



Graduação em Ciência da Computação

Thalles Cezar Rodrigues de Lima

**VISBLOCKS: UMA FERRAMENTA WEB PARA DESIGN DE
SISTEMAS DE VISUALIZAÇÃO DE DADOS**

Trabalho de Graduação



Universidade Federal de Pernambuco
secgrad@cin.ufpe.br
www.cin.ufpe.br/~secgrad

RECIFE
2018



Universidade Federal de Pernambuco
Centro de Informática
Graduação em Engenharia da Computação

Thalles Cezar Rodrigues de Lima

**VISBLOCKS: UMA FERRAMENTA WEB PARA DESIGN DE
SISTEMAS DE VISUALIZAÇÃO DE DADOS**

Trabalho apresentado ao Programa de Graduação em Engenharia da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Engenharia da Computação.

Orientador: Nivan Roberto Ferreira Júnior

RECIFE
2018

Agradecimentos

Agradeço primeiramente à minha família, principalmente aos meus pais, pelo apoio à minha escolha do curso de graduação e por me oferecer todas as oportunidades que me fizeram ser capaz de concluir minha formação sem dificuldades. Sem eles eu não alcançaria o que alcancei.

Também quero agradecer a todos os meus professores que, durante toda minha graduação, me fizeram aprender mais e me apaixonar pelos processos de desenvolvimento de software. Especialmente ao meu orientador Nivan, que me fez conhecer a área de visualização de dados e enxergar a beleza que existe em torno dela.

Deixo um agradecimento especial a Hiago Rodrigues que, com paciência e conselhos sábios, me fez conseguir seguir e concluir este projeto da forma mais tranquila possível. Sua ajuda foi essencial para eu seguir por cada etapa do trabalho.

Por fim, agradeço à empresa In Loco que me fez crescer como um bom desenvolvedor em um ambiente de trabalho livre de energia negativa, me fazendo conhecer as tecnologias mais recentes e eficientes em relação ao desenvolvimento web. Além de ser grato por todo o apoio e espaço necessário que meus líderes e colegas me ofereceram para que fosse possível a conclusão deste trabalho.

Resumo

O processo de criação de visualizações de dados frequentemente está associado ao desenvolvimento de sistemas dedicados à visualização de dados específicos. Isso geralmente exige um certo conhecimento técnico em relação a desenvolvimento de software, design e à própria área de visualização de dados. Aspectos que enriquecem uma visualização de dados, como design responsivo e a escolha de elementos gráficos apropriados acabam aumentando ainda mais a complexidade do processo de criação de visualizações.

Este trabalho introduz uma ferramenta que auxilia o design rápido de visualizações de dados em um ambiente web, de acesso livre. A ferramenta permite a leitura e interpretação de dados através de elementos conectados em um modelo de diagrama de fluxo de dados. Neste diagrama é possível inserir módulos de visualizações de diversos tipos, além ser possível a interação com os dados através de ações que refletem no comportamento dos elementos conectados. O sistema é desenvolvido com base em tecnologias recentes de desenvolvimento web e estruturado de forma que possa ser facilmente estendido com novas funcionalidades no futuro.

Palavras-chave: Visualização de dados, Web, Design Responsivo, Código Aberto, React, Diagrama de Fluxo de Dados

Abstract

The process of creating data visualization models is often associated with the development of systems dedicated to the visualization of specific data. This often requires a specific level of technical knowledge regarding software development, design and the field of data visualization. Some aspects that enhance a data visualization, such as responsive design and the choice of proper graphical elements end up increasing the complexity of the process of designing visualizations.

This work introduces a tool that helps quickly designing data visualizations in a freely accessible web environment. The tool allows data to be read and analyzed through a series of connected elements in a data flow diagram model. In this diagram it is possible to add visualization modules of different natures, in addition to allowing the interaction with the data through actions that reflect in the behaviour of the connected elements. The system is developed based on recent web development technologies and structured in such a way that it may be later easily extended with new functionalities.

Keywords: Data Visualization, Web, Responsive Design, Open Source, React, Data Flow Diagram

Lista de Figuras

2.1	Interface da ferramenta VisFlow	12
2.2	Interface da ferramenta Resonant Laboratory	13
2.3	Interface da ferramenta ExPlates	14
2.4	Editor online da ferramenta VegaLite	15
3.1	Interface do VisBlocks	19
3.2	Bloco de dados e seu menu de configurações.	20
3.3	Bloco de visualização de tabela e seu menu de configurações.	21
3.4	Bloco de visualização de gráfico circular e seu menu de configurações.	21
3.5	Bloco de visualização de gráfico de linha e seu menu de configurações.	22
3.6	Bloco de visualização de histograma e seu menu de configurações.	23
3.7	Bloco de visualização de gráfico de dispersão e seu menu de configurações.	24
3.8	Bloco de visualização geoespacial e seu menu de configurações.	25
3.9	Visualização utilizando o Bloco de filtro	26
4.1	Anomalia nas informações de geolocalização dos anúncios de imóveis coletados do Zap Imóveis	28
4.2	Filtragem dos elementos com informações de geolocalização anômalas.	29
4.3	Distribuição de imóveis residenciais e comerciais em um gráfico circular e um mapa	29
4.4	Processo de filtragem de anomalias e seleção dos imóveis mais caros.	30
4.5	Bloco de tabela exibindo o dicionário de variáveis do conjunto de dados de educação à distância	31
4.6	Visualização dos dados de alunos de educação á distância através com um bloco de gráfico circular e um bloco de filtro	32
4.7	Inclusão de um bloco de histograma para visualizar o horário no qual os alunos mais realizaram atividades	32
4.8	Visualização dos dados de alunos de educação á distância incluindo um gráfico de dispersão relacionando o envio de mensagens de alunos a professores e a tutores	33

Lista de Tabelas

2.1	Comparação de ferramentas de construção de visualizações de dados.	16
-----	--	----

Lista de Acrônimos

CSV	<i>Comma Separated Values</i>
TSV	<i>Tab Separated Values</i>
DSV	<i>Delimiter Separated Values</i>
HTML	<i>HyperText Markup Language</i>
IoT	<i>Internet of Things</i>
JSON	<i>JavaScript Object Notation</i>
PNG	<i>Portable Network Graphics</i>
SVG	<i>Scalable Vector Graphics</i>
RSS	<i>Rich Site Summary</i>
XML	<i>Extensible Markup Language</i>

Sumário

1	Introdução	9
1.1	Contribuições	10
1.2	Estrutura do trabalho	10
2	Trabalhos Relacionados	11
2.1	VisFlow	11
2.2	Resonant Laboratory	13
2.3	ExPlates	13
2.4	VegaLite	14
3	VisBlocks	17
3.1	Implementação	17
3.2	VisBlocks	18
3.2.1	Diagrama de fluxo de subconjuntos	19
3.2.2	Blocos	19
3.2.2.1	Bloco de inserção de dados	20
3.2.2.2	Bloco de visualização de Tabela	20
3.2.2.3	Bloco de gráfico circular (<i>Pie Chart</i>)	21
3.2.2.4	Bloco de gráfico de linha (<i>Line Chart</i>)	22
3.2.2.5	Bloco de histograma (<i>Histogram</i>)	22
3.2.2.6	Bloco de gráfico de dispersão (<i>Scatter Plot</i>)	23
3.2.2.7	Bloco de visualização geoespacial (<i>Pin Map</i>)	24
3.2.2.8	Bloco de filtro	25
4	Casos de uso	27
4.1	Analisando anúncios de imóveis em Recife	27
4.2	Analisando dados sobre alunos de educação à distância	31
5	Conclusão	34
5.1	Melhorias e Trabalhos futuros	34
	Referências	36

1

Introdução

Ferramentas de visualização de dados buscam prover formas intuitivas de explorar e analisar dados através de representações gráficas. Essas ferramentas auxiliam na compreensão de informações ao representá-las em um formato de fácil entendimento em comparação a representações puramente textuais, permitindo usuários identificarem padrões e correlações que possam dar suporte a tomadas de decisões.

O número de ferramentas de visualização de dados vem crescendo à medida que o volume e a variedade de dados têm aumentado nos últimos anos, devido a fatores como incorporação de Internet das Coisas, ou *Internet of Things* (IoT), sensores em diversos tipos de ambientes e digitalização de dados físicos. As ferramentas de visualização de dados disponíveis vão desde bibliotecas que auxiliam no desenvolvimento de interfaces especializadas, como Leaflet¹ ou D3.js², até ambientes complexos que abstraem boa parte dos processos de processamento sobre os dados e provêm uma interface intuitiva, como Tableau³ e Qlik⁴. Entretanto, boa parte das ferramentas mais intuitivas exigem um investimento financeiro para serem utilizadas, enquanto as outras exigem um conhecimento técnico a fim de construir visualizações adequadas para problemas específicos. Porém é comum que pessoas sem conhecimento técnico específico de desenvolvimento de visualização de dados precisem acessar indicadores sobre dados de seu interesse e dependam de desenvolvedores para a criação de ferramentas de visualização para que os dados possam ser propriamente analisados. Existe portanto uma necessidade no mercado de ferramentas mais acessíveis a usuários leigos em relação a desenvolvimento de software de visualização que possam diminuir a distância entre estes usuários e a área de visualização de dados.

¹<https://leafletjs.com/>

²<https://d3js.org/>

³<https://www.tableau.com/>

⁴<https://www.qlik.com/us/>

1.1 Contribuições

Este trabalho introduz uma ferramenta em ambiente web, chamada VisBlocks, para criação de visualização de dados de forma intuitiva e rápida. O ambiente de desenvolvimento da ferramenta segue uma estrutura baseada no modelo de fluxo de dados YOURDON; CONSTANTINE (1979) onde cada elemento de visualização é representado por um bloco que pode ser conectado a outros elementos fazendo fluir os dados na representação. Estes blocos podem ser ajustados de forma a personalizar a visualização ao gosto do usuário.

A ferramenta foi desenvolvida utilizando a biblioteca React⁵, voltada para desenvolvimento de aplicações web. A utilização desta ferramenta permite a fácil construção de uma estrutura responsiva, além de bem modularizada, o que permite que a ferramenta seja facilmente estendida com novas funcionalidades futuramente.

1.2 Estrutura do trabalho

A organização deste trabalho segue a estrutura descrita a seguir. O capítulo 2 apresenta trabalhos da literatura com um objetivo semelhante ao trabalho proposto, i.e. trabalhos que facilitam o design de visualização de dados abstraindo parcialmente ou totalmente a necessidade de escrita de códigos para os sistemas. O capítulo 3 descreve o processo de criação do sistema VisBlocks bem como o detalhamento de seus componentes. No capítulo 4 são analisados alguns casos de uso nos quais foi utilizada a ferramenta proposta para construção de visualizações de dados e seu entendimento. Finalmente, no capítulo 5 é apresentada a conclusão e considerações finais do trabalho, bem como a proposta de possíveis trabalhos futuros para a melhoria e extensão da ferramenta.

