

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
ESPECIALIZAÇÃO EM DESENVOLVIMENTO WEB**

RENAN AUGUSTO SGORLOM

PROJETO PREVISÃO DO TEMPO

MONOGRAFIA

LONDRINA

2015

RENAN AUGUSTO SGORLOM

PROJETO PREVISÃO DO TEMPO

Trabalho de Conclusão de Curso apresentado a Diretoria de Pesquisa e Pós-Graduação da Universidade Tecnológica Federal do Paraná do Câmpus Londrina, como requisito parcial a obtenção do grau de especialista em Desenvolvimento Web.

Orientador: Prof. Ms. Thiago Prado Campos

LONDRINA

2015



TERMO DE APROVAÇÃO

Título da Monografia

PROJETO PREVISÃO DO TEMPO

por

Renan Augusto Sgorlom

Esta monografia foi apresentada às 16h30 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.

Prof. Me. Thiago Prado de Campos
(UTFPR)

Prof. Dr. César Augusto Cusin
(FAP-CE)

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 Senso

AGRADECIMENTOS

Agradeço ao meu orientador Prof. Ms. Thiago Prado Campos, pela sabedoria com que me guiou nesta trajetória.

Aos meus colegas de sala.

A Secretaria do Curso, pela cooperação.

Gostaria de deixar registrado também, o meu reconhecimento à minha família, pois acredito que sem o apoio deles seria muito difícil vencer esse desafio.

Enfim, a todos os que por algum motivo contribuíram para a realização desta pesquisa.

RESUMO

SGORLOM, Renan Augusto. **Projeto previsão do tempo**. 2015. 35 páginas. Monografia (Especialização em Desenvolvimento Web) - Universidade Tecnológica Federal do Paraná. Londrina, 2015.

Projeto de desenvolvimento de uma Interface de previsão do tempo para exibir informações sobre o tempo em qualquer dispositivo, inicialmente nas dependências da Universidade Tecnológica Federal do Paraná. O projeto tem como função adquirir informações de previsão do tempo de várias plataformas já existentes como por exemplo Yahoo Weather, OpenWeather e outras, com essas informações disponíveis será feito um refinamento das informações através de uma tabela de equivalência onde os dados serão filtrados para que fiquem uns equivalentes os outros, sendo assim um serviço ininterrupto.

Palavras-chave: Previsão do tempo, Yahoo Weather, API, Animação, SVG, SMIL, JSON.

ABSTRACT

SGORLOM, Renan Augusto. **Weather forecast project**. 2015. 35 pages.
Monograph (Web development especialization) – Federal Technology University of Paraná. Londrina, 2015.

Development project of a weather Interface to display weather information on any device, initially on the premises of Federal University of Parana. The project's function acquire weather information from multiple existing platforms such as Yahoo Weather, OpenWeather and others with this information available will be a refinement of information through an equivalence table where the data will be filtered to make them equivalent each other, thus uninterrupted service.

Key-words: Weather Forecast, Yahoo Weather, API, Animation, SVG, SMIL, JSON.

LISTA DE FIGURAS

Figura 1 Exemplo de tela de tempo para computadores.....	30
Figura 2 Exemplo da tela de tempo para dispositivos móveis.....	31

LISTA DE TABELAS

Tabela 1 Tabela de Equivalencia	28
Tabela 2 Tabela de tradução de ícones	28

SUMÁRIO

1 INTRODUÇÃO	13
1.1 JUSTIFICATIVA.....	13
1.2 OBJETIVO	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 CONDIÇÕES DO TEMPO	15
2.2 RESULTADOS DA PREVISÃO DO TEMPO	15
2.3 MEIOS DE OBTER A PREVISÃO DO TEMPO	16
2.4 TRABALHOS RELACIONADOS.....	16
2.4.1 ÍCONES.....	16
2.4.2 API'S DE PREVISÃO DO TEMPO.....	17
2.5 FERRAMENTAS DISPONÍVEIS	18
2.5.1 SVG	18
2.5.1.1 MANEIRAS DE SE UTILIZAR O SVG	19
2.5.2 SMIL	20
3 DESENVOLVIMENTO.....	22
3.1 DESENVOLVIMENTO DOS ÍCONES E ANIMAÇÕES.....	22
3.1.1 ESTRUTURA SVG	22
3.1.2 ANIMANDO O SVG	25
3.2 OBTENDO INFORMAÇÕES SOBRE O TEMPO.....	26
3.2.1 TABELA DE EQUIVALENCIA.....	26
3.3 DESENVOLVIMENTO DA INTERFACE PARA O USUÁRIO	28
3.3.1 INTERFACES	29
4 TESTES E RESULTADOS.....	32
4.1 ACESSO A PREVISÃO DO TEMPO	32
4.2 TESTES NO LAYOUT	33
4.3 TESTES NA OBTENÇÃO DE INFORMAÇÕES DA API.....	34
5 CONCLUSÃO.....	35
5.1 ALCANCE DE OBJETIVOS	35
5.2 LIMITAÇÕES	35
REFERÊNCIAS.....	37
APÊNDICE A - CÓDIGO-FONTE UTILIZADO PARA PROVER INFORMAÇÕES DO TEMPO USANDO A API DO YAHOO	38

1 INTRODUÇÃO

A previsão do tempo sempre fez parte do dia-a-dia da população, para algumas pessoas é a principal informação para o início de sua semana, para outras é possível que mude totalmente a forma de trabalho. As previsões do tempo podem ainda ajudar na previsão de desastres ou percas nas plantações.

Segundo Ahrens (2006) as previsões do tempo são responsáveis por salvar vidas e propriedades.

Atualmente as pessoas vivem correndo na pressa, e as vezes ficam sem informações básicas para o dia-a-dia como a previsão do tempo, ou simplesmente não tem acesso a essa informação em qualquer lugar.

