

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

LÍGIA FALCÃO NOGUEIRA MIRANDA

**AMIGOS DO REI: UMA APLICAÇÃO WEB PARA ESTUDOS BÍBLICOS
INFANTIS**

GUARAPUAVA

2023

LÍGIA FALCÃO NOGUEIRA MIRANDA

**AMIGOS DO REI: UMA APLICAÇÃO WEB PARA ESTUDOS BÍBLICOS
INFANTIS**

King's Friends: A web application for children's bible studies

Projeto de Trabalho de Conclusão de Curso de Graduação apresentado à disciplina de Trabalho de Conclusão de Curso 1, do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná, Campus Guarapuava, como requisito parcial para obtenção do título de Tecnólogo em Tecnologia em Sistemas para Internet.

Orientador: Prof. Dr. Roni Fabio Banaszewski

GUARAPUAVA

2023



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

RESUMO

O presente trabalho aborda a temática da organização e gestão das aulas da Escola Bíblica Dominical (EBD) no âmbito do Ministério Infantil da Igreja Presbiteriana Bonsucesso (IPB) de Guarapuava. A EBD é um espaço de ensino cristão voltado para crianças de diferentes faixas etárias, e sua eficácia é essencial para a transmissão de valores e princípios religiosos. No entanto, o atual sistema de organização das aulas apresenta desafios significativos. As informações sobre escalas, materiais, e outros aspectos relevantes são compartilhadas por meio de ferramentas como WhatsApp e Google Drive. Isso tem levado a problemas de comunicação, confusão e dificuldades na preparação das aulas por parte dos professores. Para abordar essas questões, este projeto propõe o desenvolvimento de uma aplicação web que centraliza e simplifica as informações relacionadas à EBD. O sistema fornecerá uma interface intuitiva e acessível para professores, alunos e pais, visando melhorar a qualidade das aulas e a participação ativa dos pais na educação religiosa de seus filhos. O sistema permitirá aos professores acessar facilmente suas escalas e materiais necessários para as aulas. Além disso, oferecerá recursos multimídia, ideias de atividades e um espaço para compartilhar informações sobre as aulas realizadas. Através de uma área dedicada aos pais, eles poderão acompanhar o progresso de seus filhos na EBD e participar ativamente do processo educacional. Espera-se que esse projeto contribua para uma maior eficiência na organização das aulas da EBD, tornando o ensino mais eficaz e acessível para as crianças. Ao adotar um ambiente digital dedicado à educação religiosa, a Igreja Presbiteriana Bonsucesso busca fortalecer sua missão de transmitir valores e princípios cristãos às gerações mais jovens.

Palavras-chave: estudo bíblico; angular; laravel; infantil; educação.

ABSTRACT

This paper addresses the theme of organization and management of Sunday School classes within the Children's Ministry of the Presbyterian Church of Bonsucesso in Guarapuava. The Sunday School (SS) is a space for Christian education aimed at children of different age groups, and its effectiveness is essential for the transmission of religious values and principles. However, the current system for organizing classes presents significant challenges. Information regarding schedules, materials, and other relevant aspects is shared through tools such as WhatsApp and Google Drive. This has led to communication issues, confusion, and difficulties in lesson preparation for teachers. To address these issues, this project proposes the development of a web application that centralizes and simplifies information related to the SS. The system will provide an intuitive and accessible interface for teachers, students, and parents, aiming to improve the quality of lessons and active parental participation in the religious education of their children. The system will enable teachers to easily access their schedules and the materials required for classes. Additionally, it will offer multimedia resources, activity ideas, and a space to share information about completed lessons. Through a dedicated section for parents, they will be able to monitor their children's progress in the SS and actively engage in the educational process. It is expected that this project will contribute to greater efficiency in the organization of SS classes, making teaching more effective and accessible for children. By adopting a digital environment dedicated to religious education, the Presbyterian Church of Bonsucesso aims to strengthen its mission of conveying Christian values and principles to younger generations.

Keywords: bible study; angular; laravel; children; education.

LISTA DE FIGURAS

Figura 1 – Exemplo de tela inicial desenvolvido no Moodle	12
Figura 2 – Exemplos de telas desenvolvido no Moodle	13
Figura 3 – Os dez mandamentos para crianças no site Rosângela Trajano	13
Figura 4 – Exemplo de histórias no app Bíblia para Crianças	14
Figura 5 – Tela inicial no app Bíblia para Crianças	14
Figura 6 – Dinâmica do Scrum	16
Figura 7 – Quadro Kanban	17
Figura 8 – Ciclo do TDD	18
Figura 9 – Fluxo de Trabalho	20
Figura 10 – Tela de Login	29
Figura 11 – Tela Inicial	30
Figura 12 – Tela de Perfil	30
Figura 13 – Arquitetura do Banco de Dados	33

LISTA DE TABELAS

Tabela 1 – Histórias de Usuário (Ordenado por Peso	27
---	-----------

LISTA DE ABREVIATURAS E SIGLAS

Siglas

ACID	Atomicidade, Consistência, Isolamento e Durabilidade
CLI	Command Line Interface
EBD	Escola Bíblica Dominical
IPB	Igreja Presbiteriana Bonsucesso
MVC	Model-View-Controller
SQL	Structured Query Language
SS	Sunday School
TDD	Test Driven Development
VCS	Sistema de Controle de Versão

SUMÁRIO

1	INTRODUÇÃO	8
1.1	Justificativa	9
1.2	Objetivos	10
1.2.1	Objetivo geral	10
1.2.2	Objetivos específicos	11
2	TRABALHOS RELACIONADOS	12
2.1	Moodle	12
2.2	Rosângela Trajano	13
2.3	Bíblia para Crianças	14
3	REFERENCIAL TEÓRICO	15
3.1	Processo de Desenvolvimento	15
3.1.1	Método <i>Scrum</i>	16
3.1.2	Método Kanban	17
3.1.3	Desenvolvimento Orientado a Testes	18
3.2	Ferramentas de desenvolvimento	19
3.2.1	GitHub	19
3.2.2	Git e GitFlow	19
3.3	Tecnologias do lado do cliente	20
3.3.1	Angular	21
3.4	Tecnologias do lado do servidor	21
3.4.1	Laravel	21
3.4.2	Docker	22
3.4.3	MySQL	22
4	MATERIAIS E MÉTODOS	24
4.1	Materiais	24
4.2	Métodos	25
5	ANÁLISE E PROJETO DO SISTEMA	27
5.1	Análise do sistema	27
5.2	Prototipação das Telas	28
5.3	Modelagem do Banco de Dados	31

6	CONSIDERAÇÕES FINAIS	34
	REFERÊNCIAS	35

1 INTRODUÇÃO

O primeiro encontro da criança com o ambiente escolar desempenha um papel crucial no seu desenvolvimento integral, abrangendo aspectos psicológicos, físicos, intelectuais e sociais. Portanto, esse período na escola é essencial para o crescimento da criança, ajudando-a a aprimorar suas habilidades cognitivas e motoras. A Educação Infantil desempenha um papel fundamental ao permitir que a criança interaja socialmente além do círculo familiar, proporcionando-lhe a oportunidade de aprender a conviver e se relacionar em sociedade. Isso, por sua vez, contribui para o desenvolvimento de habilidades cruciais para sua formação como ser humano (LÜCKE, 2019).

No entanto, a escola não se limita a ser o único ambiente onde as crianças adquirem conhecimentos acadêmicos, como língua portuguesa e matemática. Antes mesmo de frequentar a escola, as crianças muitas vezes têm a oportunidade de aprender em outros ambientes, como creches e igrejas.

Ensinar crianças com idades entre zero e doze anos é um desafio significativo, mas instruí-las nos princípios cristãos de acordo com a Bíblia representa um desafio ainda maior. Os educadores das escolas dominicais nas igrejas evangélicas em todo o Brasil e no mundo se concentram nos conceitos fundamentais que moldarão a personalidade e o caráter dessas crianças. Eles buscam mantê-las envolvidas e concentradas, garantindo que a mensagem seja transmitida de maneira clara e simples, adaptada a cada faixa etária. O objetivo principal é tocar o coração das crianças, de modo que levem esses ensinamentos consigo ao longo de toda a vida (BERTINATTI; NASCIMENTO, 2011).

Conforme mencionado no livro de Provérbios, capítulo 22, versículo 6, é de extrema importância iniciar a educação religiosa desde tenra idade, para que esses princípios se tornem uma parte integral de suas vidas ao longo do tempo. Com o objetivo de atender às necessidades das diversas faixas etárias das crianças, as igrejas evangélicas introduziram o conceito de Escola Bíblica Dominical (EBD) especializadas, que se concentram em oferecer um ensino com linguagem adequada para cada grupo etário.