⁵<https://reactjs.org/>

2

Trabalhos Relacionados

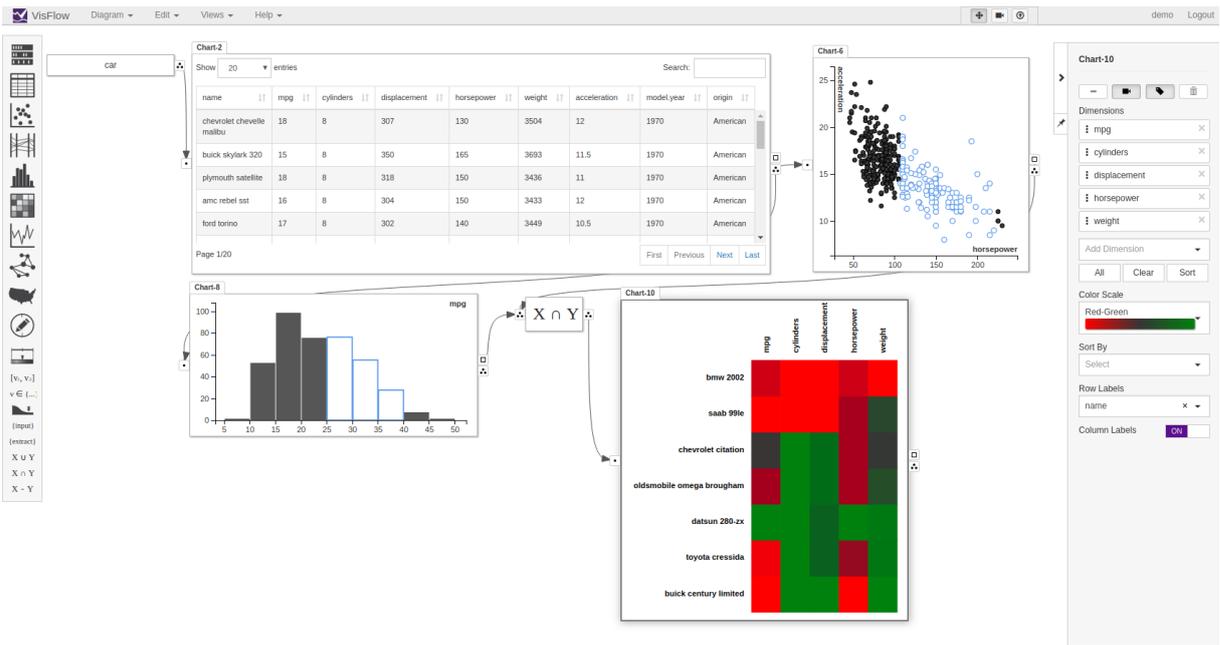
Este capítulo discute algumas ferramentas voltadas a facilitar o design visualizações de dados. Dentre essas ferramentas, algumas provêm uma interface gráfica onde o usuário tem acesso a uma previsualização do resultado e outras são construídas para serem incluídas no código do ambiente onde será apresentada a visualização. Nesta seção são enfatizadas ferramentas que abstraem em determinado nível os detalhes da implementação gráfica das visualizações.

2.1 VisFlow

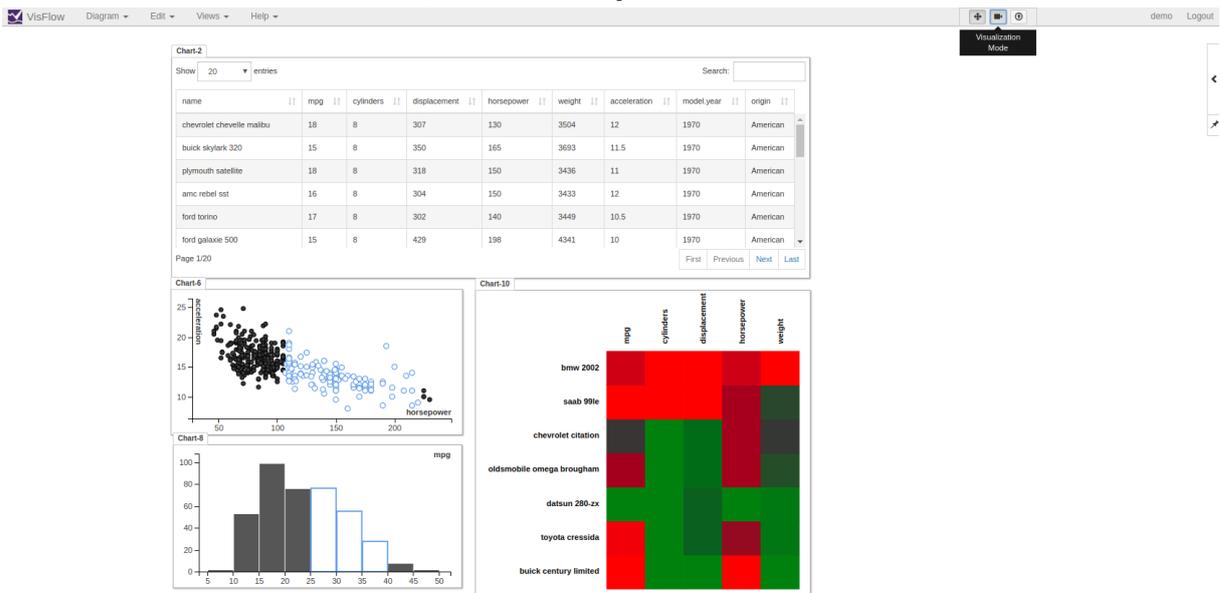
O VisFlow¹ é uma ferramenta web para visualização de dados tabulares baseado em diagramas de fluxo de dados YU; SILVA (2017). A ferramenta provê uma área de trabalho onde o usuário é capaz de inserir e conectar blocos contendo visualizações ou operações sobre os dados inseridos (Figura 2.1a). Cada bloco ativa um painel de controle próprio onde é possível ajustar suas propriedades como dimensões da visualização e suas descrições. Aspectos visuais como cores e tamanhos também podem ser ajustados com um bloco próprio de estilo, que aplica as propriedades ajustadas aos dados. A ferramenta possui oito tipos de visualização diferentes, incluindo um mapa para análise visualização geoespacial.

O ambiente do VisFlow também apresenta um modo de visualização (Figura 2.1b), onde é possível organizar os componentes de visualização em um dashboard sem a conturbação visual dos componentes de operações e injeção de dados ou as conexões entre cada componente. O usuário pode então salvar seu ambiente de visualização em sua conta e compartilhá-la com outros usuários do site. Não está disponível a opção de download da visualização para utilização em outros ambientes.

¹<https://visflow.org/>



(a) Modo de edição do VisFlow



(b) Modo de visualização do VisFlow

Figura 2.1: Interface da ferramenta VisFlow ao ser utilizada para visualizar dados sobre modelos de automóveis.

2.2 Resonant Laboratory

O Resonant Laboratory² é uma aplicação web voltada à criação de uma visualização de um conjunto de dados e compartilhar os resultados BIGELOW; CHOUDHURY; BAUMES (2016). Na ferramenta o usuário pode criar um projeto onde é possível subir um arquivo com um conjunto de dados e a partir deste criar uma das diversas opções de visualizações disponíveis. Porém, diferente das outras ferramentas analisadas, o Resonant Laboratory não suporta diversas visualizações simultâneas interagindo.

O ambiente de trabalho do projeto é dividido em três seções especializadas (Figura 2.2). A seção à esquerda é usada para carregar os dados e visualizar seus elementos. Na seção à direita o usuário pode escolher qual das visualizações será utilizada. A seção central é utilizada para configurar as propriedades da visualização associando-as aos atributos dos elementos do conjunto de dados. Os usuários então podem salvar o projetos nos servidores da aplicação e compartilhá-los com outros usuários ou baixar as visualizações nos formatos PNG, SVG ou HTML e utilizá-las em outros ambientes.

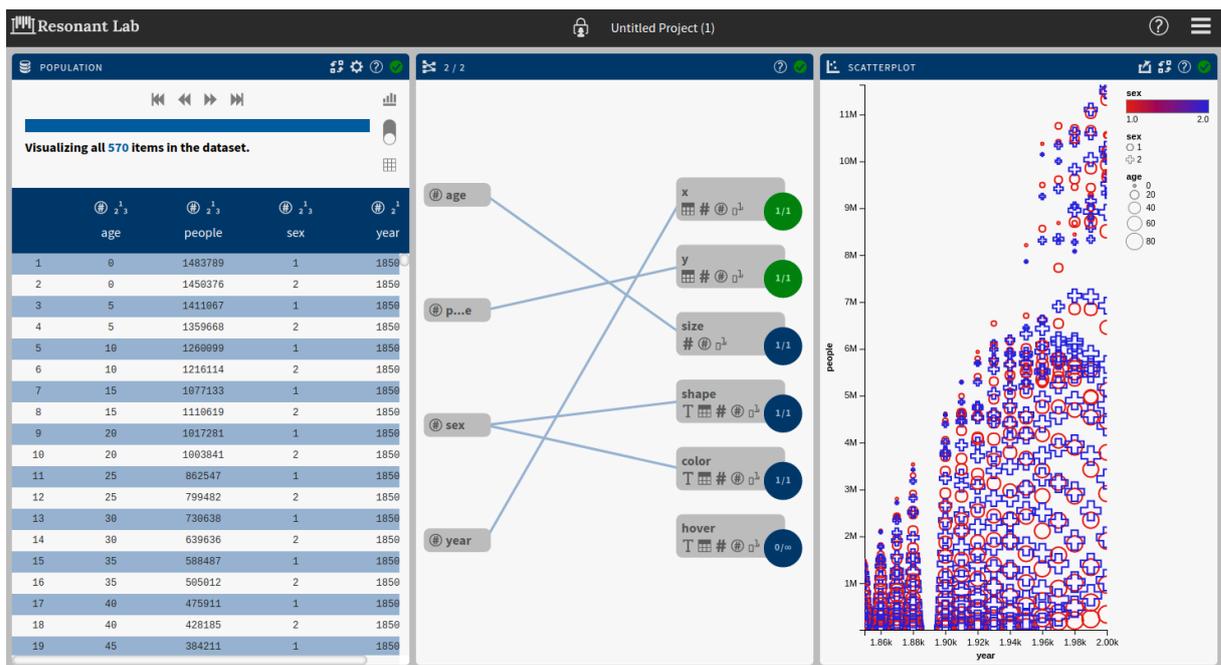


Figura 2.2: Interface do Resonant Laboratory ao ser utilizada para visualizar dados sobre a população de homens e mulheres entre as décadas de 1950 e 2000.

