

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DE SÃO PAULO**

Campus São João da Boa Vista

Trabalho Final de Curso

4º ano – Curso Técnico em Informática

Prof. Breno Lisi Romano

**Uso de *Framework Bootstrap* e aplicação de *template* alterado às
necessidades do projeto Indra**

Aluno: Rafael Silva Braganholli

Prontuário: 1320157

São João da Boa Vista – SP

2016

Resumo

Este relatório técnico visa expor a relação entre o uso do *framework Bootstrap* no projeto Indra, seus *plug-ins* e a identidade visual desenvolvida para o mesmo, buscando exemplificar as motivações que levaram à decisão de utilizar padrões de design no desenvolvimento da aplicação.

Dentre os elementos presentes nesta obra, pode-se destacar: O *Framework Bootstrap*, suas funcionalidades, história e justificativa; O *template Freelancer*, cujo papel foi servir como modelo para o design do projeto; A criação da identidade visual do Indra; e a adaptação e integração de todos os elementos citados à versão final do design da aplicação.

Sumário

1	Introdução	4
2	Desenvolvimento	6
2.1.	Bootstrap	6
2.1.1.	O <i>Framework</i>	6
2.1.2.	Funcionalidades	6
2.2.	Template “Freelancer”	7
2.3.	Criação de Identidade Visual	8
2.4.	Adaptação ao Projeto Indra	9
3.	Conclusões e Recomendações	10
4.	Referências Bibliográficas	17

1 Introdução

Com o começo do Quarto Ano do Ensino Médio Integrado ao Técnico em Informática, foi proposto um projeto para a matéria de PDS (Projeto de Desenvolvimento de Sistemas). Este projeto consiste em criar uma plataforma para controle de enchestes e dados meteorológicos em São João da Boa Vista, visto que este tem sido um problema recorrente no município. O projeto envolve vários setores acadêmicos da instituição, além do Quarto Ano Integrado, conta-se com a participação de projetos de Iniciação Científica, Extensão e Cursos Superiores.

Uma das primeiras tarefas do projeto consistiu em definir um *template*, isto é, um conjunto de arquivos que trazem “[...] toda a estrutura, diagramação e design do site, blog ou loja virtual [...]” [1]. Foi decidido que seria implementado, então, um *template* pronto baseado no *Framework Bootstrap*, apenas adaptando-o às necessidades e à identidade visual do projeto, que, mais tarde, veio a ser nomeado “Indra”. A seguir, nota-se um trecho de código do arquivo de estilo do *Bootstrap*, na Figura 01.

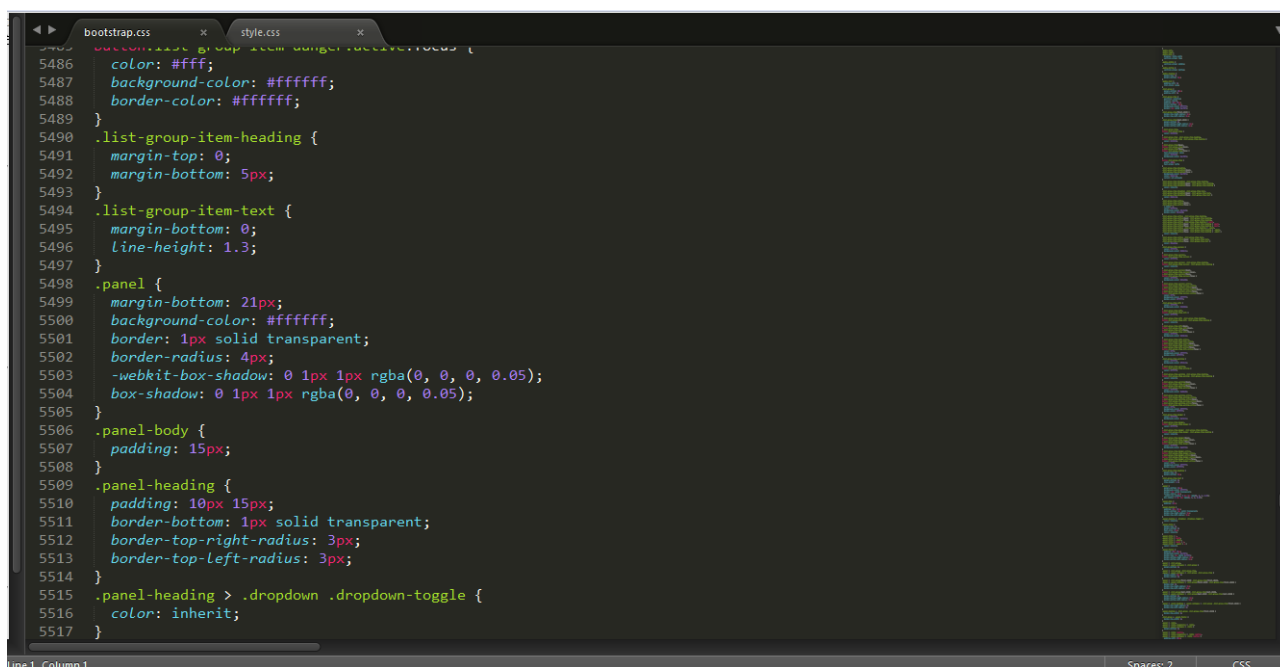
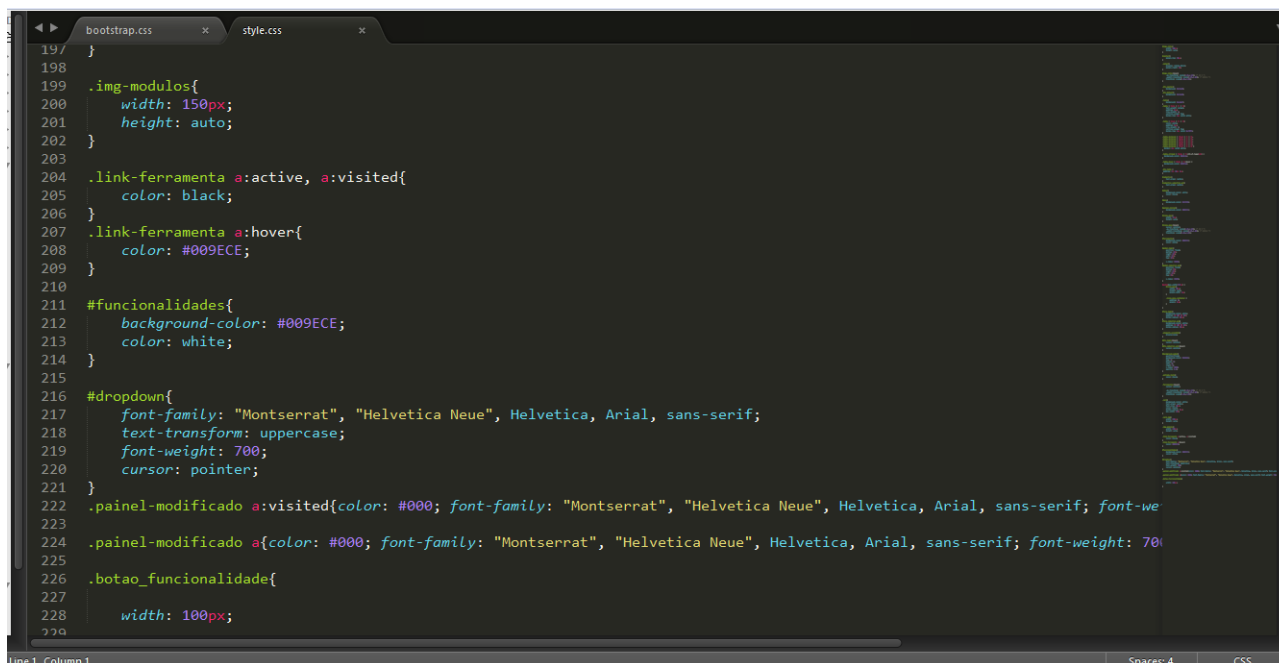


