

COVIDAPP: Aplicativo para triagem de sintomas e obtenção de informações sobre a COVID-19

Natalia Maria K. da Silva¹, Daniela Sbizera Justo²

¹Instituto Federal de Santa Catarina – Campus Gaspar (IFSC)
R. Adriano Kormann, 510 – 89111-009 – Gaspar, SC – Brasil

²Instituto Federal de Santa Catarina – Campus Gaspar (IFSC)
R. Adriano Kormann, 510 – 89111-009 – Gaspar, SC – Brasil

natalia.mks@aluno.ifsc.edu.br, daniela.sbizera@ifsc.edu.br

Abstract. *Amid the COVID-19 pandemic, which more than two million cases had been hit yet (BRASIL, 2020), it was noticed the need for the development of an app to help people gather clear and accessible information about the disease, quantities of sick and recovered people, and symptoms screening. The project was developed by partnering with Deusto University (Spain), using technologies for mobile development. A comparison with related applications was made to determine the needed features and a user satisfaction survey proved that the app features and design are effective.*

Resumo. *Em meio à pandemia do COVID-19, que até o momento no Brasil já atingiu mais de dois milhões de casos (BRASIL, 2020), notou-se a necessidade do desenvolvimento de um aplicativo para auxiliar a população na obtenção de informações claras e acessíveis sobre a doença, quantidades de recuperados e de doentes e triagem de sintomas. O projeto foi desenvolvido em parceria com a Universidade de Deusto (Espanha), utilizando tecnologias de desenvolvimento para plataformas móveis. Um comparativo com aplicativos correlatos foi realizado para determinar as funcionalidades necessárias e uma pesquisa de satisfação com usuários demonstrou que o design e a funcionalidades do aplicativo são eficazes.*

1. Introdução

Neste artigo apresenta-se uma descrição do projeto desenvolvido em cooperação com o eVida Lab, que se trata de um grupo de pesquisa da Universidade de Deusto (Espanha). Devido a pandemia do COVID-19, com início no final de 2019 na província de Wuhan, na China, e que rapidamente se espalhou pelo mundo, esse novo coronavírus foi escolhido como objeto de pesquisa deste projeto devido à grande importância deste assunto mundialmente, e porque também ocasionou impactos na saúde e economia do mundo inteiro, forçando países a adotarem medidas de isolamento em massa e de distanciamento social.

Como Bill Gates (2015) disse em um TED Conferences, o mundo contemporâneo não está preparado para esse tipo de surto, em termos de (i) vigilância e dados, (ii) pessoal e (iii) tratamento. Especialmente, não foi qualificado nesses termos para uma resposta rápida e coordenada em escala global. Atualmente na Era da Informação, acredita-se que as pessoas podem coletar mais dados por meio da tecnologia existente (como aplicativos para *smartphones*), a fim de melhorar as políticas públicas e as ações governamentais. Diante deste panorama, este trabalho visa criar uma

aplicação móvel, a fim de ajudar a melhorar o diagnóstico do COVID-19 e fornecer melhores informações ao público em geral.

2. Objetivos

A seguir nas subseções 2.1 e 2.2 são abordados o objetivo geral do projeto com ênfase no desenvolvimento do aplicativo e seus objetivos específicos divididos em tecnológicos e sociais.

2.1. Objetivo geral

O foco principal deste trabalho é projetar e implementar um aplicativo móvel para plataforma Android, para a população geral, a fim de fornecer uma interface amigável para obter informações atualizadas e confiáveis sobre o COVID-19 e também para compartilhar o estado de sua saúde. É importante observar que, embora o aplicativo tenha sido projetado para sistemas Android, devido à tecnologia escolhida (Dart/Flutter), ele também pode ser implementado em *smartphones* baseados em IOS.

2.2. Objetivos específicos

Os objetivos específicos para maior clareza foram divididos em duas vertentes: objetivos tecnológicos na subseção 2.2.1 focado no desenvolvimento do aplicativo, e objetivos sociais na subseção 2.2.2 que se trata dos objetivos que focam em benefícios para os usuários do aplicativo.