2.3 ExPlates

O ExPlates é uma ferramenta de exploração de dados e construção de visualizações dos mesmos JAVED; ELMQVIST (2013). De forma semelhante ao VisFlow, o ExPlates utiliza um modelo de fluxo de dados para explorar e conectar conjuntos de dados e suas visualizações

²<https://resonantlab.kitware.com>

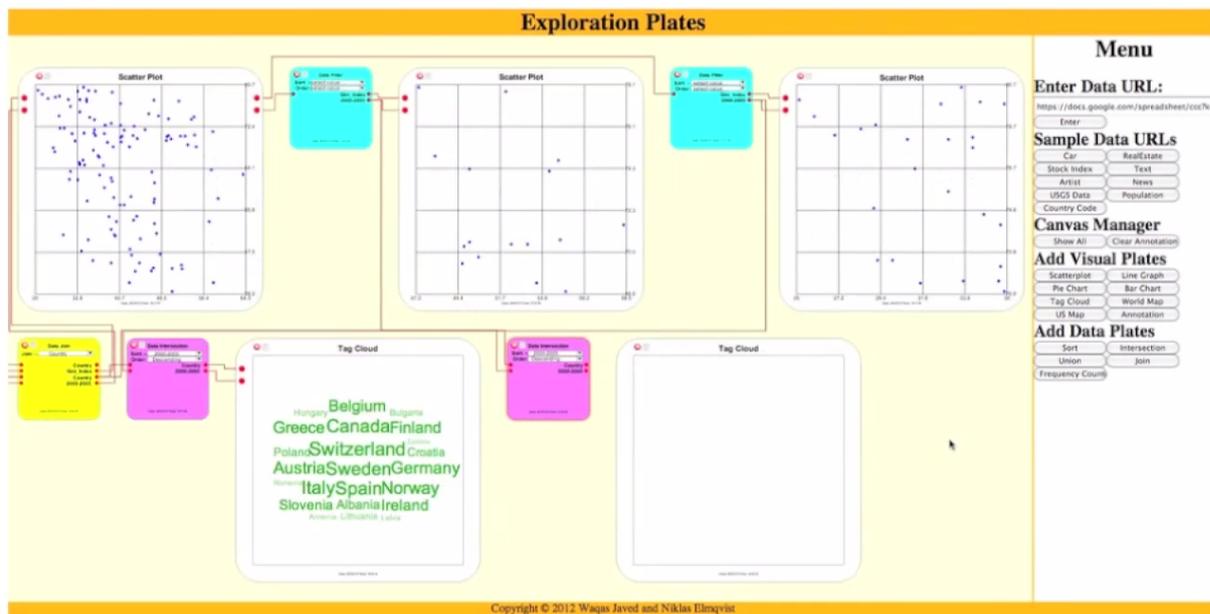


Figura 2.3: Interface do ExPlates exibida no vídeo de demonstração da ferramenta ³

de forma que subconjuntos dos dados explorados possam gerar diferentes visualizações em sequência. Porém, ao invés de as visualizações reagirem a interações com os módulos conectados, uma interação com uma visualização gera um novo módulo com um subconjunto dos dados refletindo a mudança aplicadas - sejam filtros, anotações ou alteração da técnica de visualização JAVED; ELMQVIST (2013). O ambiente de trabalho da ferramenta é composto por uma tela com um menu lateral no qual o usuário pode adicionar dados ou módulos de visualização (Figura 2.3). As operações sobre as visualizações são feitas diretamente nos seus respectivos módulos.

A ferramenta foi apresentada no *Eurographics Conference on Visualization* (EuroVis) 2013, mas atualmente nem seu código ou aplicação não estão disponíveis para acesso.

2.4 VegaLite

O VegaLite⁴ é uma gramática de alto nível que permite a definição de diversos tipos de visualizações de dados interativas SATYANARAYAN et al. (2017). Diferente das outras ferramentas analisadas, o VegaLite não apresenta uma interface para a criação instantânea de visualizações. Ao invés disso, seu objetivo é permitir que o usuário defina suas visualizações em seus códigos de forma descritiva, abstraindo apenas os detalhes da implementação dos elementos gráficos da visualização. Por conta disso, ainda é necessário algum nível de conhecimento em desenvolvimento de software para que se possa aproveitar o potencial desta ferramenta. A gramática permite um alto nível de controle dos elementos das visualizações, permitindo a criação de diversos tipos diferentes de representações, em contraste com alguns poucos pre-definidos

⁴Página para a ferramenta VegaLite: <https://vega.github.io/vega-lite/>

como nas outras ferramentas analisadas anteriormente.

Além de disponibilizar um conjunto de bibliotecas que podem ser incluídos nos projetos do usuário⁵, o VegaLite possui um editor online (Figura 2.4) onde é possível testar a gramática a partir de conjuntos de dados públicos fornecidos como exemplos. Diversas visualizações podem ser criadas simultaneamente de forma customizada e podem interagir entre si. No editor online é possível exportar os exemplos como SVG ou PNG, mas no projeto no qual a gramática é usada o compartilhamento depende de sua implementação.

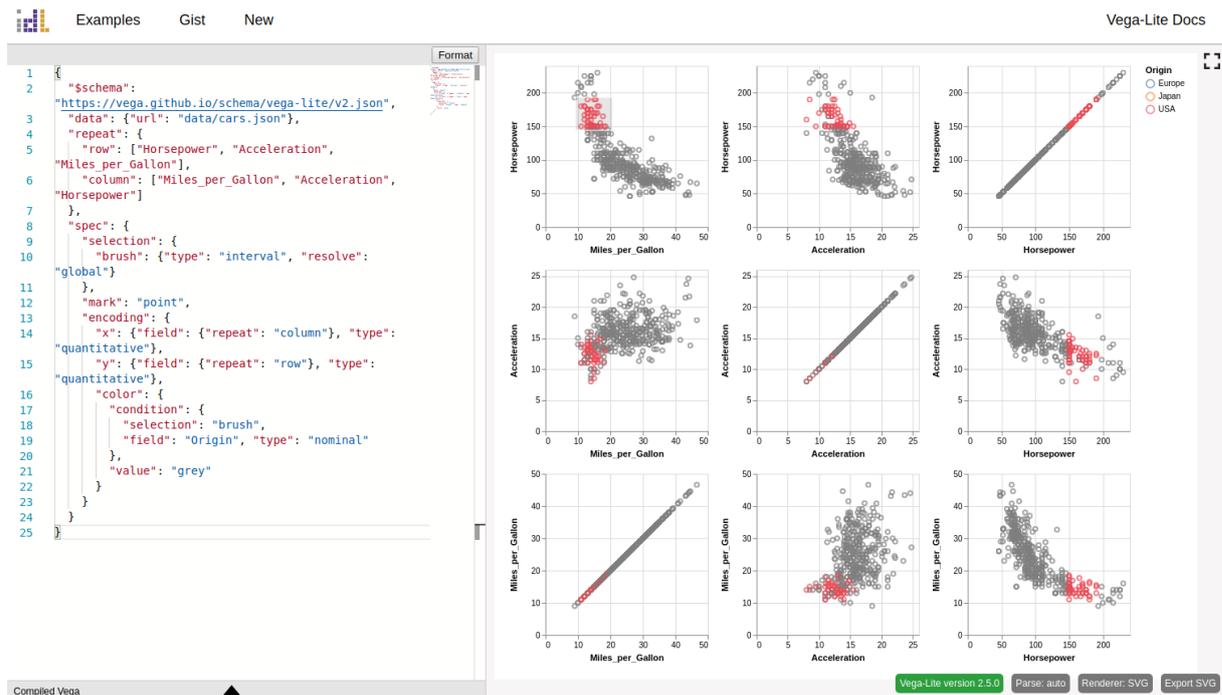


Figura 2.4: Editor online do VegaLite reproduzindo a visualização de dados sobre modelos de automóveis.

Na Tabela 2.1 a seguir são comparadas as quatro ferramentas analisadas em relação aos aspectos que enriquecem e auxiliam a experiência de criação de visualizações de dados.

⁵Repositório com a biblioteca VegaLite: <https://github.com/vega/vega-lite>

Tabela 2.1: Comparação de ferramentas de construção de visualizações de dados.

	VisFlow	Resonant Lab	ExPlates	VegaLite
Interface com módulos de visualização	Sim	Sim	Sim	Não
Formatos de leitura	CSV	CSV, TSV, JSON	Google Docs, Feeds Atom e RSS, XML, CSV	CSV, TSV, DSV, JSON
Múltiplas visualizações	Sim	Não	Sim	Sim
Tipos de visualização	8	14	7	Customizável
Open Source	Sim	Sim	Não	Sim
Compartilhamento	Entre usuários	PNG, SVG, HTML	Não	Não
Diagrama de fluxo de dados	Sim	Não	Sim	Não
Ativo	Sim	Sim	Não	Sim

3

VisBlocks

Este capítulo apresenta o processo de criação e funcionalidades da ferramenta VisBlocks. A ferramenta dispõe uma área de trabalho onde um usuário pode adicionar em um canvas elementos - denominados blocos - que operam e visualizam dados inseridos à sua escolha. É utilizado um modelo de fluxo de dados YOURDON; CONSTANTINE (1979) para explicitar a conexão entre os blocos e o fluxo de dados entre estes. A implementação da ferramenta foi feita de forma incremental, seguindo as etapas descritas na seção 3.1. Seu código fonte está disponível em um repositório no GitHub¹.

O objetivo da ferramenta é permitir o *design* de visualizações de dado de forma simples, rápida mas mantendo o sistema rico em informações. Seu público alvo vai desde usuários especialistas em visualizações de dados até, principalmente, usuários inexperientes e sem noções de desenvolvimento de sistemas de visualização. Portanto a ferramenta foi desenvolvida de forma a ter seu uso simples e intuitivo e suas informações claras para o usuário.

3.1 Implementação

Com o intuito de criar uma ferramenta que facilitasse o design de visualizações de dados abstraindo qualquer necessidade de criação de códigos, foi definido que o VisBlocks seguiria um modelo de fluxo de dados aplicado a blocos de visualizações pré-definidos. Foram definidas 5 tarefas iniciais a serem cumpridas pela ferramenta:

- (T1) Leitura de dados em formatos bem utilizados;
- (T2) Fornecimento de modelos de visualizações relevantes que possam ser aplicados aos dados inseridos;
- (T3) Conexões entre visualizações para fornecer diversas formas de analisar um mesmo dado;
- (T4) Interações com as visualizações de forma a enriquecer o entendimento do dado analisado;

¹<https://github.com/TCezarRod/visblocks>

(T5) Operações sobre as visualizações de forma a possibilitar a customização da análise ao gosto do usuário;

Para criar a ferramenta em ambiente Web com uma interface responsiva foi escolhida a ferramenta React² como base da implementação. React é uma biblioteca baseada em componentes criada e mantida pelo Facebook para o desenvolvimento web em JavaScript. A biblioteca permite o desenvolvimento de aplicações complexas que podem ter dados constantemente atualizados sem a necessidade de atualizações de página desnecessárias AGGARWAL (2018). Isso permite que a ferramenta possa operar sobre os conjuntos de dados e suas representações gráficas de forma eficiente. Além disso, o processo de desenvolvimento modular do React permitiu a criação de componentes básicos reutilizáveis através do código da ferramenta. Isso significa que futuramente novas formas de visualização podem ser incluídas na ferramenta com o mínimo de alteração em sua arquitetura.

