

# APLICAÇÃO MOBILE PARA GESTÃO DE AULAS EAD: UTILIZANDO AS TECNOLOGIAS *OFFLINE FIRST* E *OPTIMISTIC USER INTERFACE*<sup>1</sup>

Edson Junio Malaquias<sup>2</sup>  
Juliana Lilis da Silva<sup>3</sup>

**RESUMO:** Neste artigo, trata-se sobre o desenvolvimento de uma aplicação voltada para a área da *EdTech*, essa área realiza a junção da educação com a tecnologia. A aplicação tem como objetivo auxiliar alunos de instituições que estudam através da modalidade de educação a distância (EAD) progredirem com o aprendizado, realizando o estudo de diversos tipos de conteúdos e acompanhando o progresso acadêmico. Ela também permite que o aluno faça o download do conteúdo para o dispositivo móvel, fazendo com que ele tenha acesso mesmo sem ter uma conexão estável com a internet. Para a realização do desenvolvimento, foram utilizados alguns conceitos como o *offline first* e *optimistic user interface* juntamente com a linguagem de programação *React Native* e o banco de dados local *RealmDB*.

**PALAVRAS-CHAVE:** *Edtech; Offline First; React Native; RealmDB*.

**ABSTRACT:** This article is about the development of an application focused on the EdTech area, that combines education with technology. The application aims to help students from institutions that study through distance education progress with learning, carrying out the study of various types of content and monitoring academic progress. It also allows students to download the content to the mobile device giving them access even without having a stable internet connection. To carry out the development, some concepts were used, such as *offline first* and *optimistic user interface* together with the *React Native* programming language and the local database *RealmDB*.

**KEYWORDS:** *Edtech; Offline-First; React-Native; RealmDB*.

## 1 INTRODUÇÃO

O avanço da tecnologia nas últimas décadas vem revolucionando o modo como a educação é distribuída entre professores e alunos. Esse avanço permitiu a popularização da internet e, com isso, diversas áreas foram beneficiadas, sendo uma delas a educação a distância. A educação a distância (EAD) vem crescendo muito nos últimos anos o que gera um grande desafio para os fornecedores de conteúdo como a criação e disponibilização de conteúdos em larga escala.

De acordo com o censo realizado pela ABED (Associação Brasileira de Educação a Distância), no ano de 2018 foram matriculados um total de 2.358.934 alunos em cursos totalmente a distância, com um crescimento de aproximadamente 56%. E nos cursos semipresenciais, foram matriculados 2.109.951 alunos, com um crescimento

---

<sup>1</sup> Artigo apresentado com a finalidade de obter o título de bacharel de Sistemas de Informação.

<sup>2</sup> Estudante de Graduação do Curso de Sistemas de Informação do UNIPAM. E-mail: [edsonmalaquias@unipam.edu.br](mailto:edsonmalaquias@unipam.edu.br).

<sup>3</sup> Orientadora, Mestre em Ciência da Computação (UFU) e docente do UNIPAM. E-mail: [juliana@unipam.edu.br](mailto:juliana@unipam.edu.br).

aproximado de 53% comparado ao ano de 2017 (ABED, 2018). Assim, pode-se constatar que a quantidade de alunos matriculados em cursos EAD vem crescendo consideravelmente nos últimos anos e, também, a necessidade de boas conexões com a internet para acessar as aulas e os conteúdos dos cursos.

No entanto, percebe-se que no mundo, a conexão à internet faz com que a maioria das aplicações fiquem dependentes desta tecnologia praticamente o tempo todo. A dependência de conexão gera grandes problemas aos usuários, como o custo elevado para manter aplicações que precisam de uma conexão com a internet a todo momento. Outra situação que pode ser relatada é que pessoas de comunidades mais remotas geralmente têm um problema ainda maior, porque em muitos momentos elas não têm um bom sinal de internet e raramente conseguem se conectar a uma rede. As pessoas que passam por esses problemas evitam utilizar aplicações que consomem todos os seus dados móveis, como por exemplo, assistir uma aula remotamente.

