DESENVOLVIMENTO DE APLICATIVO GAMIFICADO PARA MEDIÇÃO DE ESTRESSE EM PROFISSIONAIS DA ÁREA DE ENFERMAGEM

Anderson Freitas dos Santos[1], Davi Santoro Stocco[1], Débora Pelicano Diniz[1]

[1]Faculdade de Tecnologia de FATEC Ribeirão Preto (FATEC)
Ribeirão Preto, SP – Brasil

anderson.santos175@fatec.sp.gov.br, davi.stocco@fatec.sp.gov.br,
debora.diniz2@fatec.sp.gov.br

Resumo. *Este artigo descreve o processo de construção de um aplicativo gamificado para auxiliar profissionais da área da saúde a monitorarem seus níveis de estresse, utilizando o índice de Burnout Syndrome. O trabalho foi desenvolvido atendendo uma solicitação de um mestrando do curso de Enfermagem da USP – Ribeirão Preto, que auxiliou no entendimento dos conceitos relacionados à Síndrome de Burnout. O aplicativo foi construído seguindo as melhores práticas da Engenharia de Software e se encontra, atualmente, na fase de testes beta.*

Abstract. *This article describes the process of building a gamified application to help health care professionals to monitor their stress levels using the Maslach Burnout Inventory. The work was developed attending the solicitation of a master's student in nursing at USP – Ribeirão Preto, which assisted on the understanding of the concepts related to the Burnout Syndrome. The app was built following the best practices of the Software Engineering and it is currently on the beta-test stage.*

1. Introdução

É um fato que o estresse em local de trabalho é algo corriqueiro nas vidas dos profissionais atualmente, e existem diversos fatores que podem causá-lo, tais como hiperatividade, exaustão, atividade reduzida, reações emocionais, entre outros. Todos esses sintomas podem desencadear o que é chamado de Síndrome de Burnout (SBO), como apontam Weber e Jaekel-Reinhard (2000).

Com a pandemia que acometeu a sociedade em respeito ao SarS-Cov2 é notório que os profissionais de saúde ficaram extremamente sobrecarregados. Esses profissionais tendem a desenvolver diversos sintomas de estresse, por conta de suas horas de trabalho.

A avaliação da síndrome de Burnout é feita por um psicólogo, que por meio de perguntas pré-definidas, pode encontrar o nível de estresse do paciente e escolher uma estratégia mais assertiva para o tratamento. Um aplicativo gamificado poderia ser utilizado com a finalidade de auxiliar no diagnóstico, incentivando o profissional a buscar ajuda médica e ajudando a conscientizar sobre a existência da doença.

Assim o trabalho apresentado neste artigo teve como objetivo o desenvolvimento de um aplicativo gamificado na categoria *mHealth*, que poderá auxiliar profissionais na área de saúde a monitorarem seus níveis de estresse dentro do ambiente de trabalho, por meio de um questionário estratégico que auxilia na medição do desgaste do profissional.

O aplicativo também possui conteúdo sobre desgaste profissional, o que deve auxiliar os usuários a entender melhor a doença e seus sintomas. Ao finalizar os questionários, o usuário ganha um prêmio em moedas virtuais (na maior parte dos níveis) o que pode ser utilizado para desbloquear mais níveis.

2. Síndrome de Burnout

No início da década de 70 foi observada a Síndrome de Burnout (SBO), principalmente nos profissionais da área de saúde.

> SBO foi descrita como uma inabilidade de lidar com o estresse emocional no trabalho ou como um uso excessivo de energia e recursos, levando aos sentimentos de falha e exaustão. Apesar de depressão afetar quase todos os aspectos da vida de uma pessoa, os sintomas de Burnout ocorre apenas no trabalho; no entanto, SBO também diminui o bem-estar geral. Os sintomas clínicos de SBO são inespecíficos e incluem cansaço, dores de cabeça, problemas de alimentação, insônia, irritabilidade, instabilidade emocional e rigidez em relacionamento com outras pessoas (PONCET et al, 2006, p. 01, tradução livre).

Para se medir a SOB utiliza-se o MBI (Ma*slach Burnout Inventory*) que compreende 3 subescalas: exaustão emocional (composta por 9 itens), despersonalização (composta por 5 itens) e realização pessoal (composta por 8 itens). Cada item é medido em uma escala de 0 a 6, apresentada na Tabela 1. "Altos níveis de pontos de exaustão emocional e despersonalização e baixos pontos de realização pessoal resultam em um alto nível de SBO." (PONCET et al, 2006, p. 01).