1.1 JUSTIFICATIVA

Para pessoas comuns mesmo que trabalhem em escritórios ou em centros comerciais caso elas não tenham acesso a informação da previsão do tempo poderia ser desastrosa no caso de uma tempestade de neve atingi-los durante o percurso para seu trabalho ou para casa. Pessoas que tem trabalhos sazonais como um vendedor de sorvete, necessitam saber a previsão do tempo para prever a necessidade de mandar vendedores ambulantes ou até mesmo se compensa abrir as portas para as vendas no dia. Grandes lojas de departamento também fazem uso dessa informação para saber qual tipo de produto colocaram em destaque na venda.

Atualmente existem diversos serviços que disponibilizam essa informação, tanto é a variedade de serviços como a divergências dessa informação, serviços que podem indicar previsões do tempo complicadas para o entendimento básico do usuário final, ou a informação em excesso para tal usuário.

Apesar de importante quem não depende totalmente da previsão do tempo para seu dia pode ser prejudicado por não possuir essa informação, como por exemplo estudantes que tem que percorrer um longo caminho até sua instituição de ensino e dependem de meios públicos de transporte, ou até mesmo estudantes que vão a pé, podem ser pegos de surpresa por uma tempestade ou condições adversas, pelo simples fato de não terem prestado atenção na previsão do tempo pois estava sendo exibida de forma complicada afetou o seu dia.

1.2 OBJETIVO

Este trabalho tem como objetivo desenvolver uma aplicação web que além de adaptável a qualquer dispositivo tenha a flexibilidade de receber informações de previsões do tempo de qualquer empresa que forneça a API de previsão do tempo.

Assim independente das divergências nas variações da previsão do tempo oferecidas pelas diversas empresas o usuário final receberá a informação. Caso o serviço que esteja sendo usado fique indisponível ainda é possível trocar para outro serviço rapidamente, não prejudicando nenhum usuário que dependa desse serviço.

Por ser uma aplicação que pode se adequar a qualquer dispositivo naturalmente se torna uma aplicação leve e portátil, podendo assim auxiliar em diversos locais, como por exemplo televisores espalhados por campus de faculdades, ou até mesmo em futuros displays eletrônicos nos pontos de ônibus, celulares e em qualquer outra situação onde seja possível a conexão com a internet e haja um navegador web.

Com o objetivo de uma mais rápida compreensão da previsão do tempo esse projeto espera alcançar todos os públicos, facilitando assim seu planejamento de todos para o dia, sem imprevistos com a previsão do tempo.

O projeto pretende ajudar tanto em problemas cotidianos simples como levar guarda-chuvas ou não, até prever saídas em meio a chuvas onde há risco de alagamentos.

Além disso é esperado que com base nesse projeto seja desenvolvido vários outros que seguem o mesmo padrão informativo podendo aproveitar até a mesma plataforma, onde desenvolvedores poderão exibir desde horários de ônibus e metrô até identificar e exibir em tempo real rotas de alagamentos ou nevascas que fecharam estradas. Podem ainda através da mesma plataforma informativa avisar no risco de se ir a um lugar desprotegido em determinado período e dar de cara com tempestades adversas como ventanias e tornados.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 CONDIÇÕES DO TEMPO

Tempo é um fenômeno meteorológico ocorrido na terra desde os mais comuns como: chuva, neve, nevoeiro, nuvens, tempestades de areia e ventos. Eventos menos comuns como desastres naturais, tornados, ciclones e tempestades de gelo.

“Previsões do tempo são responsáveis por salvar vidas, salvar propriedades e plantações, e nos dizer o que esperar do meio atmosférico.” (AHRENS, 2006, p 336, tradução nossa).

Para a maioria das pessoas interessa saber informações tais como se está nublado, se irá chover, permitindo assim que as pessoas planejem melhor suas atividades diárias, sem serem surpreendidas com eventos naturais indesejados para a atividade escolhida naquele dia.

“Uma previsão do tempo com chuva forte estendida no verão levaria supervisores de construção a planejar o trabalho sobre um teto protetor, lojas de departamento anunciando guarda-chuvas ao invés de roupas de banho e vendedores de sorvete pegar uns dias de folga já que seus negócios vão estacionar.” (AHRENS, 2006, p 336, tradução nossa).

Apesar da responsabilidade que uma previsão do tempo possui ela não é uma ciência exata, e ocasionalmente pode estar errada, para que a previsão se torne o mais correta possível é necessário que seja verificada frequentemente como é o caso da maioria dos aeroportos que verifica a condição do tempo a cada hora.

2.2 RESULTADOS DA PREVISÃO DO TEMPO

Para se obter a previsão do tempo é necessário um monitoramento de uma grande área que normalmente é efetuado através de satélites, este monitoramento é de vinte e quatro horas por dia, observando as massas de ar, temperatura, umidade, pressão do ar além de velocidade do vento, em seguida esses dados são calculados, dando assim a previsão do tempo com até 90% de precisão na previsão. Para Ahrens (2006, p 342, tradução nossa) “Atualmente os modelos de previsão de tempo, tem uma previsão boa para os próximos 4 ou 6 dias”.

Após analisados se obtém o resultado do tempo, esses dados podem ser obtidos por diferentes empresas, que os interpretaram de diferentes maneiras. O Yahoo Weather nos retorna 48 tipos de tempo diferentes de tempo, enquanto o Climatempo retorna apenas 11 tipos de tempo. Para uma pessoa comum o que altera na verdade é apenas o nome pois para ela céu limpo e um céu sem nuvens seria a mesma coisa, para os estudiosos da área de meteorologia tem outro significado, sendo este um dos problemas que este trabalho pretende resolver.

2.3 MEIOS DE OBTER A PREVISÃO DO TEMPO

As pessoas estão constantemente recebendo informações de previsão do tempo, seja por jornal, televisão, rádio ou internet.