Ensine a criança no caminho em que deve andar, e ainda quando for velho não se desviará dele. (BÍBLIA. . . , 2017).

Na Igreja Presbiteriana Bonsucesso (IPB) de Guarapuava, por meio do seu Ministério Infantil conhecido como "Amigos do Rei", o ensino voltado para crianças é estruturado em quatro faixas etárias distintas. O primeiro grupo, chamado de "babies", abrange crianças de zero a três anos. O segundo, denominado de "primário", engloba crianças de quatro a seis anos. O terceiro grupo, intitulado "juniores", atende crianças de sete a nove anos. Por fim, o grupo de "pré-adolescentes" inclui crianças de dez a doze anos. Além disso, o ministério conta com uma equipe de teatro e diversas professoras que se revezam no atendimento às diferentes turmas. No entanto, devido à complexidade das atividades e ao envolvimento de muitas pessoas no

ministério, a forma atual de disponibilização das informações tem gerado confusões para alguns professores. Eles têm dificuldades em saber qual será a aula no próximo domingo, quais materiais estão disponíveis para conduzir a aula, quem estará responsável pelo ensino e para qual turma, além de como manter a continuidade e a conexão com o conteúdo ensinado pelo professor da semana anterior.

Com o intuito de contribuir para a organização eficaz dos cronogramas e programas de aula, bem como para melhorar a preparação das professoras, será abordado ao longo deste trabalho de conclusão de curso, o desenvolvimento de um sistema para auxiliar na organização da EBD. Este sistema será restrito, exigindo autenticação para acesso. Apresentará uma interface simplificada, abrangendo todos os conteúdos anuais e áreas multimídias, que incluem atividades, vídeos e músicas direcionadas a diversas faixas etárias. Além disso, oferecerá a funcionalidade de emitir relatórios e registrar o andamento das aulas diárias. Também, terá uma área especial para os pais dos alunos da EBD, onde ele poderá ver um *feed*¹ com resumos da aula do dia da turma do seu filho e atividades realizadas, para assim poder acompanhar em casa o que o filho aprendeu e também reforçar o que foi aprendido. Poderá ver também, se disponível, foto da turma com marca d'água, alertando que aquelas fotos não podem ser divulgadas em redes sociais, são apenas para métricas internas e para os pais poderem ver seus filhos em aula.

1.1 Justificativa

O Ministério atualmente utiliza um arquivo de texto no Word para registrar as datas dos domingos em que haverá aulas e compartilha-o por meio do Google Drive no grupo de WhatsApp dos professores do ministério infantil. Este arquivo contém a lista de domingos, geralmente abrangendo três meses, listados um abaixo do outro com o nome do professor(a) designado para cada turma. No entanto, esta não é a maneira mais prática de localizar suas escalas.

Quando ocorrem mudanças na escala, os professores as comunicam pelo WhatsApp. Contudo, com frequência, essas informações se perdem no meio do conteúdo das aulas e dos compartilhamentos de materiais, o que resulta em problemas como a dificuldade de substituir um professor ou, em último caso, a necessidade de juntar classes às pressas. Isso pode levar a situações em que os professores não estão devidamente preparados para assumir uma turma que precisa ser coberta.

Quanto a conteúdos e materiais, estes estão disponíveis em forma de imagens escaneadas da apostila usada nas aulas, armazenadas em um Google Drive². Além disso, algumas

¹ Ferramenta que permite facilitar o consumo de conteúdo pelo usuário, oferecendo na tela inicial conteúdos direcionados para o usuários para aquilo que lhe interessa (CARREIRO, 2022).

² Plataforma de compartilhamento de arquivos com opção de armazenamento em nuvem (GOOGLE, 2012).

atividades também são escaneadas para serem usadas durante as aulas. No entanto, existem algumas limitações notáveis, como:

- **Qualidade das Imagens:** A qualidade das imagens não é muito alta, o que pode prejudicar a legibilidade e utilidade do material.
- **Cores Limitadas:** As imagens são em preto e branco, com poucas opções de atividades ou diversificação adequada para atender a diferentes faixas etárias.
- **Fotos das Aulas:** As fotos das aulas são compartilhadas no grupo, o que aumenta a quantidade de conversas no grupo, consome espaço nos celulares devido às mídias e torna difícil a tarefa de categorizar as fotos em relação às classes a que se referem, ao dia em que foram tiradas e à compilação desse material completo para métricas de final de ano.

Devido às dificuldades em localizar informações no grupo, muitas vezes é necessário repetir detalhes, como a próxima lição a ser ministrada no domingo seguinte ou a lista de professores escalados para a semana em questão. Isso frequentemente ocorre em um momento muito próximo ao evento, tornando difícil realizar substituições, e já houve casos em que os professores não tinham conhecimento de sua inclusão na escala. Quando é necessário realizar pesquisas, agendar reuniões ou comunicar informações cruciais, muitas vezes não se consegue atrair a devida atenção, uma vez que não há uma maneira eficaz de destacar essas informações.

Visando corrigir estes problema, foi idealizado um sistema para centralizar todas as informações para aulas de maneira simplificada e organizada, uma área dedicada para avisos e alertas, em forma de landing page³ contendo um *feed* de informações e fotos organizadas por prioridade e data.

1.2 Objetivos

1.2.1 Objetivo geral

Desenvolver uma aplicação web para centralizar os dados da EBD voltada para crianças de zero a doze anos, visando facilitar a preparação eficiente de aulas pelos professores e aproximar os pais do conteúdo ministrado.

³ Definido como página de aterrissagem, de conversão, de captura ou de destino com objetivo de atrair o usuário com informações sobre o que existe no sistema e lhe trazer alertas direcionados (SEBRAE, 2022)

1.2.2 Objetivos específicos

- Facilitar a identificação da próxima data de escala e da respectiva turma na página inicial.
- Possibilitar a visualização de todas as datas relevantes para o usuário por meio de um calendário.
- Disponibilizar materiais para a próxima aula, incluindo dados da apostila, sugestões de atividades, vídeos e músicas.
- Permitir o compartilhamento de informações da aula realizada no domingo, como fotos e um breve resumo dos materiais utilizados.
- Oferecer uma seção multimídia com músicas, vídeos e ideias de atividades.
- Apresentar um feed de notícias na forma de uma "landing page" com as últimas aulas ministradas, alertas, enquetes importantes e informações gerais fornecidas pelos administradores.
- Gerar relatórios de desempenho das turmas.
- Disponibilizar uma área dedicada aos pais dos alunos.
- Promover um ambiente de fácil utilização, direcionando o grupo do WhatsApp exclusivamente para informações específicas, priorizando o sistema para todas as demais comunicações.

2 TRABALHOS RELACIONADOS

Atualmente, há uma variedade de sites e aplicativos que permitem o armazenamento de dados com o objetivo de centralizar conteúdos didáticos, organizados por disciplinas e professores. Além disso, existem algumas opções voltadas especificamente para conteúdo bíblico. Nos próximos tópicos, serão apresentados sistemas disponíveis no mercado que compartilham partes desta proposta, embora não sejam idênticos em sua abordagem.

2.1 Moodle

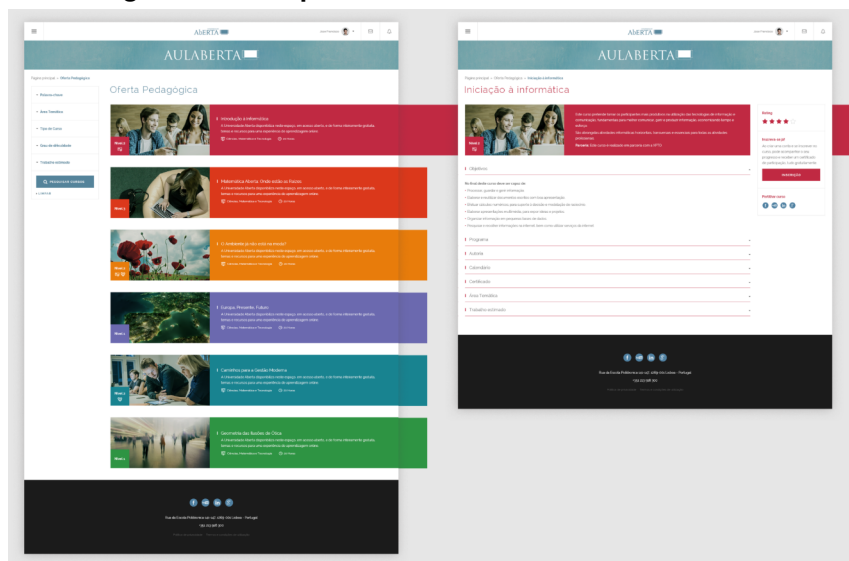
Utilizado no mundo inteiro por ser um código aberto multi-idioma, o Moodle é uma plataforma de ensino projetada para oferecer a educadores, administradores e alunos um sistema único, robusto e integrado para criar ambientes de aprendizagem personalizados. De fácil entendimento, uma vez que usa uma interface simples, recursos de arrastar e soltar para montagem com documentação abrangente e atualizada, juntamente com melhorias contínuas na usabilidade. Além disso, essa plataforma oferece a mais ampla variedade de ferramentas para apoiar tanto a aprendizagem presencial quanto cursos totalmente online, permitindo personalizar ativando ou desativando recursos básicos e integrando facilmente todos os elementos necessários para um curso, aproveitando sua ampla gama de recursos incorporados, que incluem ferramentas colaborativas externas, como fóruns, wikis, chats e blogs (MOODLEDOCS, 2021). A figura 1 exibe um exemplo de tela inicial no Moodle, enquanto a figura 2 mostra exemplos de telas também do Moodle.

