

HELPER TBL: um software para auxílio da aplicação da TBL em sala de aula

FERREIRA, Robert S.¹; CAMPOS, Saulo C.²;
TREVIZANO, Waldir Andrade²; DAIBERT, Marcelo Santos²



¹ Graduando em Ciência da Computação - UNIFAGOC

² Docente do curso de Ciência da Computação - UNIFAGOC

robertsilva-ferreira@hotmail.com

saulo@unifagoc.edu.br

waldir@unifagoc.edu.br

daiibert@unifagoc.edu.br

RESUMO

O presente trabalho consiste na elaboração de uma plataforma web da segunda fase da metodologia TBL, realizando a automação dos processos manuais, como o de criação dos quizzes e das folhas de resposta. O TBL é uma metodologia ativa com objetivo de tornar o estudante protagonista na sua aprendizagem. Nele são demonstrados o processo de desenvolvimento e o resultado final do software, apresentado por meio das telas da aplicação. Ao final, são analisados os resultados de uma avaliação de experiência de uso e praticidade realizada com alunos, na qual eles utilizaram o software em uma aplicação real da metodologia em sala de aula. Constatou-se uma boa experiência de uso e praticidade para o aluno, apontando um resultado positivo.

Palavras-chave: Metodologia TBL. Metodologia Ativa. Flutter. Dart.

INTRODUÇÃO

A partir da observação da linha temporal da educação brasileira, será possível perceber que houve pequenas mudanças na educação, como resultado de uma sequência de sucessos e tropeços (Carneiro, 2008). Para se obter uma compreensão da evolução e das conquistas ocorridas referentes ao ato formal de educar no Brasil, tende a ser necessário abordar a educação desde a chegada dos portugueses até a contemporaneidade (David *et al.*, 2014).

A educação foi iniciada e cuidada pelos jesuítas nos primeiros anos até acontecer a expulsão da Companhia de Jesus, sofrendo alterações somente em 1808, com a chegada da família real; essas alterações, voltadas para o nível superior, compreendiam a criação de escolas no Rio de Janeiro, como a Escola Anatômica e Cirúrgica, e a autorização de cursos de Matemática, Engenharia, entre outros (David *et al.*, 2014).

Somente a partir da década de 1960 iniciou-se a incrementação e estruturação do ensino (David *et al.*, 2014). Partindo de mobilizações sindicais e movimentos populares ocorreram as primeiras experiências de popularizações das escolas, mas com o início do Governo Militar passou por alterações através de leis e atitudes, censurando o desenvolvimento do raciocínio e a defesa de ideias (David *et al.*, 2014). Após o fim do período militar, na década de 1980, o sistema escolar se reorganizou; em 1996, é lançada a nova LDB, regendo o ensino até o ano de 2021 (David *et al.*, 2014), passando por alterações entre 2021 e 2022; assim, a Lei nº 14.407, de 12 de julho de 2022 (Brasil, 2022), tornou-se a mais atual até o ano de 2022 (Filpo, 2022).

Pode-se observar que foram propostas mudanças no âmbito de institucionalizar e expandir o direito à educação; contudo, com todas essas mudanças, houve poucas discussões referentes à maneira como os professores lecionavam dentro de sala de aula, no campo da didática de ensino.

Dentro desse cenário de estagnação, ocorreram casos de estudos para desenvolver a educação, e a metodologia ativa é umas das metodologias escolhidas para se estudar. As metodologias ativas possuem a principal característica de tornar o estudante protagonista na sua aprendizagem e tornar o professor um facilitador do processo, buscando conhecimento além do material repassado pelo professor, o qual deixa de ser um meio exclusivo do saber na sala de aula. Seguindo essa ideologia, foram desenvolvidas ao longo do tempo várias técnicas. Pode-se citar alguns exemplos como: sala de aula invertida, PBL (*Problem-Based learning* ou aprendizagem baseada em problemas), Gamificação e TBL (*Team-Based Learning* ou Aprendizado Baseado em Equipes). As pesquisas para aprofundar o conhecimento sobre as metodologias foram iniciadas a partir da década de 1990 por alguns autores como Bonwell, Eison, Barnes, os quais defendiam formas e estratégias novas para o ensino/aprendizagem, transformando a sala de aula em um ambiente mais adequado para adquirir os saberes (Beck, 2018).

Neste trabalho o foco será na metodologia *Team-Based Learning* (TBL) ou Aprendizado Baseado em Equipes (ABE), que foi desenvolvida na Universidade de Oklahoma para cursos de administração em 1970 por Larry K. Michaelsen, baseando-se no trabalho em equipe, raciocínio profundo e pensamento crítico, as quais, mesmo sendo projetadas para grandes turmas, podem ser aplicadas em turmas pequenas. A ideia inicial era a de criar oportunidades e obter os benefícios do trabalho em equipe, através de pequenos grupos de aprendizagem, que trabalham juntos dentro da mesma sala de aula (Bollela *et al.*, 2014).

Possuindo algumas particularidades, o TBL se diferencia de estratégias voltadas para o ensino em pequenos grupos, como a PBL (*Problem-based learning* ou aprendizagem baseada em problemas). Não é necessário o uso de várias salas especiais; é adaptável para pequenos e médios grupos de trabalho; e também não necessita de vários professores trabalhando juntos (Bollela *et al.*, 2014).

Considerando o contexto educacional, percebe-se que ocorreram poucas mudanças, tanto na didática de ensino, quanto no uso de métodos ativos de ensino, mas principalmente no âmbito da tecnologia, diferentemente do que ocorreu em várias outras áreas. O TBL demonstra benefícios como aulas mais dinâmicas, participativas, produtivas entre outros, desta maneira a proposta deste trabalho é o desenvolvimento de um software que possa automatizar e facilitar a aplicação da TBL em sala de aula (Loureiro, 2020).

Em resumo, este artigo tem como objetivo geral desenvolver uma plataforma web para automatização do método de aplicação da metodologia ativa TBL, trazendo alguns benefícios para o professor, como a remoção do trabalho de desenvolver os questionários impressos, a remoção da necessidade de preparar as folhas de respostas manualmente e facilitar correção, enquanto para o aluno facilita a revisão dos próprios erros, centraliza o material para estudo e revisão, material que o próprio questionário aplicado na metodologia se torna após o término da sua aplicação. Além disso, também se objetiva compreender o arcabouço teórico por trás da TBL; estudar as tecnologias

Dart, Flutter e GraphQL; levantar requisitos; desenvolver um protótipo da plataforma; e aplicar a ferramenta em uma atividade real de aula.

