



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE  
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**IGOR SANTA RITTA SEABRA**

**MEMORIZAPP: UM SISTEMA DE APRENDIZADO BASEADO EM  
REPETIÇÃO ESPAÇADA**

**CAMPINA GRANDE - PB**

**2023**

**IGOR SANTA RITTA SEABRA**

**MEMORIZAPP: UM SISTEMA DE APRENDIZADO BASEADO EM  
REPETIÇÃO ESPAÇADA**

**Trabalho de Conclusão Curso apresentado ao  
Curso Bacharelado em Ciência da Computação  
do Centro de Engenharia Elétrica e Informática  
da Universidade Federal de Campina Grande,  
como requisito parcial para obtenção do título  
de Bacharel em Ciência da Computação.**

**Orientador: João Arthur Brunet Monteiro**

**CAMPINA GRANDE - PB**

**2023**

**IGOR SANTA RITTA SEABRA**

**MEMORIZAPP: UM SISTEMA DE APRENDIZADO BASEADO EM  
REPETIÇÃO ESPAÇADA**

**Trabalho de Conclusão Curso apresentado ao  
Curso Bacharelado em Ciência da Computação  
do Centro de Engenharia Elétrica e Informática  
da Universidade Federal de Campina Grande,  
como requisito parcial para obtenção do título  
de Bacharel em Ciência da Computação.**

**BANCA EXAMINADORA:**

**João Arthur Brunet Monteiro  
Orientador – UASC/CEEI/UFCG**

**José Antão Beltrão Moura  
Examinador – UASC/CEEI/UFCG**

**Francisco Vilar Brasileiro  
Professor da Disciplina TCC – UASC/CEEI/UFCG**

**Trabalho aprovado em: 28 de JUNHO de 2023.**

**CAMPINA GRANDE - PB**

## **RESUMO**

Considerando a relevância da disponibilidade de ferramentas inovadoras e acessíveis para um aprendizado eficaz e contínuo, este artigo descreve o MemorizApp, um Sistema de Repetição Espaçada (SRS) gratuito e de código aberto. Baseado em cartões de memorização digitais personalizados, seu objetivo é auxiliar os usuários a memorizar conteúdos de maneira eficiente, utilizando um algoritmo de agendamento de revisões baseado na técnica da repetição espaçada. A aplicação visa superar as limitações de outras opções disponíveis no mercado, oferecendo uma variedade de tipos de teste para os cartões e buscando simplificar o uso da ferramenta. Algumas melhorias e funcionalidades adicionais são sugeridas para versões futuras do sistema.

# **MEMORIZAPP: A SPACED REPETITION-BASED LEARNING SYSTEM**

## **ABSTRACT**

Considering the relevance of the availability of innovative and accessible tools for effective and continuous learning, this article describes MemorizApp, a free and open-source Spaced Repetition System (SRS). Based on customised digital flashcards, its objective is to assist users in efficiently memorising content by utilising a scheduling algorithm based on the spaced repetition technique. The application aims to overcome the limitations of other options available in the market by offering a variety of card test types and simplifying the tool's usability. Several improvements and additional features are suggested for future versions of the system.

# MemorizApp: um sistema de aprendizado baseado em repetição espaçada

Igor Santa Ritta Seabra  
Departamento de Sistemas e Computação  
Universidade Federal de Campina Grande  
Campina Grande, Paraíba, Brasil  
igor.seabra@ccc.ufcg.edu.br

## RESUMO

Considerando a relevância da disponibilidade de ferramentas inovadoras e acessíveis para um aprendizado eficaz e contínuo, este artigo descreve o MemorizApp, um Sistema de Repetição Espaçada (SRS) gratuito e de código aberto. Baseado em cartões de memorização digitais personalizados, seu objetivo é auxiliar os usuários a memorizar conteúdos de maneira eficiente, utilizando um algoritmo de agendamento de revisões baseado na técnica da repetição espaçada. A aplicação visa superar as limitações de outras opções disponíveis no mercado, oferecendo uma variedade de tipos de teste para os cartões e buscando simplificar o uso da ferramenta. Algumas melhorias e funcionalidades adicionais são sugeridas para versões futuras do sistema.

## PALAVRAS-CHAVE

Aprendizado, memorização, repetição espaçada, cartões de memorização.

## REPOSITÓRIO

<https://github.com/igorseabra4/MemorizApp>

## 1. INTRODUÇÃO

O aprendizado constitui em uma mudança, seja ela temporária ou permanente, na estrutura cognitiva do ser humano. É um processo dinâmico que resulta da interação da pessoa com o meio e se manifesta na incorporação de novos conhecimentos, resultando em mudança na conduta ou desenvolvimento de novas habilidades [19]. Na sociedade contemporânea, é essencial, em diversas áreas da vida pessoal, profissional e acadêmica, que o aprendizado se dê de forma eficaz e contínua para acompanhar as demandas em constante evolução.

No entanto, os métodos tradicionais de aprendizado podem ser limitados em sua eficácia, já que não levam em conta as diferenças individuais no processo de aprendizagem [12]. Nesse contexto, ganha um papel relevante a disponibilidade de métodos e ferramentas inovadores e acessíveis para auxiliar no aprendizado.

Um dos principais objetivos do estudante é reter em memória, da melhor forma possível, o conteúdo estudado. Entretanto, a curva do esquecimento, modelo desenvolvido inicialmente pelo pesquisador alemão Hermann Ebbinghaus, mostra que a maior parte do conhecimento recém-adquirido é esquecido nos primeiros momentos após a aprendizagem, caso não seja utilizado ou revisado [13]. No gráfico da figura 1, temos em vermelho uma representação da capacidade de recordação de uma informação à medida que o tempo passa a partir de seu aprendizado.

Conforme evidenciado pelo modelo, a técnica de aprendizagem "em massa", que consiste em estudar uma quantidade grande de informações em um período de tempo reduzido [16], mostra-se

ineficiente na retenção de conteúdos a longo prazo. Isso ocorre porque qualquer material estudado de maneira intensa e concentrada, mesmo que seja possível de ser recordado no curto espaço de tempo subsequente, tende a ser esquecido ao longo do tempo.

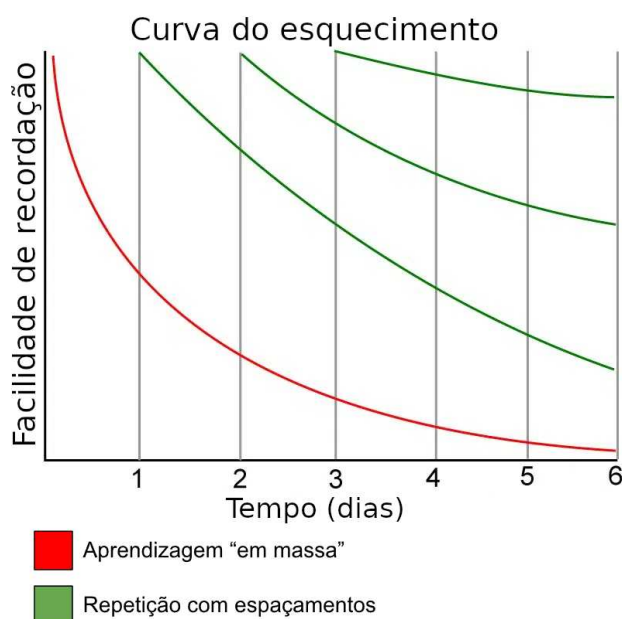


Figura 1 - Curva do esquecimento [18].

A repetição espaçada é uma técnica baseada na premissa de que, para que haja melhor retenção do conhecimento, devem ser feitas revisões do conteúdo estudado, distribuídas ao longo do tempo, com intervalos progressivamente espaçados. Essa abordagem busca encontrar parâmetros que auxiliem na determinação dos momentos ideais para realizar as revisões, lembrando o estudante de um determinado tópico no momento em que ele está mais propenso a esquecê-lo, buscando assim garantir que o conteúdo se consolide a longo prazo em sua memória. No gráfico da figura 1, temos em verde uma representação para a capacidade de recordação de uma mesma informação, caso seja revisada em intervalos espaçados. É notável que, a cada revisão subsequente, a facilidade de recordação diminui mais lentamente. Com base nisso, fica evidente a possibilidade de utilizar ferramentas computacionais em apoio à técnica, como aplicativos que aplicam a repetição espaçada ao estudo de *flashcards*, ou cartões de memorização.