Figura 1 – Exemplo de tela inicial desenvolvido no Moodle



Fonte: Adaptado de MoodleDocs (2021).

Figura 2 – Exemplos de telas desenvolvido no Moodle



Fonte: Adaptado de MoodleDocs (2021).

2.2 Rosângela Trajano

Dedicada ao ensino infantil, a página de Rosângela Trajano abriga uma ampla gama de materiais de estudo, abrangendo desde o primeiro até o nono ano e englobando disciplinas como filosofia, sociologia, artes, ensino religioso, antropologia, educação financeira, entre outras. Organizado em subcategorias, o site oferece sugestões de atividades, incluindo material para colorir, além de histórias escritas de forma acessível para garantir que as crianças compreendam o conteúdo em uma linguagem adaptada a elas (TRAJANO, 2023). A figura 3 apresenta 'Os Dez Mandamentos para Crianças' no site Rosângela Trajano.

A plataforma seria utilizada de referência

Figura 3 – Os dez mandamentos para crianças no site Rosângela Trajano

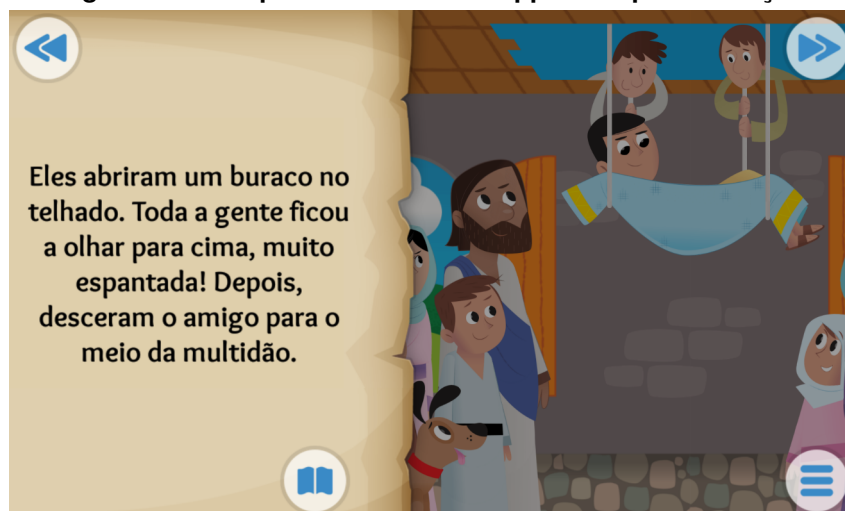


Fonte: Adaptado de Trajano (2023).

2.3 Bíblia para Crianças

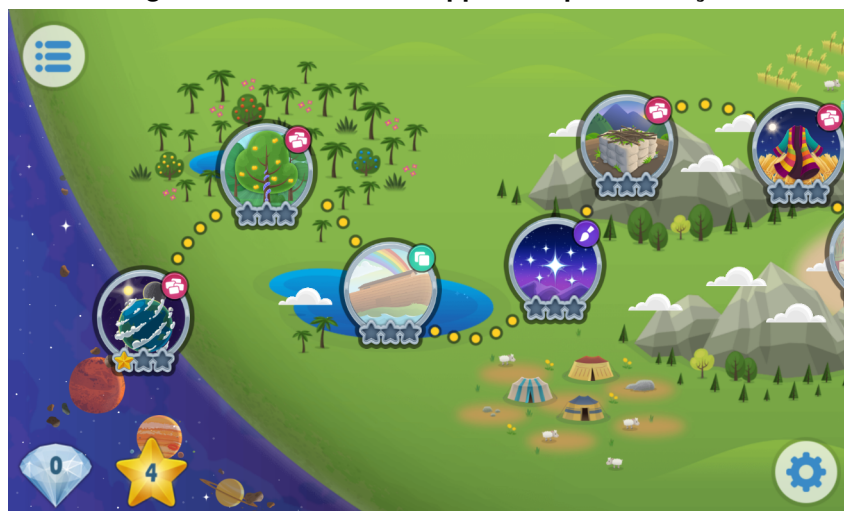
Do grupo YouVersion, o aplicativo Bíblia para Crianças está disponível para smartphones e tablets Android e IOS. É uma aplicação totalmente gratuita, disponível em várias línguas em todo o mundo. O objetivo desse aplicativo é apresentar as histórias da Bíblia de forma ilustrada e com linguagem simplificada para crianças, oferecendo aventuras interativas, animações, perguntas sobre a história e jogos que buscam proporcionar uma experiência agradável e cuidadosamente projetada. O objetivo é motivar as crianças a retornarem sempre, contribuindo para o desenvolvimento de um amor pela Palavra de Deus desde tenra idade (YOUVERSION, 2013). As figuras 4 e 5 ilustram exemplos de histórias e a tela inicial do aplicativo "Bíblia para Crianças."

Figura 4 – Exemplo de histórias no app Bíblia para Crianças



Fonte: Adaptado de YouVersion (2013).

Figura 5 – Tela inicial no app Bíblia para Crianças



Fonte: Adaptado de YouVersion (2013).

3 REFERENCIAL TEÓRICO

Para o desenvolvimento de uma aplicação web simples, organizada e destinada ao uso das professoras, com o objetivo de facilitar a condução de suas aulas e otimizar as operações gerais do Ministério Infantil, faz-se necessário realizar um levantamento do que será necessário e entender qual o foco final da aplicação. Neste caso, busca-se com a ferramenta não apenas beneficiar os líderes do Ministério, proporcionando uma visão unificada e simplificada das atividades realizadas ao longo dos anos, mas também apoiar as professoras na manutenção de uma sequência lógica de ensino. Isso é especialmente relevante devido à prática de rodízios de professoras a cada domingo, visando evitar a fadiga, permitindo assim uma continuidade coerente no desenvolvimento das aulas, além de oferecer ideias que facilitam o processo de ensino, centralizando o material necessário para a preparação das aulas das professoras. Mas acima de tudo, entregar as crianças que frequentam o Amigos do Reino uma experiência que permita com que elas absorvam da melhor maneira possível os ensinamentos de Deus para a vida delas.

Para viabilizar esta aplicação, é imprescindível contemplar as diversas ferramentas disponíveis no mercado, abrangendo linguagens de programação, frameworks e bibliotecas. A seleção criteriosa dessas tecnologias influenciará diretamente na qualidade e no prazo de implementação do sistema. Portanto, é de suma importância definir minuciosamente quais tecnologias serão adotadas, assegurando o êxito da aplicação e contribuindo positivamente para a educação cristã das crianças.

Nesse contexto, é crucial seguir boas práticas de programação e empregar técnicas que facilitem a manutenção do sistema, garantindo seu funcionamento contínuo ao longo do tempo. A busca por uma abordagem cuidadosa na escolha e implementação das tecnologias visa não apenas atender às necessidades imediatas, mas também estabelecer uma base sólida para futuras expansões e aprimoramentos, consolidando assim o impacto positivo desejado na experiência educacional das crianças.

Neste capítulo, serão exploradas tecnologias essenciais para impulsionar o desenvolvimento do projeto, abrangendo ferramentas e metodologias que serão empregadas ao longo da aplicação.

3.1 Processo de Desenvolvimento

O processo de desenvolvimento abrange todo o período de criação do sistema. Por mais que no desenvolvimento deste sistema tenha apenas um desenvolvedor, é crucial utilizar práticas e metodologias que auxiliem a organizar todas as etapas do projeto, para garantir que todas as partes importantes do sistema sejam desenvolvidas e entregues ao final da aplicação no tempo estipulado. Neste contexto, para que isso possa ser possível, serão utilizadas técnicas de gerenciamento e organização de um sistema.

