



**SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO
SERTÃO PERNAMBUCANO - *CAMPUS FLORESTA***

GESTÃO DA TECNOLOGIA DA INFORMAÇÃO

FÁBIO LUIZ DA SILVA SÁ

**ESTUDO DE *WEB DESIGN* RESPONSIVO APLICADO A UM
SISTEMA DE COMPARTILHAMENTO DE ARQUIVOS.**

Floresta – PE

2015

FÁBIO LUIZ DA SILVA SÁ

**ESTUDO DE *WEB DESIGN* RESPONSIVO APLICADO A UM
SISTEMA DE COMPARTILHAMENTO DE ARQUIVOS.**

Monografia apresentada como requisito para obtenção de graduação em Gestão da Tecnologia da Informação, do Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano – *Campus Floresta*.

Orientador(a): Prof. Msc. Felipe Pinheiro
Correia

Floresta – PE

2015

S1e Sá, Fábio Luiz da Silva
Estudo de Web Design Responsivo Aplicado a um Sistema de
Compartilhamento de Arquivos. / Fábio Luiz da Silva Sá – 2015.

49f.: II.

TCC (Gestão de Tecnologia da Informação) – Instituto Federal de Educação,
Ciência e Tecnologia do Sertão Pernambucano – Campus Floresta-PE, 2015.

Orientação – Professor Msc. Felipe Pinheiro Correia

1. Dispositivos móveis . 2. Web design responsivo . 3. Internet. I. Título

CDD 006.74

FÁBIO LUIZ DA SILVA SÁ

ESTUDO DE *WEB DESIGN* RESPONSIVO APLICADO A UM
SISTEMA DE COMPARTILHAMENTO DE ARQUIVOS.

Aprovado em: 29/07/15

Nota: 9,43

BANCA EXAMINADORA

Felipe Pinheiro Correia

Professor MSc. Felipe Pinheiro Correia

IF-SERTÃO – PE

Cassiano Henrique de Albuquerque

Professor MSc. Cassiano Henrique de Albuquerque

IF-SERTÃO – PE

Eziom Alves de Oliveira

Professor Eziom Alves de Oliveira

IF-SERTÃO - PE

"Tente uma, duas, três vezes e se possível tente a quarta, a quinta e quantas vezes for necessário. Só não desista nas primeiras tentativas, a persistência é amiga da conquista. Se você quer chegar a onde a maioria não chega, faça o que a maioria não faz."

Bill Gates

AGRADECIMENTOS

Agradeço a Deus criador do céu e da terra, por toda saúde, força e coragem que me deu durante esta longa caminhada, e por ser essencial em minha vida.

Aos meus familiares em especial a minha mãe, Maria de Lourdes de Sá (*in memoriam*), pelo exemplo de mulher e dedicação a família e que se faz presente em todos os dias da minha vida.

Ao meu pai Luiz Manoel de Sá, homem simples, de coragem, trabalhador e que com certeza é um exemplo a ser seguido.

Aos meus irmãos por todo apoio dado nesse e em outros momentos marcantes dessa grande jornada da vida.

Ao meu orientador Professor Felipe Pinheiro Correia pela grande contribuição e sabedoria, paciência e também por acreditar em mim.

A todos os professores pelo convívio, pelas trocas de conhecimento e experiência que foram de grande importância para o amadurecimento da vida acadêmica.

Ao meu amigo Phablo Henrique de Melo França, pela parceria e por compartilhar comigo a ideia inicial do SISCA (Sistema de Compartilhamento de Arquivos).

Aos amigos e colegas do Curso de Gestão da Tecnologia da Informação pela convivência nesses longos e proveitosos anos de aprendizado.

Muito obrigado a todos!

RESUMO

O número de usuários que acessam a internet por meio de dispositivos móveis cresce a cada ano, entre 2011 a 2013 foi registrado um crescimento de 106%. Vende-se mais *smartphones* e *tablets* que computadores. Houve mudança de paradigma onde o acesso à internet não é mais exclusivo a computadores *desktop* e *notebooks*, e *sim*, a novos meios de acesso e que requerem novos hábitos. A utilização de novos conceitos tecnológicos torna possível unificar a experiência de navegação em *sites* ou aplicações *web* independente do dispositivo que o acessa. O presente trabalho tem como objetivo abordar as técnicas de *Web Design* Responsivo e aplica-las na transformação de um protótipo de *layout* fixo para responsivo. Como resultado se obteve uma aplicação *web*, capaz de se adaptar de modo usável e acessível a diversas resoluções de tela.

Palavras-chaves: Dispositivos móveis. *Web Design* Responsivo. Internet.

ABSTRACT

The number of users accessing the Internet via mobile devices grows every year, between 2011 to 2013 was registered a growth of 106%. For sale more smartphones and tablet that computers. There has been a paradigm shift where Internet access is no longer exclusive to desktop computers and notebooks, and yes the, new means of access that require new habits. The use of new technological concepts makes it possible to unify the browsing experience on web sites or web applications regardless of the device that accesses. The present work aims to address the Responsive Web Design techniques and apply, in the transformation of a fixed-layout prototype for responsive. As a result we got an web application, able to adapt in order usable and accessible to different screen resolutions.

Keywords: Mobile Devices . Responsive Web Design . Internet.

LISTA DE FIGURAS

Figura 1 - Apresentação do SISCA.....	13
Figura 2 - <i>Layout</i> responsivo.....	15
Figura 3 - Exemplo de um <i>layout</i> responsivo.....	16
Figura 4 - Estrutura de uma regra CSS.....	18
Figura 5 - Exemplo folha de estilo CSS.....	18
Figura 6 - Utilizando a propriedade <i>max-width</i> para limitar <i>layout</i>	20
Figura 7 - Exemplo de medidas flexíveis.....	21
Figura 8 - Exemplo de proporção dos elementos.....	22
Figura 9 - Aplicando flexibilidade em imagens e outras mídias.....	25
Figura 10 - <i>Layout</i> fluido sem <i>Media Queries</i>	26
Figura 11 - Exemplo de <i>Media Queries</i>	27
Figura 12 - <i>Layout</i> fluido com uso de <i>Media Queries</i>	27
Figura 13 - Exemplo de <i>breakpoint</i> para dispositivo móvel.....	28
Figura 14 - Exemplo de <i>breakpoint</i> com <i>media query</i>	28
Figura 15 - Página de apresentação do sistema SISCA	30
Figura 16 - Apresentação do sistema SISCA em dispositivo móvel.....	31
Figura 17 - Folha de estilo CSS do <i>site</i> não responsivo.....	32
Figura 18 - Código HTML do <i>site</i> não responsivo.....	33
Figura 19 - <i>Wireframe</i> protótipo SISCA	34
Figura 20 - Código HTML do <i>site</i> responsivo	35
Figura 21 - Código CSS do <i>site</i> responsivo	36
Figura 22 - Código CSS com <i>breakpoint</i> para <i>mobile</i>	36
Figura 23 - Código CSS com <i>breakpoint</i> para <i>smartphone</i>	37
Figura 24 - Código CSS com <i>breakpoint</i> dispositivos móveis.....	37
Figura 25 - Menu <i>mobile</i> <i>site</i> responsivo.....	38
Figura 26 - <i>Site</i> responsivo renderizado em computador <i>desktop</i>	39
Figura 27 - <i>Site</i> responsivo renderizado em dispositivos móveis	40
Figura 28 - <i>Site</i> responsivo renderizado em <i>Smartphone</i>	41
Figura 29 - <i>Site</i> responsivo renderizado em <i>Smartphone</i>	42
Figura 30 - <i>Site</i> responsivo renderizado em <i>SmartTV</i>	43
Figura 31 - <i>Site</i> responsivo renderizado em diversos dispositivos.....	44

LISTA DE SIGLAS E ABREVIATURAS

CSS	Cascading Style Sheets
HTML	HyperText Markup Language
IDC	Internet Data Center
JPEG	Joint Photographic Experts Group
RESS	Responsive Web Design + Server Side Components
RWD	Responsive Web Design
SISCA	Sistema de Compartilhamento de Arquivos
TCC	Trabalho de Conclusão de Curso
URL	Uniform Resource Locator
W3C	World Wide Web Consortium

SUMÁRIO

1 INTRODUÇÃO	10
1.1 Justificativa	11
1.2 Objetivos	12
1.2.1 <i>Objetivo geral</i>	12
1.2.2 <i>Objetivo específicos</i>	12
1.3 Protótipo SISCA	13
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 Web Design Responsivo	14
2.2 Mobilidade	17
2.3 Web Design Responsivo e CSS3	17
2.3.1 <i>Tipos de medidas CSS</i>	19
2.3.2 <i>Limitando layout fluido com a propriedade CSS max-width</i>	20
2.4 Pilares de Web Design Responsivo	20
2.4.1 <i>Layout fluido ou Grid Flexível</i>	20
2.4.2 <i>Imagens e Recursos flexíveis</i>	23
2.4.3 <i>Media queries</i>	25
3 METODOLOGIA	29
4 CENÁRIO DO PROTÓTIPO SISCA	30
4.1 Adaptando o SISCA utilizando os conceitos de RWD	33
4.1.1 <i>Projeto de interface de design responsivo</i>	33
4.1.2 <i>Transformação do Protótipo SISCA em responsivo</i>	35
5 CONSIDERAÇÕES FINAIS	45
5.1 Trabalhos futuros	46
REFERENCIAS	47

1. INTRODUÇÃO

Com o crescimento de acesso aos dispositivos (*tablets*, *smartphones*) e da internet móvel entre os brasileiros, problemas como a navegação limitada, a incompatibilidade da grande maioria dos sites em ser visualizado nos diversos tipos de resoluções de tela e configurações dos dispositivos, tornam-se barreiras significativas para os usuários, tornando a navegação desgastante e limitada. A diversidade crescente de novos dispositivos torna cada vez mais difícil desenvolver *interfaces web* adequadas para contemplar os diversos perfis. Este trabalho aborda técnicas que permitem desenvolver interfaces que se adequem a diferentes tamanhos de telas de dispositivos eletrônicos. As soluções para isso são descritas nos conceitos de *Web Design Responsivo* que sugerem que a aplicação seja capaz de “responder” às características do dispositivo ao qual é servido (SILVA, 2014, p. 35). Responder, nesse contexto tem o sentido de movimentar-se expandindo e contraindo. O *design* responsivo se adapta com a finalidade de se acomodar de forma usável e acessível à área de visualização da tela, proporcionando ao usuário uma melhor experiência de navegação sem restrições de conteúdo servindo os mesmos recursos e informações para todos. O *Web Design Responsivo* é composto por técnicas avançadas de CSS3, HTML5 aliados ao *layout* fluído, imagens flexíveis e *media queries*.

Para demonstrar a viabilidade do uso das técnicas relacionadas ao RWD (*Responsive Web Design*) ou *Web Design Responsivo*, foram realizados experimentos em um *site* de compartilhamento de arquivos SISCA (Sistema de compartilhamento de arquivos). O objetivo dos experimentos foi a exploração do uso de *design* responsivo em termos de alternativas de *layout* que proporcionem a melhor experiência de navegação e a identificação das preferências de adaptação com relação às situações de visualização distintas. Além disso, o trabalho apresenta uma avaliação da qualidade percebida do *layout* por meio de testes em computadores, *notebooks*, *tablets*, *smartphones* e TVs.