Diante do contexto apresentado, vem surgindo diversas propostas para tentar solucionar alguns desses problemas com relação à conexão com a internet, como utilizar as aplicações em modo *offline* a maior parte do tempo, fazendo com que a única dependência com a internet seja por um breve período, somente para enviar os dados mais críticos da aplicação para um banco de dados.

Em meio a este cenário surgiu o conceito *offline first*, o qual visa resolver problemas de aplicações que dependem de internet a todo momento. A proposta desta tecnologia é que toda aplicação deve manter pelo menos suas funções essenciais funcionando independente da conexão de internet. Ao utilizar esta tecnologia, as pessoas que utilizam a aplicação não percebem que por um breve instante perderam a conexão ou, se perceberem, não ficam totalmente de mãos atadas sem poder utilizá-la. Neste modelo, no momento que essa pessoa se conectar novamente à internet as informações serão atualizadas.

Para atender a esta perspectiva, o objetivo geral deste estudo foi desenvolver uma aplicação mobile utilizando o *React Native* juntamente com o conceito *offline first*. A aplicação é destinada a alunos de instituições que trabalham com a modalidade EAD, possibilitando aos mesmos acessar as aulas por meio do celular, assistir às videoaulas e fazer os exercícios propostos. A aplicação também permite que o aluno faça o download de aulas para assistir offline. Assim, mesmo sem internet, o aluno pode utilizar a aplicação e acompanhar seu desenvolvimento acadêmico.

## 2 REVISÃO DE LITERATURA

Nesta seção são tratados temas relacionados ao projeto desenvolvido e que são de grande importância para entendimento do mesmo.

### 2.1 EDUCAÇÃO A DISTÂNCIA

De acordo com o Ministério da Educação (MEC):

Educação a distância é a modalidade educacional na qual alunos e professores estão separados, física ou temporalmente e, por isso, faz-

se necessária a utilização de meios e tecnologias de informação e comunicação. Essa modalidade é regulada por uma legislação específica e pode ser implantada na educação básica (educação de jovens e adultos, educação profissional técnica de nível médio) e na educação superior (MEC, 2009).

O aumento da educação a distância vem crescendo principalmente por oferecer maior flexibilidade, comodidade e um valor mais acessível. O aluno tem a chance de escolher o horário e o local em que vai realizar seus estudos, o que chama a atenção de diversas pessoas que querem atualizar seu currículo, porém têm o tempo muito corrido. As aulas a distância, por não precisarem disponibilizar um espaço físico e o deslocamento dos professores, faz com que o seu valor seja muito mais acessível do que os cursos presenciais e semipresenciais (SANTOS, 2018).

## 2.2 AMBIENTES VIRTUAIS DE APRENDIZAGEM

Hoje, na grande maioria dos casos, a educação a distância ou EAD é desenvolvida em ambientes virtuais de aprendizagem (AVA). Por meio destes ambientes ocorrem encontros entre alunos e professores algumas vezes por semestre. Normalmente os professores gravam suas aulas e disponibilizam no AVA, mas também é comum acontecerem aulas ao vivo. Já os cursos híbridos ou semipresenciais se dão por uma combinação dos cursos EAD e os cursos presenciais. O curso é considerado semipresencial se a carga horária presencial for maior que 20% (E+B EDUCAÇÃO; SILVA, 2018).

Os conteúdos de ensino são disponibilizados no ambiente virtual de aprendizagem, o qual é composto por um conjunto de ferramentas disponíveis na internet, sendo permitido também a interação entre os alunos e o acompanhamento dos alunos pelos professores. O objetivo do AVA é de simular uma sala de aula, assim, é possível realizar as atividades programadas, debater assuntos, visualizar o conteúdo das aulas e acompanhar todo o progresso através de relatórios (SILVA, 2020).

Devido ao cenário atual de crise gerado pela pandemia do Covid-19, os ambientes virtuais de aprendizagem tiveram um aumento significativo de acessos. Alguns países adotaram medidas de isolamento para evitar o espalhamento da doença, o que gerou a paralisação das aulas presenciais, com isso muitas instituições de ensino tiveram que se adaptar para conseguir fornecer seus conteúdos a distância se dirigindo para os AVA (SOUZA, 2020).