2.2.1. Objetivos tecnológicos

- Projetar e implementar uma arquitetura que possa ser facilmente escalável;
- Projetar e implementar um aplicativo Android;
- Projetar e implementar um banco de dados que pode escalar e suportar mudanças nas estruturas dos dados armazenados;
- Projetar e implementar um servidor no qual os usuários possam compartilhar suas informações - em um formato que facilite a leitura e utilização destes dados;
- Coletar dados de fontes confiáveis e atualizadas:
 - ◆ API da *John Hopkins - Center for Systems Science and Engineering* (CSSE).

2.2.2. Objetivos sociais

- Projetar e implementar um aplicativo que obtenha informações confiáveis e atualizadas sobre o COVID-19 considerando a base de dados da *John Hopkins - Center for Systems Science and Engineering* (CSSE) sobre o mundo inteiro ou sobre qualquer país em particular;
- Projetar e implementar um aplicativo que obtenha informações confiáveis e atualizadas sobre o COVID-19 a partir de canais de notícias em todo o mundo;
- Projetar e implementar um aplicativo em que o usuário possa compartilhar informações sobre seu estado de saúde: se o mesmo foi testado para COVID-19, o resultado do teste e se apresenta algum sintoma.

3. Materiais e métodos

Nesta seção são apresentadas as fases executadas para o desenvolvimento do aplicativo, bem como uma análise das ferramentas utilizadas.

3.1. Fases

Na Figura 1 mostra-se as fases executadas e nos tópicos seguintes uma breve explicação de cada passo.

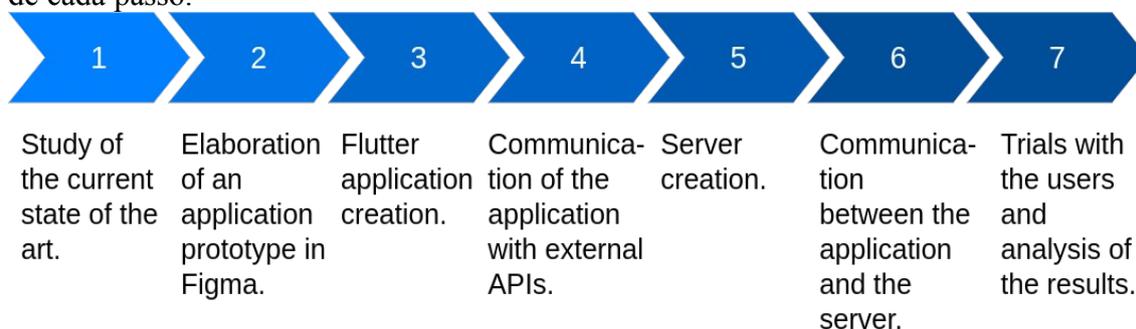


Figura 1. Fases no desenvolvimento do aplicativo

3.1.1. Fase 1: Estudo do estado da arte

- Objetivo: Análise do estado da arte
- Resultado: Definição dos objetivos e escopo do projeto.

3.1.2. Fase 2: utilizando a ferramenta colaborativa de design

- Objetivo: Desenhar um protótipo das telas do aplicativo.
- Resultado: Uma base das interfaces para desenvolvimento das telas do aplicativo e o fluxo do usuário na utilização do aplicativo.

3.1.3. Fase 3: Criação do aplicativo Flutter

- Objetivo: Projeto e implementação do aplicativo baseado no protótipo previamente projetado no Figma.
 - ◆ Escolha entre as funcionalidades levantadas, as possíveis de se implementar;
 - ◆ Design das telas baseado no protótipo feito em Figma;
 - ◆ Implementar as interações entre as telas e o usuário.
- Resultado: A aplicação pronta com o design das telas e a possibilidade de interação com o usuário.

3.1.4. Fase 4: Comunicação da aplicação com APIs externas

- Objetivo: Conectar o aplicativo com APIs externas para consumir seus dados.
 - ◆ Conectar com uma API para obtenção de notícias sobre a COVID-19;
 - ◆ Conectar com uma API para obter as taxas de casos de COVID-19 em diferentes países.

- Resultado: Por meio de requisições HTTP, é possível obter dados de APIs externas para uso da aplicação.

3.1.5. Fase 5: Criação do servidor

- Objetivo: Criação do servidor para recebimento dos dados do usuário por meio da triagem de sintomas.
- Resultados: Criado o servidor que recebe os dados e armazena em um banco de dados.

3.1.6. Fase 6: Comunicação entre o aplicativo e o servidor

- Objetivo: Projeto e implementação da comunicação entre o aplicativo e o servidor.
- Resultado: Comunicação usando métodos HTTP para troca de dados entre o aplicativo e o servidor.

