



GRADUANDO EM ENGENHARIA DE SOFTWARE  
TÉCNICO EM ELETROELETRÔNICA

NATAN OGLIARI

**PROGRESSO DE IMPLANTAÇÃO DA PLATAFORMA INTEGRADA DE  
AÇÕES PARA ENFRENTAMENTO À VIOLÊNCIA DOMÉSTICA E  
CRIMES SEXUAIS**

Maravilha/SC

2025

NATAN OGLIARI

**PROGRESSO DE IMPLANTAÇÃO DA PLATAFORMA INTEGRADA DE  
AÇÕES PARA ENFRENTAMENTO À VIOLÊNCIA DOMÉSTICA E  
CRIMES SEXUAIS**

Documento elaborado para prestar esclarecimentos sobre o andamento da implantação do Progresso de implantação da **Plataforma Integrada de Ações para Enfrentamento à Violência Doméstica e Crimes Sexuais** .

.

Maravilha/SC

2025

# Sumário

	Páginas
<b>1 Introdução</b>	<b>5</b>
<b>2 Tecnologias utilizadas</b>	<b>5</b>
2.1 Framework: Django . . . . .	6
2.2 Banco de dados: PostgreSQL . . . . .	6
2.3 Nginx . . . . .	6
2.4 Tailwind CSS . . . . .	7
2.5 Django Channels e Redis . . . . .	7
2.6 Smart Selects . . . . .	7
2.7 Chart.js . . . . .	7
2.8 Leaflet.js com OpenStreetMap . . . . .	8
2.9 jQuery . . . . .	8
2.10 Ollama: Execução Local de Modelos de Linguagem . . . . .	8
<b>3 Implementação para testes e validações</b>	<b>8</b>
3.1 Dos Scripts . . . . .	9
3.2 Hostinger . . . . .	9
3.2.1 Aplicação SaaS . . . . .	10
3.2.2 O Banco de dados . . . . .	10
3.2.3 O domínio . . . . .	11
<b>4 Métodos</b>	<b>11</b>
4.1 Estrutura de Aplicações Django . . . . .	11
4.2 Técnicas e Práticas de Desenvolvimento . . . . .	12
<b>5 Resultados</b>	<b>14</b>
5.1 Assistente virtual com IA . . . . .	19
<b>6 Conclusão</b>	<b>20</b>
<b>7 Anexos</b>	<b>23</b>

**Lista de Algoritmos**

1	Interação com o Modelo llama3:70b, tempo de resposta: 224.82 [s]	20
---	--	----

**Lista de Figuras**

1	logo PIEVDCS.	5
2	Telas da plataforma, O autor	15
3	Telas estatísticas, O autor	17
4	Telas institucionais, O autor	18
5	Modelo llama3:70b, O autor	19
6	Diploma de Técnico	23
7	Extrato de disciplinas	24

# 1 Introdução

**Figura 1.** logo PIEVDCS.

A Plataforma Integrada de Informações para Enfrentamento à Violência Doméstica e Crimes Sexuais (PI-EVDCS) é uma solução digital desenvolvida com base no *framework* Django, no modelo SaaS (Software como Serviço). Seu objetivo é integrar instituições como segurança pública, Ministério Público, Poder Judiciário, Defensoria Pública e serviços municipais de assistência, promovendo agilidade e efetividade no atendimento às vítimas e concientização dos agressores.



Fonte: O autor (2025)

Além de promover agilidade e efetividade no atendimento às vítimas, a plataforma representa uma ferramenta estratégica para romper o ciclo intergeracional da violência, que frequentemente se perpetua de pais para filhos. Estudos como o da Universidade Federal do Ceará, em parceria com a ONU Mulheres, revelam que quatro em cada dez mulheres que cresceram em lares violentos vivenciam o mesmo padrão na vida adulta, evidenciando a urgência de intervenções sistêmicas e integradas (CARVALHO; OLIVEIRA, 2017). Ao conectar instituições e facilitar o fluxo de informações, a plataforma não apenas melhora o presente - ela transforma o futuro, criando condições para que novas gerações cresçam em ambientes mais seguros, protegidos e livres da normalização da violência doméstica.

O desenvolvimento da plataforma foi viabilizado por meio de verba oriunda do Poder Judiciário da Comarca de Maravilha/SC. Um aspecto singular do projeto é a participação direta de um reeducando do Presídio Regional de Maravilha como responsável técnico pelo desenvolvimento da solução. O reeducando, com formação técnica, atua sob supervisão institucional, aplicando seus conhecimentos de forma produtiva e qualificada, demonstrando que a ressocialização é viável e impactante.

Este relatório documenta o progresso técnico da implementação, destacando as funcionalidades já desenvolvidas, os métodos utilizados e as etapas pendentes até a entrega final.

## 2 Tecnologias utilizadas

A escolha das tecnologias adotadas neste projeto foi pautada, sobretudo, por seu caráter *Open Source* (código aberto), o que proporciona significativa economia ao eliminar custos com licenciamento. Dessa forma, os investimentos se concentram exclusivamente em aspectos de

infraestrutura, como servidores e equipe técnica especializada.

Além disso, o desenvolvedor considera que as informações sensíveis contempladas por esta plataforma devem permanecer sob responsabilidade do Estado, não devendo ser compartilhadas com empresas privadas, como serviços de hospedagem em nuvem ou soluções de inteligência artificial de terceiros. Ressalta-se, contudo, que, na fase de validação, a plataforma está hospedada em serviços de terceiros.

## 2.1 Framework: Django

Segundo Django (2025), Django é um framework de desenvolvimento web escrito em Python, amplamente reconhecido por acelerar a criação de aplicações robustas e escaláveis. Seu diferencial está na abordagem *"batteries included"*, oferecendo nativamente funcionalidades essenciais como autenticação, painel administrativo, mapeamento objeto-relacional (ORM) e roteamento de URLs (*Uniform Resource Locator*).

Outro fator decisivo na escolha do Django é sua natureza nativa em Python, o que possibilita fácil integração com bibliotecas de *machine learning*<sup>1</sup>, como `scikit-learn`, `TensorFlow`, `PyTorch`, `Ollama`, dentre outras. Isso viabiliza, em futuras versões do sistema, a implementação de modelos preditivos capazes de analisar parâmetros operacionais e sugerir ações estratégicas de maneira automatizada e inteligente.

## 2.2 Banco de dados: PostgreSQL

PostgreSQL (2025) é um sistema de gerenciamento de banco de dados objeto-relacional de código aberto, com mais de 35 anos de desenvolvimento ativo, considerado um dos mais avançados e robustos do mercado. Conta com integração nativa ao Django.

