**CENTRO DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA**

**ESCOLA ESTADUAL TÉCNICA SÃO MATEUS**

**INFORMÁTICA PARA INTERNET**

**ALICE PINHEIRO TRINDADE**

**ANDREW ISMAEL RODRIGUES SILVA**

**BRENO AUGUSTO BATISTA GIANOTTI**

**GABRIEL ALVES BIZZI**

**GUILHERME DE OLIVEIRA RODRIGUES**

**ISABELA MARQUES DA SILVA SANTOS**

**ANNE**

**SÃO PAULO – SP**

**2020**

**ALICE PINHEIRO TRINDADE**

**ANDREW ISMAEL RODRIGUES SILVA**

**BRENO AUGUSTO BATISTA GIANOTTI**

**GABRIEL ALVES BIZZI**

**GUILHERME DE OLIVEIRA RODRIGUES**

**ISABELA MARQUES DA SILVA SANTOS**

**ANNE**

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Informática para Internet Integrado ao Ensino Médio da Escola Técnica São Mateus, orientado pelo Prof. Elvis Martins, como requisito parcial para obtenção do título de técnico em Informática para Internet Integrado ao Ensino Médio.

**São Paulo**

**2020**

**ANNE**

Inteligência Artificial

Relatório final, apresentado a ETEC São Mateus, como parte das exigências para a obtenção do título de Técnico em Informática para a Internet.

São Paulo, 7 de dezembro de 2020

BANCA EXAMINADORA

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. (Nome do orientador)

Afiliações

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. (Nome do professor avaliador)

Afiliações

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. (Nome do professor avaliador)

Afiliações

Dedicamos este projeto a todos professores e docentes que nos apoiaram e guiaram pelo caminho, mesmo nos dias difíceis.

Agradecimentos

Deixamos aqui nossos agradecimentos a todos professores e docentes que em tantas vezes se esforçaram para nos auxiliar em diversas questões, nos guiaram durante estes três anos e nos ensinaram muito além de bases disciplinares. Agradecemos também a nossas famílias e amigos que nos apoiaram e motivaram a seguir em frente. Que no futuro podemos contar com vocês, assim como podem contar conosco.

"...as vezes são as pessoas que ninguém espera nada, que fazem as coisas que ninguém consegue imaginar." - Alan Turing

RESUMO

Este trabalho tem como finalidade desenvolver uma inteligência artificial na qual irá escanear aplicações web, como websites, banco de dados etc., a fim de encontrar vulnerabilidades. Utilizará de aprendizado de máquina ou Machine Learning para se adequar ao cenário e conseguir, além de uma melhor performance e menos gasto de tempo, resultados muito mais precisos. Por fim, o sistema irá liberar um arquivo em formato PDF, no qual estará o resultado do escaneamento de uma forma simples e compreensível.

O presente projeto se trata da análise de vulnerabilidade através de uma inteligência artificial, uma ideia inovadora. O nosso diferencial no mercado seria a velocidade e um resultado que permite uma fácil compreensão para uma pessoa sem conhecimento prévio no assunto, mas que possibilite fácil compreensão.

ABSTRACT

This Project has as the main goal developing an artificial intelligence, which is going to be able to scan web application, such as web sites, databases etc., in order to find vulnerabilities. It will use Machine Learning to adequate itself to the scenario and get, not only a good and quick performance, but also precise results. The system will create and release a PDF file, in which contains information about the scan in a simple and understandable way.

We decided to develop our project in the analysis of vulnerabilities through an artificial intelligence with an innovative idea. The difference between our system and similar ones in the market would be the quickness and an easy result to understand, so that anyone without brief knowledge of vulnerability analysis can easily use the platform.

LISTA DE ILUSTRAÇÕES

[Figura 1 - Cronograma do projeto 16](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988552)

[Figura 2 - Lista de Top 10 principais vulnerabilidades web 20](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988553)

[Figura 3 - Países que mais sofrem cyber ataques 23](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988554)

[Figura 4 - Diagrama de Caso de Uso 25](#_Toc57988555)

[Figura 5 - Diagrama de Classe(Perfil) 28](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988556)

[Figura 6 - Diagrama de Classe (Pagamento) 29](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988557)

[Figura 7 - Diagrama de Classe (Relatório) 29](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988558)

[Figura 8 - Diagrama de sequência (Cadastro e login) 30](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988559)

[Figura 9 - Diagrama de Sequência (Pagamento) 31](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988560)

[Figura 10 - Diagrama de Sequência (Relatório) 32](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988561)

[Figura 11 - Wireframe (Home) 35](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988562)

[Figura 12 - Wireframe com layout grid (Home) 36](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988563)

[Figura 13 - Wireframe com layout grid e preenchimento de colunas (Home) 37](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988564)

[Figura 14 - Wireframe da página (Sobre nós) 38](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988565)

[Figura 15 - Wireframe com layout grid (Sobre Nós) 39](#_Toc57988566)

[Figura 16 - Wireframe com layout grid e preenchimento de colunas (Sobre Nós) 40](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988567)

[Figura 17 - Wireframe (Prices) 41](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988568)

[Figura 18 - Wireframe com layout grid (Prices) 42](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988569)

[Figura 19 - Wireframe com layout grid e preenchimento de colunas (Prices) 43](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988570)

[Figura 20 - Wireframe (Contatos) 44](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988571)

[Figura 21 - Wireframe com layout grid (Contatos) 45](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988572)

[Figura 22 - Wireframe com layout grid e preenchimento de colunas (Contatos) 46](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988573)

[Figura 23 - Wireframe com layout grid (Cadastro) 47](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988574)

[Figura 24 - Wireframe (Cadastro) 47](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988575)

[Figura 25 - Wireframe (Login) 48](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988576)

[Figura 26 - Wireframe com layout grid e preenchimento de colunas (Cadastro) 48](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988577)

[Figura 27 - Wireframe com layout grid e preenchimento de colunas (Login) 49](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988578)

[Figura 28 - Wireframe com layout grid (Login) 49](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988579)

[Figura 29 - Tela após análise 50](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988580)

[Figura 30 – Modelo ER 50](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988581)

[Figura 31 - Diagrama ER 51](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988582)

[Figura 32 - Dicionário de dados 51](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988583)

[Figura 33 - Diagrama de Pacote (Web Site) 52](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988584)

[Figura 34 - Diagrama de Pacotes (Relatório) 53](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988585)

[Figura 35 - Diagrama de Pacotes(Relatório) 53](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988586)

[Figura 36 - Diagrama de classes OO 54](#_Toc57988587)

[Figura 37 - Diagrama de sequência OO 55](file:///C:\Users\Alice%20P.%20Trindade\Desktop\Anne%20PREVIA.docx#_Toc57988588)

#### Sumário

[1 INTRODUÇÃO 12](#_Toc57988589)

[1.1 PROBLEMATIZAÇÃO 12](#_Toc57988590)

[1.2 HIPÓTESES 12](#_Toc57988591)

[1.3 OBJETIVOS 13](#_Toc57988592)

[1.3.1 OBJETIVOS GERAIS 13](#_Toc57988593)

[1.3.2 OBJETIVOS ESPECÍFICOS 13](#_Toc57988594)

[1.4 JUSTIFICATIVA 13](#_Toc57988595)

[1.5 METODOLOGIA 14](#_Toc57988596)

[1.6 CRONOGRAMA 16](#_Toc57988597)

[1.7 TEMA E TEMÁTICA 17](#_Toc57988598)

[1.6.1 TEMÁTICA 17](#_Toc57988599)

[1.6.2 TEMA 17](#_Toc57988600)

[1.8 CENÁRIO 17](#_Toc57988601)

[1.9 AGENTES 17](#_Toc57988602)

[2 REFERENCIAL TÉORICO 18](#_Toc57988603)

[2.1 TECNOLOGIA IA 18](#_Toc57988604)

[2.2 VULNERABILIDADE WEB 19](#_Toc57988605)

[2.2.1 PRINCIPAIS VULNERABILIDADES WEB 20](#_Toc57988606)

[2.2.2 PAÍSES QUE MAIS SOFREM CIBERATAQUES 23](#_Toc57988607)

[2.3 DESIGN 23](#_Toc57988608)

[3 LEVANTAMENTO DE REQUISITOS 25](#_Toc57988609)

[3.1 FUNCIONAIS 25](#_Toc57988610)

[3.1.1 FUNCIONALIDADES QUE O SISTEMA 25](#_Toc57988611)

[3.1.2 DESCRIÇÃO DO CENÁRIO 26](#_Toc57988612)

[3.2 NÃO FUNCIONAIS 26](#_Toc57988613)

[3.2.1 SEGURANÇA, PERFORMANCE, CONFIABILIDADE, USABILIDADE, ETC. 26](#_Toc57988614)

[4 ANÁLISE 27](#_Toc57988615)

[4.1 MODELO DE DOMÍNIO 27](#_Toc57988616)

[4.1.1 CLASSES CONCEITUAIS 27](#_Toc57988617)

[4.1.2 DIAGRAMA DE CLASSES 28](#_Toc57988618)

[4.2 DIAGRAMAS DE SEQUÊNCIA DE SISTEMA 30](#_Toc57988619)

[4.2.1 DIAGRAMAS DE SEQUÊNCIA DE SISTEMA 30](#_Toc57988620)

[4.3 OUTROS ARTEFATOS 32](#_Toc57988621)

[4.3.1 DOCUMENTAÇÃO DA INTERFACE GRÁFICA 32](#_Toc57988622)

[4.3.2 MODELO RELACIONAL 50](#_Toc57988623)

[5 ARQUITETURA LÓGICA 52](#_Toc57988624)

[5.1 DIAGRAMA DE PACOTES 52](#_Toc57988625)

[6 DESIGN ORIENTADO A OBJETOS 54](#_Toc57988626)

[6.1 DIAGRAMA DE CLASSES 54](#_Toc57988627)

[6.2 DIAGRAMA DE SEQUÊNCIA 55](#_Toc57988628)

[7 OUTROS TÓPICOS DE ANÁLISE E DESIGN 56](#_Toc57988629)

[7.1 DDD: DOMAIN DRIVEN DESIGN 56](#_Toc57988630)

[7.2 TDD: TEST DRIVEN DEVELOPMENT 56](#_Toc57988631)

[7.3 REFATORAÇÃO 58](#_Toc57988632)

[7.4 DESIGN PATTERNS 59](#_Toc57988633)

[7.4.1 APLICAÇÃO 60](#_Toc57988634)

[8 CONSIDERAÇÕES FINAIS 61](#_Toc57988635)

[REFERÊNCIAS 62](#_Toc57988636)

[APÊNDICES 64](#_Toc57988637)

# 1 INTRODUÇÃO

## 1.1 PROBLEMATIZAÇÃO

Segundo uma pesquisa da TIC Domicílios do NIC.br, com o crescente número de acessos à Internet, há muitas pessoas novas sem grande conhecimento na área sendo introduzidas a tudo que envolve a internet, algumas não tem um grande conhecimento sobre a tecnologia e o que pode ocorrer por trás da interface, podendo cair em plataformas vulneráveis, assim possivelmente expondo seus dados e se prejudicando. Não se limitando a apenas usuários individuais, pode ser utilizado também por empresas, que tem sites com possíveis vulnerabilidades, para gerar o relatório e ficar a par das falhas e arrumá-las com uma maior precisão.