1.1. Justificativa

Estudo da IDC (*Internet Data Center*) Brasil apontou queda de vendas no mercado brasileiro de notebooks e computadores *desktop*. No terceiro trimestre de 2014, houve um declínio nas vendas de 25% se comparado com o mesmo período de 2013 (*IDC Brazil PCs Tracker Q3, 2014*). Segundo Pedro Hagge (2014), analista de pesquisa da IDC Brasil, a concorrência com outros produtos levou a venda de computadores *desktop* para baixo “Antigamente, o PC era o único dispositivo que permitia o acesso à internet. Atualmente, o acesso está disponível em *tablets* e *smartphones*, que são produtos mais baratos”. De acordo com o Google os *smartphones* já faziam parte do dia-a-dia de 26% da população Brasileira, e seus proprietários dependem cada vez mais de seus dispositivos. O estudo ainda aponta que 46% acessam a internet todos os dias no *smartphone* e uma boa parte não saem de casa sem ele (Nosso Planeta *Mobile* 2013, 2013). Estudo da IDC Brasil mostra que vendas de *smartphones* ultrapassaram a marca de 15 milhões de unidades no terceiro trimestre de 2014. O mesmo estudo aponta crescimento de 49% em comparação ao mesmo período de 2013. De acordo com a e-bit, o *smartphone* foi o item mais procurado na *Black Friday* de 2014 (*webshoppers*, 2015). Os *tablets* também registraram crescimento constante de vendas. Conforme a IDC Brasil, no terceiro trimestre de 2014 foram comercializadas 2,3 milhões de unidades, alta de 18,1% se comparado com o mesmo período de 2013. É notável que diante desse aumento nas vendas de dispositivos móveis fez surgir uma mudança de paradigma, em que o acesso à internet não é exclusivo aos computadores, mas a diversos outros dispositivos como *smartphones*, celulares, TVs, *e-readers*, *tablets*, *notebooks* e *netbooks*.

As páginas *web* para computadores *desktop*, quando acessadas em dispositivos móveis, nem sempre proporcionam uma boa experiência de navegação para o usuário. Portanto, a adaptação das aplicações torna-se necessária, é preciso desenvolver para *web* com a finalidade de contemplar todos os diferentes tipos de aparelhos com acesso à internet independente da resolução da tela. Dennis Altermann em seu artigo classifica o *design* responsivo como uma evolução lógica

para o *design* de páginas *web*, oferecendo um único *site* ou aplicação para pessoas com diferentes perfis, adaptando-os à experiência do usuário com o dispositivo (ALTERMANN, 2012). Pensar em responsividade é repensar a *web* para o futuro, e sair das limitações de um *browser desktop* e seu tamanho previsível, e pensar em páginas com flexibilidade que suportem todo tamanho de tela (ZEMEL, 2012).

Em análise realizada através de simulações de acesso ao *site* não responsivo a partir de dispositivos distintos, foi identificado que o SISCA não atende as necessidades atuais de ser adaptável a qualquer aparelho que tenha acesso à internet e quando acessado através de dispositivos móveis proporciona uma experiência de acesso não usável, desgastante e limitada. Diante dessas barreiras apresentadas faz-se necessário a transformação do *site* seguindo a filosofia de desenvolvimento aplicando os conceitos de *Web Design Responsivo*.

1.1. Objetivos

Nesta seção serão apresentados os objetivos, geral e específico deste trabalho de conclusão de curso.

1.2.1 Objetivo geral

Estudar e avaliar os conceitos sobre *Web Design Responsivo* e demonstrar sua aplicação em um *site* de compartilhamento de arquivos (SISCA).

1.2.2 Objetivos específicos

Com base no objetivo geral, pretende-se alcançar ao final deste trabalho os seguintes objetivos específicos:

- Conhecer os princípios do *design* responsivo;
- Estudar sobre *media queries*, *layout* fluido, imagens flexíveis e CSS3;
- Converter *layout* do SISCA em responsivo.

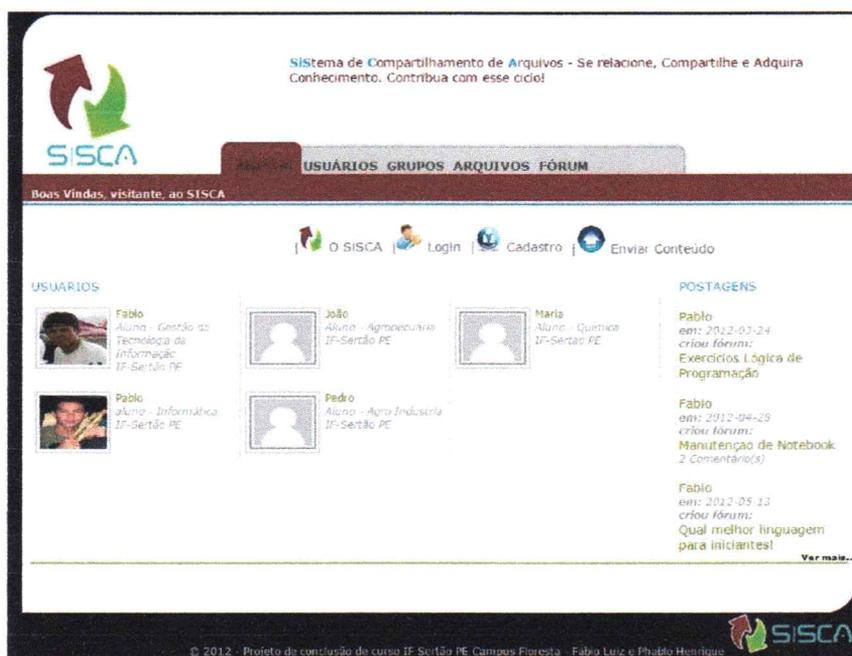
1.3. Protótipo SISCA

O sistema de compartilhamento de arquivos (SISCA), utilizado como protótipo para transformação de *layout* de largura fixa para *layout* responsivo, baseando-se nos conceitos de RWD. Foi desenvolvido em 2012, pelos discentes Fábio Luiz da Silva Sá e Phablo Henrique de Melo França, e serviu de base para o projeto de conclusão do curso técnico subsequente em informática do Instituto Federal do Sertão Pernambucano - *Campus Floresta*.

O SISCA foi desenvolvido para facilitar o acesso às informações, conteúdo acadêmico e científico produzidos pelos discentes e docentes do Instituto Federal do Sertão Pernambucano - *Campus Floresta*.

O *site* consiste em base de dados desenvolvida para reunir, organizar, armazenar, recuperar, disseminar e tornar acessível às informações entre a comunidade acadêmica.

Figura 1 – Apresentação do SISCA



Fonte: próprio autor.

2. FUNDAMENTAÇÃO TEÓRICA

Nesta etapa serão abordados os conceitos que dão sustentação e possibilitam a aplicação de projetos responsivos.

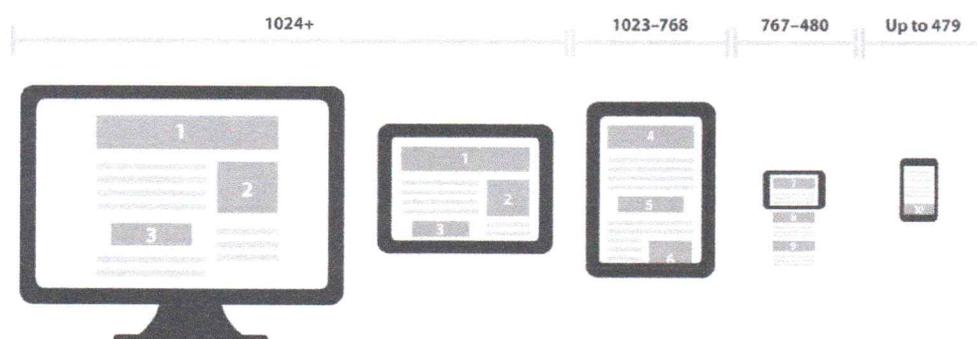
2.1. *Web Design* Responsivo

O termo em inglês "*Responsive Web Design*" (RWD) foi utilizado e abordado pela primeira vez em 2010 por Ethan Marcotte, em seu artigo publicado no site "A List Apart". Ethan Marcotte, considerado o pai do *design* responsivo, apresentou nesse artigo os pilares do RWD que abrange *Layout* fluido ou *Grids* flexíveis, Imagens e recursos flexíveis e *Media Queries* (SILVA, 2014). Esses pilares serão abordados separadamente neste trabalho.

RWD é a abordagem que sugere que o *design* e desenvolvimento devem responder ao comportamento do usuário e do ambiente baseando-se no tamanho da tela, plataforma e orientação (MARCOTTE, 2010). Sites responsivos são aquelas páginas que se adaptam a todo tipo de dispositivo (LOPES, 2013). O usuário abre a mesma página no celular, *tablet* e no *desktop* e ela se adapta para melhorar a experiência do usuário.

O *design* responsivo aborda técnicas e tecnologias capazes de adaptar páginas *web* para responderem de forma satisfatória a vários tipos de dispositivo eletrônico com acesso à internet, sem que haja perda de informações por parte do usuário. Na Figura 1 é mostrado um exemplo de *layout* responsivo apresentado em diversos dispositivos como: *desktop*, *smartphone* e *tablet* (orientação retrato e paisagem).

Figura 2 – Layout responsivo.



Fonte: Midiatismo.¹

Diante do exemplo da Figura 2 é possível observar que o *layout* de um *site* com *design* responsivo se harmoniza dentro da área que o cabe, de forma que o conteúdo se adeque mantendo a navegabilidade, visualização legível e sem perder informações, independente do tamanho de tela do dispositivo que o acessa.

Entre os princípios abordados das técnicas de RWD, um deles sugere a não utilização de linguagens *mobile*, bem como, não criar versões, subdomínios e diretórios específicos para *smartphones*, a intenção seria evitar a divisão em *web* convencional e *web mobile*, afinal a *web* é única (ZEMEL, 2013, p. 10).

O desenvolvimento de um *site* com base em cada dispositivo gera algumas dificuldades, inclusive quando é lançado um novo modelo de aparelho celular com uma resolução de tela diferente, assim, será necessário desenvolver uma nova versão do *site* que contemple a esse novo aparelho e aos aparelhos anteriores. (ZEMEL, 2013, p. 14).

¹ Disponível em <http://www.midiatismo.com.br/o-mobile/design-responsivo-entenda-o-que-e-a-tecnica-e-como-ela-funciona>

O problema relacionado ao desenvolvimento de aplicação *web* tradicional que não se adapta a qualquer dispositivo, não é técnica, mas sim conceitual, pois eles não foram projetados previamente para serem flexíveis (ZEMEL, 2013, P.13).