Em termos de gerenciamento de dados, o PostgreSQL oferece suporte a extensões como PostGIS, que adiciona funcionalidades geoespaciais essenciais para a análise e visualização de dados geográficos, como mapas de ocorrências de violência doméstica. Além disso, sua arquitetura robusta e escalável permite lidar com grandes volumes de dados, garantindo integridade, segurança e desempenho, aspectos cruciais para uma plataforma que visa integrar múltiplas instituições e gerenciar informações sensíveis.

## 2.3 Nginx

O F5, Inc. (2025) é um servidor web de alto desempenho e proxy reverso, amplamente utilizado para servir aplicações web em ambientes de produção. No contexto deste projeto, o Nginx será configurado como servidor HTTP para a versão de produção, substituindo o servidor de desenvolvimento do Django. Suas principais vantagens incluem alto desempenho,

---

<sup>1</sup> Aprendizado de máquina

baixo consumo de recursos, capacidade de lidar com grande número de conexões simultâneas e funcionalidades avançadas de balanceamento de carga e cache.

## 2.4 Tailwind CSS

O TailwindCSS (2025) é um framework de estilo baseado em classes utilitárias, que permite o desenvolvimento rápido de interfaces personalizadas e responsivas. Sua abordagem facilita a manutenção e reutilização de estilos, promovendo consistência visual e produtividade no desenvolvimento front-end, mantendo os padrões de desenvolvimento de Experiência do Usuário/Interface do Usuário (UX/UI).

## 2.5 Django Channels e Redis

Para implementação de comunicação em tempo real na plataforma, foi adotado o **Django Channels** (Django Software Foundation, 2025), uma extensão do Django que adiciona suporte a protocolos assíncronos como WebSockets<sup>2</sup>. Esta tecnologia permite o envio e recebimento de notificações instantâneas entre usuários e instituições, fundamentais para a coordenação eficiente das ações de enfrentamento à violência doméstica.

Como *backend* de mensagens, utiliza-se o **Redis** (Redis Ltd., 2025), um armazenamento de estrutura de dados em memória, que atua como *broker* de mensagens garantindo alta performance e confiabilidade na entrega de notificações em tempo real. O sistema foi configurado com capacidade de 1500 conexões simultâneas e tempo de expiração de 10 segundos, assegurando escalabilidade e eficiência.

## 2.6 Smart Selects

A biblioteca **Smart Selects** (Django Smart Selects Contributors, 2025) foi integrada ao projeto para implementar campos de seleção encadeados (*chained selects*) nos formulários Django. Esta funcionalidade é essencial para garantir a consistência de dados, especialmente na seleção de municípios baseada no estado previamente escolhido, e na vinculação de municípios às suas respectivas comarcas do Poder Judiciário.

## 2.7 Chart.js

Chart.js (2025) é uma biblioteca JavaScript especializada na renderização de gráficos interativos e responsivos. Utilizada para a visualização dos dados gerados pela aplicação, permite criar diferentes tipos de gráficos (como barras, linhas, setores dentre outras opções), proporcionando clareza e insights para os usuários.

---

<sup>2</sup>protocolo que estabelece canais de comunicações bidirecionais e persistente entre um cliente e um servidor, permitindo a troca de dados em tempo real.

A biblioteca foi criada e mantida pelos Voluntários.  
Link: (<https://github.com/chartjs/Chart.js/graphs/contributors>).

## 2.8 Leaflet.js com OpenStreetMap

Leaflet.js (2025) é uma biblioteca JavaScript leve para a criação de mapas interativos. Em conjunto com a plataforma OpenStreetMap, oferece uma base cartográfica gratuita e de alta qualidade. No contexto do projeto, foi empregada para a representação geoespacial de dados, com funcionalidades como marcação de locais, camadas e interações de usuário.

## 2.9 jQuery

jQuery (2025) é uma biblioteca JavaScript amplamente utilizada para simplificar operações no DOM<sup>3</sup>, gerenciamento de eventos e requisições assíncronas (AJAX)<sup>4</sup>. No projeto, ela contribui para tornar a navegação mais fluida e as interações mais dinâmicas, além de facilitar a integração com outras bibliotecas e funcionalidades da aplicação.

## 2.10 Ollama: Execução Local de Modelos de Linguagem

A execução de Modelos de Linguagem de Grande Escala (LLMs) será realizada utilizando a plataforma Ollama, que permite rodar modelos como LLaMA, Mistral e Gemma diretamente no ambiente local, sem necessidade de conexão com servidores externos. Segundo Ollama (2025), a proposta da ferramenta é “facilitar a execução de grandes modelos de linguagem localmente”, o que contribui para maior privacidade e controle sobre os dados processados.

A execução dos modelos será realizada com suporte de uma GPU NVIDIA RTX 2000 Ada Generation, equipada com 16GB de memória dedicada, DDR6, 128bits. Embora não seja a mais potente da linha, essa placa gráfica oferece recursos suficientes para testes locais com modelos otimizados, como os disponíveis na plataforma Ollama. Seu uso viabiliza a aceleração das inferências, mesmo que com tempos de resposta mais elevados em modelos de maior complexidade.

# 3 Implementação para testes e validações

Para os testes iniciais e validações, as informações serão geradas automaticamente através de *script python* 3.1, por se tratar de dados vulneráveis, não serão utilizadas informações reais nesta etapa do processo.

---

<sup>3</sup>Document Object Model (Modelo de Objeto de Documento) permite alterar parte do conteúdo da página web sem a necessidade de recarregar toda a página.

<sup>4</sup>(Asynchronous JavaScript and XML) Permite comunicação em segundo plano com o servidor sem recarregar a página.



### 3.1 Dos Scripts

Possui uma seção denominada de **automacoes**, esta pasta é responsável pela centralização de todos os *script* de automações para a inserção de informações no banco de dados, sendo desde a configurações de Municípios, Comarcas até a inserção de valores fictícios para validação da plataforma com os usuários finais.

Funcionalidades já implementadas nos *script*

**Cria Estado** Função que cria no banco de dados todos os estados da federação e um estado **"EX- Extrangeiro"** - Script para configuração da plataforma, pode ser executado n vezes, pois só cria se não existir.

**Atribui Município** Script que atribui todos os municípios de cada respectivo estado, evitando duplicidade e relacionando com o estado pertinente, deverá ser executado após o "Cria Estado".

**Cria Comarca** Funcionalidade que cria todas as comarcas do estado Catarinense com base no site do Tribunal de Justiça Catarinense, <https://www.tjsc.jus.br/paginas-das-comarcas> - Funcionalidade de configuração - a futuro, complementar para a inserção de todas as comarcas da federação com seus respectivos *foro*. Função a ser executada após o "Atribui Município".

