



Renan Luna

Plataforma para extrair e visualizar dados de Violência Doméstica no Estado de Pernambuco

Recife

2021

Renan Luna

Plataforma para extrair e visualizar dados de Violência Doméstica no Estado de Pernambuco

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Universidade Federal Rural de Pernambuco – UFRPE

Departamento de Computação

Curso de Bacharelado em Ciência da Computação

Orientador: Vanilson Burégio

Recife

2021

Dados Internacionais de Catalogação na Publicação
Universidade Federal Rural de Pernambuco
Sistema Integrado de Bibliotecas
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

- L961p Luna, Renan Victor Maciel de Oliveira
Plataforma para extrair e visualizar dados de violência doméstica no Estado de Pernambuco / Renan Victor Maciel de Oliveira Luna. - 2021.
38 f. : il.
- Orientador: Vanilson Buregio.
Inclui referências.
- Trabalho de Conclusão de Curso (Graduação) - Universidade Federal Rural de Pernambuco,
Bacharelado em Ciência da Computação, Recife, 2021.
1. plataforma web. 2. violência doméstica. 3. java. I. Buregio, Vanilson, orient. II. Título

CDD 004

RENAN LUNA

**Plataforma para extrair e visualizar dados de Violência Doméstica
no Estado de Pernambuco**

Trabalho de Conclusão de Curso apresentado ao Departamento de Computação, da Universidade Federal Rural de Pernambuco, como parte dos requisitos necessários à obtenção do título de Bacharelado em Ciência da Computação.

Recife, 13 de dezembro de 2021.

BANCA EXAMINADORA

Prof. Vanilson Burégio
Universidade Federal Rural de Pernambuco

Prof. Vanilson Burégio
Universidade Federal Rural de Pernambuco

Prof. Rinaldo Lima
Universidade Federal Rural de Pernambuco

Dedico esse TCC aos meus pais

Agradecimentos

A Deus por toda a força e coragem em perseverar no curso.

A UFRPE por todo conhecimento que me foi proporcionado.

Aos meus professores, coordenadores, ex-coordenadores e colegas do curso de Bacharelado em Ciência da Computação, pelo incentivo que me deram nessa jornada.

Ao meu orientador, professor Vanilson Burégio, pela motivação e toda a orientação que, generosamente, recebi de sua parte.

Aos funcionários da UFRPE, em especial, à secretária Sandra Xavier por suas palavras de sabedoria.

Aos meus familiares por toda a ajuda, orientação e incentivo ao meu crescimento pessoal e profissional.

A todos, o meu muito obrigado.

*“A persistência é o caminho do êxito.”
(Charles Chaplin)*

Resumo

Mensalmente, a SDS-PE divulga, por meio do portal da secretaria, uma planilha com dados de crimes de violência doméstica registrados desde 2015. O objetivo de divulgar esses dados consiste em tornar transparente para a população a situação da violência doméstica no estado de Pernambuco. Um problema na divulgação desses dados consiste no fato de que são divulgados como uma tabela em uma planilha no formato *excel* (proprietário), o que dificulta um pouco o processamento e a interpretação dos dados. Para mitigar esses problemas, foi desenvolvida uma plataforma web onde qualquer pessoa pode consultar esses dados de uma forma visual, fazer comparações, e ajudar o governo na elaboração de políticas públicas de combate à violência doméstica no estado de Pernambuco.

Palavras-chave: Violência doméstica, Pernambuco, Dados Abertos, Spring Boot, Java, Plataforma Web.

Abstract

Monthly, the SDS-PE publishes, through its portal, a spreadsheet with domestic violence data registered since 2015. The goal of publish those data consists in to make transparent to the population the situation of domestic violence in the state of Pernambuco. One problem in publicizing the data consists that the table inside of the spreadsheet is in the XSLX format, which makes a little bit difficult to process and to make some inference of the data. To mitigate those problems, a web platform was developed where anyone can consult those data in a visual way , make comparisons, and help the government to make public policies to combat domestic violence in the state of Pernambuco.

Keywords: Domestic violence, Pernambuco, Open data, Spring Boot, Java, Web Platform.

Lista de ilustrações

Figura 1 – Gráficos mostrando quem eram os agressores em 2019 e 2021 . . .	12
Figura 2 – Gráficos mostrando onde as vítimas eram agredidas em 2019 e 2021	12
Figura 3 – Planilha no formato <i>XLSX</i> onde os crimes são divulgados pela SDS-PE	19
Figura 4 – Processo de extração dos dados	19
Figura 5 – Diagrama mostrando as camadas do sistema/plataforma <i>web</i>	21
Figura 6 – Diagrama de classes do <i>backend</i> da plataforma <i>web</i>	22
Figura 7 – Classe “ <i>CrimeUtils</i> ” com a assinatura dos métodos	23
Figura 8 – Interface “ <i>CrimeRepository</i> ” com a assinatura dos métodos	23
Figura 9 – Classe “ <i>CrimeService</i> ” com a assinatura dos métodos	23
Figura 10 – Classe “ <i>CrimeController</i> ” com a assinatura dos métodos	24
Figura 11 – Modelo da classe de Crime	24
Figura 12 – Classe de Crime com os atributos e anotações	25
Figura 13 – A primeira metade da classe que faz a extração dos dados da planilha	26
Figura 14 – A primeira metade da classe que faz a extração dos dados da planilha	26
Figura 15 – Os métodos do repositório de crimes.	27
Figura 16 – Último método do repositório de crimes.	27
Figura 17 – Os métodos do <i>controller</i> de crimes.	29
Figura 18 – Outros métodos do <i>controller</i> de crimes.	29
Figura 19 – Últimos métodos do <i>controller</i> de crimes.	30
Figura 20 – Tabela mostrando os dez mais recentes crimes de Violência Doméstica no Estado de Pernambuco. Em seguida, está o gráfico que mostra a quantidade total de crimes de Violência Doméstica no Estado de Pernambuco por ano.	31
Figura 21 – <i>Dropdown</i> que mostra os anos selecionáveis	31
Figura 22 – Gráfico de rosca mostrando as regiões do Estado de Pernambuco onde acontecem mais crimes de violência doméstica. Em seguida, está um gráfico de barra vertical onde é mostrado o quantitativo de crimes de violência doméstica no Estado de Pernambuco	32
Figura 23 – Gráfico de pizza mostrando o quantitativo de crimes de Violência Doméstica por natureza. Em seguida, está um gráfico de barra vertical onde é mostrado o quantitativo de crimes de Violência Doméstica por natureza.	32
Figura 24 – Gráfico de rosca mostrando o quantitativo de crimes de Violência Doméstica por faixa etária da vítima.	33

Lista de tabelas

Lista de abreviaturas e siglas

SDS-PE	Secretaria de Defesa Social do Estado de Pernambuco
XLSX	Excel Microsoft Office Open XML Format Spreadsheet File
CSV	Comma Separated Values
TJPE	Tribunal de Justiça de Pernambuco
API	Application Programming Interface
DOJ	Department of Justice
CRUD	Create, read, update and delete
SENASP	Secretaria Nacional de Segurança Pública
BD	Banco de Dados
JPQL	Java Persistence Query Language
DAO	Data Access Object
HTTP	Hypertext Transfer Protocol
URL	Uniform Resource Locator
MIT	Massachusetts Institute of Technology
PDF	Portable Document Format
CVLI	Crimes Violentos Letais Intencionais
MVC	Model View Controller
CF	Constituição Federal
SSP	Secretaria de Segurança Pública
UI	User Interface

Sumário

	Lista de ilustrações	7
1	INTRODUÇÃO	11
1.1	Motivação	11
1.2	Objetivos	13
1.2.1	Objetivo Geral	13
1.2.2	Objetivos Específicos	13
1.3	Contribuições	13
1.4	Estrutura do Trabalho	13
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	Dados abertos	14
2.2	Padrões de arquitetura	15
2.2.1	Arquitetura Cliente-Servidor	15
2.2.2	Arquitetura em Camadas	15
2.2.3	Paradigma Orientado a Objetos	15
2.3	Violência Doméstica	16
3	EXTRAÇÃO E VISUALIZAÇÃO DOS DADOS	18
3.1	Extraindo os dados	18
3.2	Arquitetura do Sistema	20
3.3	Tecnologias e <i>Frameworks</i>	24
3.4	Implementação	30
4	CONCLUSÃO	34
4.1	Trabalhos Relacionados	34
4.2	Considerações finais	34
4.3	Trabalhos futuros	35
	REFERÊNCIAS	36

1 Introdução

”Dados são o novo petróleo”, essa frase foi dita por Clive Humby, um matemático especialista em ciência de dados. Uma diferença fundamental dos dados para o petróleo consiste no fato de que os dados não são escassos, ou seja, a cada segundo (MAVUDURUI, 2020), novos dados são criados, seja ao interagir em uma rede social, ou até ao assistir uma produção audiovisual em um serviço de *streaming*.

No Brasil, existem iniciativas para disponibilizar dados por parte de governos, seja na esfera federal, estadual ou municipal. No caso deste trabalho de conclusão de curso, será limitado ao poder executivo estadual, onde, mesmo com todo o esforço de disponibilizar dados, ainda possuem alguns problemas, sendo o principal, o fato de os dados estarem em um formato proprietário *XLSX*, quando deveriam ser em um formato aberto, como é o caso do *CSV*, ferindo um dos princípios dos dados abertos: o princípio dos formatos não proprietários.

Ao tornar público alguns dados dos crimes, não só de Violência Doméstica, é verificado, de acordo com Wartell e McEwen (WARTELL; MCEWEN, 2001), a percepção de algumas vantagens e desvantagens. Como vantagens, podemos destacar uma possibilidade na diminuição da quantidade de trabalho para a polícia, pois mapeado o mapa de crimes, pode haver um policiamento ostensivo em áreas consideradas perigosas. Uma outra vantagem é para a população, pois tendo mais conhecimento sobre os crimes, maior é a cooperação com a polícia para manter a lei e a ordem. Infelizmente, divulgar dados de crimes pode ter também suas desvantagens, uma desvantagem consiste no fato de que, como os dados são abertos, alguma empresa pode usar esses dados para fins comerciais, vendendo algum produto ou serviço para as pessoas em bairros perigosos, e potenciais criminosos podem usar esses dados para cometerem crimes em áreas onde a polícia não é muito presente.

