

**CENTRO PAULA SOUZA  
FACULDADE DE TECNOLOGIA DE FRANCA  
“Dr. THOMAZ NOVELINO”**

**TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**KEFFLEN MORENO RAMOS**

**APLICAÇÃO DE *FLASH CARDS* COMPARTILHÁVEIS**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Franca - “Dr. Thomaz Novelino”, como parte dos requisitos obrigatórios para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Fausto Cintra

**FRANCA/SP**

**2022**

## APLICAÇÃO DE *FLASH CARDS* COMPARTILHÁVEIS

Kefflen Moreno Ramos<sup>1</sup>

### Resumo

A fim de solucionar um problema que afeta muitos estudantes e pessoas que queiram aprender novas línguas, que é a dificuldade de memorizar inúmeras palavras novas, este projeto busca adaptar o sistema de *flash cards*, já utilizado por autodidatas no mundo todo, para utilização também por um professor em sala de aula. A proposta é criar uma aplicação em que o professor ou instrutor crie grupos de alunos e compartilhe baralhos de *flash cards* com os grupos. Os baralhos e os *flash cards* são adicionados e atualizados pelo professor. A aplicação poderia então ser utilizada por qualquer pessoa interessada em ensinar a grupos de pessoas por meio de *flash cards*.

**Palavras-chave:** Aprendizagem de línguas. Aprendizagem em grupo. Desenvolvimento *Web*. Memorização de palavras.

### Abstract

*To solve a problem that affects many students and people who want to learn new languages, which is the difficulty of memorizing a wide variety of new words, this project aims at adapting the flash card system, already used by self-taught students around the world, for use also by a teacher in the classroom. The objective is to create an application in which the teacher or instructor creates groups of students and shares decks of flash cards with the groups. Decks and flash cards are added and updated by the teacher. The application could then be used by anyone who is interested in teaching groups of people through flash cards.*

**Keywords:** *Group learning. Language learning. Web development. Word memorization.*

### 1 Introdução

Flash cards já são usados por estudantes para auxiliar no estudo de matérias para concursos e por estudantes de línguas estrangeiras para a memorização de novas palavras. A vantagem de usar *flash card* é a revisão contínua e espaçada, que ajuda na memorização do conteúdo. Um *flash card* consiste basicamente em uma pergunta e uma resposta, o estudante olha para a pergunta e deve se lembrar da resposta.

Atualmente, a tecnologia está presente em todas as áreas da sociedade, e na educação não é diferente. As novas tecnologias são aliadas da educação, podendo

---

<sup>1</sup> Graduando em Análise e Desenvolvimento de Sistema pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: kefflen.ramos@fatec.sp.gov.br

dinamizar o processo de ensino e aprendizagem (EDUCA MAIS BRASIL, 2022). E é por intermédio da tecnologia que este projeto busca adaptar o sistema de *flash cards* para um sistema que possa ser utilizado por um educador.

Esse sistema, por exemplo, poderia ser usado por um professor de língua estrangeira para auxiliar seus alunos a memorizar outras línguas. O grande diferencial é o esquema de grupos, que é a solução proposta deste projeto, para auxiliar um professor ou qualquer pessoa interessada em ensinar algum conteúdo para outras pessoas usando o sistema de *flash cards*.

Além da possibilidade de o professor criar os *decks* de *flash cards*, os alunos também podem criar seus próprios *decks*. E não fica restrito ao estudo de línguas, podendo ser utilizado para a memorização de qualquer assunto de interesse de quem o utiliza.

## 2 Viabilidade do projeto

A Figura 1 representa o Canvas do projeto, auxiliando a visualização de questões relevantes à viabilidade do projeto.

Figura 1 – Modelo Canvas



Fonte: Elaborado pelo autor

Analisando o Canvas, é possível notar que o projeto possui dois possíveis clientes, que seriam alunos e professores, sendo que o aluno estará interessado a entrar em grupos de estudos e o professor, em administrar grupos de estudos. Além disso, é possível identificar uma fonte de receita, que seriam os planos, voltados para escolas ou cursos, oferecendo a possibilidade de criar mais grupos de estudos do que o estabelecido para o plano gratuito, e com um limite maior de membros.

### 3 Levantamento de Requisitos

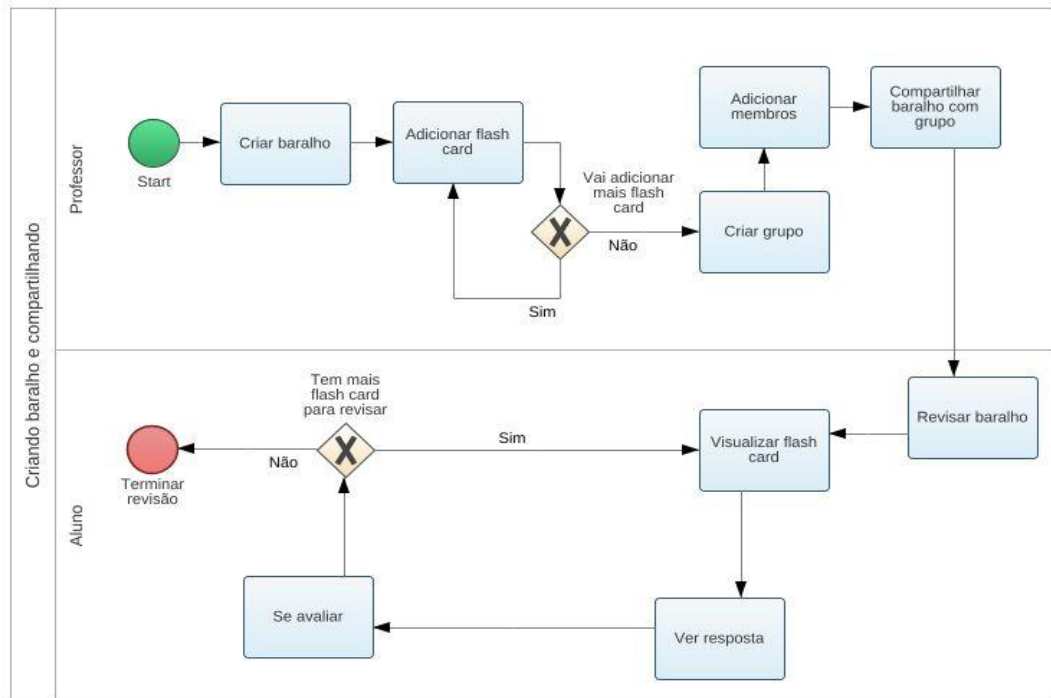
#### 3.1 Elicitação e especificação dos Requisitos