**Gera Vítimas** Cadastra aleatoriamente vítimas geradas randomicamente. Pode ser utilizada quantas vezes for necessárias.

**Gera Agressores** Cadastra aleatoriamente Agressores gerados randomicamente.

**Cadastro de MP e MPU** Gera aleatoriamente Medidas protetivas e Medidas protetivas de urgência e cadastra na seção competente a **Defensoria Pública**.

**Cria Grupo de Usuários** Cria grupos institucionais para que possa ser inserido usuários pertinentes as suas instituições e suas permissões. [*CAPS, CRAS, CREAS, Conselho Tutelar, Secretaria de Saúde, Polícia Penal, Polícia Militar, Polícia Civil, Polícia Científica, Poder Judiciário, Ministério Público e Defensoria Pública*].

### 3.2 Hostinger

A Hostinger é uma empresa global de hospedagem de sites que oferece serviços como hospedagem compartilhada, VPS (Servidor Virtual Privado), hospedagem em nuvem, registro de domínios e criador de sites. Fundada em 2004, a empresa se destacou no mercado por oferecer soluções acessíveis, com boa performance e suporte técnico multilíngue. Atualmente, a Hostinger atende milhões de clientes em todo o mundo e se tornou uma das principais opções para

desenvolvedores, empreendedores e pequenas empresas que buscam presença online confiável e de fácil gerenciamento (Hostinger International Ltd., 2025).

Esta seção será subdividida em três partes: a aplicação da plataforma como serviço, o banco de dados em *PostgreSQL* e o domínio de acesso via rede mundial de computadores.

### 3.2.1 Aplicação SaaS

A aplicação foi implementada em um servidor VPS (*Virtual Private Server*)<sup>5</sup> contratado junto à Hostinger, utilizando a infraestrutura como serviço (IaaS) oferecida pela plataforma. A instância VPS utilizada possui as seguintes configurações: **2 núcleos de CPU, 8 GB de memória RAM e 100 GB de armazenamento em disco SSD.**

O plano contratado teve início em **05 de agosto de 2025**, com vigência de **1 ano**, totalizando um custo de **R\$563,88** para o período. Essa infraestrutura garante a escalabilidade e estabilidade necessárias para o funcionamento da aplicação, permitindo seu acesso contínuo via rede pública.

A seguir, são descritos os principais recursos da VPS utilizada e o impacto de cada um no desempenho da aplicação:

- **CPU (2 núcleos):** A quantidade de núcleos da CPU influencia diretamente na capacidade de processamento paralelo do servidor. Com dois núcleos, é possível atender múltiplas requisições simultâneas de forma eficiente, garantindo responsividade adequada da aplicação em cenários de uso médio/moderado.
- **Memória RAM (8 GB):** A memória RAM é essencial para o carregamento de aplicações, bibliotecas, cache e o gerenciamento de múltiplas sessões de usuários. Com 8 GB, a aplicação possui espaço suficiente para operar com fluidez, mesmo sob cargas de trabalho maiores, como múltiplos acessos simultâneos.
- **Armazenamento (100 GB SSD):** O espaço em disco é utilizado para armazenar o código da aplicação, arquivos estáticos, banco de dados local e arquivos temporários. O uso de disco SSD, em comparação com HDs tradicionais, oferece maior velocidade de leitura e escrita, resultando em tempos de resposta mais rápidos e menor latência nas operações de I/O.

### 3.2.2 O Banco de dados

Com relação ao banco de dados, a sua configuração ficou a cargo do PostgreSQL 14, com acesso local e configurações passadas para o Django através de variáveis de ambientes.

---

<sup>5</sup>Servidor Virtual Privado: uma instância virtual isolada dentro de um servidor físico, com recursos próprios de CPU, RAM e armazenamento. Ideal para aplicações que exigem controle total do ambiente.

### 3.2.3 O domínio

O domínio a ser utilizado é o <https://www.redecontraaviolencia.org>, seu registro foi feito no dia 05 de agosto de 2025, terá validade de 1 ano (05/08/2026), o custo do domínio foi de R\$51,08 e o registro foi pela empresa Hostinger. Podendo a aplicação ser acessada através de seu IP<sup>6</sup> (Protocolo de Internet), e de sua porta de serviço. 62.72.9.77:8000.

Para a aplicação foi configurado um certificado TLS (Transport Layer Security), garantindo a segurança na comunicação entre o servidor e os usuários. O certificado é válido até 12/11/2025.

## 4 Métodos

A estrutura de desenvolvimento adotada é modular, baseada em aplicações Django (apps), com automações para carga de dados, visualização via dashboards<sup>7</sup> e controle granular de permissões por perfil institucional.

### 4.1 Estrutura de Aplicações Django

O projeto está organizado em aplicações Django independentes e interconectadas, seguindo o princípio de separação de responsabilidades:

**MAIN** Aplicação principal que gerencia a estrutura base do projeto, incluindo configurações globais, roteamento de URLs, página inicial com conteúdos dinâmicos e cálculo de variáveis estatísticas compartilhadas entre módulos.

**usuarios** Gerencia autenticação, autorização e perfis de usuários customizados (`CustomUser`). Implementa grupos personalizados (`CustomGroup`) com permissões específicas por instituição, validação de CPF, telefone e comprovantes de vínculo institucional em formato PDF.

**sistema\_justica** Centraliza os módulos do sistema de justiça, subdividido em:

- **Poder Judiciário:** Gestão de comarcas, cadastro de vítimas e agressores, integração com chatbot de IA;
- **Ministério Público:** Interface específica para promotores de justiça;
- **Defensoria Pública:** Formulários de solicitação de medidas protetivas de urgência (MPU) com mais de 30 campos detalhados sobre tipos de violência.

**seguranca\_publica** Engloba as instituições de segurança pública:

---

<sup>6</sup>IP é como se fosse o endereço da sua casa, mas na internet.

<sup>7</sup>Os dashboards são interfaces gráficas que permitem a visualização de dados de forma interativa e dinâmica.

- **Polícia Penal:** Registro de atendimentos a agressores (assistência social, psicologia, grupos reflexivos);
- **Polícia Civil:** Registro de ocorrências, investigações e flagrantes;
- **Polícia Militar:** Registro de atendimentos por tipo de patrulha (Radio Patrulha, Emergência 190, Policiamento Montado, etc.).

**município** Abrange serviços municipais de assistência:

- CRAS (Centro de Referência de Assistência Social);
- CREAS (Centro de Referência Especializado de Assistência Social);
- CAPS (Centro de Atenção Psicossocial);
- Conselho Tutelar;
- Secretaria da Saúde.