Apesar das vantagens e desvantagens de abrir alguns dados de crimes, é necessário alguma forma para tornar esses dados mais compreensíveis para a população.

1.1 Motivação

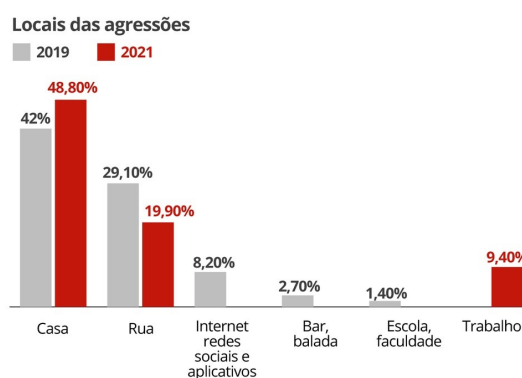
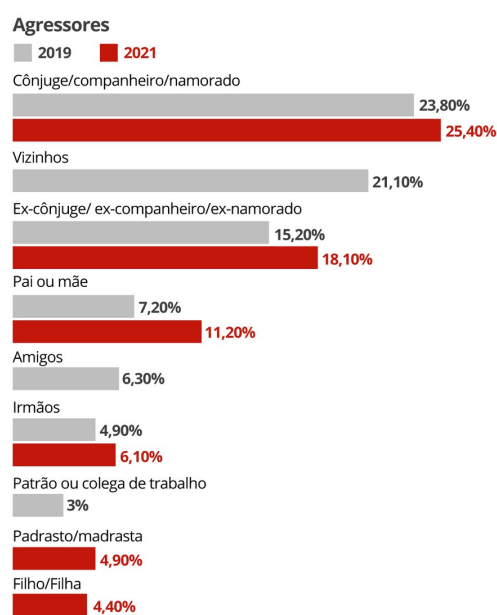
A violência provavelmente sempre fez parte do ser humano, o impacto resulta em milhões de mortes e feridas não fatais por ano em todo o mundo. Apesar do custo das despesas para remediar a violência, seja tratando os feridos, ou levando os acusados aos tribunais, o custo da perda de um vida humana por causa da violência é

incalculável (DAHLBERG; KRUG, 2006).

Segundo uma pesquisa do Instituto Datafolha, 24,4% das mulheres ouvidas pelo instituto sofreram algum tipo de violência doméstica durante a pandemia de COVID-19. No levantamento anterior, as entrevistadas disseram que os vizinhos as agrediam, ficando em segundo lugar. Em comparação ao levantamento atual, os vizinhos sumiram da lista, tendo membros da família como agressores, tornando a casa dessas vítimas um lugar muito perigoso, o que deveria ser o contrário.

Cai violência na rua e aumentam agressões dentro de casa

"Vizinho" some das respostas e entram pessoas da família



Fonte: Instituto Datafolha e Fórum Brasileiro de Segurança Pública



Infográfico elaborado em: 07/06/2021

Figura 2 – Gráficos mostrando onde as vítimas eram agredidas em 2019 e 2021

(PAULO, 2021)

Figura 1 – Gráficos mostrando quem eram os agressores em 2019 e 2021 (PAULO, 2021)

Para remediar o problema da violência doméstica no estado de Pernambuco, o TJPE criou uma iniciativa para apoiar e orientar as vítimas de violência doméstica. Essa iniciativa consiste em um canal de comunicação entre a vítima, ou outra pessoa que queira denunciar, e o poder Judiciário, onde a denúncia é analisada e o local adequado para a vítima é informado podendo ser uma delegacia, casa de acolhimento, entre outros (NOTÍCIAS, 2020).

Para a população entender onde estão ocorrendo os crimes de violência doméstica, estudar a planilha dos microdados de crimes não é viável. Porém, traduzir a planilha em uma linguagem visual para que os dados possam ser compreendidos torna viável o entendimento dos dados dos crimes, possibilitando a criação de políticas públicas de prevenção e combate aos crimes de violência doméstica no estado de Pernambuco.

1.2 Objetivos

1.2.1 Objetivo Geral

Como a planilha de microdados de violência doméstica não está em um formato adequado para a divulgação, este trabalho tem como objetivo apresentar uma plataforma *web* que mapeia os crimes de Violência Doméstica do Estado de Pernambuco por meio dos dados abertos divulgados pela SDS-PE, visando uma maior transparência das informações para a população.

1.2.2 Objetivos Específicos

1. Realizar uma extração dos microdados de violência doméstica divulgados pela SDS-PE;
2. Persistir os microdados em um BD;
3. Desenvolver uma *API* onde esses dados podem ser consultados;
4. Implementar uma plataforma *web* para a visualização dos dados que vai consumir a *API*.

1.3 Contribuições

As principais contribuições oferecidas por esse trabalho são:

1. Uma apresentação detalhada sobre o *backend* e *frontend*;
2. Implementação de uma ferramenta para a sociedade poder fazer uma análise visual dos dados de crimes de violência doméstica.

1.4 Estrutura do Trabalho

Os próximos capítulos desse trabalho são:

Capítulo 2 - Mostrar um resumo dos principais fundamentos que servem de base para esse trabalho.

Capítulo 3 - Apresentação do sistema e tecnologia utilizadas.

Capítulo 4 - Conclusão do que foi desenvolvido e sugestões para trabalhos futuros.

2 Fundamentação Teórica

Nesse capítulo haverá várias abordagens sobre os fundamentos básicos para esse trabalho. Na seção 2.1 será abordado o atual estado dos dados abertos. Na 2.2, será discutido os conceitos de padrão de arquitetura de projetos de software. Na seção 2.3, será mostrado o conceito de violência doméstica.

2.1 Dados abertos

(WANG et al., 2018) Dados abertos se referem ao conceito de que os dados são publicados para que qualquer pessoa, empresa ou organização, seja ela governamental ou não, possa acessar e ver tais dados gratuitamente. Os dois principais motivos para os governos estarem dispostos a disponibilizarem seus dados são:

1. Aumentar a transparência;
2. Estimular o reuso de dados e a criação ou melhora de aplicações.

Para um dado ser considerado aberto, esse dado precisa seguir três "leis" e oito princípios. As três leis dos dados abertos são:

1. Se o dado não pode ser encontrado e indexado na Web, ele não existe;
2. Se não estiver aberto e disponível em formato compreensível por máquina, ele não pode ser reaproveitado;
3. Se algum dispositivo legal não permitir sua replicação, ele não é útil.

(WANG et al., 2018) Já os oito princípios que os dados precisam seguir para serem considerados abertos são:

1. **Completo.** Todos os dados públicos devem ser disponibilizados. Dados públicos são dados que não estão sujeitos a restrições de privacidade, segurança ou privilégios de acesso.
2. **Primários.** Os dados devem ser coletados na fonte com o maior nível de detalhamento possível, e não de forma agregada ou modificada.
3. **Oportunidade.** Sua disponibilidade deve ser feita tão rapidamente quanto necessário para preservar o valor dos dados.

4. **Acessibilidade.** Os dados devem estar disponíveis para a mais ampla gama de usuários e as mais diversas finalidades.
5. **Processável por máquinas.** Os dados devem ser razoavelmente estruturados de modo a permitir o processamento automatizado.
6. **Não-discriminatório.** Os dados devem estar disponíveis para qualquer pessoa, sem necessidade de registro.
7. **Não-proprietário.** Os dados devem estar disponíveis em um formato sobre o qual nenhuma entidade tem o controle exclusivo.
8. **Licença livre** Os dados não estão sujeitos a quaisquer direitos de autor, patentes, marcas comerciais ou regulamento secreto. Pode ser permitida uma razoável privacidade e restrições de privilégio e segurança.

2.2 Padrões de arquitetura

Desenvolver software vai além da linguagem de programação e de algoritmos e estruturas de dados. É necessário um padrão de arquitetura para uma melhor modularidade e baixo acoplamento dos componentes. Tendo conhecimento dessa necessidade de padrões, vários padrões de arquitetura foram desenvolvidos nas últimas décadas. Nesse trabalho, foram utilizadas as arquiteturas cliente-servidor, arquitetura em camadas, e o paradigma de programação orientado a objetos.

2.2.1 Arquitetura Cliente-Servidor

A arquitetura cliente-servidor é um modelo na qual existe uma parte responsável pela manutenção (CRUD) das informações é chamado de servidor, e onde existe uma outra parte responsável pela requisição desses dados, chamado de cliente (TI, 2018).

2.2.2 Arquitetura em Camadas

Um tipo de arquitetura onde o sistema é organizado em um conjunto de camadas, onde cada camada oferece um conjunto de serviços. Cada camada oferece serviços que podem ser solicitados pela camada inferior e fornece serviços para a camada superior (FIGUEIREDO, 2011).

2.2.3 Paradigma Orientado a Objetos

O paradigma OO (Orientada a Objetos) é um paradigma de programação baseado em uma representação do mundo real no mundo virtual através de classes, objetos, atributos, métodos, entre outros (NOLETO, 2020).

2.3 Violência Doméstica

O controle dos níveis de violência pelo Estado deve ser uma preocupação constante para o bem-estar social. Esse controle é exercido através dos órgãos e instituições que compõem o Sistema de Segurança Pública, que visam à preservação da ordem pública e da segurança das pessoas e do patrimônio, posto que a segurança pública é tida como um dever do Estado e direito e responsabilidade de todos, nos exatos termos do artigo 144, *caput*, da Constituição Federal (BRASIL, 1988). O conteúdo do artigo 144 da CF é: A segurança pública, dever do Estado, direito e responsabilidade de todos, é exercida para a preservação da ordem pública e da incolumidade das pessoas e do patrimônio, através dos seguintes órgãos:

1. polícia federal;
2. polícia rodoviária federal;
3. polícia ferroviária federal;
4. polícias civis;
5. polícias militares e corpos de bombeiros militares.
6. polícias penais federal, estaduais e distrital. (Redação dada pela Emenda Constitucional nº 104, de 2019).