Os possíveis *stakeholders* foram procurados em uma comunidade do Discord (aplicação pela qual pessoas podem criar ou participar de comunidades e se comunicar com amigos), voltada ao estudo de línguas estrangeiras, tendo sido feita uma entrevista com algumas perguntas sobre o sistema almejado, para validar requisitos já levantados e elicitar eventuais novos requisitos. Além disso, também foram feitos diagramas para auxiliar o processo de elicitação de requisitos, como BPMN e os casos de usos. Também foram feitas análises de outros sistemas de *flash cards* que já existentes, como por exemplo, o Anki (2022).

#### 3.2 BPMN

Na Figura 2 é apresentado o BPMN, apresentando um processo do sistema. Foi apresentado o processo de criar um baralho e depois compartilhar. Ele se inicia na criação de um baralho que vai conter *flash cards* e será compartilhado com um grupo para que seus membros possam estudar o baralho. Neste BPMN há dois atores: professor e aluno.

Figura 2 - BPMN



Fonte: Elaborado pelo autor

### 3.3 Requisitos Funcionais

**Quadro 1 – Requisitos Funcionais do sistema**

<b>RF001-Cadastro de usuário</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve cadastrar usuários		
<b>RF002-Cadastrar baralho</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve cadastrar baralho		
<b>RF003-Cadastro de <i>flash cards</i></b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> O sistema deve cadastrar <i>flash cards</i>		
<b>RF004-Compartilhar baralho</b>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média

		( ) Baixa
<b>Descrição:</b> Usuário podem compartilhar baralhos com grupos		
<b>RF005</b> -Revisar <i>flash cards</i>	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> Usuários podem revisar os <i>flashs cards</i> do baralho		
<b>RF006</b> -Criar grupo	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> Usuário pode criar grupos para compartilhar baralhos		
<b>RF007</b> -Convidar usuário para grupo	Categoria: ( ) Oculto (X) Evidente	Prioridade: (X) Altíssima ( ) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> Usuário pode convidar membros para grupo		
<b>RF008</b> -Autoavaliação	Categoria: ( ) Oculto (X) Evidente	Prioridade: ( ) Altíssima (X) Alta ( ) Média ( ) Baixa
<b>Descrição:</b> Usuário pode se autoavaliar		

### 3.4 Requisitos Não Funcionais

**Quadro 2** – Requisitos Não Funcionais do sistema

<b>RNF001</b> - Sistema Web e Mobile	O sistema será <i>Web</i> e <i>mobile</i> alimentado por uma API Rest	Tipo	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF002</b> - Grupo precisa de membros	O grupo precisa de membros para ser criado	Tipo	(X) Desejável ( ) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF003</b> - Baralho sincronizado	Os baralhos compartilhados com o grupo devem estar sincronizados com o do dono	Tipo	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF004</b> - Membro deve ser convidado para grupo	Para o membro participar de um grupo ele precisa ser convidado pelo dono	Tipo	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF005</b> - Usuario precisa estar logado	Para usar as funcionalidades do sistema é necessário estar logado	Tipo	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF006</b> - Membro pode ser convidado por <i>e-mail</i> ou pelo nome de usuário	Um membro pode ser convidado pelo seu <i>e-mail</i> ou pelo nome de usuário	Tipo	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório

<b>RNF007-</b> Apagar grupos vazios	Grupos sem membros além do dono devem ser apagados depois de uma semana	Tipo	(X) Desejável ( ) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF008-</b> Oferecer opções para autoavaliação	Ao ver a resposta do <i>flash card</i> , enquanto revisa o baralho, deve ser oferecido opções como certo e errado para o usuário	Tipo	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório

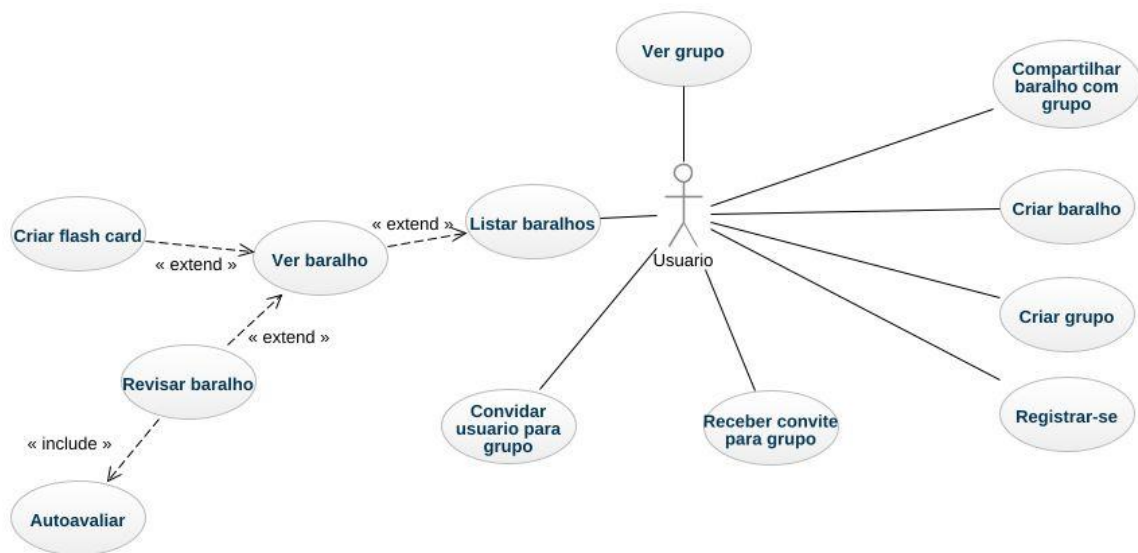
### 3.5 Regras de Negócio

**Quadro 3** – Regras de Negócio do sistema.

<b>RN001 – Flash cards possui pergunta e resposta</b>
<b>Descrição:</b> Flash cards devem ser criados com pergunta e respostas

### 3.6 Casos de Uso

Figura 3 - Diagrama de casos de uso



Fonte: Elaborado pelo autor

Especificação de cada um dos casos de uso.

**Quadro 4** – Criar baralho

<b>Caso de Uso – Criar baralho</b>	
<b>ID</b>	UC 001
<b>Descrição</b>	Este caso de uso tem por objetivo criar um baralho

<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Nenhuma
<b>Cenário Principal</b>	1. O caso de uso inicia quando o usuário seleciona a opção cadastro de criar um baralho. 2. O usuário dá um nome para o baralho 3. O usuário confirma sua criação
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	2a. O usuário dá um nome de baralho existente 2a.1 O sistema informa que já existe um baralho com esse nome 2a.2 O sistema pede para que seja inserido um novo nome

**Quadro 5 – Criar flash card**

<b>Caso de Uso – Criar flash card</b>	
<b>ID</b>	UC 002
<b>Descrição</b>	Adicionar novos <i>flash cards</i> em um baralho
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Precisa de um baralho já criado
<b>Cenário Principal</b>	1. O caso de uso se inicia quando o usuário seleciona criar <i>flash card</i> 2. O sistema carrega a página para criação do <i>flash card</i> 3. O usuário digita a pergunta 4. O usuário digita a resposta da pergunta 5. O usuário confirma a criação do <i>flash card</i>
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	Nenhum

**Quadro 6 – Listar baralhos**

<b>Caso de Uso – Listar baralhos</b>	
<b>ID</b>	UC 003
<b>Descrição</b>	Tem o objetivo de mostrar os baralhos assim como interagir com elas
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Nenhuma
<b>Cenário Principal</b>	1. O usuário clica para ver seus baralhos 2. O sistema traz todos os baralhos disponíveis
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	Nenhum

**Quadro 7 – Ver baralho**

<b>Caso de Uso – Ver baralho</b>	
<b>ID</b>	UC 004
<b>Descrição</b>	Este caso de uso tem como objetivo ver dados estatísticos, revisá-lo e inserir <i>flash card</i>
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Ter baralho criado
<b>Cenário Principal</b>	1. Inicia quando o usuário clica sobre o baralho para visualizá-lo 2. O sistema mostra a última vez que foi revisado, quantos <i>flash cards</i> possui e se tem <i>flash card</i> para revisar
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	3a Criar <i>flash card</i> 2a.1 Inclui caso de uso "UC 002"



	3b. Revisar <i>flash cards</i> 3b.1 Inclui caso de uso "UC 006"
--	--

Quadro 8 – Revisar baralho

Caso de Uso – Revisar baralho	
<b>ID</b>	UC 005
<b>Descrição</b>	Este caso de uso tem por objetivo revisar <i>flash cards</i> do baralho
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Precisa ter <i>flash cards</i> para revisar
<b>Cenário Principal</b>	1. Inicia quando selecionado revisar <i>flash cards</i> do baralho 2. É mostrado um card com a pergunta a mostra e a resposta oculta 3. O usuário aperta para ver a resposta 4. Inclui caso de uso "UC 007" 5. O sistema mostra o próximo <i>flash card</i> retornando para o passo 2 do fluxo principal
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	5a Não tem mais <i>flash cards</i> para revisar 5a.1 O sistema mostra que não tem mais <i>flash cards</i> para revisar 5a.2 Retorna para a tela anterior

Quadro 9 – Autoavaliar

Caso de Uso – Autoavaliar	
<b>ID</b>	UC 006
<b>Descrição</b>	Tem o objetivo de o usuário avaliar-se, e o sistema decidir quando será a próxima revisão
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Nenhuma
<b>Cenário Principal</b>	1. Inicia depois que o usuário vê a resposta do <i>flash card</i> durante a revisão 2. O sistema disponibiliza opções como certo ou errado
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	Nenhum