**mensageria** Sistema de notificações em tempo real utilizando Django Channels e WebSockets, permitindo envio de mensagens para usuários individuais ou grupos institucionais, com classificação de prioridade (normal, urgente, crítica) e status de leitura.

## 4.2 Técnicas e Práticas de Desenvolvimento

Algumas técnicas adotadas incluem:

- **Automação de Dados de Teste:** Scripts Python localizados no diretório `automacoes/` para geração automática de dados fictícios (vítimas, agressores, municípios, comarcas, medidas protetivas), permitindo validação realista da plataforma sem comprometer dados reais. Exemplos incluem `gera_formularios_mp.py`, `atribui_municipio.py` e `cria_grupos_usuarios.py`.
- **Modelos de Dados Complexos:** Implementação de relacionamentos avançados entre entidades utilizando:
  - `ChainedForeignKey` para seleções dependentes (município baseado em estado);
  - `ManyToManyField` para relações múltiplas (agressores em grupos de atendimento);
  - Cálculo automático de idade baseado em data de nascimento;
  - Formatação automática de CPF e validações customizadas.
- **Sistema de Notificações em Tempo Real:** Arquitetura baseada em WebSockets com Django Channels, permitindo:
  - Envio de notificações para usuários individuais ou grupos institucionais;

- Classificação por prioridade (normal, urgente, crítica);
- Rastreamento de status (não lida, lida, arquivada);
- APIs REST para contadores e listagens de notificações recentes.
- **Visualizações Dinâmicas:** Criação de dashboards interativos com *Chart.js* para análise de tipos de violência, medidas protetivas por comarca, e mapas geográficos com *Leaflet.js* para visualização espacial de ocorrências.
- **Controle Granular de Permissões:** Sistema de grupos customizados (*CustomGroup*) com permissões específicas por instituição, implementando diferentes níveis de acesso (visualização, adição, modificação) conforme o perfil institucional.
- **Templates Responsivos:** Interface desenvolvida com *TailwindCSS*, garantindo adaptabilidade a diferentes dispositivos (desktop, tablets, smartphones).
- **Segurança Avançada:**
  - Proteção CSRF<sup>8</sup>;
  - Configuração de CORS para APIs;
  - Sessões com tempo limite configurável (9000 segundos);
  - Cookies HTTP-only e SameSite;
  - Separação de variáveis sensíveis com *Dotenv*;
  - Sistema de *logging* customizado para auditoria.
- **Integração com IA Local:** Implementação e testes com modelos de linguagem via Ollama (*llama3:70b*, *mixtral*, *gpt-oss:120b*), com configurações em *settings.py* permitindo alternância entre modelos. Sistema de fallback com respostas demo quando Ollama não está disponível.
- **Decoradores Customizados:** Implementação de decoradores Python para:
  - Cálculo de tempo de execução de views (*@calcula\_tempo*);
  - Controle de acesso por grupo institucional (*@grupos\_permitidos*).

O desenvolvimento da plataforma foi iniciado em ambiente Windows 11, com ambiente virtual Python. Atualmente esta sendo desenvolvido no sistema operacional com base Linux (Ubuntu 24.04.3 LTS), tal migração se justifica e se sustenta em virtude da sua flexibilidade, autonomia na configuração de ambientes complexos e implementações de inteligências artificiais em ambiente de desenvolvimento local.

---

<sup>8</sup>Cross-Site Request Forgery (falsificação de solicitação entre sites)

Estão sendo implementados modelos de inteligência artificial utilizando o Ollama, com restrições específicas ao tema de violência doméstica e à Lei nº 11.340/2006 (Lei Maria da Penha). O objetivo é desenvolver uma assistente virtual capaz de oferecer atendimento qualificado às vítimas e ao público em geral, garantindo conformidade legal e segurança nas respostas.

Para fins de teste, está sendo utilizado o servidor de desenvolvimento do Django o (*Django server*<sup>9</sup>). Na versão de produção, o servidor elencado foi o **Nginx**. O *deploy* da aplicação esta sendo implementado na plataforma **Hostinger**, permitindo a validação da interface do usuário e de suas funcionalidades em ambiente real.

## 5 Resultados

A seguir estão listadas as principais funcionalidades já implementadas:

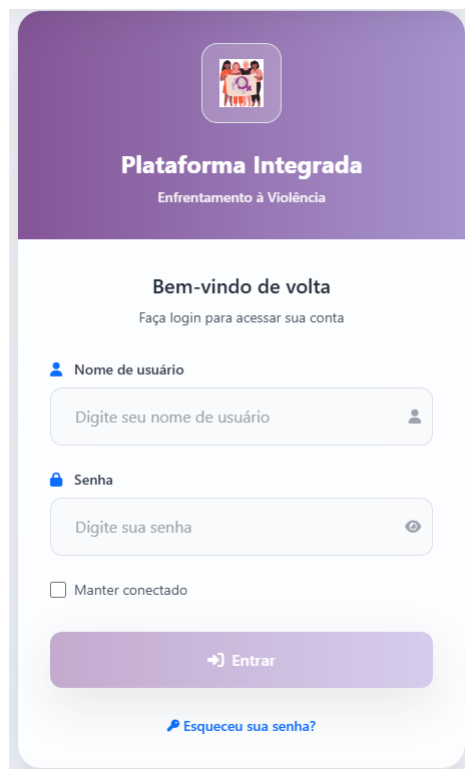
- **Estrutura de dados completa:** Modelos Django para vítimas, agressores, filhos, usuários, estados, municípios, comarcas e formulários especializados;
- **Sistema de autenticação robusto:** Login customizado com usuários estendidos (`CustomUser`), incluindo validação de CPF, telefone, data de nascimento e vinculação institucional;
- **Gestão de grupos e permissões:** Sistema automatizado de criação de grupos institucionais (12 grupos) com permissões granulares específicas por app (Sistema de Justiça, Segurança Pública, Município);
- **Formulário de Medida Protetiva extenso:** Implementado na Defensoria Pública com mais de 35 campos detalhados sobre tipos de violência, condutas de controle, ameaças, uso de tecnologia/IA para manipulação e violência vicária;
- **Registro de Ocorrências:** Polícia Militar (com 14 tipos de patrulha), Polícia Civil (investigação e flagrantes), Polícia Penal (atendimentos individuais e em grupo com agressores);
- **Sistema de Notificações em Tempo Real:** WebSocket com Django Channels, envio para usuários/grupos, prioridades (Normal, Urgente, Crítica), status (Não lida, Lida, Arquivada);
- **Dashboards estatísticos públicos:** Gráficos de tipos de violência, medidas protetivas por comarca, mapas geográficos (Leaflet.js + OpenStreetMap) com dados anonimizados;
- **Interface institucional diferenciada:** Cada instituição possui home page customizada com contador de encaminhamentos, notificações e acesso aos formulários específicos;
- **Chatbot com IA:** Interface para interação com modelos Ollama, sistema de fallback com respostas demo, configuração flexível de host e modelo;