Tabela 1 Escala de estresse percebido

Nível	**Descrição**
0	Nunca ocorre
1	Algumas vezes no ano ou menos
2	Uma vez por mês ou menos
3	Algumas vezes por mês
4	Uma vez na semana
5	Algumas vezes na semana
6	Todo dia

Fonte: (MASLACH; JACKSON, 1981)

3. Aplicativo gamificado

Conforme o crescimento de aplicações mobile na última década, o termo **Gamificação** tem ganhado cada vez mais força. Apesar de ser derivada da palavra inglesa *game*, a gamificação não tem ligação direta com jogos, mas ela abstrai algumas características que estão presentes neles, tal como a principal delas, que é divertir o público.

A área de *research* é um dos principais segmentos em que a gamificação é aplicada, pelo fato de que responder uma pesquisa pode se tornar uma tarefa repetitiva e maçante. Esse problema pode ser resolvido aplicando estratégias como sistemas de pontuação, *ranking* e recompensas, as quais ativam os instintos de competitividade e prazer do ser humano, o que gera motivação, engajamento, produtividade e lucidez para a execução de determinada atividade. O aplicativo em pauta abrange os segmentos de

saúde e *research*, porém a mecânica de gamificação também pode ser utilizada em inúmeros outros segmentos, tais como educação e treinamentos.

Jogos estão cada vez mais populares e isso faz com que um aplicativo que utilize propriedades deles, seja mais atrativo ao público.

> Motivação pode ser definida como o porquê de determinado comportamento de uma pessoa, sendo extrínseca, quando a realização de uma atividade é feita em troca de alguma recompensa. A motivação intrínseca ocorre na realização de uma atividade por vontade própria, e não por troca de recompensa. Estudos mostram que a utilização de elementos de jogos em um ambiente sério pode tornar suas atividades mais motivadoras, conceito chamado de gamificação. Esta abordagem em ambientes de aprendizagem está sendo foco de muitos artigos científicos para tentar provar a sua eficiência quanto a motivação gerada na aprendizagem. Ressalta-se ainda, dependendo de sua aplicação pode ser gerada motivação intrínseca nos usuários do ambiente (POFFO e AGOSTINI, 2018, p. 1).

4. Desenvolvimento do Jogo

A aplicação foi dividida em duas partes: (i) o *backend*, que não é visível ao usuário, sendo responsável pelas funcionalidades principais do app, como persistência e recuperação de dados por exemplo; e (ii) o *frontend*, que é a aplicação móvel que o usuário vê e interage - ao clicar em um botão, por exemplo, o *frontend* pode se comunicar com o *backend*.

O desenvolvimento do aplicativo seguiu as etapas da Engenharia de Software, sendo gerados artefatos intermediários. Todos os documentos gerados estão armazenados no GitHub e podem ser consultados por meio dos links:

- https://github.com/andersonfds/burnout-app
- https://github.com/andersonfds/burnout-api

4.1. Elicitação de Requisitos de sistema

O desenvolvimento do aplicativo foi uma solicitação de um mestrando em Enfermagem da USP – Ribeirão Preto que já possuía um protótipo do sistema e precisava acabar seu desenvolvimento. Assim, na etapa de elicitação de requisitos foram feitas várias reuniões com o cliente para melhor entender sua necessidade. O cliente também apresentou um protótipo do sistema que auxiliou muito na definição dos requisitos funcionais, que foram descritos no Documento de Requisitos (disponível no GitHub) e estão alguns estão apresentados na Figura 2.

A partir dos requisitos funcionais definidos, e considerando a principal característica solicitada, que é ter um aplicativo gamificado, ficou definido que o sistema seja composto por *cards* apresentados na tela inicial, com cada *card* contendo um questionário. Cada questionário possuí um preço em moedas virtuais para ser desbloqueado. Há níveis bloqueados e, nestes casos, uma taxa para desbloqueio é aplicada. O aplicativo deve liberar níveis e moedas gratuitos para o usuário conforme ele completa níveis.

Ficou definido também que o app seria desenvolvido utilizando o kit de desenvolvimento Flutter (2021), permitindo coleta de dados de forma remota, necessitando da construção de uma *API* que deveria conter uma forma de autenticação e forma de cadastro; as requisições para a API precisariam ser autenticadas, exceto a de login e cadastro.