## 2.3 OFFLINE FIRST

*Offline First* é um conceito de arquitetura para aplicações mobile. Esse conceito propõe que a aplicação mantenha o funcionamento de suas principais funções independente da conexão com a internet. Sua implementação é de suma importância, pois desenvolver uma aplicação pensando que o usuário sempre terá conexão com a internet é um erro, pois muitos usuários sofrem com perda de sinal de internet ou até mesmo gastos exagerados em sua conta de telefone (FENANDES, 2018).

*Optimistic User Interface* é outro padrão muito utilizado com o *offline first* pois ele simula os resultados de uma ação em uma aplicação antes mesmo de receber a resposta do servidor. Esse padrão foi criado fornecendo uma maneira mais simples de fazer com que a interface do usuário responda mais rapidamente, e garantindo que os dados se tornem consistentes com a resposta real quando ela chegar do servidor (MICHEL, 2020).

### 3 METODOLOGIA

Durante o processo de desenvolvimento deste projeto foi utilizada a metodologia ágil Kanban. Normalmente, propõe a divisão de tarefas em etapas, as quais são colocadas em um quadro facilitando a visualização e compreensão para aqueles envolvidos no processo. As etapas normalmente são divididas em *to do* (a fazer), *doing* (fazendo) e *done* (feito), porém o quadro é bem flexível, permitindo que sejam realizadas adaptações de acordo com cada desenvolvimento. Desde o planejamento até a entrega de cada uma das tarefas, o quadro tem que ser mantido atualizado, realocando os *post-its* de acordo com a descrição de cada etapa.

Outros processos que foram utilizados são as técnicas comentadas na revisão de literatura, o *Offline first* juntamente com o *Optimistic User Interface*. Essas técnicas foram utilizadas para melhorar a *user interface* (UX), fazendo com que o usuário tenha uma melhor usabilidade da aplicação.

Durante o processo de desenvolvimento da aplicação foi necessário a utilização de algumas ferramentas para auxiliar e acelerar o processo de criação. A lista das ferramentas juntamente com a descrição de cada uma está apresentada no Quadro 1.

**Quadro 1:** Ferramentas e descrições

Ferramentas	Descrição
Trello	É uma ferramenta que possibilita organizar tarefas em quadros, listas e cartões. É muito utilizado em metodologias ágeis e foi utilizado para organização e acompanhamento do quadro Kanban.
Visual Studio Code (VS Code)	O VS Code é uma ferramenta para edição de códigos e foi utilizado para escrever os códigos em <i>React Native</i> .
<i>React Native</i>	<i>React Native</i> é uma linguagem de programação utilizada para o desenvolvimento de aplicativos móveis, foi utilizada para a criação das telas e lógica da aplicação.
<i>RealmDB</i>	<i>RealmDB</i> é um banco de dados criado especificamente para aplicações móveis, e foi utilizado para armazenar os dados offline da aplicação (DRUMOND, 2016).
Android Studio – Emulador	O Android Studio é uma plataforma para desenvolvimento de aplicações mobile android, porém foi utilizado somente o emulador para fins de acompanhamento de como o aplicativo estava sendo desenvolvido.
Lucid Chart	Lucidchart é um website que permite a criação de fluxogramas e modelagens em geral foi utilizado para criação da arquitetura do sistema.

Moqups	Moqups é um website que permite a criação de protótipos de interfaces e foi utilizado para a prototipação das interfaces da aplicação.
Git Hub	O Github é "um website que fornece serviço de hospedagem online e gerenciamento de código fonte, bem como controle de versão distribuído aos usuários" (BATISTA, 2017, p. 224). Este recurso foi utilizado para armazenamento e controle de versão do código fonte.

Fonte: Elaborado pelo autor, 2020.