3.1.1 Método Scrum

Apesar de método no nome, *Scrum* não se propõe a ser uma metodologia com processos padronizados onde segue-se de maneira metódica uma série de etapas sequenciais para que obtenha os resultados desejados. O *Scrum* é um *framework* ágil amplamente utilizado no desenvolvimento de software para gerenciar e organizar projetos complexos, proporcionando uma abordagem flexível e iterativa para o desenvolvimento de produtos, enfatizando a colaboração, a adaptabilidade e a entrega contínua de valor ao cliente. Ele divide projetos em períodos denominados como *sprints*, que duram algumas semanas e a definição de papéis como o *Product Owner*, *Scrum Master* e *Developers*. O *Product Owner* representa os interesses do cliente, logo, será o responsável em fornecer os requisitos do sistema juntamente com a ordem em que devem ser executados, além disso, ele quem irá avaliar a qualidades das entregas. O *Scrum Master* é o facilitador do processo, sendo assim, será o responsável de garantir com que o *Scrum* seja compreendido e aplicado de forma correta. Os *Developers* serão responsáveis em criar o produto seguindo os pedidos do cliente segundo a organização do *Scrum Master*. Para gestão de atividades, existem reuniões regulares, como o *Sprint Planning* para planejamento, o *Daily Scrum* para sincronização diária, o *Sprint Review* para mostrar o que foi feito e o *Sprint Retrospective* para avaliar e melhorar.

Figura 6 – Dinâmica do Scrum



Fonte: Adaptado de MindMaster (2023).

O *Scrum* utiliza listas de tarefas chamadas *Product Backlog* e o *Sprint Backlog*, que ajudam a priorizar o que precisa ser feito. O *Product Backlog* é uma lista em constante evolução, onde os itens podem ser adicionado, excluídos ou revisto pelo *Product Owner*, nela existem todas as funcionalidades, requisitos, melhorias e correções que precisam ser feitas ao produto, representando as demandas do cliente e também requisitos identificados pela equipe de desenvolvimento. O *Sprint Backlog* é uma lista dos itens do *Product Backlog* selecionados durante a

parte de planejamento para a *Sprint* que tem duração média de duas a quatro semanas, sendo o plano da entrega do produto ao longo do período, sendo exemplificada na figura 6 (MIND-MASTER, 2023).

3.1.2 Método Kanban

Implementado juntamente com o *Scrum*, o método Kanban é uma abordagem visual para gerenciar o trabalho, originalmente desenvolvida no Japão para otimizar a produção. Hoje em dia, é amplamente utilizado em diversos contextos, incluindo desenvolvimento de software, gerenciamento de projetos, e até mesmo em tarefas do dia a dia. Para criar um quadro Kanban, é utilizado um painel subdividido em colunas que representam os estágios do processo, tais como "A fazer", "Em andamento" e "Concluído", conforme ilustrado na Figura 7. Em cada coluna, as tarefas são representadas por cartões que incluem informações cruciais, como descrição e responsável. Esses cartões são movidos entre as colunas de acordo com o progresso do desenvolvimento, proporcionando uma visualização clara e dinâmica do fluxo de trabalho. Para evitar sobrecarga, cada coluna tem um limite de cartões que podem estar em andamento simultaneamente.

Figura 7 – Quadro Kanban



Fonte: Adaptado de Freitas (2022).

Com este processo, todos os membros da equipe podem visualizar facilmente o fluxo de trabalho, identificar gargalos e entender o que está sendo feito, além disso, é adaptável a mudanças e permite ajustes contínuos. Se algo não está funcionando, a equipe pode modificá-lo facilmente. Ao usar o Método Kanban, as equipes podem manter um fluxo de trabalho mais suave, priorizar tarefas de forma eficaz e responder rapidamente às mudanças nas demandas

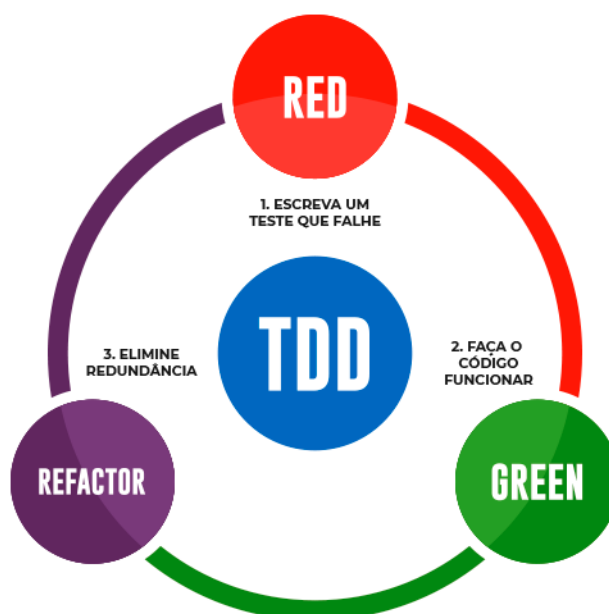
do projeto. É uma ferramenta valiosa para promover a colaboração e aumentar a eficiência em diversas áreas de trabalho (SABINO, 2023).

3.1.3 Desenvolvimento Orientado a Testes

O Desenvolvimento Orientado a Testes, do inglês Test Driven Development (TDD) é uma abordagem no processo de desenvolvimento de software que enfatiza a criação de testes automatizados antes da implementação real do código. Tem como principal objetivo garantir a qualidade do software desde o início do processo de desenvolvimento. Ao criar testes antes da implementação do código, os desenvolvedores buscam detectar precocemente erros, melhorar a estrutura do código, facilitar a manutenção futura e manter o foco nas necessidades do usuário.

A abordagem visa criar uma suíte de testes abrangente que promova a confiança na funcionalidade e estabilidade do software, permitindo mudanças e atualizações com maior segurança. O ciclo do TDD segue as seguintes etapas *Red-Green-Refactor*, como pode ser visualizado na figura 8. Na etapa *Red* os desenvolvedores escrevem um teste automatizado que inicialmente falha, representando a necessidade de uma nova funcionalidade ou alteração. Na fase *Green*, o código é implementado de maneira mínima para fazer o teste passar, garantindo que a funcionalidade desejada seja alcançada. Por fim, na fase *Refactor*, os desenvolvedores aprimoram o código sem alterar o comportamento, mantendo a integridade dos testes. Esse ciclo contínuo promove o desenvolvimento incrementado e aprimora a qualidade do código ao longo do tempo (GUEDES, 2019).

Figura 8 – Ciclo do TDD



Fonte: Adaptado de Guedes (2019).

3.2 Ferramentas de desenvolvimento

As ferramentas de desenvolvimento consistem em utilizar ferramentas apropriadas para aumentar a eficiência do desenvolvimento, reduzindo a necessidade de realizar tarefas manuais e entregando a organização e qualidade do sistema em construção. Neste contexto, para que isso possa ser possível, serão apresentadas ferramentas que facilitam o desenvolvimento de aplicações, contribuindo para organização, uso de boas práticas e manutenção do sistema.

3.2.1 GitHub

O *GitHub* é uma plataforma de desenvolvimento colaborativo baseado em nuvem que hospeda um Sistema de Controle de Versão (VCS) chamado *Git* que fornece um ambiente para que desenvolvedores possam colaborar em projetos de software, rastrear mudanças no código-fonte, gerenciar problemas e realizar diversas outras tarefas relacionadas ao desenvolvimento de software. Tem como principal funcionalidade permitir que os desenvolvedores armazenem seus projetos em repositórios públicos ou privados, facilitando, assim, trabalho em equipe, pois dessa forma diferentes desenvolvedores podem colaborar em um projeto simultaneamente, fazendo alterações que ficam todas registradas pelo *GitHub* (LONGEN, 2023b).

Algumas ferramentas de desenvolvimento integradas ao *GitHub* serão de suma importância para concretização do projeto, além de centralizar todas as ideias em apenas um local. Estas serão discutidas nos seguintes subtópicos.