3.2 VisBlocks

Nesta seção são descritas a interface e funcionalidades da ferramenta VisBlocks e como ela atende às tarefas listadas na seção anterior. A Figura 3.1 mostra a interface da ferramenta. Esta interface é composta por três componentes principais: o menu de seleção de blocos à esquerda; um canvas no centro da página, onde o usuário pode mover livremente os blocos, além de conectá-los para fazer seguir seus dados; e um menu de controle à direita, variante para cada tipo de bloco selecionado, onde o usuário pode controlar as opções da visualização ativa.

Para atender à tarefa T3, cada bloco disponível pode ser conectado com outro bloco no canvas através do clique no botão de seta na parte superior de cada bloco. Alguns dos blocos de visualização disponibilizam algum tipo de interação, como seleção dos dados mostrados, fazendo com que os blocos seguintes sejam atualizados com o subconjunto de dados selecionado, atendendo à tarefa T4 e permitindo o usuário analisar os dados com um aspecto específico. Na seção a seguir são detalhados cada um dos blocos disponíveis.

²<https://reactjs.org/>



Figura 3.1: Interface do VisBlocks composta por um menu de seleção de blocos (A), um canvas (B) onde os blocos (C) podem ser inseridos e conectados, e o menu de controle dos blocos (D) à direita.

3.2.1 Diagrama de fluxo de subconjuntos

O VisBlocks implementa o modelo de diagrama de fluxo de subconjuntos semelhante ao proposto na ferramenta VisFlow YU; SILVA (2017), no qual é restrito no diagrama a transferência e manipulação de subconjuntos de dados, produzindo um fluxo mais simples em comparação a outras ferramentas com modelos semelhantes (e.g. VTK utilizado no VisTrails BAVOIL et al. (2005)). Yu e Silva sugerem um diagrama onde os nós são definidos por módulos com diferentes portas. Essas portas são utilizadas para conectar os módulos de duas formas, diferenciando os dados transmitidos em uma conexão entre o conjunto completo dos dados no módulo de origem e um subconjunto desses dados selecionado na visualização. No VisBlocks os blocos se comunicam através de apenas um tipo de conexão, que transmite o conjunto completo dos dados no módulo ou o subconjunto selecionado caso haja uma seleção ativa. Uma conexão que transmite o conjunto completo de dados do bloco é redundante no modelo, dado que pode ser substituída por uma conexão partindo de um bloco anterior no fluxo. Manter apenas um tipo de conexão entre os blocos simplifica a usabilidade do modelo. Segundo Roberts, um ambiente com múltiplas visualizações de um dado proveem ao usuário uma melhor compreensão dos dados, mas é encorajado desde que o sistema permita que essas visualizações sejam geradas e gerenciadas facilmente ROBERTS (1998).

3.2.2 Blocos

No VisBlocks estão disponíveis 8 tipos diferentes de blocos que podem ser conectados e ajustados ao gosto do usuário. Um deles é um bloco de dados, responsável por incluir dados no sistema e cumprir a tarefa T1. Seis dos blocos são utilizados para visualização dos dados recebidos como entrada, cumprindo a tarefa T2. Com exceção do bloco de tabela e o bloco

de visualização geoespacial, as visualizações foram em boa parte implementadas utilizando a biblioteca de visualização Victory³ desenvolvida pela Formidable⁴, que provê componentes de visualização construídos com a biblioteca D3⁵. Por fim, existe um bloco de operações que recebe um conjunto de dados e envia aos blocos seguintes um subconjunto destes dados de acordo com suas operações, cumprindo a tarefa T5.

3.2.2.1 Bloco de inserção de dados

O primeiro bloco no menu de seleção é o bloco para inserção de dados (Figura 3.2). Ao escolher esta opção, um diálogo de arquivos é aberto para que o usuário possa escolher um conjunto de dados, nos formatos CSV ou JSON. Ao ser inserido, o bloco possui apenas uma configuração, que é o nome a ser exibido no bloco. Isso ajuda a identificar os dados analisados caso a visualização contenha mais de um bloco de dados. Apesar da simplicidade, este bloco é indispensável para a visualização, pois todos os fluxos devem partir de um deste.

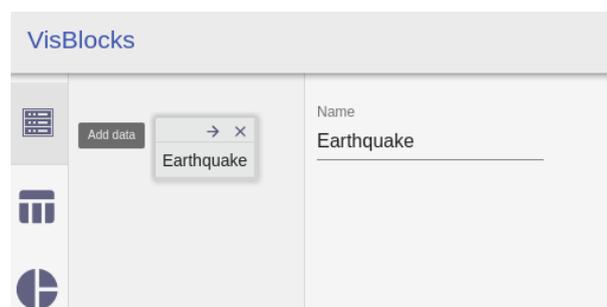


Figura 3.2: Bloco de dados e seu menu de configurações.

3.2.2.2 Bloco de visualização de Tabela

O bloco de Tabela (Figura 3.3) exibe todo o conjunto de dados recebido em uma tabela onde cada linha é um elemento do conjunto de dados e suas colunas representam cada um de seus atributos. Foi utilizado o componente Handsontable⁶ para a implementação do conteúdo interno do bloco. No seu menu de configurações é possível definir, além do nome do bloco, quais colunas são ocultadas da visualização. Isto não afeta os dados que seguem o fluxo. Na visualização, o usuário pode interagir com os dados selecionando linhas da tabela. Ao fazer isto, os blocos seguintes no fluxo de dados operam apenas sobre os dados cujas linhas estão selecionadas. Este bloco é útil para visualizar o conjunto completo dos dados analisados e selecionar elementos específicos do conjunto de dados.

³<https://formidable.com/open-source/victory/>

⁴<https://formidable.com/>

⁵<https://d3js.org/>

⁶<https://handsontable.com/>

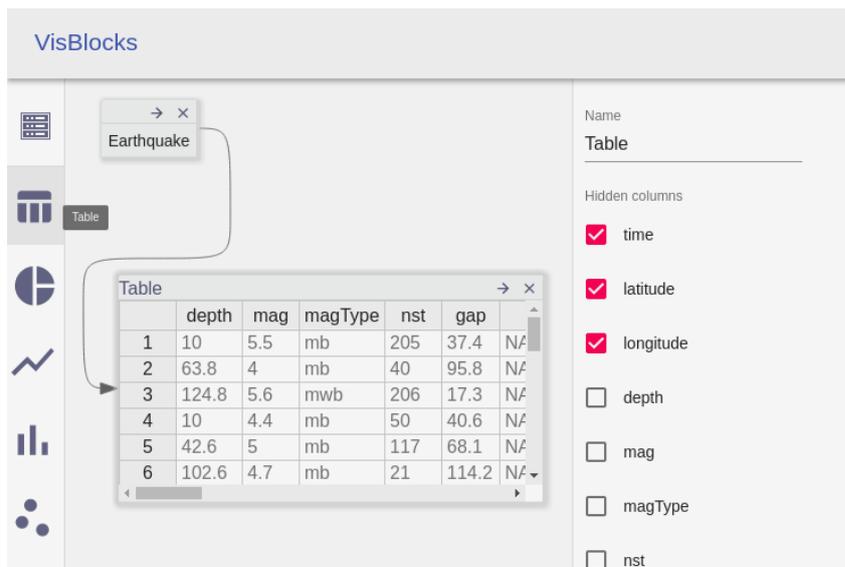


Figura 3.3: Bloco de visualização de tabela e seu menu de configurações.

3.2.2.3 Bloco de gráfico circular (*Pie Chart*)

O bloco de gráfico circular (Figura 3.4) exibe o conjunto de dados separado em setores onde são agrupados de acordo com algum atributo categórico. Cada setor é descrito pelo valor do atributo categórico agrupado e a quantidade de elementos naquela categoria. No menu de controle o usuário pode definir o nome do bloco, o atributo dos dados a ser usado para o agrupamento e as cores de cada setor do gráfico. Este bloco é útil caso o usuário busque a quantidade de elementos em cada ou determinada categoria.

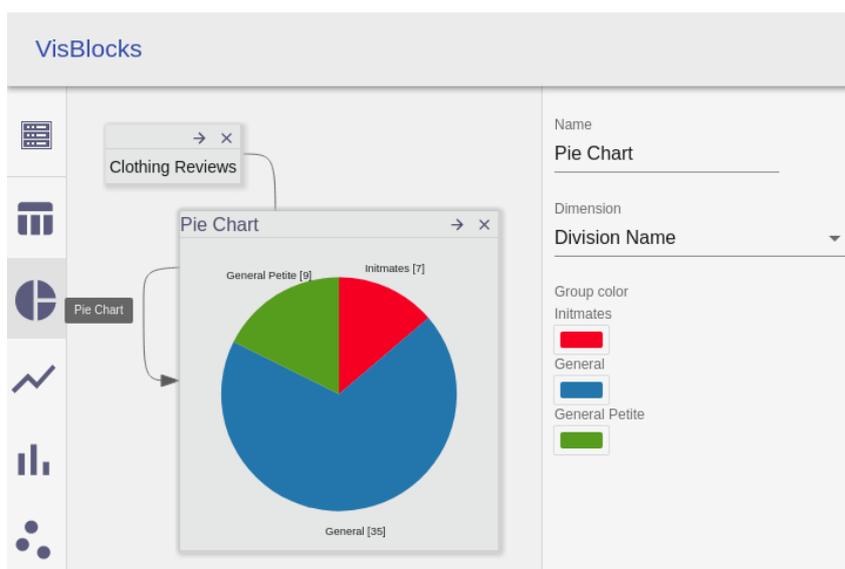


Figura 3.4: Bloco de visualização de gráfico circular e seu menu de configurações.

3.2.2.4 Bloco de gráfico de linha (*Line Chart*)

O bloco de gráfico de linha (Figura 3.5) exibe o conjunto de dados de forma na qual cada elemento do conjunto é representado por um vértice em uma linha representada entre um eixo horizontal (*Domain*) e um eixo vertical (*Range*). Este bloco pode ser utilizado para visualizar alguma propriedade através de uma dimensão contínua com um elemento para cada valor da propriedade, como a variação de um valor ao longo do tempo, por exemplo. No menu de configuração é possível ajustar os atributos utilizados nos dois eixos, além da cor da linha e o nome exibido no bloco.



Figura 3.5: Bloco de visualização de gráfico de linha e seu menu de configurações.

3.2.2.5 Bloco de histograma (*Histogram*)

O bloco de histograma (Figura 3.6) agrupa o conjunto de dados em intervalos iguais de algum atributo numérico e contabiliza os elementos que fazem parte de cada intervalo, exibindo os dados processados em uma coluna para cada intervalo cujas alturas representam a quantidade de elementos. Essas colunas podem ser selecionadas fazendo com que os blocos subsequentes a este processem apenas o subconjunto de dados selecionado. Este bloco pode ser utilizado para contabilizar elementos que estão dentro de uma variação de algum atributo numérico e filtrar apenas determinado intervalo de valores.