3.1.7. Fase 7: Teste com os usuários e análise dos resultados

- Objetivos:
 - ◆ Disponibilizar o aplicativo para os usuários testarem;
 - ◆ Receber *feedback* de potenciais usuários por meio de um formulário.
- Resultado: Algumas das sugestões foram implementadas, criação da versão 1.1 do aplicativo. Os resultados foram detalhados na seção 6.

3.2. Técnicas

Essa seção contém as principais tecnologias utilizadas no desenvolvimento do projeto.

3.2.1. Banco de dados

Para o projeto, foi escolhido o banco de dados NoSQL (*Not Only SQL*), do qual Lóscio (2019) expôs que tem como propósito gerenciar um grande volume de dados (*Big Data*), estes que podem estar de forma estruturada ou semiestruturada, e que exige alta escalabilidade e disponibilidade.

A escolha de um banco de dados NoSQL foi devido ao tipo de dados que esse projeto exige. O objetivo deste banco de dados é armazenar os sintomas dos usuários, que são reunidos por meio de um formulário, estes sintomas estão em constante mudança junto com a pesquisa científica, e em conjunto a sua estrutura de dados. E também, porque este projeto foi pensado para ser dimensionado, e assim, podendo atender a um grande número de usuários

Desta forma, os seguintes pontos levaram a escolha do banco de dados NoSQL:

- A estrutura do dado a ser armazenado;
- Rapidez para consulta e inserção;
- Simplicidade para criação e manutenção do banco de dados.

Atualmente no mercado, temos diversos bancos de dados não relacionais disponíveis, como: Cassandra, Hypertable, MongoDB, Redis, CouchDB e Dynamo. Além disso, possui-se diferentes tipos de modelos de dados NoSQL, alguns deles mais

conhecidos são: *key-value*, orientado a colunas, orientado a documentos e orientado a grafos.

Neste projeto optou-se pelo MongoDB do qual utiliza o modelo orientado a documentos, neste modelo conforme descrito por Lóscio (2019) se pode armazenar coleções de documentos, sendo um documento, um objeto com um identificador único e um conjunto de campos, que podem ser strings, listas, tendo assim como característica importante na escolha o fato de não ter um esquema rígido, como ocorre nos bancos relacionais. Desta forma pode-se atualizar a estrutura do documento, com novos campos, sem causar problemas no banco de dados.

Na Figura 2 é apresentada a estrutura de documento utilizada para armazenar os dados dos sintomas dos usuários do aplicativo.

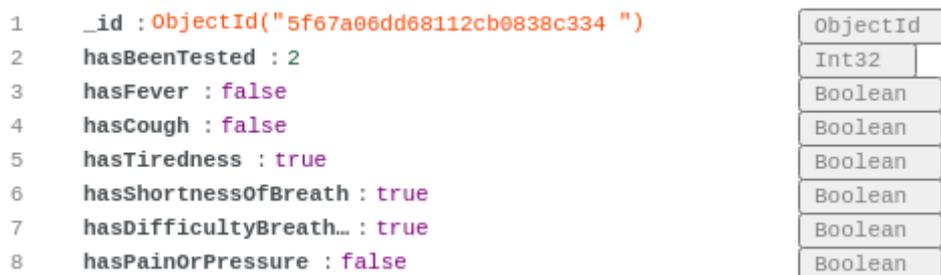


Figura 2. Estrutura do documento no banco de dados

3.2.2. Desenvolvimento mobile

No desenvolvimento *mobile* deste projeto escolheu-se pela utilização do *framework* Flutter, o qual se trata de um projeto *open source* criado pela empresa Google na linguagem Dart, que tem como objetivo o desenvolvimento multiplataforma, com foco na construção de interfaces, possibilitando assim a construção de um aplicativo de alta performance e com uma interface gráfica de qualidade.

Flutter possibilita com o desenvolvimento de um único código fonte ter disponível um aplicativo para uso nos sistemas Android e IOS, visto que os dois sistemas operacionais são os mais conhecidos e utilizados atualmente nos *smartphones*, sendo este um ponto positivo se comparado a ter que escrever o código nativo para Android e IOS (BASSETO, 2019), tendo que ter conhecimento sobre as duas plataformas para desenvolver em cada uma, aumentando o tempo gasto com desenvolvimento, e dificultando atualizações do aplicativo, pois teria que alterar em dois códigos fontes diferentes.

