

SMART SYSTEM TO COMBAT UNDERREPORTING OF DISEASES



SISTEMA INTELIGENTE DE COMBATE À SUBNOTIFICAÇÃO DE DOENÇAS

ALVES, Antônio Brasília de Araújo; OLIVEIRA, Luan Felipe; REIS, Rodrigo Antônio; CARVALHO, Marcos Alberto; CARVALHO, Jaqueline Corrêa Silva; SANTOS, Flávia Aparecida Oliveira Silva; BASTOS, Camila; SOUZA, Patrícia Carolina; RAMOS, Celso de Ávila; SILVA, Vinícius Duarte Esteves da

Antônio Brasília de Araújo Alves, UNIFENAS, Brasil

Luan Felipe Oliveira, UNIFENAS, Brasil

Rodrigo Antônio Reis, UNIFENAS, Brasil

Marcos Alberto Carvalho, UNIFENAS, Brasil

Jaqueline Corrêa Silva Carvalho, UNIFENAS, Brasil

Flávia Aparecida Oliveira Santos, UNIFENAS, Brasil

Camila Bastos, UNIFENAS, Brasil

Patrícia Carolina Souza, UNIFENAS, Brasil

Celso de Ávila Ramos, UNIFENAS, Brasil

Vinícius Duarte Esteves da Silva, UNIFENAS, Brasil

ABSTRACT: The general population does not have quick means of notifying health problems that can get worse without adequate assistance. This work intends to present a solution that allows the population to notify the symptoms that indicate diseases that can be treated by the health system, allowing professionals to assist these people efficiently. It is intended to make the population aware of the importance of notifying symptoms to public health professionals, allowing them to provide adequate care. As a result, a Web system open to the entire population was developed, completely free of charge, with the aim of disseminating information to combat and prevent diseases. Containing a personalized area for the administrator to publish news and images related to the information available on the platform and a personalized area for registered users, containing exclusive news and information related to their health. An area with news was also built in which the user can leave his comment for everyone to see. A prototype of the system screens was built using the Figma tool. The Cacao tool was used to create use case and class diagrams, necessary for the development of the project. For development, we used Microsoft's VsCode software as our main tool, along with its various extensions, associated with JavaScript, TypeScript, C#, and their frameworks, Angular and .NET. As data storage, PostgreSQL relational database was chosen. The Toxicity Classifier, a library with TensorFlow artificial intelligence resources, was used to prevent and classify inappropriate comments.

KEYWORDS: System, smart, underreporting.

RESUMO: A população em geral não possui meios ágeis de notificar problemas de saúde que podem se agravar sem a assistência adequada. Este trabalho pretende apresentar uma solução que permita à população notificar os sintomas que indiquem doenças passíveis de serem tratadas pelo sistema de saúde, permitindo aos profissionais assistirem estas pessoas de forma eficiente. Pretende-se conscientizar a população sobre a importância de notificar sintomas aos profissionais da saúde pública, permitindo aos mesmos providenciar os cuidados adequados. Com isso, foi desenvolvido um sistema Web aberto a toda a população, totalmente gratuito, com o intuito de divulgar informações para combate e prevenção de doenças. Contendo uma área personalizada para o administrador publicar notícias e

Revista Científica da UNIFENAS
Universidade Professor Edson Antônio Velano, Brasil
ISSN: 2596-3481
Publicação: Trimestral
vol. 6, nº. 5, 2024
revista@unifenas.br

Recebido: 08/07/2024
Aceito: 28/08/2024
Publicado: 09/09/2024

URL: <https://revistas.unifenas.br/index.php/revistaunifenas/issue/view/52>

DOI: 10.29327/2385054.6.5-2

imagens relacionadas às informações disponibilizadas na plataforma e uma área personalizada para usuários cadastrados, contendo notícias exclusivas e informações relacionadas à sua saúde. Foi construído, também, uma área com notícias em que o usuário poderá deixar seu comentário para que todos vejam. Foi realizado a construção de um protótipo das telas do sistema utilizando a ferramenta Figma. Foi utilizado a ferramenta Cacao para a criação de diagramas de casos de uso e de classes, necessários para o desenvolvimento do projeto. Para o desenvolvimento, utilizamos como ferramenta principal o software VsCode da Microsoft, juntamente com suas diversas extensões, associados às linguagens JavaScript, TypeScript, C#, e seus frameworks, Angular e .NET. Como armazenamento de dados, foi escolhido banco de dados relacional PostgreSQL. Foi utilizado o Toxicity Classifier, uma biblioteca com recursos de inteligência artificial do TensorFlow, para coibir e classificar comentários inapropriados.