Quadro 10 – Criar grupo

Caso de Uso – Criar grupo	
<b>ID</b>	UC 007
<b>Descrição</b>	Tem o objetivo de criar grupos/salas para que seja compartilhado baralhos
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Nenhuma
<b>Cenário Principal</b>	1. Inicia na tela inicial ao clicar criar grupo 2. O usuário digita o nome do grupo 3. O usuário adiciona os baralhos que serão compartilhados 4. O usuário adiciona os membros por <i>e-mail</i> 5. O usuário confirma a criação do grupo
<b>Pós-condição</b>	Ter pelo menos um membro ativo no grupo
<b>Cenário Alternativo</b>	2a O usuário já tem um grupo com esse nome 2a.1 O sistema pede para que seja inserido um novo nome 2a.2 O usuário digita o novo nome 2a.3 Retorna para o fluxo principal no passo 3

Quadro 11 – Receber convite para grupo

Caso de Uso – Receber convite para grupo	
<b>ID</b>	UC 008
<b>Descrição</b>	Este caso de uso tem por objetivo possibilitar receber convite e entrar em grupo
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Nenhuma
<b>Cenário Principal</b>	1. O usuário recebe uma notificação de convite para grupo 2. Ele abre o convite 3. O sistema mostra nome do usuário e do grupo 4. O usuário aperta sim para confirmar que quer entrar no grupo
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	4a O usuário não aceitar entrar para o grupo

Quadro 12 – Registrar-se

Caso de Uso – Registrar-se	
<b>ID</b>	UC 009
<b>Descrição</b>	Este caso de uso tem como objetivo descrever o processo de registro de usuário
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Nenhuma
<b>Cenário Principal</b>	1. O usuário clica para registrar-se 2. O sistema pede o <i>e-mail</i> e a senha 3. O sistema pede um nome para entrar no sistema 4. O sistema pede um apelido que será público para os outros usuários
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	2a E-mail já existe 2a.1 O sistema informa que o <i>e-mail</i> já existe 2a.2 O sistema encerra o processo de registro 3a Nome de usuário já existe 3a.1 O sistema informa que o nome de usuário já existe 3a.2 Retorna para o fluxo principal no passo 3

Quadro 13 – Compartilhar baralho com grupo

Caso de Uso – Compartilhar baralho com grupo	
<b>ID</b>	UC 010
<b>Descrição</b>	Este caso de uso tem como objetivo compartilhar baralho com um grupo
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Possuir baralho Ser dono do grupo
<b>Cenário Principal</b>	1. inicia quando o usuário clica para compartilhar baralho 2. O sistema mostra todos os grupos que o usuário possui 3. O usuário seleciona o grupo ou grupos com quem deseja compartilhar 4. O usuário confirma o compartilhamento
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	2a O usuário não possui grupo 2a.1 O sistema avisa que não tem grupo para compartilhar 4a O usuário cancela compartilhamento

**Quadro 14 – Convidar usuário para grupo**

<b>Caso de Uso – Convidar usuário para grupo</b>	
<b>ID</b>	UC 011
<b>Descrição</b>	Este caso de uso tem como objetivo convidar usuário para grupo
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Possuir grupo Ser dono do grupo
<b>Cenário Principal</b>	1. inicia quando o usuário clica para convidar usuário no grupo 2. O sistema pede a identificação do usuário que deseja convidar 3. O usuário depois de inserir a identificação confirma
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	2a O usuário clica em cancelar