Figura 01 – Trecho de Código CSS do Bootstrap.

Justificou-se o uso do *framework Bootstrap*, principalmente, pela sua praticidade, facilidade em criar layouts responsivos e compatibilidade com todos os atuais navegadores (Thiago Nascimento, 2013). Abaixo, um exemplo do código personalizado de estilo do projeto.



```
197 }
198
199 .img-modulos{
200     width: 150px;
201     height: auto;
202 }
203
204 .link-ferramenta a:active, a:visited{
205     color: black;
206 }
207 .link-ferramenta a:hover{
208     color: #009ECE;
209 }
210
211 #funcionalidades{
212     background-color: #009ECE;
213     color: white;
214 }
215
216 #dropdown{
217     font-family: "Montserrat", "Helvetica Neue", Helvetica, Arial, sans-serif;
218     text-transform: uppercase;
219     font-weight: 700;
220     cursor: pointer;
221 }
222 .painel-modificado a:visited{color: #000; font-family: "Montserrat", "Helvetica Neue", Helvetica, Arial, sans-serif; font-we
223
224 .painel-modificado a{color: #000; font-family: "Montserrat", "Helvetica Neue", Helvetica, Arial, sans-serif; font-weight: 70
225
226 .botao_funcionalidade{
227
228     width: 100px;
229 }
```

Figura 02 – Folha de Estilo personalizada do Portal Indra

Além do uso de framework para acesso às classes CSS (*Cascade Style Sheets*) (SILVA, 2017), foi de grande importância criar a identidade visual do projeto, sendo assim, houve a necessidade de trabalhar com o próprio CSS do Projeto, isto é, com instruções e classes que definiriam as características visuais únicas do portal. Assim, conclui-se que a identidade visual engloba os conceitos e valores do projeto.

Portanto, a tarefa deste trabalho consiste em apresentar como e porque o uso de um *framework* pronto aliado a um *template* é justificado às necessidades do projeto Indra.

2. Desenvolvimento

2.1. Bootstrap

Para fugir da necessidade de escrever sempre milhares de classes CSS com o intuito de estilizar aplicações na web, muitos projetos utilizam do *Framework Bootstrap*, uma ferramenta de desenvolvimento *front-end* criada para desenvolver *templates* responsivos.

2.1.1. O Framework

Atualmente em sua versão 3.3.7, o Bootstrap possui CSS para estilização de ambientes web e tem como foco templates dinâmicos e resposivos, isto é, capazes de se adaptar a qualquer dispositivo, seja ele um computador, um *smartphone* ou *tablet*.

Desenvolvido por Mark Otto e Jacob Thornton, inicialmente, o *framework* levava o nome de “*Twitter Blueprint*”, hoje, o *Bootstrap* é mantido por uma pequena equipe de desenvolvedores, junto aos criadores previamente citados, por consequência, atualizações são lançadas periodicamente, sempre atualizando as tecnologias para proporcionar uma plataforma consistente aos *web designers*.

A ideia primordial ao desenvolver o *Twitter Blueprint* era encorajar a consistência entre ferramentas internas da empresa, incentivando o uso de padrões. Segundo Mark Otto:

“Um grupo super pequeno de desenvolvedores e eu nos reunimos para construir uma ferramenta interna e vimos a oportunidade de fazer algo a mais. Nesse processo, nos vimos criar algo muito mais substancial do que mais uma ferramenta interna. Meses depois, nós acabamos com uma versão jovem do Bootstrap, como uma maneira de documentar e compartilhar padrões de design dentro da companhia”[5].