Diante dos problemas de adaptar o *site* para que seja acessado em qualquer dispositivo eletrônico, algumas estratégias são utilizadas, como o "m ponto", que ao acessar uma página *web* através do *Smartphone*, o navegador redirecionará para uma URL semelhante à `http://m.website.com.br`, porém, se a URL for acessada em um computador *desktop*, o conteúdo não terá uma boa apresentação e nem uma boa navegabilidade, pois se trata de uma versão *mobile* (ZEMEL, 2013, p. 14).

Outro conceito encontrado na obra de ZEMEL é o *One Web*, que consiste em uma única URL e um único sistema com o mesmo conteúdo para todos. Ainda há outra técnica chamada RESS (*Responsive Design + Server Side Components*), onde se serve a mesma URL e a mesma página, porém o conteúdo é ajustado no *server-side*, conforme o navegador (ZEMEL, 2013, p. 14).

Na Figura 3 é apresentado como a técnica de *design* responsivo permite que a página *web* responda a qualquer dispositivo, de modo que não haja perda de conteúdo e proporcione ao usuário uma navegação agradável.

Figura 3 - Exemplo de um *layout* responsivo.



Fonte: (ZEMEL, 2013, p. 21)

2.2. Mobilidade

O acesso à *web* por dispositivos móveis (*laptops*, *smartphones* e *tablets*), difere do acesso tradicional através dos computadores *desktop*. (LOUREIRO; MATEUS, 1998, p.2), antes do surgimento dos dispositivos móveis, para acessar a Internet tínhamos como única alternativa o computador (*Desktop*), ao mesmo tempo a mobilidade nos oferece dispositivos móveis (*smartphones* e *tablets*) que se encontram presentes conosco na maior parte do tempo (DARIVA, 2011, p. 3).

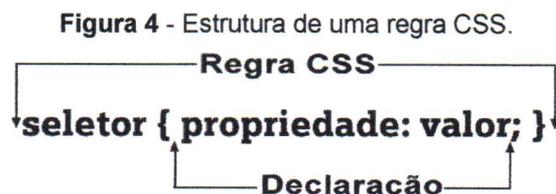
O número de usuários que acessam websites por meio de dispositivos móveis cresceu 106 % entre 2012 a 2013. Além da presença crescente de *smartphones* nas casas dos brasileiros, o número de domicílios com computadores e que possuem *tablets* passou de 4% em 2012 para 12% em 2013 (CETIC, 2013).

De acordo com levantamento da Associação Brasileira de Telecomunicações (Telebrasil), a banda larga móvel, pelas redes de 3G e 4G, cresceu 69% em 2013, se comparado a 2012. O levantamento também constatou que das 111,4 milhões de conexões registradas em 2013, 96,1 milhões partiram de telefones celulares, e 15,3 milhões foram por outros meios de acesso e terminais de dados (TELEBRASIL, 2014).

2.3. Web Design Responsivo e CSS3

O conhecimento de CSS (Folhas de Estilo em Cascata, do inglês *Cascading Style Sheets*) é primordial para o desenvolvimento de *layout* responsivo. Convém deixar claro que não é escopo deste trabalho abordar todas as funcionalidades das CSS, e sim, algumas das funcionalidades que frequentemente são utilizadas no *design* responsivo.

A definição mais precisa e simples para folha de estilo encontra-se na *homepage*² das CSS no *site* do W3C e diz: “Folha de estilo em cascata é um mecanismo simples para adicionar estilos (por exemplo: fontes, cores, espaçamentos) aos documentos *web*.”. Na Figura 4 é possível visualizar a estrutura da regra CSS.



Fonte: (SILVA, 2014, p. 72)

De acordo com a Figura 4, uma regra CSS é formada por duas partes: um **seletor** e uma **declaração**. Uma declaração CSS é composta de uma propriedade CSS seguida por um valor CSS e dele separado por dois-pontos (:) (SILVA, 2014, p. 72). Vale ressaltar que uma regra CSS poderá possuir mais de uma declaração CSS, desde que sejam devidamente separadas por ponto e vírgula (;).

A Figura 5 apresenta um exemplo de uma folha estilo CSS:

Figura 5 - Exemplo folha de estilo CSS.



Fonte: (SANTOS, 2014)

² Disponível em: <http://www.webstyles-portuguese.info/Style/CSS/>

2.3.1 Tipos de medidas CSS

As medidas no CSS mais conhecidas são: pixel, ponto, em e porcentagem (ZEMEL, 2013, p. 24). A definição e utilização de cada uma delas serão descritas a seguir:

- Pixel (px): é o menor elemento num dispositivo de exibição. É a unidade de medida fixa mais utilizada no CSS. Os profissionais de *web design* utilizam essa medida em uma técnica chamada de “*pixel perfect*”. Essa técnica consiste em criar uma estrutura HTML/CSS idêntica à imagem gráfica criada, por exemplo, no *photoshop* pelo *designer*.
- Ponto (*point*): é unidade de tamanho fixo utilizado para CSS de impressão. 1 ponto é igual a 1/72 polegadas.
- Ems (em): é uma unidade escalável. Com relação ao tamanho de fonte, 1em, é o mesmo tamanho da fonte atual do elemento pai. Sendo 12px o elemento pai, 1em seria igual a 12px, 2em seria 24px.
- Porcentagem (%): é semelhante a unidade “em”, mas possui algumas diferenças. Entre as semelhanças temos que as duas unidades são escaláveis. Ao utilizar a unidade porcentagem, o alvo permanece totalmente escalável.

É possível perceber uma diferença básica entre *pixel*, ponto, em e porcentagem, os dois primeiros são unidades de medidas fixas, os dois últimos, são medidas relativas.

A técnica do *design* responsivo sugere que os desenvolvedores de páginas *web* optem por usar a unidade de medida porcentagem para o tamanho do *layout* (larguras, margens, espaçamentos). As medidas com valor em % se alteram de maneira conveniente, permitindo que a legibilidade seja preservada (ZEMEL, 2013, p, 25). Outra sugestão é usar em para as fontes. A medida em "em" pode até ser

usada fora de textos, mas vai ser sempre uma medida relativa ao tamanho da fonte (*font-size*) (ZEMEL, 2013, p, 25).

2.3.2 Limitando layout fluido com a propriedade CSS *max-width*

A propriedade CSS *max-width* é utilizada para restrição de um elemento específico ou a página inteira. A Figura 6, mostra o trecho de código usado para restringir a largura em 2000 pixels da página e centralizá-la através da utilização dos atributos *margin-left* e *margin-right*.

Figura 6 – Utilizando a propriedade *max-width* para limitar *layout* fluido.

```
1 body {  
2   max-width: 2000px;  
3   margin-left: auto;  
4   margin-right: auto;  
5   width: 100%;  
6 }
```

Fonte: (LOPES, 2013, p. 57)

2.4. Pilares de *Web Design* Responsivo

Segundo o autor da técnica de *Web Design* Responsivo (RWD), Ethan Marcotte, há três pilares para o desenvolvimento do *design* responsivo. A seguir serão descritos os principais conceitos de RWD: *Layout* fluido, Flexibilidade de imagens e recursos e, *Media Queries*.

2.4.1. *Layout* fluido ou *Grid* Flexível

O *Layout* fluido, também chamado de *Grid* flexível, é o passo inicial a ser dado para o desenvolvimento do projeto de *design* responsivo (MARCOTTE, 2011). A

ideia principal para se chegar a um *layout* fluido é “não utilizar medidas absolutas no CSS” como *pixel*, ponto e outras (ZEMEL, 2013, p. 23). Esse conceito sugere o uso de medidas relativas ao invés de absolutos, as medidas relativas devem ser usadas para especificar valores relacionados ao tamanho do elemento pai (LOPES, 2013, p. 49).

A simples mudança na declaração de valores CSS absolutos como *pixel(px)* por valores CSS relativos (% ou em) proporciona grande vantagem, pois, leva o elemento pai e suas subpartes a abranger um espaço proporcional à resolução do dispositivo. O exemplo da Figura 7 apresenta trecho de código com uso de medidas relativas.

Figura 7 – Exemplo de medidas flexíveis.

```
1 body {  
2     /* a página ocupa a largura da tela toda */  
3     width: 100%;  
4 }  
5  
6 article {  
7     /* o article ocupa 3/4 da página */  
8     width: 75%;  
9     /* e tem uma margem interna de 10% do tamanho do pai */  
10    padding: 10%;  
11 }
```

Fonte: (LOPES, 2013, p.49)

Diante das dificuldades em utilizar medidas flexíveis, por estar acostumado com medidas fixas copiadas direto do *photoshop*. A recomendação é pensar na proporção dos elementos para se ter um *layout* flexível (LOPES, 2013, p 53).

Figura 8 – Exemplo de proporção dos elementos.



Fonte: (LOPES, 2013, p.53)

Inicialmente o desenvolvedor deverá não se atentar ao tamanho em px (*pixel*) de cada bloco definido no arquivo gráfico feito pelo *designer* no *photoshop* ou *software* semelhante. A ideia é visualizar o desenho, por exemplo, o da Figura 8 e enxergar um painel de notícias dividido em quatro partes que somadas ocupem os 100% da tela (LOPES, 2013, p.53).

No exemplo da Figura 8, a proporção dos elementos da página é estabelecida com base na porcentagem.

Na proporção dos elementos, à medida que a resolução mude, muda-se o contexto e o tamanho final é recalculado automaticamente pelo navegador.

Lopes (2013) sugere que, com relação às fontes, seja utilizada a mesma estratégia da proporção dos elementos na página, se o título principal é 50% maior que o título secundário, aplica-se essa mesma relação, ou seja, um título com 2em e outro com 3em, caso a fonte base seja alterada os demais valores em “em” mudam proporcionalmente (LOPES, 2013, p. 54).

Para se chegar a um *layout* fluido, é fundamental a consolidação do entendimento e aplicação das técnicas, fazendo com que, de acordo com (SILVA,

2014, p.225) a unidade de medida relativa **em** “responda” em relação ao tamanho de fonte (*font-size*) e a unidade de medida **porcentagem** “responda” a largura da *viewport*.

Em caso de um *layout* existente baseado na técnica tradicional de “*pixel perfect*”, deve ser realizada a conversão de valores absolutos para valores relativos.

Na fase de transformação de *layout* de largura fixa para responsivo, muitas vezes os desenvolvedores envolvidos se deparam com problemas de conversão de medidas absolutas para medidas relativas (ZEMEL, 2013). Mas, felizmente, (MARCOTTE, 2011) sugeriu em seu artigo uma fórmula de realizar a conversão de valores absolutos para valores relativos.

A fórmula proposta por Ethan Marcotte foi a seguinte:

$$\text{Target} / \text{Context} = \text{Result}$$

Em tradução livre quer dizer:

$$\text{Alvo} / \text{Contexto} = \text{Resultado}$$

Aplicação deste cálculo, por exemplo, em um título com tamanho de fonte de 20px, seria:

$$20 \text{ (alvo)} / 16 \text{ (contexto)} = 1,25 \text{ (resultado)}$$

A fórmula é utilizada para calcular medidas relativas de fontes e *layout*. (ZEMEL, 2013, p.46) frisa que ao usarmos medidas relativas para fontes, devemos definir a medida com **em** e quando se tratar de medidas de *layout* utilizar **porcentagem**.