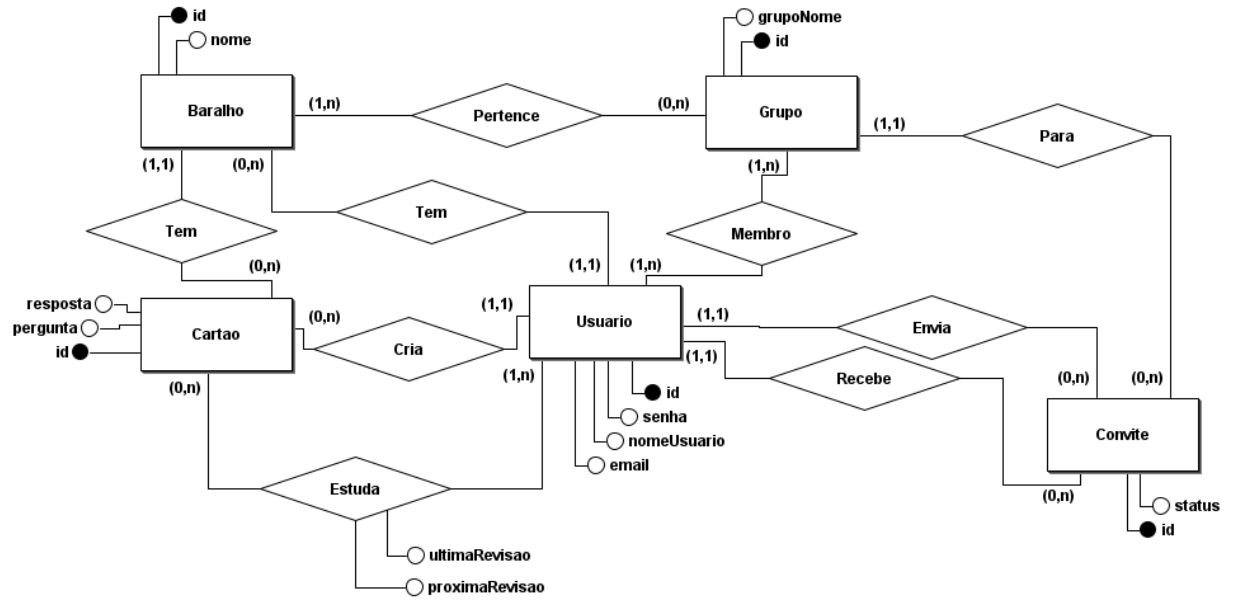
**Quadro 15 – Ver grupo**

<b>Caso de Uso – Ver grupo</b>	
<b>ID</b>	UC 012
<b>Descrição</b>	Este caso de uso tem como ver grupo
<b>Ator Primário</b>	Usuário do sistema
<b>Pré-condição</b>	Possuir grupo
<b>Cenário Principal</b>	1. inicia quando o usuário clica para ver o grupo 2. O sistema mostra dados os baralhos do grupo e a identificação do dono
<b>Pós-condição</b>	Nenhuma
<b>Cenário Alternativo</b>	Nenhum

### 3.7 Diagrama Entidade-Relacionamento

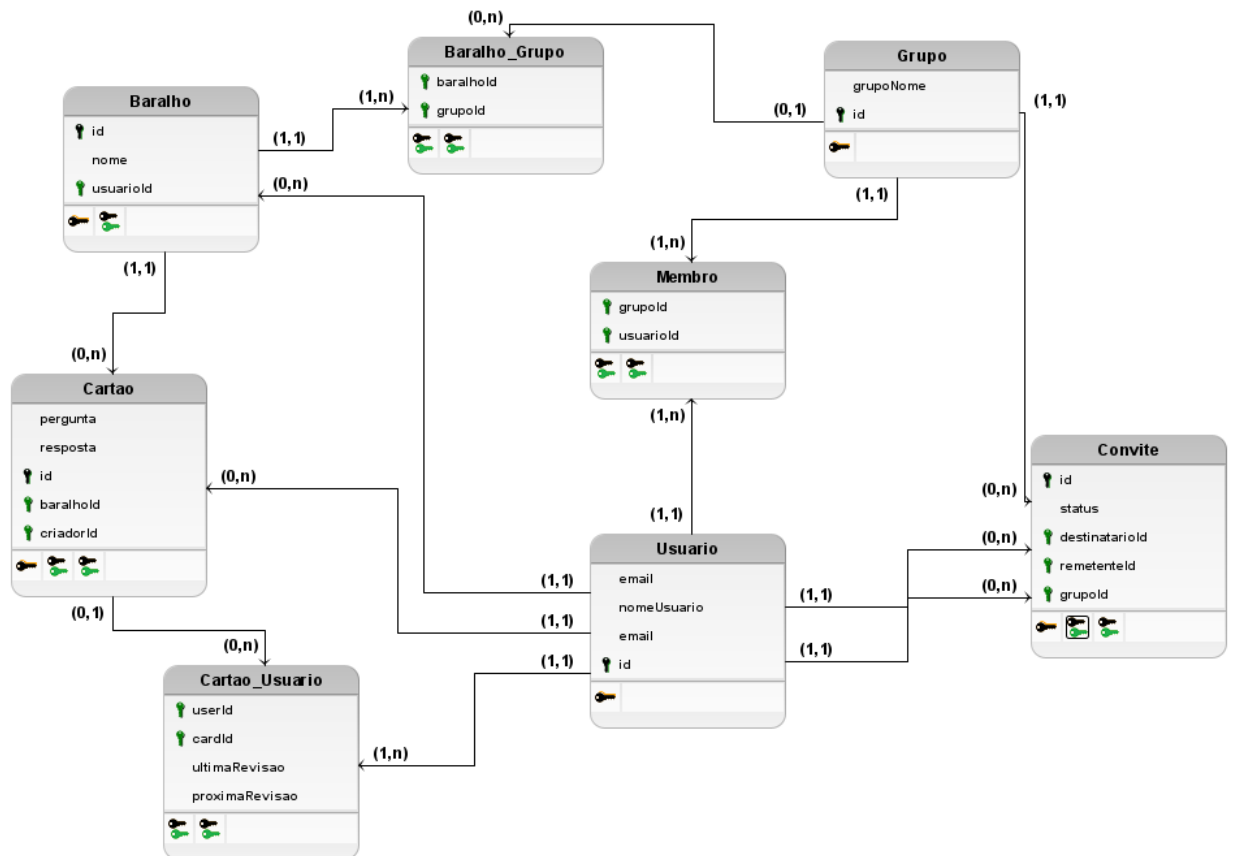
A Figura 4 traz o diagrama conceitual que representa a modelagem do banco de dados e a Figura 5 é a versão lógica. Pela Figura 4, podemos observar a presença de cinco tabelas que se relacionam, sendo a principal delas a tabela de usuário. Um usuário poderá se relacionar com vários baralhos, cartões, grupos e convites. O baralho tem relações com vários grupos e pode ter vários cartões. Na versão lógica (Figura 5), nota-se a presença de tabelas intermediárias para fazer os relacionamentos de muitos para muitos.

Figura 4 - Diagrama Entidade-Relacionamento - Conceitual



Fonte: Elaborado pelo autor

Figura 5 - Diagrama Entidade-Relacionamento - Lógico



Fonte: Elaborado pelo autor

## 4 Ferramentas e Métodos de Desenvolvimento

### 4.1 Ferramentas

#### a) **Typescript**

Typescript é uma linguagem de programação altamente tipada que foi construído com base no Javascript (TYPESCRIPT, 2022). Foi necessário para o projeto pois, no domínio da aplicação, é importante fazer uso de interfaces que não existem no Javascript puro. Além disso, mantém o código mais fácil de ler e descobrir erros de tipos.