RF 1. O aplicativo deve ser tratado como um jogo, para que os usuários fiquem mais relaxados ao utilizarem;

RF 2. O aplicativo deve permitir 3 tipos de questões: (i) questões fechadas com múltipla escolha, que permite que o usuário selecione mais de uma opção; (ii) questões fechadas com escolha binária, na qual o usuário poderá responder sim ou não, ou mesmo ter apenas uma ação de continuar; (iii) nível de *Burnout*, em que o usuário poderá escolher o valor de *burnout*;

RF 3. A ação sim ou não ou ação de continuar devem ser dinâmicas, podendo cada botão ser configurado para fazer uma ação específica de voltar para a tela inicial, ou prosseguir com o questionário.

Figura 1. Requisitos de sistema
Fonte: Autoria própria

Após o cliente ter validado o Documento de Requisitos, foram construídos diagramas para representar a modelagem funcional do sistema e todos esses documentos estão disponíveis no GitHub.

4.2. Escolha das tecnologias

Para desenvolvimento do *frontend*, foi utilizado o Flutter (2021), que é um kit de ferramentas desenvolvido pela Google, que permite que o aplicativo seja executado tanto em iOS, quanto em Android com apenas um código base; essa decisão foi tomada, pois os dois sistemas operacionais requerem uma sintaxe específica para que as coisas funcionem corretamente, no Android, por exemplo, isso pode ser feito com a linguagem de programação Java ou Kotlin; já no iOS pode ser feito apenas com Objective-C ou Swift; essas limitações, são designadas pelas manufaturadoras dos sistemas operacionais, no caso do Android, a Google e no iOS, a Apple; dessa forma para desenvolver uma aplicação com disponibilidade nos dois sistemas operacionais, caso não fosse utilizado algum toolkit multiplataforma, seria necessário fazer duas vezes o app: uma para Android, utilizando Java ou Kotlin e uma para iOS, utilizando Objective-C ou Swift.

4.3. *Backend*

Para o *backend* foi utilizado o NestJS, um framework para NodeJS que permite que uma aplicação web seja executada sobre o protocolo *HTTP*. Apesar de suportar outras tecnologias, como comunicação bilateral por *websocket*, o NestJS vai além de um micro-framework como o Express, pois ele é modularizado, permitindo que toda a aplicação possua partes independentes que podem ser reaproveitadas em outros módulos, sem a necessidade de reescrever código (NESTJS, 2021).

O SGBD (Sistema de Gerenciamento de Banco de Dados) utilizado foi o PostgreSQL (2021). A estrutura inicial do banco de dados foi criada por meio de entidades do TypeORM (2021), como podem ser vistas na Figura 3.

A implantação da API e do banco de dados foram feitas com o Heroku (2021), isso pois o Heroku possui planos acessíveis, e pode, inclusive, ser utilizado gratuitamente caso não haja uma demanda alta de recursos.

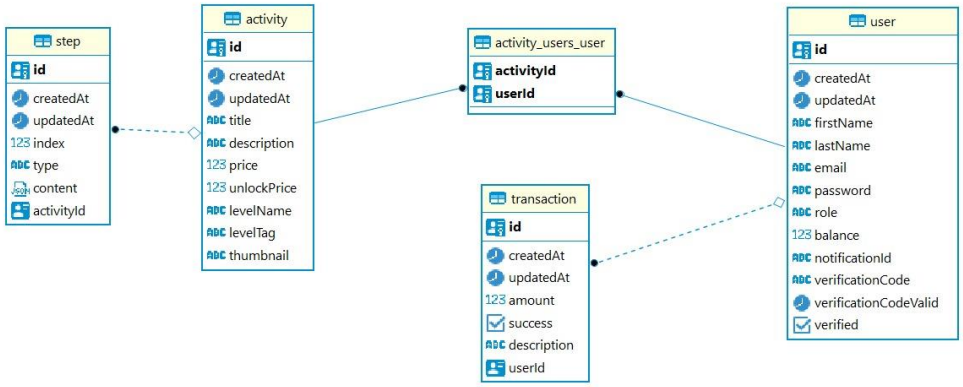

Figura 2. Modelo relacional do banco de dados

Fonte: Autoria própria

A escolha pela utilização do TypeScript se deve ao fato da linguagem permitir seguir de maneira consistente os princípios da programação orientada a objetos, além de tornar a codificação mais rápida e conveniente.

A organização dos domínios da aplicação é feita dentro do diretório ***resources***, como pode ser observado na Figura 4.