---

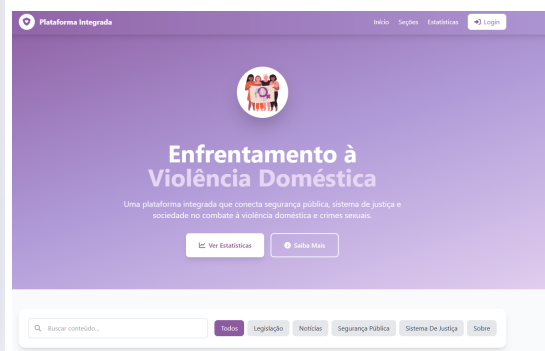
<sup>9</sup>Pode ser acessado dentro da rede do estado Catarinense através do 10.40.22.46:8000

- **Scripts de automação:** Criação de estados, municípios, comarcas de SC, geração de 500+ registros fictícios de MPU, criação automática de grupos com permissões;
- **Documentação:** Instruções técnicas e README detalhado de cada aplicação Django;
- **Páginas Concluídas:** Login (com recuperação de senha), Página Inicial pública (conteúdos dinâmicos), Relatórios públicos, Home institucional das 12 instituições, Formulários de cadastro, Interface de notificações.

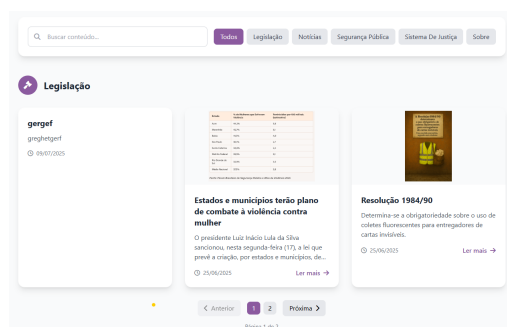
Incluo algumas telas da plataforma, a sequência das imagens são aleatórias.



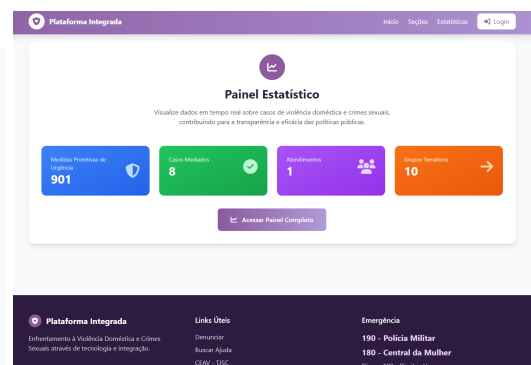
(a) Tela de Login.



(b) Página aberta, estilo one-page.



(c) Página aberta, estilo one-page 1.



(d) Página aberta, estilo one-page 2.

**Figura 2.** Telas da plataforma, O autor

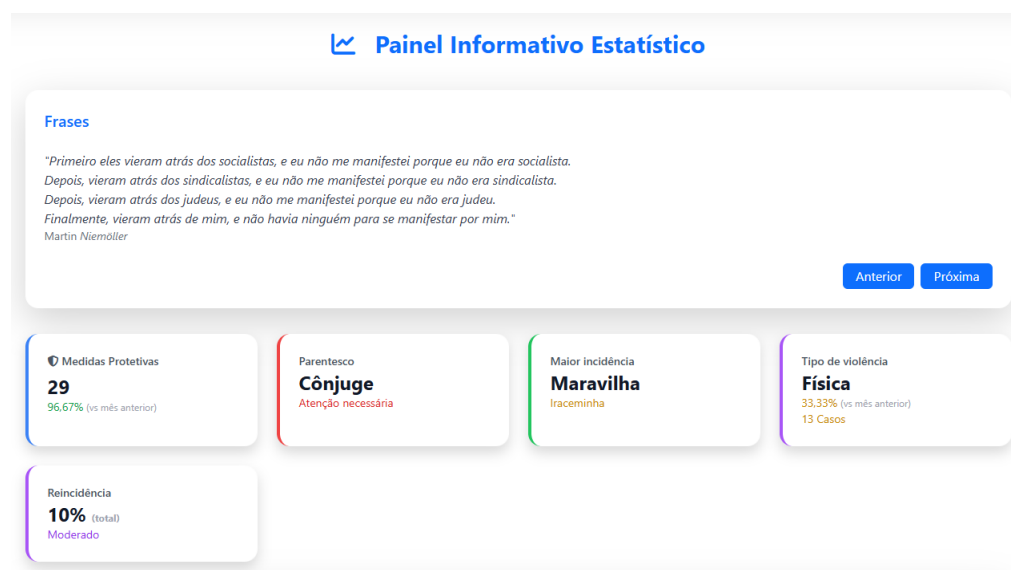
Foi desenvolvida uma seção pública de relatórios, permitindo que portais de notícias ou

entidades interessadas acessem estatísticas em tempo real. As informações apresentadas são reais e baseadas nos registros da plataforma, mas tratadas com rigor técnico para garantir o anonimato das vítimas e envolvidos. As localizações exibidas são referenciadas a centros comunitários ou órgãos públicos de cada bairro, evitando a identificação direta dos locais de ocorrência.