Como a Segurança da informação anda crescendo continuamente segundo a IDC, International Data Corporation, a área de Segurança da Informação gerou cerca de 107 bilhões de dólares para o mercado em 2019. O Brasil se encontra em 10º lugar de acordo com o nível de gastos, com US$ 1,6 bilhão ao todo. O aumento do consumo em software dentro das empresas evidencia a importância na aquisição das novas tecnologias. Tendo isso em vista focamos em acompanhar o mercado juntamente de suas evoluções a fim de desenvolver uma plataforma de simples entendimentos para pessoas leigas no assunto.

A IDC ressalta, ainda, que as organizações devem realizar um reposicionamento a respeito do uso de tecnologia. Técnicas isoladas para a proteção dos sistemas apresentarão uma efetividade cada vez menor na hora de oferecer um ambiente saudável, o que vai exigir investimentos sobre diferentes tipos de processos e soluções. Logo, a capacidade de gerenciar diferentes serviços de modo exponencial não será um requisito apenas no processamento e armazenamento dos dados, mas também na segurança deles.

## 1.2 HIPÓTESES

Temos como objetivo final criar uma inteligência artificial voltada para a análise de aplicações web, onde o usuário irá colocar o endereço desejado e a análise será feita.

Assim que terminar, nossa IA apresentará um relatório completo sobre a varredura solicitada e possíveis erros e vulnerabilidades encontrados, a fim de que o representante ou proprietário possa decidir qual a melhor alternativa para a correção e dependendo do problema nossa plataforma também será capaz de solucioná-la de forma simples e eficaz.

Embora existam outras empresas e projetos que possuem um foco semelhante ao nosso, nos diferenciamos em termos de desenvolvimento, uma vasta gama de dados analíticos para uma melhor composição, e proporcionamos um relatório ao usuário dos erros e falhas encontrados, sem deixar de pensar na facilidade de uso para qualquer pessoa e em uma performance muito boa, já que a IA se adequará ao ambiente para explorá-lo e conseguir coletar informações rapidamente.

Procuramos entregar até o fim do período de criação do projeto uma versão beta do que planejamos.

## 1.3 OBJETIVOS

### 1.3.1 OBJETIVOS GERAIS

Nosso projeto tem como objetivo utilizar o sistema para analisar dados das plataformas do usuário e apresentar os erros e vulnerabilidades presentes, sendo uma ferramenta de fácil uso, sem a necessidade de passar horas lendo códigos.

### 1.3.2 OBJETIVOS ESPECÍFICOS

* Coletar dados sobre o conhecimento do usuário e analisar tais informações para obter um maior entendimento do usuário e desempenho no sistema.
* Criar um sistema que analisa plataformas Web
* Procurar falhas e erros
* Criar relatórios sobre estes.
* Apresentar o resultado da análise.

## 1.4 JUSTIFICATIVA

Nos tempos atuais, costuma-se inserir em sites e plataformas informações como, e-mail, senha do e-mail, cartão de crédito ou débito, tanto para uma doação ou pagamento de serviços. A segurança do site se torna algo contestável devido ao fato de serem informações pessoais importantes. Anne se torna a resolução deste problema, sendo uma IA que testara a vulnerabilidade do site, apontando as falhas na segurança do site.

Acreditamos que este teste de vulnerabilidade nos sites seja primordial para atrair mais clientes e, assim, a confiança deles. Nosso projeto pretende fazer uma leitura intensa do banco de dados em MySQL, analisando os códigos, identificando erros, como de semântica ou no próprio HTML e CSS, e os consertando.

Para, deste modo, garantir o conforto e tranquilidade do cliente enquanto este navega. Anne envia um relatório para o desenvolvedor do Software Web analisado, informando o nível da segurança do site e mostrando os erros para que possam ser consertados o mais rápido possível, e com a autorização do desenvolvedor Anne os arruma automaticamente.

## 1.5 METODOLOGIA

Quando comentamos ou falamos sobre Inteligência Artificial logo pensamos em sistemas e humanos-robôs substituindo os humanos em determinada coisa. Muitos não percebem, mas a Inteligência Artificial está no nosso dia a dia. Sabe o corretor ortográfico do smartphone que vai completando as palavras e frases que você escreve com mais frequência? Isso é uma Inteligência Artificial que aprendeu e entendeu seus hábitos, onde soube quais são as palavras mais utilizadas por você, usuário, fazendo sugestões.

Uma Inteligência Artificial pode ser definida como a capacidade de sistemas e dispositivos de identificar e solucionar problemas. E muitos já atuam de maneira autônoma, ou seja, não precisam da supervisão do ser humano. As Inteligências Artificiais conseguem processar uma grande quantidade de dados, que uma pessoa não tem condições de fazer, desta forma, estão assumindo atividades repetitivas e que não exigem empatia ou criatividade, qualidades humanas que uma IA não consegue suprir.

A equação básica da tecnologia da Inteligência Artificial atualmente é: Big Data + Modelos de Dados Eficientes + Computação em Nuvem = Máquinas Inteligentes ou Inteligência Artificial.

Estes modelos de dados envolvem algumas áreas da Ciência da Computação, em especial Processamento de Linguagem Natural, Deep Learning e Machine Learning.

A lógica de fazer uma IA para websites, é de melhorar a viabilidade do usuário e do desenvolvedor, onde podemos encontrar problemas na navegação do usuário, como, lentidão; perda de informações cadastradas, entre outras. Ou seja, esta IA irá analisar erros contidos em Softwares Web através de uma intensa análise no seu código fonte e conexões PDO por MySQL ou NoSQL, como falhas de injeção, que são comuns em website, apontando os canais de vulnerabilidade que tem no site. Especificando e corrigindo as brechas de segurança.

Esse tipo de injeção ocorre quando:

* Os dados fornecidos pelo usuário são enviados a um interpretador contendo comandos mal-intencionados.
* Credenciais de acesso e token de sessão não protegidos apropriadamente. Atacantes comprometem senhas, chaves ou tokens de autenticação de forma a assumir a identidade de outros usuários.
* Restrições de perfil de acesso não devidamente aplicadas.
* Configuração incorreta de Segurança.
* APIs não protegem adequadamente dados confidenciais.

É uma ferramenta de teste de intrusão, open source, que automatizará o processo de detecção e exploração de falhas de injeção SQL e Erro de semântica no código fonte.

Como base nas informações, foram pesquisados assuntos como:

* Python e Segurança da Informação;
* JavaScript e TypeScript;
* Análise de Sistemas;
* Kali Linux;
* Programação Python in Terminal;
* Indentação.

Este projeto foi desenvolvido em um tempo máximo de 7 meses, em que na equipe de desenvolvimento havia: 1 programador; 2 designers; 2 copyright e 1 administrador. Foi desenvolvida na linguagem Python, versão 3.8.0; Utilizando aplicações de Machine Learning, um método de análise de dados que automatiza a construção de modelos analíticos, sendo um subcampo da Engenharia e da ciência da computação que evoluiu do estudo de reconhecimento de padrões e da teoria do aprendizado computacional em inteligência artificial; Biblioteca PyBrain, para a criação de redes neurais, automatizadas para não só a otimização da aplicação a ser desenvolvida, mas também para Anne mostrar o que está acontecendo com o código, detalhando seu erro com outras bibliotecas.

Anne apontará erros de maneira geral em um relatório que é gerado dentro de seu sistema, auxiliando o desenvolvedor (com baixo ou alto conhecimento) a melhorar e otimizar seus softwares web.

## 1.6 CRONOGRAMA

Figura 1 - Cronograma do projeto

## 1.7 TEMA E TEMÁTICA

### 1.6.1 TEMÁTICA

Temos como temática a Inteligência Artificial.

### 1.6.2 TEMA

Uso de Inteligência Artificial para a análise de vulnerabilidades e erros comuns em sistemas e softwares Web.

## 1.8 CENÁRIO

A princípio nosso projeto é um aplicativo com inteligência artificial que procura as vulnerabilidades em aplicações web e disponibiliza um relatório com as vulnerabilidades encontradas. Como o tema pode ser um pouco complexo e abordar diversas áreas nossos principais cenários estão localizados em Internet, Inteligência artificial, hacking e aplicativos. Mais especificamente nossos cenários são: Inteligência artificial voltada para pentesting, vulnerabilidades de aplicações web e aplicativos para teste de vulnerabilidades.

## 1.9 AGENTES

Os agentes do projeto compõem-se das pessoas envolvidas no desenvolvimento e as que estão tendo o recebimento do projeto, ou seja, os clientes.

Para o desenvolvimento, temos seis membros: Alice, responsável pela parte escrita e documentação de todos os passos do projeto; Andrew, responsável por dialogar e ter acesso ao público, praticar entrevistas e consultas; Breno, líder do grupo, responsável pela administração do desenvolvimento; Guilherme, responsável pelo design e desenho das partes visuais do projeto; Gabriel Bizzi, responsável pela programação em geral; e Isabela, responsável também pela documentação e design de partes visuais.

Para o recebimento do projeto, temos uma empresa ou pessoa física que deseja analisar a segurança de seu site de forma simples, eficaz e rápida, sem a contratação de técnicos para realizar tal feito, sendo utilizado de forma interativa e acessível.

# 2 REFERENCIAL TÉORICO

## 2.1 TECNOLOGIA IA

Com o decorrer do tempo o uso da Internet foi crescendo, e suas utilidades aumentando. Com estas novas utilidades surgiram diversas plataformas, como para compras online por exemplo, o que requer informações privadas do usuário, como números de documentos e senhas de cartões. Muitas plataformas podem possuir vulnerabilidades, causando problemas, aos desenvolvedores e em alguns casos, clientes também. Por conta destes fatores que pensamos em nosso projeto, que busca, por meio de uma IA encontrar erros e falhas em plataformas Web.

Neste referencial teórico apresentaremos os conceitos básicos que nos trouxeram a este projeto, como os métodos utilizados, a vulnerabilidade de sistemas Web, o design e até o próprio IA.

A partir das pesquisas que foram realizadas, a IA pode ser uma opção, se não a melhor para resolver problemas que exigiriam muito da mão humana. Esperava-se que, em muitas empresas, fossem usadas IA's para cibersegurança. De fato, isso existe, mas muitas empresas ainda necessitam de segurança cibernética e uma IA é o que mais poderia se encaixar nessa atividade de segurança, já que há poucas pessoas no mundo que tem as habilidades de um profissional da segurança da Informação, onde este fica horas trabalhando para a eliminação de malwares e também o x de funcionários que fazem isso na mesma empresa. "Para ilustrar isso, suponha que o professor diga à máquina: "Faça sua lição de casa agora". Isso pode fazer com que "o professor diga 'faça sua lição de casa agora'" seja incluído entre os fatos bem estabelecidos. Outro fato pode ser: "Tudo o que o professor diz é verdade"." (COMPUTING MACHINERY AND INTELLIGENCE, Alan Turing)

A indentação apesar de ser importante para manutenção do código, é um recurso muito mais estético do que funcional, no que diz respeito a legibilidade do código, tendo em vista de que é muito importante, pois consiste na organização ou formatação do seu código para que ele fique legível e de fácil entendimento para outros programadores que forem trabalhar com ele. Veja bem que na maioria das linguagens não é obrigatório, mas a linguagem python liga para a identação. É ela que define os blocos, não existem begins-ends e chaves. Deste modo, a indentação do código em HTML, apesar de ser importante para manutenção do código, é um recurso muito mais estético do que funcional.

Já a hierarquia e a semântica é o extremo oposto disso, não vai alterar em nada a estética da aplicação, mas pode ser fundamental para um código funcional e semântico. Existe muitas formas de definir tecnicamente para que realmente serve a hierarquia de um documento ou como defini-la. "O termo semântico-html é um bocado e desmente o quão simples é, o quão bem estabelecido é entre os web designers modernos e o fato de ter benefícios muito além do óbvio, fazer a coisa certa para a Web usando a marcação semântica." (POSH, 2007)

O Certificado SLL (Secure Sockts Layer) é um protocolo criado para gerar mais segurança entre os servidores web e os computadores. Ele usufrui de criptografia para proteger a conexão entre o navegador e o servidor onde está sendo acessado, ele aumenta a proteção dessa conexão entre ambos ao criar uma espécie de “chave” que somente ambos os lados possuem, sendo uma barreira fortíssima na proteção de dados.