A violência, que é uma das faces da criminalidade, tende a escapar desse controle estatal, assumindo patamares indesejados, o que exige a constante implementação de políticas públicas e sociais em prol da preservação da ordem pública, notadamente com ações preventivas desenvolvidas pelos órgãos policiais.

Como a Violência Doméstica tende a fugir do controle estatal, o Estado busca criar dispositivos legais que ajudem no enfrentamento desse delito, visando ao controle social. Para tanto, surgiu a Lei Maria da Penha (Lei nº 11.340, de 7 de agosto de 2006), que criou mecanismos para coibir a violência doméstica e familiar, notadamente contra a mulher, “para Prevenir, Punir e Erradicar a Violência contra a Mulher e de outros tratados internacionais ratificados pela República Federativa do Brasil”, como pode ser verificado no artigo 1º.

Violência Doméstica é uma ação ou omissão baseada no gênero que cause alguma consequência como a morte, a lesão, o sofrimento físico, sexual ou psicológico e o dano moral ou patrimonial (PENHA, 2018a). A violência doméstica é um fenômeno que existe em 5 tipos (PENHA, 2018b):

1. **FÍSICA** - Conduta que ofenda a integridade ou saúde corporal da mulher.

2. **PSICOLÓGICA** - Qualquer conduta que: cause dano emocional e diminuição da autoestima; prejudique e perturbe o pleno desenvolvimento da mulher; ou vise degradar ou controlar suas ações, comportamentos, crenças e decisões;
3. **SEXUAL** - Conduta que constranja a presenciar, a manter ou a participar de relação sexual não desejada mediante intimidação, ameaça, coação ou uso da força;
4. **PATRIMONIAL** - Conduta que configure retenção, subtração, destruição parcial ou total de seus objetos, instrumentos de trabalho, documentos pessoais, bens, valores e direitos ou recursos econômicos, incluindo os destinados a satisfazer suas necessidades;
5. **MORAL** - Conduta que configure calúnia, difamação ou injúria.

No site oficial do Instituto Maria da Penha, podemos encontrar a explicação para um dos mitos de que “a violência doméstica só acontece em famílias de baixa renda e pouca instrução.”, “A violência doméstica é um fenômeno que não distingue classe social, raça, etnia, religião, orientação sexual, idade e grau de escolaridade” (PENHA, 2018b).

Assim, pela força e abrangência do tema, a importância do presente trabalho justifica-se pelo seu objetivo de apresentar uma plataforma *web* que mapeia os crimes de violência doméstica do estado de Pernambuco por meio dos dados abertos divulgados pela SDS-PE, visando trazer informações à população em geral, bem assim oferecer tratamento de dados técnicos que possam contribuir para a implementação de estratégias para os órgãos de segurança pública no enfrentamento da Violência Doméstica.

3 Extração e Visualização dos Dados

Esse capítulo descreve como a plataforma *web* foi implementada desde a extração e persistência dos dados de crimes de violência doméstica em um banco de dados, e disponibilização dos dados como uma *API web*. Na seção 3.1, será mostrado como foi feita a extração dos dados. A seção 3.2 mostra os detalhes do sistema. A seção 3.3 detalha as tecnologias e *frameworks* utilizados no desenvolvimento da plataforma. E a seção 3.4 mostra o sistema desenvolvido e seus componentes.

3.1 Extraíndo os dados

Os dados de crimes de violência doméstica são divulgados mensalmente no portal da SDS-PE. O único problema na divulgação é que os dados são divulgados em uma planilha no formato *XLSX*, o que fere um dos princípios dos dados abertos. Esses dados contém crimes ocorridos desde 2015 e que, infelizmente, cresce em número todo mês, onde cada linha, exceto o cabeçalho, é um crime cometido.

Os campos na planilha de microdados de crimes de Violência Doméstica são os seguintes:

1. **MUNICÍPIO DO FATO** - Município onde ocorreu o crime;
2. **REGIÃO GEOGRÁFICA** - Região do Estado de Pernambuco onde o município do fato é localizado;
3. **NATUREZA** - Natureza do crime;
4. **DATA DO FATO** - Data do crime;
5. **ANO** - Ano do crime;
6. **SEXO** - Sexo da vítima;
7. **IDADE SENASP** - Faixa etária da vítima;
8. **TOTAL DE ENVOLVIDOS** - Quantidade de envolvidos;

	A	B	C	D	E	F	G	H
1	MUNICÍPIO DO FATO	REGIAO GEOGRÁFICA	NATUREZA	DATA DO FATO	ANO	SEXO	IDADE SENASP	TOTAL DE ENVOLVIDOS
2	AFOGADOS DA INGAZEIRA	SERTÃO	INJÚRIA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
3	AFRANIO	SERTÃO	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
4	AFRANIO	SERTÃO	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	4) 25-29	1
5	AGUAS BELAS	AGRESTE	ESTUPRO DE VULNERÁVEL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	2) 12-17	1
6	ALAGOINHA	AGRESTE	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
7	ALAGOINHA	AGRESTE	DIFAMAÇÃO POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
8	ALIANCA	ZONA DA MATA	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
9	ANGELIM	AGRESTE	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
10	BELO JARDIM	AGRESTE	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	3) 18-24	1
11	BEZERROS	AGRESTE	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
12	BODOCO	SERTÃO	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
13	BOM JARDIM	AGRESTE	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
14	CABO DE SANTO AGOSTINHO	REGIÃO METROPOLITANA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	3) 18-24	1
15	CABO DE SANTO AGOSTINHO	REGIÃO METROPOLITANA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
16	CABO DE SANTO AGOSTINHO	REGIÃO METROPOLITANA	INJÚRIA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	3) 18-24	1
17	CABO DE SANTO AGOSTINHO	REGIÃO METROPOLITANA	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
18	CABO DE SANTO AGOSTINHO	REGIÃO METROPOLITANA	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
19	CABROBO	SERTÃO	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
20	CAMARAGIBE	REGIÃO METROPOLITANA	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	4) 25-29	1
21	CAMARAGIBE	REGIÃO METROPOLITANA	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	2
22	CARNAUBEIRA DA PENHA	SERTÃO	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	3) 18-24	1
23	CARUARU	AGRESTE	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
24	CARUARU	AGRESTE	ESTUPRO DE VULNERÁVEL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	3) 18-24	1
25	CARUARU	AGRESTE	INJÚRIA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	3) 18-24	1
26	CARUARU	AGRESTE	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	3) 18-24	1
27	CARUARU	AGRESTE	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
28	CARUARU	AGRESTE	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
29	EXU	SERTÃO	OUTROS CRIMES POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
30	FEIRA NOVA	AGRESTE	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	5) 30-34	1
31	FERNANDO DE NORONHA	REGIÃO METROPOLITANA	DANO POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
32	FLORES	SERTÃO	OUTROS CRIMES POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	MASCULINO	6) 35-64	1
33	ITAMARACA	REGIÃO METROPOLITANA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
34	ITAQUITINGA	ZONA DA MATA	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1
35	JABOATAO DOS GUARARAPES	REGIÃO METROPOLITANA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	4) 25-29	1
36	JABOATAO DOS GUARARAPES	REGIÃO METROPOLITANA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	2
37	JABOATAO DOS GUARARAPES	REGIÃO METROPOLITANA	INJÚRIA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	01/01/2015	2015	FEMININO	6) 35-64	1

Figura 3 – Planilha no formato XLSX onde os crimes são divulgados pela SDS-PE

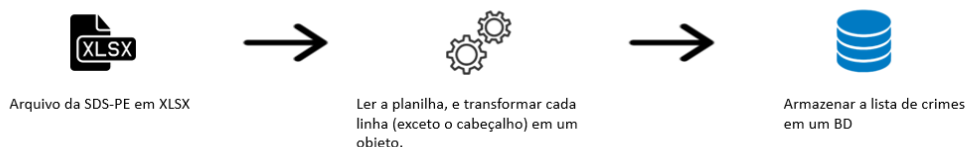


Figura 4 – Processo de extração dos dados

Após a extração dos dados, cada linha é armazenada como um objeto da classe Crime. Os atributos dessa classe são:

1. **id** - O identificador do crime no BD;
2. **municipioDoFato** - Município onde ocorreu o crime;
3. **regiaoGeografica** - Região geográfica onde o município é localizado;
4. **natureza** - Tipo do crime;
5. **dataDoFato** - Data do crime;
6. **ano** - Ano do crime;

7. **sexo** - Sexo da vítima;
8. **faixaEtaria** - Faixa etária da vítima;
9. **totalDeEnvolvidos** - Total de envolvidos no crime.

3.2 Arquitetura do Sistema

(BURÉGIO; MEIRA; ROSA, 2013) propôs o paradigma “Máquinas Sociais” que representa um paradigma promissor para lidar com a complexidade dessa nova *Web* que está emergindo ao nosso redor. Do ponto de vista de software, pode ser vista como uma maneira simples, formal e unificada maneira para descrever um software na *Web*. Esse paradigma resulta da convergência de três visões:

1. **Software Social** - As versões primárias dos *softwares* sociais, são os sistemas *web* que fazem parte da chamada “*web 2.0*”, a segunda geração de sistemas *web* onde existe um compartilhamento de informações online (CARVALHO, 2008).
2. **Pessoas como Unidades Computacionais** - Essa visão refere-se à esforços de pesquisa que integram pessoas e *softwares* em um sistema composto.
3. **Software como Entidades Sociáveis** - Considerando apenas o contexto da *Web*, essa visão tem como foco, trabalhos que tentam entrelaçar elementos sociais em *softwares* visando a “socialização” desses *softwares* permitindo a interação de um com o outro.