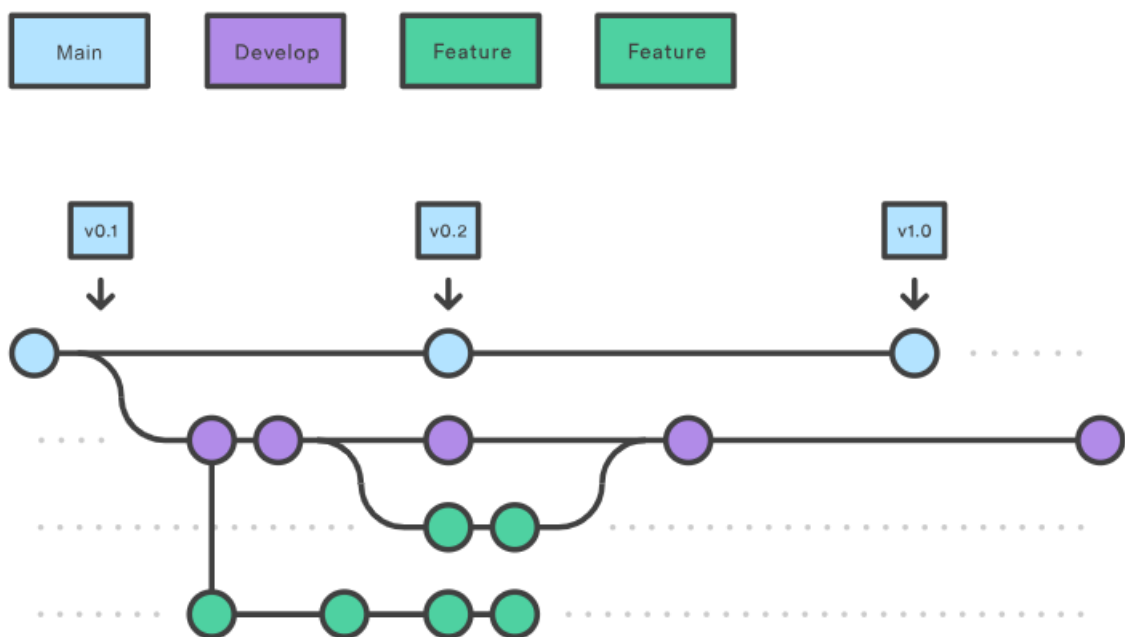
3.2.2 Git e GitFlow

O *Git* é um sistema de controle de versão distribuído amplamente utilizado no desenvolvimento de software, criado por Linus Torvalds em 2005 para ajudar no desenvolvimento do kernel do Linux, mas desde então tornou-se uma ferramenta essencial para projetos de software em geral. Por meio dele, qualquer desenvolvedor autorizado numa equipe pode gerenciar o código-fonte e seu histórico de mudanças usando ferramentas de linha de comandos, mas diferente dos sistemas de controle de versão centralizados, ele oferece ramificações de recursos, também conhecidos como *feature branches*, o que permite que cada desenvolvedor separe seu trabalho em ramificações locais para fazer suas mudanças no código (LONGEN, 2023b).

Para controle e uso dessas ramificações foi desenvolvido o modelo alternativo de ramificações denominado *Gitflow* que consiste no uso de ramificações de recursos e várias ramificações primárias, fornecendo um conjunto de diretrizes para organizar o fluxo de trabalho ao utilizar o *Git*, principalmente em projetos com um ciclo de vida mais complexo que venham a envolver desenvolvimento de novas funcionalidades, correções de bugs e manutenções. Comparado ao desenvolvimento baseado em troncos, o *Gitflow* tem mais ramificações de vida longa

e commits maiores, pois neste modelo, os desenvolvedores criam uma ramificação de recurso e retardam a mescla com a ramificação de tronco principal até que o recurso esteja completo. Esse fluxo trabalho é constituído por uma ramificação principal, denominada de *main*, onde armazena o histórico da versão oficial do sistema, seguido pelas ramificações de desenvolvimento, denominada de *develop*, que contém os códigos para a próxima versão do sistema. Cada nova funcionalidade deverá ser desenvolvida a partir da *develop* e quando a funcionalidade for concluída, deve ser feito a mesclagem, ou *merge*, com a *develop* e quando esta tiver recursos suficiente para ir para ramificação principal, ela deve ser bifurcada em uma ramificação de lançamento, denominada como *release*. Quando a mesma estiver pronta para ser lançada é mesclada com a *main*, mas se durante esse processo for preciso corrigir problemas com rapidez, são criadas ramificações denominadas de *hotfix*, sendo estes baseados na *main* já que reflete ao código em produção. Todo o ciclo de trabalho pode ser visto na figura 9 (ATLASSIAN, 2023).

Figura 9 – Fluxo de Trabalho



Fonte: Adaptado de Atlassian (2023).

3.3 Tecnologias do lado do cliente

O lado do cliente, caracterizado como *Frontend*, desempenha um papel crucial na criação da interface visual de uma aplicação, pois esse possibilita a interação entre o usuário e o sistema, constituída por elementos visuais como layouts, botões e elementos interativos. Além disso, o *Frontend* é responsável em interagir com o banco de dados por meios de requisições com o servidor para controle de informações, sendo para criar novos dados, editar ou

exibir dados já existentes. Neste contexto, para que isso possa ser possível, serão apresentadas ferramentas que permitam a criação de uma interface visual adequada para interação com o usuário.

3.3.1 Angular

Angular é um *framework* de desenvolvimento da web criado pelo Google, usado para construir aplicações web e mobile interativas e dinâmicas. Por utilizar a arquitetura Model-View-Controller (MVC), simplifica o desenvolvimento ao oferecer estruturas organizacionais, como componentes reutilizáveis e módulos, que facilitam a construção de interfaces de usuário complexas, além disso, oferece recursos como data binding, injeção de dependência e um sistema de roteamento integrado, o que permite aos desenvolvedores criar aplicações web robustas de forma mais eficiente. Na versão Angular, e não AngularJs, utiliza TypeScript, uma linguagem que contém todos os elementos e funcionalidades do JavaScript, mas com algumas funcionalidades a mais, como a tipagem estática, interfaces e outras construções que não estão presentes no JavaScript padrão. Com sua Command Line Interface (CLI), sendo uma ferramenta de linha de comando, os desenvolvedores podem gerenciar facilmente tarefas como compilação, teste e implantação, tornando o processo de desenvolvimento mais ágil e consistente (LONGEN, 2023a).

3.4 Tecnologias do lado do servidor

O lado do servidor, caracterizado como *Backend*, desempenha o papel de cérebro da aplicação, onde se torna o responsável por receber e executar processos por requisições via *API*, sendo responsável por armazenar os dados do sistema, manipular os dados armazenados, definir rotas, processos e tratamento de dados, além da parte de segurança do sistema. Neste contexto, para que isso possa ser possível, serão apresentadas ferramentas que permitam a realização desse tratamento de dados do sistema.

3.4.1 Laravel

O Laravel é um popular *framework* de desenvolvimento web em PHP que foi criado por Taylor B. Otwell e é conhecido por sua elegância e facilidade de uso, sendo projetado para tornar o desenvolvimento web mais eficiente e agradável, oferecendo uma variedade de recursos com um sistema robusto de gerenciamento de banco de dados e uma sintaxe expressiva denominada *Eloquent* para interação com o banco de dados, sendo um poderoso sistema de roteamento. Além disso, integra o *Composer* para gerenciamento de dependências facilitando a inclusão e atualização de bibliotecas de terceiros. Possui uma sintaxe limpa e recursos avançados que simplificam as tarefas comuns, além de incentivar boas práticas de desenvolvimento por

sua estrutura eloquente, promovendo a legibilidade e manutenção do código. É um *framework* com uma excelente curva de aprendizagem por ser fácil e aprender, além de poupar tempo por permitir que tarefas complexas sejam executadas em pouco tempo, acompanhado de uma documentação simples e direta frequentemente atualizada devido a sua comunidade ativa está sempre auxiliando (CARDOSO, 2023).

3.4.2 Docker

O Docker é uma plataforma de virtualização de contêineres que simplifica o processo de empacotamento, distribuição e execução de aplicativos em ambientes isolados, chamados contêineres. A containerização permite que vários aplicativos funcionem em diferentes ambientes complexos, pois cada contêiner contém todos os elementos necessários para executar uma aplicação. Isso possibilita que ela seja executada em qualquer ambiente que suporte o Docker, facilitando a criação e implantação de aplicativos de maneira rápida e consistente. Isso elimina as diferenças entre ambientes de desenvolvimento, teste e produção.

A arquitetura do Docker é composta por quatro componentes: Cliente Docker, Servidor Docker, Imagens do Docker e Registro do Docker. O cliente Docker é o principal componente para criar, gerenciar e executar aplicativos em contêiner, sendo o principal método de controle do servidor Docker. O servidor Docker, também conhecido como o daemon do Docker, aguarda as solicitações da *API REST* feitas pelo cliente Docker e gerencia imagens e containers.