Com a popularização do *Bootstrap*, mais versões foram lançadas. A versão 4.0 foi anunciada em 29 de outubro de 2014. Graças a essa popularização, existem cada vez mais projetos, ambientes web e *templates* responsivos no mercado. O grande sucesso da ferramenta se deve, também, ao seu código *open-source*, isto é, gratuito para uso.

2.1.2. Funcionalidades

Destacamos dentre as funcionalidades do *Framework*, principalmente, a responsividade, conceito que tem se tornado cada vez mais importante e presente nos requisitos de projetos web, uma vez que, cada vez mais usuários acessam a internet pelos seus dispositivos moveis. Segundo

uma pesquisa realizada pela WMcCann, em 2012, 41% dos entrevistados já utilizavam internet nos seus *smartphones* [6]. No Brasil, temos um crescente exponencial no uso de internet em *smartphones*. No segundo trimestre de 2015, o número de acessos à internet usando celulares ultrapassou 72 milhões, o que representou um aumento de 23,5% em relação ao semestre anterior [7]. Por essas razões, justifica-se o uso e desenvolvimento de *designs* responsivos. A seguir, na Figura 03, temos um exemplo de design responsivo exibido numa tela de *desktop* referente às ferramentas do Modulo 03 do Projeto Indra, responsável pelas Medições à esquerda, seguido da mesma página à direita, exibida em um simulador de tela de *smartphone*.

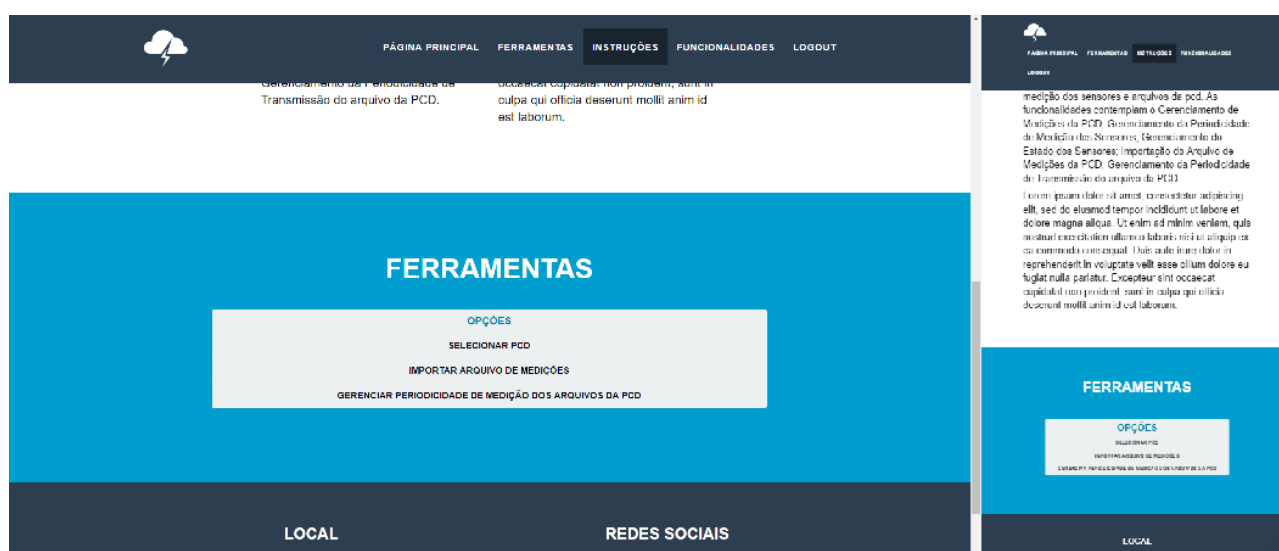


Figura 03 – Comparação de tela responsiva em diferentes tamanhos de display.

Além de suas classes CSS, o *Bootstrap* possui diversos *plug-ins* para o framework, facilitando a implementação de outras ferramentas que podem enriquecer ainda mais um projeto web [6].

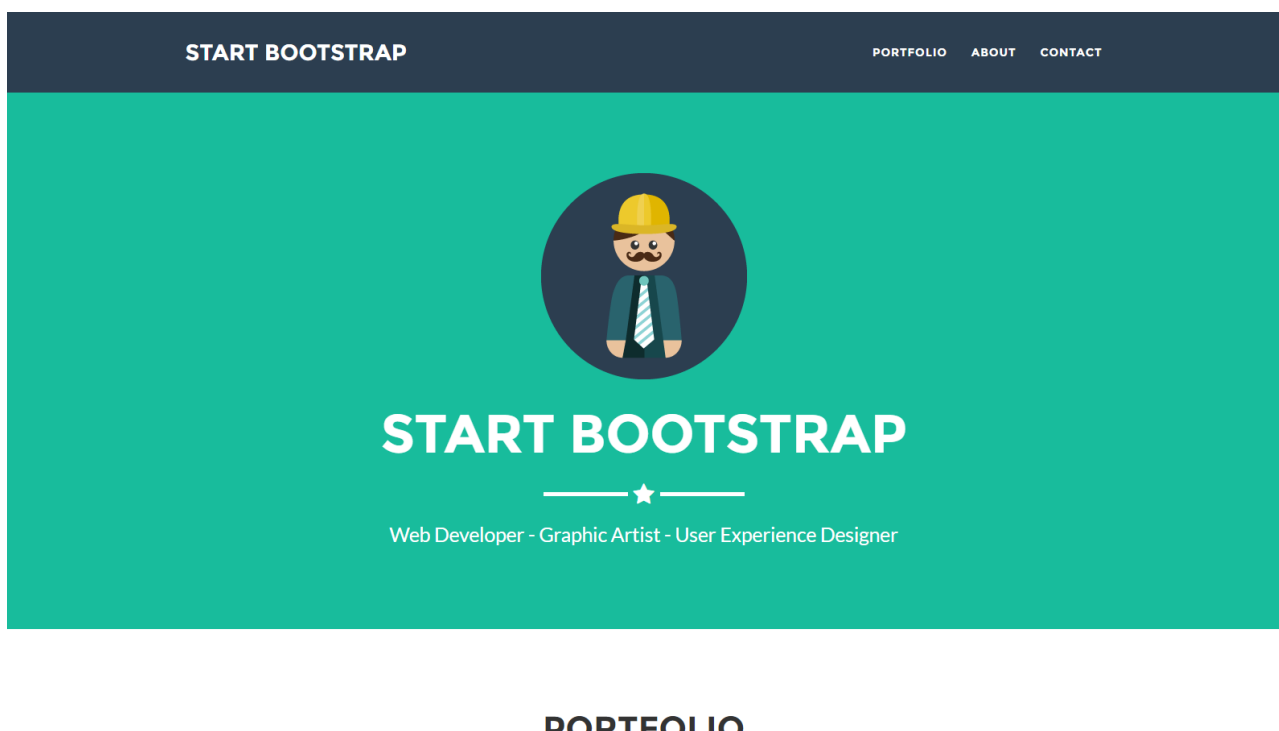
Em vista de projetos de larga escala, onde muitos trabalham juntos, a padronização de *design* é de suma importância, sendo assim, justifica-se o uso de *frameworks*.