Um cartão de memorização é tradicionalmente feito em papel, mas neste artigo trabalharemos com o cartão em sua forma digital. Em cada cartão, é escrita uma informação que se deseja aprender, seguindo o formato de uma pergunta sobre o conteúdo na frente e sua resposta no verso. Ao estudar, o estudante lê a pergunta na frente do cartão, tenta respondê-la mentalmente,

verifica a resposta no verso e, em seguida, avalia o quão fácil foi acertar aquela resposta, ou se a resposta estava errada. Essa avaliação, juntamente com outros parâmetros que serão discutidos mais à frente, alimenta um algoritmo que determina quando esse cartão deve ser estudado novamente.

Esses conceitos servem como fundamento para a criação de um Sistema de Repetição Espaçada (ou SRS, sigla em inglês para *Spaced Repetition System*), um software aplicativo que tem como objetivo ser útil aos estudantes de uma área em que as técnicas de repetição espaçada e cartões de memorização sejam aplicáveis. O SRS auxilia no processo de memorização do conteúdo, permitindo que o estudante foque no aprendizado, abstraindo questões relacionadas ao algoritmo de agendamento de revisões, que, sem o uso do software, ficariam a cargo do estudante [17].

Neste trabalho, foi concebido o protótipo de um sistema de repetição espaçada denominado MemorizApp. O objetivo principal desse SRS é fornecer aos usuários uma ferramenta acessível que permite a criação e estudo de coleções personalizadas de cartões, utilizando as técnicas descritas para estimular o aprendizado eficiente e eficaz. Além disso, o SRS busca oferecer recursos adicionais, como uma ampla variedade de tipos de teste para os cartões de memorização, que serão detalhados posteriormente. O sistema foi projetado com uma abordagem direta e intuitiva, reduzindo a necessidade do usuário lidar com configurações ou outras funcionalidades relacionadas ao seu funcionamento. Vale ressaltar que MemorizApp é um sistema de código aberto e gratuito, diferenciando-se dos aplicativos similares disponíveis no mercado. O artigo abordará a motivação que levou ao desenvolvimento do sistema, descreverá suas funcionalidades, lógica de uso e arquitetura, discutirá considerações sobre seu desenvolvimento e possibilidades para o futuro desta aplicação e outras similares.

## 2. MOTIVAÇÃO

Existem algumas opções, disponíveis no mercado, de aplicativos de aprendizado que utilizam repetição espaçada, cartões de memorização e/ou conceitos similares. Entretanto, o interesse na criação de uma nova aplicação surgiu a partir da necessidade de determinadas características que não estão presentes em algumas das opções existentes analisadas.

A principal dessas características é a possibilidade de ter diversos tipos de teste para o conteúdo dos cartões. A forma tradicional de revisão, conforme descrita anteriormente, consiste em ler a pergunta na frente do cartão, responder mentalmente e atribuir um nível de dificuldade ao cartão com base na facilidade de obtenção da resposta (iremos nos referir a esse processo como "teste tradicional" ou "forma tradicional de teste" neste artigo). No entanto, essa maneira de avaliar o conhecimento da informação pode ser falha, pois é possível que o usuário não tenha certeza sobre a dificuldade da resposta da pergunta ou forneça avaliações imprecisas [6].

Especialmente considerando as possibilidades oferecidas pelo ambiente virtual, existem outras formas de testar o conhecimento. Algumas possibilidades são tipos de questão de teste comumente usados em âmbito escolar e acadêmico, como múltipla escolha, preenchimento de lacunas e associação de colunas [3, 7]. Com essas possibilidades no sistema, os usuários podem customizar sua experiência de aprendizado de acordo com suas preferências, além de obter um resultado mais preciso na avaliação do conteúdo dos cartões.

Uma aplicação de repetição espaçada com cartões de memorização popular atualmente é o Anki [4], opção gratuita e de código aberto. Com versões para web, desktop e dispositivos

portáteis, ele permite ao usuário criar suas próprias coleções de cartões e estudá-los de acordo com parâmetros personalizados, além de disponibilizar uma plataforma online para que os usuários compartilhem suas coleções. O Anki possui alta possibilidade de customização tanto das configurações do aplicativo como dos cartões em si, entretanto, essa diversidade de opções torna a curva de aprendizado do uso da aplicação alta para alguns usuários iniciantes [15]. Além disso, ele possui apenas a forma tradicional de teste.

Outra plataforma similar existente é o Memrise [8], uma plataforma de aprendizado que utiliza os princípios da repetição espaçada e da gamificação. Focada primariamente no estudo de idiomas, opera em um modelo de negócio fechado e pago, embora também disponha de uma versão gratuita com funcionalidades limitadas. Ela permite que o usuário crie seus próprios cartões de memorização e possui como uma característica diferencial o estudo através da geração automática de alguns tipos de testes para os cartões, incluindo opções de resposta em múltipla escolha e solicitação para digitar a resposta.

Diante desse panorama, nossa proposta é desenvolver um novo sistema que aprimore a experiência do usuário por meio da capacidade de criação de cartões de memorização personalizados e uma ampla variedade de opções de teste em um sistema de fácil uso e a implementação de um algoritmo de repetição espaçada eficiente.

## 3. SOLUÇÃO

O MemorizApp é uma aplicação web de acesso gratuito por meio de qualquer dispositivo através do navegador. Qualquer pessoa pode criar uma conta, e os usuários registrados podem criar, gerenciar e estudar coleções de cartões de memorização de diversos tipos, que podem ser organizados em lições dentro das coleções. Nesta seção, veremos uma descrição da implementação e lógica de funcionamento da aplicação.

Durante o uso da aplicação, os cartões podem passar por diferentes estados, que serão detalhados na seção 3.1. Em seguida, na seção 3.2, abordaremos os diversos tipos de cartões disponíveis no sistema, ou seja, os tipos de testes que podem ser aplicados. Na seção 3.3, veremos uma descrição da lógica de uso da aplicação. O algoritmo de agendamento de revisões usando a técnica de repetição espaçada e sua implementação serão discutidos na seção 3.4. Por fim, na seção 3.5, está descrita a arquitetura do sistema.

### 3.1 Estados de cartão

Esses são os estados nos quais um cartão pode estar:

#### 3.1.1 Novo

Esse é o estado que um cartão recebe ao ser criado. Cartões "novos" são cartões que ainda não foram aprendidos ou estudados e não possuem revisões agendadas. Eles não aparecerão em uma sessão de estudo.

#### 3.1.2 Aprendendo

Esse é o estado que um cartão recebe ao ser movido para estudo. Em uma sessão de estudo, os cartões nesse estado sempre aparecem primeiro, até que não haja mais nenhum. Cartões nesse estado possuem um contador, que inicia em 5. Ao ser respondido corretamente, o valor do contador diminui em 1. Ao ser respondido incorretamente, o valor do contador aumenta em 2, para um máximo de 10. O cartão cujo contador chega em 0 é movido para o estado "revisando" e sua primeira revisão é agendada para um valor padrão de duas horas após o horário em que o cartão foi respondido.

### 3.1.3 Revisando

Um cartão em estado "revisando" já foi aprendido pelo usuário e possui uma revisão agendada para uma determinada data e horário. Caso esse momento já tenha passado, o cartão torna-se disponível para revisão em uma sessão de estudo. Durante a sessão de estudo, o cartão respondido corretamente terá sua revisão reagendada de acordo com o algoritmo de repetição espaçada (descrito na seção 3.4). O cartão respondido incorretamente será movido para o estado "falhado". Os cartões em "revisando" aparecem na sessão de estudo quando não há mais nenhum cartão em "aprendendo".