Na internet além dos serviços citados anteriormente existem sites como o OpenWeatherMap¹, esta empresa disponibiliza um serviço detalhado de previsão do tempo, mostrando informações como horário do pôr do sol, força do vento, para qual direção o vento está indo, pressão atmosférica e umidade. Ele oferece também a opção de o usuário ver as mudanças diárias na previsão ou mudanças por hora, além de como a previsão do tempo foi anos atrás para a mesma época do ano. Porém as pessoas não querem perder tempo com informações que elas julgam desnecessárias para suas atividades do dia-a-dia e desejam as informações básicas de se irá fazer sol e poderão ir à praia amanhã por exemplo.

2.4 TRABALHOS RELACIONADOS

2.4.1 ÍCONES

Na internet é possível encontrar outros trabalhos representativos de condições do tempo “Climacons é uma adorável coleção construída de ícones do tempo disponível em PSD, ESP, Fireworks, PNG e formatos SVG.” (WITHCROFT, 2015, tradução nossa). O trabalho de Adam Withcroft é disponível gratuitamente para qualquer um que deseja utilizá-lo.

¹ <http://openweathermap.org/>

Alessio Atzeni também disponibiliza diferentes tipos de representações do tempo em seu trabalho em diferentes formatos como PSD, CSH, EPS, SVG e fontes para desktop e a web. Assim como Adam, Alessio também disponibiliza seu trabalho para ser usado gratuitamente.

Estes autores oferecem uma proposta para o problema da representação visual, que é necessário para que as pessoas identifiquem com facilidade o tempo que vai estar hoje por exemplo. Porém assim como as fontes de informação do tempo, elas têm divergências e precisam ser simplificadas junto com as informações do tempo para que haja a equivalência destas informações.

2.4.2 API'S DE PREVISÃO DO TEMPO

API's são conjuntos de rotinas e padrões de programação para o acesso a um aplicativo específico ou software ou plataforma na web.

As API's encontradas todas têm a mesma função de mostrar a previsão do tempo, porém todas têm variações de quantos tipos de tempo diferentes podem exibir, API's com fontes de informação confiáveis como por exemplo Yahoo Weather trazem a informação necessária. O local onde você estaria exibindo as informações de previsão do tempo seria totalmente dependente apenas do Yahoo Weather caso ele saísse fora do ar por algum tempo indeterminado, as pessoas dependentes desse serviço seriam prejudicadas.

Este projeto visa resolver o problema de incompatibilidade desses dados oferecidos com outras API's como por exemplo a do OpenWeather, seria necessária uma adaptação dos tipos de tempo oferecido por ambas, para que caso seja necessária a troca do serviço de informação da condição do tempo o resultado mostrado para o usuário não seja alterado, mantendo as informações finais simples e objetivas.

Apesar disso essas API's são de total importância pois a empresa responsável por elas recebe dados de satélites, convertem as informações distribuindo assim a previsão do tempo obtida por eles. A maioria das empresas com este tipo de serviço disponibiliza essas informações gratuitamente abrindo espaço para que desenvolvedores formatem como bem entender a previsão do tempo.

2.5 FERRAMENTAS DISPONÍVEIS

O projeto será desenvolvido na plataforma web e usará de ferramentas já disponíveis para que seja concluído.

Ferramentas básicas tais como HTML, CSS, Javascript, JQuery, Illustrator e Photoshop. Além de ferramentas diferentes como o SVG e SMIL.

2.5.1 SVG

SVG (scalable vector graphics), são códigos que renderizados por qualquer leitor SVG (navegadores web, programas específicos como por exemplo Illustrator), se tornam imagens vetoriais.

“SVG que significa gráficos vetoriais escaláveis, é uma aplicação do XML que torna possível representar informações gráficas de uma maneira compacta e portátil”. (EISENBERG, ROYDS, 2014, p. 1).

“SVG é uma linguagem de aplicação do XML, uma linguagem de marcação extensiva. Todos documentos SVG seguem a regras de sintaxe do XML 1.0. ” (WATT, LILLEY, 2002, p. 8, tradução nossa).

Atualmente para se conseguir uma imagem com transparência de uma qualidade superior existem arquivos PNG, porem estes arquivos tem um peso enorme, além de tornarem a navegação lenta para dispositivos que utilizam conexão 3g por exemplo, como é o caso dos celulares.

“Um dos objetivos nos requerimentos originais do documento SVG era que ele fosse disponível em uma grande variação de dispositivos, não apenas nos navegadores convencionais do desktop”. (WATT, LILLEY, 2002, p. 24, tradução nossa).

Ainda assim não é possível manipular o elemento de maneira livre ou fazer animações como é possível com arquivos do flash, ou .GIF, o SVG une as características de diversos elementos e torna possível a animação de uma imagem vetorial e com peso reduzido.

“É a combinação das características que faz o SVG ser potencialmente poderoso e usável para programação ou gráficos.” (WATT, LILLEY, 2002, p. 12, tradução nossa).

Seu peso depende da complexidade do gráfico vetorial gerado, quantas linhas, quantos elementos ele possui, cores e etc, tudo isso influencia no resultado final do peso do SVG.

“O tamanho do arquivo SVG naturalmente depende da complexidade da imagem em vetor que será renderizado.” (WATT, LILLEY, 2002, p. 12, tradução nossa).

O SVG possui ainda diversas vantagens sobre outros elementos gráficos disponíveis atualmente “SVG tem muitas vantagens em potencial sobre os padrões de gráficos existentes.” (WATT, LILLEY, 2002, p. 12, tradução nossa). Como por exemplo é possível criar cores dinâmicas para um único SVG, onde essas cores podem ser manipuladas simplesmente por CSS, existe também a possibilidade do usuário interagir diretamente com o SVG através de javascript, outro recurso é a possibilidade de aplicar zoom a uma imagem SVG sem que a mesma perca qualidade e sem outras requisições no servidor “Imagens em SVG podem receber zoom sem a necessidade de voltar ao servidor e carregar outra versão da imagem” (WATT, LILLEY, 2002, p. 12, tradução nossa).

SVG possui um código aberto, ou seja, ele permite você estudar, modificar e distribuir o software de graça para qualquer finalidade “O código pode ser inspecionado para ajudar no entendimento de como a imagem foi construída, e a opção de salvar o código SVG para o disco também está disponível” (WATT, LILLEY, 2002, p. 19, tradução nossa).