PALAVRAS-CHAVE: Sistema, inteligente, subnotificação

1 INTRODUÇÃO

Atualmente, várias doenças e alguns de seus sintomas são de conhecimento da população, porém em muitos desses casos, sintomas que podem evoluir para doenças mais graves não são considerados. Esses sintomas, muitas vezes leves, são ignorados em uma consulta médica pelo próprio paciente, e que se evoluir, pode trazer grandes consequências para a sua saúde. A população em geral não possui meios ágeis de notificar problemas de saúde. Com o tempo, estes problemas podem se agravar, gerando maiores demandas e gastos no sistema de saúde.

Com a informatização e a facilidade de acesso às informações cada vez mais precisa, muitas pessoas acabam buscando soluções para seus problemas de saúde nos diversos meios de busca pela internet. Muitas das vezes acabam localizando resultados corretos, mas nem sempre esses resultados corretos levam a um diagnóstico certo, que só pode ser feito por um acompanhamento de um profissional de saúde.

A subnotificação de doença de notificação compulsória é uma irregularidade que fragiliza o encadeado sistema de saúde brasileiro, gerando prejuízos substanciais à medida que as doenças subnotificadas constituem risco à saúde da população; o conhecimento destas e de seus agravos são imprescindíveis para a promoção de ações de controle [1]. Se torna obrigatório aos profissionais da saúde notificar junto às autoridades competentes as doenças encontradas em seus pacientes, pois isso garante o investimento necessário para a prevenção de uma doença e o impedimento que novos casos possam aparecer,

seja uma doença infecciosa e transmissível ou não. Entretanto, esse primeiro contato com o profissional é de pura responsabilidade do paciente, pois visto que com um sintoma leve pode se agravar causando grandes consequências a ele próprio [1].

É possível elaborar um método ou uma solução que consiga diminuir o problema da subnotificação de doenças. Vivendo-se na era da tecnologia, pode-se utilizar de um meio em que grande parte da população tem acesso e está devidamente acostumada: a internet. Partindo-se desse princípio, uma solução Web é bastante adequada para conseguir auxiliar o SUS com as informações obtidas de pacientes e pessoas que precisam/buscam por orientação na parte saúde, na notificação de doenças.

O sistema contra a subnotificação de doenças, conta com recursos especialmente desenvolvidos para os dois lados envolvidos no percalço, tanto o profissional da saúde, quanto a pessoa que está passando por problemas e precisa notificar sua situação.

Valendo-se disso, o sistema web possui uma área que será destinada aos usuários na qual eles irão conseguir navegar pelo website, com diversas notícias, informações e um ambiente extremamente acessível e fácil para que se consiga notificar as doenças, assim como também a parte que será utilizada pelo profissional de saúde, mostrando todos os dados que se mostrem pertinentes ao avaliar o cenário de subnotificação do país (ou região escolhida pelo profissional) como um todo. Essa área contém informações da região, tipo de sintomas, agrupamento por tipos de sintomas, sexo, idade e posteriormente, quaisquer dados que sejam solicitados e se mostrem úteis. Além disso, é possível cadastrar as notícias, antes citadas, e outros tipos de informação, para manter o cidadão devidamente informado sobre o que fazer, como fazer e como agir, em situações que se demonstre necessária a notificação de doenças.

Os dois tipos de usuários, poderão ainda se comunicar através de um fórum que será disponibilizado em qualquer notícia/informação cadastrada, para que haja uma comunicação de duas vias, melhorando o desempenho e colhendo feedbacks de ambas as partes.

Visando mitigar o problema da subnotificação desses sintomas, e consequentemente das doenças que se evoluem de forma lenta, esse trabalho visa o desenvolvimento de sistema web voltado à população em geral, com o objetivo de informar e conscientizar pessoas sobre as mais variadas doenças e seus sintomas, além de como devem ser tratadas. Além disso, o sistema será utilizado para publicação de notícias, dicas de bem-estar e informações úteis no combate e prevenção de doenças.

O software visa, através da integração entre os cursos de Ciência da Computação e Biomedicina, alcançar o propósito de contribuir efetivamente com a sociedade, disponibilizando informações úteis à população acerca de cuidados com a saúde e manter contato individualizado com os cidadãos cadastrados para orientações direcionadas.