### 3.1.4 Falhado

Um cartão recebe o estado "falhado" quando possuía o estado "revisando" e é respondido incorretamente durante a sessão de estudo. Os cartões em "falhado" são apresentados na sessão de estudo por último, ou seja, quando não há mais nenhum cartão em "aprendendo" e nenhum cartão "revisando" passou de seu horário de revisão. Cartões falhados respondidos incorretamente continuam com o estado de falhado; cartões falhados respondidos corretamente são movidos de volta para o estado "revisando" e sua revisão é agendada para o momento presente, ou seja, serão revisados imediatamente.

### 3.1.5 Suspenso

O estado "suspenso" é atribuído aos cartões que o usuário deseja excluir das sessões de revisão, independentemente de seu estado anterior. Os cartões "suspensos" não serão exibidos durante uma sessão de estudo. O usuário pode mover ou remover um cartão desse estado na tela de edição de coleção.

Ao remover a condição de suspenso de um cartão, o cartão que estava previamente no estado "revisando" será movido de volta para esse estado, com a data previamente agendada para revisão mantida. Cartões previamente nos estados "novo", "aprendendo" e "falhado" serão movidos para o estado "novo".

## 3.2 Tipos de cartão

O sistema MemorizApp possui como um diferencial a diversidade de tipos de cartão, o que permite que o conhecimento do usuário seja testado de várias formas. O resultado desses testes alimentará o algoritmo de repetição espaçada, descrito na seção 3.4. Esses são os tipos de cartão disponíveis:

### 3.2.1 Frente e verso

O cartão "frente e verso" é o tipo tradicional de cartão de memorização. Na figura 2, podemos ver o formulário de edição do cartão e seus campos, "frente" e "verso":

Escolha o tipo de cartão:  
Frente e verso

Frente:  
Estados Unidos da América

Verso:  
Washington, D.C.

Salvar Apagar

Figura 2 - Formulário de edição do cartão "frente e verso".

Durante a sessão de estudo, ao ser testado, a frente do cartão é apresentada ao usuário (figura 3):

Capitais do Mundo

Frente

Estados Unidos da América

Revelar

Figura 3 - Cartão "frente e verso" durante estudo.

É neste momento, ao ler a frente do cartão, que o usuário deve respondê-la mentalmente. A resposta é revelada ao pressionar o botão "revelar", como mostra a figura 4.

Capitais do Mundo

Frente

Estados Unidos da América

Verso

Washington, D.C.

Errei Acertei (Difícil) Acertei (Ok) Acertei (Fácil)

Figura 4 - Cartão "frente e verso" durante estudo com resposta revelada.

Com a resposta revelada, o usuário deve indicar se errou, ou no caso de ter acertado, avaliar o quão fácil foi para ele acertar a resposta. Essa avaliação existe apenas para cartões com status "revisando"; para cartões "aprendendo" e "falhado", ele deve indicar apenas se a resposta estava correta ou não.

### 3.2.2 Resposta por escrito

O cartão "resposta por escrito" funciona de forma similar ao cartão "frente e verso", com a diferença de que, no teste, o usuário deve digitar a resposta do cartão, e a resposta é considerada correta apenas se ele digitar a resposta correta. Essa determinação é feita ignorando a distinção entre letras maiúsculas e minúsculas. Na figura 5, vemos os campos "pergunta" e "resposta" no formulário de edição.

Escolha o tipo de cartão:  
Resposta por escrito

Pergunta:  
México

Resposta:  
Cidade do México

Salvar Apagar

Figura 5 - Formulário de edição do cartão "resposta por escrito".

Na figura 6, vemos o cartão durante a sessão de estudo. A resposta correta é exibida depois de clicar em "Enviar".

Capitais do Mundo

Pergunta

México

Sua Resposta

Resposta

Enviar

Figura 6 - Cartão "resposta por escrito" durante estudo.

Na figura 7, vemos o cartão com a resposta correta revelada.



Figura 7 - Cartão "resposta por escrito" após revelação da resposta.

### 3.2.3 Múltipla escolha

O cartão "múltipla escolha" permite ao usuário digitar uma pergunta ou proposição e uma lista de alternativas. A figura 8 mostra o formulário de edição para este tipo de cartão.

No formulário, o botão "+" permite adicionar uma nova alternativa. O botão "-" ao lado de cada alternativa permite removê-la. A caixa de seleção ao lado de cada alternativa define se ela é correta ou não. A caixa de seleção com o rótulo "Ordem aleatória" define se, durante o teste, as alternativas serão exibidas em ordem aleatória ou na ordem que foram inseridas no cartão.

Figura 8 - Formulário de edição do cartão "múltipla escolha".

Na figura 9, vemos o cartão durante a sessão de estudo. Para a resposta ser considerada correta, o usuário deve selecionar exatamente a alternativa correta (ou alternativas, caso haja mais de uma correta). As alternativas corretas são reveladas depois de clicar em "Enviar".

Na figura 10, vemos o cartão após o usuário clicar em "Enviar". As respostas corretas são mostradas à direita das respostas do usuário.

Figura 9 - Cartão "múltipla escolha" durante estudo.

Figura 10 - Cartão "múltipla escolha" exibindo a resposta correta.

### 3.2.4 Associar colunas

O tipo de cartão "associar colunas" permite que o usuário insira pares de itens em duas colunas, esquerda e direita. Existe uma associação entre os dois itens de uma mesma linha, que deve ser reproduzida pelo usuário durante a sessão de estudo. Veja o formulário de edição desse tipo na figura 11: o botão "+" permite adicionar um novo item e o botão "-" ao lado de cada item permite removê-lo.

Figura 11 - Formulário de edição do cartão "associar colunas".

Na figura 12, podemos ver o cartão "associar colunas" durante a sessão de estudo. Os itens são exibidos em ordem aleatória e, à direita, dois botões com setas (uma para cima e uma para baixo) permitem reordenar a coluna da direita.

**Capitais do Mundo**

Associe as capitais de alguns dos países continentais da América Central.

Honduras	Cidade da Guatemala	↕
Panamá	São José	↕
Nicarágua	Cidade do Panamá	↕
Costa Rica	Manágua	↕
Guatemala	São Salvador	↕
El Salvador	Tegucigalpa	↕

**Figura 12 - Cartão "associar columnas" durante estudo.**

Após clicar em "Enviar", as respostas corretas são reveladas. Os itens com respostas corretas ganham um fundo verde, e os itens com respostas erradas um fundo vermelho. A resposta só é considerada correta com uma pontuação de 100%, ou seja, se todos os itens tiverem sido associados corretamente, como mostra a figura 13.

**Capitais do Mundo**

Associe as capitais de alguns dos países continentais da América Central.

**Resposta**

Honduras	São Salvador	Tegucigalpa
Panamá	Cidade do Panamá	Cidade do Panamá
Guatemala	Cidade da Guatemala	Cidade da Guatemala
El Salvador	São José	São Salvador
Costa Rica	Tegucigalpa	São José
Nicarágua	Manágua	Manágua

Pontuação: 50,00%  
✘ Errou!

**Figura 13 - Cartão "associar columnas" após revelação das respostas.**

### 3.2.5 Verdadeiro ou falso

O tipo de cartão "verdadeiro ou falso" permite que o usuário insira uma afirmação e defina se ela é verdadeira ou falsa através da caixa de seleção no formulário, como mostra a figura 14:

Escolha o tipo de cartão:

Verdadeiro ou falso

**Afirmação:**

O assassinato do arquiduque austríaco Francisco Ferdinando é considerado um dos principais eventos que levaram à Primeira Guerra Mundial.

Verdadeiro:

**Figura 14 - Formulário de edição do cartão "verdadeiro ou falso".**

Durante a sessão de estudo, a afirmação é mostrada ao usuário, que deve responder se é verdadeira ou falsa (figura 15). A resposta correta é revelada posteriormente (figura 16).

**Afirmação**

O assassinato do arquiduque austríaco Francisco Ferdinando é considerado um dos principais eventos que levaram à Primeira Guerra Mundial.

