar dois ou três campos de qualquer tipo na mesma linha.

Figura 35: Estrutura de classes para organização visual dos elementos



Fonte: Acervo pessoal

Os botões também são estilizados por classes globais reaproveitáveis que podem ser utilizadas tanto em âncoras quanto em botões e inputs de envio. Uma classe formata o botão e outras duas definem as cores.

Figura 36: Estrutura de classes para botões



Fonte: Acervo pessoal

A próxima página a ser desenvolvida era a das galerias de fotos. A listagem foi simples. Eram quatro galerias por linha com informações que ficavam ocultas até que o usuário posicionasse o mouse sobre a galeria específica. Essa interação do mouse foi feita com CSS 3 alterando a opacidade e a posição do div com as informações.

Figura 37: Representação da interação do usuário com o item da lista



Fonte: Acervo pessoal

A página de cadastro de galerias tinha uma peculiaridade, era necessário criar um botão para subir fotos ou arquivos compactados. No antigo sistema, esse botão fazia parte de um plugin e era renderizado em flash, o que impossibilitava o uso em determinados aparelhos celulares. Para que o novo sistema fosse 100% responsivo e acessível, o novo botão foi desenvolvido como outro botão qualquer e a equipe de back-end desenvolveu uma forma de upload simplificada e rápida sem usar nenhum plugin.

A página de vídeos seguiu os mesmos padrões da página de fotos. O cadastro de vídeos também foi simples pois necessitava de apenas dois campos de texto e um campo para marcar ou não como destaque. A página de músicas também seguiu os padrões da página de galerias com quatro capas por linha. Talvez a página de cadastro mais trabalhosa de todas tenha sido a de cadastro de álbuns.

Nela, o usuário podia subir uma capa para o álbum, preencher o título, ano de lançamento, descrição e um link para compra na loja de preferência e, até aí, tudo normal. O trabalhoso é que o cliente poderia subir as músicas daquele álbum e renomear uma a uma. A solução encontrada foi criar campos para que o usuário preenchesse o número e o título, música a música. Ao invés de apenas listar os nomes das músicas, foram listados campos de texto descaracterizados contendo as informações para que o usuário pudesse alterá-las e só depois concluir o cadastro.

Figura 38: Personalização de títulos de músicas



Fonte: Acervo pessoal

A página de configurações foi facilmente resolvida pois as classes globais para formulários fizeram que quase toda a página estivesse pronta apenas escrevendo o HTML. Esse era um dos objetivos do novo painel, facilitar o desenvolvimento front-end para que o foco principal ficasse nas funcionalidades, e

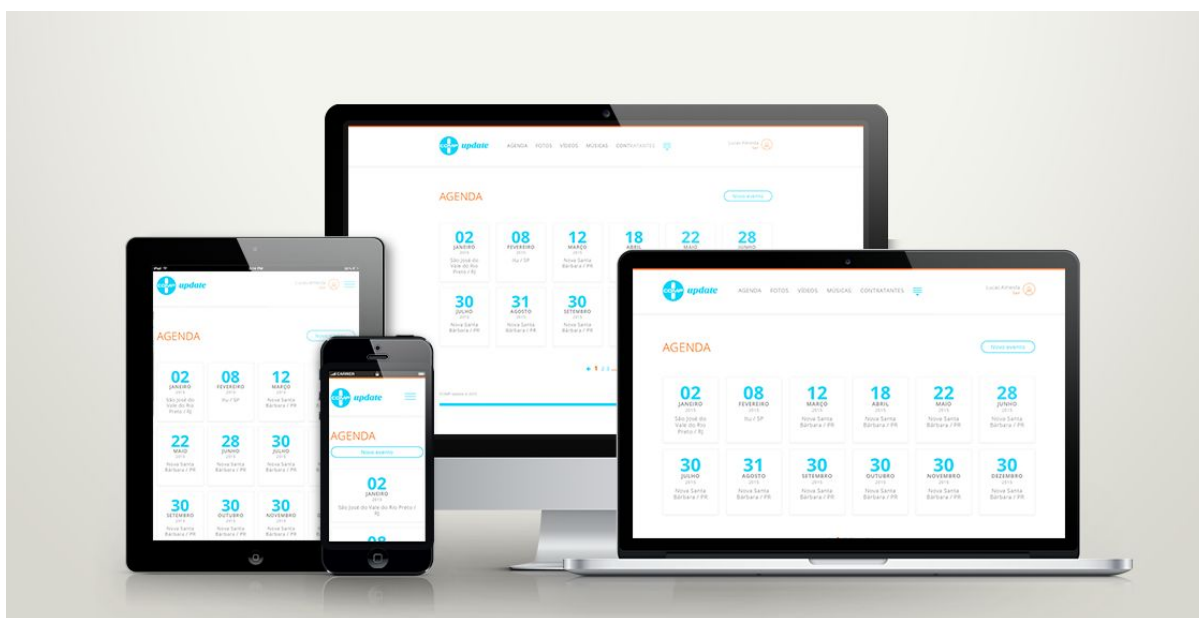
não na montagem das páginas.