O Flutter tem como conceito a componentização, onde cada componente seria um Widget, de forma que facilita a reutilização dos mesmos componentes em diferentes telas, criando uma padronização no *layout* das telas e ainda reduzindo o tempo de trabalho do desenvolvedor, e outro ponto positivo também é o fato de possuir componentes prontos e bonitos, e caso necessário também é fácil realizar uma personalização dos componentes, sem o desenvolvedor precisar gastar muito tempo no desenvolvimento de interfaces bonitas. Exemplos de aplicativos que passaram a utilizar Flutter, conforme descrito no próprio site do Flutter (2020) são: Nubank, Philips Hue, Ebay, Groupon, entre outros.

Além disso algumas outras vantagens do Flutter (2020) mencionadas são:

- Facilidade na utilização de recursos nativos, de forma que basta apenas instalar a

biblioteca, salvar e estará pronto para utilizar;

- Documentação detalhada e eficiente;
- Possibilidade de usar a mesma linguagem sendo tipada ou não, a critério do desenvolvedor;
- *Hot Reload*, isso reflete em uma alta produtividade, as alterações no código são refletidas no celular ou emulador de forma quase instantânea.

Devido a todos os pontos positivos citados, à necessidade de um rapidez no desenvolvimento das funcionalidades e ainda visando a obtenção rápida de uma interface gráfica amigável utilizando do Material UI, o Flutter se mostrou uma ótima opção para desenvolvimento do aplicativo.

3.2.3. Desenvolvimento do backend

Para o desenvolvimento do *backend*, foi utilizado o NodeJS. Criado no final de 2009 por Ryan Dahl e mais catorze funcionários, trata-se de uma plataforma popular para construções de aplicações *server-side* em Javascript e que roda em cima da Javascript Engine V8 da Google (NODEJS, 2020), criada inicialmente para utilização no navegador Chrome.

Um diferencial do NodeJS em comparação a outras tecnologias para *web* no desenvolvimento como Python, Ruby e Java é o fato de possuir uma arquitetura *non-blocking thread*, de forma que visa uma alta performance e baixo consumo da memória, com o objetivo de utilizar completamente e de forma eficaz a capacidade de processamento dos servidores, com isso o desenvolvedor aguarda menos tempo por um resultado de uma solicitação (PEREIRA, 2014).

De acordo com Pereira (2014), o NodeJS torna-se uma plataforma altamente escalável, e possui características como ser *single-thread* e orientado a eventos, e por meio do NPM (Node Package Manager) é possibilitado ao desenvolvedor integrar um imenso conjunto de módulos e pacotes externos elaborados pela comunidade de desenvolvedores que disponibilizam estes de forma pública, assim qualquer desenvolvedor pode distribuir ou reutilizar código de terceiros de forma prática.

4. Desenvolvimento

Após uma análise do estado da arte, três funcionalidades foram escolhidas sendo elas: histórico de casos, triagem de sintomas e canal de notícias, para a primeira versão do aplicativo, especialmente por questões técnicas, mas também devido à duração do projeto.

4.1. Página inicial

A página inicial do aplicativo, que pode ser visualizada na Figura 3, possui três cartões que conduzem os usuários às funcionalidades implementadas. Cada cartão possui um título e uma explicação sobre a respectiva funcionalidade.

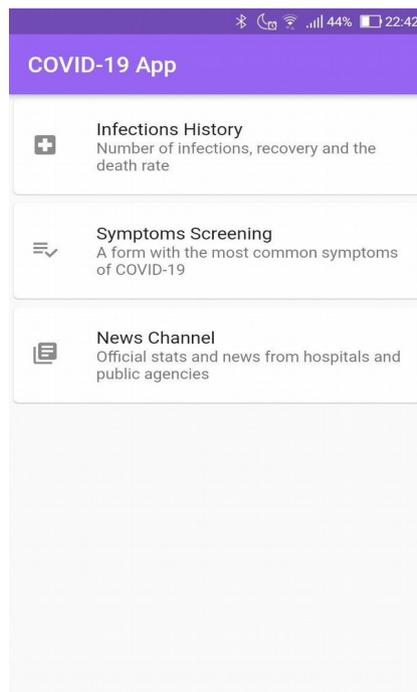


Figura 3. Tela inicial do aplicativo

4.2. Histórico de casos

Na tela de histórico de infecções é mostrado o número total de casos confirmados, recuperados e mortos no mundo, conforme mostrado na Figura 4. Todas as informações desta tela, bem como as subsequentes quando o usuário filtrar por país, são coletadas da base de dados da John Hopkins.