Além da padronização e responsividade, o *Bootstrap* possui documentação muito bem definida, facilitando o entendimento e uso de suas classes e funcionalidades.

2.2. Template “Freelancer”

Na etapa de definição do template a ser utilizado no projeto, foi proposto que houvesse uma votação entre *templates* sugeridos pelos alunos.

Um dos grupos propôs utilizar o tema “*Freelancer*”. Deste tema, foram feitas alterações para se adequar ao projeto. A seguir, na Imagem 04, o tema *Freelancer*.



PORTFOLIO
Imagem 04 – Tema “*Freelancer*” com *Bootstrap*.

A ideia principal do tema é servir como expositor para, como diz o nome, *freelancers*, possuindo sessões para portfolio e contato, tudo em uma página. Isso fica claro no *slogan* do tema: “*One Page Theme*”.

Além de apresentar uma interface limpa, como era procurado, o tema já possuía a disposição de elementos da página bem-feita, e, além disso, uma barra de navegação que veio a ser útil e agregou valores de design ao projeto.

2.3. Criação de Identidade Visual

Convém que todo produto tenha uma identidade visual, assim, não é diferente com uma aplicação *web*.

Em primeiro momento, é necessário entender que uma identidade visual deve representar de forma sistematizada e resumida a principal ideia do produto, desta forma, esta identidade deve conter, agregados, algo que represente o produto, como um logotipo; cores que se relacionem com a temática do produto; na fonte dos textos, caso exista algum. Em poucas palavras, definir uma identidade visual é definir a aparência pela qual seu produto será reconhecido pelo público [9].

Ao criar uma identidade visual para o projeto Indra, foi relacionado seu objetivo e seu nome, advindo de um deus da mitologia Hindu, responsável pelas tempestades, desta forma, foi concebido o logotipo exibida abaixo na Figura 05.



Figura 05 – Logotipo do projeto Indra.

Também foi concebido um logotipo minimalista para agir como um resumo do logotipo mostrado na figura anterior. Este logotipo está representado a seguir, na Figura 06.



Figura 06 – Logotipo minimalista do projeto Indra.

Além do logotipo, foi almejado alcançar uma relação entre as cores do projeto e sua temática, sem perder o aspecto *clean* de seu *design*. O objetivo foi alcançado utilizando tons de azul em contraste com branco e fontes brancas e pretas. A cor azul é culturalmente associada ao clima, uma vez que relacionamos o mesmo à chuva.

2.4. Adaptação ao Projeto Indra

Justificou-se o uso de um *template* (*Freelancer*) devido à praticidade, uma vez que seria possível focar esforços em outras atividades, graças às qualidades do *template* escolhido, desta

forma, foi possível concentrar esforços apenas na adaptação do mesmo, sem a necessidade de construir um próprio *template* do início.

Em primeiro lugar, foi feita a aplicação da identidade visual ao *template*, assim, o *Freelancer* (Figura 04 – Página 08), começou a tomar forma como o projeto Indra. Nota-se a aparência da página principal do projeto na Figura 09, abaixo.



Figura 09 – Página principal do projeto Indra

Além de simplesmente aplicar as alterações ao *template* inicial, padrões foram definidos para manter o *design* único do portal. O primeiro padrão definido, além de cores, foi o padrão de tabelas. Para tal, foi utilizada um *plug-in* JQuery chamado *Data Tables*. Dentre as funcionalidades do *Data Tables* temos o padrão de paginação a cada 10 itens; barra de busca funcional que realiza pesquisas pelo próprio JQuery nos itens inseridos na tabela; botão de ordenação crescente/decrescente; responsividade; estilo de botões. Além de suas funcionalidades, o uso deste *plug-in* foi possível em todos os módulos do projeto, reforçando a ideia de padronização. Abaixo, é notável um exemplo do *plug-in* advindo de seu site oficial na Figura 10.

Show entries

Search:

Name	Position	Office	Age	Start date	Salary
Airi Satou	Accountant	Tokyo	33	2008/11/28	\$162,700
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09	\$1,200,000
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12	\$86,000
Bradley Greer	Software Engineer	London	41	2012/10/13	\$132,000
Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07	\$206,850
Brielle Williamson	Integration Specialist	New York	61	2012/12/02	\$372,000
Bruno Nash	Software Engineer	London	38	2011/05/03	\$163,500
Caesar Vance	Pre-Sales Support	New York	21	2011/12/12	\$106,450
Cara Stevens	Sales Assistant	New York	46	2011/12/06	\$145,600
Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29	\$433,060

Showing 1 to 10 of 57 entries

Previous 2 3 4 5 6 Next

Figura 10 – Exemplo do *plug-in Data Tables* e suas funcionalidades.