2.5.1.1 MANEIRAS DE SE UTILIZAR O SVG

O SVG é uma imagem, sendo assim pode ser utilizada como tal pelo HTML através da tag .

Mesmo utilizando o SVG com a tag ele não perde suas propriedades vetoriais e pode ser utilizado em quaisquer tamanhos de telas e dispositivos “Imagens em SVG são renderizadas de uma maneira independente da resolução do dispositivo que ele está sendo renderizado. “ (WATT, LILLEY, 2002, p. 24, tradução nossa).

O problema ao utilizar o SVG com a tag do HTML é não é possível assim manipular a sua estrutura XML, trocando cores e adicionando animações por exemplo.

É possível também adicionar o SVG diretamente através do CSS com a propriedade *background-image* “O mais comumente utilizado é a propriedade *background-image*, que coloca a imagem atrás do texto do elemento que está sendo estilizado.” (EISENBERG, ROYDS, 2014, p. 17, tradução nossa).

Mas assim como utilizado pela tag do HTML não é possível manipular os elementos do SVG dessa maneira.

Outra maneira de se utilizar o SVG é adicionando o seu código diretamente no HTML, dando assim total liberdade ao desenvolvedor em manipula-lo da maneira que quiser usufruindo de todos os recursos disponíveis, “No entanto também é possível misturar o código SVG com a marcação HTML ou XML em um único arquivo”. (EISENBERG, ROYDS, 2014, p. 20, tradução nossa).

2.5.2 SMIL

Smil (pronunciado smile e sua sigla significa “Synchronized Multimedia Integration Language”), permite interatividade com elementos áudio visuais “SMIL 2.0 foi desenvolvido para trazer uma apresentação multimídia interativa para a web” (BULTERMAN, 2004, p. 03, tradução nossa), é uma linguagem de fácil aprendizado assim como o HTML, pois muito de suas apresentações são escritas na forma de texto.

Smil é baseado na linguagem XML “É baseada em texto, não proprietário, formato padrão XML, que é tecida da família XML cooperativa de linguagens interdependentes ” (BULTERMAN, 2004, p. 03, tradução nossa).

Seu objetivo ainda é trazer essa função para todos os dispositivos.

“SMIL foca em fazer em multimedia o que o HTML faz com o hiper texto. Trazer para todo tipo de dispositivo de apresentação (PCs, mobiles dispositivos de mão até TVs) com formato de facil autor descritivo que pode ser mostrado em uma diversidade de plataformas” (BULTERMAN, 2004, p. 10, tradução nossa).

"A funcionalidade do SMIL também é integrada em várias outras linguagens" (BULTERMAN, 2004, p. 11, tradução nossa), como por exemplo SVG e XMT (MPEG-4 multimídia).

3 DESENVOLVIMENTO

Para resolver o problema foi desenvolvida uma ferramenta web que é responsável por resgatar informações meteorológicas e mostra-las de forma que o usuário final possa ter um entendimento rápido e fácil do que está acontecendo.

A ferramenta possui como requisitos acesso um dispositivo com acesso à internet, um browser capaz de renderizar SVG, e um display ou monitor para mostrar as informações.

Como linguagem de marcação da ferramenta foi utilizado HTML, para a criação dos ícones que representam a condição do tempo foi utilizado SVG, para animá-los e torna-los adaptáveis a qualquer resolução foi utilizado CSS, para capturar as informações de previsão do tempo foram utilizados objetos JSON.

3.1 DESENVOLVIMENTO DOS ÍCONES E ANIMAÇÕES

3.1.1 ESTRUTURA SVG

Como citado na fundamentação teórica o SVG possui uma estrutura XML, o código SVG pode ser complexo como os gerados automaticamente por ferramentas como o Illustrator ou simples (digitados pelo próprio desenvolvedor em um editor de texto).

Para criar ícones em SVG primeiro precisamos entender a sua estrutura básica:

```
<svg xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink" version="1.1" width="800"
height="800" viewBox="0 0 800 800">
  <g id="background">
    <rect x="-1" fill="#99CCCC" width="800" height="800"/>
    <defs>
      <rect id="SVGID_1_" x="-1" width="800"
height="800"/>
    </defs>
    <clipPath id="SVGID_2_">
      <use xlink:href="#SVGID_1_" />
    </clipPath>
    <polygon clip-path="url(#SVGID_2_)" fill="#7CB5AB"
points="683.2 969.4 978 553.7 471.2 128.7 432.6 146 389.7 503.7 176.4 544.
"/>
  </g>
<g id="objects">
```

```

        <path fill="#FFFFFF" d="M397.8 746.2h0.7v-0.8c0-0.9 0.1-1.6
0.3-2.1 0.2-0.5 0.6-0.9 1.1-1.1 0.5-0.2 1.3-0.4 2.2-0.4 1.6 0 2.5 0.4 2.5
1.2 0 0.3-0.1 0.5-0.3 0.7 ..."/>
    <!-- ... -->
</g>
</svg>

```

A tag <svg> é a que define para o browser que tipo de documento ele está lendo, além disso ela pode carregar outras informações importantes como a versão do svg, suas dimensões e o viewBox (área visível do svg, mais conhecido como canvas size no Illustrator por exemplo).

A tag <g> significa que todos os itens dentro dela fazem parte de um mesmo grupo, aplicando assim configurações exclusivas ou apenas para fins de organização.

A tag <rect> define um retângulo e ainda nela se pode definir a suas dimensões com as propriedades width e height além de fill que serve para preencher o objeto com alguma cor e stroke que troca o contorno do objeto, no exemplo o “x” significa a posição que ele estará.

No caso da tag <defs> permite que objetos sejam definidos para usa-los novamente no futuro.

Já o <clipPath> define que qualquer elemento que ultrapasse seus limites seja oculto.