#### b) **React**

Trata-se de uma biblioteca Javascript popular para construir interfaces de usuários *Web* (REACT, 2022). Escolhida por ser muito fácil de construir interfaces modernas e responsivas.

#### c) **Prisma**

Prisma é um ORM (*Object Relational Mapper*), para executar comandos no banco de dados com código Typescript (PRISMA, 2022). Oferece recursos de migração e é compatível com os bancos de dados mais populares no mercado.

#### d) **MySQL**

O banco de dados escolhido para o projeto é o MySQL versão 8 com licença GPL (General Public Licence) (MYSQL, 2022). Escolhido pela familiaridade do autor com a ferramenta, e por oferecer alguns recursos que podem ser utilizados na aplicação, como *enums*.

#### e) **Node.js**

Node é uma aplicação multiplataforma, *open source*, usada para executar código Javascript (NODE.JS, 2022). Foi utilizado na versão 18.10.0 para rodar o código Javascript que é compilado do TypeScript.

## 4.2 Métodos de Desenvolvimento

### a) **Domain Driven Design (DDD)**

O Domain Driven Design (DDD) fornece estrutura para o desenvolvimento de *software* centrado no domínio do negócio, por meio de princípios e padrões. Por exemplo, a separação da aplicação em camadas, tendo uma camada central que é o domínio da aplicação, onde é encapsulada toda a lógica do negócio e desacoplada de dependências. De acordo com Eric Evans (2017), a capacidade de resolver problemas relacionadas ao domínio é o lugar onde está o coração do *software*.

O projeto usa os padrões estruturais sugeridos pelos DDD, como entidades, repositórios, serviços e a separação da aplicação em várias camadas.

### b) **Ports and Adapters**

É uma arquitetura de *software* que separa as aplicações em camadas, em que as camadas superiores não têm conhecimentos das camadas inferiores. Aplica os conceitos estruturais do DDD, e, para se comunicar com a camada de domínios, é usado a estrutura de *ports*, que seriam as interfaces que dizem o que será feito no sistema, e os *adapters*, que são as implementações (HOLMSTRÖM, 2020).

Foi utilizado por ser compatível com os princípios apresentados no DDD, para que a aplicação tenha escalabilidade, e seja de fácil manutenção.

### c) **API Rest**

API (*Application Program Interface*) é um conjunto de regras e definições para que sistemas se comuniquem, e Rest são restrições de como uma API deve funcionar (RED HAT, 2020). Por exemplo, para uma API ser *restful*, é necessário que apresente uma interface uniforme, ausência de estado etc. Seu uso é necessário pois o projeto deve atender tanto a *Web* como a *mobile* e é importante ter uma API que possa servir estas plataformas.

## 5 Resultados e Discussão

A aplicação inicia na tela de *login*, mostrada na Figura 6, onde o usuário pode entrar no sistema usando seu *e-mail* e sua senha. Caso ainda não tenha cadastro, ele pode clicar no botão para se cadastrar e irá ser encaminhado para tela da Figura 7.

**Figura 6** - Tela de *login*

Flash card

Email:

Senha:

Entrar

Ainda não tem uma conta?  
Registre-se

Fonte: Elaborado pelo autor

**Figura 7** - Tela de cadastro

Flash card

Nome de usuário:

Senha:

Email:

Nome:

Registrar-se

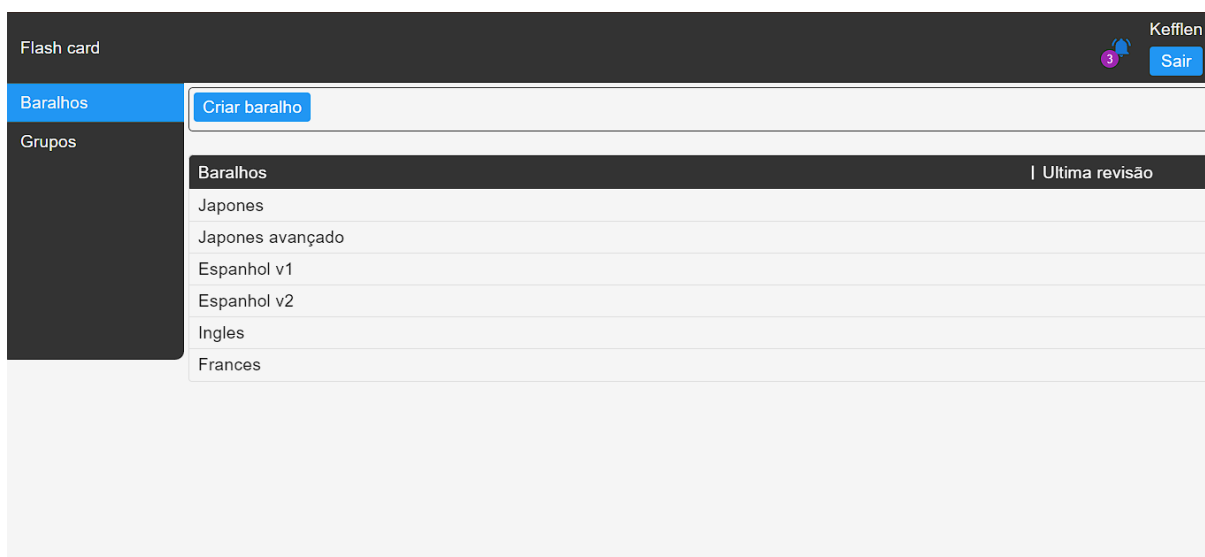
Ja tem uma conta?  
Entre

Fonte: Elaborado pelo autor

Após entrar na aplicação, o usuário é direcionado para a tela principal (Figura 8), na qual são mostrados todos os baralhos que o usuário tem para revisar. Nessa

tela, o usuário pode criar um *deck* informando um nome, e pode clicar diretamente na tabela para ver mais informações do baralho, assim como revisá-lo.