No menu de configurações é possível ajustar, além do nome do bloco, qual atributo numérico é utilizado para o agrupamento e quantos intervalos (*Bins*) serão utilizados. Além disso é possível alterar a cor padrão das colunas e a cor que indica que elas estão selecionadas.

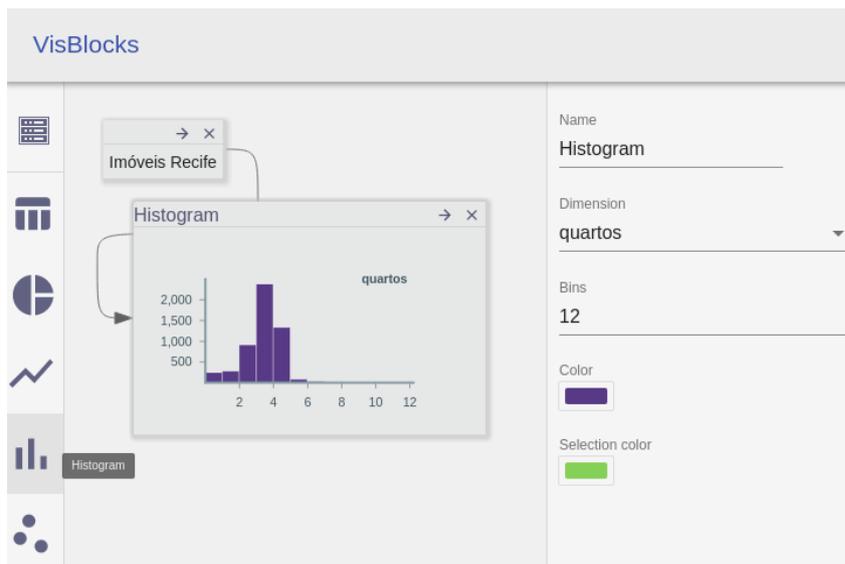


Figura 3.6: Bloco de visualização de histograma e seu menu de configurações.

3.2.2.6 Bloco de gráfico de dispersão (*Scatter Plot*)

O bloco de gráfico de dispersão (Figura 3.7) exibe os elementos do conjunto de dados representados por pontos dispersos entre duas dimensões numéricas, posicionados de acordo com seu valor correspondente para cada uma das dimensões. Esses pontos podem ser selecionados, de forma semelhante ao bloco de histograma, fazendo com que os blocos subsequentes processem apenas os elementos contidos na seleção. Além disso os pontos podem ser agrupados por cores diferentes de acordo com uma terceira dimensão, categórica. Este gráfico pode ser utilizado para avaliar relações entre variáveis do conjunto de dados, o que pode indicar uma relação de causa e efeito.

No menu de controle deste bloco o usuário pode escolher os atributos numéricos utilizados nos eixos vertical e horizontal e o atributo categórico utilizado para agrupar por cores. Além disso, a visualização pode ser customizada alterando o tamanho dos pontos no gráfico, assim como suas cores. É possível também alternar entre usar as cores dos agrupamentos ou pintar todos os pontos de uma cor padrão única.

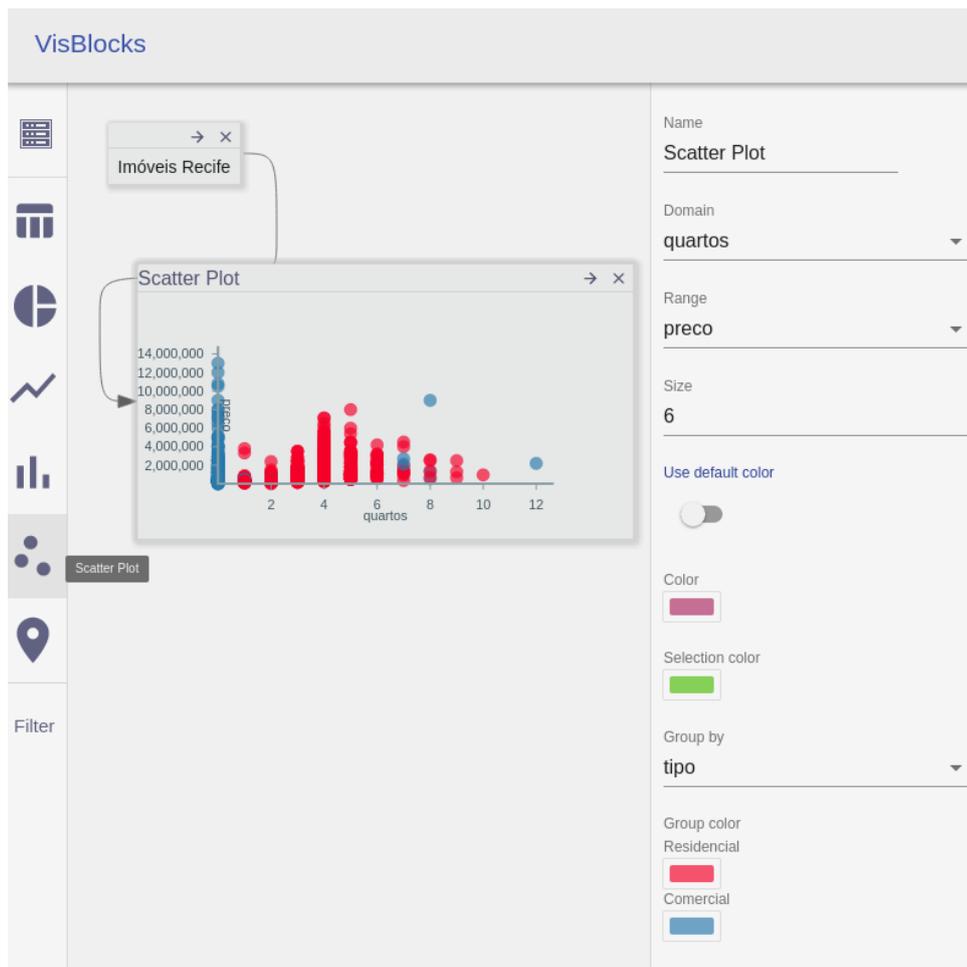


Figura 3.7: Bloco de visualização de gráfico de dispersão e seu menu de configurações.

3.2.2.7 Bloco de visualização geoespacial (*Pin Map*)

O bloco de visualização geoespacial (Figura 3.8) distribui os elementos do conjunto de dados de acordo com suas coordenadas geoespaciais, i.e. latitude e longitude. Para que os elementos sejam exibidos no mapa é necessário que possuam duas dimensões com os valores de suas coordenadas: “latitude” ou “lat” para a latitude e “longitude” ou “lng” para a longitude. Os pontos no mapa também podem ser agrupados de acordo com atributos categóricos, de forma semelhante ao bloco de gráfico de dispersão. Além disso, cada pino no mapa pode ser clicado para exibir o valor de algum de seus atributos em um *popup* acima do pino. Esta visualização é bastante rica, pois permite analisar o espaço físico relacionado aos dados, além de observar a relação de seus atributos com sua distribuição geográfica. Foi utilizada a biblioteca Leaflet⁷ para a implementação da visualização neste bloco.

No menu de controle deste bloco é possível definir o tamanho dos pinos, qual atributo será exibido nos *popups* acima deles, qual atributo será utilizado para seu agrupamento por cores, além da cor específica para cada valor. É possível também alternar entre usar as cores dos agrupamentos ou pintar todos os pinos de uma cor única.

⁷<https://leafletjs.com/>

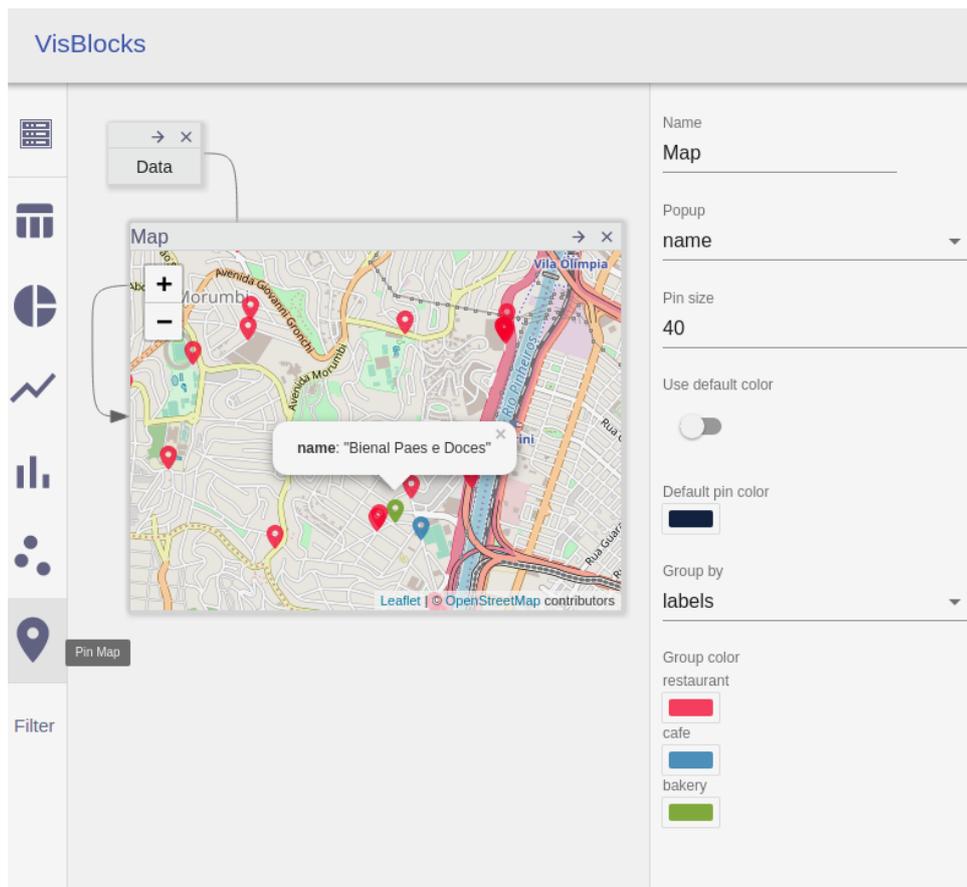


Figura 3.8: Bloco de visualização geoespacial e seu menu de configurações.

3.2.2.8 Bloco de filtro

O bloco de filtro, mostrado na Figura 3.9 é utilizado para selecionar um subconjunto dos dados de acordo com um atributo categórico. Assim como o bloco de dados, seu interior não apresenta nenhuma informação além do nome do bloco. Porém no seu menu de controle é possível escolher qual atributo será utilizado para a filtragem dos dados e quais valores serão selecionados. Os blocos seguintes a este irão operar apenas sobre os elementos do conjunto de dados que apresentam os valores marcados no menu de controle.