REFERENCIAL TEÓRICO

Metodologias Ativas: TBL (*Team-Based Learning*)

A *Team-Based Learning* (TBL) ou Aprendizado Baseado em Equipes (ABE) foi criada em 1970 por Larry Michaelsen, para uso acadêmico na Universidade de Oklahoma. Em 2001, com o objetivo de incorporar novas metodologias nos cursos de ciência da saúde, o governo americano escolheu o TBL para se aplicar e financiou educadores para aplicá-la em sala de aula (Bollela *et al.*, 2014). Hartz e Schalatter descrevem o objetivo desta estratégia de ensino como “ir além de simplesmente cobrir conteúdos, mas se concentrar em garantir que os alunos tenham a oportunidade de praticar usando os conceitos do curso para resolver problemas”.

As atividades dentro da TBL são divididas em 3 etapas. A primeira etapa é chamada de **Preparo** ou *Preparation*, na qual o aluno faz, externamente à sala de aula, um preparo inicial referente à tarefa (Krug *et al.*, 2016). Para executar esse estudo, o professor disponibiliza o material voltado ao tema, podendo ser, por exemplo, um livro, um artigo. Essa fase o objetivo de fazer com que o aluno obtenha um entendimento base sobre o tema (Hartz; Schalatter, 2016).

A segunda etapa, denominada de **Garantia de Preparo** ou *Readiness Assurance*, é constituída por dois testes. Enquanto o primeiro é realizado individualmente, o segundo é realizado em grupo. Nesta etapa, há um feedback sobre a resposta da questão indicando se a resposta é a correta; em sequência, o aluno pode fazer uma apelação referente ao teste (Krug *et al.*, 2016). Os dois testes são compostos por questões de múltipla escolha, mas ocorre uma diferença na maneira de responder a eles (Bollela *et al.*, 2014). Para responder ao teste individual, o aluno deve verificar as alternativas que considera corretas e distribuir os pontos totais entre elas. Por exemplo: em 1 questão de 4 alternativas, ele tem um total de 4 pontos; caso tenha dúvida entre as alternativas “b” e “d”, ele pode distribuir dois pontos em cada alternativa e no momento da correção irá contabilizar somente os pontos da alternativa correta. Já no teste em grupo, cada aluno irá receber uma folha de respostas para verificar a sua resposta; primeiramente, após um debate do time, eles devem selecionar uma alternativa em conjunto e verificar se é a correta. Caso tenha errado, ele terá uma nova chance, mas seus pontos irão cair de 1 em 1 ponto a cada alternativa que verificar, e, quando acertar, os pontos serão contabilizados de acordo com a quantidade que foi retirada (Bollela *et al.*, 2014). Após os dois testes, o aluno terá a opção de solicitar uma apelação, caso não concordem com a resposta indicada pelo professor como correta. Para validar a apelação é necessário argumentar utilizando referências para embasar o argumento (Bollela *et al.*, 2014).

Na última etapa, chamada de **Aplicação dos Conceitos** ou *Application of Course*, são propostas atividades em equipe voltadas ao cotidiano profissional (Krug *et al.*, 2016). Essa etapa é desenvolvida seguindo 4 princípios básicos, conhecidos no inglês como 4's: Problema significativo (*Significant*), em que os estudantes irão resolver problemas do cotidiano; Mesmo Problema (*Same*), no qual todas as equipes recebem o mesmo problema ao mesmo tempo; Escolha específica (*Specific*), em que as equipes irão gerar respostas curtas e de fácil entendimento para as outras equipes; e Relatos

simultâneos (*Simultaneous report*), em que todas as equipes devem demonstrar as respostas ao mesmo tempo, evitando alterações na resposta por influência da argumentação de outra equipe (Bollela *et al.*, 2014).

Com o objetivo de obter níveis altos de aprendizagem do aluno, 4 pilares são necessários: equipes permanentes, formadas de maneira estratégica e adequada condução; alunos responsabilizados pela qualidade do trabalho realizado em equipe ou individualmente; fornecimento frequente de *feedback*; tarefas para desenvolvimento do trabalho em equipe e promover a aprendizagem individual (Krug *et al.*, 2016).

Segundo Terçariol (2017), a TBL demonstra ser um meio colaborativo para incentivar a aprendizagem com maior eficácia, fazendo com que o aluno tenha experiências de trabalho em grupo. Algumas vantagens são demonstradas com a sua aplicação, por exemplo: o estudante acostuma-se a buscar o conhecimento por mais de uma fonte; a partir dos estudos iniciais, o aluno torna-se mais presente nas participações das aulas; e desenvolve e melhora a comunicação interpessoal através das atividades em grupo e entre outras.

Dart

Em 2011 o Google apresentava o Dart, uma linguagem de programação cujo principal objetivo era substituir a principal linguagem embutida nos navegadores, o JavaScript (Andrade, 2019).

O Dart tornou-se a base da codificação no desenvolvimento das aplicações com o framework Flutter graças a algumas características, como: alta performance, orientação a objetos, produtividade de desenvolvimento e etc. A sintaxe é baseada na linguagem C e traz algumas características principais, como a forte tipagem, tipos genéricos, orientação objeto e entre outros.

Framework Flutter

O Flutter é um conjunto de ferramentas para interface gráfica desenvolvido pela Google para a criação de belos aplicativos e compilados nativamente, disponibilizando a possibilidade de funcionamento multiplataforma através da mesma base de código (FLUTTER, 2015). Foi anunciado em 2015 por Eric Seidel; na primeira fase experimental, recebeu o nome de Sky e, ao passar do tempo, recebeu o nome de Flutter. A versão 1.0 foi lançada em dezembro de 2018 como a versão estável (Bueno, 2021).

O Flutter usa uma estratégia de criação de interfaces de usuários de maneira declarativa, ou seja, o desenvolvimento inteiro é feito através de linhas de código, utilizando a unidade básica para construção o widget (Bueno, 2021). Com o seu desenvolvimento baseado em widgets, existem tipos diferentes para cada finalidade, por exemplo, widgets para elementos estruturais como botões, menus, etc., elementos de estilização como fontes, cores, etc. Dentro de toda essa separação, há widgets específicos para as plataformas, seja mobile, web ou desktop (Corazza, 2022).

O Flutter tem um diferencial para as outras opções de criação, e Corazza o descreve com as seguintes palavras: “O que torna o Flutter diferente da maioria das outras opções para criar aplicativos móveis é que ele não utiliza os widgets fornecidos com o dispositivo. Em vez disso, utiliza o seu próprio mecanismo de renderização de alto desempenho para desenhar widgets (Corazza, 2022).

GraphQL

O GraphQL pode ser descrito formalmente como uma linguagem de consulta e mecanismo de descrição dos recursos e requisitos de modelo de dados feito para aplicações cliente-servidor (Quiña-Mera *et al.*, 2022). Sem a necessidade de uma linguagem específica ou mecanismo de persistência, o GraphQL converte os recursos em uma linguagem uniforme seguindo 5 princípios: hierarquias, centrada no produto, de tipagem forte, consultas específicas pelo cliente e introspectivas (GraphQL, 2018).