**Figura 15 - Cartão "verdadeiro ou falso" durante o estudo.**

**Afirmação**

O assassinato do arquiduque austríaco Francisco Ferdinando é considerado um dos principais eventos que levaram à Primeira Guerra Mundial.

**Resposta**

Verdadeiro

Acertou!

**Figura 16 - Cartão "verdadeiro ou falso" com a resposta revelada.**

### 3.2.6 Ordenação de parágrafos

O cartão "ordenação de parágrafos" permite ao usuário digitar uma lista de parágrafos ou frases, como mostra a figura 17. O botão "+" permite adicionar um novo item e o botão "-" ao lado de cada item permite removê-lo.

Escolha o tipo de cartão:

Ordenar parágrafos

Proposta (pode ser vazio):

Ordene os acontecimentos históricos abaixo em ordem cronológica.

Parágrafos:

Criação da imprensa de Gutenberg

Proposta da reforma protestante por Martinho Lutero

Invenção do telescópio por Galilei

**Figura 17 - Formulário de edição do cartão "ordenar parágrafos".**

Durante a sessão de estudo, os itens são exibidos inicialmente em ordem aleatória, como mostra a figura 18. À esquerda de cada item, existem dois botões com setas (uma para cima e uma para baixo) para reordenar os itens. Ao clicar em "Enviar", a resposta só é considerada correta se os itens estiverem na ordem correta. A resposta correta é revelada posteriormente (figura 19).

Ordene os acontecimentos históricos abaixo em ordem cronológica.

↕ Invenção do telescópio por Galilei

↕ Criação da imprensa de Gutenberg

↕ Proposta da reforma protestante por Martinho Lutero

**Figura 18 - Cartão "ordenar parágrafos" durante o estudo.**

Ordene os acontecimentos históricos abaixo em ordem cronológica.

Criação da imprensa de Gutenberg  
  Proposta da reforma protestante por Martinho Lutero  
  Invenção do telescópio por Galilei

**Resposta**

1. Criação da imprensa de Gutenberg

2. Proposta da reforma protestante por Martinho Lutero

3. Invenção do telescópio por Galilei

Acertou!

Continuar

Figura 19 - Cartão "ordenar parágrafos" após revelação da ordem correta.

### 3.3 Lógica da aplicação

No protótipo desenvolvido, o acesso é livre para qualquer usuário e requer criação de uma conta com nome de usuário e senha.

#### 3.3.3 Registro e login

Ao entrar no sistema pela primeira vez, o usuário vê a seguinte tela (figura 20):

MemorizApp Página Inicial Login Cadastro

Você não está logado. Por favor [faça login](#) ou [crie uma conta](#).

Figura 20 - Página inicial (sem usuário logado).

Ao clicar no botão "Cadastro", o usuário é direcionado para a tela de cadastro, onde pode criar uma conta com nome de usuário e senha (figura 21):

Nome de usuário:

Nome de usuário

Senha:

Senha

Cadastrar

Figura 21 - Formulário de cadastro.

O usuário que já possui cadastro pode realizar login clicando no botão "Login", onde será redirecionado para a tela de login (figura 22):

Nome de usuário:

Nome de usuário

Senha:

Senha

Login

Figura 22 - Formulário de login.

Após inserir credenciais válidas e submeter o formulário, o usuário será redirecionado para a tela de listagem de coleções, conforme mostra a figura 23. O usuário já logado pode

encontrar no cabeçalho o botão "Logout", que permite ao usuário sair de sua conta.

#### 3.3.2 Listagem e criação de coleções

Na tela de listagem de coleções (figura 23), o usuário pode ver em uma tabela todas as suas coleções e a quantidade de cartões que a coleção possui no momento, em um agrupamento que informa ao usuário o que ele pode fazer com a coleção naquele momento.

MemorizApp Coleções Logout

Nome	Cartões (estudar agora)	Cartões	Cartões (novos)	Cartões (em revisão)	Cartões (suspensos)	
<a href="#">Coleção de teste</a>	15	16	1	15	0	<a href="#">Editar</a>
<a href="#">Capitais do Mundo</a>	0	14	13	0	1	<a href="#">Editar</a>

Adicionar Coleção

Figura 23 - Tabela de listagem de coleções.

- Estudar agora: somatório da quantidade de cartões nos estados "aprendendo", "falhado" e "revisando" cuja data de revisão já passou. É a quantidade de cartões que o usuário tem para estudar caso inicie a sessão de estudo naquele momento.
- Cartões: total de cartões daquela coleção.
- Novos: cartões no estado "novo".
- Em revisão: cartões no estado "revisando".
- Suspensos: cartões no estado "suspense".

Clicando em "Adicionar Coleção", o usuário tem acesso ao formulário para criar uma nova coleção, definindo seu nome (figura 24).

Cancelar

Nome da coleção:

Nome da coleção

Adicionar

Figura 24 - Formulário para adicionar nova coleção.

Ao clicar no nome da coleção, o usuário será redirecionado para a tela de estudo da coleção, que detalharemos na seção 3.3.6. O botão "Editar" leva à tela de edição daquele item da tabela, que veremos agora na seção 3.3.3.

#### 3.3.3 Edição de coleção

MemorizApp Coleções Logout

Capitais do Mundo [Editar](#)

Configurações

Estudar em ordem aleatória:

Novos cartões por vez (padrão):

Salvar

Figura 25 - Página de edição de coleção.

Na tela de edição de coleção, o usuário pode editar o nome e as configurações da coleção, assim como as lições e cartões daquela coleção, como mostra a figura 25.

O botão "Editar" abre o formulário para edição da coleção. Nele, é possível mudar o nome ou apagar a coleção, como visto

na figura 26. Apagar a coleção removerá todas as suas lições e cartões.

Nome da coleção: Capitais do Mundo

Salvar Apagar Cancelar

Figura 26 - Edição de coleção.

O formulário de configurações da coleção, visto na figura 25, permite editar duas configurações:

- Estudar em ordem aleatória: habilitada por padrão.
  - Quando ativada, os cartões na sessão de estudo serão exibidos em uma sequência aleatória, mas ainda seguindo a organização pelos status de "aprendendo", "revisão" e "falhado", nessa ordem.
  - Quando inativa, os cartões serão mostrados na ordem definida nesta mesma tela (ver seção 3.3.4).
- Novos cartões por vez: o valor é 5 por padrão. Esse número define a quantidade de cartões que serão movidos de "novo" para "aprendendo" por vez, por padrão, na tela de estudo da coleção (vista na seção 3.3.6).

Configurações

Estudar em ordem aleatória:

Novos cartões por vez (padrão): 5

Salvar

Figura 27 - Configurações da coleção.

### 3.3.4 Criação e edição de lição

Abaixo do formulário de configurações da coleção, o usuário pode ver a lista de lições daquela coleção (figura 28). Cada coleção é iniciada com uma lição, por padrão, e o usuário tem a liberdade de adicionar ou remover quantas lições desejar. As lições são uma forma do usuário agrupar os cartões de uma coleção em grupos menores a seu próprio critério, podendo ter um texto introdutório para explicação do conteúdo daquela lição.

Lição #4: Lição de exemplo 4  
Exemplo de texto descritivo da lição.  
Esta lição ainda não possui cartões.  
Adicionar Cartão Editar

Lição #5: Lição de exemplo 5  
Esta lição ainda não possui cartões.  
Adicionar Cartão Editar

Adicionar Lição

[Voltar](#)

Figura 28 - Lista de lições, mostrando as duas últimas lições de uma coleção.

No exemplo mostrado na figura 28, a lição número 5 possui um texto descritivo, e a lição 6 não. Não foram adicionados ainda cartões nestas lições.

Ainda nesta tela, podemos ver que as lições possuem dois botões com setas, um com a seta para cima e um com a seta para baixo, ao lado de seu nome, para que o usuário possa reordenar as lições a seu critério. O botão para mover a lição

para cima não está disponível na primeira lição e o botão para mover a lição para baixo não está disponível na última lição.

O botão "Adicionar Lição", no final da lista de lições, mostra o formulário para adicionar uma nova lição ao final daquela coleção, como mostra a figura 29.