2 METODOLOGIA

2.1 Sistema inteligente

É de conhecimento geral que a internet nem sempre é o local mais amigável, visto que o anonimato e a facilidade de

expressar suas opiniões favorecem isso. Pensando neste problema, foi utilizado e aprimorado a tecnologia Toxicity Classifier. Essa biblioteca de Inteligência Artificial, criada utilizando-se do TensorFlow (uma biblioteca de código aberto para aprendizado de máquina aplicável a uma ampla variedade de tarefas), tem o objetivo de coibir e classificar comentários inapropriados, de acordo com sua toxicidade, nível de machismo, homofobia, racismo, xenofobia e ofensividade. O Toxicity Classifier, originalmente capaz de entender a toxicidade apenas em inglês, foi adaptada para que consigamos traduzir as palavras enviadas ao fórum em tempo real e após classifica-las, o comentário será barrado, sendo enviado aos administradores para posterior aprovação ou negação do comentário, podendo-se ter a opção de negar automaticamente todos comentários impróprios [2].

2.2 Desenvolvimento da aplicação

Para o desenvolvimento desse software, foram utilizadas algumas ferramentas e frameworks disponibilizados gratuitamente, com o objetivo de atender a cada um dos propósitos à que eles servem. A arquitetura Cliente/Servidor, uma estrutura de aplicação distribuída que distribui as tarefas e cargas de trabalho entre os fornecedores de um recurso ou serviço, designados como servidores, e os requerentes dos serviços, designados como clientes [3].

Seguindo esse conceito, a aplicação desenvolvida nesse trabalho é baseada no modelo de arquitetura cliente/servidor, sendo dividida em dois projetos: uma desenvolvida em Angular, contando com a biblioteca Toxicity Classifier; e uma API desenvolvida em .NET utilizando Entity Framework com banco de dados em PostgreSQL.

2.2.1 Visual Studio Code

Para o desenvolvimento do sistema, foi utilizado o Visual Studio Code. Ele é um editor de código-fonte desenvolvido pela Microsoft para Windows, Linux e macOS. É um software livre e de código aberto, baseado no Electron, um framework que é usado para desenvolver aplicativos desktop utilizando uma estrutura baseada em web. O desenvolvimento de toda parte de digitação de código envolveu o VSCode, bem como o uso de suas diversas extensões.

2.2.2 Framework Angular

O Angular, uma estrutura feita em JavaScript, é uma plataforma de aplicações web de código-fonte aberto e front-end baseado em TypeScript. Criado por Miško Hevery e Adam Abrons em 2009, é uma plataforma para a construção da interface de aplicações usando HTML, CSS e, principalmente, JavaScript para manipular elementos de forma eficiente. Com o foco em criar SPAs, o Angular utiliza de conceitos como componentes, templates,

diretivas, roteamento, módulo, serviços, injeção de dependências e ferramentas de infraestrutura. Utilizando o framework o desenvolvimento de 17.000 linhas de código inicial pode ser reduzido em 1.500 linhas de código, aumentando drasticamente a produtividade do desenvolvedor [3].

A parte visual em que o cliente vê da aplicação, denominamos de “front”. Nessa aplicação o front foi totalmente desenvolvido nas linguagens HTML, CSS e Javascript, utilizando as facilidades do framework Angular na versão 14, que é a versão atualmente estável. Utilizando-se de rotas, uma aplicação SPA pode conter “páginas” navegáveis, visto que essas rotas separam qual o componente será carregado em determinada região em que o cliente está. Com isso, foram feitas as páginas da aplicação com as áreas destinadas aos usuários cadastrados, ao público geral (Figura 1) e a área administrativa (Figura 2).