Assim como o <rect> a tag <polygon> forma elementos básicos nesse caso um polígono o atributo “points” utilizado no exemplo é o caminho percorrido por esse polígono, ou seja, seu formato.

Não necessariamente você precisa definir o formato de um objeto antes de desenhá-lo no SVG, caso você queira fazer qualquer forma pode ser utilizado a tag <path> que através do atributo “d” é designado seu formato, a definição de formatos no SVG por outro lado torna o SVG humanamente legível.

É possível ainda com o SVG criar sprites de imagens, utilizando das tags <g> para agrupar os elementos e para chama-lo no seu HTML é necessária a seguinte estrutura:

```

<svg viewBox="0 0 16 16">
    <use xlink:href="#icone "></use>
</svg>

```

Onde o viewBox deve ser exatamente o mesmo do viewBox do arquivo SVG criado individualmente para o grupo designado, o primeiro número significa sua posição “top”, o segundo sua posição “left”, o terceiro sua largura e o quarto sua altura (note que quando se refere a altura e largura em SVG é sobre seu estado inicial, tendo em mente que são gráficos vetoriais e são sujeitos a mudanças de dimensões sem perder a qualidade).

A tag <use> faz referência ao ID da tag <g> fazendo assim a referência necessária para resgatar o que há dentro deste grupo. Para que seja possível utilizar este método é necessária que toda a estrutura do SVG esteja presente no arquivo HTML utilizado.

SVGs ainda podem conter acessibilidade dando a oportunidade para usuários especiais entender o que está se passando.

```
<svg viewBox=".." role="img" aria-labelledby="Descrição do título">
  <g>
    <title id="title">Titulo do elemento</title>
    <desc id="description">Descrição do element svg</desc>
  </g>
</svg>
```

Onde o atributo “role” indica para o programa que o usuário de necessidades especiais está utilizando que objeto é esse, aria-labelledby indica o título do componente global, <title> dentro do <g> indicando o título daquele grupo de elementos, e a tag <desc> a descrição do item.

Há várias outras tags SVGs para grupos como o <g> por exemplo: <a>, <glyph>, <marker>, <mask>, <missing-glyph>, <pattern>, <switch>, <symbol>.

Além de outras tags para formas básicas, descrições, filtros, fontes, gradiente, elementos gráficos, elementos com fonte de luz, elementos de forma, elementos estruturais, elementos de texto que não iremos abordar neste projeto.

Os ícones utilizados neste projeto foram todos manipulados com a ferramenta da Adobe o Illustrator, sendo assim todo código SVG utilizado neste projeto foi gerado através da interface do programa.

3.1.2 ANIMANDO O SVG

Como citado no desenvolvimento há várias formas de se aplicar animações no SVG, neste projeto foi utilizado o método do CSS para realizar as animações.

Para que o CSS tenha acesso a todo elemento do SVG é necessário que sua estrutura esteja “crua” no html.

```
<svg “...”>
<path class="nuvem" d="M38.999..."></path>
<path class="gota1" d="M17.001..."></path>
    <path class="gota2" d="M24.999..."></path>
    <path class="gota3" d="M32.999..."></path>
</svg>
```

Com este exemplo temos uma nuvem com o total de três pingos de chuva, através das classes atribuídas a cada elemento <path> é possível agora manipulá-los com o CSS.

O projeto consistiu do uso de CSS-animation e CSS-keyframes para realizar as animações, onde o CSS-animation é responsável por definir o tempo que a animação irá durar, a forma que a animação será executada, e qual animação do CSS-keyframes utilizar. O CSS-keyframes é responsável por definir o que fazer em cada estágio da animação como por exemplo em que posição (ou qualquer outro tipo de personalização) o elemento vai estar no início da animação e em qual posição ele estará na metade e/ou no fim da animação.

Exemplo de CSS-keyframes:

```
@-webkit-keyframes anim-pNub{
    0% {
        -webkit-transform: translateX(0px) translateY(0px)rotate(0deg);
    }
    ...
    100% {
        -webkit-transform: translateX(24px) translateY(-
12px)rotate(42deg);
    }
}
```

O resultado final da combinação dessas duas ferramentas é uma imagem escalável e animada.

3.2 OBTENDO INFORMAÇÕES SOBRE O TEMPO

Para resgatar as informações sobre a previsão do tempo foi utilizada a API do Yahoo Weather, utilizando a query de pesquisa:

```
SELECT * FROM weather.forecast WHERE woeid IN (SELECT woeid FROM geo.places(1) WHERE text="londrina")
```

Assim o Yahoo Weather retorna um objeto JSON, onde se pode conferir informações como a força e direção do vento, umidade e pressão da atmosfera, máximas e mínimas para o dia atual e próximos quatro dias além da previsão do tempo.

Através de JavaScript, é possível organizar as informações que o objeto JSON apresenta, sendo possível exibir as informações necessárias.

Para que fosse possível trabalhar com mais de uma fonte de dados foi necessária a criação de uma tabela de equivalências das previsões do tempo que cada fonte fornece.



















3.2.1 TABELA DE EQUIVALENCIA

A tabela de equivalência foi criada com o intuito de refinar e filtrar os diferentes tipos de previsão do tempo que as diferentes fontes fornecem. Para o usuário final e comum não interessa saber qual o exato tipo de chuva dentre os vários tipos que existem que vai acontecer hoje, interessa saber apenas que vai chover para que ele se prepare com um guarda-chuvas e etc.

O refinamento foi criado com base nos principais tipos de tempo que acontecem e que afetam a vida urbana tipos de previsão do tempo menos comuns e desastres naturais que ocorrem com menor frequência foram adicionadas na categoria “tempestades adversas”.

Sempre que uma nova fonte de previsão do tempo for adicionada ela terá que passar pela tabela de equivalência, que é feito assim que é identificado o que cada código que a fonte de previsão do tempo fornece.

Após os dados serem filtrados e ganharem um novo código é possível transformá-los em ícones seguindo uma segunda tabela.