Nome da lição: Nome da lição

Descrição da lição: Descrição da lição

Adicionar Cancelar

Figura 29 - Formulário de criação de lição.

O botão "Editar", ao lado do nome da lição, abre o formulário de edição da lição (figura 30) para que o usuário possa definir o nome e a descrição daquela lição, ou apagar a lição, removendo assim todos os seus cartões.

Lição #5

Nome da lição: Lição 5

Descrição da lição: Exemplo de texto descritivo c

Salvar Apagar Cancelar

Figura 30 - Formulário de edição de lição.

O botão "Adicionar Cartão", presente ao final de cada lição, abre o formulário de adição de cartão (figura 31), que veremos em seguida na seção 3.3.5.

O botão "Voltar", no final da página, redireciona o usuário de volta para a sua lista de coleções.

### 3.3.5 Criação e edição de cartões

Escolha o tipo de cartão:

Tipo... (dropdown menu)

- Associação de colunas
- Frente e verso
- Múltipla escolha
- Ordenar parágrafos
- Resposta por escrito
- Verdadeiro ou falso

Cancelar

Figura 31 - Formulário de criação e edição de cartão.

A figura 31 mostra o estado inicial do formulário de criação ou edição de cartões. Nele, é possível escolher qual será o tipo daquele cartão e, feita a seleção, quais serão os valores dos campos, que variam de acordo com o tipo, como descrito anteriormente na seção 3.2.

Podemos ver, na figura 32, um exemplo de lição com 3 cartões. A lição possui uma descrição e cada cartão é mostrado em uma linha de uma tabela.

Lição #1: América do Norte

Nesta lição, estudaremos as capitais dos países da América do Norte. Editar

#	Tipo	Cartão	
1	Frente e verso	<p><b>Frente</b> Estados Unidos da América</p> <p><b>Verso</b> Washington, D.C</p>	<span>Editar</span> <span>Suspender</span>
2	Resposta por escrito	<p><b>Pergunta</b> México</p> <p><b>Resposta</b> Cidade do México</p>	<span>Editar</span> <span>Suspender</span>
3	Múltipla escolha	<p><b>Pergunta</b> Canadá</p> <p><b>Opção 1:</b> X (incorreta) Toronto</p> <p><b>Opção 2:</b> W (correta) Ottawa</p> <p><b>Opção 3:</b> X (incorreta) Montreal</p> <p><b>Opção 4:</b> X (incorreta) Vancouver</p> <p><b>Ordem aleatória</b>  <input checked="" type="checkbox"/> Sim</p>	<span>Editar</span> <span>Suspender</span>

Adicionar Cartão

Figura 32 - Tabela de cartões em uma lição.

Na primeira coluna da tabela, é mostrado o número do cartão nesta coleção. Essa ordenação de cartões definirá a ordem na qual os cartões serão movidos de "novo" para "aprendendo" na tela de estudo da coleção (mostrada na seção 3.3.6) e, caso a configuração "Estudar em ordem aleatória" esteja desativada para a coleção, também será a ordem em que os cartões serão estudados ou revisados durante a sessão de estudo. Ainda nesta coluna, cada cartão possui dois botões com setas, um com a seta para cima e um com a seta para baixo, para que o usuário possa reordenar os cartões a seu critério. O botão para mover o cartão para cima não está disponível no primeiro cartão e o botão para mover o cartão para baixo não está disponível no último cartão.

Na segunda coluna, vemos o tipo daquele cartão.

Na terceira coluna, vemos os campos daquele cartão. Quais campos estão presentes em um cartão depende de seu tipo.

Na quarta coluna, temos os botões "Editar", que abre o formulário de edição daquele cartão (figura 31), e "Suspender", que move o cartão para o estado "suspenso" (ver seção 3.1.5). O cartão que está suspenso no momento possui o texto "Suspenso" nesta coluna e um botão "Retornar", que remove o estado "suspenso" do cartão, como vemos na figura 33:

<span>↑</span> 7 <span>↓</span>	Frente e verso	<p><b>Frente</b> Portugal</p> <p><b>Verso</b> Lisboa</p>	<span>Editar</span> <span>Retornar</span> <span>Suspenso</span>
---------------------------------------	----------------	--	---

Figura 33 - Cartão suspenso.

### 3.3.6 Estudo da coleção

Na tabela de listagem de coleções (vista na seção 3.3.2), clicar no nome da coleção levará o usuário à tela de estudo da coleção, vista na figura 34.

O texto "Não há cartões para estudar no momento" é exibido quando não há cartões na coleção nos estados "aprendendo" ou "falhado", ou no estado "revisando" com uma data de revisão agendada que já tenha passado. Isso significa que todos os cartões da coleção estão nos estados "novo" ou "suspenso", ou no estado "revisando" com uma data de revisão futura.

Na caixa intitulada "Quantidade para mover para estudo", cujo valor padrão é definido nas configurações da coleção (consulte a seção 3.3.3, figura 27), o usuário pode definir a quantidade de

cartões que serão movidos do estado "novo" para "aprendendo" ao clicar nos botões "Mover para estudo" de cada lição.

MemorizApp Coleções Logout

**Capitais do Mundo**

Não há cartões para estudar no momento.

Quantidade para mover para estudo:  
5

---

**Lição #1: América do Norte (3 cartões novos)**

Mover 3 para estudo

Nesta lição, estudaremos as capitais dos países da América do Norte.

---

**Lição #2: América Central (2 cartões novos)**

Mover 2 para estudo

---

**Lição #3: Europa (7 cartões novos)**

Mover 5 para estudo

Figura 34 - Tela de estudo da coleção sem cartões para estudar.

Abaixo, temos a lista das lições da coleção. O indicador de cartões novos mostra a quantidade de cartões no estado "novo" presentes em cada lição, que é a quantidade máxima que pode ser movida para estudo.

Os botões "Mover para estudo" em cada lição movem a quantidade específica de cartões daquela lição do estado "novo" para "aprendendo", o que permite que sejam estudados na sessão de estudo. A quantidade máxima a ser movida será determinada na caixa "quantidade para mover para estudo" mencionada acima, sendo limitada pela quantidade de cartões novos presentes naquela lição.

A figura 35 mostra a mesma coleção da figura 34, após o botão "Mover para estudo" ser pressionado uma vez em cada uma das três lições mostradas.

MemorizApp Coleções Logout

**Capitais do Mundo**

Estudar

Quantidade para mover para estudo:  
5

---

**Lição #1: América do Norte (0 cartões novos)**

Nesta lição, estudaremos as capitais dos países da América do Norte.

---

**Lição #2: América Central (0 cartões novos)**

---

**Lição #3: Europa (2 cartões novos)**

Mover 2 para estudo

Figura 35 - Tela de estudo da coleção com cartões para estudar.

Quando há cartões para estudar, ou seja, há cartões na coleção nos estados "aprendendo" ou "falhado", ou no estado "revisando" com uma data de revisão agendada que já tenha passado, torna-se disponível o botão "Estudar", que inicia uma sessão de estudo de cartões.

Na sessão de estudo, os testes dos cartões disponíveis para estudo serão exibidos sequencialmente para o usuário. O formato do teste depende do tipo do cartão, como visto na seção 3.2.

Observe que a sessão de estudo inicia automaticamente ao entrar nesta página, caso existam cartões para estudar no momento.

### 3.4 Algoritmo de repetição espaçada

Nesta seção, iremos fornecer uma visão aprofundada sobre a escolha e os detalhes de implementação do algoritmo de repetição espaçada usado no sistema, uma versão modificada do algoritmo SM-17 [20].

O algoritmo SM-17 foi escolhido devido à sua eficácia e adoção em sistemas de repetição espaçada. Desenvolvido entre os anos de 2014 e 2016, ele é uma versão atualizada do algoritmo SuperMemo, que é utilizado no software de aprendizado com base em repetição espaçada de mesmo nome, desenvolvido desde 1987. O SM-17 visa otimizar o processo de retenção do conteúdo através de uma heurística que leva em consideração os padrões de memorização humanos.