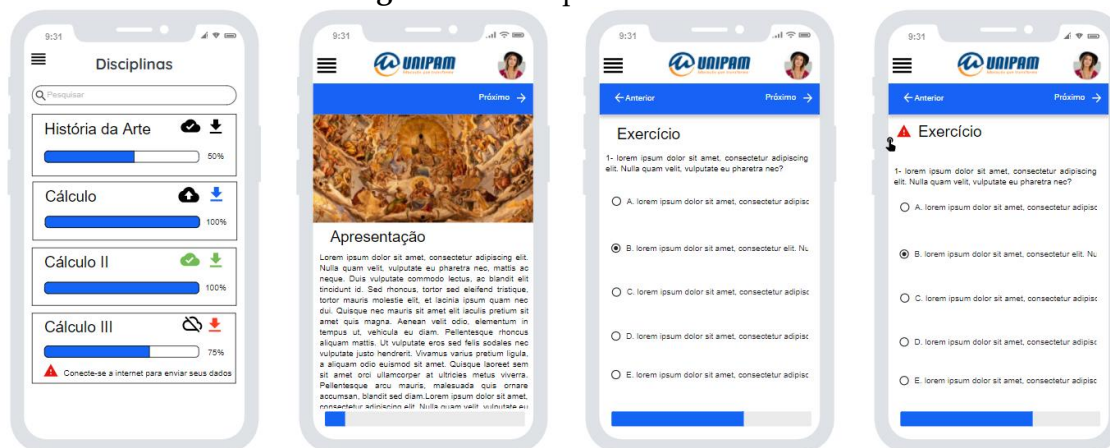
As pesquisas sobre as partes teóricas e práticas necessárias para o desenvolvimento do projeto foram feitas através de consultas e estudos de artigos acadêmicos, fóruns, sites e outros meios. Essas pesquisas proporcionaram o conhecimento necessário para desenvolver o projeto.

#### 4 DESENVOLVIMENTO E DISCUSSÃO DOS RESULTADOS

Nesta seção são apresentados os pontos mais relevantes da construção do sistema, descrevendo desde a fase inicial com a prototipação até os resultados das principais funções do sistema.

Na fase inicial do desenvolvimento foi criada a prototipação das telas da aplicação com o Moqups, conforme demonstrado na Figura 1. Essa prototipação foi uma forma rápida de destacar as principais funcionalidades e validar algumas hipóteses antes de começar o desenvolvimento das funcionalidades e também funcionou como modelo para as telas da aplicação.

Figura 1: Protótipo das interfaces

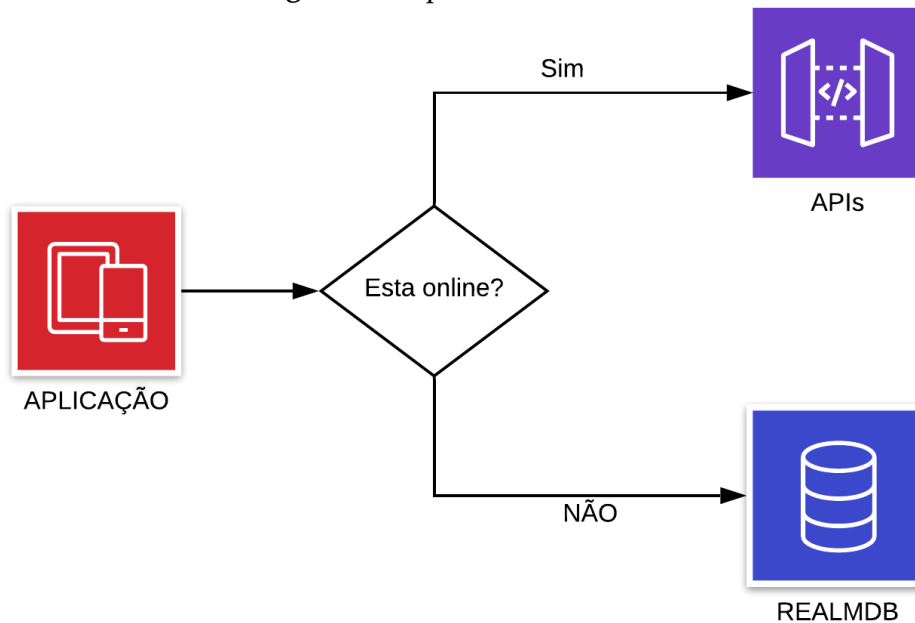


Fonte: Elaborado pelo autor, 2020.