"SSL é um protocolo em camadas. Em cada camada, as mensagens podem incluir campos de comprimento, descrição e conteúdo. SSL leva mensagens para ser transmitido, fragmenta os dados em blocos gerenciáveis, opcionalmente comprime os dados, aplica um MAC, criptografa e transmite o resultado. Os dados recebidos são descriptografados, verificados, descompactados, remontados e entregue a clientes de nível superior." (A. Freier; P. Karlton; P. Kocher, 2011)

## 2.2 VULNERABILIDADE WEB

Vulnerabilidades web são falhas encontradas em sistemas e plataformas que se exploradas por pessoas, mal-intencionadas ou não, podem causar falhas de segurança a tal sistema. É o resultado de alguns fatores, sendo esses, o processo de criação, manutenção e uso da plataforma (Monteverde,2014).

### 2.**2.1 PRINCIPAIS VULNERABILIDADES WEB**

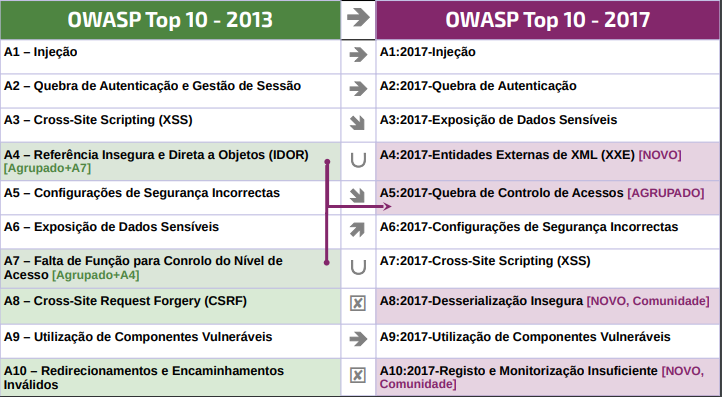
A OWASP (Open Web Application Security Project), uma organização mundial sem fins lucrativos, que procura divulgar aspectos relacionados a segurança de aplicativos Web, possui arquivos e matérias para pessoas interessadas, sendo um deles o Top 10, uma lista que mostra as 10 vulnerabilidades mais presentes em aplicações Web. Aqui citaremos e explicaremos as que envolvem e são mais relevantes em relação a nosso projeto com base nas informações descritas no site da OWASP e em outros artigos.

Figura 2 - Lista de Top 10 principais vulnerabilidades web

#### 2.2.1.1 INJEÇÃO

A injeção foi a vulnerabilidade mais explorada em 2003 e continua sendo em 2017, e a mais comum é a injeção SQL, que ocorre quando códigos maliciosos são injetados pelo formulário que entra em contato com o banco de dados, que executa o código. “A injeção de falhas consiste na introdução de dados não confiáveis ou maliciosos que são enviados a uma aplicação como parte de um comando ou consulta.” (Carvalho, Pereira, Alves, Ribeiro, 2013).

#### 2.2.1.2 CROSS SITE SCRIPTING

O Cross Site Scripting acontece quando o atacante utiliza uma aplicação Web para mandar um código malicioso ao navegador do usuário, que o executa e dá acesso as informações do navegador do usuário. “Atacantes podem utilizar essas falhas para roubar sessões de usuário da vítima, desfigurar sites, inserir conteúdos maliciosos, redirecionar usuários, sequestrar o navegador da vítima usando o malwares etc.” (Carvalho, Pereira, Alves, Ribeiro, 2013).

#### 2.2.1.3 REFERÊNCIA INSEGURA E DIRETA A OBJETO

Ocorre quando uma referência é exposta a um objeto interno da aplicação, como um arquivo de configuração. “Sem uma política correta de verificação de acesso, atacantes podem manipular referências internas da aplicação e acessar dados sensíveis sem autorização.” (OWASP, 2013).

#### 2.2.1.4 CONFIGURAÇÕES DE SEGURANÇA INCORRETAS

A pessoa que ataca por meio disto acessa contas padrões e páginas não utilizadas etc., para obter acesso não autorizado e conhecimento do sistema. Este tópico envolve a falta de configurações seguras em aplicações Web, servidores Web, servidores de aplicação e de banco de dados. “Todas essas configurações devem ser definidas, implementadas e mantidas, já que geralmente a configuração padrão é insegura.” (OWASP, 2013).

#### 2.2.1.5 EXPOSIÇÃO DE DADOS SENSÍVEIS

Dados sensíveis são dados confidencias como senhas, dados pessoais, senhas de cartões etc. Os atacantes não quebram a criptografia, eles procuram outros meios roubar chaves, ou dados apagados do servidor. “Os dados sensíveis merecem proteção extra como criptografia no armazenamento ou em trânsito, bem como precauções especiais quando trafegadas pelo navegador.” (OWASP, 2013).

#### 2.2.1.6 FALTA DE FUNÇÃO PARA CONTROLE DO NÍVEL DE ACESSO

A exploração desta vulnerabilidade acontece quando o atacante, que possui autorização no sistema, muda a URL ou o parâmetro de alguma função, com o objetivo de obter acesso. “Aplicações Web nem sempre protegem adequadamente as funções do aplicativo. Muitas vezes o nível de proteção da função é gerenciado através de configuração, e o sistema pode estar configurado incorretamente. (Carvalho, Pereira, Alves, Ribeiro, 2013).

#### 2.2.1.7 REDIRECIONAMENTOS E ENCAMINHAMENTOS INVÁLIDOS

A falha de redirecionamento ocorre quando utiliza um parâmetro e o usuário é redirecionado para o valor do parâmetro sem validação. “Sem uma validação adequada, os atacantes podem redirecionar as vítimas para sites de phishing ou malware, ou usar encaminhamentos para acessar páginas não autorizadas.” (OWASP, 2013).

#### 2.2.1.8 QUEBRA DE AUTENTICAÇÃO

Muitas vezes as funções de autenticação do usuário e ao gerenciamento de sessões são implementadas incorretamente, o que pode permitir que ataquem senhas, chaves e tokens de sessão e explorar outras falhas para assumir a identidade de outros usuários. “Uma vulnerabilidade de autenticação quebrada pode permitir que o invasor use métodos manuais e/ou automáticos para obter controle sobre qualquer conta que desejar em um sistema, ou obter controle completo sobre o sistema.” (SUCURI, 2020)

#### 2.2.1.9 REGISTRO E MONITORIZAÇÃO INSUFICIENTE

O registro e monitorização insuficiente permitem com que a pessoa que ataca, possa explorar o sistema persistentemente. “Não ter um processo eficiente de registro e monitoramento em vigor pode aumentar os danos ao comprometimento de um site” (SUCURI, 2020).

### 2.2.2 PAÍSES QUE MAIS SOFREM CIBERATAQUES

Uma pesquisa feita pela Statista de novembro de 2017 até abril de 2018 revelou os cinco países que mais sobrem em ataques. Como podemos ver a seguir o Estados Unidos lidera com 30,1% de todos os ataques feitos no mundo, logo após temos a Horlanda com 11,9%, China com 7,1%, Brasil com 6,2% e Rússia com 4,4%.

Figura 3 - Países que mais sofrem cyber ataques

Os Estados Unidos sofreram com mais de 238,6 milhões de ataques durantes o quarto trimestre de 2017 e com um custo total dos crimes cibernéticos de US$ 21,22 milhões, o mais caro do mundo.

## 2.3 DESIGN

A área tem como objetivo planejar o estilo da interface, mas não focando na cor ou no formato do botão como muitos pensam, o design irá primeiro pensar que tipo de experiência ou sentimento irá passar para o usuário, pois é a parte que todos terão o primeiro contato até mesmo na hora de instalá-lo.

Ou seja, o primeiro foco é pensar na impressão dos usuários terão sobre a ferramenta certificando que atinja ou até mesmo supere suas expectativas.

Uma parte importante que será utilizada é a UX/UI, UX se tratando do comportamento, e a UI de tratando da aparência. Colocando em questão primeiro da Ui temos em mente as cores, tipos e formatos dos botões e temos a tipografia sendo a escrita. Com isso temos a seguinte fala "...Eu considero que

Existem 3 pilares essenciais da Tipografia em interfaces, sendo eles: 1 Escala de tamanhos; 2 Use boas fontes; 3 Legibilidade..." Isso foi uma fala retirada do ebook escrito por Gabriel Silvestri, com isso eles quis mencionar que é muito importante ter uma variação boa no tamanho das fontes, assim como utilizar uma boa fonte ainda citando que é melhor utilizar uma fonte já pronta do que desenvolver uma, tendo uma maior legibilidade deixando mais aparente para o usuário. Em questão de cores e estilos de botões deve se levar em questão que tipo de experiência você quer passar ao usuário tendo uma paleta específica para cada situação, assim como as formas que se encaixam em certas paletas às deixando mais completas.

E agora levando em questão a UX ela é uma das partes mais importantes pois ela irá interagir com os usuários de várias maneiras, sempre pensando na melhor hipótese de satisfação para o usuário desde interação na instalação, interface enquanto se utiliza a plataforma, usada com peso em cima de apps nas suas interfaces. O design se interface define o mundo dos botões, Campos e outros componentes da interface que lhe fornece as habilidades de fazer as coisas com mais facilidade.

A interface de navegação é responsável pelos espaços de informação fornecida ao usuário possibilitando uma melhor deslocação unindo software e hipertexto, ajudando também na informação para uma comunicação afetiva com os usuários.

Em geral essas são uma das partes mais importantes do design dentro da interface, são os principais pontos a se seguir para atingir uma expectativa para ser passada ao usuário, pensando sempre em obter o melhor resultado na satisfação do próprio, além de também dar personalidade à interface fazendo com que ela seja de certa forma mais aceita e chamativa. Tratando-se disso o design sem dúvidas é uma das principais partes mais importantes durante o desenvolvimento de um produto.

# 3 LEVANTAMENTO DE REQUISITOS

## 3.1 FUNCIONAIS

A função do software a ser desenvolvido é auxiliar usuários e desenvolvedores de sites da internet a encontrar erros em plataformas web, assim obtendo uma maior segurança na navegação para os usuários.

### 3.1.1 FUNCIONALIDADES QUE O SISTEMA

Este sistema tem como principais funcionabilidades:

Permitir a escolha da plataforma pelo usuário;

Coleta de dados da plataforma;

Análise de códigos de plataformas;

Realizar relatórios sobre a plataforma.3.1.2 DIAGRAMA DE CASOS DE USO

Diagrama

Descrição gerada automaticamente

Figura 4 - Diagrama de Caso de Uso

### 3.1.2 DESCRIÇÃO DO CENÁRIO

**Escolha do site:** Um site de escolha do usuário tem o link anexado ao aplicativo e enviado ao sistema.

**Coleta de dados:** O sistema coleta as informações da plataforma.

**Análise:** O sistema realiza uma análise no código da plataforma.

**Relatório:** Sistema elabora um relatório em pdf para a visualização do cliente e faz o download automaticamente.

## 3.2 NÃO FUNCIONAIS

Requisitos não funcionais não possuem uma funcionabilidade, mas ainda assim são necessários no sistema para que o mesmo obtenha um melhor desempenho.