Tendo como inspiração o paradigma de Máquinas Sociais, e a ideia de tornar o Governo uma máquina social (TOWARDS..., 2015), a arquitetura do sistema consiste em uma plataforma *web* que utiliza dados da planilha de microdados divulgada pela SDS-PE desde 2015, para mostrar, através de elementos visuais, a situação da violência doméstica no Estado de Pernambuco. Uma visão mais geral da arquitetura é mostrada na figura 5.

Na camada de dados, existe somente uma única fonte de dados. Essa fonte é a planilha de microdados.

Acima da camada de dados, consta a camada de processamento. Nessa camada, existe uma classe que é a responsável por ler a planilha no formato *XLSX* e salvar os dados em um BD, chamada “*CrimeUtils*”. Para poder salvar os dados e realizar as consultas ao BD, foi feita uma *interface* chamada “*CrimeRepository*” que possui os *scripts* de consulta e as assinaturas dos métodos.

Na camada de serviços, existem duas classes. Uma se chama “*CrimeService*”, onde nessa classe é feita a implementação dos métodos do repositório de crimes. A

outra classe é chamada de “*CrimeController*”, onde são implementados os *endpoints* da API de crimes de violência doméstica. Todas as classes e a interface serão mais detalhadas na próxima seção. Abaixo, na figura 6 está o diagrama de classes do *backend* da plataforma web. No diagrama é mostrado os relacionamentos das classes “*CrimeUtils*” com a interface “*CrimeRepository*”

Para mostrar os dados da planilha de crimes, a camada de aplicação consome os dados retornados dos *endpoints*, e converte esses dados de uma linguagem verbal para uma não verbal por meio de gráficos.

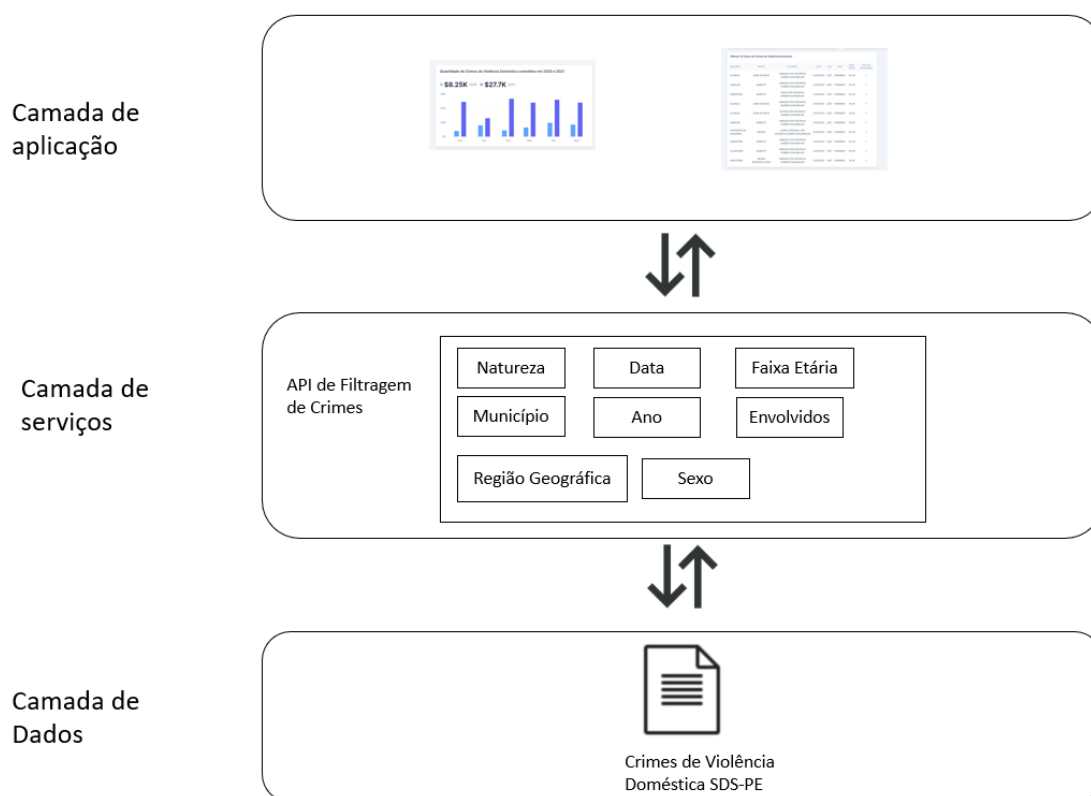


Figura 5 – Diagrama mostrando as camadas do sistema/plataforma web

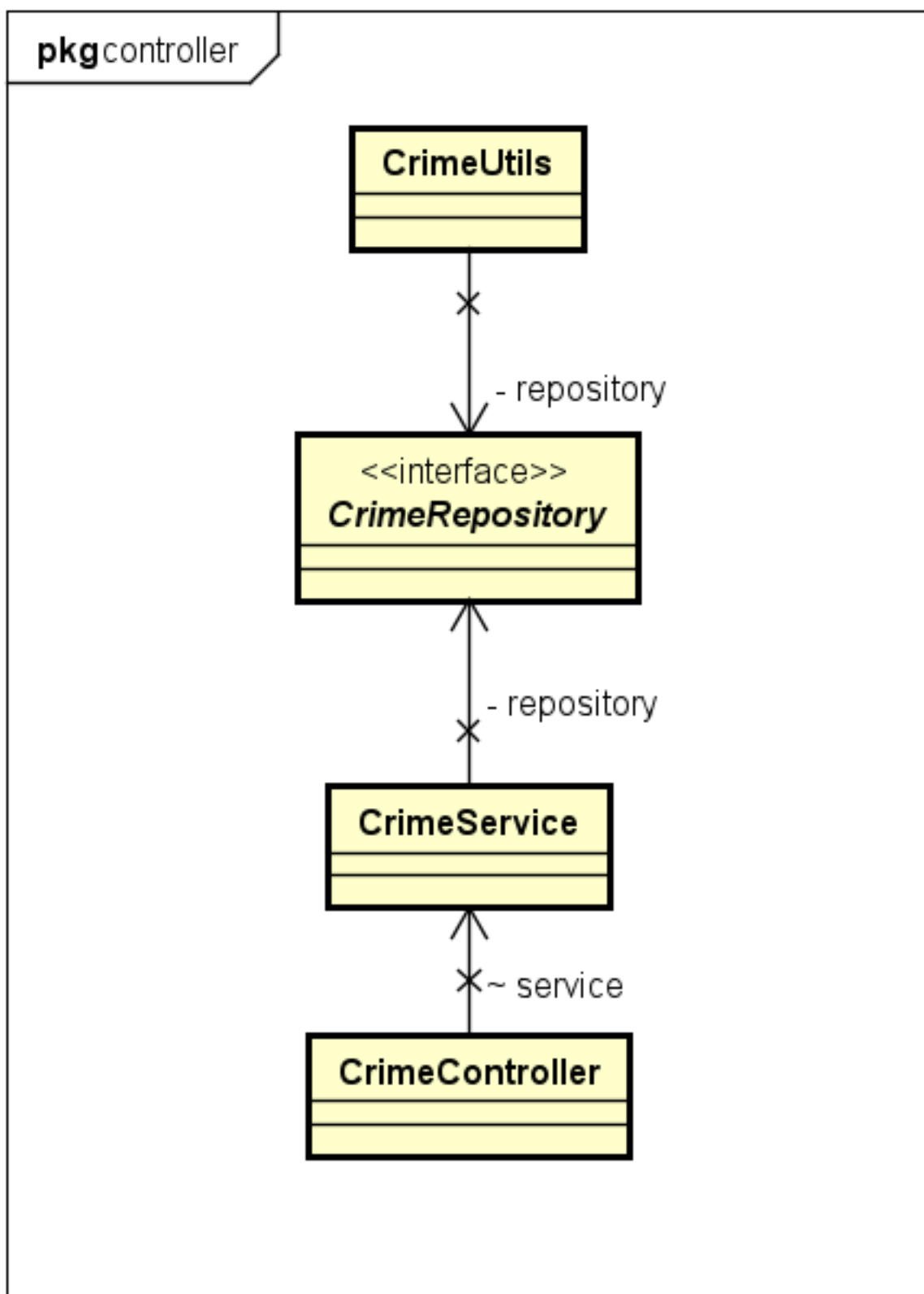
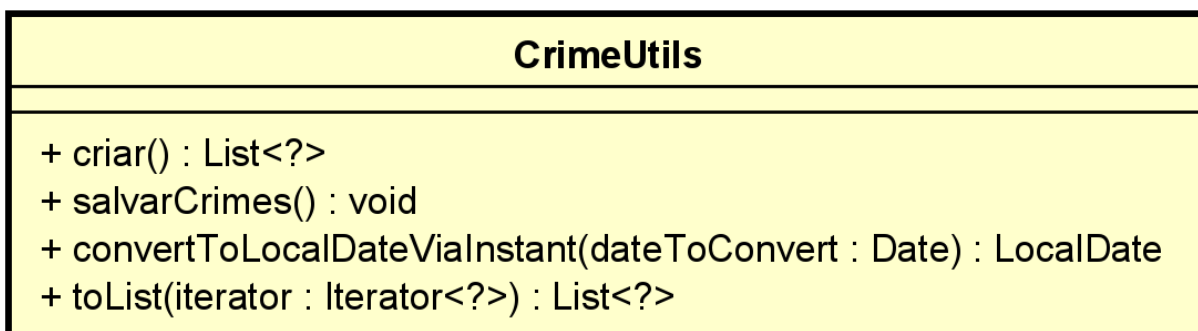
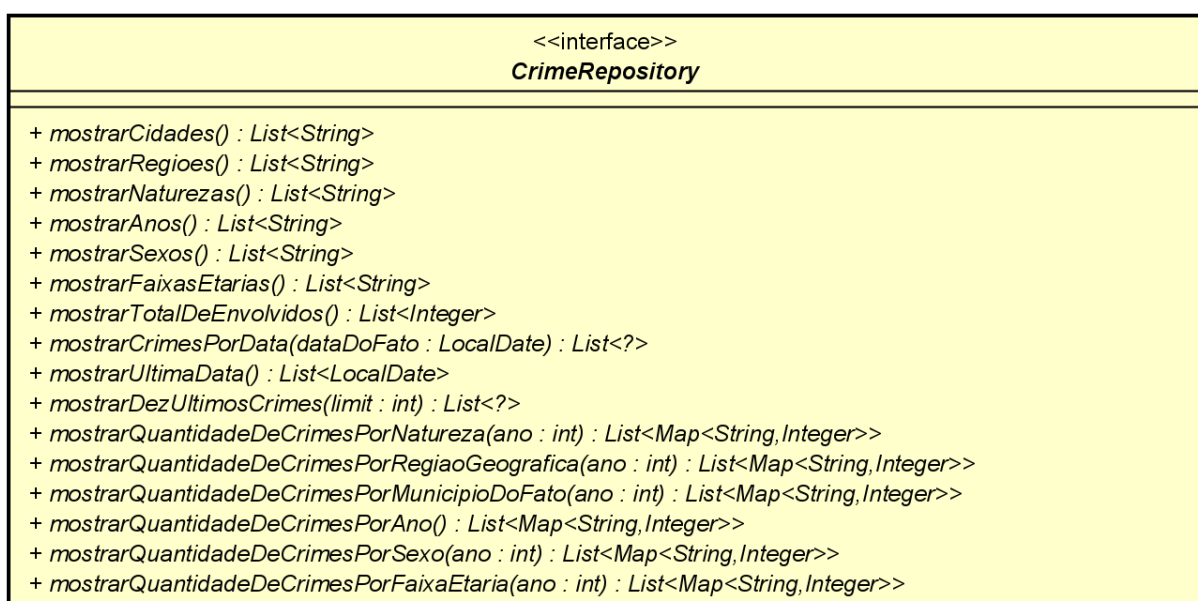
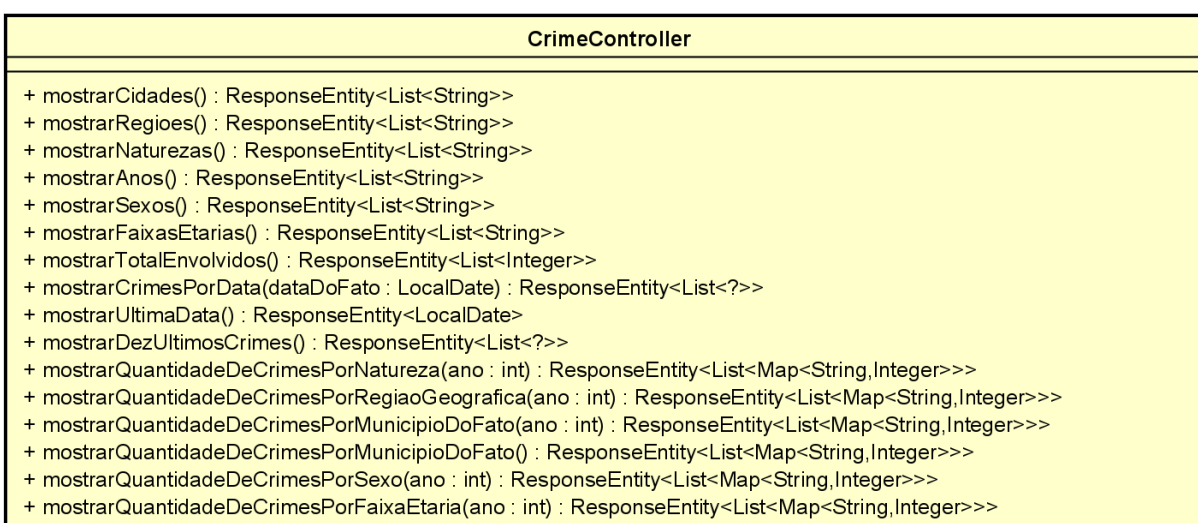