Após a prototipação foi realizada uma análise de arquitetura para definir como a aplicação se comportaria quando estivesse offline. Essa análise foi baseada no conceito de arquitetura *offline first*, na qual o foco era a aplicação manter as suas principais funções, independente da conexão com a internet. O desenho dessa arquitetura foi criado utilizando o Lucid Chart e pode ser observado na Figura 2. Nesta Figura foi demonstrado os dois acessos aos conteúdos que são via API e RealmDB.

Assim, pode-se observar que a aplicação acessa o conteúdo por padrão na API, mesmo que o conteúdo esteja salvo no *RealmDB*, pois dessa forma sempre que o usuário estiver online é garantido que a fonte de dados que ele está utilizando é a mais atualizada e ao perder a conexão com a internet ele pode continuar o estudo do conteúdo acessando as unidades de aprendizagem que estão salvas no *RealmDB*.

**Figura 2:** Arquitetura do sistema



Fonte: Elaborado pelo autor, 2020.

Após ter os levantamentos de telas e arquitetura definidos, foram detectadas diversas tarefas a serem realizadas. Essas foram listadas no Trello para facilitar o controle do fluxo e especificação de quais tarefas eram prioritárias para as entregas do projeto. Também, foi levantada a demanda do aluno poder acessar o conteúdo de aprendizagem mesmo sem internet, pois vários alunos não têm uma conexão estável nos dispositivos móveis. Devido a essas tarefas e demandas foi realizado um estudo para selecionar algumas bibliotecas para ajudar a iniciar o desenvolvimento da aplicação, sendo elas:

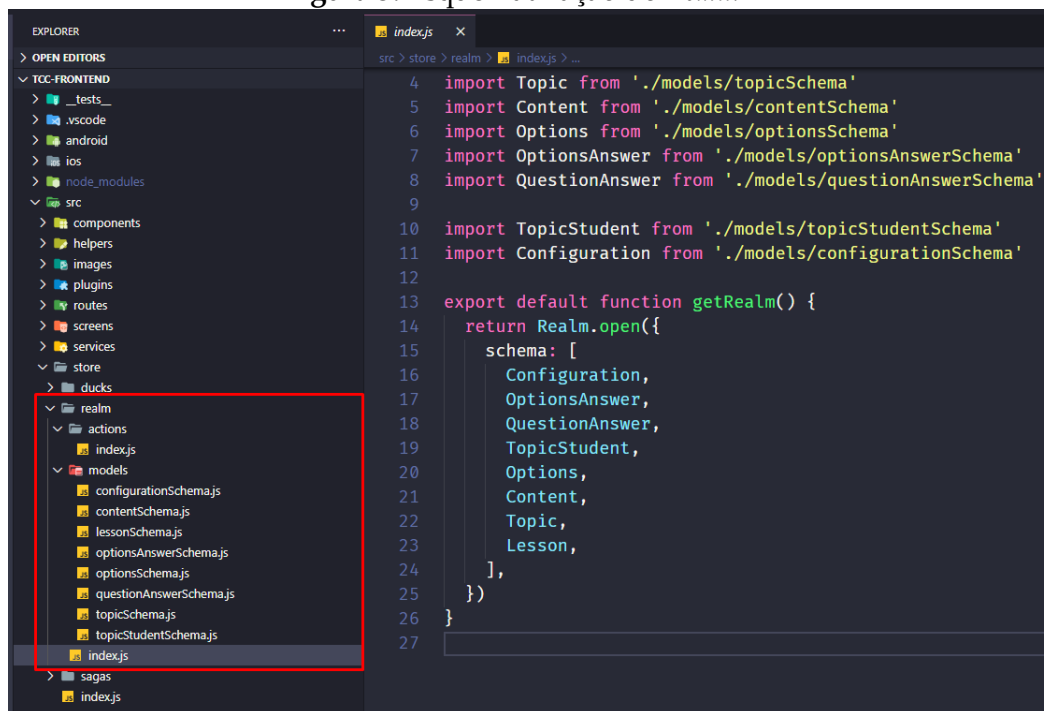
- *rn-fetch-blob*: proporcionou uma forma fácil de realizar o download dos arquivos contidos dentro das unidades de aprendizagem.
- *Cheerio*: foi utilizado para manipulação e validação dos conteúdos *HTML* (*Hypertext Markup Language*).
- *Redux*: para realizar o controle de estado, facilitando a comunicação de componentes dentro do *React Native*.
- *react-native-render-html*: foi muito utilizado na tela de conteúdo para ler os diversos tipos de conteúdo *HTML* e os transformar em elementos que o *React Native* conseguisse interpretar para realizar a renderização da interface.