### 3.2.1 SEGURANÇA, PERFORMANCE, CONFIABILIDADE, USABILIDADE, ETC.

**Segurança:** Uma das partes mais importantes de um sistema é a segurança. Por isso é necessário que haja proteção e privação de dados e contas pessoais como documentos pessoais, e-mails, entre outros dados, para fazer com que os usuários e suas informações estejam o mais seguros o possível. Nossa plataforma procura ter controle do tratamento de dados, para que não sejam vazados e assim proporcionar uma experiencia segura aos usuários.

**Performance:** Procuramos ter uma plataforma pratica e dinâmica, de fácil entendimento com um bom desempenho na eficiência, que faça o usuário se sentir confortável a usando.

**Confiabilidade:** Temos conhecimento de que falhas podem ocorrer, porém procuramos estar à frente, verificando o sistema a procura de falhas para evitar danos ao sistema, também realizando manutenção.

**Usabilidade:** Temos um design agradável e posicionamento de elementos favorável à plataforma e ao usuário, para que seja de fácil uso.

# 4 ANÁLISE

## 4.1 MODELO DE DOMÍNIO

Anne é um aplicativo para **computadores** que tem como principal objetivo ajudar **usuários** comuns e **desenvolvedores** a saberem sobre a segurança de **plataformas** **web** visitadas ou criadas, no caso dos desenvolvedores. Assim ajudando pessoas a saberem se suas informações, ou a de seus clientes estão seguras.

Nosso projeto está dividido em duas partes importantes. A plataforma online e o aplicativo onde tudo acontece. Para obter **acesso**, o usuário deve acessar o web site. Lá há informações sobre o projeto, indo do objetivo até a **inscrição** e **download**. Para conseguir o aplicativo o usuário realiza um **cadastro**, um **login**, e assim realiza o **pagamento**, em duas formas: **boleto ou cartão**. O serviço emite o boleto ou gera a cobrança no cartão e quando/se o pagamento for autorizado a plataforma libera o download do **arquivo** **de** **instalação** do sistema.

Dentro do aplicativo o **usuário** poderá inserir um **link** de sua preferência para ser posto pelo sistema por uma análise, que retorna como resposta ao cliente um **relatório** sobre a plataforma em formato pdf, que é salvo no computador automaticamente.

### 4.1.1 CLASSES CONCEITUAIS

**Perfil:** Usuário, endereço, login, tipo.

**Pagamento:** Cartão, perfil, pedido.

**Relatório:** Site, resultado da Análise, autor.

|  |  |
| --- | --- |
| Perfil | Usuário |
| Endereço |
| Login |
| Tipo |
| Pagamento | Cartão |
| Perfil |
| Pedido |
| Relatório | Site |
| Resultado da análise |
| Autor |

### Diagrama Descrição gerada automaticamente4.1.2 DIAGRAMA DE CLASSES

Figura 5 - Diagrama de Classe(Perfil)

Diagrama

Descrição gerada automaticamente

Figura 6 - Diagrama de Classe (Pagamento)

Diagrama

Descrição gerada automaticamente

Figura 7 - Diagrama de Classe (Relatório)

## 4.2 DIAGRAMAS DE SEQUÊNCIA DE SISTEMA

### 4.2.1 DIAGRAMAS DE SEQUÊNCIA DE SISTEMA

Diagrama

Descrição gerada automaticamenteDiagrama

Descrição gerada automaticamente**Cadastro e login no Web Site**

Figura 8 - Diagrama de sequência (Cadastro e login)

Diagrama, Esquemático

Descrição gerada automaticamenteDiagrama

Descrição gerada automaticamente**Cobrança e pagamento**

Figura 9 - Diagrama de Sequência (Pagamento)

Diagrama

Descrição gerada automaticamente**Funcionamento do software**

Figura 10 - Diagrama de Sequência (Relatório)

# 4.3 OUTROS ARTEFATOS

### 4.3.1 DOCUMENTAÇÃO DA INTERFACE GRÁFICA

#### 4.3.1.1 AS 10 HEURÍSTICAS DE NIELSEN (SITE)

**01. Visibilidade do status do sistema**

Todo sistema depende integralmente da visão para situar o usuário. Dessa forma, nosso sistema conta com objetos estrategicamente posicionados, a fim de não prejudicar a visão do usuário e fazer com que ele se sinta confortável e em um local profissional – porém familiar –, como visto no wireframe.

**02. Correspondência entre o sistema e o mundo real**

Dentro da interface, utilizamos uma linguagem mais próxima do dia a dia do usuário e símbolos de fácil identificação, retornando ao primeiro tópico, frisando mais uma vez que os objetos são estrategicamente posicionados, visando a confortabilidade do usuário.

Um bom exemplo é a página de contato, onde temos um mapa interativo com o endereço e as redes sociais logo abaixo, sendo uma simples, porém forte mensagem para quem vê a página.

**03. Controle e liberdade para o usuário**

No nosso site, o usuário tem liberdade para utilizar todas as ferramentas disponíveis e voltar atrás. Pode ver as imagens pelo slide, mexer no mapa interativo e voltar ao ponto original, apagar as informações de contato etc.

**04. Consistência e padronização**

Todas as telas são padronizadas, a logo sempre na mesma posição juntamente com o rodapé e a barra de menu. Cores, fontes e tamanhos uniformes, como a página de “sobre” e a página de “download”.

**05. Prevenção de erros**

Não possuímos prevenção de erros em nosso site, pois ele é apenas explicativo e serve de promoção ao sistema.

**06. Reconhecimento em vez de memorização**

Para que o usuário reconheça elementos dentro do site, optamos por inserir os caminhos para outras páginas na parte superior de todas as telas, onde se encontram os botões “home”, “download”, “sobre nós” e “FAQ”.

**07. Flexibilidade e eficiência de uso**

Para atingir usuários leigos e experientes, deixamos os elementos de fácil acesso. Como, por exemplo, o botão de download se encontra logo na tela inicial.

**08. Estética e design minimalista**

O site é muito objetivo. Focamos em não encher a tela de elementos, mas sim, deixar com a menor quantidade possível. Em todas as páginas é perceptível o minimalismo.

**09. Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros**

Para isso, na tela de contato, por exemplo, teríamos uma mensagem de aviso no formulário, caso o usuário não preenchesse corretamente os campos obrigatórios.

**10. Ajuda e documentação**

Na aba de contato, um pouco abaixo do formulário, encontram-se as perguntas frequentemente feitas. Dessa forma, o usuário poderá “se encontrar”, como ter conhecimento de quais os requisitos necessários de uma máquina para poder rodar bem o sistema. Na página inicial também se encontra o botão de download, mais nítido do que os outros elementos presentes.

#### 4.3.1.2 AS 10 HEURÍSTICAS DE NIELSEN (APLICAÇÃO)

**01. Visibilidade do status do sistema**

Todas as telas da aplicação são limpas, a fim de evitar mau entendimento do usuário e prejudicar sua compreensão, incluindo um design minimalista e letras bem legíveis.

**02. Correspondência entre o sistema e o mundo real**

A linguagem utilizada é simples, de forma que guia o usuário até os botões e ações dentro da interface.

**03. Consistência e padronização**

Todas as telas da interface estão padronizadas, respeitando os limites, margens, fontes e cores.

**04. Flexibilidade e eficiência de uso**

Os botões e a aba para inserir a URL, por exemplo, estão de fácil acesso, com letras legíveis e linguagem acessível.

**05. Estética e design minimalista**

Por ser acessível para pessoas leigas, as telas não possuem muitas informações e são simples, com poucas frases e botões básicos.

#### 4.3.1.1 PROTÓTIPO (WIREFRAME)

Wireframe para o site:

Figura 11 - Wireframe (Home)

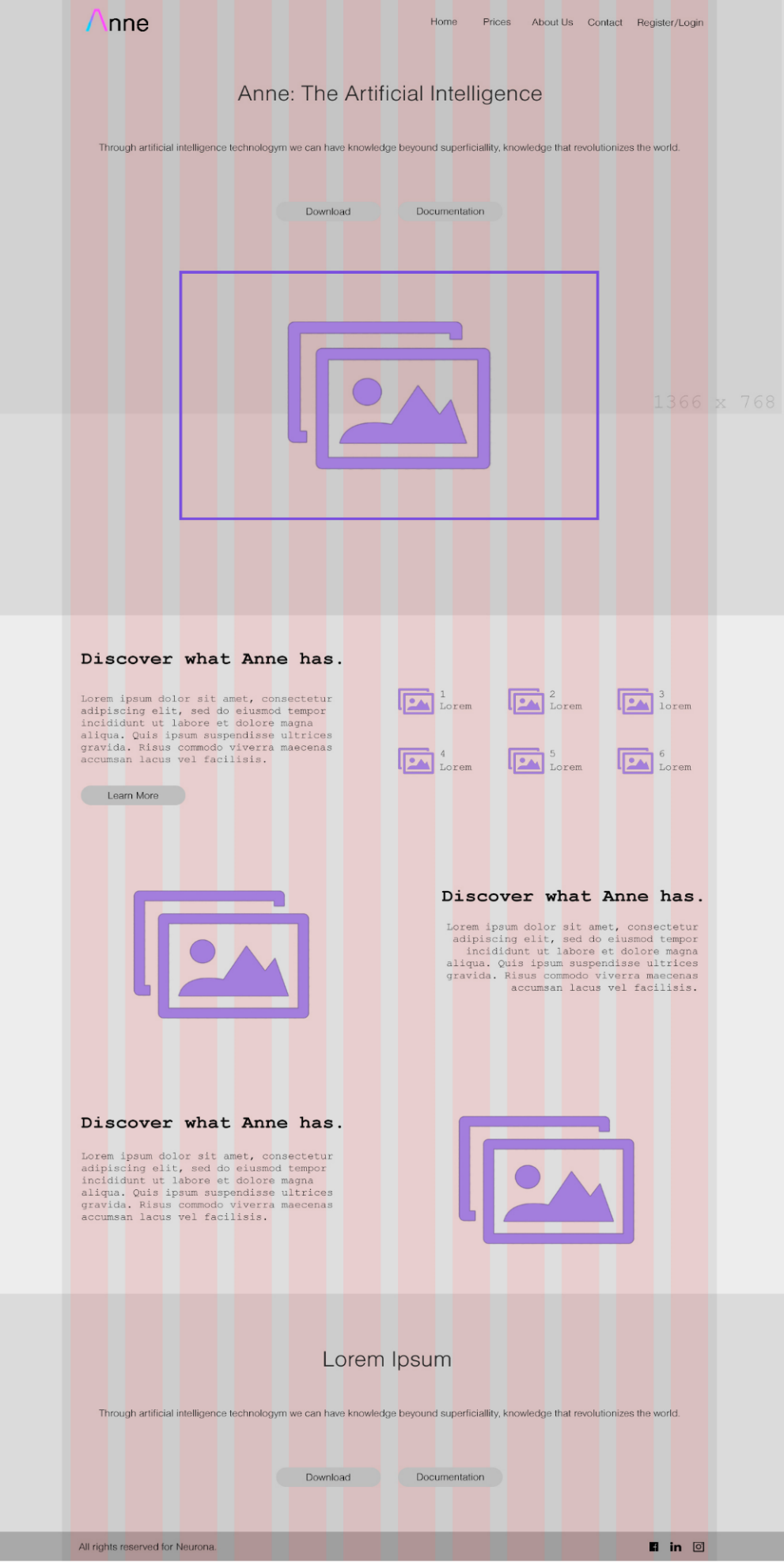


Figura 12 - Wireframe com layout grid (Home)