Bem como os demais aspectos do portal, as tabelas foram padronizadas em um estilo visual que correspondesse ao sistema como um todo. Nota-se este estilo na Figura 11, abaixo.


FERRAMENTA 3 PÁGINA PRINCIPAL LOGIN

SELEÇÃO DE PCDS

Procurar:

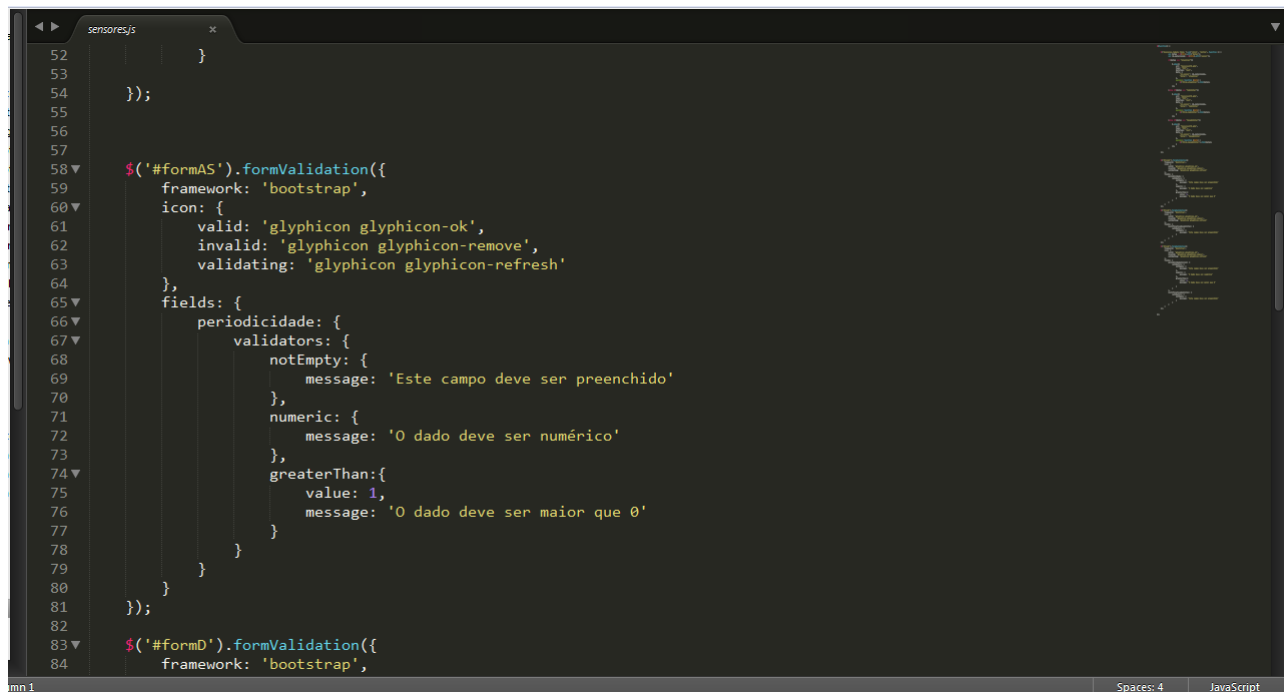
Codigo da PCD	Latitude	Longitude		
1	-21.9695	-46.7989	Gerenciar Medições	Gerenciar Sensores
2	-24.98	-47.7232	Gerenciar Medições	Gerenciar Sensores
3	-24.9582	-47.5632	Gerenciar Medições	Gerenciar Sensores
4	-30.9622	-66.6666	Gerenciar Medições	Gerenciar Sensores
100	-21.9695	-46.7989	Gerenciar Medições	Gerenciar Sensores

Mostrando página 1 de 1

Anterior Próximo

Figura 11 – Tabela de PCDs utilizando o *Data Tables* com estilo padronizado.

Outro padrão definido no projeto corresponde às validações dos campos de inserção. Foi utilizado outro *plug-in*. O *FormValidation*, adaptado para funcionar em conjunto ao *Bootstrap*, tem a função de, de acordo com restrições definidas pelos desenvolvedores, permita ou bloqueie o envio de formulários onde os campos correspondem ou não às restrições definidas. Por exemplo, é possível restringir um campo a apenas dados numéricos. Na Figura 12, abaixo, nota-se o código necessário para realizar tal restrição.



```
52 }
53
54 });
55
56
57
58 $('#formAS').formValidation({
59   framework: 'bootstrap',
60   icon: {
61     valid: 'glyphicon glyphicon-ok',
62     invalid: 'glyphicon glyphicon-remove',
63     validating: 'glyphicon glyphicon-refresh'
64   },
65   fields: {
66     periodicidade: {
67       validators: {
68         notEmpty: {
69           message: 'Este campo deve ser preenchido'
70         },
71         numeric: {
72           message: 'O dado deve ser numérico'
73         },
74         greaterThan: {
75           value: 1,
76           message: 'O dado deve ser maior que 0'
77         }
78       }
79     }
80   }
81 });
82
83 $('#formD').formValidation({
84   framework: 'bootstrap',
```

Figura 12 – Código JavaScript para restrição de inserção em campo de texto. Apenas dados numéricos.

Para exemplificar esta autenticação, mostra-se abaixo um formulário para alterar a periodicidade de medição de um sensor instalado em uma PCD na Figura 13.