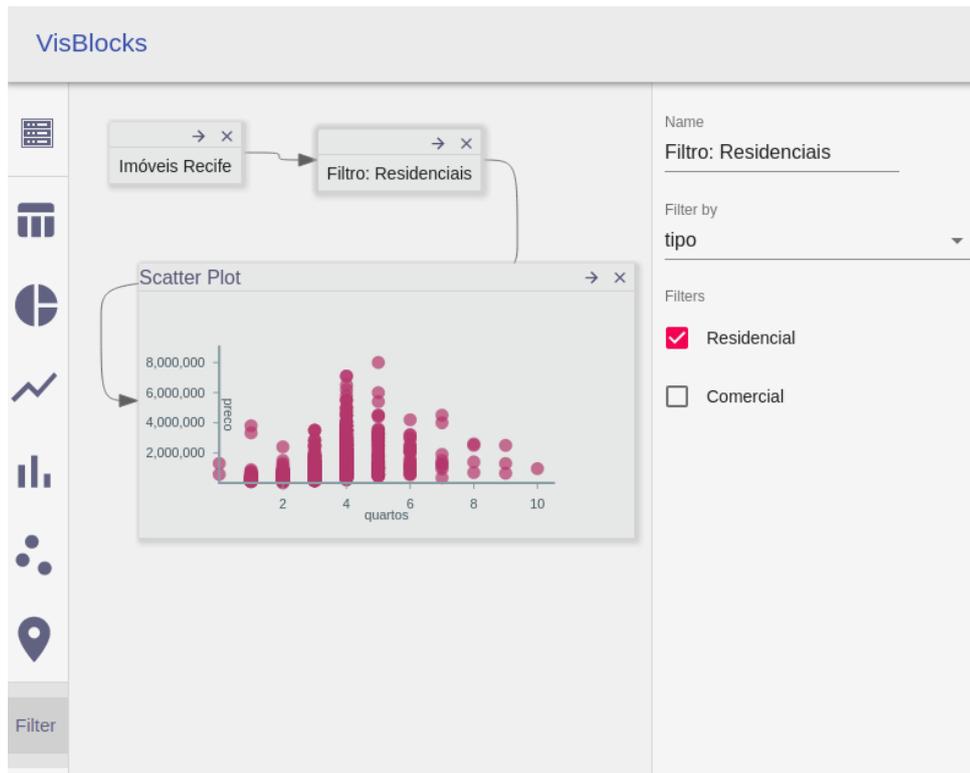


Figura 3.9: Visualização onde é utilizado um bloco de filtro para selecionar um subconjunto de dados com o valor Residencial para o atributo tipo e enviá-los a um gráfico de dispersão.

4

Casos de uso

Este capítulo demonstra a utilidade do VisBlocks analisando dois conjuntos de dados distintos. Assim será demonstrada a versatilidade da ferramenta ao fazer possível tanto a detecção de anomalias nos dados como a identificação de padrões e busca de informações. Esses conjuntos de visualizações podem ser construídos em poucos minutos em contraste com as várias horas que um desenvolvedor levaria para implementar cada visualização em sistemas dedicados a cada um dos conjuntos de dados.

4.1 Analisando anúncios de imóveis em Recife

Nesta análise foi utilizado um conjunto de dados sobre anúncios de imóveis na região de Recife, disponibilizado na plataforma de compartilhamento de dados data.world pelo usuário pauloromeira¹. O Conjunto de dados foi obtido a partir de um *webscraper*² no site de anúncios imobiliários Zap Imóveis³ e utilizado para a população da ferramenta de visualização de dados Sweet Home LIMA FARIAS; FERREIRA (2018). Ao analisar o conjunto de dados utilizando o bloco de mapa é possível notar imediatamente que existem diversos pontos com a informação de geolocalização incorretas. O webscraper deveria ter coletado anúncios de imóveis apenas da região de Recife-PE, porém existem pontos em outros estados do Brasil e até mesmo fora do país (Figura 4.1). É necessário então filtrar apenas os elementos que estão na região desejada.

Observando o mapa, pode-se notar que na longitude da região desejada estão apenas os pontos corretos, então podemos usar este valor para filtrar os elementos. Porém, pode-se não saber qual o intervalo correto da longitude desejada. Se for inserido um bloco de histograma é possível reparar que existe uma quantidade consideravelmente elevada por volta do valor -35.0 (Figura 4.2a), então certamente é naquele intervalo que estão a maioria dos elementos do conjunto de dados, i.e. os dados corretos. Aquele intervalo pode ser selecionado e o bloco do histograma utilizado como entrada para o bloco de mapa, fazendo apenas os elementos

¹<https://data.world/pauloromeira/anuncios-de-imoveis-recife-pe>

²<https://github.com/pauloromeira/realestate-scraper>

³<https://www.zapimoveis.com.br/>

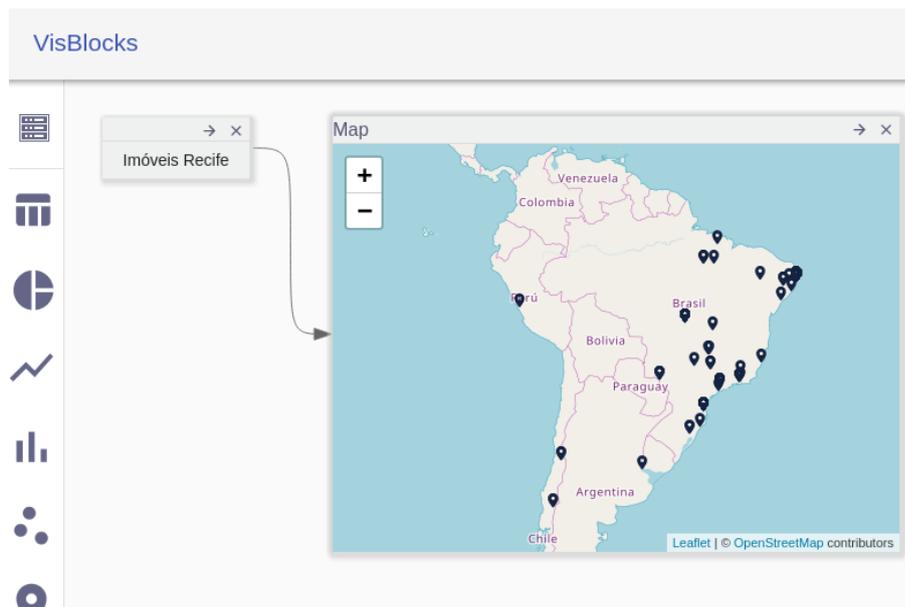
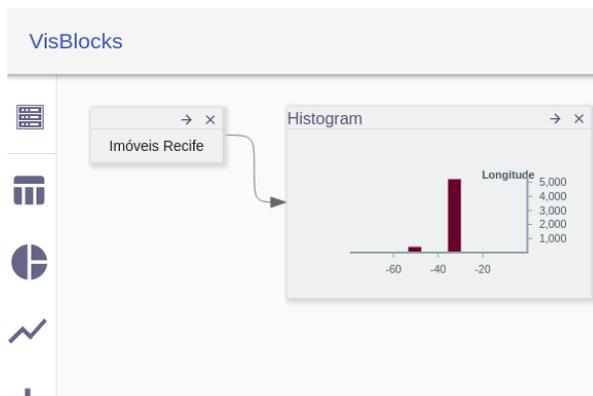


Figura 4.1: Anomalia nas informações de geolocalização dos anúncios de imóveis coletados do Zap Imóveis

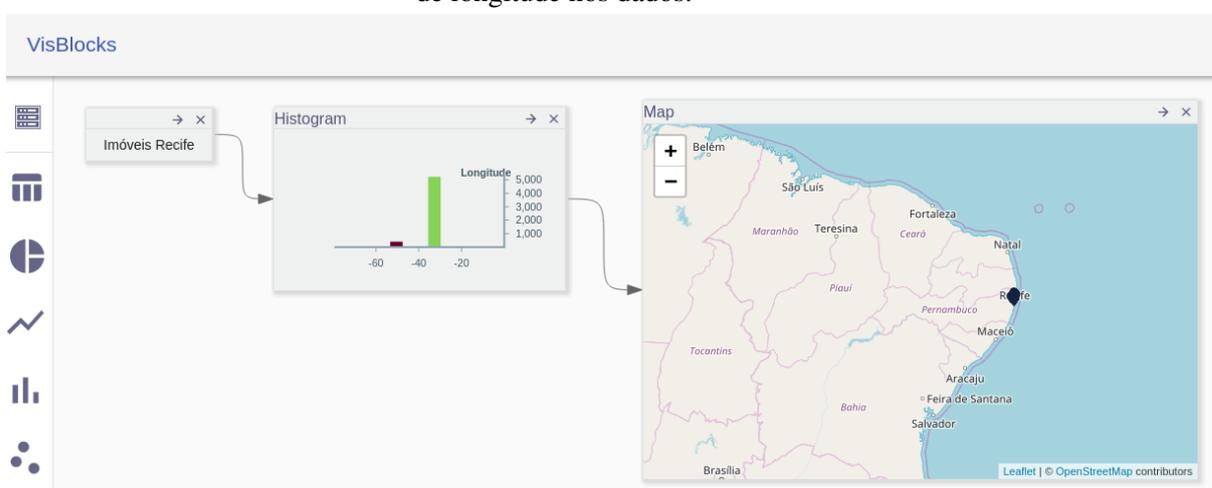
selecionados serem processados. Como esperado, todos os pontos no mapa se encontram na região do Recife (Figura 4.2b).

Os imóveis anunciados podem ser do tipo Comercial ou Residencial, e a distribuição de cada tipo pode ser observada no mapa ao agrupar os elementos por cores, além de sua quantidade exata exibida em um gráfico circular (Figure 4.3). Como pode ser notado, o site de anúncios possui muito mais imóveis residenciais anunciados em comparação a imóveis comerciais. Um bloco de filtro poderia ser utilizado caso se desejasse analisar apenas os elementos residenciais ou comerciais.

Com as coordenadas propriamente filtradas, as outras propriedades dos anúncios podem ser analisadas. Na Figura 4.4a, o gráfico circular é descartado e é utilizado um gráfico de dispersão para verificar a relação entre o preço e a área útil dos imóveis. Pode-se observar que alguns pontos fogem do padrão da maioria com um preço muito alto para pouca área útil ou vice-versa. Pode ser criado então um segundo gráfico de dispersão com apenas os pontos que estão na mesma região do primeiro gráfico, deixando a visualização mais compreensível. Neste segundo gráfico podem ser buscados diversos tipos de anúncios ao gosto do usuário. Conectá-lo a um bloco de mapa ou um de tabela fornece informações suficientes para buscar anúncios específicos. Por exemplo, o usuário pode observar a localização dos apartamentos mais baratos com quatro quartos ou observar que os anúncios mais caros estão concentrados na zona norte e na orla da zona sul da cidade, como é mostrado na Figura 4.4b.



(a) Utilização de um histograma para mostrar um pico em um intervalo de longitude nos dados.



(b) Seleção dos elementos no pico do histograma resulta na filtragem dos elementos na região de Recife.