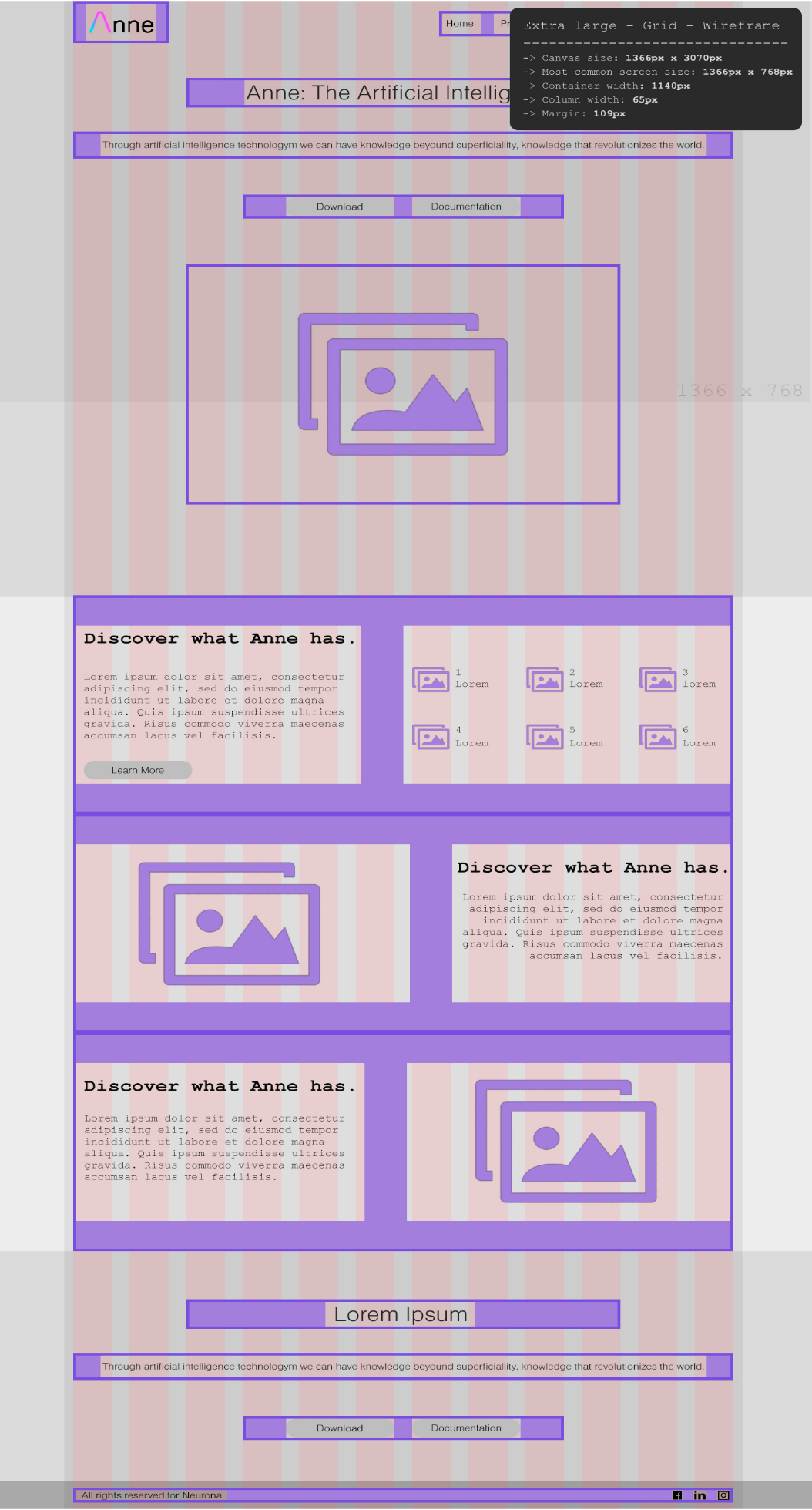


Figura 13 - Wireframe com layout grid e preenchimento de colunas (Home)

Interface gráfica do usuário, Diagrama, Teams

Descrição gerada automaticamente

Figura 14 - Wireframe da página (Sobre nós)

Uma imagem contendo texto, lousa

Descrição gerada automaticamente

Figura 15 - Wireframe com layout grid (Sobre Nós)

Uma imagem contendo Gráfico de barras

Descrição gerada automaticamente

Figura 16 - Wireframe com layout grid e preenchimento de colunas (Sobre Nós)

Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamente

Figura 17 - Wireframe (Prices)

Linha do tempo

Descrição gerada automaticamente

Figura 18 - Wireframe com layout grid (Prices)

Linha do tempo

Descrição gerada automaticamente

Figura 19 - Wireframe com layout grid e preenchimento de colunas (Prices)

Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamente

Figura 20 - Wireframe (Contatos)

Uma imagem contendo texto, lousa

Descrição gerada automaticamente

Figura 21 - Wireframe com layout grid (Contatos)

Gráfico

Descrição gerada automaticamente

Figura 22 - Wireframe com layout grid e preenchimento de colunas (Contatos)

Interface gráfica do usuário

Descrição gerada automaticamenteInterface gráfica do usuário, Teams

Descrição gerada automaticamente

Figura 23 - Wireframe com layout grid (Cadastro)

Figura 24 - Wireframe (Cadastro)

Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamenteDiagrama

Descrição gerada automaticamente

Figura 25 - Wireframe (Login)

Figura 26 - Wireframe com layout grid e preenchimento de colunas (Cadastro)

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Figura 27 - Wireframe com layout grid e preenchimento de colunas (Login)

Figura 28 - Wireframe com layout grid (Login)

Wireframe da aplicação:

Figura 29 - Tela após análise

Figura 29 - Tela após análise

### 4.3.2 MODELO RELACIONAL

#### 4.3.2.1 MODELO ER

Figura 30 – Modelo ER

#### Diagrama Descrição gerada automaticamente4.3.2.2 DIAGRAMA ER

Figura 31 - Diagrama ER

#### Tabela Descrição gerada automaticamente4.3.2.3 DICIONÁRIO DE DADOS

Figura 32 - Dicionário de dados

# 5 ARQUITETURA LÓGICA

## Diagrama Descrição gerada automaticamente5.1 DIAGRAMA DE PACOTES

Figura 33 - Diagrama de Pacote (Web Site)

Diagrama, Esquemático

Descrição gerada automaticamente

Figura 34 - Diagrama de Pacotes (Relatório)

Figura 35 - Diagrama de Pacotes(Relatório)

# 6 DESIGN ORIENTADO A OBJETOS

## Diagrama Descrição gerada automaticamente6.1 DIAGRAMA DE CLASSES

Figura 36 - Diagrama de classes OO

## Diagrama Descrição gerada automaticamente6.2 DIAGRAMA DE SEQUÊNCIA

Figura - Diagrama de sequência OO

# 7 OUTROS TÓPICOS DE ANÁLISE E DESIGN

## 7.1 DDD: DOMAIN DRIVEN DESIGN

Domain Driven Design significa Projeto Orientado a Domínio. Ele veio do título do livro escrito por Eric Evans, dono da DomainLanguage, uma empresa especializada em treinamento e consultoria para desenvolvimento de software. O livro de Evans é um grande catálogo de Padrões, baseados em experiências do autor ao longo de mais de 20 anos desenvolvendo software utilizando técnicas de Orientação a Objetos

Existe 3 pilares do DDD, o domínio, exploração de modelos de formas criativas e definir e falar a linguagem Ubíqua.

E o que seria esses 3: DOMÍNIO “*Domínio é a base, o coração de todo projeto ou negócio”,* ou seja, ele é a razão do projeto existir. É baseado em um conjunto de ideias, conhecimento e processos de negócio.

EXPLORAÇÃO DE MODELOS DE FORMAS CRIATIVAS: “*O Desenvolvedor não mais apenas codifica, ele participa de todo o processo”,* fazendo o desenvolvedor ter mais contato com o Domain Expert entre outros dentro do processo de desenvolvimento.

LINGUAGEM UBÍQUA: “*ter uma linguagem onde todos possam compreender”,* linguagem onde todos os envolvidos vão usar para melhora da comunicação, como base pode ser atribuído um glossário com as palavras que podem ocorrer confusão, deixando todos cientes de que uma vez definido o glossário, essas palavras serão utilizadas durante todo o processo.

## 7.2 TDD: TEST DRIVEN DEVELOPMENT

TDD é o Desenvolvimento Orientado por Testes (Test Driven Development). Isso mesmo! Desenvolvemos o software baseado em testes que são escritos antes do nosso código de produção, basicamente é feito alguns testes para cada funcionalidade feito em pequenos ciclos de repetições.

* Escrevemos um Teste que inicialmente não passa (Red)
* Adicionamos uma nova funcionalidade do sistema
* Fazemos o Teste passar (Green)
* Refartáramos o código da nova funcionalidade (Refactoring)
* Escrevemos o próximo Teste

este novo teste criado inicialmente falha, já que ainda não temos a implementação da funcionalidade em questão, e em seguida, implementamos a funcionalidade para fazer o teste passar, e usando essa estratégia simples, temos alguns benefícios como:

* Feedback rápido sobre a nova funcionalidade e sobre as outras funcionalidades existentes no sistema.
* Código mais limpo, já que escrevemos códigos simples para o teste passar
* Segurança no Refactoring pois podemos ver o que estamos ou não afetando
* Segurança na correção de bugs
* Maior produtividade já que o desenvolvedor encontra menos bugs e não desperdiça tempo com depuradores
* Código da aplicação mais flexível, já que para escrever testes temos que separar em pequenos "pedaços" o nosso código, para que sejam testáveis, ou seja, nosso código estará menos compacto.

Muitos não gostam da ideia de TDD pois aumenta na produção de códigos, mas usando as estratégias do TDD, é possível criar menos bugs durante o desenvolvimento, temos outros motivos para explicar o porquê a técnica é ótima, mesmo o projeto sendo de curto ou médio prazo:

* Confiança do desenvolvedor na correção de bugs, pois qualquer passo errado será mostrado pelos testes
* Tempo de desenvolvimento menor na adição de funcionalidades, já que o sistema é mais flexível e o código mais limpo
* Menor tempo do desenvolvedor ao corrigir bugs após aquelas incessantes brigas com o pessoal de qualidade depois da famosa frase, "Mas na minha máquina funciona!"
* Possibilidade de Integração Contínua, com builds automáticos e feedbacks rápidos de problemas.

Podemos fazer esse processo de uma maneira diferente, testando e depois codificando, o fato de termos um teste primeiro que o código garante que daremos passos simples para a codificação da funcionalidade, afim de fazer o teste passar, ou seja, seremos "obrigados" a escrever uma implementação simples para que o teste passe.

## 7.3 REFATORAÇÃO

A função da refatoração é melhorar a legibilidade e também a eficiência do projeto de forma que ele mantenha sua funcionalidade original deixando mais organizado e corrigindo bugs que muitas vezes atrapalham nas linhas de código, então é recomendado revisar todo código seja para verificar se tudo estar nos conformes, diminuir códigos ou concertar bugs, muitos programadores costumam dizer que quando isso ocorre significa que o código está perdendo a integridade, isso é gerado através de vários fatores desde; ter um prazo curto de entrega, falta de padronização, falta de comunicação da equipe etc. esses um dos principais motivos que geram erro dentro dos projetos, mas a também por exemplo erros na estruturas onde acaba gerando uma situação em que você deve usar a refatoração como códigos longos demais que acabam ficando muitas vezes difíceis de se compreender, trechos de códigos duplicados ou seja linhas de código que se repetem, métodos de subclasses que fazer a mesma função e funções muito longas que n são reutilizadas.