2.4.2. *Imagens e Recursos flexíveis*

Componentes como imagens, vídeos, e *widgets* de forma geral aparecem com frequência em páginas ou aplicação *web*. Obviamente que inseridos em um projeto responsivo esses componentes devem ser flexíveis (SILVA, 2014, p.161).

As imagens e os recursos flexíveis são partes bem complicadas e de difícil resolução em um projeto de *design* responsivo. Pois, no caso das imagens, normalmente elas contêm número fixo de *pixel*, colocar porcentagem fará essas imagens aumentarem ou diminuir de acordo com o tamanho da tela ou do elemento pai. Por exemplo, uma imagem de 400px, sendo esticada a 100% de uma tela de 1280px, ficaria com uma aparência bem ruim (LOPES, 2013, p. 131).

A cada dia são desenvolvidas diferentes técnicas com objetivo de solucionar o problema com imagens flexíveis, tais como o uso de *JavaScript*, linguagem *server side* entre outros. Porém, essas técnicas, sejam elas já existentes ou recém-inventadas não resolvem totalmente o problema. Ou seja, nenhuma delas é perfeita ou ideal. De acordo com (SILVA, 2014, p. 161) a recomendação seria o desenvolvedor assumir a responsabilidade e ter conhecimento das técnicas disponíveis, podendo citar: *Adaptive images*, *picturefil*, *jQuery Picture*, *Riloadr* e entre outras, e assim decidir qual a mais conveniente de acordo com as características do projeto responsivo.

De forma geral, fazer com que uma imagem se torne fluída e contraia a medida que a resolução mude é uma tarefa relativamente simples e pode ser feita utilizando apenas CSS, no caso, com uso da propriedade *max-width*. Utilizando essa propriedade é possível que as imagens possam ter qualquer largura até no máximo 100% do elemento que estão contidas (ZEMEL, 2013, p. 57).

Diante dessa afirmativa, surgem possíveis questionamentos: qual a dificuldade então? Se o comportamento esperado para uma imagem flexível é fluir junto com o *layout*.

A resposta para esse questionamento é exemplificada na obra de (SILVA, 2014, p. 162). Supondo que uma imagem com extensão *jpeg* que tenha um tamanho de 800px por 600px com peso total de carregamento igual a 140 Kb a ser servida em uma página responsiva. Perceba que a imagem fluirá, mas não trará um resultado satisfatório para dispositivos móveis, em especial aos que utilizam conexões de telefonia móvel. Esses usuários serão penalizados por não disporem de uma conexão mais veloz no momento do acesso. Para solucionar esse problema Zemel

(2013, p 67) sugere que quando o acesso se der via *mobile*, o mais conveniente é que seja criada uma imagem, de menos peso e tamanho para ser servida a fim de aprimorar a experiência da visita.

A regra aplicada com a propriedade CSS *max-width* assim como em imagens, também pode ser utilizada para outros tipos de recursos e mídias, tais como: *iframe*, *object*, *embed* e *video* (ZEMEL, 2013, p. 60). Como mostra o trecho do código CSS da Figura 9.

Figura 9 – Aplicando flexibilidade em imagens e outras mídias.

```
img,  
iframe,  
object,  
embed,  
video {  
    max-width: 100%;  
}
```

Fonte: (ZEMEL, 2013, p. 60)

2.4.3. Media queries

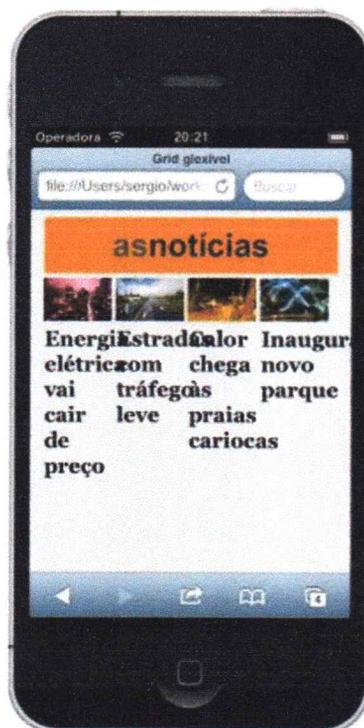
São três as técnicas apresentadas por Ethan Marcotte, para desenvolver um projeto responsivo: *layout* fluido, imagens e recursos flexíveis e *media queries*.

Uma página construída com medidas flexíveis é totalmente fluida e já está 90% no caminho de um design adaptável a todo tipo de tela. Mas é difícil fazer tudo ficar bonito e ajustado a todo tipo de resolução só com **porcentagens** e **em**. Portanto, o *design* fluido não resolve todos os problemas. Aí que as *media queries* entram. Elas são essenciais para um bom *design* responsivo. (LOPES, 2013, p. 59).

Na Figura 10 temos um exemplo de *layout* fluido sem o uso de *media queries*, percebe-se que o *layout* flui, porém, elementos como textos ficam desajustados e

sobrepostos, tornando a página pouco usável. O ideal seria em telas pequenas organizar as notícias verticalmente de forma que ocupe 100% da tela (LOPES, 2013, p. 60).

Figura 10 - *Layout fluido sem Media Queries.*



Fonte: (LOPES, 2013, p. 60)

A técnica mais apropriada para solucionar o problema apresentado na Figura 9, é a adição do *design* condicional que de acordo com (LOPES, 2013, p. 60) só será aplicado em situações específicas, permitindo readaptar o *design* da página de acordo com características do navegador. O local no CSS em que é colocada uma *media query* é chamado de *breakpoint*, que é o ponto de quebra do *layout fluido* em que necessita de ajuste no *design* para melhorar a experiência do usuário.

O uso de *media queries*, é a principal aliada do *layout fluido*, as duas combinadas a mídias flexíveis, tornam a experiência de navegação mais prazerosa.

Na Figura 11 é apresentado o trecho do código utilizado para solucionar o problema mostrado na Figura 9.

Figura 11 – Exemplo de *Media Queries*.

```

1 .noticia {
2     float: left;
3     width: 25%;
4 }
5
6 @media (max-width: 400px) {
7     .noticia {
8         float: none;
9         width: 100%;
10    }
11 }

```

Fonte: (LOPES, 2013, p. 60-61)

Na Figura 11, na linha seis vemos uma declaração de *media query* que define uma regra para largura de *viewport* com no máximo 400 *pixel*, para que elementos declarados com a classe CSS *.noticia* ocupem 100% da largura da página. O resultado pode ser visto na Figura 12.

Figura 12 – Layout fluido com uso de *Media Queries*.



Fonte: (LOPES, 2013, p. 61)

Na Figura 12, percebe-se que o comportamento dos elementos mudou em relação à Figura 10, com intuito de melhorar a experiência de acesso do usuário.

Também é possível observar que o uso da *Media Queries* apresentou um resultado satisfatório a página.

Analisando a Figura 12, há o emprego da técnica *breakpoint* e dispõe o conteúdo em apenas uma coluna.

A Figura 13 ilustra a técnica que contempla resoluções no formato paisagem em dispositivos móveis.

Figura 13 - Exemplo de *breakpoint* para dispositivo móvel



Fonte: (LOPES, 2013, p.63)

A regra CSS que define o *breakpoint* no exemplo da Figura 13 pode ser visualizada conforme a Figura 14.

Figura 14 – Exemplo de *breakpoint* com *media query*.

```

1 @media (max-width: 600px) {
2   .noticia {
3     width: 50%;
4   }
5 }
  
```

Fonte: (LOPES, 2013, p. 62)

Ao utilizar várias *media queries*, é recomendável respeitar a ordem em que os *breakpoints* forem escritos no código fonte, pois uma regra sobrescreve a anterior (LOPES, 2013, p. 63).

3. METODOLOGIA

Este trabalho de conclusão de curso teve como alicerce metodológico a pesquisa bibliográfica a respeito do tema RWD.

Foi realizada pesquisa bibliográfica relacionada à transformação e implementação de *layout* responsivo. O conjunto de técnicas e conceitos foi aplicado para adaptação de um protótipo destinado aos dispositivos com tamanhos de tela variados. O resultado desse estudo, os testes e um comparativo da versão anterior sem o uso das técnicas de RWD, bem como a versão implementada e os resultados obtidos serão disponibilizados no final deste trabalho.

Com relação ao *layout* do protótipo SISCA, inicialmente foram realizadas simulações, análises e testes para verificar possíveis incompatibilidades na navegação em diversos dispositivos tais como: *smartphone*, *tablet*, *notebook*, *desktop*, *smart TV* e também emuladores na *internet*, podendo citar: *ipad peek*³ e *mobile phone emulator*⁴, que permitiram simular o comportamento do site no navegador de vários dispositivos móveis.

Tais simulações de acesso tanto em dispositivos físicos quanto em emuladores virtuais, tiveram como objetivo, levantar informações e identificar deficiências, das quais foram solucionadas ao longo do processo de adaptação do *site* para responsivo, que primeiramente definiu-se através de *wireframe* o formato de *layout* responsivo que se pretendia obter. Na sequência foram analisados todo o código HTML e CSS e aplicar o conceito de *layout fluido* para que elementos da página tornassem flexíveis. Em seguida foram inseridos as *media queries* nos *breakpoint* identificados que necessitavam de maiores ajustes e também aplicação do design condicional. Em seguida foi empregado o conceito do terceiro pilar que compõe o RWD, que consiste em aplicar flexibilidade de imagens e recursos. E por fim, necessitou-se hospedar o SISCA em servidor *web* para que fosse possível realizar o acesso por meio de qualquer dispositivo eletrônico com internet.