Cod.	Tipos de tempo	Yahoo Weather	Proposta
1	Ensolarado	32, 34, 36 	
2	Tempo limpo (noite)	31, 33 	
3	Nublado	26, 27, 28 	
4	Parcialmente nublado	29, 30, 44 	
5	Chuva	8, 9, 10, 11, 12, 40 	
6	Tempestade (raios)	1, 3, 4, 23, 37, 38, 39, 45, 47, 35 	
7	Neve	13, 14, 16, 25 	
8	Nevasca	15, 41, 42, 43, 46, 8, 5, 6, 7, 18 	
9	Tornado	0, 2 	






			
10	Névoa (neblina)	20, 21, 22 	
11	Tempestades adversas	19, 24 	

Tabela 1 Tabela de Equivalencia

Código	Nome do SVG responsável
1	sol.svg
2	lua.svg
3	nuvem.svg
4	parcialmenteNub.svg
5	chuva.svg
6	tempestade.svg
7	neve.svg
8	nevasca.svg
9	tornado.svg
10	neblina.svg
11	tempestadeAdv.svg

Tabela 2 Tabela de tradução de ícones

Assim conclui-se a organização da distribuição dos ícones necessários para representação gráfica da previsão do tempo independente da fonte utilizada para buscar as informações.

3.3 DESENVOLVIMENTO DA INTERFACE PARA O USUÁRIO

A interface final foi desenvolvida fazendo uso de técnicas de design responsivo, que permite que o layout se adapte a qualquer dispositivo como resultado final.

Além do mais a distribuição dos conteúdos se deve de forma que o usuário possa com apenas um olhar rapidamente identificar as informações de seu interesse, para isso foram utilizados esquemas de cores e ícones animados.

Como um exemplo quando o tempo está “fechado”, ou seja, com previsão de chuva, a linha onde se representa a previsão do tempo do dia atual assume uma cor mais escura representando o tempo, quando o tempo está ensolarado e limpo esta mesma linha assume uma cor mais clara, assim o layout representa a previsão do tempo de duas formas: A forma básica a cor, e uma forma um pouco mais detalhada o ícone.

Este projeto conta com onze tipos diferentes tipos de ícones para representações das previsões do tempo: ensolarado, tempo limpo a noite, nublado, parcialmente nublado, chuvoso, tempestuoso, neve, nevasca, tornados, neblina ou névoa e tempestades adversas.

A interface exibe as seguintes informações:

- Na primeira linha: Dia atual, ícone representação da previsão (chuva, sol e etc), representação escrita deste ícone, temperatura máxima, temperatura mínima, cidade e estado da previsão.
- Nas linhas respectivas informa a data, ícone representação da previsão, representação escrita deste ícone, temperatura máxima e mínima.

A ferramenta utilizada para prototipação da interface foi Adobe Photoshop.

3.3.1 INTERFACES

Na figura 1 é possível visualizar um exemplo de como é interface do projeto exibida em televisores e computadores:

Hoje		Chuva 15°C 10°C	Londrina Paraná
05/05		Tempestade 10°C 5°C	
06/05		Ensolarado 35°C 25°C	

Figura 1 Exemplo de tela de tempo para computadores

Na figura 2 é possível visualizar um exemplo de como é interface do projeto exibida em televisores e computadores.

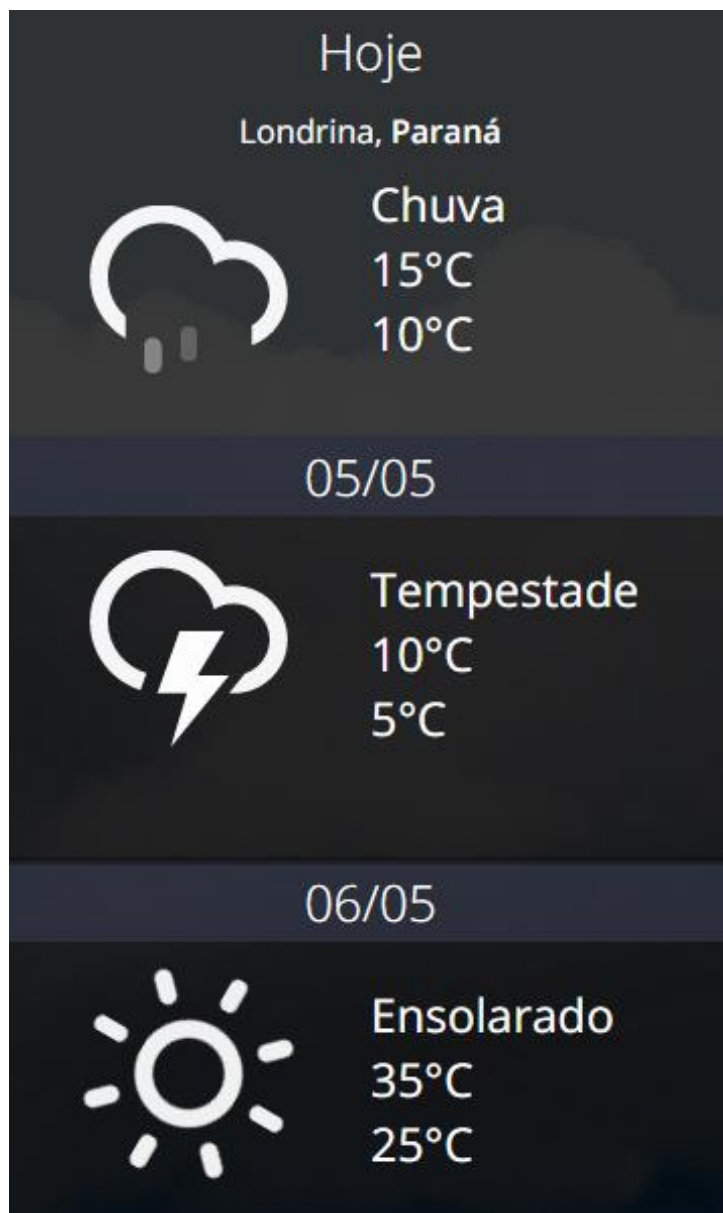


Figura 2 Exemplo da tela de tempo para dispositivos móveis

4 TESTES E RESULTADOS

Foram realizados teste que garantem a resolução dos problemas apresentados neste trabalho testes como: Como a previsão do tempo afeta a vida das pessoas (qual a importância de fato dessa informação na vida urbana), como elas tem acesso a essa informação, se elas têm acesso a essa informação por acaso ou se vão atrás de saber essa informação, e se em algum momento que elas se depararem com essa informação como por exemplo na espera de um ônibus, entrada em prédios e etc. qual o interesse em saber as informações. Além disso foram realizados testes na tabela de equivalência do projeto para verificar se de fato o resultado era o esperado, além de testes no layout para verificação de experiência do usuário.