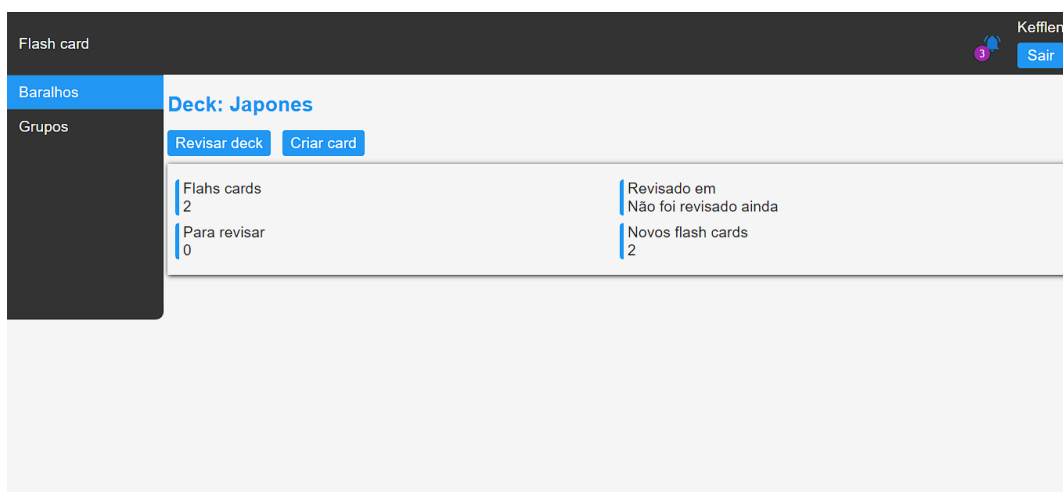
**Figura 8 - Home/Baralhos**



Fonte: Elaborado pelo autor

A tela de informações do baralho (Figura 9) mostra o total de *flash cards* que há no baralho, quantos existem para revisar e os *flash cards* novos para ver no dia. Também nesta tela o usuário pode iniciar a revisão e criar *flash cards*.

**Figura 9 - tela do baralho**



Fonte: Elaborado pelo autor

Na Figura 10, podemos observar a tela para criar o *flash card*, em que é necessário escrever a pergunta e a resposta. Na Figura 11, vê-se a tela de revisão no



ponto em que o sistema está esperando o clique do usuário para mostrar a resposta e oferecer opções para o usuário se avaliar, como pode ser visto na Figura 12.

**Figura 10 - Criar *flash card***



The screenshot shows a web application interface for creating a flash card. The top navigation bar includes the text 'Flash card' on the left and the user name 'Kefflen' with a profile icon and a 'Sair' button on the right. A sidebar on the left contains two menu items: 'Baralhos' (highlighted in blue) and 'Grupos'. A 'Voltar' button is located at the top of the main content area. The main content area is divided into two sections: 'Pergunta' and 'Resposta', each with a large white text input field. A blue 'Criar' button is positioned at the bottom right of the 'Resposta' field.

Fonte: Elaborado pelo autor

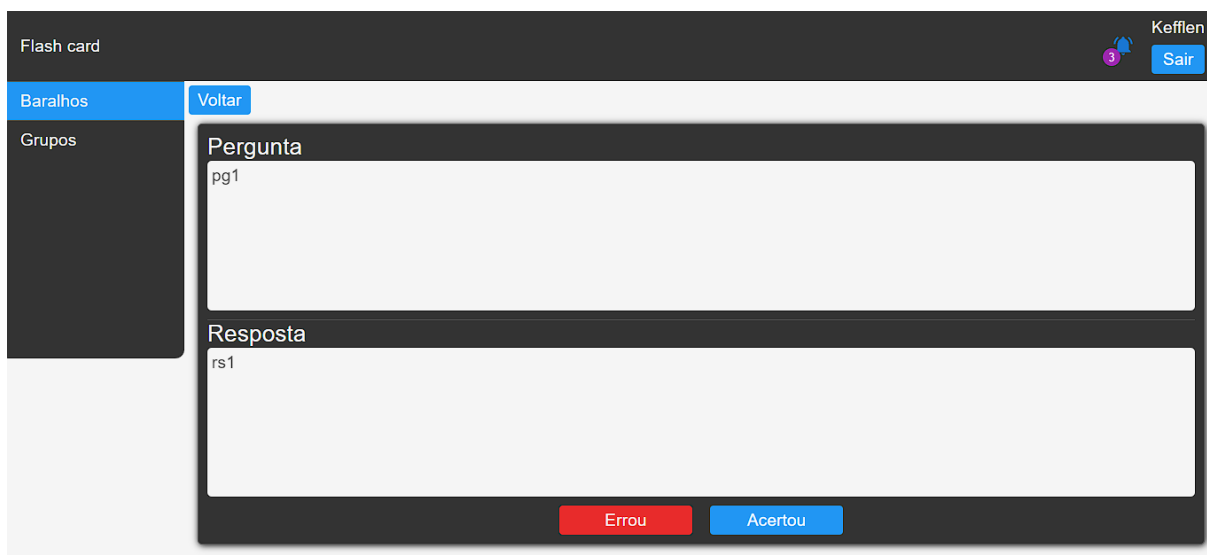
**Figura 11 - Revisar *flash card***



The screenshot shows the same web application interface as Figure 10, but in the 'Revisar flash card' state. The 'Pergunta' input field now contains the text 'pg1'. The 'Resposta' input field is currently empty. The 'Criar' button is no longer visible, indicating that the card has been saved and is now in a review state.