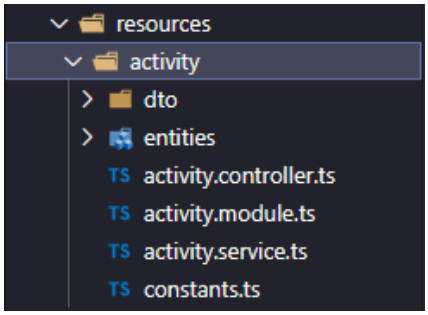

Figura 3. Estrutura de pastas do recurso Activity
Fonte: Autoria própria

Figura 4. Documentação do *endpoint* POST de *Activity*
Fonte: Autoria própria

Foi utilizada a ferramenta Swagger para gerar a documentação automática dos *endpoints* da A (Figura 5), que são agrupados a partir dos domínios da aplicação (Figura 6).

Figura 5. Agrupamento dos recursos
Fonte: Autoria própria

Cada *endpoint* é uma interface de comunicação que será utilizada para executar as funcionalidades da aplicação via protocolo HTTP.

Neste caso, o *client* que irá realizar chamadas e interagir com cada *resource* da aplicação será o aplicativo mobile, com chamadas para conseguir acesso ao sistema (***Auth***) e obter as atividades que serão realizadas pelo usuário. A partir do *resource* de etapas (***Step***), é possível inserir e buscar no banco de dados os passos que deverão ser executados para a conclusão de cada atividade da aplicação (***Activity***).

4.4. Processo de implantação do *Backend* na nuvem

Para implantar a aplicação no *Heroku*, foi necessário criar uma aplicação dentro do dashboard da plataforma, essa etapa é necessária para obter um URL único na internet para essa aplicação, em seguida foi necessário conectar ao repositório do GitHub.

Ao conectar ao repositório, o *Heroku* é capaz de se comunicar diretamente e assim fazer a implantação a partir de uma *ramificação*, sendo necessário apenas configurar as *variáveis de ambiente*.

4.5. *Frontend*

Para desenvolvimento do *frontend*, foi utilizado o Flutter (2021) pois assim foi desenvolvido apenas um código que pode ser executado tanto em equipamentos com sistema operacional Android quanto iOS.

A primeira tela do aplicativo é a tela de **Login** (Figura 7a). Caso o usuário ainda não seja cadastrado, ele deve clicar no link **'Ainda não tem cadastro? Cadastre-se'** que o direciona para tela de **Cadastro** (Figura 7b). Ao fazer o cadastro, receberá um e-mail com um código de 4 dígitos para que possa confirmar a autenticidade de seu e-mail (Figura 7c). Caso o usuário já seja cadastrado, basta inserir seu e-mail e senha para ir para a tela **Principal**.

Na página principal o usuário poderá responder aos questionários dinâmicos (cujo conteúdo é carregado dinamicamente da API). Existem alguns tipos de telas:

- A tela com questões do tipo **Burnout** (Figura 8a), que contém uma pergunta e uma linha horizontal que o usuário poderá deslizar sobre e escolher uma opção, entre a extrema esquerda, que é equivalente ao valor 0 na escala MBI, e a extrema direita, equivalente ao valor 6.
- A tela com questões do tipo **Opções** (Figura 8b), que contém uma lista de possibilidades que o usuário poderá escolher. Existem casos em que é possível

escolhas múltiplas e existem casos que apenas um valor poderá ser escolhido; ainda, há casos em que a lista pode apenas ser vista e não modificada pelo usuário.

- A tela do tipo **texto**. O tipo texto é um pouco mais complexo, pois ele possui a ação de continuar, no entanto pode também aparecer opções sim e não, e cada uma das opções podem ter uma ação específica; nas Figuras 8c e 8d estão exemplos dos 2 cenários para a atividade texto.
- A tela do tipo **imagem,** que mostra uma imagem em modo de tela cheia.