4.1 ACESSO A PREVISÃO DO TEMPO

É indiscutível que a previsão do tempo faz parte do dia-a-dia das pessoas e que pode afetar em muito o dia das mesmas, “Tempo e clima tornaram-se tanto parte de nossas vidas que a primeira coisa que muitos de nós fazemos é ver a previsão do tempo local ou olhar ela em nossos smartphones. (AHRENS, 2014, p 24, tradução nossa).

Alguns costumam como mencionado assistir pela manhã no noticiário a previsão do tempo, outros no meio do caminho dão uma olhada em seus smartphones, mas e se essa informação fosse disponível de forma que uma pessoa no ponto de ônibus tivesse acesso mesmo sem o seu smartphone, ou até mesmo disponibilizar essa informação em faculdades escolas e vias de carros.

Com um teste de observação feito em aeroportos pode-se notar que um simples telão de previsão do tempo sustentou a atenção de muitos que passaram por ele, indicando assim que essa informação apesar da chance da pessoa ter visualizado anteriormente sempre terá uma atenção das pessoas que passam por ela.

Não há pesquisas e nem como assumir que a informação de previsão do tempo é adquirida de forma involuntária ou ao acaso, porém pode-se afirmar que caso essa informação fosse visível em qualquer lugar como por exemplo telões e

tvS, as pessoas provavelmente notariam, e dependendo da previsão do tempo poderiam alterar os planos para o dia.

Para a resolução final deste problema seria necessária uma conscientização do local onde deveriam ser disponibilizadas estas telas. Algumas regiões do mundo oferecem condições climáticas que caso a pessoa não saiba pode oferecer resultados catastróficos como é o caso de tornados, enchentes e outros desastres que são de extrema importância para a população.

4.2 TESTES NA TABELA DE EQUIVALÊNCIA

A tabela de equivalência foi testada após o recolhimento de informações dos provedores de previsão do tempo. Foram definidos os números e o que cada um deveria representar, sendo assim se uma API de previsão do tempo o código um retornasse ensolarado a tabela deveria enquadrar este código no número que melhor corresponde o termo ensolarado unificando termos como tempo limpo e céu aberto por exemplo. Além disso termos como garoa, chuva, pancadas de chuva também foram unificados em um único código na tabela, afim da maior compatibilidade entre todos os provedores de previsão do tempo.

Além disso após a atribuição inicial na tabela de equivalência foram atribuídos ícones para cada um concedendo assim uma representação visual adequada para cada um deles.

4.2 TESTES NO LAYOUT

O layout foi desenvolvido com o intuito da melhor experiência de usuário possível, traz as informações de forma simplificada e de fácil entendimento. Além disso o layout propõe uma experiência para qualquer dispositivo onde for acessado, seja um televisor ou um smartphone. O layout oferece a mesma experiência para todos os tipos de usuários pois nenhum deles perdem informações.

Inicialmente foram testados se as informações exibidas de fato eram necessárias, como por exemplo normalmente no dia-a-dia não é necessário para a maioria das pessoas saber a força e a posição do vento.

As informações que foram descritas como de maior importância são a situação do tempo (chuvoso, ensolarado e etc.) temperatura máxima e mínima e previsões para os próximos dias.

Além disso a função de cores do layout ajuda o usuário a associar a situação do tempo atual com a da previsão exibida na tela.

4.3 TESTES NA OBTENÇÃO DE INFORMAÇÕES DA API

Ao obter informações da API da Yahoo Weather, foram realizados testes para identificar a veracidade das informações, estes testes consistiam em comparar com outras fontes de previsão do tempo foram ainda realizados testes para comparar o resultado filtrado na tabela de equivalência com o resultado exibido do projeto.

Foram realizados testes para saber quais os tipos de informações que poderiam ser obtidos de cada API, para que se houver no futuro necessidade de adição de novas funções o projeto se adaptar de forma correta.

5 CONCLUSÃO

Neste projeto foi apontado a importância da previsão do tempo para todos, independentemente do meio rural, ou urbano a previsão do tempo pode afetar o seu dia-a-dia, seja no seu planejamento semanal, nas suas vendas, ou no seu meio de locomoção. Foi apontado também que com a correria diária que as pessoas do meio urbano vivem atualmente é de total importância que informações básicas como previsão do tempo seja disponível de forma que mesmo involuntariamente você a obtenha.

5.1 ALCANCE DE OBJETIVOS

A proposta inicial do projeto é a informação da previsão do tempo de maneira fácil, rápida, em qualquer lugar independente da pessoa possuir dispositivos para isso ou não. De forma que as pessoas não sejam pegas desprevenidas por uma chuva, riscos de enchentes ou qualquer coisa do tipo.

O objetivo básico foi alcançado, o projeto consegue se encaixar em qualquer dispositivo com os requerimentos básicos de acesso à internet, um browser e uma tela para mostrar as informações, atualmente vários dispositivos contam com esses requerimentos simples, além de haver a possibilidade de adicioná-los a painéis em pontos de ônibus por exemplo.

5.2 LIMITAÇÕES

O projeto conta com algumas limitações que podem ser melhoradas através de pesquisas.