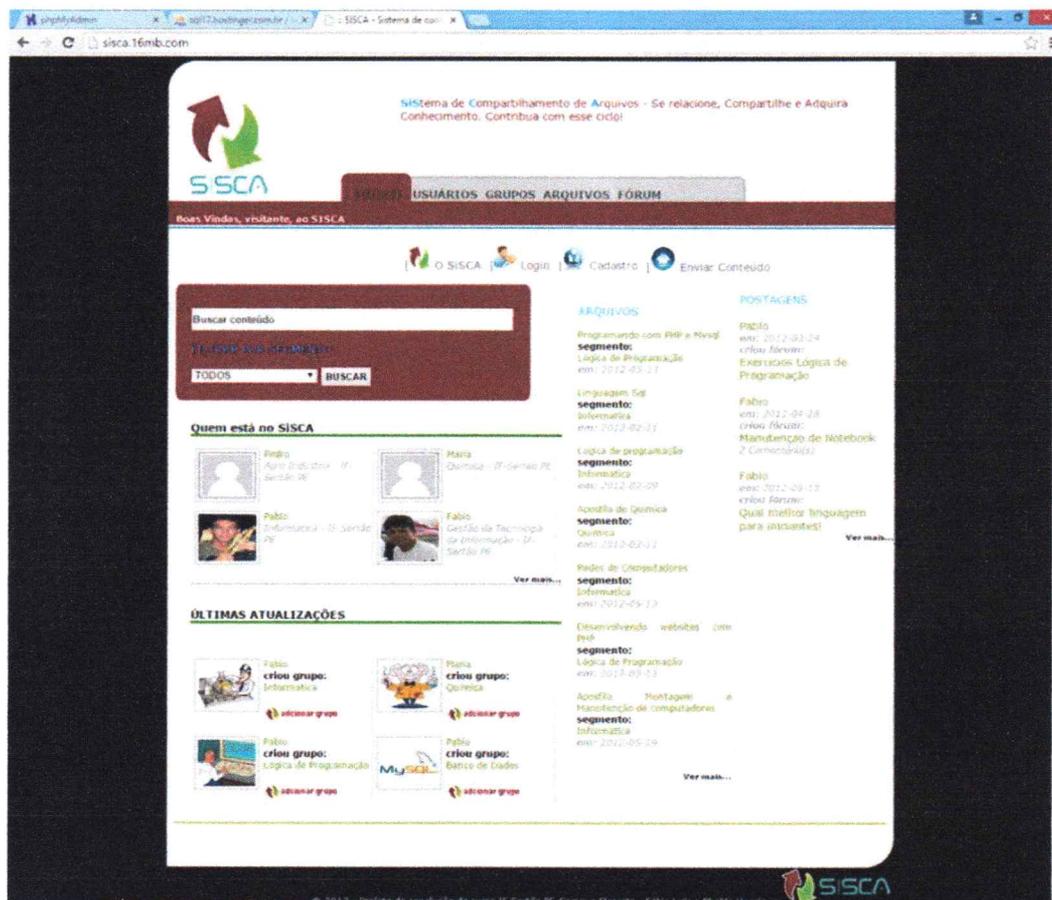
³ Disponível em: <http://ipadpeek.com/>

⁴ Disponível em: <http://www.mobilephoneemulator.com/>

4. CENÁRIO DO PROTÓTIPO SISCA

A Figura 15 ilustra a página inicial do *site* de largura fixa sendo renderizada através de navegador de computador *desktop*. Covem deixar claro que a utilização do protótipo SISCA foi meramente da interface gráfica.

Figura 15 - Página de apresentação do sistema SISCA.



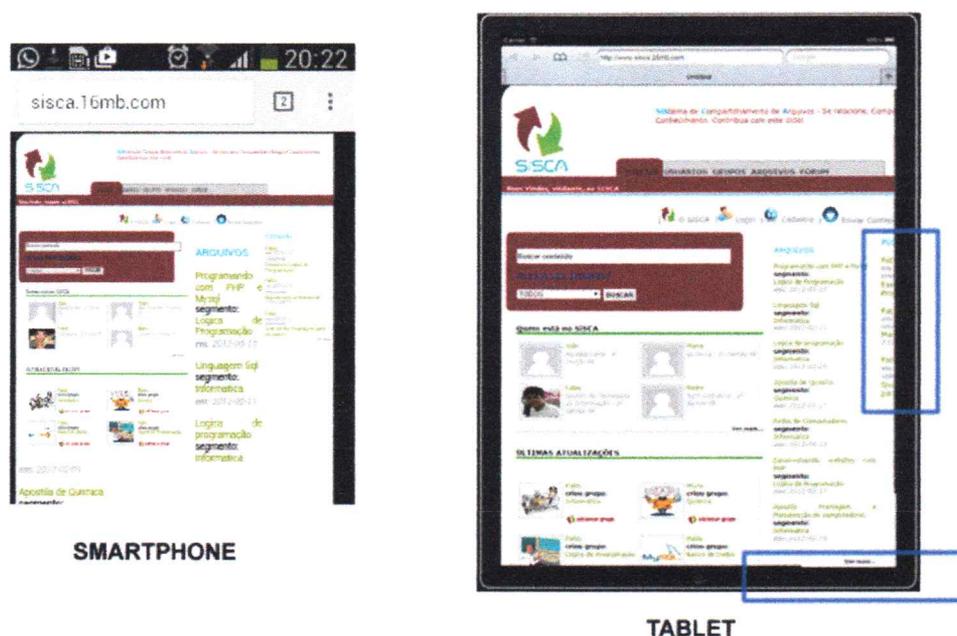
Fonte: arquivo pessoal

É possível identificar que o *layout* apresenta uma boa navegabilidade e legibilidade do conteúdo, assim, não proporcionaria grandes problemas aos usuários com telas de tamanhos superiores a 800 *pixels*. Em contrapartida, o *site* não foi desenvolvido seguindo a filosofia do RWD, que torna o *layout* adaptável aos diversos tipos de resoluções de tela dos dispositivos eletrônicos. Sendo assim, quando o acesso se dava através de outros dispositivos, tais como: *tablet* e

smartphone, o resultado não era tão satisfatório, pois a primeira versão do protótipo foi pouco usável pelos usuários de dispositivos móveis, logo, era difícil e desagradável o acesso às informações.

Na Figura 16 é possível visualizar através de simulações, como o *site* é renderizado quando acessado em diferentes dispositivos.

Figura 16 - Apresentação do sistema SISCA em dispositivo móvel.



Fonte: arquivo pessoal

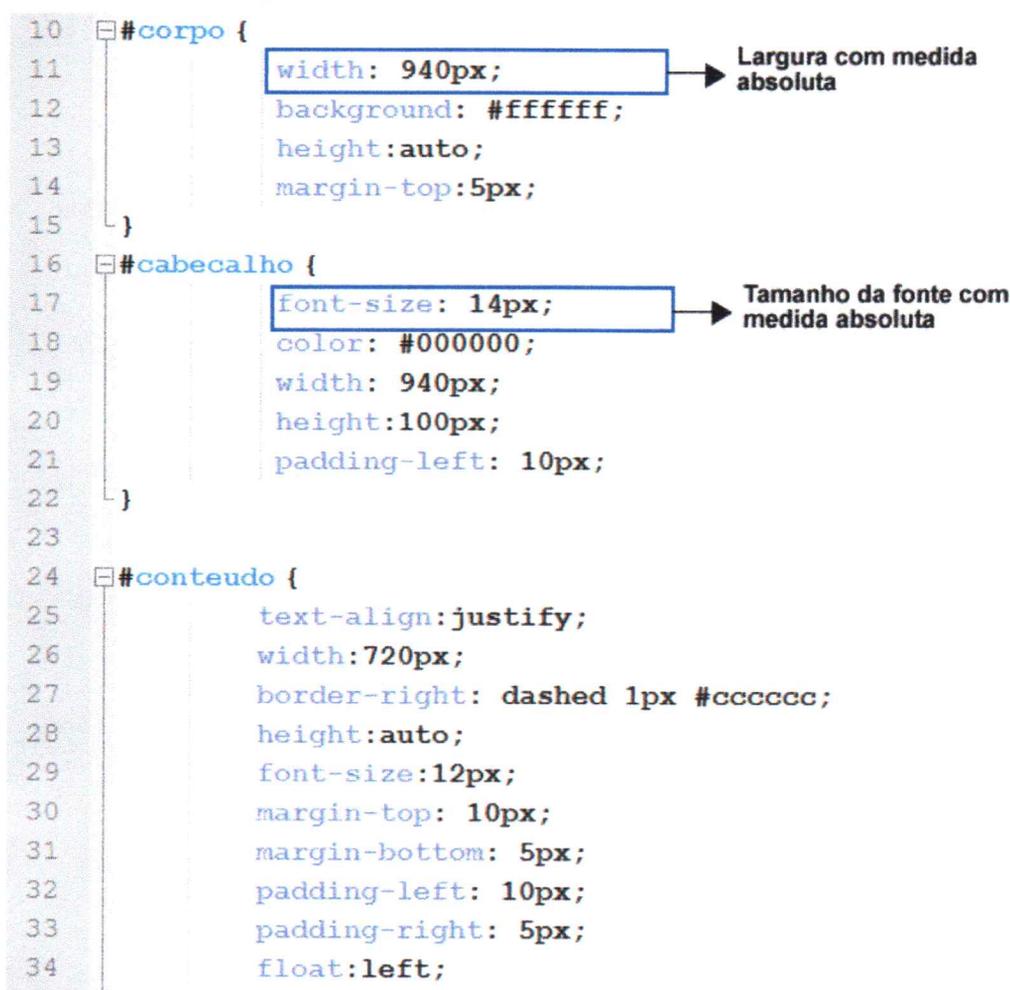
A Figura 16 mostra simulação do *site* de largura fixa sendo acessado em dispositivos móveis. Na simulação são notáveis as deficiências ao ser visualizado, identifica-se que o *site* de largura fixa tem sua usabilidade comprometida, pois não se adapta de forma usável, ficando com o aspecto encolhido, os textos ilegíveis dificulta a leitura e links difíceis de ser tocados nos *smartphones*.

Ao ser acessado em *tablets* em modo retrato, nota-se que elementos posicionados à direita, ultrapassam a largura da tela do dispositivo e ficam cortados gerando incômoda barra de rolagem horizontal.

Nas figuras seguintes serão ilustrados trechos dos códigos CSS e HTML do *layout* atual, para que possam ser identificadas as características que o impossibilita de ser responsivo.

Figura 17 - Folha de estilo CSS do *site* não responsivo.

```
10  #corpo {
11      width: 940px;
12      background: #ffffff;
13      height:auto;
14      margin-top:5px;
15  }
16  #cabecalho {
17      font-size: 14px;
18      color: #000000;
19      width: 940px;
20      height:100px;
21      padding-left: 10px;
22  }
23
24  #conteudo {
25      text-align:justify;
26      width:720px;
27      border-right: dashed 1px #cccccc;
28      height:auto;
29      font-size:12px;
30      margin-top: 10px;
31      margin-bottom: 5px;
32      padding-left: 10px;
33      padding-right: 5px;
34      float:left;
```



Fonte: arquivo pessoal

Analisando a Figura 17, observa-se que na linha 11 do código CSS, a largura (*width*) é declarada com medida absoluta, *pixel* (px), bem como a propriedade fonte (*font-size*) que também tem seu valor declarado com medida absoluta, as declarações de valores em *pixel* se estendem ao longo de todo estilo do código CSS

e impedem que o *site* se comporte de forma responsiva, desse modo, fazendo-se necessário a reescritura do código CSS, alterar todos os valores declarados com medidas absolutas para medidas relativas, aplicar flexibilidade de imagens e recursos e a inserção das *media queries*.

Outro fator que incapacita o *layout* de ser responsivo pode ser visualizado na Figura a seguir que mostra trechos de códigos HTML da página de largura fixa.