Muitas vezes no desenvolvimento do código pode surgir um bug sem que seja percebido por isso a refatoração é recomendada ser feita várias vezes à 3 situações em que a refatoração é obrigatória são elas; quando se adiciona uma nova funcionalidade à estrutura, ao corrigir um bug e também ao fazer a revisão do código que é algo bem comum.

Isso resultará em uma estrutura mais compreensiva e fácil de se entender e um baixo custo de manutenção pois um bug que é consertado em desenvolvimento do projeto custa muito mais barato que um encontrado pós lançamento. A refatoração tem como principais métodos os seguintes:

**Pull up method:** Quando duas classes contém o mesmo método elas têm que ser movidas para superclasse.

**Pull up field:** Quando o atributo está duplicado em subclasses ele também precisa ser movido para a superclasse.

**Push down method:** É o contrário dos outros métodos quando uma classe está presente na superclasse, mas ela é irrelevante apenas na subclasse ela deve ser movida para a mesma.

**Push down fiel:** Descer atributos para subclasses.

Extract subclasse: criar uma especialização para a subclasse que se aplica em funções limitadas.

**Extrac superclass:** classes com características incomuns devem ser levadas a superclasse e especializadas.

Transformar variáveis temporárias em métodos assim elas poderam também ser utilizadas por outros métodos, essa é a estrutura básica para se realizar uma refatoração.

## 7.4 DESIGN PATTERNS

Os designs patterns são basicamente padrões de projetos muitos pensam que se tratam de códigos estruturais ou wireframes que já vem prontos mas eles são padrões de uso onde você utiliza como base uma estrutura já existente padronizada como alto nível, muitos utilizam dos designs de patterns depois que sua estrutura deu algum problema, mas é muito mais conveniente utilizar os padrões desde o início do desenvolvimento, por exemplo se você começa um projeto que está ocorrendo tudo certo até tal momento que acontece um erro, e então resolve utilizar os padrões de projeto para retomar tudo, não seria mais fácil ter usado desde o início? Sim até porque eles existem para evitar erros, porem utilizar para consertar um erro já feito não está errado é apenas mais simples de se resolver.

Utiliza-los traz um grande benefício de produtividade e organização principalmente para desenvolvedores já que facilitam bastante suas vidas, além de ajudar em trabalho de equipe pois todas aquelas discussões técnicas acabam ficando mais claras tanto estruturalmente quanto visual mente. Um dos mais conhecidos são os padrões de GoF considerado uma das melhores práticas do mercado tanto que autores utilizaram do GoF para escrever o livro *“Design patterns elements of reusable object-oriented”* onde ele aplica os elementos de criação, estrutura e comportamental.

Claro que nem todos os padrões de projeto serão sempre utilizados ao decorrer do tempo alguns padrões saem para que outros entrem ou se atualizem e assim sucessivamente.

### 7.4.1 APLICAÇÃO

Para a criação de nosso projeto decidimos aplicar os conceitos de DDD e TDD. Como é um projeto novo e não temos nada hospedado na internet por agora, o DDD seria a melhor opção para o site, pois não vamos mexer nas funcionalidades dele. Quanto ao software o TDD seria uma boa opção pois, ele está em um repositório do GitHub, assim, podemos desenvolver novas funcionalidades sem interromper o funcionamento da atual versão que está disponível para download.

# 8 CONSIDERAÇÕES FINAIS

Concluímos que, através deste trabalho, é necessária a varredura frequente de websites para garantir a segurança do usuário e tornar a internet um lugar mais seguro e tranquilo para a navegação de pessoas leigas, tornando a inteligência artificial Anne um programa de extrema importância.

Além disso, foi muito enriquecedor para todos os integrantes o desenvolvimento da Anne, pois assim tivemos acesso à diversas áreas de trabalho, como design e programação.

# REFERÊNCIAS

MONTEVERDE, Wagner Aparecido. Estudo e Análise de Vulnerabilidades Web. Universidade Tecnológica Federal do Paraná, 2014. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/5823/1/CM\_COINT\_2013\_2\_02.pdf>. Acesso em: 11 de maio de 2020.

LEMOS, Robert. The state of vulnerability reports: What the CVE surge means. TechBeacon. Disponível em: <<https://techbeacon.com/security/state-vulnerability-reports-what-cve-surge-means>> . Acesso em:11 de maio de 2020.

(Monteverde,2014).<<http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/5823/1/CM_COINT_2013_2_02.pdf>>. Acesso em: 29 de maio, 14:30.

(Carvalho, Pereira, Alves, Ribeiro, 2013) – <http://revistas.unifenas.br/index.php/RE3C/article/view/60/19>. Acesso em: 29 de maio, 15:12.

OWASP, 2017 <https://wiki.owasp.org/images/0/06/OWASP\_Top\_10-2017-pt\_pt.pdf> Acesso em: 29 de maio, 15:40.

OWASP 2013 - <https://owasp.org/www-pdf-archive/OWASP\_Top\_10\_2013\_PT-BR.pdf>. Acesso em: 29 de maio, 16:20.

SUCURI, 2020. OWASP Top 10 Security Risks & Vulnerabilities – <https://sucuri.net/guides/owasp-top-10-security-vulnerabilities-2020/>. Acesso em: 29 de maio, 17:20.

CLEMENT, J. Share of global web application attack traffic as of April 2018, by originating country. 2018. Disponível em: <https://www.statista.com/statistics/276425/internet-attack-traffic-by-originating-country/>. Acesso em 26 de julho de 2020.

CAELUM ENSINO E INOVAÇÃO. UX e Usabilidade Aplicados em Mobile e Web. Acesso em: 29 de maio 17:33

BAMAM, Charles. Design de livros didáticos digitais. Acesso em: 29 de maio 15:10

SILVESTRI, Gabriel. Design de interfaces. Acesso em: 29 de maio, 14:23

Refatoração. [c2011]. Disponível em: <<https://www.devmedia.com.br/introducao-a-refatoracao/21377>> Acesso em: 14 de nov. de 2020. [c2011]

TDD. 2010. Disponível em: <https://www.devmedia.com.br/test-driven-development-tdd-simples-e-pratico/18533#:~:text=TDD%20é%20o%20Desenvolvimento%20Orientado,do%20nosso%20código%20de%20produção>> Acesso em: 14 de nov. de 2020. [c2010]

Design Patterns, 2019. Disponível em: <<https://www.opus-software.com.br/design-patterns/>>. Acesso em: 14 de nov. de 2020.

DDD. 2019. Disponível em: <https://fullcycle.com.br/domain-driven-design/>> Acesso em: 14 de nov. de 2020.

# APÊNDICES

**ANÁLISE DE PROBLEMA**

**DECLARAÇÃO DO PROBLEMA**

Segundo uma pesquisa da TIC Domicílios do NIC.br, com o crescente número de acessos à Internet, há muitas pessoas novas sem grande conhecimento na área sendo introduzidas a tudo que envolve a internet, algumas não tem um grande conhecimento sobre a tecnologia e o que pode ocorrer por trás da interface, podendo cair em plataformas vulneráveis, assim possivelmente expondo seus dados e se prejudicando. Não se limitando a apenas usuários individuais, pode ser utilizado também por empresas, que tem sites com possíveis vulnerabilidades, para gerar o relatório e ficar a par das falhas e arrumá-las com uma maior precisão.

Como a Segurança da informação anda crescendo continuamente segundo a IDC, International Data Corporation, a área de Segurança da Informação gerou cerca de 107 bilhões de dólares para o mercado em 2019. O Brasil se encontra em 10º lugar de acordo com o nível de gastos, com US$ 1,6 bilhão ao todo. O aumento do consumo em software dentro das empresas evidencia a importância na aquisição das novas tecnologias. Tendo isso em vista focamos em acompanhar o mercado juntamente de suas evoluções a fim de desenvolver uma plataforma de simples entendimentos para pessoas leigas no assunto.

A IDC ressalta, ainda, que as organizações devem realizar um reposicionamento a respeito do uso de tecnologia. Técnicas isoladas para a proteção dos sistemas apresentarão uma efetividade cada vez menor na hora de oferecer um ambiente saudável, o que vai exigir investimentos sobre diferentes tipos de processos e soluções. Logo, a capacidade de gerenciar diferentes serviços de modo exponencial não será um requisito apenas no processamento e armazenamento dos dados, mas também na segurança deles.

|  |  |
| --- | --- |
| **Elementos** | **Descrição** |
| O problema | Perigo de perda/roubo de dados pessoais na internet |
| afeta | Usuários e desenvolvedores da web |
| Devido | Vulnerabilidades em plataformas web e falta de conhecimento na área. |
| Os Benefícios desse | Mais segurança na navegação para usuários e melhor desempenho nos sites dos desenvolvedores. |

Diagrama

Descrição gerada automaticamente**ANÁLISE DAS CAUSAS RAIZES**

**USUÁRIOS E OUTROS STAKEHOLDERS**

|  |  |
| --- | --- |
| **Usuários** | **Responsabilidade** |
| Auxiliador | Auxilia os usuários |
| Desenvolvedores | Desenvolve o site e o software |
| Cliente | Utiliza o site e o software |
| Designers | Desenvolvem a parte visual do site |

Diagrama

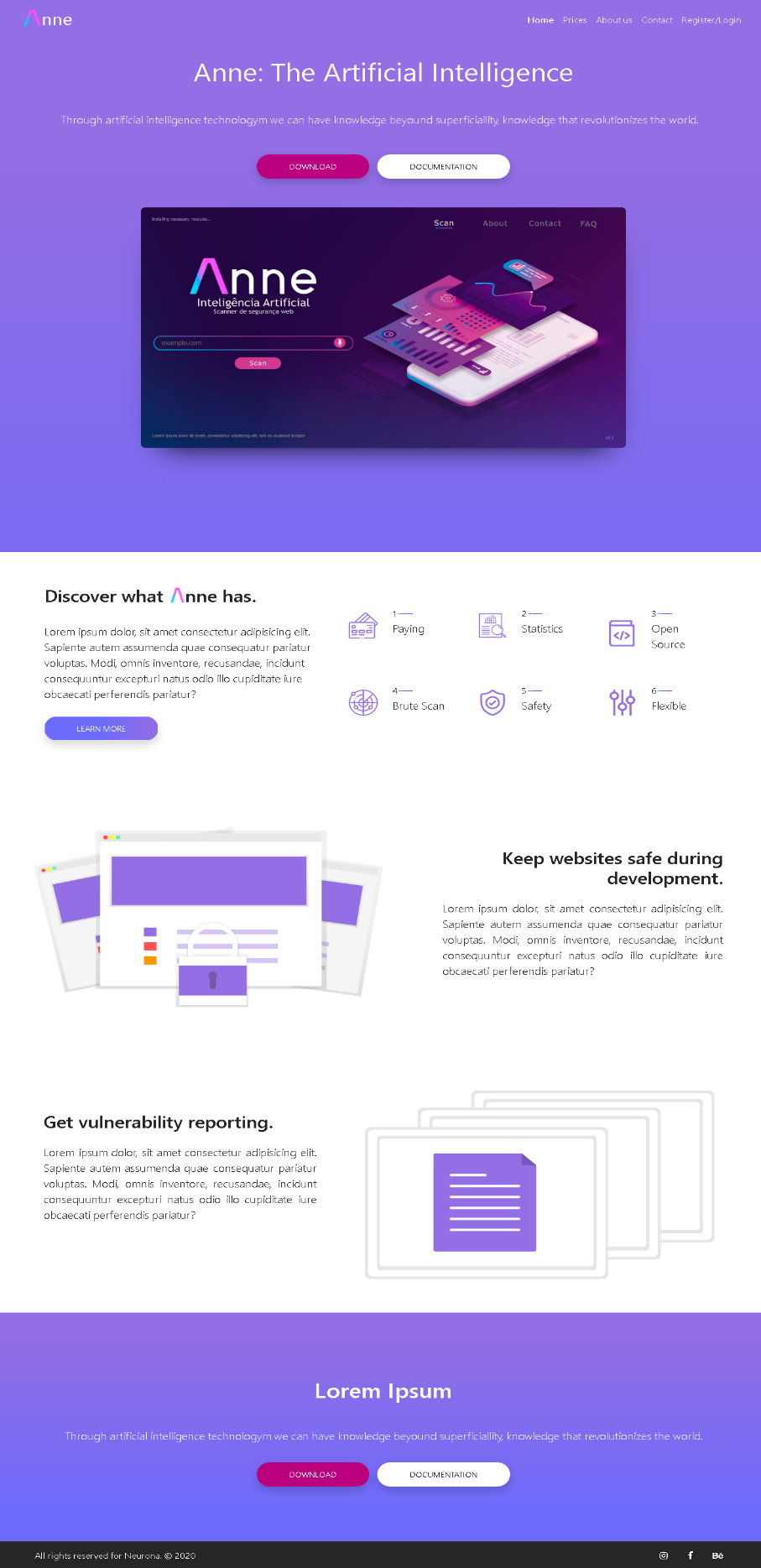
Descrição gerada automaticamente**DELIMITAÇÃO DA FRONTEIRA SISTÊMICA**