Figura 1. Protótipo da tela inicial

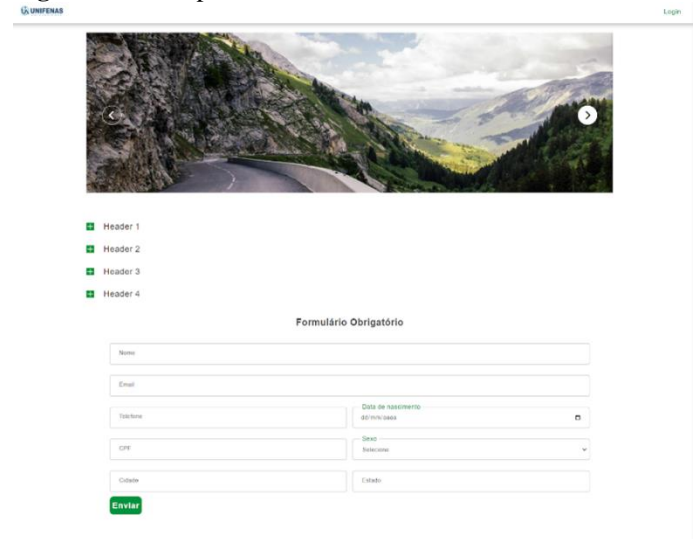
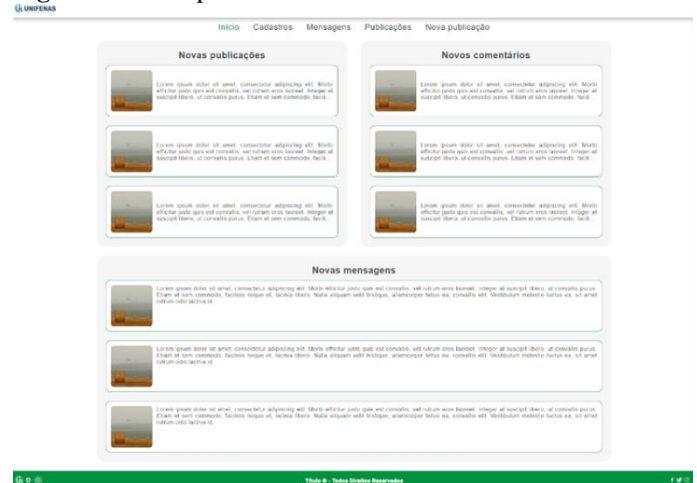


Figura 2. Protótipo da tela administrativa



2.2.3 Microsoft .NET

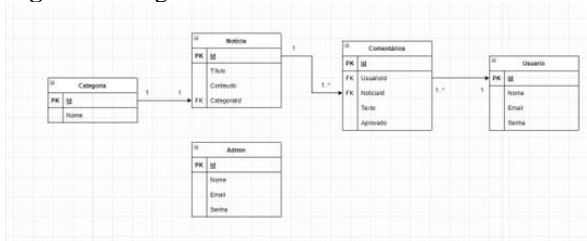
O .NET é uma plataforma de código aberto para a criação de aplicações de desktop, Web e móveis que podem ser executadas nativamente em qualquer sistema operacional. O sistema .NET inclui ferramentas, bibliotecas e linguagens que oferecem suporte ao desenvolvimento de software

moderno, escalável e de alta performance [4]. O .NET é gratuito, de código aberto e é um projeto da .NET Foundation. O .NET é mantido pela Microsoft e pela comunidade no GitHub em vários repositórios.

Nesse sistema, foi criada uma API contendo os métodos CRUD necessários para a aplicação funcionar, com a segurança necessária para permitir o acesso por meio de requisições cliente/servidor somente para a aplicação em questão. O Entity Framework permite que os desenvolvedores trabalhem com dados na forma de objetos e propriedades específicos de domínio, como clientes e endereços de clientes, sem ter que se preocupar com as tabelas e colunas de banco de dados subjacentes em que esses dados são armazenados [4].

Utilizando as facilidades que o Entity Framework possui, o mapeamento das classes, bem como sua alocação no banco de dados, se torna muito prático. Assim, foi construída uma aplicação back-end no padrão MVC, que é uma arquitetura composta por três elementos, Model (Modelo), View (Visão) e Control (Controle), porém sem o elemento da View, que está representada como parte da aplicação do front, feito em Angular. Desse modo as classes foram construídas via código e migradas para o banco de dados. A figura 3 apresenta o diagrama de classes considerado para a aplicação.