Figura 18 - Código HTML do *site* não responsivo.

```

1 <html>
2 <head>
3   <title>:: SISCA - Sistema de compartilhamento de arquivo acadêmico ::</title>
4   <meta name="description" content="sistema compartilhamento de arquivos acadêmicos">
5   <meta name="keywords" content="compartilhamento de arquivos acadêmicos, Floresta, :
6   <link rel="stylesheet" type="text/css" href="css/estilo.css">
7 </head >

```

Fonte: arquivo pessoal

Analisando a Figura 18 acima entre as linhas 2 e 7, é possível identificar que não foi inserida no cabeçalho (HEAD) do arquivo HTML a meta *tag viewport*, que de acordo com (SILVA, 2014 pág. 56) ela fornece ao navegador informações que permitem que ele escale de modo conveniente o conteúdo destinado a ser servido em telas e larguras de *viewport* variadas. O *viewport* é o espaço disponível para a página ser renderizada no navegador (LOPES, 2013, pag. 68). Ou seja, é o espaço em branco em que a página abre.

4.1. Adaptando o SISCA utilizando os conceitos de RWD

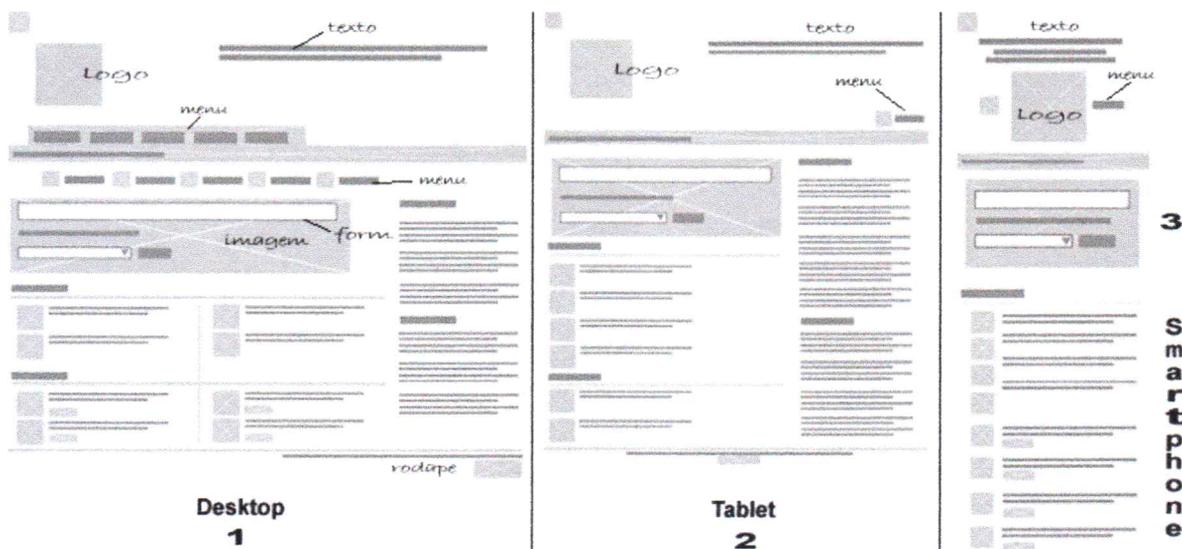
Essa etapa destina-se a aplicar as técnicas de *Web Design* Responsivo a fim de reparar as deficiências apresentadas anteriormente com relação ao *layout* de largura fixa e torná-lo responsivo.

4.1.1. Projeto de interface de design responsivo

Nesta seção será abordado o projeto de *interface* que possibilita ter uma visão previa e simplificada de como se organizará o conteúdo do SISCA em diferentes dispositivos, tais como computadores, *tablets* e *smartphones*.

Para tal finalidade teve-se como base a representação de baixa fidelidade com o uso de *wireframe*, que é uma espécie de esqueleto do projeto. Com o uso de *wireframe* foi possível propor uma representação de *layout* responsivo que pode ser visto na Figura 19 abaixo.

Figura 19 – *Wireframe* protótipo SISCA.



Fonte: arquivo pessoal

A Figura 19 mostra três *wireframes*, sendo cada um para determinado dispositivo. O primeiro *wireframe* destina-se a computadores *desktop*, TVs ou grandes resoluções, e tem como característica a divisão do *layout* em três colunas.

O segundo *wireframe* destina-se aos *tablets* e resoluções intermediárias, pode-se perceber que a disposição de alguns elementos é alterada com relação ao *wireframe* de *desktop*, e assume uma divisão do conteúdo em duas colunas. Para a versão *mobile* o *menu* tradicional é substituído por um *menu* de seleção.

O terceiro *wireframe* foi para o grupo dos *smartphones* e pequenas resoluções, os elementos são estruturados em uma única coluna, um abaixo do outro, fazendo com que a página se expanda na vertical, de forma a acomodar todo conteúdo principal, aprimorando a experiência do usuário possibilitando uma navegação simplificada e intuitiva.

Com os *wireframes* definidos para cada grupo de dispositivos, inicia-se a parte de transformação do código fonte do protótipo SISCA para que se torne responsivo.

4.1.2. Transformação do Protótipo SISCA em responsivo

A primeira transformação a ser realizada foi na marcação HTML, com a inserção da meta *tag viewport*, etapa fundamental para o correto funcionamento das *media queries*. As alterações podem ser vistas conforme é mostrado na Figura 20.

Figura 20 - Código HTML do site responsivo.

```

1 <!doctype html>
2 <html lang="pt-br">
3 <head>
4 <title>:: SISCA - Sistema de compartilhamento de arquivo acadêmico ::</title>
5 <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
6 <meta name="viewport" content="width=device-width,initial-scale=1">
7 <meta name="description" content="sistema compartilhamento de arquivos acadêmicos">
8 <meta name="keywords" content="compartilhamento de arquivos acadêmicos, Floresta,

```

Fonte: arquivo pessoal

Com a adição da meta *tag viewport* na linha seis, no trecho do código HTML, a atribuição do parâmetro *content="width:device-width"*, quer dizer que o *viewport* assumirá um valor específico de cada dispositivo. Exemplo, em um *smartphone* com 320px de largura, o *viewport* é de 320px.

A meta *tag* pode receber outros parâmetros, porém o *content="width:device-width" initial-scale=1* é o mais adequado para grande maioria dos sites *mobile*. (LOPES, 2013, pag. 77). O parâmetro *initial-scale=1*, indica que o *site* será renderizado com o zoom no tamanho real dos elementos.

Assim como o código HTML, os códigos CSS também precisaram ser reescritos conforme as recomendações de *web design* responsivo.

A etapa seguinte foi substituir as medidas absolutas para medidas relativas, o objetivo é fazer com que o *layout* se torne fluido. Para tal finalidade foi necessário realizar a conversão de medida com base na formula proposta por Ethan Marcotte. Ao longo de toda folha de estilo CSS, as propriedades que continham medidas em pixel (px), aplicou-se a formula de conversão para obter o valor equivalente em porcentagem (%). Algumas alterações no código são vistas na Figura 21.

Figura 21 - Código CSS do site responsivo.

```

14 #corpo {
15     width: 68.814055636896%; /*940px * 100 div 1336px = 68.814055636896%*/ max-width:940px;
16     background: #ffffff;
17     height:auto;
18     margin-top:1% /*5px */
19 }
20
21 #cabecalho {
22     width:100%; /*940px 100%*/
23     height: auto;
24     padding-left: 3%; /*20px;*/
25 }
26
27 #conteudo {
28     width:100%; /*940px*/
29     height:auto;
30     font-size: 0.750em; /*12px / 16px = 0.750em */
31     margin-top: 1.3888888888%; /*10px;*/
32     margin-bottom: 0.6944444444%; /*5px;*/
33     padding-left: 0.4166666666%; /*3px;*/
34     padding-right: 0.4166666666%; /*3px;*/
35     float:left;
36 }
37
38 #logo-rodape {
39     width:100%; /*940px 100%*/
40     height:4.3617021277%; /*41px;*/

```

Fonte: arquivo pessoal

Após transformar os valores da folha de estilo que continham medidas absolutas e alterá-las para medidas relativas, o passo seguinte foi à adição de *media queries* com *breakpoints* para os dispositivos móveis e *desktop*.

Figura 22 - Código CSS com *breakpoint* para *mobile*.

```

536 /* inicio do codigo responsivo para dispositivos moveis*/
537
538 @media screen and (max-width:820px) {
539
540 #corpo {
541     width: 84.4%; /* 794px x 100 div 940px = 84.4%*/
542     background: #ffffff;
543     height:auto;
544     margin-top:1%;
545 }
546
547 #cabecalho {
548
549     width:100%;
550     height:140px;
551     padding-left: 2%;

```

Fonte: arquivo pessoal

Na Figura 22 do código CSS na linha 538 foi declarado o primeiro *breakpoint*. A partir das simulações de acesso, foi constatado que a primeira quebra significativa do *layout* ocorre para uma largura de *viewport* de 820px.

A segunda quebra significativa foi voltada ainda para os dispositivos móveis, em especial, os diversos *smartphones*.

Figura 23 - Código CSS com *breakpoint* para smartphone.

```

751  /* inicio do codigo responsivo para smartphones*/
752  @media screen and (max-width:370px) {
753
754  #cabecalho{
755      text-align:center;
756      height:130px;
757  }
758  #conteudo {
759
760      width:90%; /*335px x 100 div 370px = 90%*/
761  }
762
763  #busca {
764      width:95%; /*350px*/
765      height: auto;
766      background-image: url('../images/img-busca.png');
767      background-size:100% 15%;
768      background-repeat: no-repeat;
769      padding: 2%;

```

Fonte: arquivo pessoal

Figura 24 - Código CSS com *breakpoint* dispositivos móveis

```

901  /* codigo responsivo, mantém os destaques em coluna unica*/
902  @media screen and (max-width:768px){body {background:#000000;}}
903  #break-destaque{
904      width:100%; /* 480px*/
905      float:left;
906      padding-left:1px;
907      padding-right:3px;
908      border-right: none;
909
910  }
911  .estilo_arq {width:90%; float:left; border-right:none; padding-left: 1%;}
912
913  }