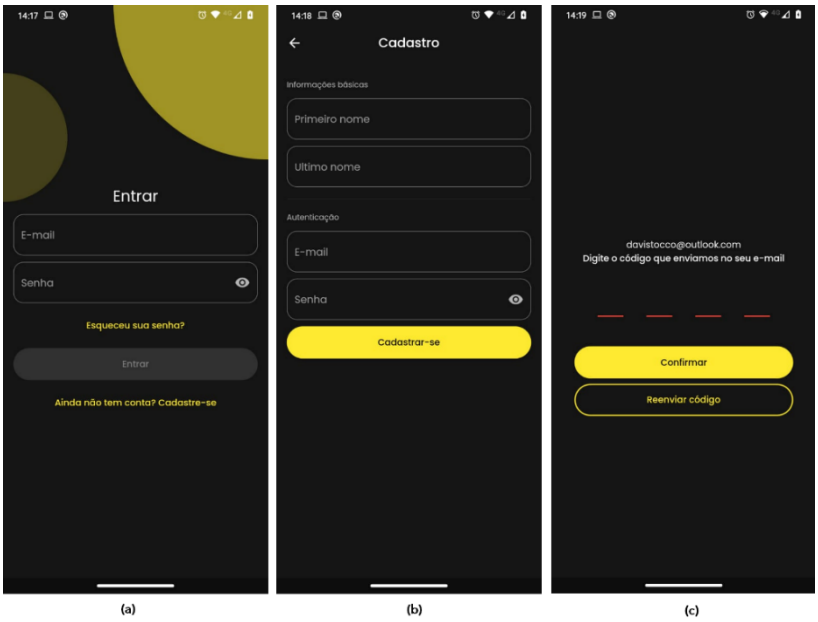

Figura 6 a) Tela de Login; b) Tela de Cadastro; c) Tela de Confirmação de Cadastro
Fonte: Autoria própria

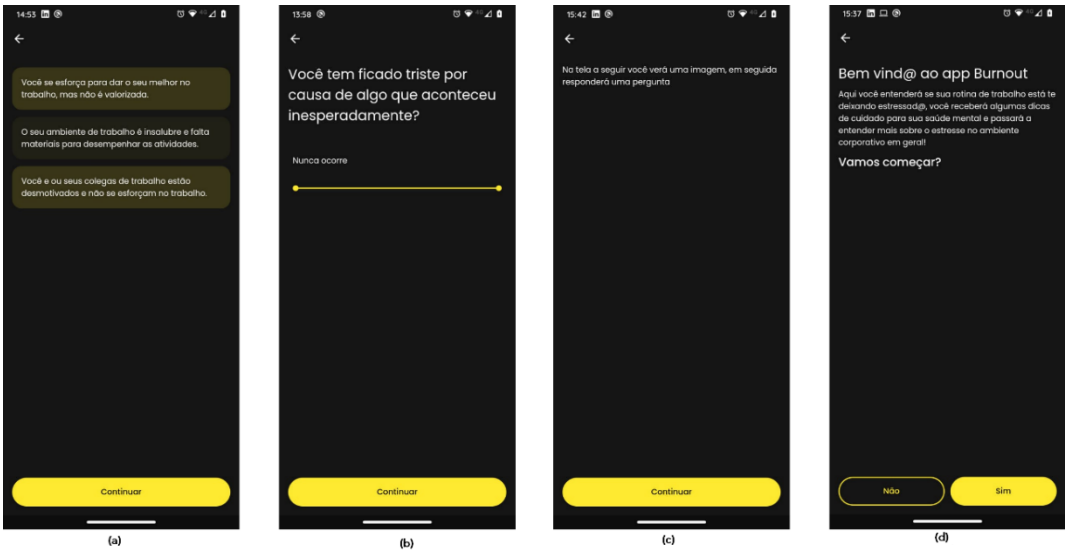

Figura 7: Atividades de opções, Burnout e texto
Fonte: Autoria própria

Ao finalizar as tarefas, o usuário poderá ganhar um prêmio em moedas virtuais, que servirão para adquirir novos níveis e completar mais questionários. Uma outra possibilidade é o nível de Burnout do usuário ser recalculado baseando-se nas suas respostas. Na Figura 9 estão apresentados exemplos das duas possibilidades.

4.6. Arquitetura da aplicação mobile

Cada tela da aplicação foi dividida em 2 componentes principais, sendo eles, o componente que irá controlar a página, que pode ser chamado de **controlador**, e a página, que por sua vez contém instruções de como dispor os elementos na tela para o usuário.

O gerenciamento de estado foi feito utilizando o GetX. Na Figura 10 está apresentada a estrutura de pastas da tela de Login, que foi replicada para todas as outras páginas, de acordo com a necessidade de cada uma.