Figura 3. Diagrama de classes



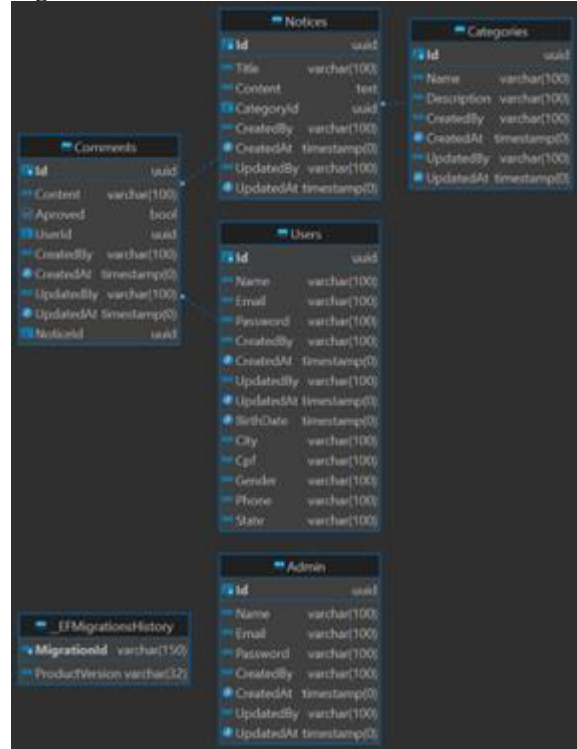
A parte da regra de negócio, utilizando o Entity Framework, foram criadas interfaces de acesso para a conexão com o banco de dados que, utilizando a biblioteca Npgsql, que é um plugin open source utilizado para a conexão entre a aplicação e o banco de dados, fazem todo processo de CRUD do sistema.

2.2.4 Banco de Dados PostgreSQL

Um SGBD é o conjunto de programas de computador (softwares) responsáveis pelo gerenciamento de bases de dados. O principal objetivo é retirar da aplicação cliente a responsabilidade de gerenciar o acesso, manipulação e organização dos dados. O SGBD disponibiliza uma interface para que os seus clientes possam incluir, alterar ou consultar dados. Em bancos de dados relacionais a interface é constituída pelas APIs ou drivers do SGBD, que executam comandos na linguagem SQL [6]. O PostgreSQL é um poderoso Gerenciado de

Banco de Dados Relacional que é caracterizado por ser um SGBD open source e que ganhou uma grande popularidade nos últimos anos devido sua facilidade no manuseio. Sua abordagem é relacional, ou seja, os dados estão organizados em forma de tabela (Figura 4, garantindo assim fácil integração com uma API desenvolvida com o auxílio do Entity Framework.

Figura 4. Modelo ExR



2.2.5 TensorFlow Toxicity

Em 2011, a equipe do Google Brain (segmento do Google voltado pesquisa na área de inteligência artificial) criou um sistema chamado DistBelief, que era voltado para redes neurais e deep learning. Como a utilização estava crescendo dentro da empresa, o Google designou uma equipe de cientistas da computação para redesenhar o DistBelief, com o objetivo de torná-lo mais simples e mais rápido. Esse aperfeiçoamento acabou transformando o DistBelief no TensorFlow.

O TensorFlow é uma biblioteca de código aberto utilizada para o desenvolvimento de aplicações com o uso de machine learning. Nessa biblioteca temos vários modelos de machine learning com bases gratuitas já treinadas para uso nas mais diversas aplicações. Em 30 de março de 2018, foi anunciado pelo Google o TensorFlow.js, que trouxe a biblioteca para JavaScript. Com ele, você pode importar modelos existentes ou já treinados para realizar inferências.

Além disso, é possível treinar novamente os modelos importados ou desenvolver novos modelos de aprendizado de máquina diretamente no seu browser, utilizando JavaScript. Dentro da biblioteca do TensorFlow, temos o Toxicity Classifier que é um verificador e classificador de ofensividades. O modelo Toxicity é responsável por analisar o conteúdo digitado pelo usuário, classificar e retornar o tipo de toxicidade daquele conteúdo.

Para o modelo Toxicity utilizado nesse sistema, utilizou-se uma base de autoria da Google com aproximadamente 2 milhões de comentários que é alimentada e atualizada constantemente pela comunidade. Entretanto, o conteúdo dessa base está totalmente em inglês, sendo assim, foi preciso a utilização de uma API de terceiros que nos fornecem tradução instantânea entre o conteúdo no sistema e o conteúdo do Toxicity. As figuras 5 e 6 mostram a utilização desta API.

Figura 5. Parte de comentários do sistema

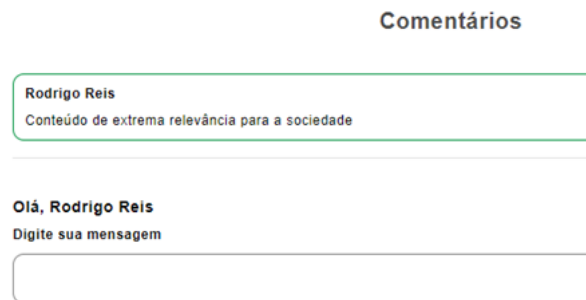


Figura 6. Função que utiliza os serviços do Toxicity Classifier

```

toxicityVerify(text: string) {
  this.toxicityService.toxicityModel
    .classify(text)
    .then((predictions: any) => {
      const mappedData = predictions.map((data: any) => {
        return {
          label: data.label,
          match: data.results[0].match
        };
      });
      this.insertComment(mappedData);
    }).catch((err: any) => { console.log(err) });
}
    
```