```

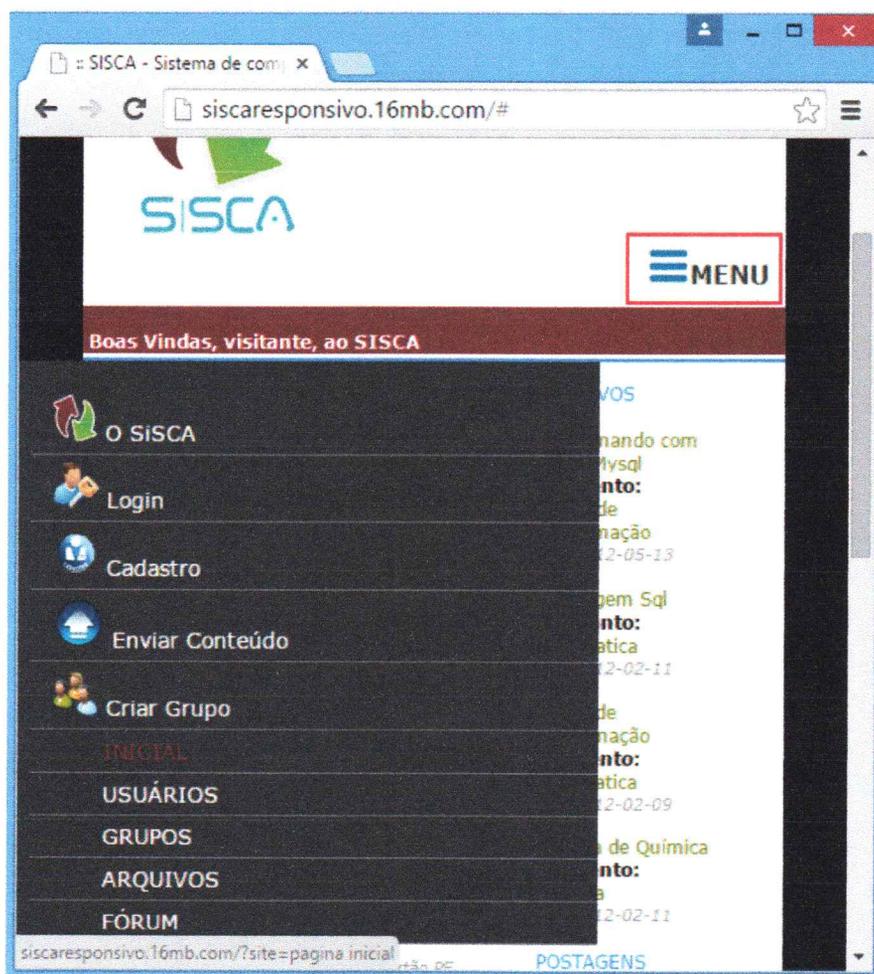
Fonte: arquivo pessoal

O terceiro e último *breakpoint* tem como objetivo dispor os elementos do conteúdo para que sejam organizados em uma única coluna de acordo com o que foi definido previamente no *wireframe* do projeto, contemplando a categoria dos dispositivos móveis.

Durante o processo de revisão do estilo CSS, foram usados recursos para testar o comportamento do *layout* em simulações com dispositivos de telas de tamanhos variados e redimensionamento do navegador.

Por fim, foi implementado um novo elemento na página HTML, voltado aos dispositivos móveis, que foi o *menu* de seleção, que de acordo com o tamanho da resolução, muda a sua disposição no *layout* proporcionando uma navegação simplificada para usuários *mobile*.

Figura 25 - Menu *mobile* site responsivo

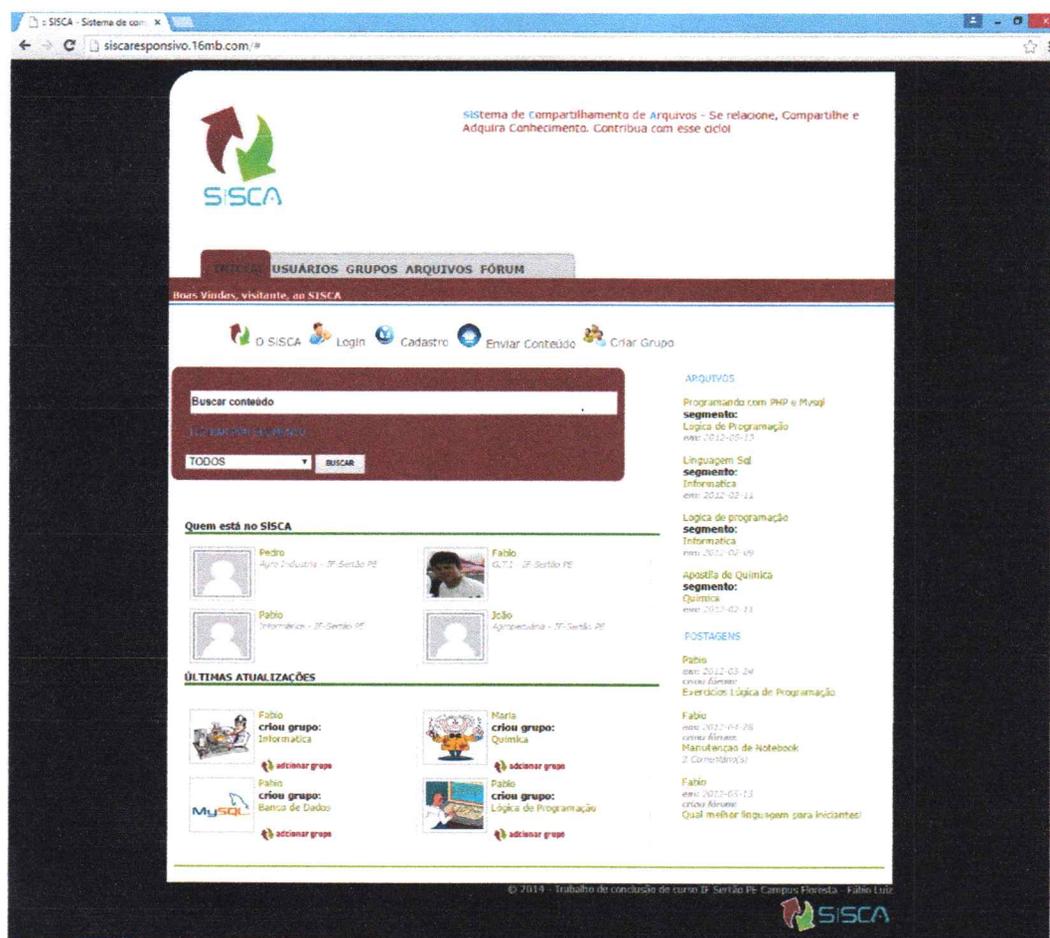


Fonte: arquivo pessoal

Nas Figuras 26, 27, 28, 29, 30 e 31 é possível ter uma visão de como o *site* responsivo se comporta quando acessado por diversos dispositivos e a atuação dos *breakpoints* com cada *layout* sendo utilizado.

A Figura 26 ilustra o SISCA sendo visualizado em navegador *Google chrome* versão 43.0.2357.124, resolução de 1280 x 1024 de computador *desktop*.

Figura 26 - Site responsivo renderizado em computador *desktop*



Fonte: arquivo pessoal

Figura 27 - Site responsivo renderizado em dispositivos móveis

PAISAGEM



RETRATO



Fonte: arquivo pessoal

A Figura 27 ilustra o SISCAR sendo visualizado através de *ipad* em modo retrato e paisagem. Pode-se perceber que a disposição dos elementos da página muda em

relação à configuração do dispositivo, isso porque, quando o aparelho está em modo retrato e em seguida é rotacionado para modo paisagem, automaticamente a largura da *viewport* muda, em consequência, o *layout* se adapta em tempo de execução para se acomodar de forma mais conveniente com a resolução da tela.

Outro ponto a ser observado com relação ao *layout* responsivo do site SISCA, é que houve uma melhora na visualização dos elementos da página se comparado com a versão não responsiva. O conteúdo do site é totalmente visível e se adequa proporcionalmente à área de exibição da tela, solucionando a problemática de surgimento da barra de rolagem horizontal.

Figura 28 - Site responsivo renderizado em *Smartphone*



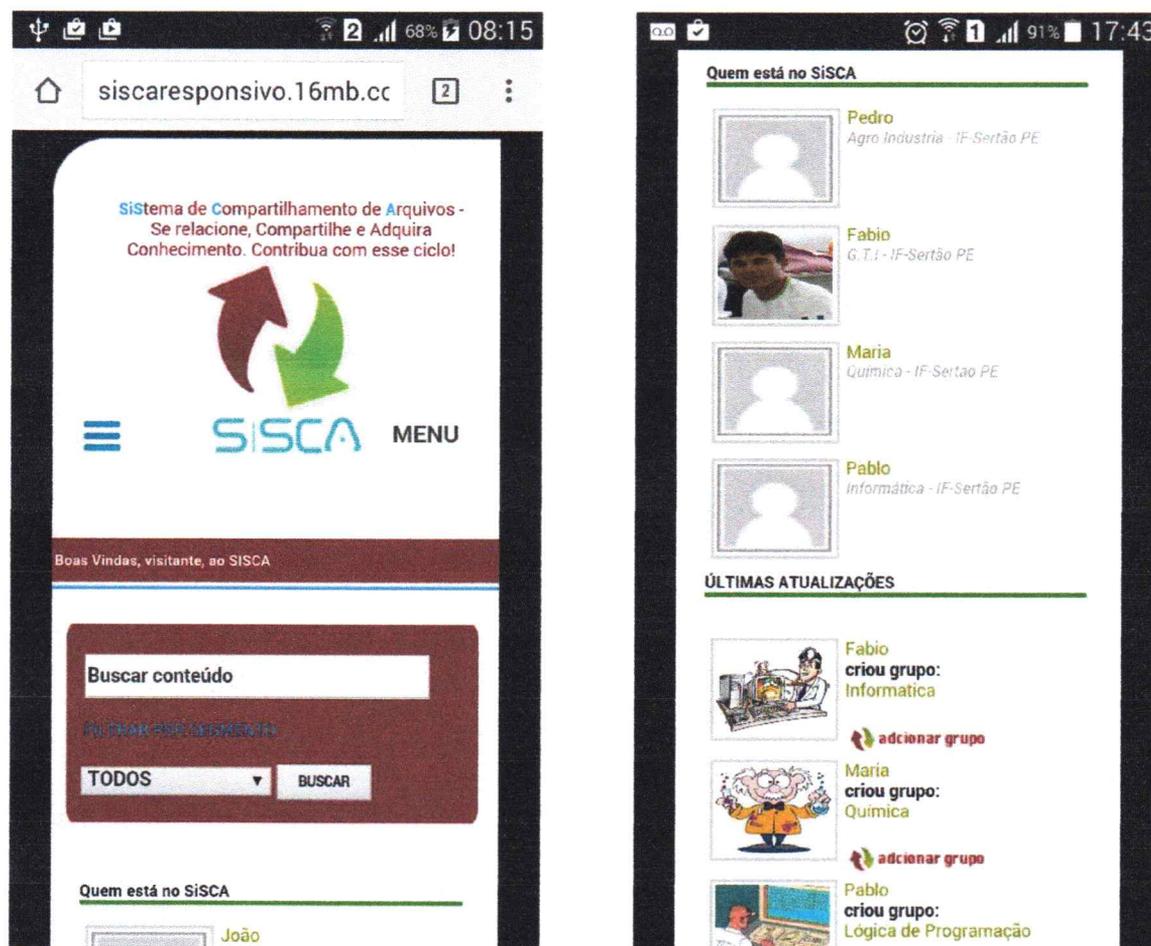
Fonte: arquivo pessoal

A Figura 28 apresenta o site responsivo sendo renderizado em navegador de simulador virtual ⁵ que simula a visualização em *smartphone* no modo de orientação retrato e paisagem. Da mesma forma que ocorre com o exemplo anterior da Figura

⁵ Simulador utilizado para realizar testes de acesso em diferentes resoluções, Disponível em: <http://ipadpeek.com/>

27, é possível perceber que a disposição dos elementos dá página sofrem mudanças a fim de acomodarem-se ao novo contexto.