O GraphQL foi criado pelo Facebook em 2012 e, de acordo com Byron (2015), o começou a ser implementado em 2012, por causa da decisão de reconstruírem as suas aplicações móveis. O autor descreve um dos motivos: “Quando construímos os aplicativos móveis do Facebook, precisávamos de uma API que busca dados poderosa o suficiente para descrever todo o Facebook, mas simples o suficiente para ser fácil de aprender e usar por nossos desenvolvedores de produtos” (Byron, 2022).

Embora o projeto tenha sido iniciado pelo *Facebook Inc.* a partir do ano de 2020, o grupo *GraphQL Foundation* assumiu os direitos atuais e o Facebook tornou-se membro da *GraphQL Foundation* (GRAPHQL, 2015).

Hasura

Hasura é descrita como uma plataforma de código aberto, com a capacidade de fornecer interfaces para aplicativos (APIs) de alto nível industrial, baseando-se nos dados fornecidos, retirando a exigência de que os desenvolvedores se preocupem com a criação, funcionamento e escalonamento de um servidor GraphQL (Clark, 2020).

O funcionamento do GraphQL no Hasura é eficiente e veloz, disponibilizando em tempo real APIs GraphQL sobre o Postgres (banco padrão utilizado), esquemas remotos e webtriggers de eventos no banco de dados. Para ser utilizado o desenvolvedor precisará executá-lo em um sistema *cloud* ou localmente em um servidor próprio, em sequência conectar-se a novos bancos ou bancos já existentes para obter os recursos disponíveis da plataforma (CLARK, 2020).

METODOLOGIA

Para o desenvolvimento deste projeto foram necessários seguir as seguintes etapas: Estudo dos conceitos e formas de aplicação da TBL; Análise e levantamento de requisitos para construção de software que possa apoiar a aplicação da TBL; Estudo das tecnologias para desenvolvimento do protótipo de software; Implementação de um protótipo de software; Aplicação de testes em sala de aula para medir o resultado.

Para a compreensão e estudo da metodologia foram realizadas pesquisas em busca de artigos ou trabalhos sobre a metodologia TBL. Através desta pesquisa foi encontrado o artigo “Aprendizagem baseada em equipes: Da teoria à prática.” de BOLLELA *et al.*, 2014. Um artigo que contextualiza a metodologia e explica as etapas da metodologia enquanto repassa alguns pontos principais de sua aplicação.

Também se observou algumas aplicações da metodologia por parte dos professores do Centro Universitário Governador Ozanam Coelho. O autor do texto participou das aulas para entender o desenvolvimento da metodologia entre os alunos e o professor aplicador. Esta experiência ajudou na análise para encontrar os principais

pontos que são necessários para se ter na aplicação, nomeados como requisitos funcionais.

Para atingir o objetivo final houve a necessidade de selecionar uma ou mais linguagens. Para isto foram escolhidos um Framework e uma linguagem, que atendiam as necessidades para front-end e back-end, sendo a linguagem Dart e o Framework Flutter para o front-end e o somente o Dart para back-end assim mantendo a base de código para os dois lados da aplicação. Para persistência dos dados foi escolhido o PostgreSQL juntamente com o Hasura para se ter uma plataforma de comunicação com o banco externa a aplicação e o GraphQL como linguagem de consulta.

Todas as tecnologias foram escolhidas por serem tecnologias onde já existe um domínio sobre elas e tornando a curva de aprendizagem sobre o uso de certos recursos menor e por conseguir manter a base de código nas aplicações, economizando algumas questões de reescrita de código.

Após a aquisição dos conhecimentos necessários, decidiu-se por um projeto focado na etapa 2 da TBL (garantia de preparo), para ficar adequado ao tempo disponível para o desenvolvimento do trabalho e por ser considerado a etapa mais trabalhosa ao professor.

Em seguida, iniciou-se a fase de implementação do software, conforme descrito na seção Desenvolvimento deste artigo. Após a construção do software, com o objetivo de analisar o resultado e a eficácia da hipótese levantada por este artigo, o software foi utilizado em uma aula real, da disciplina de Sistemas de Informação, no curso de Ciência da Computação do UNIFAGOC.

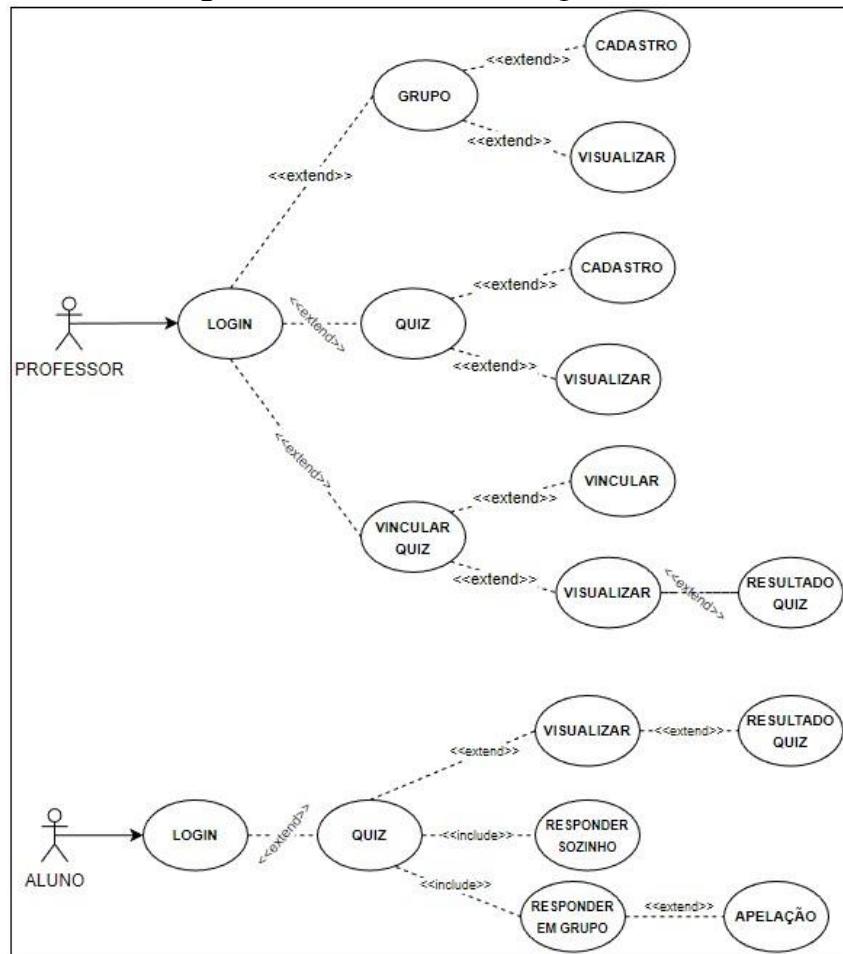
Após a aula, os alunos foram convidados a responder a um questionário através do *Google Forms*, objetivando avaliar a eficácia do software e a satisfação do aluno. O resultado dessa avaliação é exibido na sessão Resultados.