Durante o desenvolvimento foi realizada a criação das telas em *React Native* seguindo os modelos de interface e de arquitetura comentados anteriormente. Para o

versionamento e armazenamento do código fonte foi utilizado o *GitHub*, pois fornece segurança no armazenamento, acesso rápido ao código e possibilita a criação de várias versões do código fonte. Nessa fase do desenvolvimento, foram criados os esquemas do *RealmDB* que definiram como o conteúdo seria salvo dentro do banco de dados local.

Como comentado, o banco de dados local da aplicação é o *RealmDB*. Este foi estruturado em um arquivo de controle, *models* e *actions*. Essa estrutura pode ser observada na Figura 3, onde a direita é mostrada a função *getRealm*, que retorna a instancia da base de dados, e à esquerda é demonstrada a divisão de pastas das *actions* e *models*.

Figura 3: Esquematização do *RealmDB*



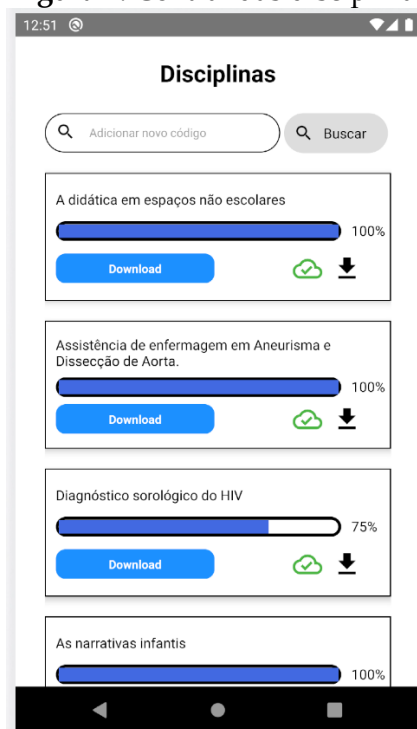
Fonte: Elaborado pelo autor, 2020.

O arquivo de controle mostrado na Figura 3 é responsável por criar uma nova instância ou retornar a instância já existente do banco de dados. Essa instância é esquematizada pelas configurações contidas nos esquemas. Os esquemas são salvos dentro da pasta *models* e são responsáveis por definir como os dados são salvos no *realmDB* e como se relacionam com os outros esquemas. As *actions* são responsáveis por intermediar toda a comunicação do banco de dados com o resto da aplicação, nelas ficam as lógicas de controle das operações do *RealmDB* como inserção, atualização, exclusão e busca.

A Figura 4 mostra a tela da central das disciplinas onde o aluno tem acesso a suas unidades de aprendizagem. Nessa tela, ele pode visualizar o progresso da unidade, adicionar novas unidades e também baixar grande parte do conteúdo através da opção de download.

Para realizar a função de download, todo o conteúdo *HTML* (*Hypertext Markup Language*) é manipulado utilizando a biblioteca *cheerio* seguindo a esquematização do banco de dados da aplicação e sendo salvo no *RealmDB*. Os arquivos são salvos no armazenamento interno do celular com o auxílio da biblioteca *rn-fetch-blob*. Essa função permite ao aluno acessar o conteúdo em qualquer lugar, independente da conexão com a internet. Lembrando que todo o conteúdo será sincronizado com a API quando o usuário tiver novamente uma conexão com a internet.