As imagens do Docker consistem nas instruções dadas ao servidor Docker com os requisitos sobre como criar um contêiner Docker. Por fim, o registro do Docker é um aplicativo do lado do servidor de código aberto usado para hospedar e distribuir imagens. É extremamente útil para armazenar imagens localmente e manter controle total sobre elas (CHAN, 2023).

Com o Docker e sua arquitetura, os desenvolvedores podem criar contêineres a partir de imagens padronizadas, compartilhando facilmente suas aplicações e suas dependências com outros membros da equipe, com contêineres leves que inicializam rapidamente, otimizando os recursos do sistema. Essa flexibilidade e portabilidade fazem do Docker uma ferramenta valiosa para orquestração de aplicativos em ambientes de nuvem, permitindo escalabilidade eficiente e facilitando a integração contínua e a implantação contínua (CI/CD) (CHAN, 2023).

3.4.3 MySQL

O MySQL é um sistema código aberto de gerenciamento de base de dados relacional Structured Query Language (SQL), ou seja, ele armazena essas informações em tabelas separadas e as conecta com chaves, tendo sido desenvolvido e suportado pela Oracle, amplamente utilizado, conhecido por sua confiabilidade e desempenho. Permite aos desenvolvedores armazenar, organizar e recuperar dados de maneira eficiente. Com suporte a SQL, facilita a criação

e a manutenção de bancos de dados relacionais para uma variedade de aplicativos, desde sistemas simples até sistemas mais complexos (KINSTA, 2021).

Os recursos do MySQL incluem a capacidade de lidar com grandes volumes de dados, suporte a transações Atomicidade, Consistência, Isolamento e Durabilidade (ACID) para garantir a integridade dos dados, e a capacidade de ser integrado a várias linguagens de programação. A atomicidade é uma propriedade que garante que cada transação seja tratada como uma entidade única, ou seja, para a transação ter sucesso necessita que todas as operações a ser executadas passem com sucesso.

A consistência permite assegurar que uma transação somente leve o banco de dados se todos os dados forem válidos, mantendo a estabilidade do banco. O Isolamento permite que várias tabelas sejam lidas ou alteradas por vários usuários simultaneamente. A durabilidade garante que uma transação, uma vez executada, permanecerá neste estado mesmo que haja um problema grave no sistema (REIS, 2018).

4 MATERIAIS E MÉTODOS

4.1 Materiais

Para a organização e gestão do projeto, serão empregadas diversas ferramentas que desempenham papéis fundamentais na execução e no acompanhamento das tarefas.

A primeira delas é o *GitHub*, uma plataforma de controle de versão. Esse sistema de versionamento permitirá o acompanhamento e registro das alterações no código-fonte, facilitando a colaboração entre os membros da equipe. Além disso, o *Git Flow* será adotado como metodologia de trabalho, o que implica a criação de branches para desenvolvimento, revisão e implementação de novas funcionalidades. Essa abordagem contribuirá para a organização e rastreamento eficazes das alterações no projeto.

O *GitHub Projects* será a ferramenta para o gerenciamento de tarefas. Esse recurso permite a criação de painéis de tarefas, organizadas em colunas que representam etapas do projeto, tornando mais claro o andamento das atividades. O modo roadmap proporciona uma visão geral do progresso e das metas a serem alcançadas, auxiliando na definição de prioridades e prazos. Portanto, será usado o modo roadmap.

No processo de prototipagem, o Figma será empregado na criação de protótipos de telas. Essa ferramenta facilita a representação visual dos elementos que comporão o sistema, permitindo a visualização e a validação das interfaces de forma mais prática.

Para a modelagem do banco de dados, será adotado o MySQL Workbench, uma ferramenta que possibilita a criação de diagramas e a definição das tabelas e relações necessárias para o funcionamento do sistema. Isso proporcionará uma compreensão mais clara da estrutura de dados do projeto.

No desenvolvimento propriamente dito, para a parte frontend, o framework Angular será utilizado. Sua escolha se deve à capacidade de criar interfaces de usuário interativas e responsivas, essenciais para a usabilidade do sistema. No que diz respeito ao backend, será empregado o framework Laravel, que facilita o desenvolvimento de APIs e a integração com o banco de dados MySQL. O acesso ao banco será realizado por meio do MySQL Workbench, proporcionando uma interface amigável para gerenciar os dados.

Para garantir a consistência e compatibilidade dos componentes necessários, será desenvolvido um ambiente Docker. Isso assegurará que todas as partes do sistema estejam funcionando de forma adequada e estejam nas versões corretas.

O ambiente de desenvolvimento escolhido para a elaboração do código será o Visual Studio Code, uma ferramenta que oferece recursos essenciais para a programação, como depuração e extensões que auxiliam no desenvolvimento.

4.2 Métodos

Neste projeto, será adotada uma metodologia de desenvolvimento incremental, inspirada no Scrum. Embora o Scrum seja comumente utilizado por equipes, ele será adaptado para atender às necessidades de um projeto desenvolvido individualmente. O desenvolvimento contará com o apoio do orientador e reuniões regulares com os líderes do ministério infantil "Amigos do Rei". O orientador e os líderes do ministério assumirão papéis importantes na metodologia Scrum, atuando como Product Owners, enquanto o aluno desempenhará o papel de Developer. O Professor orientador também desempenhará o papel de Scrum Master.

O desenvolvimento das atividades será dividido em Sprints quinzenais, que representam ciclos de trabalho. Para o gerenciamento de tarefas e acompanhamento do progresso, será utilizado o recurso Projects do GitHub. Isso proporcionará uma visão geral do sistema, seus requisitos e o acompanhamento efetivo do que está sendo realizado, com graus definidos de importância para as tarefas.

Para identificação das tarefas, será adotado um número de ID único, seguindo um padrão de cinco dígitos. Esse ID será replicado na branch do projeto, facilitando a compreensão de qual tarefa está relacionada a cada commit. A estrutura da tarefa incluirá um título composto pelo ID e uma breve descrição do que deve ser feito. Em seguida, a tarefa será detalhada de forma clara, incluindo a prioridade atribuída para indicar a urgência de sua realização. As prioridades podem ser:

- **Normal:** tarefa que pode ser desenvolvida quando possível.
- **Importante:** tarefa que precisa ser desenvolvida sem tanta urgência.
- **Urgente:** tarefa que precisa ser desenvolvida de forma prioritária.
- **Bug:** tarefa que precisa ser resolvida urgentemente para corrigir falhas no sistema causadas por bugs.

Cada tarefa terá um tempo previsto para conclusão, alinhado com a carga horária semanal de 15 horas dedicadas ao projeto pelo aluno. A organização das tarefas em sprint iniciarão sendo subdividas e definidas dentro do Backlog para serem distribuídas a cada sprint seguindo o total de 30 horas, subdivido por listas identificadas pelo número da sprint e período definido. Cada coluna dessas views serão divididas em:

- **To Do:** tarefas que precisam ser desenvolvidas ao longo da sprint.
- **In Progress:** tarefa está em fase de desenvolvimento.
- **Review:** tarefa finalizada pronta para ser revisada ao fim da sprint e testada.
- **Done:** tarefa finalizada e aprovada.

Quanto à construção do projeto, será empregado o Docker para criar contêineres controlados, o que facilitará o desenvolvimento, evitará problemas de versionamento de componentes e permitirá manutenções mais eficientes. Os contêineres serão divididos em:

- **amigos-back** : responsável por executar o framework Laravel e com todos os componentes necessários para sua utilização nas versões corretas, onde estará concentrada toda a aplicação backend do sistema, como as APIs que serão consumidas, toda a lógica de tratamento dos dados enviados, além de segurança dos dados do sistema.
- **amigos-front** : responsável por executar o framework Angular e com todos os componentes necessários para sua utilização nas versões corretas, onde estará concentrada toda a aplicação frontend do sistema, como as telas que serão acessadas pelo cliente.
- **amigos-db** : responsável por executar o MySQL, onde serão armazenados todos os dados gerados pelos usuários que servem para a funcionalidade do sistema.

A adoção das metodologias e materiais descritos visam contribuir na organização e sucesso desse projeto, oferecendo um sistema funcional e de agrado do cliente, permitindo uma distribuição coerente de tarefas para que seguindo esse roteiro o sistema possa ser desenvolvido no tempo previsto e de fácil acompanhamento do desenvolvimento do sistema, permitindo ter tempo para testes e aprimoramentos.

5 ANÁLISE E PROJETO DO SISTEMA

5.1 Análise do sistema

Com base na metodologia *Scrum* explanada na seção 3.1.1, serão definidas as histórias dos usuários utilizando a estrutura "*COMO [persona], QUERO [funcionalidade], PARA [algum fim].*" e os critérios de aceitação.