A primeira limitação se identifica na forma como o projeto identifica a localização que se deve exibir a previsão do tempo que é feita de forma manual, onde o programador deve alterar a linha de código referente a cidade que exibirá a previsão. Essa limitação poderia ser melhorada de forma que o projeto identifica automaticamente a localidade em que está, fazendo assim a previsão do tempo do mesmo.

Outra limitação é que o projeto requer o uso da internet constante para fazer as consultas de previsão do tempo, caso fosse implementado além da internet uma forma de receber as informações seria de grande ajuda.

Além disso o projeto não é flexível a ponto de trocar as informações exibidas, como por exemplo no caso de se aplicar em um ponto de ônibus seria interessante se utilizasse a mesma estrutura e tecnologias para que pudesse alternar a exibição da previsão do tempo e de horários de ônibus, se estão atrasados ou não e etc.

REFERÊNCIAS

AHRENS, C. Donald. **Essentials of Meteorology: An Invitation to the Atmosphere**. Cengage Learning, 2014, 244 p.

ATZENI, Alessio; **Meteocons** ícones de tempo em svg e diversos formatos: <http://www.alessioatzeni.com/meteocons/> ultimo acesso em 02/05/2015;

BARRY, Roger, CHORLEY, Richard, BARRY. **Atmosphere, Weather and Climate**. Routledge 2004, 472 p.

BULTERMAN, Dick C. A., RUTLEDGE, Lloyd. **SMIL 2.0: Interactive Multimedia for Web and Mobile Devices**. Springer Science & Business Media, 2004, 439 p.

EISENBERG, J. David, ROYDS, Amelia Bellamy. **SVG Essentials**. O'Reilly Media, Inc. 2014, 366 p.

OpenWeather API. Disponível em: <<http://openweathermap.org/api>>. Último acesso em 12 de dezembro de 2015.

PEARLMAN, Ellen, HOUSE, Lorien. **SVG for web developers**. Prentice Hall Professional, 2003, 413 p.

Yahoo Weather. Disponível em: <<https://developer.yahoo.com/weather/>>. Último acesso em 12 de dezembro de 2015.

WATT, Andrew, LILLEY, Chris. **SVG Unleashed**. Sams Publishing 2002, 1117 p.

WITHCROFT, Adam; **Climacons** ícones de tempo em svg disponível em: <http://adamwhitcroft.com/climacons/> último acesso em 02/05/2015;

APÊNDICE A - CÓDIGO-FONTE UTILIZADO PARA PROVER INFORMAÇÕES DO TEMPO USANDO A API DO YAHOO

```

var callbackFunction = function(data) {

    console.log('JSON Retorno (Select *): ');
    console.log(data);

    var pais = data.query.results.channel.location.country;
    var estado = data.query.results.channel.location.region;
    var cidade = data.query.results.channel.location.city;
    var codigo = data.query.results.channel.item.condition.code;
    var dataDia = data.query.results.channel.item.condition.date;
    var temperaturaAtual = data.query.results.channel.item.condition.temp;
    var textoAtual = data.query.results.channel.item.condition.text;

    document.getElementById('pais').innerText = pais;
    document.getElementById('estado').innerText = estado;
    document.getElementById('cidade').innerText = cidade;
    document.getElementById('codigo').innerText = codigo;
    document.getElementById('dataDia').innerText = dataDia;
    document.getElementById('temperaturaAtual').innerText =
temperaturaAtual;
    document.getElementById('textoAtual').innerText = textoAtual;

    var previsaoAmanha = data.query.results.channel.item.forecast[1];
    // [0] -> Dia atual
    // [1] -> Amanha
    // [2] -> Depois de amanhã
    // ...
    var previsaoAmanhaCodigo = previsaoAmanha.code;
    var previsaoAmanhaData = previsaoAmanha.date;
    var previsaoAmanhaDia = previsaoAmanha.day;
    var previsaoAmanhaMax = previsaoAmanha.high;
    var previsaoAmanhaMin = previsaoAmanha.low;
    var previsaoAmanhaTexto = previsaoAmanha.text;

    console.log('Previsão Amanhã: ');
    console.log(previsaoAmanha);

    document.getElementById('previsaoAmanhaCodigo').innerText =
previsaoAmanhaCodigo;
    document.getElementById('previsaoAmanhaData').innerText =
previsaoAmanhaData;
    document.getElementById('previsaoAmanhaDia').innerText =
previsaoAmanhaDia;
    document.getElementById('previsaoAmanhaMax').innerText =
previsaoAmanhaMax;
    document.getElementById('previsaoAmanhaMin').innerText =
previsaoAmanhaMin;
    document.getElementById('previsaoAmanhaTexto').innerText =
previsaoAmanhaTexto;

    // depois de amanhã

    var previsaoDepAmanha = data.query.results.channel.item.forecast[2];

```

```
var previsaoDepAmanhaCodigo = previsaoDepAmanha.code;
var previsaoDepAmanhaData = previsaoDepAmanha.date;
var previsaoDepAmanhaDia = previsaoDepAmanha.day;
var previsaoDepAmanhaMax = previsaoDepAmanha.high;
var previsaoDepAmanhaMin = previsaoDepAmanha.low;
var previsaoDepAmanhaTexto = previsaoDepAmanha.text;

console.log('Previsão depois de Amanhã: ');
console.log(previsaoDepAmanha);

document.getElementById('previsaoDepAmanhaCodigo').innerText =
previsaoDepAmanhaCodigo;
document.getElementById('previsaoDepAmanhaData').innerText =
previsaoDepAmanhaData;
document.getElementById('previsaoDepAmanhaDia').innerText =
previsaoDepAmanhaDia;
document.getElementById('previsaoDepAmanhaMax').innerText =
previsaoDepAmanhaMax;
document.getElementById('previsaoDepAmanhaMin').innerText =
previsaoDepAmanhaMin;
document.getElementById('previsaoDepAmanhaTexto').innerText =
previsaoDepAmanhaTexto;
};
```