Figura 6 – Diagrama de classes do *backend* da plataforma *web*

Em seguida, são apresentadas as classes e a *interface* com os métodos.

Figura 7 – Classe “*CrimeUtils*” com a assinatura dos métodosFigura 8 – Interface “*CrimeRepository*” com a assinatura dos métodosFigura 9 – Classe “*CrimeService*” com a assinatura dos métodos

CrimeService
<pre> + mostrarCidades() : List<String> + mostrarRegioes() : List<String> + mostrarNaturezas() : List<String> + mostrarAnos() : List<String> + mostrarSexos() : List<String> + mostrarFaixasEtarias() : List<String> + mostrarTotalDeEnvolvidos() : List<Integer> + mostrarCrimesPorData(dataDoFato : LocalDate) : List<?> + mostrarUltimaData() : List<LocalDate> + mostrarDezUltimosCrimes() : List<?> + mostrarQuantidadeDeCrimesPorNatureza(ano : int) : List<Map<String,Integer>> + mostrarQuantidadeDeCrimesPorRegiaoGeografica(ano : int) : List<Map<String,Integer>> + mostrarQuantidadeDeCrimesPorMunicipioDoFato(ano : int) : List<Map<String,Integer>> + mostrarQuantidadeDeCrimesPorAno() : List<Map<String,Integer>> + mostrarQuantidadeDeCrimesPorSexo(ano : int) : List<Map<String,Integer>> + mostrarQuantidadeDeCrimesPorFaixaEtaria(ano : int) : List<Map<String,Integer>> </pre>

Figura 10 – Classe “*CrimeController*” com a assinatura dos métodos

Crime
<pre> - serialVersionUID : long = 1L - id : Long - municipioDoFato : String - regioaoGeografica : String - natureza : String - ano : int - sexo : String - faixaEtaria : String - totalDeEnvolvidos : int + Crime(municipioDoFato : String, regioaoGeografica : String, natureza : String, dataDoFato : LocalDate, ano : int, sexo : String, faixaEtaria : String, totalDeEnvolvidos : int) + Crime() + getId() : Long + setId(id : Long) : void + getMunicipioDoFato() : String + setMunicipioDoFato(municipioDoFato : String) : void + getRegiaoGeografica() : String + setRegiaoGeografica(regiaoGeografica : String) : void + getNatureza() : String + setNatureza(natureza : String) : void + getDataDoFato() : LocalDate + setDataDoFato(dataDoFato : LocalDate) : void + getAno() : int + setAno(ano : int) : void + getSexo() : String + setSexo(sexo : String) : void + getFaixaEtaria() : String + setFaixaEtaria(faixaEtaria : String) : void + getTotalDeEnvolvidos() : int + setTotalDeEnvolvidos(totalDeEnvolvidos : int) : void </pre>

Figura 11 – Modelo da classe de Crime

3.3 Tecnologias e *Frameworks*

Para construir o *backend* do sistema, foi usado o *framework* [Spring Boot](#), usando a linguagem de programação Java. O *backend* do projeto usa como BD o PostgreSQL, um BD relacional.

Para fazer o mapeamento objeto relacional, foi usado o *framework* [Hibernate](#), que visa simplificar o mapeamento objetos em BD, tornando cada classe em uma ta-

bela no BD e cada objeto dessa classe, uma linha no BD. A classe Crime possui como atributos os nomes das colunas que estão na planilha de microdados, exceto por uma coluna, onde na planilha se chama "IDADE SENASP" e no sistema, foi renomeada para faixaEtaria para ficar mais lógico, já que na planilha não é mostrada a idade exata da vítima.

```
11 @Entity
12 @Table(name = "tb_crimes")
13 public class Crime implements Serializable{
14
15     private static final long serialVersionUID = 1L;
16
17     @Id
18     @GeneratedValue(strategy = GenerationType.IDENTITY)
19     private Long id;
20
21     private String municipioDoFato;
22     private String regioaoGeografica;
23     private String natureza;
24     private LocalDate dataDoFato;
25     private int ano;
26     private String sexo;
27     private String faixaEtaria;
28     private int totalDeEnvolvidos;
29
30     public Crime(String municipioDoFato, String regioaoGeografica, String natureza,
31                 LocalDate dataDoFato, int ano, String sexo, String faixaEtaria, int totalDeEnvolvidos) {
32         this.municipioDoFato = municipioDoFato;
33         this.regiaoGeografica = regioaoGeografica;
34         this.natureza = natureza;
35         this.dataDoFato = dataDoFato;
36         this.ano = ano;
37         this.sexo = sexo;
38         this.faixaEtaria = faixaEtaria;
39         this.totalDeEnvolvidos = totalDeEnvolvidos;
40     }
41
42     public Crime() {
43     }
44 }
```

Figura 12 – Classe de Crime com os atributos e anotações

Como podemos ver na figura acima, na linha 11, 12, 17 e 18 constam anotações. Anotações são fragmentos de código criados para descrever classes, atributos e métodos na linguagem Java (ARAÚJO, 2012). Na linha 11, consta a anotação `@Entity`, que mapeia a classe e cria uma tabela no BD. Para deixar um nome de tabela que segue as boas práticas, foi usado, na linha 12, uma anotação chamada `@Table`, e nessa anotação foi adicionada a propriedade `name`, para mudar o nome da tabela para "tb_crimes". No *hibernate*, para definir o id da tabela, são necessárias duas anotações:

1. **@Id** - define o atributo como um id
2. **@GeneratedValue** - Define a estratégia para a geração de IDs, nessa anotação tem uma propriedade chamada `strategy`, que especifica qual estratégia será usada para gerar um id incrementável.