Os níveis de acesso de usuários ao sistema são: Administrador, Professor e Pais. Assim, estes usuários podem utilizar recursos específicos de acordo com suas categorias, contribuindo assim para a melhoria do aprendizado das crianças participantes da EBD e proporcionando uma melhor interação entre professores e os pais no ensino bíblico das crianças como objetivo focal do sistema.

As Histórias do usuário são apresentadas na tabela 1 a seguir, formando o backlog do sistema. Sendo subdividida por ID referente a tarefa, o peso para representar o grau de prioridade e a descrição explicando as histórias do usuários.

Tabela 1 – Histórias de Usuário (Ordenado por Peso

ID	Peso	Descrição
1	5	COMO administrador eu QUERO cadastrar usuários PARA acessarem ao sistema
6	5	COMO professor eu QUERO acessar calendário PARA me programar para aula
2	4	COMO administrador eu QUERO inserir o cronograma PARA aulas dos professores
7	4	COMO professor eu QUERO acessar materiais de aulas PARA preparar aula
8	4	COMO professor eu QUERO documentar aula PARA mostrar o que foi feito
14	4	COMO administrador eu QUERO alterar cronogramas PARA quanto trocar professores
3	3	COMO administrador eu QUERO salvar apostila em aula PARA professores acessarem
9	3	COMO professor eu QUERO confirmar aula PARA mostrar que eu dei a aula marcada
11	3	COMO professor eu QUERO alterar perfil PARA definir senha e atualizar dados
15	4	COMO pai eu QUERO saber o que meu filho aprendeu PARA reforçar em casa
5	2	COMO administrador eu QUERO cadastrar eventos PARA professores acessarem
4	2	COMO administrador eu QUERO emitir relatórios PARA professores acessarem
10	2	COMO professor eu QUERO acessar grupo do WhatsApp PARA dúvidas e avisos
12	1	COMO professor eu QUERO enviar conteúdos PARA auxiliar a criar aulas por tema
13	1	COMO administrador eu QUERO aprovar conteúdos PARA disponibilizar a todos

Fonte: Autoria própria (2023).

Os pesos foram utilizadas a técnica de Moscow, que é uma técnica de priorização de requisitos que classifica os requisitos em quatro categorias, sendo elas *Must have* (Deve ter),

Should have (Deveria ter), *Could have* (Poderia ter) e *Won't have* (Não terá). No entanto, a categoria *Could have* foi subdividida com base no grau de necessidade mais elevado (PM3, 2022). A seguir, apresenta-se a distribuição desses pesos dividido de 1-5 onde 1 o menor grau de importância e o 5 o maior:

- **5 - *Must have* (Deve ter)**: sendo como requisito de maior grau de importância e precisa ter no sistema, algo indispensável.
- **4 - *Should have* (Deveria ter)**: sendo como requisito do sistema que deveriam ter no sistema, logo podem não ter a importância do anterior, mas está logo em seguida.
- **3 - *Could have* (Poderia ter)**: sendo um requisito do sistema que pode vir a ter mas sendo melhor que tenha do que o de peso 2.
- **2 - *Could have* (Poderia ter)**: sendo um requisito do sistema que pode vir a ter, mas menos importante que o de peso 3.
- **1 - *Won't have* (Não terá)**: sendo um requisito do sistema que não crítico para o sistema funcionar e caso não possa vir a ser implementado não seria realmente um problema.

5.2 Prototipação das Telas

A aplicação proposta nesse trabalho busca entregar ao usuário uma experiência de fácil utilização tanto para obter informações de aulas, materiais e cronogramas, como para fornecer dados de suas aulas que foram ministradas. Todo acesso ao sistema precisaria ser apenas por meio de login, onde este seria dividido pelas seguintes categorias:

- **Admin**: qualidade de administrador do sistema, tendo acesso a todas as áreas do sistema, além de áreas exclusivas que permitem com que ele possa gerar relatórios.
- **Professor**: o acesso de professor vem com mais uma categoria de restrição, sendo por turmas as quais ministra suas aulas.
- **Pais**: o acesso dos pais possibilita que eles entrem em uma página de *feed*, proporcionando uma experiência simplificada e direta para acompanhar o que o filho aprendeu no último domingo. Isso inclui a visualização de fotos da aula e a leitura de recados importantes.

Para efetuar o login, e também sendo a tela inicial do sistema, foi feito um protótipo com auxílio do Figma, como apresentado na seção 4.1 e pode ser visto na figura 10. Em caso de ao mesmo tempo ser professora e mãe de alguma criança que frequenta o ministério, ao login, o usuário escolhe qual tipo de perfil quer acessar.

Figura 10 – Tela de Login



Fonte: Autoria Própria (2023).

Em todos os níveis de acesso todos os usuários terão acesso ao *feed* de notícia, sendo essa a única tela de acesso no login dos pais, em formato de *landing page* com todas as informações em uma página só, permitindo que desça pela página para mais informações. Nela conterà as últimas fotos das aulas que foram dadas e as informações resumidas do conteúdo que foi dado, possibilitando ver mais para mais informações. Além disso, na parte superior em forma de carrossel de imagens, ficarão as informações principais, como enquetes, encontros, reuniões e eventos.

Ao realizar o login, tanto professoras quanto administradores têm acesso direto à tela inicial do sistema, onde encontram informações cruciais sobre as próximas aulas, conforme evidenciado nas Figuras 11 *a* e *b*. Essas informações incluem o dia da aula (retângulo azul), o tema a ser abordado (retângulo verde), a turma responsável pela aula (retângulo rosa) e o texto base para a próxima aula (retângulo lilás). Além disso, ambos têm acesso a um menu lateral (círculo com o logo e retângulo amarelo) para explorar outras áreas do sistema. Esse menu é flexível, podendo ser ocultado ou exibido conforme o clique do usuário.

Na barra de navegação (*navbar*), destacada na parte superior em degradê rosa nas Figuras 11 *a* e *b*, os usuários podem acessar a área de calendários, onde visualizam os cronogramas relacionados ao seu perfil. Para o administrador, o calendário exibirá informações de todos os professores. Adicionalmente, a barra de navegação oferece acesso à área de mensagens, direcionando para o *Whatsapp* atual das professoras. Isso facilita a comunicação em caso de trocas de escala, necessidade de assistência ou dúvidas. Os usuários também podem acessar a tela de perfil e efetuar logout do sistema. Clicar no logo sempre redireciona para a tela inicial, independentemente da parte do sistema em que estejam. As representações visuais dessas funcionalidades podem ser visualizadas nos protótipos das Figuras 11 *a* e *b*.

Figura 11 – Tela Inicial

(a) Sem Menu

(b) Com Menu



Fonte: Autoria Própria (2023).

Na tela de perfil, o usuário encontra informações sobre o usuário logado, como nome, foto, formas de contato e para qual turma ou turmas a professora está no cronograma para dar suas aulas. Na figura 12, representado pelo retângulo azul terão todas as informações da professora um breve comentário sobre quem é a professora, explicando se ela tem filhos do ministério e quais são suas faixas etárias que mais se identifica, além de informar se tem alguma limitação ou auxilia em mais partes do ministério, como em cultos noturnos, teatro ou música. No retângulo lilás, terão informações resumidas das últimas aulas dadas pela professora em questão.

Figura 12 – Tela de Perfil



Fonte: Autoria Própria (2023).

Além disso, através do menu terá acesso a telas de multimídias, onde professores irão encontrar vídeos, atividades e músicas como materiais de auxílios para ajudar professoras a dar suas aulas e até fazer a hora da música inicial. Terão também acesso para a página da apostila, nela terá todas as aulas da apostila separadas por temas, onde dentro terá os capítulos

respectivos àquele tema e qualquer arquivo auxiliar. Outra página será a de registro de aula, onde a professora depois de dar a aula, ela nessa página registra qual foi a aula que ela deu naquele dia, dá uma breve explicação do que foi feito, incluindo atividade e anexa fotos dos alunos em aula. Todas essas fotos receberão marcas d'água e apenas o administrador terá acesso a essas fotos sem marcas para respeito de imagem das crianças e professores. Pelo menu lateral, também terá acesso ao *feed* explicado posteriormente nesta seção.