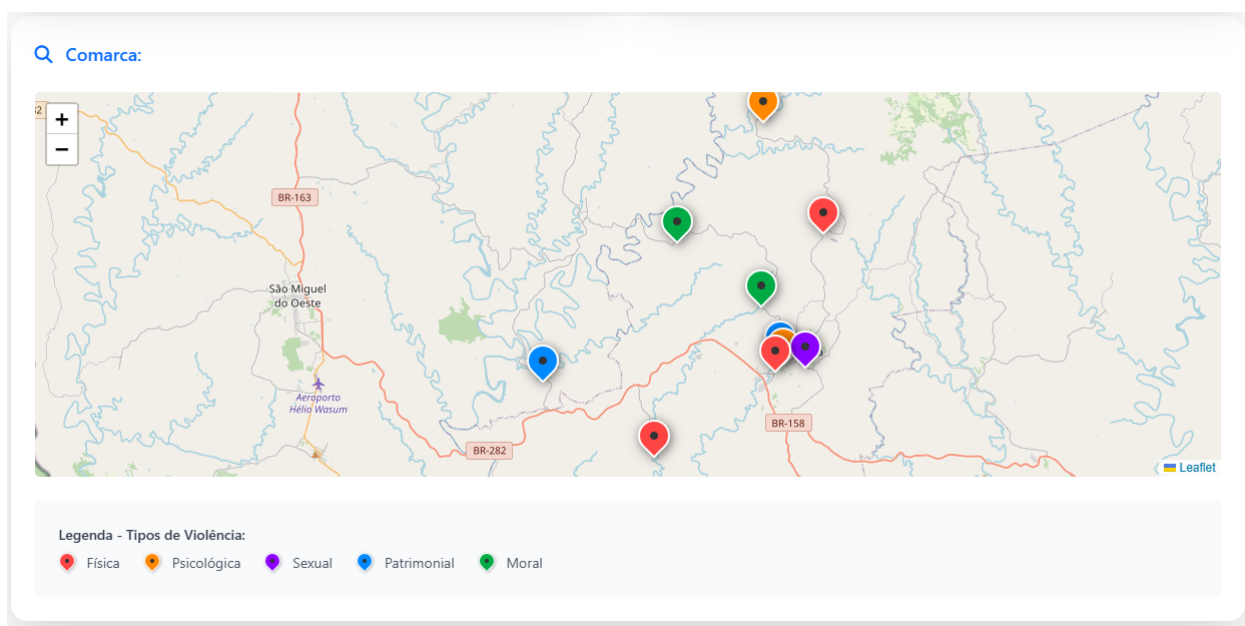




(a) Gráfico da Etnia e Classe Econômica. (b) Gráfico do Grau de Instrução e Grau de Parentesco.



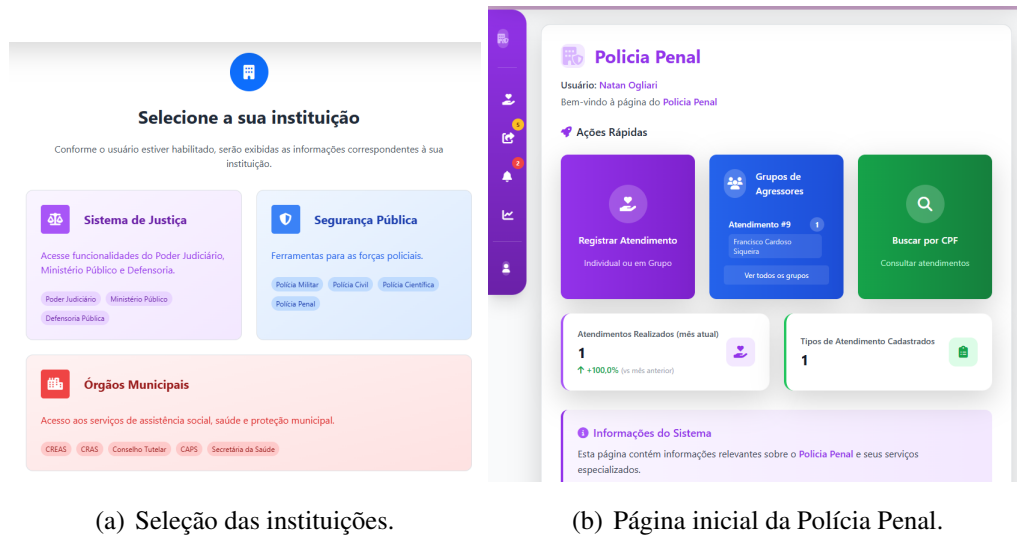
(c) Frases Motivacionais e alguns indicadores.



(d) Gráfico Geográfico dos tipos de violência.

**Figura 3.** Telas estatísticas, O autor

Esta sendo criada telas para cada instituição que terá acesso a plataforma e cada instituição terá determinado acesso e cada funcionalidade. Os *layout* das instituições ficará a cargo de cada representante da referida em ajustar com o desenvolvedor, para usabilidade e funcionalidades de cada instituição.



**Figura 4.** Telas institucionais, O autor

Funcionalidades em desenvolvimento:

- **Módulo de Atendimento Municipal** CRAS, CAPS, CREAS, Conselho Tutelar e Secretaria da Saúde
- **Módulo da segurança pública** PM, PP, PC, PCI
- **Integração Avançada entre Apps** Fluxos completos de encaminhamento entre órgãos
- **Notificações e Alertas** Sistema de notificações internas e externas
- **Painel Administrativo Avançado** Customização do admin para fluxos institucionais
- **Testes Automatizados** Cobertura de testes unitários e de integração
- **Aprimoramento de Relatórios** Novos filtros, exportação de dados, relatórios customizados por perfil
- Implantação em ambiente de produção e automação de deploy na plataforma *Hostinger*
- Sistema de mensagem com o *Django Channels*<sup>10</sup>

Funcionalidades Pendentes:

- Integração com sistemas externos (e-proc, MP, Defensoria, SISP)

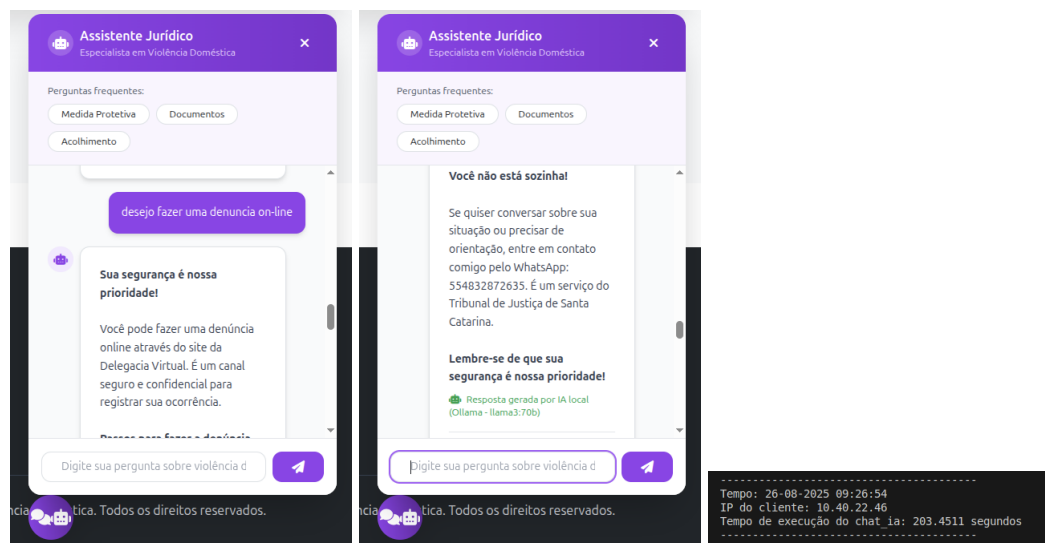
<sup>10</sup>Permite comunicação em tempo real, usado para troca de notificações entre usuários e instituições.