Por fim, a página de usuários usava praticamente a mesma formatação das páginas com dados tabulares. Alguns ajustes foram feitos nas informações que eram exibidas, mas nada que tomasse muito tempo. Depois que todas as páginas da nova estrutura padrão estavam desenvolvidas, o foco era transformar tudo aquilo em algo que pudesse ser utilizado em celulares.

As media queries para ajustar os elementos padrão já haviam sido escritas no início do desenvolvimento front-end, junto com o término das duas páginas piloto. O menu, até um determinado tamanho de tela ficou oculto e podia ser acessado por meio de um ícone no canto superior direito da tela.

O comportamento padrão do sistema quando exibido em celulares era de mostrar os itens de lista um abaixo do outro, para facilitar a visualização e aumentar a área de acesso com o toque.

Figura 39: Visão geral do sistema em diferentes tamanhos de tela



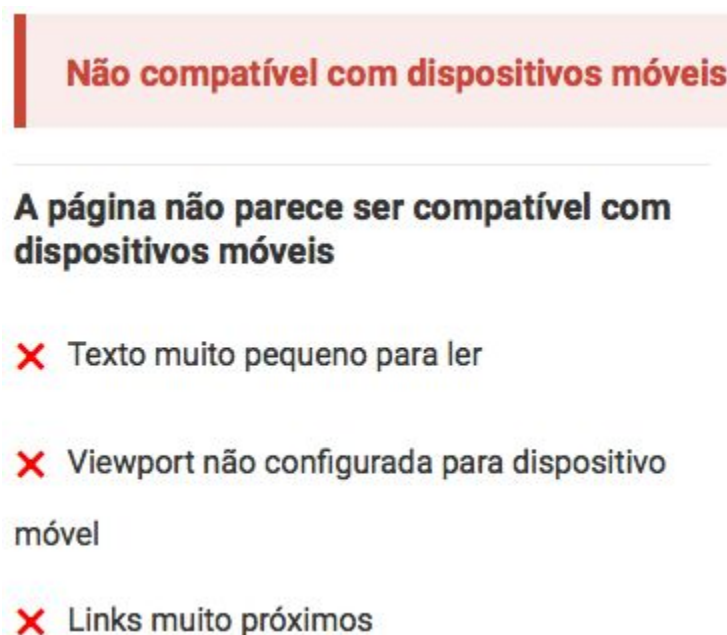
Fonte: Acervo pessoal

6 TESTES

Os primeiros testes foram feitos na parte responsiva do sistema. Durante o desenvolvimento foram utilizados um Apple iPad 2 representando os tablets e um Apple iPhone 5C representando os smartphones. Dessa forma, os ajustes na finalização do projeto não seriam tão intensos. Para testar a resposividade, o Google disponibiliza uma ferramenta para testes de compatibilidade com dispositivos móveis.