5.3 Modelagem do Banco de Dados

A modelagem do banco de dados para esse sistema pode ser visualizado na figura 13, onde consiste nas seguintes tabelas principais:

- **users:** tabela que irá conter todos os usuários do sistema, com suas informações de contato, como nome, e-mail e telefone, tipo de login (administrativo, professor ou pais), status de ativo ou inativo, além de login e senha para acessar o sistema.
- **children:** tabela que irá conter o nome da criança e sua data de nascimento para determinar sua turma vigente.
- **classes:** tabela que irá conter as divisões de classes, sendo esta uma tabela de valores já pré-determinados como *babies*, primário, juniores e pré-adolescentes, contendo quais idades pertencem a essa turma e se o turno é da manhã ou da noite.
- **schedule:** tabela que irá conter as aulas e programações do ministério, sendo subdivida em tipos (aula, reunião, confraternização e eventos externos), data que irá acontecer, para qual usuário representa aquela programação e qual classe se refere.
- **multimedia:** tabela que irá conter os arquivos de multimídia do banco referente a arquivos de aula e relacionados com temas definidos, contendo o nome, url, tipos de arquivos (documento, vídeo ou foto), usuário que adicionou e seu status (aprovado, reprovado e aguardando).
- **themes:** tabela que irá conter os temas que serão encaixados os arquivos de multimídia, sendo definida apenas por seu id e nome.

As tabelas de conexão auxiliares serão:

- **professor-has-class:** tabela de conexão de 1 para n de professores com classes, onde um professor pode estar em uma ou mais classes diferentes.
- **parent-has-children:** tabela de conexão entre os pais e seus filhos em uma conexão n para n, uma vez que os dois pais podem ter acesso e ter mais de um filho.

- **schedule-has-photos**: tabela que conecta a programação com fotos que serão inseridas para demonstrar o que aconteceu, sendo uma conexão um para n, onde cada programação pode ter de uma a n fotos.
- **theme-has-multimedia**: tabela de conexão de n para n de temas com os arquivos adicionados, onde um tema pode ter diversos arquivos multimídia e um arquivo multimídia pode ter mais de um tema.
- **class-has-multimedias**: tabela de conexão de 1 para n de classe com arquivos multimídia, onde uma classe pode ter um ou vários arquivos multimídia de referência.

Por fim, a tabela de histórico será:

- **log**: tabela que irá conter tudo que aconteceu no sistema, gerando histórico que será utilizado para gerar relatórios no sistema sobre tudo que aconteceu ao longo do período. Contendo descrição do que aconteceu, qual tipo de ação foi feita (adicionar, editar, deletar, trocar, dentre outras), qual tabela se refere a informação e qual o id nessa tabela se refere, além disso, que usuário realizou essa ação.

6 CONSIDERAÇÕES FINAIS

A solução tecnológica desenvolvida ao longo deste projeto possibilitará que o Ministério Infantil da IPB organize o trabalho realizado pelos professores e membros do teatro para as crianças que participam do programa "Amigos do Rei". Isso tornará mais fácil o planejamento das aulas, o desenvolvimento de atividades criativas e a manutenção de uma agenda de aulas organizada e de fácil acesso. Além disso, a capacidade de documentar as aulas com fotos das atividades realizadas e seus temas permitirá aos pais que tenham acesso ao sistema acompanhar o que foi ensinado aos seus filhos, promovendo um ensino contínuo do que foi abordado em sala de aula.

Além de solucionar o problema atual de organização na EBD, o sistema terá como principal objetivo possibilitar que as crianças tenham aulas mais direcionadas, facilitando uma aprendizagem mais sólida e concreta, independentemente de suas idades. Isso permitirá que elas cresçam e se tornem adultos que seguem esses princípios, trilhando os caminhos de Deus e evitando desvios em sua fé.

REFERÊNCIAS

- ATLASSIAN. **Saiba tudo sobre o Gitflow Workflow**. 2023. Disponível em: <https://www.atlassian.com/br/git/tutorials/comparing-workflows/gitflow-workflow>. Acesso em: 17 nov. 2023.
- BERTINATTI, N.; NASCIMENTO, E. Fraga Vilas-Bôas Carvalho do. **Os modelos de organização das escolas dominicais presbiterianas no Brasil: o ideal e o possível**. 2011. Disponível em: https://www.snh2011.anpuh.org/resources/anais/14/1300850691_ARQUIVO_TRABALHOCOMPLETOANPUH2011.pdf. Acesso em: 08 out. 2023.
- BÍBLIA Sagrada. Nova Almeida atualizada. [S.l.]: Sociedade Bíblica do Brasil, 2017.
- CARDOSO, R. **Laravel: Conheça o framework PHP mais utilizado hoje em dia!** 2023. Disponível em: <https://www.locaweb.com.br/blog/temas/codigo-aberto/laravel-framework-php/>. Acesso em: 17 nov. 2023.
- CARREIRO, A. P. **O que é um feed?** 2022. Disponível em: <https://apublicacao.com.br/o-que-e-feed/>. Acesso em: 08 out. 2023.
- CHAN, D. **O Que é Docker e Como Ele Funciona? – Docker Explicado**. 2023. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-docker>. Acesso em: 17 nov. 2023.
- FREITAS, A. V. **Kanban: o método de produção da Toyota que se tornou aliado da gestão de projetos**. 2022. Disponível em: <https://www.promadjr.com/post/kanban-o-m%C3%A9todo-de-produ%C3%A7%C3%A3o-da-toyota-que-se-tornou-aliado-da-gest%C3%A3o-de-projetos>. Acesso em: 17 nov. 2023.
- GOOGLE. **Google Drive**. 2012. Disponível em: <https://www.google.com/intl/pt-BR/drive/>. Acesso em: 08 out. 2023.
- GUEDES, M. **Afinal, o que é TDD?** 2019. Disponível em: <https://www.treinaweb.com.br/blog/afinal-o-que-e-tdd>. Acesso em: 17 nov. 2023.
- KINSTA. **O Que é MySQL? Uma Explicação Simples para Quem Está Começando**. 2021. Disponível em: <https://kinsta.com/pt/base-de-conhecimento/o-que-e-mysql/>. Acesso em: 17 nov. 2023.
- LONGEN, A. S. **O Que É Angular: O Framework Que Transformará Seu Código**. 2023. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-angular>. Acesso em: 17 nov. 2023.
- LONGEN, A. S. **O Que é GitHub, Para Que Serve e Como Usar**. 2023. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-github>. Acesso em: 17 nov. 2023.
- LÜCKE, N. C. F. S. A importância do estímulo no desenvolvimento da criança. **Revista Científica Multidisciplinar Núcleo do Conhecimento**, São Paulo, v. 12, n. 06, p. 33–44, jun 2019. ISSN 2448-0959.
- MINDMASTER. **Scrum: A Metodologia Ágil Explicada de forma Definitiva**. 2023. Disponível em: <https://mindmaster.com.br/scrum/>. Acesso em: 17 nov. 2023.
- MOODLEDOCS. **Sobre o Moodle**. 2021. Disponível em: https://docs.moodle.org/all/pt_br/Sobre_o_Moodle. Acesso em: 08 out. 2023.

PM3. **Método MoSCoW: framework para ajudar a priorizar tarefas**. 2022. Disponível em: https://www.cursospm3.com.br/blog/metodo-moscow-framework-para-priorizar-tarefas/?gad_source=1&gclid=CjwKCAiA-P-rBhBEEiwAQEXhHyRKjkfx4Owjvd9E7zITLOVxVBgJ89BHESM78bf2DTWLXYBB-hyKhoCR7BwE. Acesso em: 15 dez. 2023.

REIS, F. dos. **Conceitos de Bancos de Dados – O que significa ACID**. 2018. Disponível em: <http://www.bosontreinamentos.com.br/bancos-de-dados/conceitos-de-bancos-de-dados-o-que-significa-acid/>. Acesso em: 17 nov. 2023.

SABINO, R. **Kanban: o que é, o Método Kanban, principais conceitos e como funciona no dia a dia**. 2023. Disponível em: <https://www.alura.com.br/artigos/metodo-kanban>. Acesso em: 17 nov. 2023.

SEBRAE. **O que é uma landing page**. 2022. Disponível em: <https://sebrae.com.br/sites/PortalSebrae/artigos/o-que-e-uma-landing-page,3e1e097399323810VgnVCM100000d701210aRCRD#:~:text=Uma%20landing%20page%20tem%20como,de%20captura%20ou%20de%20destino>. Acesso em: 08 out. 2023.

TRAJANO, R. **Material didático de estudo bíblico para crianças**. 2023. Disponível em: <https://rosangelatrajano.com.br/material-didatico-de-estudo-biblico-para-criancas/>. Acesso em: 08 out. 2023.

YOUVERSION. **Bíblia para Crianças**. 2013. Disponível em: https://play.google.com/store/apps/details?id=com.bible.kids&hl=pt_BR&gl=US&pli=1. Acesso em: 08 out. 2023.