O algoritmo usa uma abordagem adaptativa para realizar o agendamento das revisões, determinando os intervalos a partir da dificuldade percebida de cada cartão, bem como o desempenho do usuário e intervalo anterior.

Para isso, cada cartão possui um valor, denominado "fator de facilidade", ou *easiness factor*, que não está acessível ao usuário. Todo novo cartão inicia com esse valor em 2,0. Esse valor, que é modificado durante as sessões de revisão, aumenta quando a resposta está correta e diminui quando a resposta está errada. Ele é atualizado sempre que o cartão recebe uma resposta, esteja ele no estado "aprendendo", "revisando" ou "falhado". Na figura 36, podemos observar o pseudocódigo do algoritmo que calcula esse fator, da forma que foi implementado no sistema.

```
function getNewEasinessFactor(status, answer, easinessFactor) {
  var result
  if (answer == ERRADO) {
    result = easinessFactor - 0.8
  } else if (status == APRENDENDO || status == FALHADO) {
    result = easinessFactor + 0.1
  } else if (status == REVISANDO) {
    if (answer == DIFICIL)
      var factor = 0
    else if (answer == CORRETO)
      var factor = 1
    else if (answer == FACIL)
      var factor = 2
    result = easinessFactor + 0.1 + factor * (0.08 - factor * 0.02)
  }
  return max(result, 1.2)
}
```

Figura 36 - Função que retorna o fator de facilidade.

A função recebe três parâmetros: o estado atual do cartão (*status*), a resposta que foi dada (*answer*), e o fator de facilidade anterior (*easinessFactor*). A resposta dada sempre terá o valor CORRETO ou ERRADO, exceto no cartão de tipo "frente e verso", onde as indicações do usuário de facilidade da resposta "difícil", "ok" e "fácil" correspondem aos valores de entrada DIFICIL, CORRETO e FACIL, respectivamente. A resposta é considerada incorreta para o valor ERRADO e correta para qualquer outro valor.

Veja, na lista abaixo, o comportamento do algoritmo para cada valor possível dos parâmetros:

- Caso a resposta tenha sido incorreta, o fator de facilidade resultante será o fator original decrescido de 0,8.
- Caso a resposta tenha sido correta e o estado atual do cartão seja "aprendendo" ou "falhado", o fator de facilidade resultante será o fator original acrescido de 0,1.

- Caso a resposta tenha sido correta e o estado atual do cartão seja "revisando", o acréscimo do fator de facilidade dependerá da indicação de facilidade do cartão.
  - Se a resposta foi DIFICIL em um cartão "frente e verso", o acréscimo é de 0,1.
  - Se a resposta foi CORRETO em um cartão "frente e verso" ou correta em qualquer outro tipo de cartão, o acréscimo é de 0,16.
  - Se a resposta foi FACIL em um cartão "frente e verso", o acréscimo é de 0,18.

O valor mínimo para o fator de facilidade é 1,2, o que é garantido através da função `max()` no retorno da função.

O novo fator de facilidade definido é salvo junto ao cartão. Ele só é usado posteriormente caso o cartão esteja no estado "revisando".

Todo cartão possui um valor denominado "intervalo", contado em segundos, que equivale ao espaçamento de tempo entre os agendamentos de revisões daquele cartão. Esse valor é armazenado internamente e não está acessível ao usuário.

Após uma revisão, em caso de resposta correta, o intervalo previamente registrado é multiplicado pelo fator de facilidade, resultando em um novo intervalo. Em seguida, a próxima revisão é agendada, somando-se o instante atual da revisão ao novo valor do intervalo.

Por exemplo: se um cartão do tipo "frente e verso" possui um fator de facilidade igual a 2,4 e o último intervalo de revisão foi de 2 dias, se o usuário responder o cartão corretamente clicando no botão "Acertei (Difícil)" (ver figura 20), o fator de facilidade será incrementado para 2,5, valor que irá multiplicar o intervalo anterior para obter o novo intervalo de 5 dias. A nova revisão será agendada para 5 dias depois do momento atual.

O valor inicial e mínimo para o intervalo de repetição é de 2 horas. Esse valor é aplicado quando um cartão é movido do estado "aprendendo" ou "falhado" para o estado "revisando". Considerando que o fator de facilidade nunca é inferior a 1,2, o intervalo de repetição nunca diminuirá, desde que o usuário sempre responda corretamente ao cartão. Isso está alinhado com o princípio da repetição espaçada de aumentar gradativamente os intervalos de repetição.

Dessa forma, o tempo que será acrescido para agendamento da próxima revisão depende não apenas da resposta atual ao cartão, mas também do histórico de respostas àquele cartão, pois é ele que determina o fator de facilidade.

### 3.5 Arquitetura do sistema e tecnologias usadas

O sistema MemorizApp é uma aplicação web que consiste em três componentes: o front-end, o back-end e o banco de dados. O front-end é responsável por exibir os dados obtidos do back-end ao usuário, além de capturar as entradas do usuário e enviá-las ao back-end; essa comunicação é feita através de uma API REST. O back-end lida com as regras de negócio, incluindo a captura, tratamento, validação e retorno de dados. O estado da aplicação é armazenado no banco de dados, e o back-end se comunica com ele por meio de requisições SQL para obter e modificar os dados necessários.

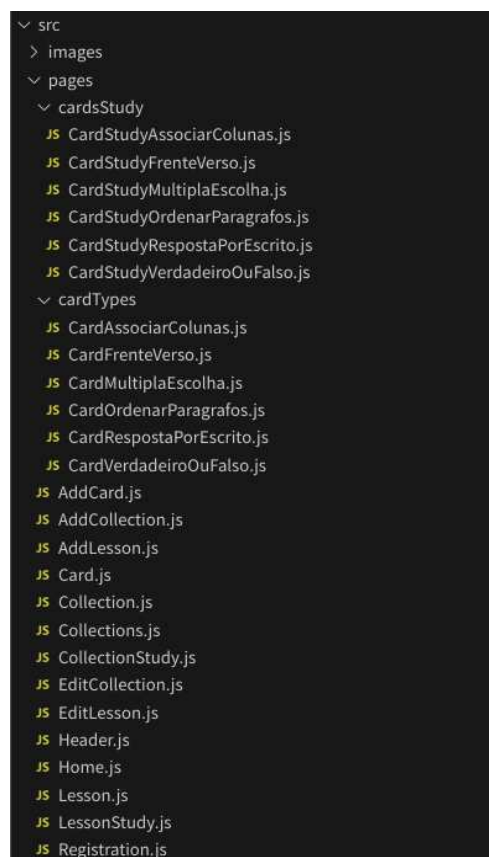
#### 3.5.1 Front-end

O front-end do sistema foi desenvolvido usando Node.js [9] e React [14]. Essa biblioteca foi escolhida para o front-end devido à sua eficiência e facilidade de uso, sua capacidade de criar interfaces responsivas, além da ampla adoção no mercado

e em comunidades open-source, o que resulta em facilidade de encontro de informações relevantes. Possuindo uma abordagem baseada em componentes reutilizáveis, a interface foi desenvolvida de forma modular.