DESENVOLVIMENTO

Construção da aplicação

Com base nos estudos sobre TBL e na experiência vivida em sala de aula, realizou-se a análise e definição dos requisitos funcionais, gerando o caso de uso apresentado na Figura 1.

Figura 1 – Caso de uso da aplicação

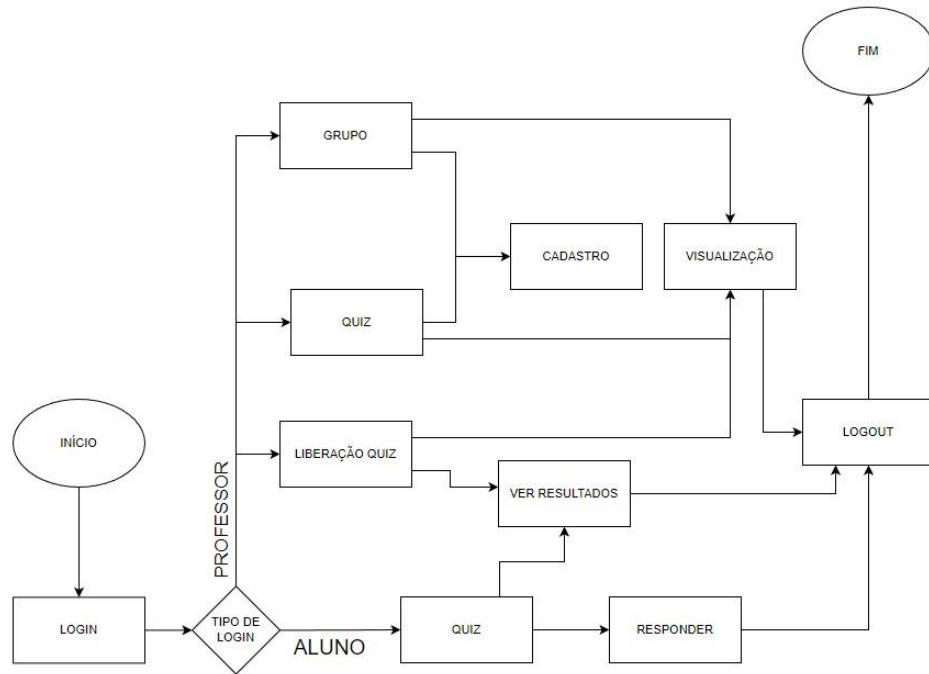


Fonte: os autores, 2022.

Como resultado, foi desenvolvido um protótipo para aplicação da metodologia TBL, cujo objetivo era fornecer ao professor a possibilidade de cadastrar quizzes e liberá-los para grupos de aluno em datas escolhidas por ele, cadastrar os grupos de alunos e visualizar os resultados dos quizzes. O aluno teria as funcionalidades de responder ao quiz e ver o seu resultado.

Dentro do projeto da aplicação foram definidos alguns detalhes, como a necessidade de gerar dois fluxos de uso, separados pelo login, o qual é dividido em duas opções: Professor e Aluno. Essa divisão pode ser observada na Figura 2, que ilustra a projeção simplificada do fluxo da aplicação.

Figura 2 – Fluxo Simplificado da Aplicação

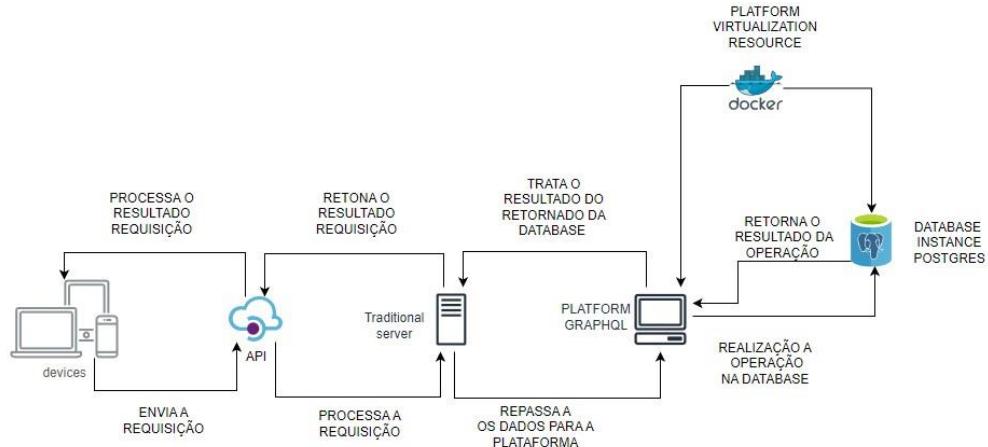


Fonte: os autores, 2022.

Como pode ser observado na Figura 2, a maior parte do fluxo fica concentrada para o Professor, o qual será responsável por toda a realização de cadastrados e controle da liberação do quiz, assim como da observação dos resultados referentes a eles. Ao aluno resta responder ao *quiz* e verificar os resultados.

Para comunicação entre as aplicações, foi estudado o método de comunicação por API, uma interface de comunicação que define como as aplicações se comunicam. Através desse sistema de API, é informado para a aplicação no servidor o que é necessário realizar de tarefa na base de dados. A aplicação repassa essas informações para a plataforma de gerenciamento do GraphQL, a qual irá fazer a manipulação dos dados. Em seguida, é feito o retorno do resultado, o qual é tratado pelo back-end e devolvido para o front-end. Este fluxo é exemplificado na Figura 3, na qual se observa a menção do Docker no fluxo.

Figura 3 – Fluxo de comunicação da Aplicação



Fonte: os autores, 2022.

Docker é uma tecnologia *open source* desenvolvida pela comunidade em conjunto com a empresa Docker Inc., com o objetivo de facilitar a criação e a distribuição de ambientes isolados. Ele empacota as aplicações e recursos dentro da ideia de um container, tornando-o portátil; assim, consegue copiar, migrar e implantar o ambiente com maior flexibilidade (Guedes, 2022). A aplicação é utilizada para manter os recursos referentes à base de dados e à plataforma de gerenciamento do GraphQL em um mesmo local, facilitando sua manipulação.

Funcionamento da Aplicação

Para utilização da aplicação, o primeiro passo para o usuário seria efetuar o login, cuja tela pode ser observada na Figura 4. O usuário utilizará o seu CPF e senha para entrar, contudo o cadastro neste protótipo ainda não está disponível.