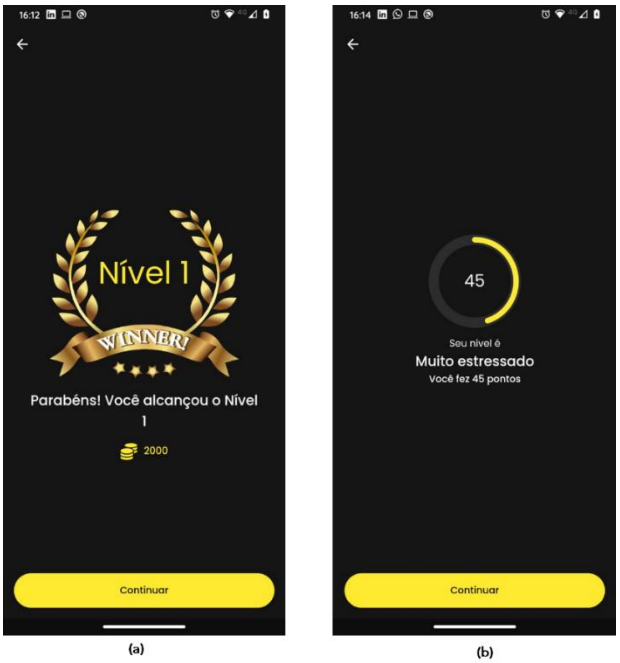

Figura 8: Possibilidades de finalização
Fonte: Autoria própria

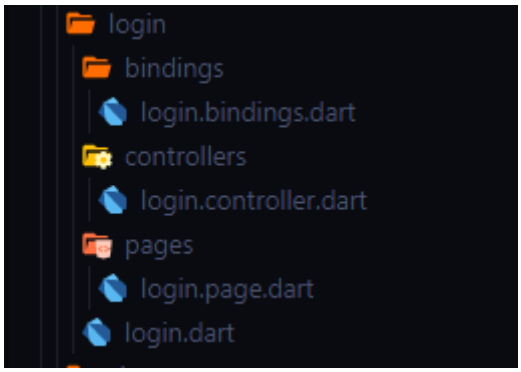

Figura 9: Estrutura de pastas da tela de login
Fonte: Autoria própria

Conclusão

Deve-se dar mais atenção quando o assunto é obre o excesso de trabalho, e com a pandemia, o esgotamento físico e mental é ainda mais evidente em profissionais que precisam não só administrar o trabalho, como também conduzir as tarefas de casa e cuidar dos filhos em tempo integral.

Devido ao aumento da carga horária dos profissionais da área de saúde no geral, por conta da COVID-19, esse aplicativo se torna útil para monitoramento e coleta de dados da saúde mental desses profissionais.

Observa-se que o objetivo proposto foi atingido, visto que o aplicativo está funcionando, já sendo usado, em testes beta, por alguns profissionais da saúde, que retornarão suas observações sobre o aplicativo.

Assim, como trabalho futuro será realizada a análise das respostas obtidas nos testes beta, corrigindo eventuais problemas apresentados e, após essa atividade, o aplicativo será utilizado pelo cliente em seu trabalho de doutorado.

Referências

FLUTTER. (2021). Disponível em: <https://flutter.dev/>. Acesso em: 17 mai. 2021.

HEROKU. (2021). Disponível em: <https://www.heroku.com/>. Acesso em: 18 mai. 2021.

MASLACH, C.; JACKSON, S. E. (1981). ***The measurement of experienced burnout.*** Journal of Organizational Behavior, Volume 2, Edição 2, p. 99-113. Disponível em:<https://onlinelibrary.wiley.com/doi/abs/10.1002/job.4030020205>. Acesso em: 20 mar. 2021.

NESTJS. (2021). Disponível em: <https://docs.nestjs.com/>. Acesso em: 18 mai. 2021.

PostgreSQL. (2021). Disponível em: <https://www.postgresql.org/>. Acesso em: 18 mai. 2021.

POFFO, M.; VOLANI, E. A. (2018). Gamificação para motivar o aprendizado. **CIET:EnPED**, São Carlos. Disponível em: <https://cietenped.ufscar.br/submissao/index.php/2018/article/view/268>. Acesso em: 20 mai. 2021.

PONCET, MARIE CÉCILE et al. (2006). ***Burnout Syndrome in Critical Care Nursing Staff*.** American Journal of Respiratory and Critical Care Medicine, Volume 175, Edição 7, p. 698-704. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/17110646/>. Acesso em 20 mar. 2021.

TypeORM. (2021). Disponível em: <https://typeorm.io/>. Acesso em: 18 mai. 2021.

WEBER A., JAEKEL-REINHARD A. (2000). ***Burnout Syndrome: A Disease of Modern Societies?*** *Occupational Medicine*, Volume 50, Edição 7, p. 512-517. Disponível em:<https://pubmed.ncbi.nlm.nih.gov/11198677/>. Acesso em 05 mai. 2021.