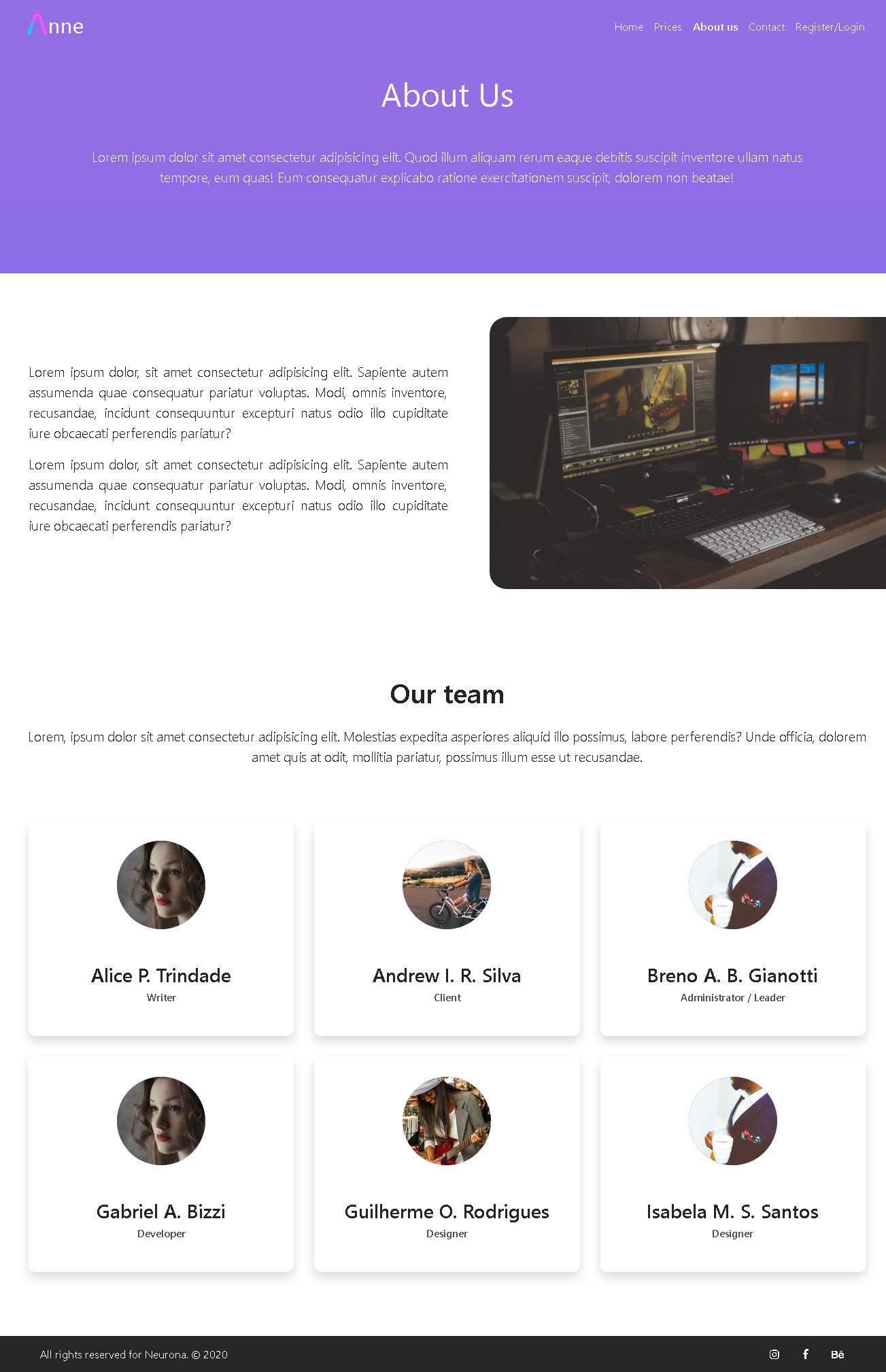
SITE E SOFTWARE

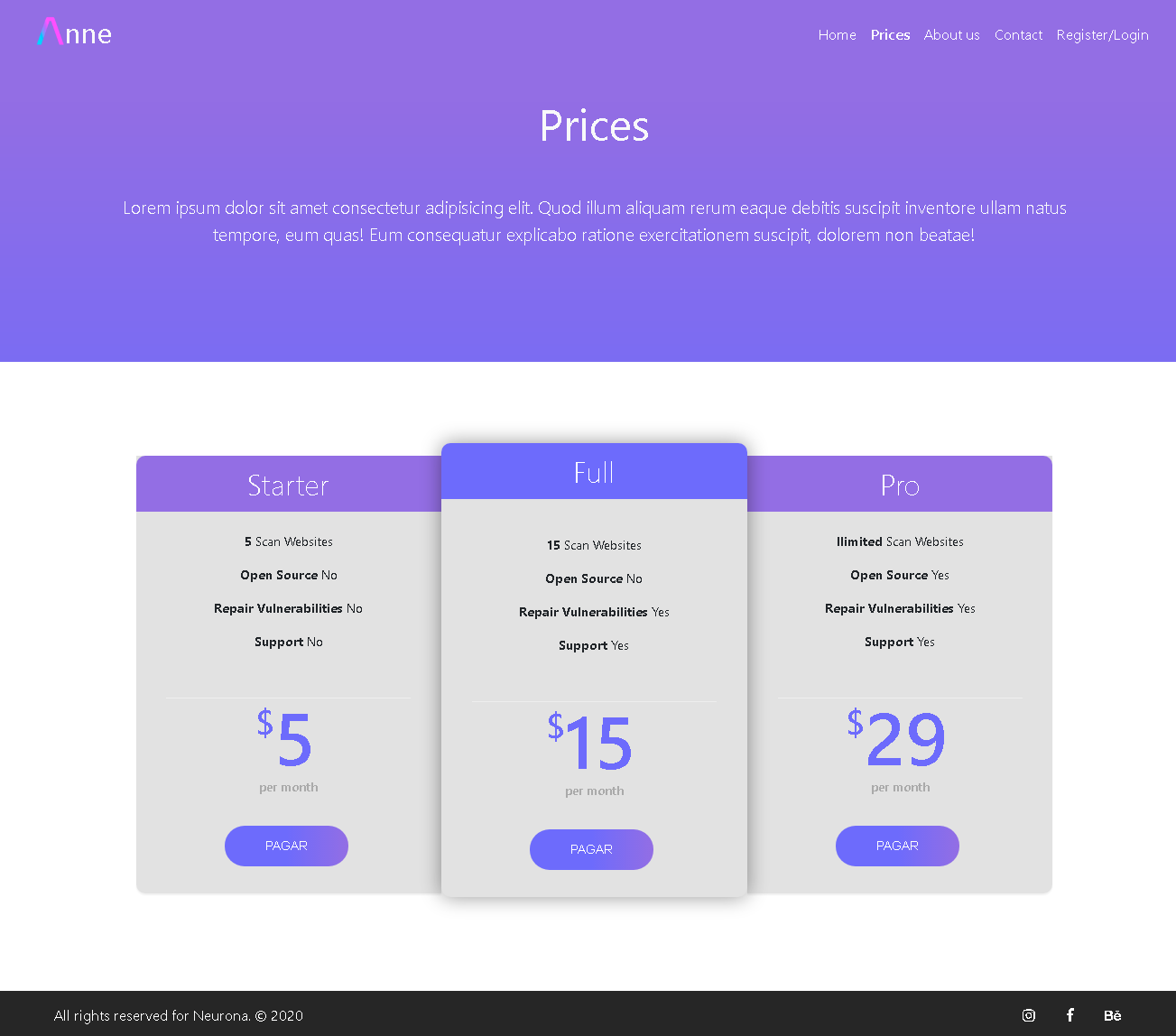
O site foi desenvolvido com Bootstrap / SCSS, Php e MySql. No qual este tem o intuito de tirar dúvidas de leigos sobre a Anne através da documentação da mesma. O sistema web conta com apenas o download, checkout e documentação. Também, sobre a equipe desenvolvedora do sistema.