Nas páginas do novo painel, os testes retornaram "Incrível! Esta página é compatível com dispositivos móveis.". Para base de comparação, as páginas do painel antigo também foram testadas e além do retorno negativo, a ferramenta apontou os problemas, que podem ser vistos na figura 40.

Figura 40: Exemplo de resposta negativa para responsividade



Fonte: Acervo pessoal

Após a responsividade ser comprovada, a compatibilidade entre navegadores foi o próximo tópico. A ferramenta utilizada para o teste, *Browser Shots*, captura a interface em diversos navegadores e sistemas operacionais diferentes. Dessa forma, é possível fazer a análise das incompatibilidades de interface. As páginas foram testadas em 136 navegadores diferentes (levando em consideração as versões

diferentes). Nenhuma divergência visual foi detectada nas versões mais atuais dos navegadores e, por isso, não foram necessárias alterações.

Todas as páginas foram avaliadas e receberam nota 100 de 100 levando em consideração a usabilidade em cada tipo de aparelho utilizando uma ferramenta disponibilizada pelo *Google Developers* chamada *Pagespeed Insights*, que também leva em consideração a velocidade de carregamento da página. A figura 41 é da análise da página de login.

Figura 41: Resposta do Pagespeed da página de login

100 / 100 Experiência do usuário

✓ **Parabéns! Nenhum problema encontrado.**

Configure o visor
Sua página especifica um visor que corresponde ao tamanho do dispositivo. Isso permite que ela seja exibida apropriadamente em todos os dispositivos. Saiba mais sobre a [configuração de visores](#).

Dimensionar o conteúdo para se ajustar à janela de visualização
Os conteúdos da sua página se ajustam à janela de visualização. Saiba mais sobre [como dimensionar o conteúdo para se ajustar à janela de visualização](#).

Evitar plug-ins
Parece que sua página não usa plug-ins, o que previne o uso de conteúdos em muitas plataformas. Saiba mais sobre a importância de [evitar plug-ins](#).

Evite intersticiais para a instalação de aplicativos que ocultam conteúdo
Parece que sua página não tem intersticiais para a instalação de aplicativos que ocultam uma parte significativa do conteúdo. Saiba mais sobre a importância de [evitar o uso de intersticiais para a instalação de aplicativos](#).

Tamanho apropriado dos pontos de toque
Todos os links/botões da sua página são grandes o suficiente para que um usuário toque com facilidade no touchscreen. Saiba mais sobre o [tamanho apropriado dos pontos de toque](#).

Use tamanhos de fonte legíveis
O texto na sua página é legível. Saiba mais sobre o [uso de tamanhos de fonte legíveis](#).



Fonte: Acervo pessoal

O teste mais importante veio na implementação do sistema em um projeto da agência. Os primeiros problemas encontrados foram a necessidade de desenvolver páginas específicas (que não estão na estrutura padrão) e algumas funções extras, como tornar um item ativo ou inativo.

O primeiro problema solucionado foi a criação das páginas extras, específicas para cada cliente. Como o sistema foi criado com um kit de interface do usuário consistente, não foi muito trabalhoso criar as novas telas. Por exemplo, a estrutura da página Contratantes, por ser simples, foi utilizada como base para outras telas. O segundo problema foi resolvido pelo próprio programador back-end no decorrer do projeto. As novas funcionalidades eram analisadas pelo front-end, ajustadas e

aplicadas uma a uma.

Como esse tipo de sistema tem que ser atualizado com uma frequência muito grande para não se tornar obsoleto, vários testes são realizados semanalmente, a cada aplicação. Todas essas alterações são versionadas e adicionadas à versão padrão, salvo algumas situações específicas.