**Figura 4:** Central das disciplinas



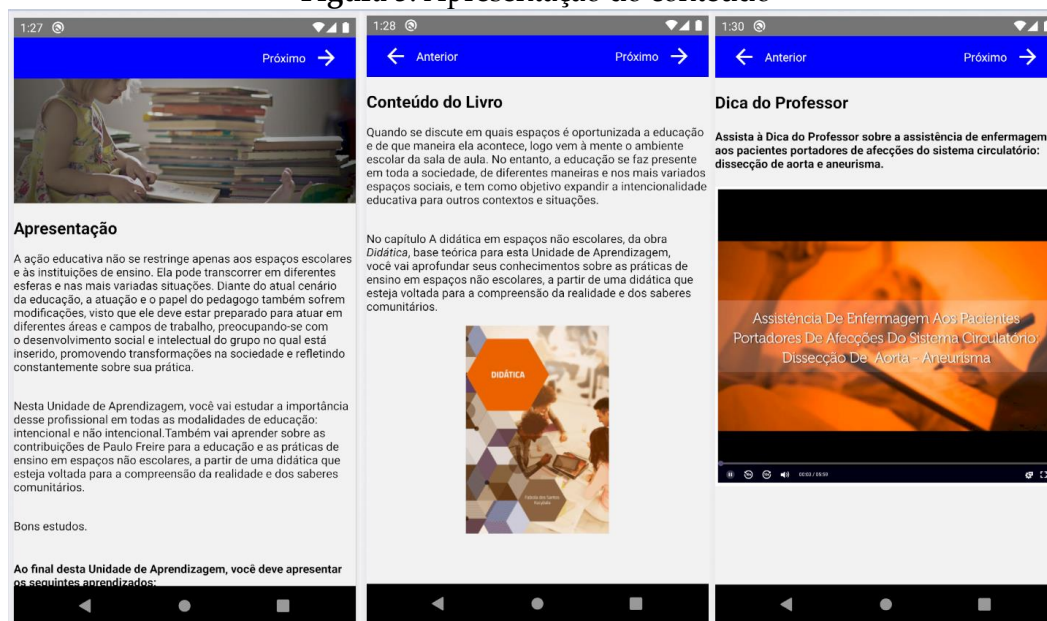
Fonte: Elaborado pelo autor, 2020.

Ao acessar uma unidade de aprendizagem o aluno entra na tela de conteúdo, conforme exemplificado na Figura 5. Nesta tela, o conteúdo dispõe de vários elementos para dinamizar a aprendizagem e tornar mais agradável a experiência do aluno no contexto de estudo de um curso EAD.

Pode-se observar que os conteúdos são abordados em vários formatos, como textos, imagens, vídeos, livros em *pdf* e conteúdos interativos que são *iframes*. Alguns dos elementos citados são representados por uma *tag HTML* (*Hypertext Markup Language*) customizada.



Figura 5: Apresentação do conteúdo



Fonte: Elaborado pelo autor, 2020.

Para realizar a renderização dos elementos que compõem as unidades de aprendizagem, como no exemplo mostrado na Figura 5, houve um desafio que era conseguir renderizar *tags* padrões do *html* e *tags* customizadas que vem da *API* e do *RealmDB*.

Para solucionar esse problema foi utilizada a biblioteca *react-native-render-html* que realiza a leitura desse conteúdo transformando-o em elementos reconhecidos pelo *React Native*. E para as *tags* customizadas, essa biblioteca permite que o desenvolvedor crie elementos específicos do *React Native* para que ela consiga realizar a transformação, assim fazendo com que o *React Native* consiga compilar o conteúdo e realizar a renderização da unidade de aprendizado.