Figura 4.2: Filtragem dos elementos com informações de geolocalização anômalas.

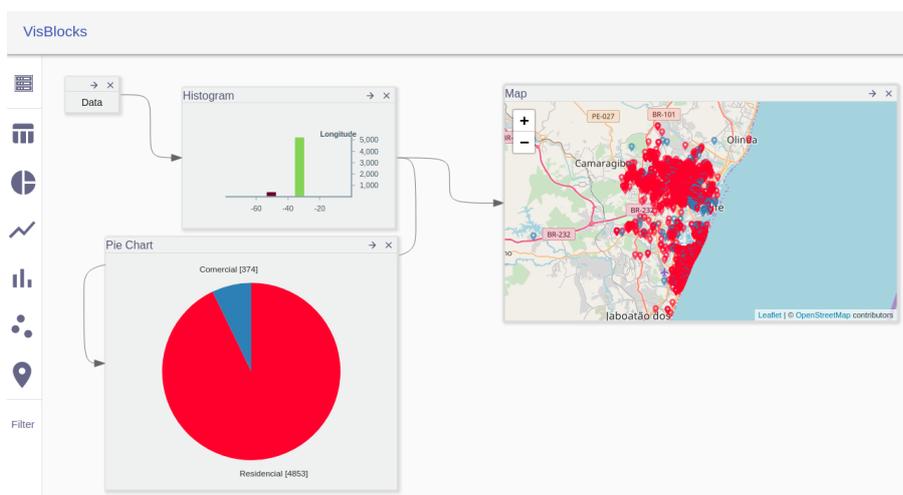
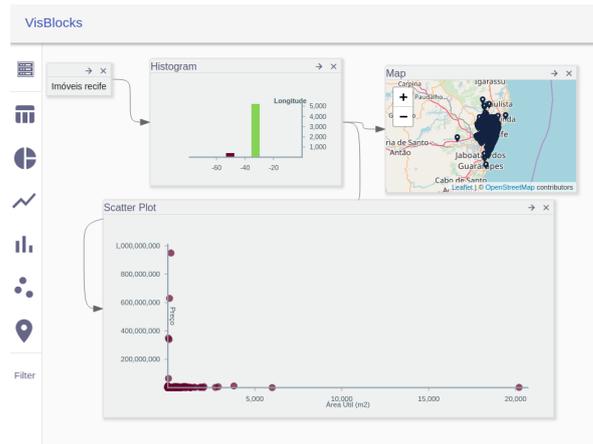
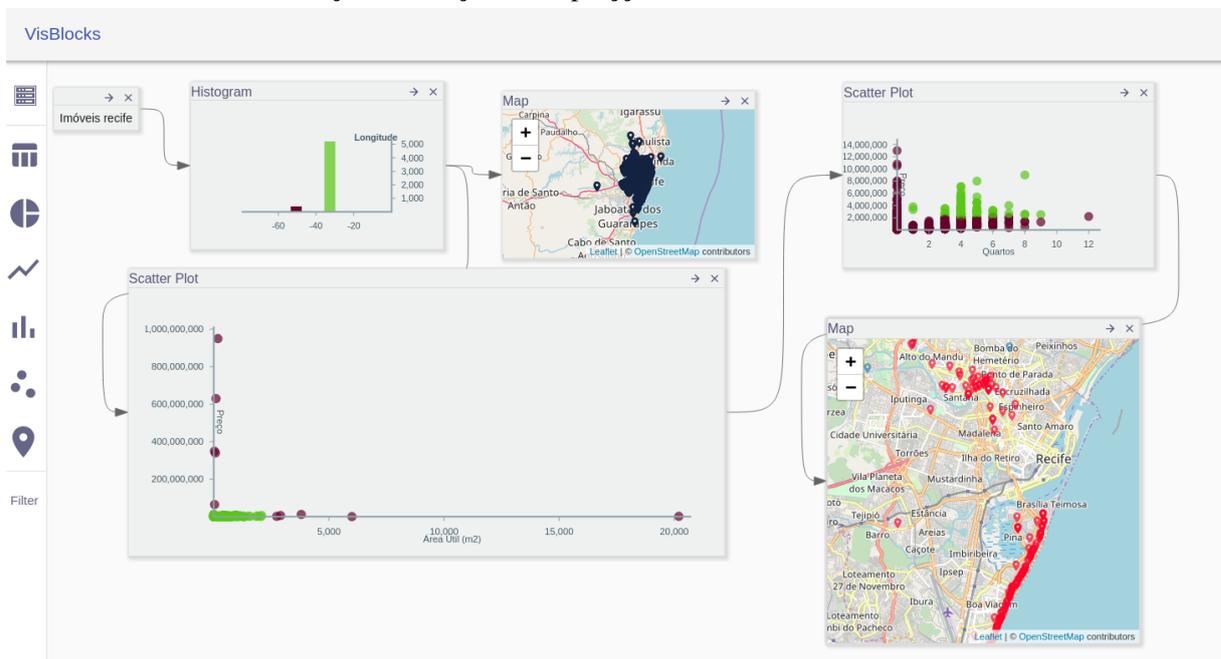


Figura 4.3: Distribuição de imóveis residenciais e comerciais em um gráfico circular e um mapa



(a) Visualização da relação entre preço e área útil dos imóveis.



(b) Filtragem dos imóveis mais caros dentro dos que não fogem do padrão de preço e sua representação em um bloco de mapa.

Figura 4.4: Processo de filtragem de anomalias e seleção dos imóveis mais caros.

4.2 Analisando dados sobre alunos de educação à distância

Neste caso foi utilizada um conjunto de dados com informações sobre a interação de alunos ao portal de educação a distância da Universidade de Pernambuco (UPE). Os atributos deste conjunto de dados estão identificados por variáveis descritas em um dicionário em um arquivo separado. Para facilitar o entendimento das informações analisadas, é útil manter um bloco de tabela com as descrições das variáveis a partir do dicionário de dados (Figura 4.5).

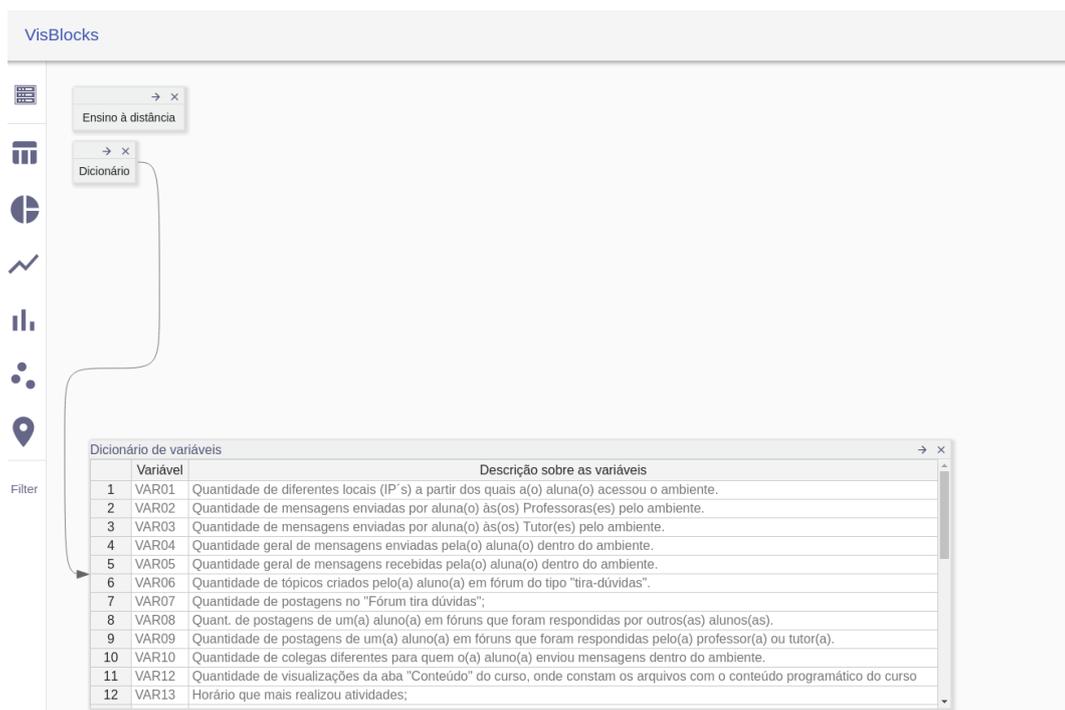


Figura 4.5: Bloco de tabela exibindo o dicionário de variáveis do conjunto de dados de educação à distância

Alunos de quatro cursos diferentes estão descritos neste conjunto de dados, como pode ser observado no gráfico circular da Figura 4.6. Será utilizado um bloco de filtro para filtrar alunos dos cursos de Administração e Letras apenas.

Segundo o dicionário de variáveis, o atributo VAR13 indica o horário que o aluno realizou mais atividades. Utilizando um bloco de histograma com 24 intervalos, mostrado na Figura 4.7, é possível observar que os horários que os alunos mais realizam suas atividades são por volta das 11h e por volta das 20h. O grande volume anômalo no valor 0 muito provavelmente indica que estes alunos estão sem informação de tal variável.

Em seguida, um diagrama de dispersão pode ser utilizado para analisar a relação entre os atributos VAR02 e VAR03, que indicam a quantidade de mensagens enviadas pelos alunos aos professores e aos tutores respectivamente (Figura 4.8). O gráfico indica que para a maioria dos alunos, com exceção de alguns fora do padrão, o auxílio solicitado ao professor é inversamente proporcional ao auxílio solicitado aos tutores. Isso pode indicar o auxílio dos tutores é capaz de suprir as necessidades dos alunos no curso e reduzir a carga de mensagens dos alunos para os