Figura 4 – Tela de Login



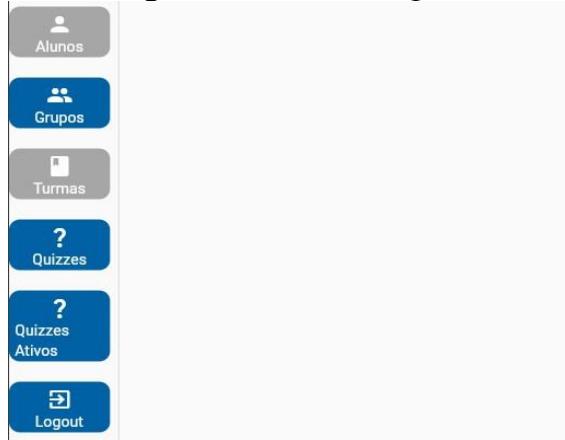
Fonte: os autores, 2022.

Após efetuado o login, o sistema identifica o tipo de usuário, fazendo a separação dos acessos pelos tipos de usuário, professor ou aluno. A seguir, será apresentada a visão do professor e a do aluno sobre o funcionamento do software.

Visão do Professor

Caso o usuário seja do tipo Professor, ele terá na lateral esquerda um menu com 5 opções além da opção de Logout (Figura 5): acessar a área do *quiz*, a de liberação do *quiz* e a de grupos; as outras duas opções são de área a serem desenvolvidas.

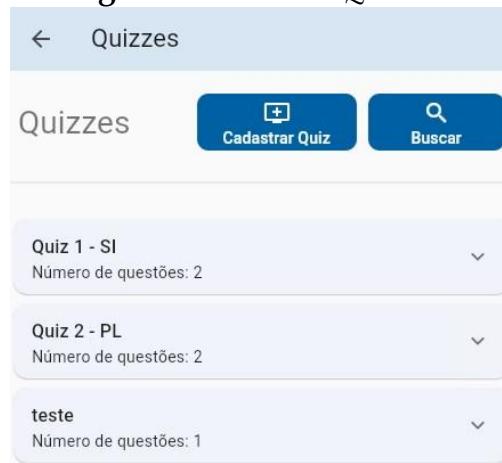
Figura 5 - Tela Principal



Fonte: os autores, 2022.

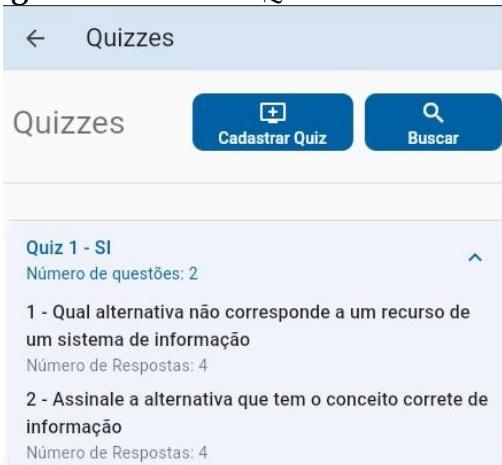
Acessando a opção *quiz*, serão apresentados todos os *quizzes* já cadastrados no sistema, como pode ser observado na Figura 6. Nessa apresentação serão informados o título do quiz e a quantidade de questões que há nele; ao clicar em cima dele, o card irá expandir, demonstrando as perguntas (Figura 7).

Figura 6 - Tela dos Quizzes



Fonte: os autores, 2022.

Figura 7 – Tela dos Quizzes Detalhados



Fonte: os autores, 2022.

Observando a Figura 6, no canto superior direito, há a opção de cadastrar um *quiz*. Ao se clicar nessa opção, abrirá uma tela para preencher o título do *quiz* e a quantidade de questões dele, como mostra a Figura 8. Na parte inferior da tela, observa-se um botão para confirmar esses dados; após essa confirmação, será repassado para a tela onde será feito o cadastro das questões e respostas (Figura 9).

Na imagem dessa tela, existem algumas numerações demonstrando alguns detalhes do cadastro. O (1) indica o número de questões a se cadastrar, com um contador ao lado, demonstrando quantas já foram adicionadas; o (2) indica o local onde será escrita a pergunta; o (3) indica o número de respostas referentes àquela pergunta; modificando-se este dado através dos botões ao lado do campo, o sistema já irá corrigir a quantidade de campo que será disponibilizada para preencher com o texto referente à resposta; esses campos podem ser identificados pelo número (4); e, ao final, para a conclusão do preenchimento dos dados, estará disponível o botão “adicionar” na parte inferior da tela, dando a possibilidade de salvar essa pergunta vinculada ao *quiz*; após essa adição, irá aparecer a pergunta na lista abaixo do texto “Perguntas Cadastradas”, área que pode ser identificada pelo número (5).

Figura 8 – Tela de Cadastro Inicial do Quiz

A interface de usuário (UI) da Figura 8 é uma tela de cadastro inicial para um quiz. No topo, uma barra azul contém o link "[← Quiz Cadaster](#)". Abaixo, há dois campos de texto: "Título" e "Número de questões". O campo "Número de questões" contém o valor "1". Na base da tela, há um botão cinza com o texto "Próximo".

Fonte: os autores, 2022.

Figura 9 – Tela de Cadastro de Perguntas e Respostas

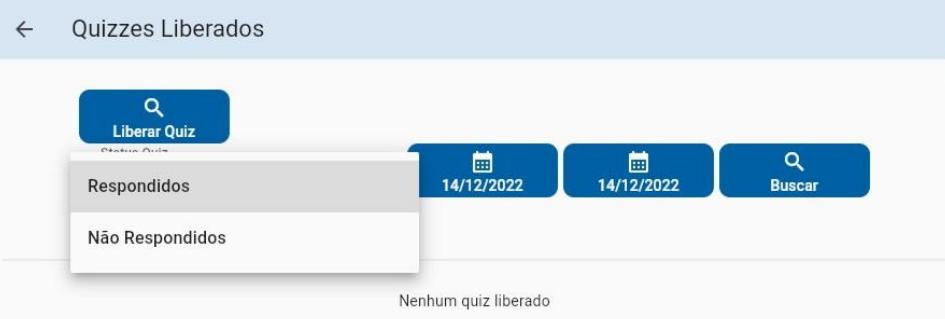
A interface de usuário (UI) da Figura 9 é uma tela para cadastrar perguntas e respostas. No topo, uma barra azul contém o link "[← Cadastrar Perguntas](#)" e o texto "(1) Nº Perguntas Cadastradas: (0/1)". Abaixo, há campos para "Descrição" (2) e "Pontuação Máxima" (3), com o valor "1" digitado. A seção "Respostas a Cadastrar" contém campos para "Resposta 1" (4) e "Pontuação Máxima" (5), com o valor "4" digitado. Um checkbox "Resposta Correta" está desmarcado. Abaixo, uma lista mostra "Perguntas Cadastradas" (5) e um botão cinza com o texto "Salvar".