Para ler o arquivo XLSX, foi necessário um *framework* desenvolvido pela *Apache Foundation* chamado de [Apache POI](#), esse framework permite a leitura do arquivo

XLSX e extração dos conteúdos das células da planilha em tipos de dados que a linguagem Java entende. Para extrair esses dados, foram necessários alguns passos.

```

@Component
public class CrimeUtils {

    @Autowired
    private CrimeRepository repository;

    public List<Crime> criar() throws FileNotFoundException {
        List<Crime> crimes = new ArrayList<Crime>();

        try {
            // lendo o arquivo
            FileInputStream file = new FileInputStream(
                "src/main/resources/MICRODADOS_DE_VIOLENCIA_DOMESTICA_JAN_2015_A_OUT_2021.xlsx");
            Workbook workbook = new XSSFWorkbook(file);
            // Setando a aba
            Sheet sheet = workbook.getSheetAt(0);

            // Setando as linhas
            List<Row> rows = (List<Row>) toList(sheet.iterator());

            // Remove os cabeçalhos
            rows.remove(0);

            // Setando as células (colunas)
            rows.forEach(row -> {
                List<Cell> cells = (List<Cell>) toList(row.cellIterator());

                String faixaEtariaFormatada = cells.get(6).toString();
                faixaEtariaFormatada = faixaEtariaFormatada.substring(3);

                if (faixaEtariaFormatada.equals("zio")) {
                    faixaEtariaFormatada = "Não informada";
                }

                LocalDate dataConvertida =
                    this.convertToLocalDateViaInstant(cells.get(3).getDateCellValue());

                Crime crime = new Crime();

```

Figura 13 – A primeira metade da classe que faz a extração dos dados da planilha

```

                Crime crime = new Crime();
                crime.setMunicipioDoFato(cells.get(0).toString());
                crime.setRegiaoGeografica(cells.get(1).toString());
                crime.setNatureza(cells.get(2).toString());
                crime.setDataDoFato(dataConvertida);
                crime.setAno((int) cells.get(4).getNumericCellValue());
                crime.setSexo(cells.get(5).toString());
                crime.setFaixaEtaria(faixaEtariaFormatada);
                crime.setTotalDeEnvolvidos((int) cells.get(7).getNumericCellValue());

                crimes.add(crime);
            });
            workbook.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return crimes;
    }

    public void salvarCrimes() throws FileNotFoundException {
        List<Crime> crimes = this.criar(); // Retorna uma lista com todos os crimes na planilha.
        try {
            crimes = repository.saveAll(crimes);
            System.out.printf("Foram adicionadas %d linhas", crimes.size());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    public LocalDate convertToLocalDateViaInstant(Date dateToConvert) {
        return dateToConvert.toInstant().atZone(ZoneId.systemDefault()).toLocalDate();
    }

    public List<T> toList(Iterator<T> iterator) {
        return IteratorUtils.toList(iterator);
    }
}

```

Figura 14 – A primeira metade da classe que faz a extração dos dados da planilha

Na linha 23 na figura 9, foi necessário usar a anotação `@Component`. Essa anotação permite ao bean registrado no *Spring* ser utilizado em qualquer outro *bean*, seja ele um serviço, um *controller*, etc (ALLET, 2013). *Controller* é uma camada do padrão de projeto MVC responsável por ser um intermediário entre a camada de dados e a camada de aplicação (ZUCHER, 2020). Na 27, foi “instanciado” um objeto de *repository* da interface *CrimeRepository*, essa interface será detalhada mais a frente nesse documento. Esse “instanciamento” foi possível com a anotação `@Autowired`, essa anotação fornece controle sobre onde e como a ligação entre os beans deve ser realizada (ALLET, 2013). Na linha 29, está declarado o método `criar`, que não recebe argumentos e pode lançar uma exceção de arquivo não encontrado. Na 31 foi criada um *ArrayList* onde será armazenada a lista de crimes. Dentro de um bloco *try*, foi lida a planilha nas linhas 35 e 36; na 39, foi configurada qual aba do arquivo vai ser lida, como só tem uma aba, foi lida a aba da posição 0; na linha 42, foi setada uma lista de *rows* usando o método `toList` tendo como parâmetro um iterador da planilha que será detalhado mais adiante; na 45, é removido o cabeçalho da planilha; para configurar os objetos da classe *Crime*, foi necessário usar uma função *lambda* onde a cada linha lida, é feita uma conversão para um objeto do tipo *Crime*; nas linhas 51 até 59 são feitas algumas conversões, as conversões são uma mudança na *string* de faixa etária para caso a faixa etária seja terminada com “zio” para “Não informada” e a conversão do tipo *Date* para *LocalDate*. Com o objeto de crime criado, e com os atributos setados, esse objeto é adicionado na lista de crimes. Após percorrer toda a planilha, o arquivo é fechado e a lista é retornada. Com essa lista retornada, é possível executar o método

salvarCrimes. Localizado na linha 81 da classe, esse método cria uma lista de crimes retornada do método “criar” mencionado anteriormente nesse parágrafo. Depois que a lista é obtida, o repositório de crimes é chamado e é invocado o método “saveAll” do repository. Caso haja alguma exceção na persistência dos dados, o *log* de exceções será mostrado no *console* do *Spring Boot*. Na linha 91, está o método que converte um dado do tipo *Date* para *LocalDateTime*. E na linha 95, está o método para converter o iterator em uma List.

```
12 @Repository
13 public interface CrimeRepository extends JpaRepository<Crime, Long> {
14
15     @Query(
16         value = "SELECT DISTINCT obj.municipioDoFato FROM Crime AS obj ORDER BY obj.municipioDoFato")
17     List<String> mostrarCidades();
18
19     @Query(
20         value = "SELECT DISTINCT obj.regiaoGeografica FROM Crime AS obj ORDER BY obj.regiaoGeografica")
21     List<String> mostrarRegioes();
22
23     @Query(value = "SELECT DISTINCT obj.natureza FROM Crime AS obj ORDER BY obj.natureza")
24     List<String> mostrarNaturezas();
25
26     @Query(value = "SELECT DISTINCT obj.ano FROM Crime AS obj ORDER BY obj.ano")
27     List<String> mostrarAnos();
28
29     @Query(value = "SELECT DISTINCT obj.sexo FROM Crime AS obj ORDER BY obj.sexo")
30     List<String> mostrarSexos();
31
32     @Query(value = "SELECT DISTINCT obj.faixaEtaria FROM Crime AS obj ORDER BY obj.faixaEtaria")
33     List<String> mostrarFaixasEtarias();
34
35     @Query(
36         value = "SELECT DISTINCT obj.totalDeEnvolvidos FROM Crime AS obj ORDER BY obj.totalDeEnvolvidos")
37     List<Integer> mostrarTotalDeEnvolvidos();
38
39     @Query(value = "SELECT obj FROM Crime as obj where obj.dataDoFato = :dataDoFato")
40     List<Crime> mostrarCrimesPorData(LocalDate dataDoFato);
41
42     @Query(value = "SELECT obj.dataDoFato FROM Crime as obj ORDER BY obj.dataDoFato DESC")
43     List<LocalDate> mostrarUltimaData();
44
45     @Query(value = "SELECT * FROM tb_crimes ORDER BY data_do_fato DESC limit :limit",
46         nativeQuery = true)
47     List<Crime> mostrarDezUltimosCrimes(@Param("limit") int limit);
48
49 }
```

Figura 15 – Os métodos do repositório de crimes.

```
50     @Query(
51         value = "SELECT obj.natureza as natureza, count(obj.natureza) as quantidade FROM tb_crimes obj where ano = :ano group by natureza",
52         nativeQuery = true)
53     List<Map<String, Integer>> mostrarQuantidadeDeCrimesPorNatureza(@Param("ano") int ano);
54 }
```

Figura 16 – Último método do repositório de crimes.

Na interface “*CrimeRepository*”, é usada a anotação *@Repository* para definir a interface como pertencente à camada de persistência (ALLET, 2013). Essa interface herda da poderosa interface “*JpaRepository*”, que possui métodos para a persistência dos dados em um BD. Essa interface tem métodos, e cada um deles possui uma query customizada. Todos os métodos da interface, exceto por “mostrarDezUltimosCrimes”,

“mostrarUltimaData”, e “mostrarQuantidadeDeCrimesPorNatureza”, retornam os valores em cada atributo da classe de Crime. O método “mostrarDezUltimosCrimes” retornam os dez últimos crimes registrados na planilha de microdados. O método “mostrarUltimaData” retorna a última data registrada na planilha de microdados de crimes. O método “mostrarQuantidadeDeCrimesPorNatureza” retorna a quantidade de crimes por natureza tendo como parâmetro, o ano em que esses crimes aconteceram. Cada método possui uma anotação `@Query`, essa anotação tem um atributo obrigatório chamado “value” que armazena o valor da *query* e um opcional chamado “nativeQuery” para ser uma *Query* do PostgreSQL. As figuras 9 e 10 mostram a *interface* e as assinaturas dos métodos. O repositório do *backend* do projeto está no repositório no *GitHub* [backend-tcc](#).

Na classe “*CrimeController*” são usadas três anotações:

1. **@RestController** - Anotação que provê dicas para quem está lendo o código e para o *Spring* que aquela classe anotada tem uma papel específico. Nesse caso é um controlador *web*, então o *Spring* considera isso quando se trata requisições *HTTP* (WEBB DAVE SYER, 2012);
2. **@RequestMapping** - Anotação que provê informação de “rotas”. Essa anotação conta ao *Spring* que cada requisição *HTTP* mapeada com um “/” pode ser mapeada para o método *home* (WEBB DAVE SYER, 2012);
3. **@CrossOrigin** - Anotação que habilita um mecanismo utilizado pelos navegadores para compartilhar recursos entre diferentes origens (MACHADO, 2012).