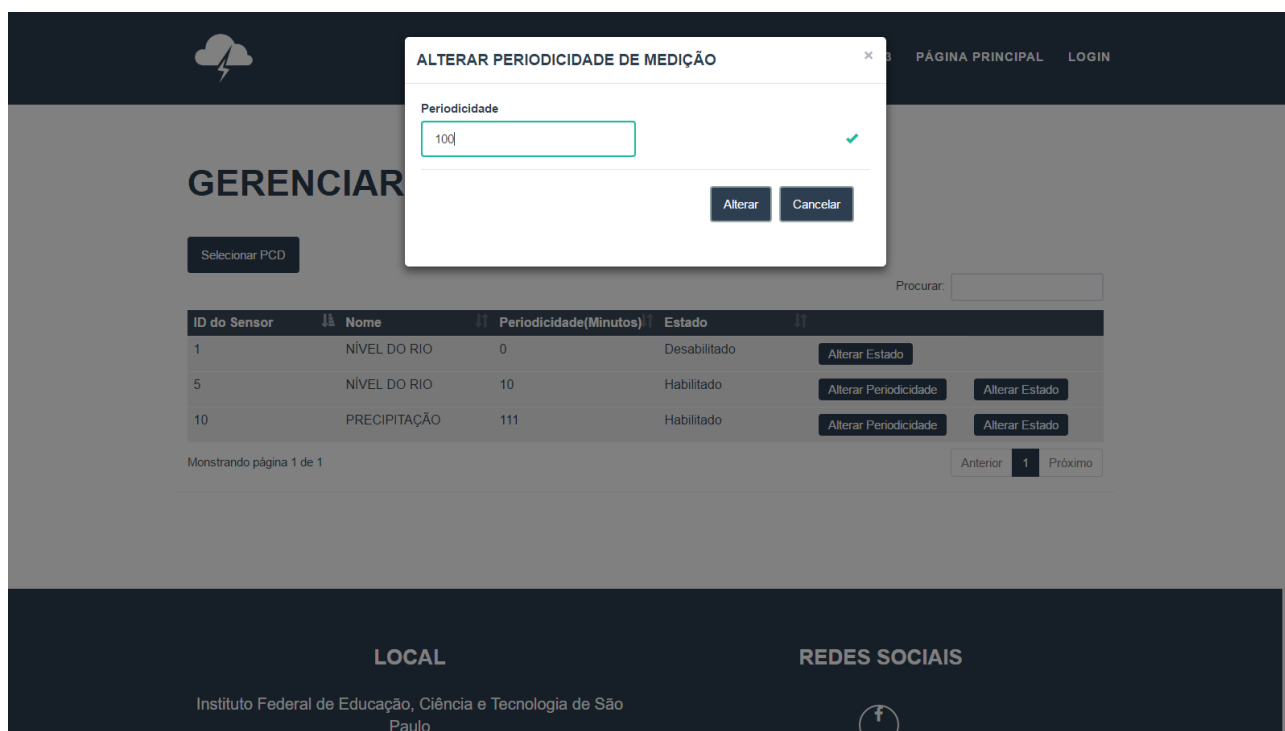


Figura 13 – Formulário para alteração de periodicidade de medição de sensores com validação.

Como é possível notar, ao inserir um dado numérico superior a 0, o campo de texto é contornado em verde e é exibido um ícone ao seu lado, vide Figura 13.

Em contrapartida, ao não atender as restrições impostas no código, o mesmo campo de texto é exibido, porém, contornado em vermelho acompanhado de uma mensagem alertando sobre o que há de errado em sua inserção. Verifica-se este caso na Figura 14, abaixo.