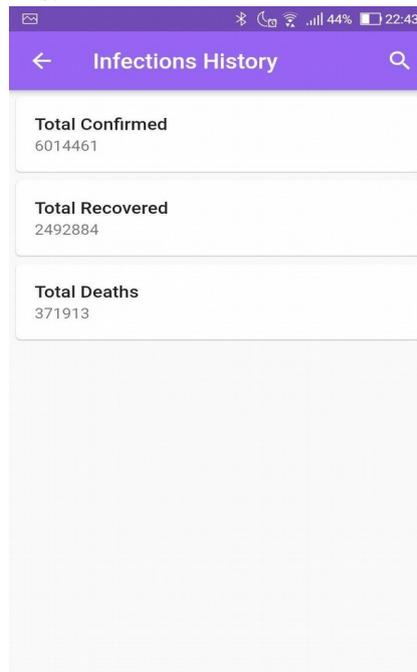


Figura 4. Tela com histórico de casos

4.3. Triagem de sintomas

Quando o usuário clica na opção de triagem de sintomas, primeiro aparece uma caixa de diálogo questionando se o usuário fez o teste para COVID-19 e o resultado. Este diálogo pode ser visto na Figura 5.

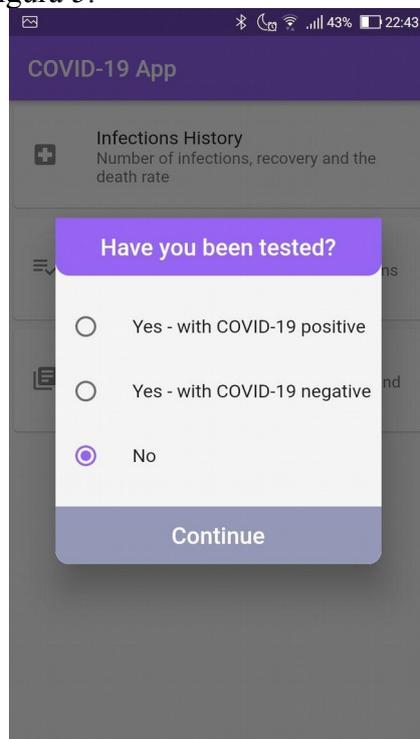


Figura 5. Caixa de diálogo da triagem de sintomas

Caso na caixa de diálogo da Figura 5, o usuário responder que não fez o teste ou obteve resultado positivo para COVID-19, uma tela de triagem de sintomas é mostrada, como pode ser visto na Figura 6.

21:09

← Symptoms Screening

Are you feeling these symptoms?
(select one or more options)

- Fever
- Cough
- Tiredness
- Shortness of breath
- Difficulty breathing
- Persistent pain or pressure in the chest

Send

Figura 6. Checklist de sintomas

E, por fim, quando o usuário envia os sintomas para o servidor, uma caixa de diálogo aparece se for bem-sucedido o envio.

4.4. Canal de notícias

A página principal do canal de notícias mostra notícias do Brasil, como pode ser visto na Figura 7, com a possibilidade de poder filtrar por país.



Figura 7. Tela principal com algumas notícias

5. Trabalhos correlatos

Devido ao enorme repercussão que o COVID-19 está causando ao redor do mundo, muitos pesquisadores já estão trabalhando em soluções utilizando aplicativos móveis para reduzir os danos causados. Algumas dessas soluções podem ser encontradas nesta seção.

5.1. Coronavírus-SUS

O governo brasileiro disponibilizou o aplicativo móvel Coronavírus-SUS para realização de triagem de pacientes, verificando se existem sintomas apresentados pelo usuário compatíveis com os apresentados pelo COVID-19. Esse aplicativo contém diversos tópicos como sintomas, como prevenir a doença, o que fazer em caso de suspeita ou infecção, e também um campo com notícias oficiais do Ministério da Saúde sobre o assunto.