Nas figuras 11 até 13, ficam os métodos do *controller* de crimes. Cada um deles chama o *service* de crimes e retorna - exceto por um método chamado *mostrarUltimaData*, que retorna somente uma data - uma lista, podendo ser de *strings* ou, crimes ou um *map* onde a chave é o atributo do crime e o valor é a quantidade. Em todos os métodos do *controller*, é retornado um objeto da classe *ResponseEntity*, esse objeto é responsável por retornar toda a resposta de uma requisição *HTTP*, com o *status code*, cabeçalhos e o corpo da requisição. Esse objeto é genérico, podendo retornar qualquer tipo de dado no corpo da requisição (FEJÉR, 2021).


```
46 @RestController
47 @CrossOrigin
48 @RequestMapping("/crimes")
49 public class CrimeController {
50
51     @Autowired
52     CrimeService service;
53
54     @GetMapping(value = "/cidades")
55     public ResponseEntity<List<String>> mostrarCidades() {
56         List<String> list = service.mostrarCidades();
57         return ResponseEntity.ok(list);
58     }
59
60     @GetMapping(value = "/regioes")
61     public ResponseEntity<List<String>> mostrarRegioes() {
62         List<String> list = service.mostrarRegioes();
63         return ResponseEntity.ok(list);
64     }
65
66     @GetMapping(value = "/naturezas")
67     public ResponseEntity<List<String>> mostrarNaturezas() {
68         List<String> list = service.mostrarNaturezas();
69         return ResponseEntity.ok(list);
70     }
71
72     @GetMapping(value = "/anos")
73     public ResponseEntity<List<String>> mostrarAnos() {
74         List<String> list = service.mostrarAnos();
75         return ResponseEntity.ok(list);
76     }
77
78     @GetMapping(value = "/sexos")
79     public ResponseEntity<List<String>> mostrarSexos() {
80         List<String> list = service.mostrarSexos();
81         return ResponseEntity.ok(list);
82     }
83 }
```

Figura 17 – Os métodos do *controller* de crimes.

```
54 @GetMapping(value = "/faixas-etarias")
55 public ResponseEntity<List<String>> mostrarFaixasEtarias() {
56     List<String> list = service.mostrarFaixasEtarias();
57     return ResponseEntity.ok(list);
58 }
59
60 @GetMapping(value = "/total-envolvidos")
61 public ResponseEntity<List<Integer>> mostrarTotalEnvolvidos() {
62     List<Integer> list = service.mostrarTotalDeEnvolvidos();
63     return ResponseEntity.ok(list);
64 }
65
66 @GetMapping(value = "/crimes-por-data")
67 public ResponseEntity<List<Crime>> mostrarCrimesPorData(
68     @DateTimeFormat(iso = DateTimeFormat.ISO.DATE) LocalDate dataDoFato) {
69     List<Crime> list = service.mostrarCrimesPorData(dataDoFato);
70     return ResponseEntity.ok(list);
71 }
72
73 @GetMapping(value = "/ultima-data")
74 public ResponseEntity<LocalDate> mostrarUltimaData() {
75     LocalDate ultimaData = service.mostrarUltimaData().get(0);
76     return ResponseEntity.ok(ultimaData);
77 }
78
79 @GetMapping(value = "/dez-ultimos-crimes")
80 public ResponseEntity<List<Crime>> mostrarDezUltimosCrimes() {
81     List<Crime> listaDezUltimosCrimes = service.mostrarDezUltimosCrimes();
82     return ResponseEntity.ok(listaDezUltimosCrimes);
83 }
84
85 @GetMapping(value = "/qtd-crimes-natureza-ano")
86 public ResponseEntity<List<Map<String, Integer>>> mostrarQuantidadeDeCrimesPorNatureza(int ano) {
87     List<Map<String, Integer>> qtdCrimesPorNaturezaEAno =
88     service.mostrarQuantidadeDeCrimesPorNatureza(ano);
89     return ResponseEntity.ok(qtdCrimesPorNaturezaEAno);
90 }
91 }
```

Figura 18 – Outros métodos do *controller* de crimes.


```
92 @GetMapping(value = "/qtd-crimes-regiao-ano")
93 public ResponseEntity<List<Map<String, Integer>>> mostrarQuantidadeDeCrimesPorRegiaoGeografica(
94     int ano) {
95     List<Map<String, Integer>> qtdCrimesPorRegiaoEAno =
96         service.mostrarQuantidadeDeCrimesPorRegiaoGeografica(ano);
97     return ResponseEntity.ok(qtdCrimesPorRegiaoEAno);
98 }
99
100 @GetMapping(value = "/qtd-crimes-municipio-ano")
101 public ResponseEntity<List<Map<String, Integer>>> mostrarQuantidadeDeCrimesPorMunicipioDoFato(
102     int ano) {
103     List<Map<String, Integer>> qtdCrimesPorMunicipioEAno =
104         service.mostrarQuantidadeDeCrimesPorMunicipioDoFato(ano);
105     return ResponseEntity.ok(qtdCrimesPorMunicipioEAno);
106 }
107
108 @GetMapping(value = "/qtd-crimes-ano")
109 public ResponseEntity<List<Map<String, Integer>>> mostrarQuantidadeDeCrimesPorMunicipioDoFato() {
110     List<Map<String, Integer>> qtdCrimesPorAno = service.mostrarQuantidadeDeCrimesPorAno();
111     return ResponseEntity.ok(qtdCrimesPorAno);
112 }
113
114 @GetMapping(value = "/qtd-crimes-sexo-ano")
115 public ResponseEntity<List<Map<String, Integer>>> mostrarQuantidadeDeCrimesPorSexo(int ano) {
116     List<Map<String, Integer>> qtdCrimesPorSexo = service.mostrarQuantidadeDeCrimesPorSexo(ano);
117     return ResponseEntity.ok(qtdCrimesPorSexo);
118 }
119
120 @GetMapping(value = "/qtd-crimes-faixa-etaria-ano")
121 public ResponseEntity<List<Map<String, Integer>>> mostrarQuantidadeDeCrimesPorFaixaEtaria(
122     int ano) {
123     List<Map<String, Integer>> qtdCrimesPorSexo =
124         service.mostrarQuantidadeDeCrimesPorFaixaEtaria(ano);
125     return ResponseEntity.ok(qtdCrimesPorSexo);
126 }
```

Figura 19 – Últimos métodos do *controller* de crimes.

3.4 Implementação

A plataforma foi implementada para ser intuitiva. Na página, existe um dashboard com uma tabela mostrando os dez últimos crimes de Violência Doméstica registrados na planilha e a quantidade registrada do total de crimes por ano. O *frontend* da plataforma foi desenvolvido na biblioteca *React* utilizando a linguagem *typescript*, usando como framework de *UI* o *React-Bootstrap*, para construir os gráficos, foi necessário o uso da biblioteca *apex-charts*, e para consumir os *endpoints* da API de crimes foi utilizado o cliente de requisições *HTTP* *axios*. No *dashboard* existe um *Dropdown* pedindo para selecionar o ano, que vai de 2015 até 2021, por padrão, o ano escolhido é o de 2015, mas ao selecionar um ano diferente, os valores nos gráficos mudam de acordo com o ano. Os gráficos mostram o quantitativo de crimes por: região, gênero da vítima, natureza, município, e faixa etária das vítimas. O código do *frontend* do projeto está nesse repositório no *GitHub* [frontend-tcc](#).

Dez mais recentes Crimes de Violência Doméstica no Estado de Pernambuco

MUNICÍPIO	REGIÃO	NATUREZA	DATA	ANO	SEXO	FAIXA ETÁRIA	TOTAL DE ENVOLVIDOS
ALIANCA	ZONA DA MATA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	35-64	1
ANGELIM	AGRESTE	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	35-64	1
AGRESTINA	AGRESTE	DANO POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	30-34	1
ALIANCA	ZONA DA MATA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	18-24	1
ALIANCA	ZONA DA MATA	INJÚRIA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	18-24	1
ANGELIM	AGRESTE	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	18-24	1
AFOGADOS DA INGAZEIRA	SERTÃO	LESÃO CORPORAL POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	35-64	1
AGRESTINA	AGRESTE	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	30-34	1
ALAGOINHA	AGRESTE	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	35-64	1
ARACOIABA	REGIÃO METROPOLITANA	AMEAÇA POR VIOLÊNCIA DOMÉSTICA/FAMILIAR	31/10/2021	2021	FEMININO	18-24	1

Quantidade de Crimes de Violência Doméstica cometidos por ano

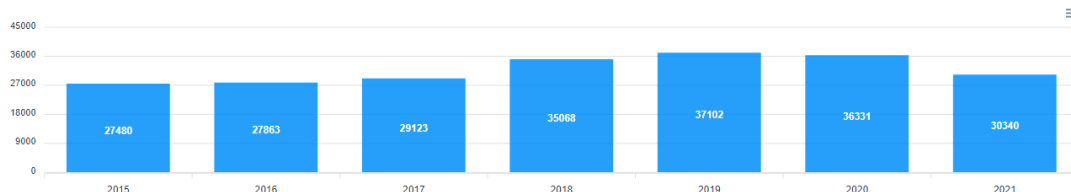


Figura 20 – Tabela mostrando os dez mais recentes crimes de Violência Doméstica no Estado de Pernambuco. Em seguida, está o gráfico que mostra a quantidade total de crimes de Violência Doméstica no Estado de Pernambuco por ano.

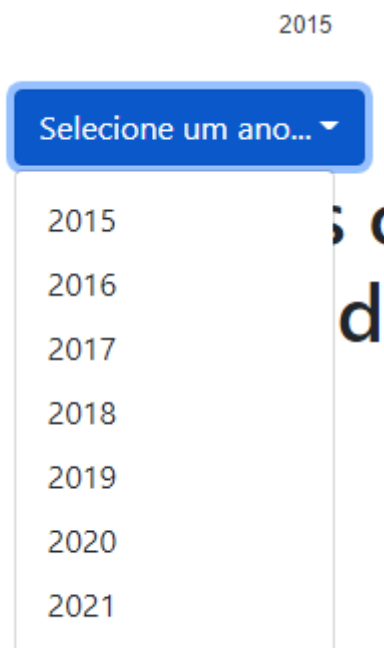


Figura 21 – Dropdown que mostra os anos selecionáveis

Regiões onde a incidência de crimes de violência doméstica é maior no ano de 2015