Fonte: os autores, 2022.

Após o cadastro, o sistema retorna para a tela principal da aplicação e, no menu lateral esquerdo, haverá a opção de vincular o *quiz*, como demonstrado na Figura 5. Ao clicar nele, abrirá uma tela que demonstra os quizzes já vinculados. A busca dos

quizzes divide-se em *quizzes* respondidos e não respondidos, conforme ilustrado pela Figura 10.

Figura 10 – Tela de Vínculo de Quiz e Resultado do Quiz



Fonte: os autores, 2022.

Figura 11 – Tela de Resultado do Quiz

Grupo	Pontuação Respondida	Pontuação Validada
000001	0.0	0.0
000002	0.0	0.0
000001	0.0	0.0

Grupo	Pontuação Respondida	Pontuação Validada
000001	1.0	1.0
000002	1.0	1.0
000001	2.0	2.0

Fonte: os autores, 2022.

Caso o *quiz* seja do tipo respondido, ao clicar sobre ele são exibidos os resultados, separados em resultado individual dos alunos e dos grupos. As perguntas serão exibidas com o resultado ocultado; para visualizá-las, é necessário clicar em cima dela. A Figura 11 demonstra como estará descrita a resposta do aluno e, abaixo, a resposta do grupo dele.

Visão do Aluno

Para o usuário do tipo aluno, ao fazer o login ele irá acessar a tela principal do aluno, demonstrada na Figura 12, onde ele, ou acessa a área dos quizzes, ou se desconecta do sistema.

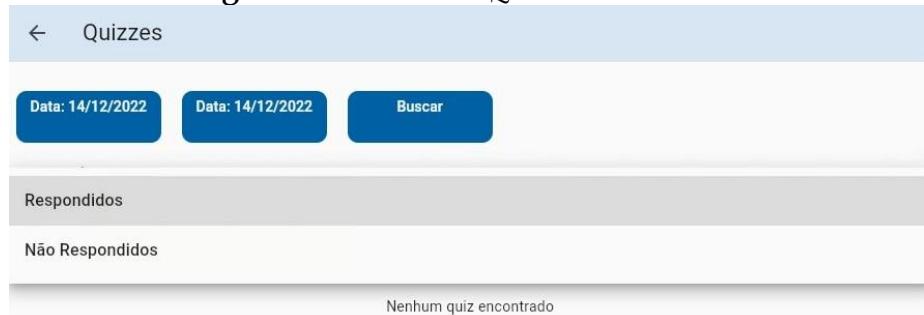
Figura 12 – Tela Principal do Aluno



Fonte: os autores, 2022.

A área dos *quizzes* apresenta a opção de buscar os *quizzes* que podem ser divididos em dois: respondidos e não respondidos, como demonstrado na Figura 13. Quando um *quiz* estiver como respondido, ao clicar em cima dele, será possível ver o resultado seguindo o padrão demonstrado na Figura 14.

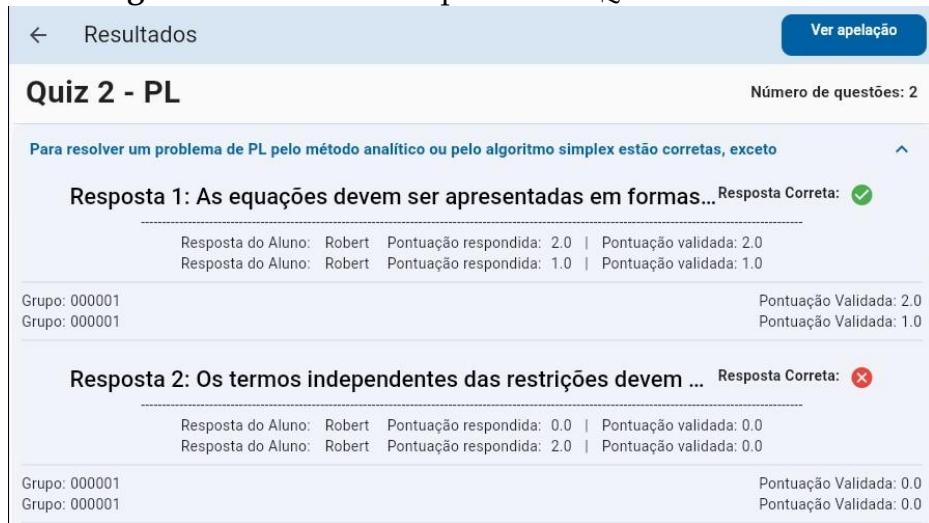
Figura 13 – Tela dos Quizzes do Aluno



Fonte: os autores, 2022.

Quando o aluno estiver pesquisando um *quiz* e este já tiver sido respondido, abrirá uma tela com a sua resposta e a do seu grupo, seguindo o padrão apresentando abaixo, na Figura 14, informando a pontuação vinculada à resposta dada pelo aluno e quantos pontos foram validados nela; no grupo, demonstra-se a pontuação obtida após as tentativas de responder à questão.

Figura 14 – Tela das Respostas dos Quizzes do Aluno



Fonte: os autores, 2022.

Se for um quiz ainda não respondido, ao clicar nele, uma tela se abrirá para que os alunos deem as suas respostas individuais. Na lateral direita da resposta, haverá um campo para o aluno digitar o valor referente à questão, como é demonstrado na

imagem da Figura 15. Para dar continuidade às respostas, haverá botões na parte inferior para avançar e retornar. Ao final da última questão, o botão para salvar será utilizado para salvar as respostas no banco de dados. Após finalizar o *quiz* individual, iniciará o *quiz* em grupo, no qual o aluno, ao clicar em cima da resposta, será informado se a resposta está correta ou não (Figura 16).

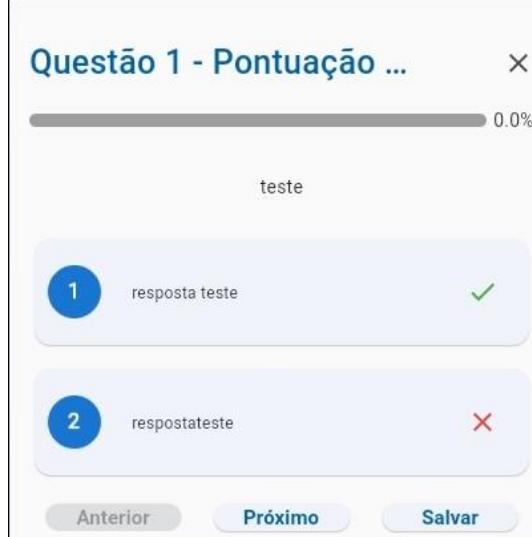
Os passos para prosseguir no *quiz* em grupo e salvá-lo será o mesmo do *quiz* feito individualmente. Ao finalizar o *quiz* em grupo, irá ser perguntado se deseja fazer a apelação. Em caso afirmativo, aparecerá uma janela para que se preencha o campo com a apelação desejada (Figura 17); caso contrário, retorna-se para a tela a fim de pesquisar os *quizzes*.