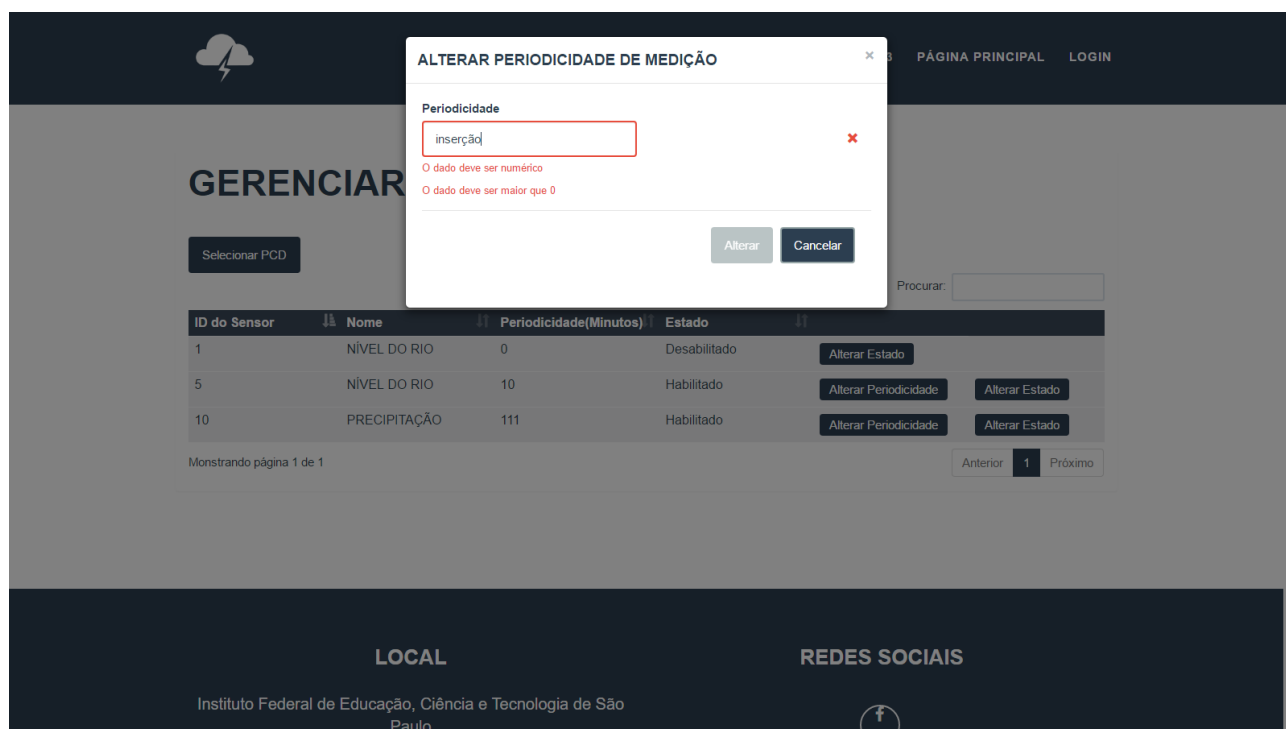
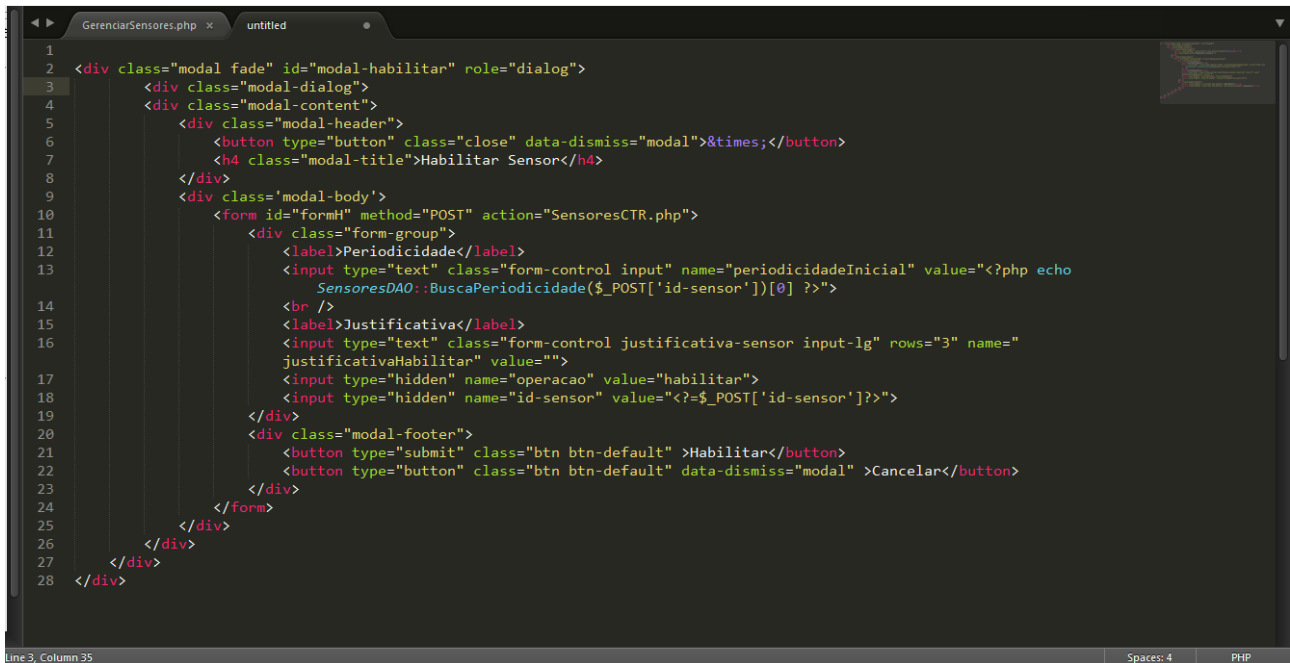


Figura 14 – Inserção que não atende aos requisitos da validação.

A última padronização de estilos foi implementada nas telas modais do projeto com as classes do próprio *Bootstrap*. São as classes, *.modal-dialog*, *.modal-content*, *.modal-header*, *.modal-body* e *.modal-footer*, cada uma responsável por uma parte da tela modal. A seguir, na figura 15, podemos notar o código HTML com as classes CSS para exibir a tela modal.



```
1 <div class="modal fade" id="modal-habilitar" role="dialog">
2   <div class="modal-dialog">
3     <div class="modal-content">
4       <div class="modal-header">
5         <button type="button" class="close" data-dismiss="modal">&times;</button>
6         <h4 class="modal-title">Habilitar Sensor</h4>
7       </div>
8       <div class="modal-body">
9         <form id="formH" method="POST" action="SensoresCTR.php">
10          <div class="form-group">
11            <label>Periodicidade</label>
12            <input type="text" class="form-control" name="periodicidadeInicial" value="<?php echo
13              SensoresDAO::BuscaPeriodicidade($_POST['id-sensor'])[0] ?>">
14          </div>
15          <label>Justificativa</label>
16          <input type="text" class="form-control justificativa-sensor input-lg" rows="3" name="
17            justificativaHabilitar" value="">
18          <input type="hidden" name="operacao" value="habilitar">
19          <input type="hidden" name="id-sensor" value="<?=$_POST['id-sensor']?>">
20        </div>
21        <div class="modal-footer">
22          <button type="submit" class="btn btn-default">Habilitar</button>
23          <button type="button" class="btn btn-default data-dismiss="modal">Cancelar</button>
24        </div>
25      </form>
26    </div>
27  </div>
28 </div>
```

Figura 15 – Código de estrutura e estilo de uma tela modal.

Citadas acima, as classes que compõem uma tela modal segregam a mesma em três principais partes, juntas em uma grande parte que as agrupa. A classe responsável por agrupar as demais é denominada, *.modal-dialog*; *.modal-header* refere-se à sessão de cabeçalho, onde podemos, por exemplo, definir o nome da tela modal; *.modal-body* refere-se ao local onde encontra-se o conteúdo da janela; e *.modal-footer* é o rodapé da janela, onde, geralmente, existem botões para concluir uma operação ou cancelar a janela modal.

Abaixo, é possível notar como a divisão dos elementos citados acima funciona, na Figura 16.