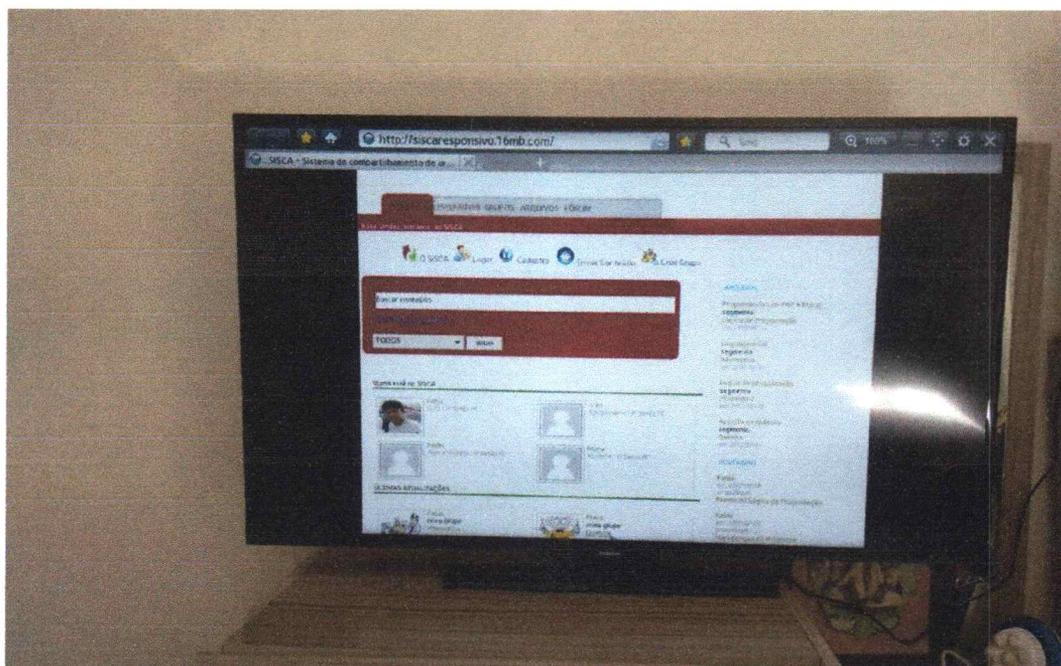
Figura 29 - Site responsivo renderizado em *Smartphone*



Fonte: arquivo pessoal

A Figura 29, também representa a forma visual de como o SISCA é renderizado em dispositivo móvel. O exemplo da Figura 29 se deu através de acesso por navegador *Google chrome* de *smartphone*. Pode-se notar que diante do acesso por meio de pequenas resoluções, o conteúdo do *site* responsivo é estruturado automaticamente em uma única coluna utilizando 100% da largura da tela do dispositivo, mantendo textos legíveis e imagens visíveis, sem perdas de conteúdo e mantendo as mesmas informações para usuários da categoria *mobile*.

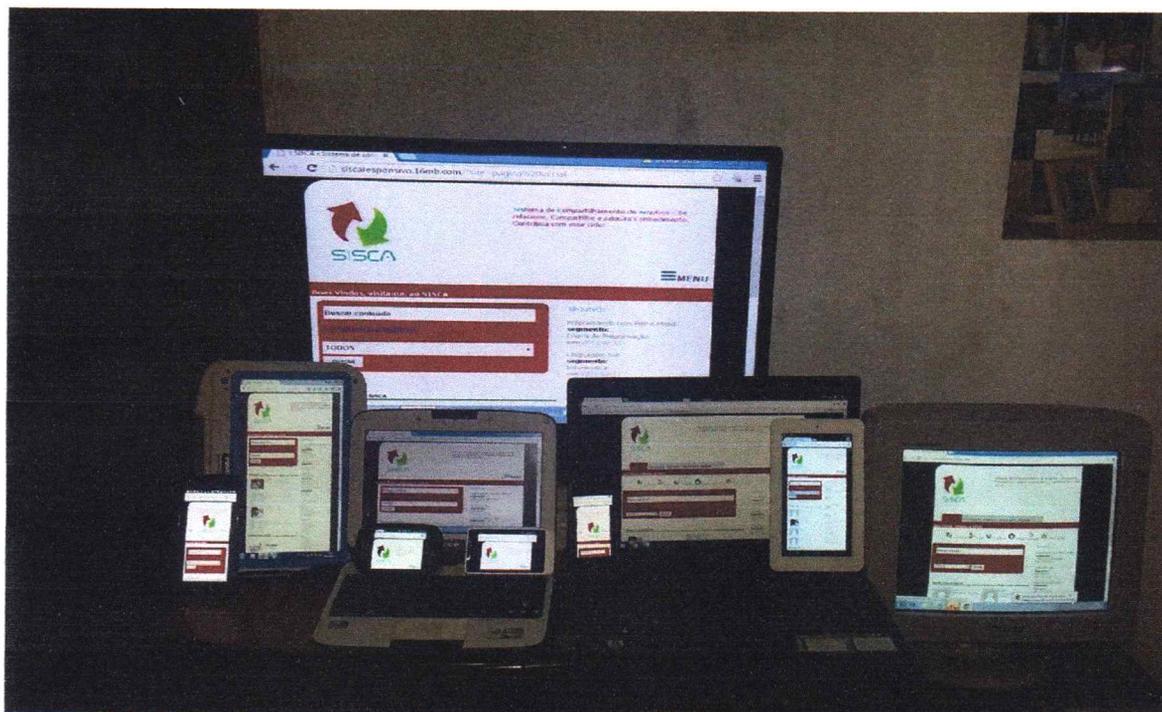
Figura 30 - Site responsivo renderizado em *SmartTV*



Fonte: arquivo pessoal

A Figura 30 ilustra o SISCAR sendo renderizado em navegador proprietário de uma *Smart TV LED 40" Full HD Samsung* com resolução de tela de 1.920 x 1.080. É possível observar que o resultado foi considerado satisfatório para os extremos de alta resolução.

Figura 31 - Site responsivo renderizado em diversos dispositivos



Fonte: arquivo pessoal

A Figura 31 ilustra um exemplo de acesso simultâneo ao *site* responsivo a partir de diversos dispositivos tais como: Computador *desktop*, *Tablet*, *notebook*, *smartphones* e *netbook*.

5. CONSIDERAÇÕES FINAIS

Diante das imagens do site responsivo, percebe-se que independentemente da largura da tela, o *layout* se adapta de modo que todos os seus aspectos principais são mantidos, sendo esse, o principal objetivo de *Web Design Responsivo*, proporcionar um único site para todos os dispositivos, servindo as mesmas informações, sem perdas, sem cortes de imagens, mantendo a legibilidade e dispondo de uma única versão de código fonte que contemple aos dispositivos de hoje, e também, os dispositivos do futuro.

A escolha do tema RWD e a implantação do mesmo no protótipo SISCA se deram pelo motivo do interesse particular pelo tema *Web Design Responsivo*, que é uma abordagem nova em relação ao desenvolvimento de aplicações *web* voltada a multi-dispositivos.

Diante do trabalho apresentado, é importante compreender que foram utilizados recursos de linguagens *web* que são usadas freqüentemente e que dão suporte suficiente para desenvolver aplicações usáveis e funcionais para os diversos tipos de dispositivos eletrônicos.

Entende-se que as orientações dadas para realização do estudo gerou bons resultados, atendendo as expectativas de obter um único *site* ou aplicação que contemplasse a diversas resoluções de tela independente do dispositivo que o acessa.

O presente trabalho de conclusão, desenvolvido no âmbito do curso de Gestão da Tecnologia da Informação do Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano, *Campus – Floresta* foi de grande relevância para a comunidade acadêmica.

5.1. Trabalhos futuros

Nesta seção serão descritas algumas sugestões de pesquisa que podem ser realizadas a partir deste trabalho.

- Avaliação de usabilidade do sistema SISCA;
- Análise da eficácia do SISCA, aplicação de estudo de caso com alunos e professores em sala de aula;
- Aplicação das técnicas atuais relacionadas a imagens flexíveis compará-las e demonstrar os resultados de uso;
- Adição de outras funcionalidades ao SISCA que contribuam para o processo de ensino aprendizagem.

REFERÊNCIAS

ALTERMANN, D. **Design Responsivo: Entenda o que é a técnica e como ela funciona.** Disponível em: <<http://www.midiatismo.com.br/o-mobile/design-responsivo-entenda-o-que-e-a-tecnica-e-como-ela-funciona>> Acesso em 10/02/2015.

CETIC, **Pesquisa de indicadores TIC domicílios 2013 – CETIC.** Disponível em: <<http://www.cetic.br/pesquisa/domicilios/indicadores>> Acesso em: 05/02/2015.

DARIVA, R. **Gerenciamento de Dispositivos Móveis E Serviços de Telecom.** São Paulo, SP - Brasil: Campus, 2011.

Ebit, Webshoppers 2015. Disponível em: <<http://www.ebit.com.br/webshoppers>> Acesso em 27/01/2015.

Google, **Nosso Planeta Mobile: Brasil. Como entender o usuário de celular.** Disponível em: <<http://pt.slideshare.net/alexandrehonorio/google-our-mobile-planet-2013>> Acesso em 19/12/2014.

IDC, **Estudo da IDC Brasil mostra recorde nas vendas de smartphones no terceiro trimestre de 2014.** Disponível em: <<http://br.idclatin.com/releases/news.aspx?id=1777>> Acesso em 26/01/2015.

IDC, **Mercado brasileiro de PCs registra queda de 25% no 3º trimestre de 2014.** Disponível em: <<http://www.idcbrasil.com.br/releases/news.aspx?id=1775>> Acesso em: 22/12/2014.

LOPES, S. **A Web Mobile. Programe para um mundo de muitos dispositivos.** São Paulo, SP - Brasil: Casa do Código, 2013.

LOUREIRO, A. A. F.; MATEUS, G. R. **Introdução à Computação Móvel. 1. ed.** Rio de Janeiro, RJ - Brasil: 11ª Escola de Computação, 1998.

MARCOTTE, E. **Responsive Web Design.** New York, NY - USA : A book apart, 2011.

SANTOS, M. **Aprendendo HTML a linguagem da web.** Disponível em: <<http://pensandonaweb.com.br/aprendendo-html-a-linguagem-da-web>> Acesso em: 05/02/2015

SILVA, S. M. CSS3: desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3. São Paulo,SP - Brasil. Novatec Editora, 2012.

SILVA, S. M. Web Design Responsivo. Aprenda a criar sites que se adaptam automaticamente a qualquer dispositivo, desde desktops até telefones celulares. São Paulo,SP - Brasil. Novatec Editora, 2014.

TELEBRASIL, Indicadores da comunicação móvel no Brasil. Disponível em:<<http://www.telebrasil.org.br/panorama-do-setor/consulta-a-base-de-dados>>
Acesso em: 05/02/2015

ZEMEL, T. Web Design Responsivo: Páginas adaptáveis para todos os dispositivos. São Paulo, SP - Brasil: Casa do código, 2013.