5.2. Private Kit: Safe Paths

O aplicativo Private Kit: Safe Paths, criado nos Estados Unidos, é um aplicativo de código aberto, desenvolvido pelo MIT em colaboração com diversos pesquisadores. Neste aplicativo, o usuário recebe uma notificação em seu celular caso esteja próximo a uma pessoa infectada pelo COVID-19. Esse reconhecimento acontece pela tecnologia *bluetooth* e afirma ser totalmente anônimo para ambos os usuários - o infectado que envia o sinal e as pessoas ao redor podem receber este sinal.

6. Resultados

Após finalizar a primeira versão do aplicativo, ele foi encaminhado para testes para um grupo de usuários, entre eles dois *designers* de UX/UI (Experiência do Usuário/Interface do Usuário) e um desenvolvedor/analista de sistemas. Em seguida,

algumas de suas sugestões foram implementadas, e a versão 1.1 foi lançada para novos testes, desta vez para mais de trinta usuários.

Em ambos os conjuntos de testes, a principal preocupação era com a usabilidade. Conforme explanado por Marcus (2015), usabilidade refere-se aos seguintes itens:

- Design intuitivo;
- Fácil de aprender;
- Eficiência de uso;
- Memorabilidade;
- Frequência ou gravidade do erro;
- Satisfação subjetiva.

Como a usabilidade é altamente importante no desenvolvimento de *software*, entre os usuários neste primeiro teste, havia dois *designers* de UX/UI.

No primeiro teste efetuado com usuários, obteve-se como retorno de um dos *designers* que o aplicativo abordava um conteúdo relevante, porém poderiam ter maiores *feedbacks* de retorno da aplicação. Um exemplo poderia ser a barra de pesquisa do aplicativo a qual deveria declarar de forma mais clara que é para pesquisar por país e quando não tivesse um país encontrado, o mesmo deveria informar também. Essa sugestão foi relevante, sendo efetuado seu desenvolvimento na versão 1.1 do aplicativo.

Foi elaborado um questionário *online* dividido em 3 grandes tópicos: as funcionalidades, o *design* e eventuais erros ou sugestões em geral. Este questionário foi encaminhado para o grupo de três usuários. As questões e seus resultados podem ser visualizados a seguir.

Em termos de funcionalidade, as questões podem ser visualizadas na Tabela 1. A opção Bom significa “bom - com alguns aspectos a serem melhorados” e a opção Ótimo, neste caso, significa “Eu posso entender do que se trata”.

Tabela 1. Resultado da pesquisa sobre funcionalidades

Questão	Ruim	Bom	Ótimo
O que achou da página inicial do aplicativo?		66,7%	33,3%
O que você achou da tela de histórico de infecções?		100%	
O que você achou da tela de triagem de sintomas?		100%	
O que você achou da tela do canal de notícias?		66,7%	33,3%
Em geral, o que você achou das funcionalidades do aplicativo?		66,7%	33,3%

Após essa pesquisa inicial com os três usuários, e elaborada a versão 1.1 do aplicativo, foi realizada uma segunda pesquisa para medir a usabilidade com foco em quatro pontos: utilidade, facilidade de uso, facilidade de aprendizado e satisfação. A

pesquisa foi aplicada por meio de um formulário, no qual 32 usuários da versão 1.1 do aplicativo responderam questões que variam em uma escala entre 1 (discordo totalmente) e 7 (concordo totalmente), e também uma opção N/A (não se aplica).

Além do questionário, também houve um espaço para apontar aspectos negativos e positivos do aplicativo, dos quais podem ser visualizados a seguir.

Aspectos negativos listados pelos usuários:

- “Demorou algum tempo para recuperar as informações.”
- “Gostaria de ver gráficos comparando os dados.”
- “A pesquisa do país é armazenada. Se eu buscar notícias de 10 países diferentes e tentar voltar, preciso clicar em desfazer 10 vezes para voltar a página inicial. E também gostaria de ver as curvas de infecção.”
- “Eu gostaria de ver o *feedback* dos meus sintomas, se eu tiver curiosidade ou não.”

Aspectos positivos listados pelos usuários:

- “É simples e direto. Eu confio nas informações fornecidas pelo aplicativo. Na utilização o fluxo é linear e conciso.”
- "Fácil de usar. Minimalista.”
- “Fácil de usar e buscar informações. Interessante buscar notícias de países diferentes. Muito bom.”
- “Fácil, intuitivo e com informações úteis.”
- “É bom buscar notícias de diferentes países.”
- “Fácil, limpo, rápido e intuitivo.”
- “Design limpo”.
- "Muito útil.”