Na figura 37, podemos ver como foram organizados os componentes da aplicação. A seguir, temos uma descrição dos principais componentes:

- Header: é o cabeçalho do aplicativo. Está sempre visível e mostra o ícone, nome e a barra de navegação. Pode ser visto nas figuras 20, 23, 25, 34 e 35.
- Home: equivale à página inicial do aplicativo para quando o usuário não está logado (figura 20).
- Registration: este componente é o formulário usado nas telas de registro (figura 21) e login (figura 22).
- Collections: este componente possui a tabela de coleções (figura 23).
- AddCollection: visto na figura 24, é o formulário para adicionar uma nova coleção.
- Collection: visto na figura 25, este componente representa a página de edição de coleção.
- EditCollection: é o formulário de edição de coleção (figura 26).
- AddLesson: é o formulário para adicionar uma lição a uma coleção, como mostrado na figura 29.
- Lesson: este componente exibe as informações de uma lição e sua lista de cartões, como mostra a figura 28. O componente Collection possui uma lista desse.
- EditLesson: é o formulário de edição de lição (figura 30).
- Card: este componente exibe um cartão na tela de edição de coleção. Uma lista desse está presente no componente Lesson, como visto na figura 32.
- AddCard: a figura 31 mostra o estado inicial deste componente, que é usado tanto para adição de um novo cartão quanto para edição de um cartão existente. Ele possui um dos componentes da pasta *cardTypes* como componente filho, dependendo do tipo do cartão selecionado.
- *cardTypes*: nesta pasta, estão 6 componentes, equivalentes aos 6 tipos de cartão presentes no sistema. Eles permitem editar os campos dos cartões e salvá-los. Descrições e figuras para cada um deles estão presentes na seção 3.2.
- CollectionStudy: este componente representa a tela de estudo de coleção, como mostram as figuras 34 e 35. Ele possui como componentes filhos uma lista de LessonStudy, além de exibir os componentes da pasta *cardsStudy* durante a sessão de estudo.
- LessonStudy: este componente mostra uma lição na tela de estudo de coleção, como mostram as figuras 34 e 35. Ele exibe o nome e descrição da lição, além de possuir o botão que permite mover os cartões de "novo" para "aprendendo".
- *cardsStudy*: nesta pasta, estão 6 componentes, equivalentes aos 6 tipos de cartão presentes no sistema. Eles exibem os cartões durante as sessões de estudo e permitem que o usuário envie as respostas. Descrições e figuras para cada um deles estão presentes na seção 3.2.



**Figura 37 - Principais componentes do front-end da aplicação.**

O front-end usa algumas bibliotecas além de React, como axios [1] para realização de requisições HTTP e universal-cookie [21] para gerenciamento de *cookies*. A autenticação do usuário é feita através de um token JWT, o que será aprofundado posteriormente. A comunicação com o back-end se dá através de requisições a uma API REST exposta pelo serviço.

### 3.5.2 Back-end

O back-end do sistema foi implementado utilizando Node.js [9] em conjunto com o framework Express.js [5]. A escolha pelo Express.js foi baseada em sua estrutura de desenvolvimento simples e flexível, permitindo a criação de uma API REST de forma rápida e eficiente. O framework inclui recursos como roteamento, o que possibilita a definição de rotas de forma organizada e modular, e middleware, o que proporciona melhor controle do fluxo de requisições e respostas, permitindo a implementação de funcionalidades como autenticação, validação de dados e tratamento de erros.

Além do Express.js, foram usadas algumas bibliotecas no desenvolvimento do back-end.

A biblioteca mysql2 [11] foi usada para realizar a comunicação com o banco de dados.

Considerando que a criptografia de dados sensíveis é uma prática recomendada para a segurança do usuário e do sistema, a biblioteca bcrypt [2] foi usada para tal. Através dela, é feito o hash da senha do usuário antes que ela seja salva no banco de dados, bem como a comparação entre os dados existentes e os recebidos no momento de login.

A autenticação de usuário é realizada através de um token JWT (JSON Web Token). A biblioteca jsonwebtoken foi usada para criação e verificação do token. No momento de login, é enviado

para o usuário um objeto encriptado, contendo o identificador único e nome do usuário que realizou login, assim como o *timestamp* da criação do token. Esse objeto é assinado com uma chave secreta que é específica à aplicação back-end; esta chave garante a autenticidade do token que foi gerado. Este token é armazenado como um *cookie* pelo front-end e enviado juntamente a toda requisição que requer autenticação, sendo verificado pelo back-end através de um *middleware* específico presente nas rotas. Como o objeto encriptado inclui o identificador do usuário, através dele é possível saber qual usuário realizou uma requisição.

O anexo A contém a lista com uma descrição de todas as rotas expostas pela API do sistema.

### 3.5.3 Banco de dados

O banco de dados MySQL [10] foi escolhido para o projeto por ser um sistema de gerenciamento de banco de dados confiável e amplamente utilizado, possuindo uma comunidade ativa e diversidade de recursos disponíveis, além da familiaridade do desenvolvedor.

O projeto do banco de dados foi realizado juntamente com o desenvolvimento da API do back-end. Foi desenvolvida uma função no back-end para realizar as migrações de tabelas para inicialização do banco.

## 4. EXPERIÊNCIA ADQUIRIDA

Nesta seção, descreveremos a experiência do desenvolvedor durante o processo de criação da aplicação MemorizApp.

A criação do projeto foi um desafio, visto que envolveu a construção de uma aplicação full stack desde o início, incluindo a concepção da ideia, que surgiu a partir de uma necessidade do próprio desenvolvedor, o planejamento das funcionalidades, a escolha das tecnologias, o processo de desenvolvimento, os testes e a documentação.

A seleção das tecnologias foi um desafio, pois apesar de alguma experiência no desenvolvimento de aplicações web, o desenvolvedor nunca havia iniciado uma aplicação full stack do zero. Foi necessário testar diversas opções de tecnologias para back-end e front-end até encontrar as que melhor se adaptassem às necessidades do projeto e à rotina de trabalho.

Além disso, para o desenvolvimento do front-end da aplicação, foi necessário aprender React, tecnologia sobre a qual o desenvolvedor não possuía conhecimento, além de conceitos de HTML e CSS, linguagens com as quais possuía pouca familiaridade. A criação do front-end foi a etapa mais desafiadora do projeto, mas resultou no adquirento e aplicação de habilidades importantes.

Infelizmente, devido ao cronograma e à complexidade do projeto, nem todas as funcionalidades planejadas puderam ser implementadas; estas serão detalhadas na próxima seção. Houve necessidade de estabelecer uma relação de prioridade entre as funcionalidades planejadas, para que aquelas consideradas essenciais para a demonstração da aplicação fossem desenvolvidas primeiro, deixando para segundo plano funcionalidades adicionais. Além disso, não foi possível realizar a etapa de avaliação, que consiste no teste e validação do protótipo da aplicação pelo público-alvo.

Apesar desses desafios, mas também através deles, a experiência de desenvolvimento da aplicação MemorizApp e da construção deste trabalho foram enriquecedoras, pois representaram a possibilidade de desenvolver novas habilidades ao mesmo tempo em que se aplicaram conhecimentos desenvolvidos ao longo da graduação em um projeto real.

## 5. CONCLUSÃO E TRABALHOS FUTUROS

Em seu estado atual, o sistema MemorizApp se revela uma ferramenta útil para estudantes que buscam memorizar conteúdos e reter conhecimentos de forma eficiente, observável através de suas funcionalidades implementadas com sucesso.

No entanto, devido à limitação do tempo e à complexidade do projeto, algumas das funcionalidades e aspectos originalmente planejados não puderam ser implementadas nesta versão. Esses aspectos seriam relevantes para uma melhora na capacidade da aplicação e na experiência do usuário, e podem ser implementadas em uma versão futura do projeto, ou mesmo em um novo projeto de âmbito similar a ser desenvolvido. São elas:

- Aprimoramento do design de interface do usuário (UI) e experiência do usuário (UX). A interface atual é um protótipo, desenvolvida para apresentar as funcionalidades do sistema, mas sem se preocupar com as melhores práticas de design de interface e experiência. Um refatoramento do design estaria alinhado com um dos propósitos originais do sistema, que é ser fácil de usar para iniciantes;
- Inclusão de novos tipos de teste para cartões, incluindo verificação da resposta através de reconhecimento de fala e de escrita à mão, úteis para o aprendizado de, por exemplo, novos idiomas;
- Capacidade de criar cartões e descrições de lições em *rich text*, ou seja, possibilidade de formatar os elementos de texto, como escolher o tipo e o tamanho da fonte, aplicar estilos como negrito ou itálico, e adicionar elementos multimídia, como imagens, áudio e vídeo, proporcionando uma experiência mais diversificada de aprendizado;
- Importação de dados para cartões através de formatos externos ao sistema, como CSV ou JSON, para que os usuários gerem automaticamente coleções a partir de dados de outras fontes;
- Possibilidade de lembrar o usuário, por meio de uma notificação diária por e-mail, por exemplo, sobre a existência de revisões pendentes, auxiliando na manutenção de uma rotina de estudos mais consistente;
- Permitir que os usuários compartilhem as coleções que criaram em um ambiente público, onde outros usuários poderão salvá-las para sua conta e estudar com elas.