- Documentação Técnica Detalhada (Swagger/OpenAPI)
- Aprimoramento de UX/UI para acessibilidade
- Aprimoramento de Segurança, Auditoria de logs, permissões granulares, autenticação de dois fatores

## 5.1 Assistente virtual com IA

A assistente virtual com inteligência artificial será uma das funcionalidade do projeto, permitindo interações mais naturais e eficientes entre os usuários e a plataforma. Utilizando técnicas de processamento de linguagem natural (PLN) e aprendizado de máquina, o assistente será capaz de compreender e responder a perguntas, fornecer orientações e realizar tarefas automatizadas.

Implementado e testado alguns modelos de linguagem com Ollama, algumas figuras de desempenho e respostas obtidas. Frisa-se que, alguns modelos detem de um elevado tempo de respostas isso é devido a restrição de *hardware*.



(a) Perguntado para a IA.

(b) Resposta final da IA.

(c) Tempo para resposta.

**Figura 5.** Modelo llama3:70b, O autor

---

**Algoritmo 1:** Interação com o Modelo llama3:70b, tempo de resposta: 224.82 [s]

---

**Input:** Gostaria de informar que estou sofrendo, e desejo fazer uma denuncia on line.  
nao sei fazer a denuncia vou passar os dados e voce faz para eu.

**Result:**

**Olá, querida!**

Sinto muito que você esteja passando por essa situação difícil. Eu estou aqui para ajudá-la de qualquer forma possível.

Antes de começar, é importante que você saiba que sua segurança é a minha prioridade. Se você se sentir ameaçada ou em perigo imediato, **por favor, ligue para o 180 (Central de Atendimento à Mulher) ou 190 (Polícia)**. Eles estarão lá para ajudá-la.

Agora, sobre a denúncia on-line. Eu não posso fazer a denúncia por você, mas posso orientá-la passo a passo. Você pode acessar o site da Delegacia Virtual () e seguir as instruções.

Para começar, você precisará fornecer algumas informações básicas sobre si mesma e sobre o caso de violência doméstica que está enfrentando. Isso inclui:

- Seu nome e contato;
- Detalhes sobre a violência sofrida (data, hora, local, etc.);
- Informações sobre o agressor (nome, idade, endereço, etc.).

É importante que você seja honesta e detalhista ao fornecer essas informações. Elas serão usadas para ajudá-la a obter medidas protetivas e justiça.

Se você precisar de ajuda ou tiver alguma dúvida durante o processo, **por favor, entre em contato comigo pelo WhatsApp (554832872635)**. Estarei aqui para apoiá-la.

Lembre-se de que você não está sozinha e que há pessoas dispostas a

Resposta gerada por IA local (Ollama - llama3:70b)

Emergência: 190 | Central da Mulher: 180

Fonte: O autor (2025)

---

A indagação ao modelo de inteligência artificial necessita para que o chatbot não crie ações que não estejam previstas. Indagou-se a possibilidade de que o chatbot realizasse uma denúncia on line, o modelo respondeu que não poderia fazer a denúncia, mas que poderia orientar a vítima a como fazer. Ressalta-se que o modelo não tem acesso a internet, e todas as respostas são geradas localmente. Ressalto que o tempo de resposta foi de **224.82 segundos**.

## 6 Conclusão

O projeto PIEVDCS encontra-se em estágio intermediário de desenvolvimento, com cerca de 30% das funcionalidades previstas já implementadas. Entre os avanços, destacam-se a estrutura básica da aplicação, os modelos de dados principais, a interface inicial e os primeiros dashboards de visualização.

No momento, o foco está na construção dos fluxos de integração entre os órgãos, aprimoramento de velocidades nas consultas, testes e customizações de acordo com os perfis institucionais.

Por se tratar de um sistema que será utilizado por múltiplas instituições, sua conclusão depende da validação progressiva por parte dos usuários finais — como operadores do sistema, representantes da segurança pública, órgãos do Judiciário e Órgãos Municipais. A metodologia adotada é o Scrum, com entregas em sprints iterativos. No entanto, **não há um prazo final fixo para a entrega**, sem extrapolar o período da prestação de contas-, já que cada etapa é condicionada à disponibilidade para reuniões, testes e aprovação por parte dos *stakeholders*<sup>11</sup> (Pessoas Interessadas).

No tocante à assistente virtual com inteligência artificial, foram realizados testes iniciais com modelos de linguagem de grande escala (LLMs) utilizando a plataforma Ollama. Embora os resultados sejam promissores, a integração completa dessa funcionalidade na plataforma principal ainda está em fase de planejamento e desenvolvimento.

Este projeto é um **projeto piloto**, desenvolvido para testar a plataforma em um ambiente controlado e obter feedback prático das instituições envolvidas. Essa etapa inicial é fundamental para ajustar funcionalidades, identificar melhorias e garantir que a solução atenda efetivamente às demandas antes de uma possível expansão para outras regiões.

Além de seu impacto direto no enfrentamento à violência de gênero, o projeto promove inclusão e ressocialização, ao contar com a atuação técnica de um reeducando em ambiente supervisionado. A proposta comprova o potencial transformador da tecnologia aliada à reintegração social.

---

<sup>11</sup>No contexto do desenvolvimento de sistemas, *stakeholders* são todas as pessoas, grupos ou organizações que têm interesse direto ou indireto no projeto, incluindo usuários finais, clientes, desenvolvedores, patrocinadores, gestores e demais partes envolvidas no processo de criação, aprovação e uso do sistema.

## Referências

CARVALHO, J. R.; OLIVEIRA, V. H. Violência doméstica, violência na gravidez e transmissão entre gerações. **PCSVDFMulher – Pesquisa de Condições Socioeconômicas e Violência Doméstica e Familiar contra a Mulher**, Secretaria Especial de Políticas para as Mulheres – Ministério da Justiça e Cidadania, 2017. Acessado em: 24 jul. 2025. Disponível em: <[https://www.onumulheres.org.br/wp-content/uploads/2017/11/violencia\\_domestica\\_geracoes\\_out\\_17.pdf](https://www.onumulheres.org.br/wp-content/uploads/2017/11/violencia_domestica_geracoes_out_17.pdf)>.