O TensorFlow atua como parte integrante dos comentários das notícias do sistema. A cada comentário feito por um usuário, ele passa pela verificação do Toxicity que nos retorna o nível de toxidade daquele comentário. Caso não possua nenhum dos níveis de toxidade informado pelo Classifier ele automaticamente é adicionado à lista de comentários permitidos na publicação, caso contrário ele é enviado para avaliação e aprovação dos moderadores e administradores do sistema. A figura 7 mostra exemplos de aprovação de comentários.

Figura 7. Aprovação de comentários

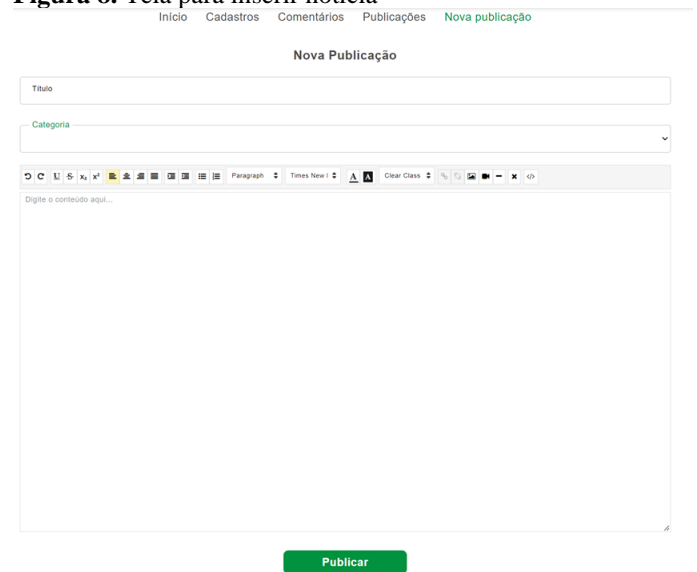
Comentários							
Notícia	Nome	Email	Whatsapp	Data	Mensagem	Status	Ações
teste	Rodrigo Reis	rodrigo.reis@aluno.unifenas.br	(35) 99151-9999	06/12/2022	Conteúdo de extrema relevância para a sociedade	Aprovado	🗑️
teste	Rodrigo Reis	rodrigo.reis@aluno.unifenas.br	(35) 99151-9999	06/12/2022	uma notícia idônea	Reprovado	🗑️

3 RESULTADOS E DISCUSSÃO

Com as ferramentas mencionadas, foi criada a primeira versão inicial do software contendo alguns recursos básicos como um portal de notícias, com o intuito de divulgar informações e orientações para as pessoas, uma parte específica para comentários (sendo essa um fórum capaz de filtrar comentários impróprios, caso estes contenham toxidade e sejam ofensivos) e áreas de acesso restrito para usuários cadastrados e para os administradores, cada qual com sua funcionalidade específica, que auxilia na navegação do site.

O software conta com um dashboard exclusivo para o administrador. Neste local, é possível inserir notícias contendo imagens e qualquer outro conteúdo de interesse para todos com suporte à seleção de categorias, como mostrado na Figura 8.

Figura 8. Tela para inserir notícia



Como mostrado na Figura 9, o administrador possui acesso aos dados dos usuários cadastrados, seguindo todas as normas da LGPD com relação à dados sensíveis, como telefone, cidade, data de nascimento, etc.. que podem ser usados para entrar em contato com o mesmo, ou para fins de análise, podendo extrair informações como, qual a região que possui mais notificações e qual a faixa etária predominante.

Figura 9. Dados dos usuários cadastrados

Cadastros				
Nome	Documento	Data de Nascimento	Cidade	Informações
Rodrigo Reis	205.255.200-50	20/02/2000	Atenas - MG	(35) 9 9161-9999 rodrigo_reis@aluno.unifenas.br
Luan Felipe	696.201.060-59	20/02/2000	Atenas - MG	(35) 9 9916-0559 luan_ofenra@aluno.unifenas.br
Antonio Brasili	075.378.940-00	20/02/2000	Atenas - MG	(35) 4 9434-0448 antonio_brasili@aluno.unifenas.br

Também, conta com uma tela de aprovação de comentários que foram feitos nas notícias publicadas (Figura 10). Nessa tela é possível alterar o “status” de um comentário que foi aprovado ou reprovado de acordo com o Toxicity Classifier, que é o classificador inteligente de toxidade da aplicação.