Crimes de Violência Doméstica Cometidos em 2015

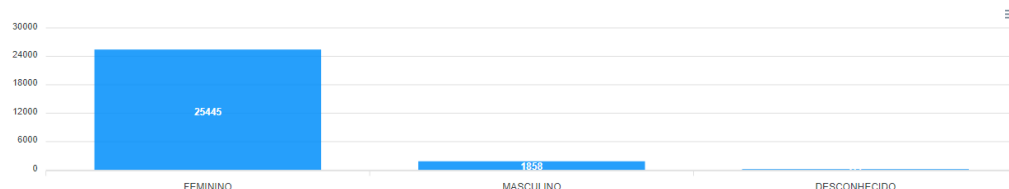
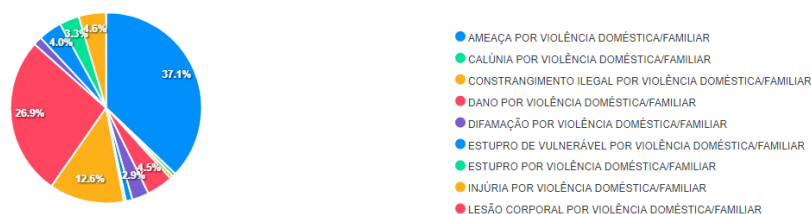


Figura 22 – Gráfico de rosca mostrando as regiões do Estado de Pernambuco onde acontecem mais crimes de violência doméstica. Em seguida, está um gráfico de barra vertical onde é mostrado o quantitativo de crimes de violência doméstica no Estado de Pernambuco

Naturezas de crimes mais cometidos em 2015



Municípios onde mais ocorreram mais Crimes de Violência Doméstica Cometidos em 2015

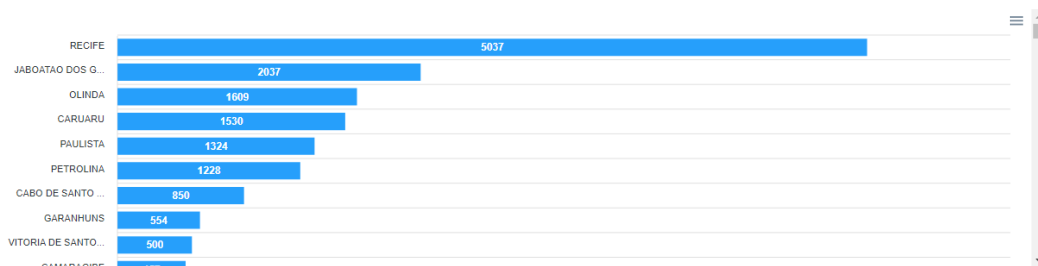


Figura 23 – Gráfico de pizza mostrando o quantitativo de crimes de Violência Doméstica por natureza. Em seguida, está um gráfico de barra vertical onde é mostrado o quantitativo de crimes de Violência Doméstica por natureza.

Faixas Etárias das vítimas de Crimes de Violência Doméstica no ano de 2015



Figura 24 – Gráfico de rosca mostrando o quantitativo de crimes de Violência Doméstica por faixa etária da vítima.

4 Conclusão

Neste trabalho, foi realizada a implementação de uma plataforma *web* que mapeia os dados de crimes de Violência Doméstica no Estado de Pernambuco. O objetivo foi tornar as informações na planilha de microdados, fornecida pela SDS-PE, mais transparentes para a sociedade, por meio da tradução de uma linguagem verbal para não-verbal através de gráficos e tabela. Por meio de quatro objetivos específicos, foi proporcionada uma ferramenta que pode ajudar a sociedade a cobrar mais investimentos nas políticas públicas de prevenção e combate à Violência Doméstica.

4.1 Trabalhos Relacionados

Uma iniciativa, que cobre crimes de um modo mais generalizado, ocorre no Reino Unido. Desde dezembro de 2008, as forças policiais da Inglaterra e do País de Gales unem esforços para publicar estatísticas de crimes usando uma ferramenta de mapeamento de crimes *online* (CHAINEY, 2012).

(SOARES et al., 2017) propôs uma plataforma que mapeia os Crimes Violentos Letais Intencionais (CVLI) no Estado de Pernambuco. Nesse trabalho as informações dos crimes, que eram fornecidas no formato *PDF*, passavam por um pré-processamento de conversão para um formato aberto muito utilizado por cientistas de dados, o formato *CSV*.

Neste trabalho, diferentemente dos acima citados, tem como objetivo transparecer a situação da violência doméstica, onde membros da família são os agressores. Sendo esse um crime difícil de se combater, pois acontece no interior dos lares, geralmente, sem a presença de testemunhas, e por não deixarem rastros.

4.2 Considerações finais

O Governo do Estado de Pernambuco, por meio da SDS-PE entende que publicar dados para a população estimula a criação e inovação de soluções para todos, estado e sociedade. Mas essa divulgação não é perfeita e sempre terá problemas, foi necessário um processamento dos dados para armazenar em um BD, pois o *Spring Boot* não tem embutido um *plugin* para ler arquivos em *XLSX*. Mesmo depois de ter lido a planilha, existiam alguns dados inconsistentes, como por exemplo, alguns valores eram escritos como "(vazio)", o que precisou ser alterado. Depois de resolvido o problema da leitura da planilha, foi possível "transformar" a planilha de microdados de

crimes de violência doméstica em uma *API* de crimes onde é possível filtrar por até dois atributos, onde um deles é o ano.

4.3 Trabalhos futuros

Como esse trabalho possui de um escopo fechado, ficam algumas sugestões para trabalhos futuros.

1. Filtrar por mais de 2 atributos dos crimes;
2. Mostrar o mapa do estado de Pernambuco, onde cada cidade é clicável e poder ver estatísticas da cidade e comparar com outras da mesma região;
3. Comparar dados de crimes de violência doméstica com os demais estados da federação.
4. Fornecer informações às vítimas sobre onde encontrar ajuda, para que sejam devidamente acolhidas e encorajadas a denunciar seus agressores, para assim quebrar esse ciclo de violência.
5. Automatizar o processo de obtenção do arquivo, pois é feito com um processo manual de baixar o arquivo, e colocar na pasta correta do projeto.

Referências

ALLET, A. *Introdução prática ao Spring Framework com uso de Anotações*. 2013. Disponível em: <<https://www.devmedia.com.br/introducao-pratica-ao-spring-framework-com-uso-de-annotacoes/27859>>.

ARAÚJO, C. *Entendendo Anotações em Java*. 2012. Disponível em: <<https://www.devmedia.com.br/entendendo-annotacoes-em-java/26772>>.

BRASIL, C. N. do. *Artigo 144 da Constituição Federal de 1988*. 1988. Disponível em: <<https://www.jusbrasil.com.br/topicos/10673132/artigo-144-da-constituicao-federal-de-1988>>.

BURÉGIO, V.; MEIRA, S.; ROSA, N. Social machines: A unified paradigm to describe social web-oriented systems. In: . [S.l.: s.n.], 2013.

CARVALHO, A. A. A. *Manual de ferramentas da Web 2.0 para professores*. [S.l.]: Ministério da Educação e Ciência. Direção Geral de Inovação e ..., 2008.

CHAINEDY, L. T. S. *Engagement, empowerment and transparency: publishing crime statistics using online crime mapping*. 2012.

DAHLBERG, L. L.; KRUG, E. G. Violence: a global public health problem. *Ciência & Saúde Coletiva*, SciELO Brasil, v. 11, 2006.

FEJÉR, A. *Using Spring ResponseEntity to Manipulate the HTTP Response*. 2021. Disponível em: <<https://www.baeldung.com/spring-response-entity>>.

FIGUEIREDO, E. *Padrões arquiteturais*. 2011. Disponível em: <https://homepages.dcc.ufmg.br/~figueiredo/disciplinas/aulas/padroes-arquiteturais_v01.pdf>.

MACHADO, G. *O que é CORS e como resolver os principais erros*. 2012. Disponível em: <<https://www.treinaweb.com.br/blog/o-que-e-cors-e-como-resolver-os-principais-erros>>.

MAVUDURUI, A. *Is Data Really the New Oil in the 21st Century?* 2020. Disponível em: <<https://towardsdatascience.com/is-data-really-the-new-oil-in-the-21st-century-17d014811b88>>.

NOLETO, C. *POO: tudo sobre Programação Orientada a Objetos!* 2020. Disponível em: <<https://blog.betrybe.com/tecnologia/poo-programacao-orientada-a-objetos/>>.

NOTÍCIAS, A. C. de. *Justiça de Pernambuco abre novo canal contra violência doméstica*. 2020. Disponível em: <<https://www.cnj.jus.br/justica-de-pernambuco-abre-novo-canal-contraviolencia-domestica>>.

PAULO, P. P. *Uma em cada quatro mulheres foi vítima de algum tipo de violência na pandemia no Brasil, aponta pesquisa*. 2021. Disponível em: <<https://g1.globo.com/sp/sao-paulo/noticia/2021/06/07/1-em-cada-4-mulheres-foi-vitima-de-algum-tipo-de-violencia-na-pandemia-no-brasil-diz-datafolha.html>>.

PENHA, I. M. da. *O QUE É VIOLÊNCIA DOMÉSTICA*. 2018. Disponível em: <<https://www.institutomariadapenha.org.br/violencia-domestica/o-que-e-violencia-domestica.html>>.

PENHA, I. M. da. *TIPOS DE VIOLÊNCIA*. 2018. Disponível em: <<https://www.institutomariadapenha.org.br/lei-11340/tipos-de-violencia.html>>.

SOARES, E.; DELGADO, A.; BURÉGIO, V.; BRITO, K.; FRANÇA, C. *Sistema Web para Mapeamento de Dados de Crimes Letais no Estado de Pernambuco*. 2017.

TI, C. *Arquitetura cliente-servidor*. 2018. Disponível em: <<https://www.canalti.com.br/arquitetura-de-computadores/arquitetura-cliente-servidor/>>.

TOWARDS Government as a Social Machine. 2015.

WANG, X.; FANG, Y.; LIU, Y.; HORN, B. *A Survey on the Status of Open Data and Its Future*. 2018.

WARTELL, J.; MCEWEN, J. *Privacy in the Information Age: A Guide for Sharing Crime Maps and Spatial Data*. [S.l.: s.n.], 2001.

WEBB DAVE SYER, J. L. P. *Spring Boot Reference Documentation*. 2012. Disponível em: <<https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>>.

ZUCHER, V. *O que é padrão MVC? Entenda arquitetura de softwares!* 2020. Disponível em: <<https://www.lewagon.com/pt-BR/blog/o-que-e-padrao-mvc>>.