7. Considerações finais

Os principais objetivos deste projeto eram conceber e implementar uma aplicação móvel, onde os usuários pudessem obter dados confiáveis e atualizados sobre a COVID-19 e também para compartilhar seu estado de saúde em um banco de dados *online* para uso futuro. Este projeto cumpriu seus principais objetivos e também abriu portas para futuras implementações e melhorias.

Por mais que se esteja na Era da Informação, e teoricamente, não é muito difícil para as pessoas obterem informações em 2020, o aplicativo projetado e desenvolvido neste projeto é útil não só porque visa concentrar informações sobre a pandemia da COVID-19, mas especialmente por causa da divulgação de notícias falsas e a falta de transparência em alguns governos.

Em 6 de junho de 2020, o Ministério da Saúde do Brasil decidiu não informar o número total de casos ativos e óbitos no país (MACHADO *et al.*, 2020). Alguns dias depois, por decisão do Supremo Tribunal Federal, os dados foram mais uma vez divulgados, visto a importância do acesso à informação por parte da população, pois somente com base em dados é possível ter uma real noção da situação.

Como *feedback* dos usuários teve-se grande aprovação do aplicativo, do qual a maior parte dos usuários comentou sobre a usabilidade do aplicativo e interface do mesmo. E também pelo assunto COVID-19 tratado neste projeto, ser um tema de importância global, este projeto visou ser construído uma arquitetura que poderia facilmente escalar e, graças a isso há muitas melhorias com o avanço dos estudos podem ser implementadas.

8. Referências

- BASSETO, Giovanni. **Comparativo entre Frameworks para desenvolvimento multiplataforma**. 47p. Trabalho de conclusão de curso (Graduação em Sistemas de Informação) - Universidade Estadual do Norte do Paraná, Campus Luiz Meneghel, 2019. Disponível em: <http://200.201.11.152/bitstream/handle/123456789/392/Projeto%20TCC%20-%20Giovanni%20vers%c3%a3o%20final.pdf?sequence=1&isAllowed=y>. Acesso em: 7 jun. 2020.
- BRASIL. Ministério da Saúde. **Covid-19 no Brasil**. Brasília, 2020. Disponível em: https://susanalitico.saude.gov.br/extensions/covid-19_html/covid-19_html.html. Acesso em: 26 jun. 2020.
- FLUTTER. **Apps take flight with Flutter**. [S.l.], [2020]. Disponível em: <https://flutter.dev/showcase>. Acesso em: 7 jun. 2020.
- GATES, Bill. **The next outbreak?: We're not ready**. [S.l.], 2015. Disponível em: https://www.ted.com/talks/bill_gates_the_next_disaster_we_re_not_ready. Acesso em: 29 mar. 2020.
- LÓSCIO, B. F. **NoSQL no desenvolvimento de aplicações Web colaborativas**. [S.l.], 2016. Disponível em: https://www.researchgate.net/profile/Bernadette_Loscio/publication/268201466_NoSQL_no_desenvolvimento_de_aplicacoes_Web_colaborativas/links/576aa72008aef2a864d1ef8c/NoSQL-no-desenvolvimento-de-aplicacoes-Web-colaborativas.pdf. Acesso em: 7 jun. 2020.
- MACHADO, Renato *et al.* Governo deixa de informar total de mortes e casos de Covid-19; Bolsonaro diz que é melhor para o Brasil. **Folha de São Paulo**, Brasília, jun. 2020. Notícias. Disponível em: <https://www1.folha.uol.com.br/equilibrioesaude/2020/06/governo-deixa-de-informar-total-de-mortes-e-casos-de-covid-19-bolsonaro-diz-que-e-melhor-para-o-brasil.shtml>. Acesso em: 10 jun. 2020.
- MARCUS, Aron. **Design, User Experience, and Usability**. 1st edition. Los Angeles: Springer, 2015.
- NODEJS. **ECMAScript 2015 (ES6) and beyond**. [S.l.], [2020]. Disponível em: <https://nodejs.org/en/docs/es6/>. Acesso em: 7 jun. 2020.
- PEREIRA, Caio Ribeiro. **Aplicações Web Real Time com Node.js**. São Paulo: Casa do Código, 2014.