Figura 10. Tela de comentários

Comentários

Notícia	Nome	Email	Whatsapp	Data	Mensagem	Status
Lorempsum	Rodrigo Reis	rodrigo.reis@aluno.unifenas.br	(35) 99161-9999	03/12/2022	legal	Aprovar
Lorempsum	Rodrigo Reis	rodrigo.reis@aluno.unifenas.br	(35) 99161-9999	03/12/2022	teste	Aprovar

Já na figura 11 são mostradas todas as notícias que estão publicadas no portal desde o início do projeto, tendo uma prévia da data em que foi publicado, título e uma breve parte do conteúdo. É possível também a exclusão, de maneira rápida e ágil, das notícias, basta que o administrador do sistema clique no botão vermelho na parte direita da notícia e confirme a exclusão.

Figura 11. Tela de notícias publicadas

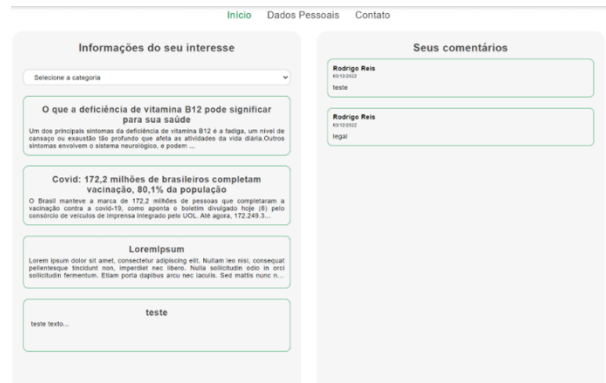
Publicações

Data	Título	Conteúdo	
01/12/2022	teste	teste teste...	✖
01/12/2022	Lorempsum	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam leo nisi, consequat pellentesque tristique non, imperdiet nec libero. Nulla sollicitudin odio in orci sollicitudin fermentum. Etiam porta dapibus arcu nec laculis. Sed mattis nunc nec mi...	✖
06/12/2022	Covid: 172,2 milhões de brasileiros completam vacinação, 80,1% da população	O Brasil manteve a marca de 172,2 milhões de pessoas que completaram a vacinação contra a covid-19, como aponta o boletim divulgado hoje (6) pelo consórcio de veículos de imprensa integrado pelo UOL. Até agora, 172.249.3...	✖
06/12/2022	O que a deficiência de vitamina B12 pode significar para sua saúde	Um dos principais sintomas da deficiência de vitamina B12 é a fadiga, um nível de cansaço no indivíduo tão profundo que afeta as atividades da vida diária.Outros sintomas envolvem o sistema neurológico, e podem ...	✖

O software conta, ainda, com um sistema de login e autenticação de usuários, em que os mesmos podem se cadastrar e fazer o login com e-mail e senha. O intuito da página de login, é que seja simples e preciso, sem muitas informações ou poluição visual, utilizando técnicas de UI/UX. O sistema de cadastro é bem objetivo e priorizando a clareza dos conteúdos e com fontes e cores recomendadas pelas práticas de UI/UX para que os usuários não tenham dificuldade para se cadastrar. A parte do cadastro é opcional, não sendo obrigatório para que o usuário disfrute da maior parte do conteúdo do site. Porém, caso o usuário tenha o desejo de cadastrar, as informações sensíveis serão mantidas em sigilo e de acordo com todas as regras e especificações da LGPD.

Os usuários que decidirem se cadastrar e usufruir de uma experiência mais personalizada e única, poderão acessar a página de início após o login, com duas opções disponíveis. A primeira das opções, situada na parte “Informações do seu interesse”, o usuário pode escolher qual a categoria de maior interesse, para que o sistema filtre dentre as opções disponíveis e mostre com maior frequência e prioridade a categoria selecionada. A segunda opção, “Seus comentários”, tem o propósito de mostrar todos os comentários feitos pelo usuário logado, mostrando a data, o próprio comentário e o seu nome que foi utilizado na época em que o comentário foi publicado.