7 CONSIDERAÇÕES FINAIS

Com a finalização da etapa crucial do desenvolvimento, a sensação foi de dever cumprido. Todos os objetivos esperados foram alcançados e o novo sistema foi completamente aceito pelos usuários e desenvolvedores. O Update foi reconstruído do zero, mas manteve a navegação facilitada e a simplicidade no uso.

A nova interface ficou moderna e foi criada para se manter atual pelo maior tempo possível. Os títulos e textos foram ampliados, os elementos foram bem demarcados e padronizados, deixando a interface amigável e limpa, sem informações desnecessárias ou imagens que comprometeriam o rendimento.

Na parte técnica, um dos pontos positivos é a diminuição da quantidade de arquivos no sistema. No anterior, passavam de mil. Hoje, são menos de cem arquivos contando com as bibliotecas javascript de terceiros e com os arquivos fonte do back-end. Essa diminuição economiza tempo na hora de subir os arquivos e também evita o uso desnecessário de espaço do servidor.

Essa diminuição se deve a nova estrutura que foi criada para ser mais simples e à diminuição do uso de imagens na parte gráfica do sistema. Também pelo fato de as bibliotecas javascript terem sido atualizadas, removendo arquivos necessários de versões antigas das mesmas.

A aplicação do sistema passou a ser feita pela equipe de back-end da agência. O Update anterior era implementado pelos programadores front-end, custando horas de trabalho em algo antiquado. Todas as alterações em listagens, paginação, novas telas específicas eram feitas pelo front-end usando como base outras versões de áreas do admin. Depois da atualização, o back-end só precisa alterar as conexões com o banco de dados e, se existir uma necessidade específica, replicar a página mais adequada. Não é mais necessário duplicar a página, alterar as chamadas de javascript, duplicar o javascript relacionado, alterar chamadas do servidor, etc.

Outro problema resolvido foi o das páginas que eram geradas a partir de arquivos javascript. O novo sistema só utiliza AJAX nas ações que não precisam ser

redirecionadas a novas páginas, como ativação e exclusão de itens. A listagem e paginação são todas feitas em linguagem back-end.

Um ponto que teve muita aceitação dos clientes foi que a nova interface foi feita para ser utilizada a partir de qualquer aparelho que possuíse um navegador e internet. O sistema é inteiramente responsável e se comporta bem em qualquer tamanho de tela, fazendo com que ele seja útil a qualquer momento.

Por fim, a atualização do sistema foi concluída com sucesso pois agradou a todos os envolvidos. O cliente final por estar utilizando um produto moderno e atualizado com frequência, a equipe de front-end por não precisar mais gastar horas para alterar vários arquivos a cada implementação e a equipe de back-end por poder implementar o sistema de forma rápida e precisa.

REFERÊNCIAS

PAGANI, Talita. **Quando utilizar (ou não) frameworks CSS**. Disponível em <<http://tableless.com.br/quando-utilizar-ou-nao-frameworks-css/>>. Acesso em 12 jul. 2013.

NIELSEN, J.; LORANGER, H. **Usabilidade na Web**: projetando websites com qualidade. Tradução por New Riders - Peachpit. São Paulo: Elsevier, 2006.

SIQUEIRA, Marcelo C. **Gestão estratégica da informação**: como transformar o conteúdo informacional em conhecimento valioso. Rio de Janeiro: Brasport, 2005.

LUPTON, E.; PHILLIPS, Jennifer C. **Novos Fundamentos do Design**. Tradução de Cristian Borges. São Paulo: Cosac Naify, 2008.

LIDWELL, W.; HOLDEN, K.; BUTLER, J. **Princípios Universais do Design**. Tradução de Francisco Araújo da Costa. Porto Alegre: Bookman, 2010.

DUCKETT, J. **Javascript & jQuery**: insteractive front-end web development. Tradução livre. Indianapolis: John Wiley & Sons, Inc, 2014.