## 6. REFERÊNCIAS

- [1] Axios. <https://axios-http.com/docs/intro>.
- [2] bcrypt. <https://www.npmjs.com/package/bcrypt>.
- [3] Buchweitz, B. Elaboração de Questões de Múltipla Escolha. Estudos em Avaliação Educacional, São Paulo, n.14, p. 105-132, 1996. <http://educa.fcc.org.br/pdf/eae/n14/n14a06.pdf>.
- [4] Elmes, D. Anki - powerful, intelligent flashcards, 2021. <https://apps.ankiweb.net/>.
- [5] Express - Node.js web application framework. <https://expressjs.com/>.
- [6] Inhae. Anki: mistakes to avoid and tips to regain control over your deck. Inside That Japanese Book, 11 jun. 2017. <https://insidethatjapanesebook.com/2017/06/12/anki-mistakes-to-avoid-and-tips-to-regain-control-over-your-deck/>.
- [7] Kroton. Manual para elaboração e revisão de questões para avaliação. Kroton, 2017.



- <http://cms.saladoprofessor.com.br/media/attachments/2017/07/10/1.-manual-para-elaboracao-de-questoes-2017.2.pdf>.
- [8] Memrise. <https://www.memrise.com>.
- [9] Node.js. <https://nodejs.org/en>.
- [10] Oracle. MySQL. <https://www.mysql.com/>.
- [11] mysql2. <https://www.npmjs.com/package/mysql2>.
- [12] Orço, C. L.; Iop, E.; Gai, N. A. Diferenças individuais no processo de aprendizagem em sala de aula. *Unoesc & Ciência - ACHS*, [S. l.], v. 9, n. 2, p. 133–138, 2018.  
<https://periodicos.unoesc.edu.br/achs/article/view/19978>.
- [13] Pergher, G. K.; Stein, L. M. Compreendendo o esquecimento: teorias clássicas e seus fundamentos experimentais. *Psicologia USP*, v. 14, p. 129–155, 2003.  
<https://www.scielo.br/j/pusp/a/vpcf6StvHKKPBzkLPBq4GG/?lang=pt#>.
- [14] React. <https://react.dev/>.
- [15] Basic Anki Setup | Refold. <https://refold.la/roadmap/stage-1/a/anki-setup>.
- [16] Rodrigo. Aprendizagem Distribuída vs. Aprendizagem em Massa: Definição e Exemplos. <https://pt.estudando.com/aprendizagem-distribuida-vs-aprendizagem-em-massa-definicao-e-exemplos/>.
- [17] Silva, D.; Carniello, A.; Carniello, A. Flashcards virtuais - técnica de repetição espaçada aplicada ao apoio na memorização do conteúdo estudado. *Gestão Universitária*, v. 3, 2015.  
<http://www.gestaouniversitaria.com.br/artigos-cientificos/flashcards-virtuais-tecnica-de-repeticao-espacada-aplicada-ao-apoio-na-memorizacao-do-conteudo-estudado--2>.
- [18] Tamm, S. Spaced Repetition: A Guide to the Technique. *E-Student*, 10 jan. 2023.  
<https://e-student.org/spaced-repetition/>.
- [19] Velásquez, F. R. Enfoques sobre el aprendizaje humano. *Universidad Simón Bolívar*, junho de 2001.
- [20] Wozniak, P. Algorithm SM-17. SuperMemo Guru. [https://supermemo.guru/wiki/Algorithm\\_SM-17](https://supermemo.guru/wiki/Algorithm_SM-17).
- [21] universal-cookie. <https://www.npmjs.com/package/universal-cookie>.

## ANEXO A - Rotas da API

Todas as rotas requerem autenticação através de token JWT, exceto onde indicado.

Rota	Método	Corpo	Retorno	Efeito
/auth/register	POST	{ username, password }		Rota pública. Criação de um novo usuário, com o nome de usuário e senha especificados.
/auth/login	POST	{ username, password }	{ accessToken, issuedAt }	Rota pública. Realiza login do usuário e retorna o token, caso o nome de usuário e senha estejam corretos.
/auth	GET			Verifica se o usuário está logado com um token válido, retornando o código HTTP 403 FORBIDDEN caso contrário.
/collections	POST	{ name }	{ idCollection }	Cria uma nova coleção.
/collections	GET		[{ id, name, cardCount, cardCountNew, cardCountLearning, cardCountReview, cardCountDue, cardCountFailed, cardCountBuried }]	Retorna a lista de coleções do usuário.
/collections/:collectionId	GET		{ name, settings: { random, defaultNewCards }, lessons: [{ id, name, description, order }], cards: [{ id, idLesson, idCardType, card, order, idCardStatus }], }	Retorna uma coleção para a tela de edição. Os parâmetros de URL "lessons" e "cards" podem ser definidos como false para não incluir as listas no retorno.
/collections/:collectionId	PUT	{ name, settings: { random, defaultNewCards } }		Altera o nome ou as configurações de uma coleção.
/collections/:collectionId	DELETE			Apaga uma coleção e todas as suas lições e cartões.
/collections/:collectionId /study	GET		[{ name, lessons: [{ id, name, description, order, countNew }], defaultNewCards }]	Retorna uma coleção para a tela de estudo.

Rota	Método	Corpo	Retorno	Efeito
/collections/:collectionId/lessons	POST	{ name, description }	{ idLesson }	Cria uma nova lição em uma coleção.
/collections/:collectionId/lessons/reorder	PUT	{ lessons: [idLesson] }		Altera a ordem das lições de uma coleção.
/collections/:collectionId/lessons/:lessonId	PUT	{ name, description }		Altera uma lição.
/collections/:collectionId/lessons/:lessonId	DELETE			Apaga uma lição e todos os seus cartões.
/collections/:collectionId/lessons/:lessonId/cards	POST	<pre>// Frente e verso: {   idCardType: "FB"   frente,   verso }  // Resposta por escrito: {   idCardType: "WA"   pergunta,   resposta }  // Verdadeiro ou falso: {   idCardType: "TF"   frente,   resposta }  // Associar colunas: {   idCardType: "CA"   proposta,   itens: [{     esquerda,     direita   }] }  // Ordenar parágrafos: {   idCardType: "OP"   proposta,   paragrafos: [texto] }  // Múltipla escolha: {   idCardType: "MC"   pergunta,   aleatorio,   opcoes: [{     texto,     correto   }] }</pre>		Cria um novo cartão em uma lição de uma coleção.
/collections/:collectionId/lessons/:lessonId/cards/reorder	PUT	{ cards: [idCard] }		Altera a ordem dos cartões de uma lição.
/collections/:collectionId/lessons/:lessonId/cards/:cardId	PUT	Igual ao da rota que cria um novo cartão.		Altera um cartão.
/collections/:collectionId/lessons/:lessonId/cards/:cardId	DELETE			Apaga um cartão.

Rota	Método	Corpo	Retorno	Efeito
/collections/:collectionId/lessons/:lessonId/cards/:cardId/bury	PUT	{ buried }		Define se um cartão está suspenso.
/collections/:collectionId/lessons/:lessonId/begin	POST			Move os cartões de uma lição de "novo" para "aprendendo". A quantidade de cartões a ser movida é 10 por padrão, mas pode ser alterada definindo o parâmetro de URL "limit".
/collections/:collectionId/pending	GET		{ isDue, card: { id, idCardStatus, idCardType, card } }	Retorna um cartão para ser estudado durante a sessão de estudo. Caso não exista nenhum cartão a ser estudado, "isDue" será false e "card" será null.
/collections/:collectionId/cards/:cardId/answer	POST	{ answer }		Responde a um cartão durante uma sessão de estudo, alterando seu estado e intervalo e agendando a revisão.