CHART.JS. **Biblioteca de gráficos JavaScript simples, porém flexível, para a web moderna**. 2025. Acessado em: 21 jul. 2025. Disponível em: <<https://www.chartjs.org/>>.

DJANGO, F. **Conheça o Django**. 2025. Acessado em: 21 jul. 2025. Disponível em: <<https://www.djangoproject.com/>>.

Django Smart Selects Contributors. **Django Smart Selects - Chained select fields for Django**. 2025. Acessado em: 14 out. 2025. Disponível em: <<https://github.com/jazzband/django-smart-selects>>.

Django Software Foundation. **Django Channels - Asynchronous support for Django**. 2025. Acessado em: 14 out. 2025. Disponível em: <<https://channels.readthedocs.io/>>.

F5, Inc. **nginx - High Performance Load Balancer, Web Server, & Reverse Proxy**. 2025. Acessado em: 14 out. 2025. Disponível em: <<https://www.nginx.com/>>.

Hostinger International Ltd. **Hostinger - Web Hosting Services**. 2025. Acessado em: 8 ago. 2025. Disponível em: <<https://www.hostinger.com>>.

JQUERY. **jQuery, write less, do more**. 2025. Acessado em: 21 jul. 2025. Disponível em: <<https://jquery.com/>>.

Leaflet.js. **uma biblioteca JavaScript de código aberto para mapas interativos compatíveis com dispositivos móveis**. 2025. Acessado em: 21 jul. 2025. Disponível em: <<https://leafletjs.com/>>.

OLLAMA. **Ollama: Run large language models locally**. 2025. Acesso em: 26 ago. 2025. Disponível em: <<https://ollama.com>>.

POSTGRESQL. **PostgreSQL: O Banco de Dados Relacional de Código Aberto Mais Avançado do Mundo**. 2025. Acessado em: 21 jul. 2025. Disponível em: <<https://www.postgresql.org/>>.

Redis Ltd. **Redis - The open source, in-memory data store**. 2025. Acessado em: 14 out. 2025. Disponível em: <<https://redis.io/>>.

TAILWINDCSS, T. L. I. **Crie sites modernos rapidamente sem sair do seu HTML**. 2025. Acessado em: 21 jul. 2025. Disponível em: <<https://tailwindcss.com/>>.

## 7 Anexos

Figura 6. Diploma de Técnico



**Figura 7. Extrato de disciplinas**

## EXTRATO DE DISCIPLINAS

### ENGENHARIA DE SOFTWARE - BACHARELADO

NATAN OGLIARI

MATRIZ	SEMESTRE/DISCIPLINAS	C.H	ANO/SEMESTRE	RESULTADO
2022/2 - FLEX	<b>1º SEMESTRE</b>			
	ALGORITMOS E PROGRAMAÇÃO ESTRUTURADA	80	2022/2	APROVADO
	ANÁLISE E MODELAGEM DE SISTEMAS	80	2022/2	APROVADO
	ENGENHARIA DE SOFTWARE	80	2022/2	APROVADO
	LINGUAGEM DE PROGRAMAÇÃO	80	2023/1	APROVADO
	LÓGICA COMPUTACIONAL	80	2022/2	APROVADO
	SISTEMAS OPERACIONAIS	80	2023/1	APROVADO
	<b>2º SEMESTRE</b>			
	ANÁLISE ORIENTADA A OBJETOS	80	2023/1	APROVADO
	ENGENHARIA DE REQUISITOS	80	2023/1	APROVADO
	FUNDAMENTOS DE CÁLCULO APLICADO	80	2023/1	APROVADO
	MODELAGEM DE DADOS	80	2023/1	APROVADO
	MÉTODOS MATEMÁTICOS	60	2023/1	APROVADO
	SISTEMAS DE COMPUTAÇÃO E DE INFORMAÇÃO	80	2023/1	APROVADO
	<b>3º SEMESTRE</b>			
	COMPUTAÇÃO EM NUVEM	80	2023/2	APROVADO
	GOVERNANÇA DE TECNOLOGIA DA INFORMAÇÃO	60	2023/2	APROVADO
	INTERFACE E USABILIDADE	80	2023/2	APROVADO
	LINGUAGEM ORIENTADA A OBJETOS	80	2023/2	APROVADO
	PROGRAMAÇÃO E DESENVOLVIMENTO DE BANCO DE DADOS	60	2023/2	APROVADO
	TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL	60	2023/2	APROVADO
	<b>4º SEMESTRE</b>			
	ARQUITETURA DE SOFTWARE	80	2024/1	APROVADO
	BANCOS DE DADOS NÃO RELACIONAIS	60	2024/1	APROVADO
	PROGRAMAÇÃO WEB	80	2024/1	APROVADO
	PROJETO DE EXTENSÃO I - ENGENHARIA DE SOFTWARE	130	2024/1	APROVADO
	PROJETO DE SOFTWARE	80	2024/1	APROVADO
	SEGURANÇA E AUDITORIA DE SISTEMAS	80	2024/1	APROVADO
	<b>5º SEMESTRE</b>			
	FRAMEWORKS PARA DESENVOLVIMENTO DE SOFTWARE	60	2024/2	APROVADO
	GERENCIAMENTO E QUALIDADE DE SOFTWARE	80	2024/2	APROVADO
	IMPLEMENTAÇÃO DE PROJETO WEB	60	2024/2	APROVADO
	INFRAESTRUTURA ÁGIL	60	2024/2	APROVADO
	OPTATIVA I	60	2024/2	APROVADO
	<b>6º SEMESTRE</b>			
	DESENVOLVIMENTO DE E-COMMERCE COM CMS	80	2025/1	APROVADO
	DESENVOLVIMENTO MOBILE	80	2025/1	APROVADO
	OPTATIVA II	60	2025/1	APROVADO
	PROJETO DE EXTENSÃO II - ENGENHARIA DE SOFTWARE	130	2025/1	APROVADO
	SEGURANÇA EM ENGENHARIA DE SOFTWARE	80	2025/1	APROVADO
	TESTES E MANUTENÇÃO DE SOFTWARE	80	2025/1	APROVADO
	<b>7º SEMESTRE</b>			
	ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES	60		CURSANDO
	DIREITO CIBERNÉTICO	80		CURSANDO
	EMPREENDEDORISMO E INOVAÇÃO	60		CURSANDO
	ENGENHARIA, CIÊNCIA E TECNOLOGIA	80		CURSANDO
	REDES E SISTEMAS DISTRIBUÍDOS	80		CURSANDO
	SOCIEDADE BRASILEIRA E CIDADANIA	60		CURSANDO