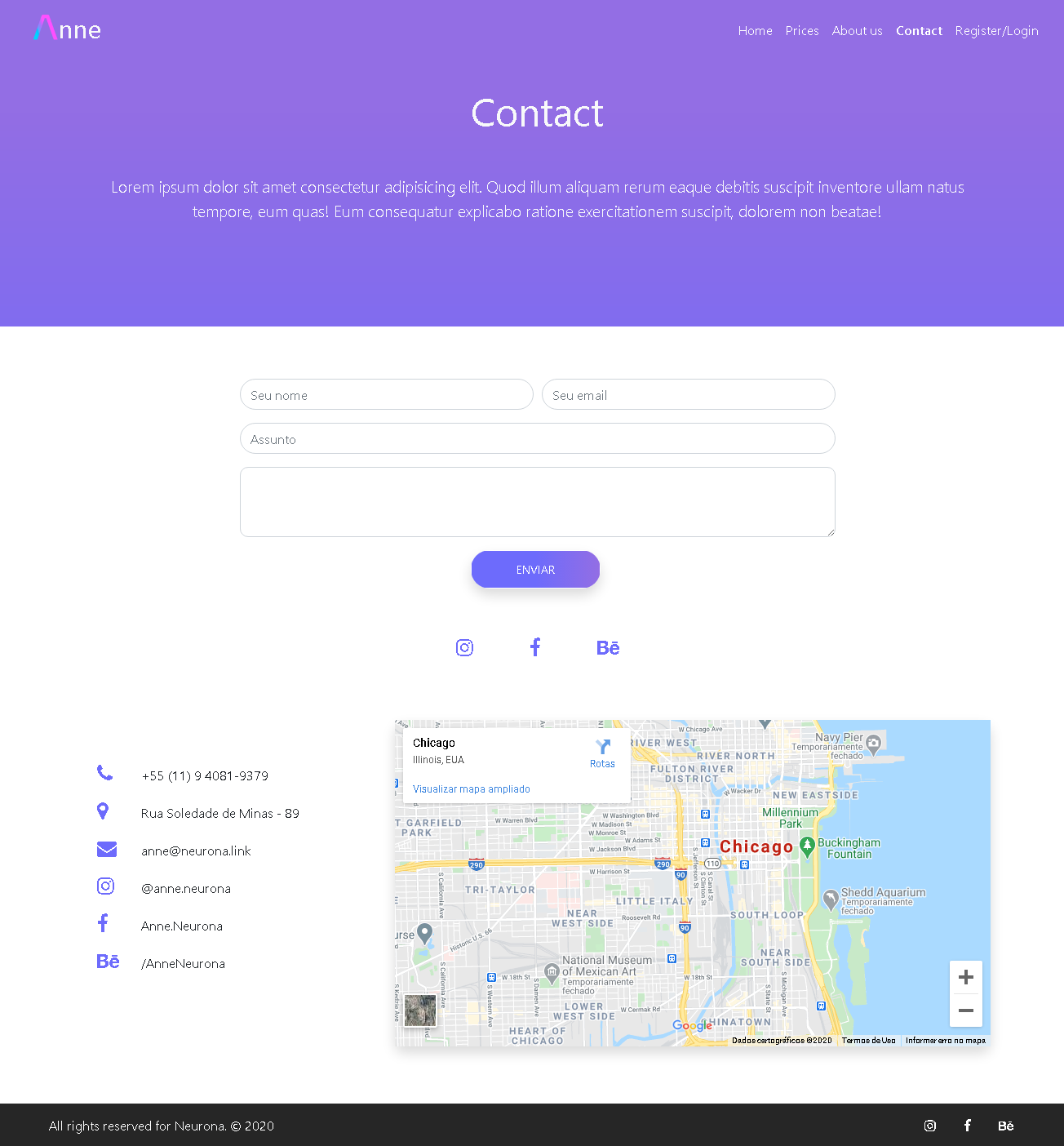
A Anne foi programada completamente em Python, desde seu processo interno até seu IU, que traz a interface do usuário. Ela funciona como um aglomerado de módulos que digitaliza o sistema web-alvo. Depois é feito o mapeamento das vulnerabilidades encontradas e, logo, apontando dicas de como evitar/resolver vulnerabilidades encontradas.

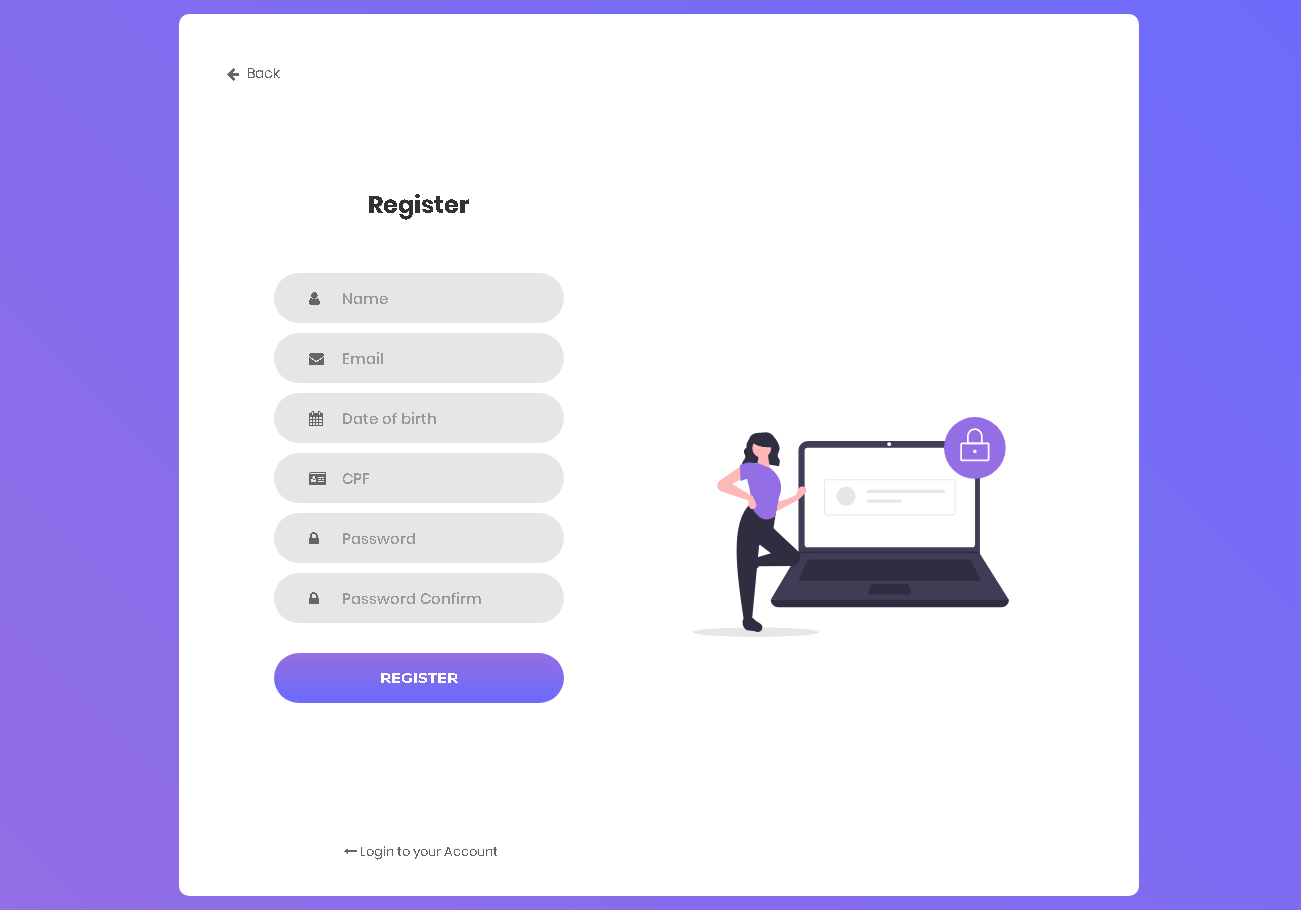
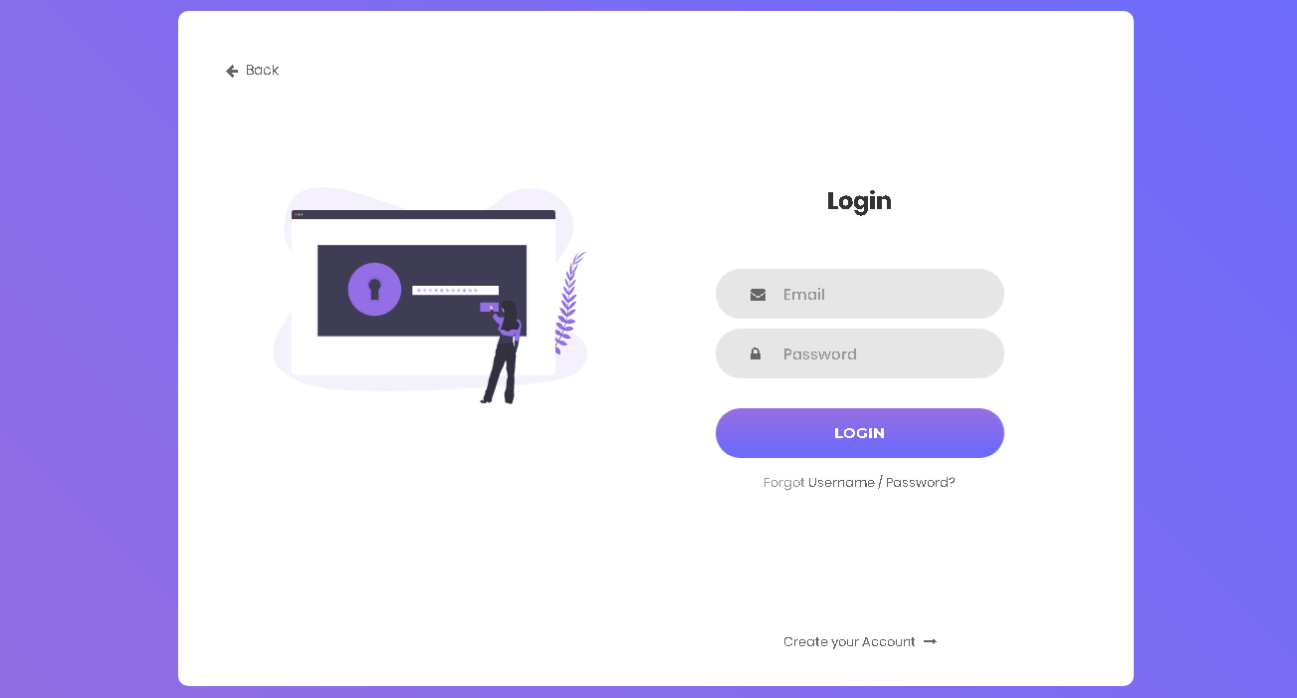
A Inteligência Artificial existente dentro do programa, é necessariamente utilizada para classificar e categorizar os o sistema web, identificando se ele está com segurança: Boa, média ou crítica.

**DESIGN**









Uma imagem contendo Linha do tempo

Descrição gerada automaticamente**MODELAGEM DE NEGÓCIOS**

**PLANO DE NEGÓCIOS**

**Sumario executivo**

* **Descrição da empresa**

A nossa empresa está desenvolvendo o que seria uma inteligência artificial apelidada de Anne, que atua analisando sites e verificando falhas na segurança dele. ela gera um relatório explicando qual seria essa falha na segurança que pode ser tanto na programação, e caso um seja um site mal intencionado, gerando conforto e segurança

* **Quem são os empreendedores**

Alice Pinheiro, Andrew Rodrigues, Breno Augusto, Gabriel Bizzi, Guilherme de Oliveira, Isabela Marques.

* **Produtos e/ou serviços que serão oferecidos**

Uma IA voltada para a segurança web.

* **Mercado de atuação**

Mercado digital.

* **Localização da empresa**

Negócio digital.

* **Missão**

Oferecer segurança e análises com objetividade e precisão

**Análise de mercado**

**Entender a situação do mercado de atuação:**

Buscamos profissionais da área de desenvolvimento de site web, eles poderão comprar a Anne a partir do nosso site ou em contato com um de nossos empreendedores, vão comprar pela incerteza na segurança do site desenvolvido, nosso produto terá como venda por um pacote anual de $$, eles devem ter no mínimo 18 anos

**Tendências de mercado:** Buscamos usuários de máquinas, como notebook e computador, que necessitam de auxílio para a navegação web e saber sobre a segurança dos sites visitados.

**Situação do mercado:** O mercado da segurança da informação está em crescimento. Existem projetos semelhantes ao nosso, mas queremos nos encaixar e aplicar mais a área.

**Concorrentes:** Redbelt, e pontuar quais as características que faz o nosso produto melhor que o do concorrente

**Plano de marketing**

Redes sociais. Facebook, Twitter, Insta, Reddit e Youtube. Fazer as propagandas a partir de vídeos promocionais e postagens nas redes.

(Assim podendo posicionar o produto perante o mercado (preço trabalhado, divulgação, canais de distribuição)).

**Plano operacional**

A equipe é de um desenvolvedor, dois designers, empresa digital, capacidade de produção, tempo necessário de desenvolvimento em cada etapa.

**Construção de cenário**

Construir diferentes situações/cenários, que podem ocorrer com a empresa ou o projeto, projetar as situações otimistas e as pessimistas ou uma mediana, entender como atuar em cada situação.

**Avaliação estratégica**

Avaliar os planos internos e externos para observar oportunidades ou possíveis ameaças, deixar claro as forças e fraquezas para entender quais os comportamentos e poder de reação.