Figura 15 – Tela das Respostas Sozinho e em Grupo



Fonte: os autores, 2022.

Figura 16 – Tela das Respostas Sozinho e em Grupo



Fonte: os autores, 2022.

Figura 17 – Tela da Opção de apelação



Fonte: os autores, 2022.

RESULTADO

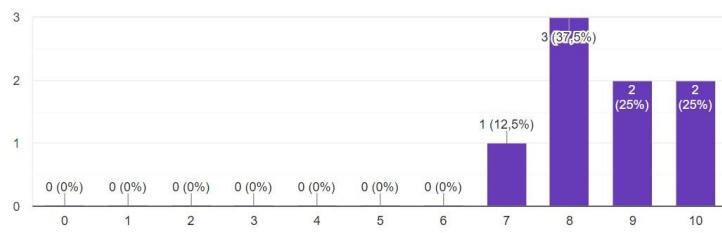
O teste do protótipo de software foi realizado no Centro Universitário Ozanam Coelho (UNIFAGOC), localizado na Rua Doutor Adjalme da Silva Botelho, nº 20, em Ubá-MG, cujo site para acesso é <https://unifagoc.edu.br/>. O software foi utilizado em uma aula real da disciplina de Sistemas de Informação do curso de Ciência da Computação. Participaram desse experimento 8 alunos, do segundo e do oitavo períodos. A partir de um tema já estudado pelos alunos, eles responderam ao questionário individualmente e em grupo, separados em dois grupos, e, ao final, consultaram os resultados e tiveram a oportunidade de realizar a apelação.

Após o experimento, os alunos responderam a um questionário para uma avaliação qualitativa do software. Cada pergunta foi avaliada em uma escala de 0 a 10. Na primeira, foi perguntado aos alunos se o software possuía todas as funcionalidades necessárias. Como pode ser observado na Figura 18, para a maioria (37,5%), o software possui funcionalidades para a metodologia, mas ainda necessita de outras para ter todas as necessárias.

Figura 18 – Resultado da Primeira Pergunta

Com base na sua experiência de estudo através do método TBL, você acredita que este software possui todas as funcionalidades necessárias para aplicação do método em sua plenitude? Considere que a resposta 0 indica que não possui nenhuma funcionalidade necessária e 10 que possui todas as funcionalidades necessárias.

8 respostas



Fonte: os autores, 2022.

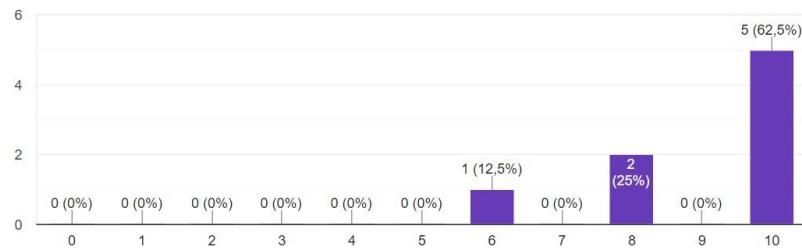
Já na segunda pergunta, indagou-se aos alunos se a utilização do software trouxe praticidade e eficiência à aplicação do processo. Ao observar o gráfico apresentado na Figura 19, verifica-se que, para a maioria dos alunos (62,5%), o

software tornou mais prático e eficiente o processo, enquanto para 12,5% dos alunos ele demonstrou uma eficiência mediana.

Figura 19 – Resultado da Segunda Pergunta

Em comparação com a sua experiência em responder as perguntas em uma folha de papel, você acredita que através da utilização do software este processo se tornou mais prático e eficiente? Considere que a resposta 0 indica que o software não prático e eficiente de nenhuma maneira e que 10 é totalmente prático e eficiente.

8 respostas



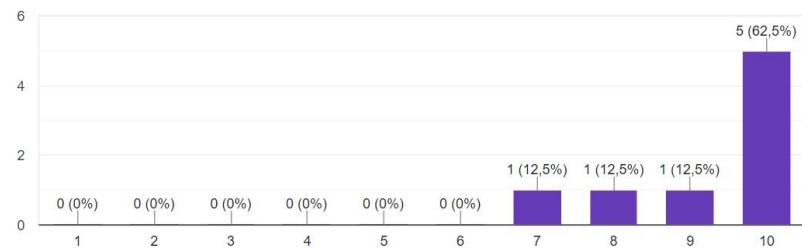
Fonte: os autores, 2022.

A terceira pergunta versou sobre a satisfação da experiência do aluno ao realizar a metodologia TBL utilizando o software. De acordo com o gráfico da Figura 20, em sua maioria (62,5%), os alunos acharam satisfatória a experiência da utilização do software na metodologia.

Figura 20 – Resultado da Terceira Pergunta

Com base na sua experiência de estudo, como você qualifica a sua experiência em realizar a metodologia TBL utilizando um software? Considere a resposta 0 totalmente insatisfatória e a resposta 10 completamente satisfatória.

8 respostas



Fonte: os autores, 2022.

CONCLUSÃO

A concepção deste trabalho resultou em um protótipo de uma plataforma web para auxiliar a aplicação da metodologia TBL, realizando a automação de alguns processos feitos até então de forma manual e em papéis. Explorando tecnologias atuais, como o Flutter, foi possível fornecer acesso fácil para o usuário, possibilitando a utilização de dispositivos móveis e a separação dos grupos em salas diferentes.

A aplicação do Docker trouxe uma agilidade na troca de localidade dos recursos referentes à base de dados e ao sistema gerenciador, em conjunto com a utilização do GraphQL, que possibilitou economia de tempo no momento do desenvolvimento pelos recursos pré-prontos oferecidos, resultando em um desenvolvimento prático e ágil.

Para verificar se o software atende ao seu propósito principal, que é apoiar a aplicação da metodologia TBL, foi aferida a experiência do usuário em relação aos critérios de satisfação, à existência das funcionalidades necessárias e à praticidade em relação ao uso de papel. Os resultados dessa avaliação mostraram-se consideravelmente positivos em todos os critérios, mas evidenciou-se também a necessidade de evolução.