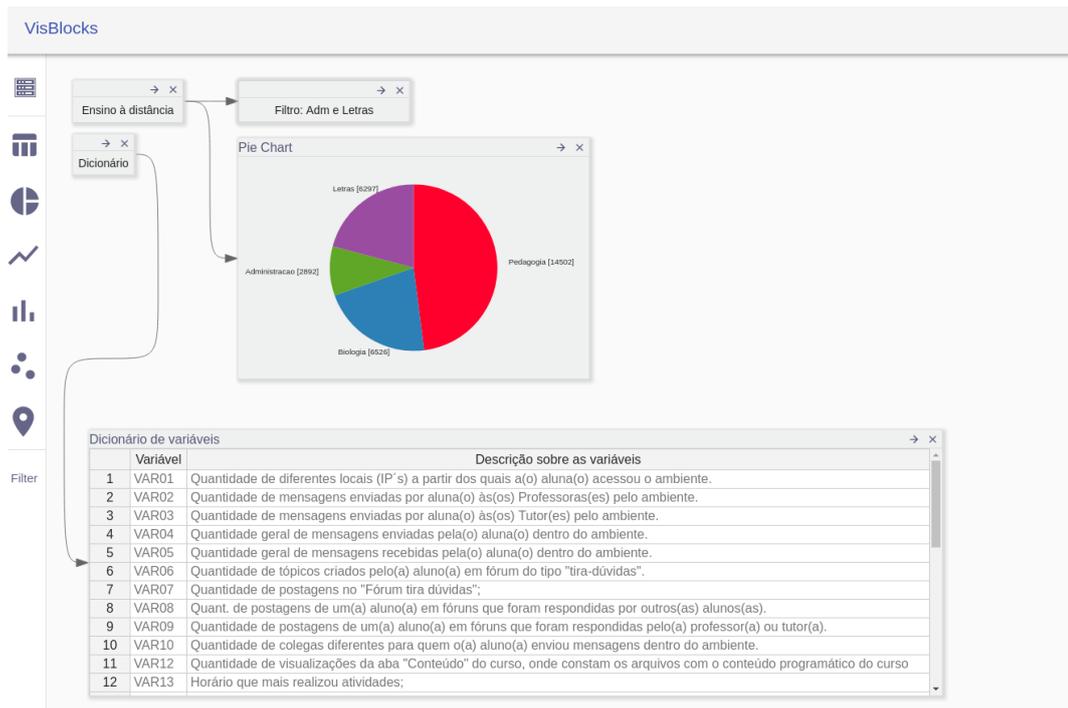


Figura 4.6: Visualização dos dados de alunos de educação á distância através com um bloco de gráfico circular e um bloco de filtro



Figura 4.7: Inclusão de um bloco de histograma para visualizar o horário no qual os alunos mais realizaram atividades

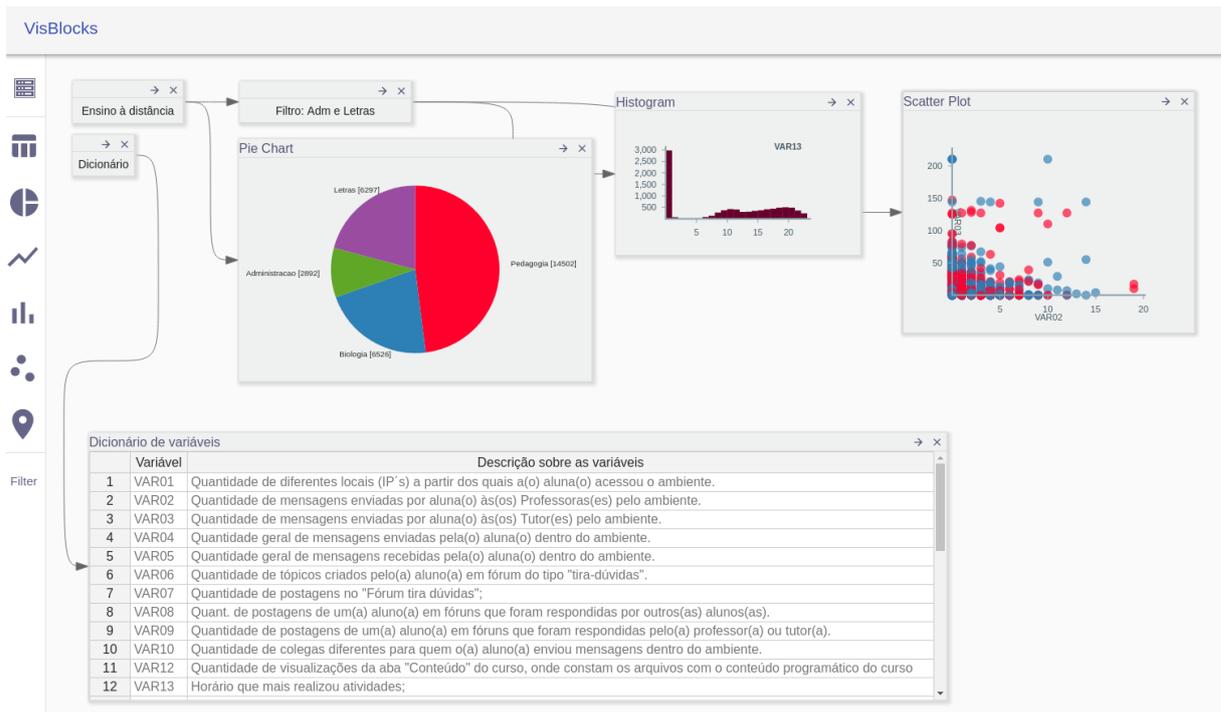


Figura 4.8: Visualização dos dados de alunos de educação à distância incluindo um gráfico de dispersão relacionando o envio de mensagens de alunos a professores e a tutores

professores. Além disso, com os pontos agrupados por curso, é possível notar que mais alunos do curso de Letras enviam mais mensagens aos professores do que os alunos de Administração.

5

Conclusão

Este trabalho introduziu uma ferramenta web para design de visualização de dados chamada VisBlocks. Foram utilizadas tecnologias atuais para desenvolvimento de aplicações web responsivas, como React e bibliotecas baseadas em D3. A ferramenta dispensa instalações e pode ser acessada a qualquer momento, possibilitando a criação de visualizações de forma imediata.

Utilizando as visualizações disponibilizadas pelo Visblocks foram feitos casos de estudo sobre dois conjuntos de dados. No primeiro foi possível identificar os aspectos e distribuição de imóveis divulgados em uma plataforma de anúncios de imóveis. O segundo caso de estudo mostrou aspectos do uso de uma plataforma de ensino a distância por seus alunos.

Dentre os trabalhos relacionados, a ferramenta VisFlow é a qual o VisBlocks mais se aproxima, pois também possibilita a criação de visualizações de dados através de um modelo de fluxo de dados. Entretanto, diferente do VisFlow, a ferramenta proposta foi desenvolvida de forma que seu código possa ser facilmente estendido com novas visualizações, pois a escolha da biblioteca React como base facilita a modularização dos componentes e é de fácil aprendizado. Dessa forma a ferramenta está pronta para evoluir e receber melhorias no futuro. A próxima seção descreve sugestões de algumas melhorias e trabalhos futuros para a ferramenta.

5.1 Melhorias e Trabalhos futuros

O VisBlocks atualmente oferece apenas algumas das formas de visualização de dados mais simples e conhecidas. Novos tipos de visualização estão sempre surgindo e existe espaço e suporte na ferramenta para inclusão de diversos tipos diferentes de visualização. Alguns mais simples que podem ser criados são diagramas de coordenadas paralelas, mapas de calor ou calendários, por exemplo. Outras visualizações mais complexas, como algumas das exibidas na galeria da biblioteca D3¹ também podem ser facilmente incluídas utilizando a base do componente de bloco do sistema.

Além das visualizações, vários tipos de operações sobre os dados podem ser incluídas

¹<https://github.com/d3/d3/wiki/Gallery>

além do simples bloco de filtro presente atualmente. É possível adicionar filtros numéricos, operações de transformação ou operações de conjuntos como união e intersecção de dados, por exemplo. Todas essas operações, assim como novas visualizações, podem ser incluídas sobre a base do modelo de bloco no VisBlocks.

Também é interessante que o usuário possa compartilhar suas visualizações. Então, assim como nas ferramentas relacionadas analisadas no Capítulo 2, o VisBlocks pode futuramente dar apoio à exportação de suas visualizações. Isso deve acontecer tanto entre usuários do sistema como ao fazer download de arquivos como SVG, PNG ou até em um formato próprio que possa ser carregado novamente de forma editável por outros usuários.

Outro aspecto a ser melhorado no VisBlocks é sua eficiência. O sistema é executado totalmente no navegador do usuário, então sua capacidade de processamento depende fortemente do ambiente onde está sendo utilizado. Para melhorar isso é sugerido a adição de um servidor na arquitetura do sistema que possa realizar processamentos muito custosos sobre dados muito grandes.

Por fim, para validar a contribuição científica do VisBlocks e fazer uma avaliação técnica para melhorar a qualidade do sistema, será necessário uma avaliação do impacto causado pela ferramenta ao seu público alvo. Filippo et al. descrevem diversos métodos de pesquisa e coleta de dados que podem ser utilizados para avaliar o sistema FILIPPO; PIMENTEL; WAINER (2011). Os mais adequados para o VisBlocks são experimentos com usuários utilizando a ferramenta proposta e outras ferramentas relacionadas para montar sistemas de visualização de dados. Os usuários podem ser selecionados dentre usuários especialistas em visualização de dados e usuários inexperientes, e os experimentos seguidos de entrevistas para a validação dos resultados com os usuários em relação às ferramentas. Usuários especialistas também podem ser reunidos em um grupo focal utilizado para gerar ideias e expor opiniões sobre a ferramenta para uma análise técnica e sugestões de melhorias. Com isso, à medida que o sistema proposto for evoluindo, será possível produzir um relatório científico descrevendo e avaliando a ferramenta em um estudo técnico sobre sua contribuição à comunidade de Visualização de Dados.

Referências

- AGGARWAL, S. Modern Web-Development using ReactJS. **International Journal of Recent Research Aspects**, [S.l.], v.5, n.1, p.133–137, 2018.
- BAVOIL, L. et al. VisTrails: enabling interactive multiple-view visualizations. In: VIS 05. IEEE VISUALIZATION, 2005. **Anais...** [S.l.: s.n.], 2005. p.135–142.
- BIGELOW, A.; CHOUDHURY, R.; BAUMES, J. Resonant Laboratory and Candela: spreading your visualization ideas to the masses. In: WORKSHOP ON VISUALIZATION IN PRACTICE (VIP '16). **Proceedings...** [S.l.: s.n.], 2016.
- FILIPPO, D.; PIMENTEL, M.; WAINER, J. Metodologia de pesquisa científica em sistemas colaborativos. In: FUKS, M. P. . H. (Ed.). **Sistemas Colaborativos**. [S.l.: s.n.], 2011. p.379–404.
- JAVED, W.; ELMQVIST, N. ExPlates: spatializing interactive analysis to scaffold visual exploration. **Eurographics Conference on Visualization (EuroVis)**, [S.l.], v.32, n.3, p.441–450, 2013.
- LIMA FARIAS, J. R. de; FERREIRA, N. Sweet Home: uma ferramenta de visualização de dados focada no mercado imobiliário. **Workshop de Computação Urbana (COURB-SBRC)**, [S.l.], v.2, n.1/2018, 2018.
- ROBERTS, J. C. On Encouraging Multiple Views for Visualisation. In: IV '98 PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON INFORMATION VISUALISATION. **Anais...** [S.l.: s.n.], 1998.
- SATYANARAYAN, A. et al. Vega-Lite: a grammar of interactive graphics. **IEEE Transactions on Visualization and Computer Graphics**, [S.l.], v.23, n.1, p.341–350, 2017.
- YOURDON, E.; CONSTANTINE, L. L. **Structured Design**: fundamentals of a discipline of computer program and systems design. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1979.
- YU, B.; SILVA, C. T. VisFlow - Web-based Visualization Framework for Tabular Data with a Subset Flow Model. **IEEE Transactions on Visualization and Computer Graphics**, [S.l.], v.23, n.1, p.251–260, 2017.