Figura 12. Tela de início do usuário logado



O sistema possui uma área de “Contato”. Neste local, há as opções de enviar uma mensagem, tais como informações sobre telefone e email. Caso a opção de mensagem seja escolhida, o texto que for enviado após clicar no botão “enviar”, irá para um email escolhido para os administradores, conseguindo analisar e posteriormente responder o feedback de maneira privada ao usuário (A informação para contato utilizada será a mesma cadastrada no sistema).

Além disso, há uma página inicial que contém algumas informações de relevância para instigar a pessoa a se cadastrar no sistema, com um formulário simples e de fácil acesso para esse cadastro. Nessa primeira página há também um carrossel de imagens para estimular os visitantes a visitar o portal de notícias e ver o que está sendo publicado pelos administradores.

O portal de notícias, é a página principal do sistema. Nele são mostradas as informações principais e mais recentes do sistema. Essas informações cadastradas são as mesmas postadas pelos administradores na aba de cadastrar notícias. Assim como mostrado na figura 13, a primeira parte conta com briefing de algumas notícias, mostrando o título em evidência, uma imagem (caso tenha uma) e uma breve parte do texto, para introduzir a pessoa que está lendo ao assunto e contexto.

Figura 13. Portal de notícias



Logo abaixo da notícia há a parte de comentários. Caso o usuário seja um visitante novo ou não tenha logado previamente, a seção de comentários estará indisponível até que o mesmo se cadastre. Após o login desse usuário o acesso se torna possível e, obedecendo todas as regras de boa conduta na internet, pode fazer seu comentário livremente. A parte do comentário é gerenciada em parte pela IA do

TensorFlow Toxicity Classifier, que bloqueia qualquer comentário impróprio e acrescenta em uma lista de comentários para que possa ser verificado e validado pelos administradores do sistema que, se aprovado futuramente, irá ser exibido.

As limitações desse trabalho residem na insuficiência de recursos para propagar e divulgar o site, assim como a compra de um domínio para mantê-lo no ar. Assim não foi possível o seu teste em um real caso de acesso simultâneo de diversas localizações do país.

CONCLUSÃO

O objetivo geral desse trabalho foi desenvolver um sistema que permita, de maneira acessível e fácil, que as pessoas consigam notificar suas doenças aos profissionais de saúde, ao mesmo tempo conseguindo orientações e informações úteis sobre a situação atual. Os profissionais por sua vez irão obter dados cruciais para conseguir desenvolver medidas e tomarem decisões que, com as informações recebidas, visam melhorar cada vez mais o sistema de saúde brasileiro.

Considerando os resultados obtidos, e as limitações existentes neste trabalho, pode-se indicar, para pesquisas futuras, a aplicação de novas funcionalidades e melhorando nas estruturas já existentes, com a finalidade de atender melhor às necessidades do usuário, produzindo dados mais completos e seguros.

Esse sistema web desenvolvido também é passível de adaptações, tornando-se algo que esteja em constante evolução para atender as mais diversas mudanças na área da saúde, tudo isso em conjunto com as necessidades dos profissionais da área.

REFERÊNCIAS

[1] SOUZA, H. C. de; ARCELINO, L. A. M.; TRINDADE, R. F. C. Doenças de Notificação Compulsória: uma análise dos atendimentos ambulatoriais no Hospital Universitário Prof. Alberto Antunes (HUPAA). 2006.

[2] PAVLOPULOS et al. Toxicity Detection: does Context Really Matter? arXiv:2006.00998v1 [cs.CL] 1 Jun 2020. Disponível em: <https://arxiv.org/format/2006.00998>>. Acesso em: 6 de março de 2024.

[3] MENDES, Antonio. Arquitetura de Software: desenvolvimento orientado para arquitetura. Editora Campus. Rio de Janeiro - RJ, 2002.

[4] BRANAS, Rodrigo. AngularJS Essentials. Packt Publishing, 2014.

[5] AWS, Amazon. O que é .NET. AWS, 2022. Disponível em <<https://aws.amazon.com/pt/what-is/net>>. Acesso em 6 de março de 2024.

[6] MICROSOFT. Visão Geral do Entity Framework. Microsoft Learn, 2022. Disponível em <<https://learn.microsoft.com/pt-br/dotnet/framework/data/adonet/ef/overview>>. Acesso em 6 de março de 2024

[7] SAÚDE, Ministério da. Sistema Gerenciador de Banco de dados. Datasus, 2019. Disponível em <<https://datasus.saude.gov.br/glossario/sistema-gerenciador-de-banco-de-dados-sgbd/>>. Acesso em 6 de março de 2024.