Com base nesses resultados, pode-se concluir que o software se demonstrou promissor, embora necessite de melhorias em alguns aspectos, podendo ser uma ótima ideia para se dar continuidade a trabalhos futuros, tais como: a implementação das funcionalidades de *UPDATE*, *DELETE* na parte dos cadastros; a adição do cadastro dos usuários; a possibilidade de cadastrar o material para estudo na plataforma; a opção de liberar atividades para os alunos seguindo a ideia da 3^a fase da metodologia para a parte dos professores. No que diz respeito aos alunos, sugere-se adicionar um mecanismo em que o aluno líder do grupo possa responder ao questionário enquanto o restante somente visualizasse as perguntas e respostas; e, no geral da aplicação, a repaginação do designer da aplicação.

Em caso de interesse em visualizar o código-fonte da aplicação por parte do *Front-end* e *back-end*, estão disponíveis os seguintes links:

- Front-End: https://github.com/RobertsFerreira/dashboard_tbl
- Back-End: https://github.com/RobertsFerreira/api_tbl_tcc

REFERÊNCIAS

ANDRADE, Kleber. Introdução a linguagem de programação Dart, 2019. Disponível em: <https://medium.com/flutter-comunidade-br/introdu%C3%A7%C3%A3o-a-linguagem-deprograma%C3%A7%C3%A3o-dart-b098e4e2a41e>. Acesso em: 22 set. 2022.

BECK, Caio. **Metodologias ativas**: conceito e aplicação, 2018. Disponível em: <https://www.andragogiabrasil.com.br/metodologias-ativas>. Acesso em: 28 ago. 2022.

BOLLELA, Valdes Roberto *et al.* Aprendizagem baseada em equipes: da teoria à prática. **Medicina**, Ribeirão Preto, v. 47, n. 3, p. 293-300, 2014. Disponível em: <https://www.revistas.usp.br/rmrp/article/view/86618>. Acesso em: 24 set. 2022.

BRASIL, Lei 14.407, de 13 de julho de 2022. Altera a Lei nº 9.394, de 20 de dezembro de 1996 (Lei de Diretrizes e Bases da Educação Nacional), para estabelecer o compromisso da educação básica com a formação do leitor e o estímulo à leitura. **Diário Oficial da União**, Brasília, 8 dez. 2022.

BUENO, Carlos Eduardo de Oliveira. **Desenvolvimento de um aplicativo utilizando o framework flutter e arquitetura limpa**, 2021. Disponível em: <https://repositorio.pucgoias.edu.br/jspui/handle/123456789/1861>. Acesso em: 20 set. 2022.

BYRON, L. **GraphQL**: a data query languages, 2015. Disponível em: <https://engineering.fb.com/2015/09/14/core-data/graphql-a-data-query-language/>. Acesso em: 17 ago. 2022.

CARNEIRO, Neri de Paula. **A educação no Brasil**: avanços e problemas, 2008. Disponível em: <https://meuartigo.brasilescola.uol.com.br/educacao/a-educacao-no-brasil-avancosproblemas.htm>. Acesso em: 02 dez. 2022.

- CLARK, M. **Hasura vs Firebase**. Disponível em: https://blog.back4app.com/pt/hasura-vs-firebase/#O_que_e_o_Hasura. Acesso em: 19 ago. 2022.
- CORAZZA, Paulo Victor. **Um aplicativo multiplataforma desenvolvido com flutter e NoSQL para o cálculo da probabilidade de apendicite**, 2018. Disponível em: <https://www.lume.ufrgs.br/handle/10183/190147>. Acesso em: 22 set. 2022.
- DAVID, Ellen Adriana *et al.* Aspectos da evolução da educação brasileira. **Revista Uniaraguaia**, v. 5, n. 5, p. 184-200, 2014. Disponível em: <http://www.faculdadearaguaia.edu.br/sipe/index.php/REVISTAUNIARAGUAIA/article/view/199/182>. Acesso em: 12 ago. 2022.
- FILPO, Sara. **LDB 2022** - alterações recentes na Lei de Diretrizes e Bases 9.394/96. Disponível em: <https://pedagogiadescomplicada.com/ldb-2022-alteracoes/>. Acesso em: 08 dez. 2022.
- FLUTTER. 2015. Disponível em: <https://flutter.dev/>. Acesso em: 19 set. 2022.
- GRAPHQL. **Introdução ao GraphQL**, 2015. Disponível em: <https://graphql.org/learn/>. Acesso em: 19 ago. 2022.
- GRAPHQL. The GraphQL Foundation, 2018. Disponível em: <http://spec.graphql.org/June2018/>. Acesso em: 10 dez 2022.
- GUEDES, Marylene. **No final das contas**: o que é o Docker e como ele funciona, 2022. Disponível em: <https://www.treinaweb.com.br/blog/no-final-das-contas-o-que-e-o-docker-e-como-ele-funciona>. Acesso em: 27 nov. 2022.
- HARTZ, Ani Mari; SCHLATTER, Gabriel Vianna. A construção do trabalho de conclusão do curso por meio da metodologia ativa Team-Based Learning. **Administração: Ensino e Pesquisa**, v. 17, n. 1, p. 73-109, 2016. Disponível em: <https://raep.emnuvens.com.br/raep/article/view/274/pdf>. Acesso: 29 nov. 2022.
- KRUG, Rodrigo de Rosso *et al.* O “bê-á-bá” da aprendizagem baseada em equipe. **Revista Brasileira de Educação Médica**, v. 40, p. 602-610, 2016. Disponível em: <https://www.scielo.br/j/rbem/a/w5Tg86RL75mkjX7yZhmNQ6F/abstract/?lang=pt>. Acesso em: 18 nov. 2022.
- LOUREIRO, Mirela Malveira *et al.* O uso da metodologia ativa TBL como método de ensino na aula de monitoria, 2020. **Brazilian Journal of Development**, v. 6, n. 7, p. 42740-42747, 2020. Disponível em: <https://ojs.brazilianjournals.com.br/ojs/index.php/BRJD/article/view/12559>. Acesso em: 09 dez. 2022.
- QUIÑA-MERA, Antonio *et al.* GraphQL: a systematic mapping study. **ACM Computing Surveys (CSUR)**, 2022. Disponível em: <https://dl.acm.org/doi/abs/10.1145/3561818>. Acesso em: 11 nov. 2022.
- TERÇARIOL, Adriana Lima. **Team based learning**: uma metodologia ativa para auxílio no processo de aprendizagem, 2017. Disponível em: <http://www.unoeste.br/site/enepe/2017/suplementos/area/Humanarum/4%20-%20Educa%C3%A7%C3%A3o/TEAM%20BASED%20LEARNING%20UMA%20METODOLOGIA%20ATIVA%20PARA%20AUXILIO%20NO%20PROCESSO%20DE%20APRENDIZAGEM.pdf>. Acesso em: 29 nov. 2022.