## 5 CONSIDERAÇÕES FINAIS

Na seção anterior foi demonstrado o desenvolvimento do aplicativo e como pode-se observar ele atingiu o seu objetivo principal que era a criação de um aplicativo mobile desenvolvido em *React Native*. O aplicativo teria que permitir ao aluno realizar o estudo de sua unidade de aprendizagem com a possibilidade de acessar o conteúdo mesmo sem internet. Essa função de acesso offline foi disponibilizada por um botão de download de fácil acesso na tela das disciplinas.

Como melhorias futuras, pretende-se desenvolver alguns elementos específicos que, por terem uma demanda menor, não foram priorizados, que são os componentes de áudio e fórmulas. Outra melhoria seria a conclusão do elemento de vídeo para ser salvo durante o download do conteúdo e também ser disponibilizado offline. Um fluxo bastante interessante de ser desenvolvido no futuro é a criação de filas para as chamadas de APIs, pois quando a aplicação ficar offline ela armazenaria essas chamadas para serem realizadas em ordem quando a aplicação retomar sua

conexão. Assim, a universidade poderia acompanhar o progresso do aluno mesmo que a internet do aluno caia por alguns minutos.

O aplicativo acabou sendo uma boa alternativa de ambiente virtual de aprendizagem por ter a função de acesso offline, a qual muitos desses ambientes não fornecem. E para os alunos funciona como mais uma forma de acesso às unidades de aprendizagem, assim podendo concluir sua formação no ensino a distância com maior comodidade.

## REFERÊNCIAS

ABED. **Censo EAD.BR**: relatório analítico da aprendizagem a distância no Brasil. 2018. Disponível em: [http://abed.org.br/arquivos/CENSO\\_DIGITAL\\_EAD\\_2018\\_PORTUGUES.pdf](http://abed.org.br/arquivos/CENSO_DIGITAL_EAD_2018_PORTUGUES.pdf). Acesso em: 16 mar. 2020.

BATISTA, Natércia A. *et al.* **GitSED**: um conjunto de dados com informações sociais baseado no GitHub. *In: SBBB-Dataset Showcase Workshop*. 2017. p. 224-233.

DRUMOND, DOUGLAS. **Iniciação ao Realm**: o que é o Realm?. 2016. Disponível em: <https://medium.com/android-dev-br/iniciacao-ao-realm-b7624e380194/>. Acesso em: 19 out. 2020.

EaD, PRESENCIAL E SEMIPRESENCIAL: qual é a diferença? 2018. Disponível em: <https://www.educamaisbrasil.com.br/educacao/noticias/ead-presencial-e-semipresencial-qual-e-a-diferenca>. Acesso em: 29 mar. 2020.

FERNANDES, Diego. **Offline first**: como funciona e como aplicar no React Native?. 2018. Disponível em: <https://blog.rocketseat.com.br/offline-first-react-native/>. Acesso em: 29 mar. 2020.

MEC. **O que é educação a distância?**. 2009. Disponível em: <http://portal.mec.gov.br/escola-de-gestores-da-educacao-basica/355-perguntas-frequentes-911936531/educacao-a-distancia-1651636927/12823-o-que-e-educacao-a-distancia>. Acesso em: 31 mar. 2020.

MICHEL, MAYKON. **Optimistic UI**. 2020. Disponível em: <https://ezdevs.com.br/offline-first-com-react-native/>. Acesso em: 29 mar. 2020.

SANTOS, Barbara. **Dados e tendências da educação a distância no Brasil**. 2018. Disponível em: <https://blog.hotmart.com/pt-br/educacao-a-distancia-no-brasil/>. Acesso em: 29 mar. 2020.

SILVA, Rafael Silvério. **O que é o ambiente virtual de aprendizagem?**. 2020. Disponível em: <https://eadbox.com/o-que-e-ambiente-virtual-de-aprendizagem/>. Acesso em: 29 mar. 2020.

SOUZA, Lays Serpa. **Os dilemas na implementação do EAD em universidades frente à Covid-19**. 2020. Disponível em: <https://www.jota.info/opiniao-e-analise/artigos/os-dilemas-na-implementacao-do-ead-em-universidades-frente-a-covid-19-10042020/>. Acesso em: 21 abr. 2020.