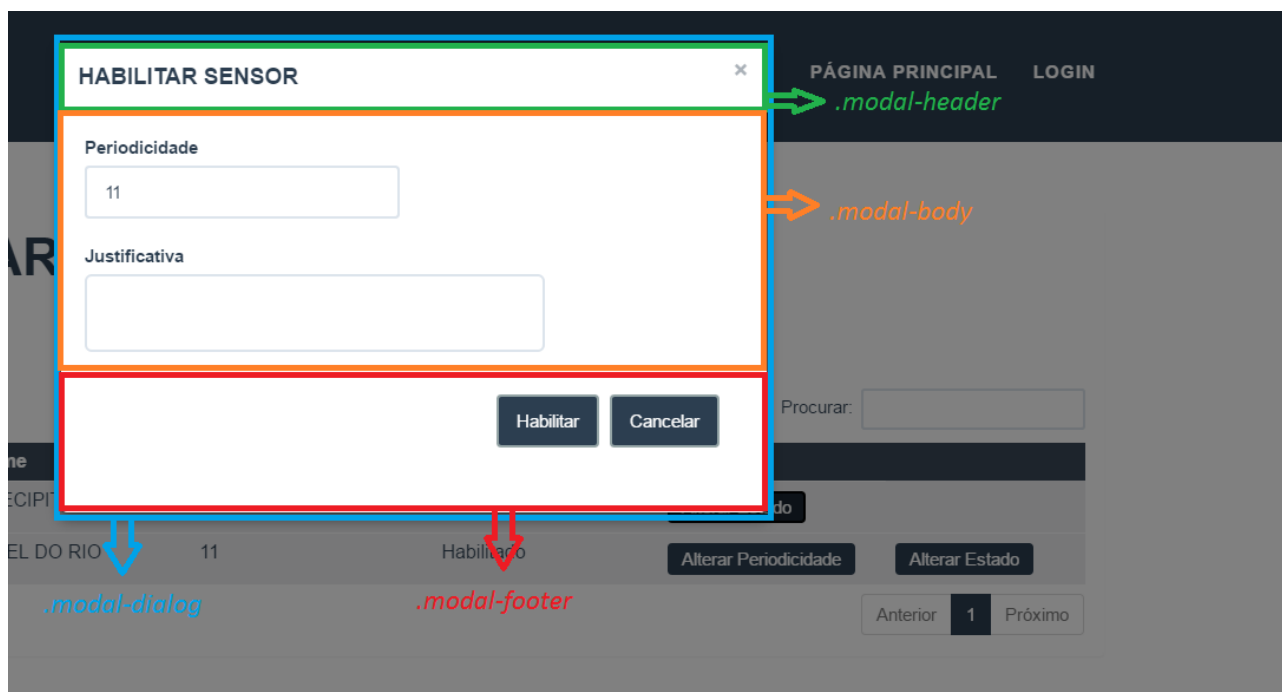


Figura 16 – Janela modal segregada nos elementos que a compõem.

3. Conclusões e Recomendações

Para todo produto, é necessário que seja definida uma identidade visual [9]. Para alcançar isto no desenvolvimento de uma aplicação web, utilizamos de diversas ferramentas, como visto neste relatório, *frameworks*, *plug-ins* e APIs para facilitar a confecção destas interfaces.

Justificamos a criação de interfaces *clean*, utilizando as ferramentas já citadas, uma vez que tal estilo de *design* nos permite alcançar resultados sucintos e amigáveis ao usuário.

Como resultado, então, conclui-se que, em projetos de pequeno e médio porte, algumas vezes até mesmo de grande porte, o uso de ferramentas como os *frameworks* e *plug-ins* trazem aspectos e padrões de *design* valiosos.

Além do *design* moderno e *clean*, as ferramentas utilizadas possibilitaram o desenvolvimento de um sistema com interface responsiva, para que fosse possível acompanhar a tendência atual do mercado de tecnologia da informação, onde o uso de aplicativos e internet nos smartphones é cada vez mais crescente e bate de frente com os convencionais computadores.

Por fim, é recomendado que as ferramentas citadas neste relatório sejam utilizadas para desenvolver interfaces que atendam tendências do mercado atual, unindo estas exigências à praticidade que as ferramentas proporcionam à praticidade que as ferramentas proporcionam.

4. Referências Bibliográficas

- [1] Fabio Lobo, O Que é Template? Disponível em: <<http://fabiolobo.com.br/o-que-e-template.html>>. Acesso em 19 de outubro de 2016.
- [2] Thiago Nascimento, Desenvolvendo com Bootstrap: um Framework front-end que vale a pena! Disponível em: <<http://thiagonasc.com/desenvolvimento-web/developendo-com-bootstrap-3-um-framework-front-end-que-vale-a-pena>>. Acesso em 19 de outubro de 2016.
- [3] SILVA, M. S. Construindo Sites com CSS e (X)HTML. São Paulo: Novatec, 2007.
- [4] Bootstrap em Português – O mais popular *framework front-end* responsivo e focado para dispositivos móveis do mundo. Disponível em: <<http://getbootstrap.com.br/>>. Acesso em 19 de outubro de 2016.
- [5] Otto, Mark (17 de janeiro de 2012). "*Bootstrap in A List Apart No. 342*". Mark Otto.
- [6] Eliane Garone, O que é Design Responsivo? Disponível em <<http://www.princiweb.com.br/blog/front-end/css/o-que-e-design-responsivo.html>>. Acesso em 19 de outubro de 2016.
- [7] Opus Software, Estatísticas de uso de celular no Brasil. Disponível em <<http://www.opus-software.com.br/estatisticas-uso-celular-brasil/>>. Acesso em 19 de outubro de 2016.
- [8] Gabriel Costa, O Que É Bootstrap? Disponível em: <<http://www.tutorialwebdesign.com.br/o-que-e-bootstrap/>>. Acesso em 19 de outubro de 2016.
- [9] Clifton, Rita. O Mundo das Marcas. Lisboa: Editorial, 2004.