Fonte: Elaborado pelo autor

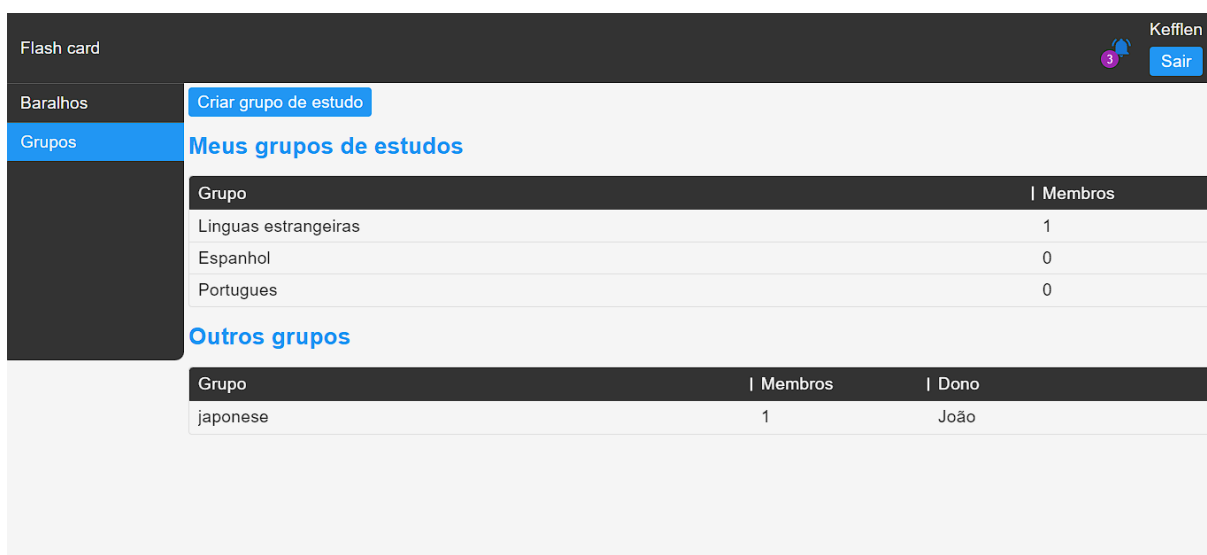
**Figura 12** - Ver resposta e se avaliar



Fonte: Elaborado pelo autor

Na Figura 13, é possível ver os grupos que o usuário criou e também aqueles que em que o usuário está inserido, sendo possível, ainda, criar um novo grupo. Ao clicar sobre um grupo, informações sobre ele são exibidas.

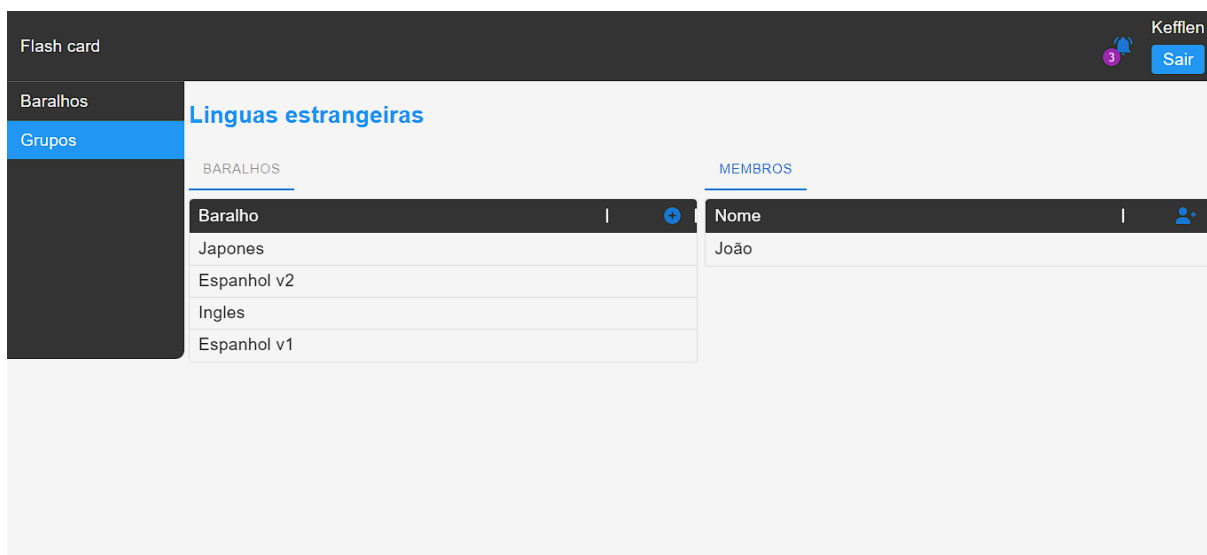
**Figura 13** - Tela de grupos



Fonte: Elaborado pelo autor

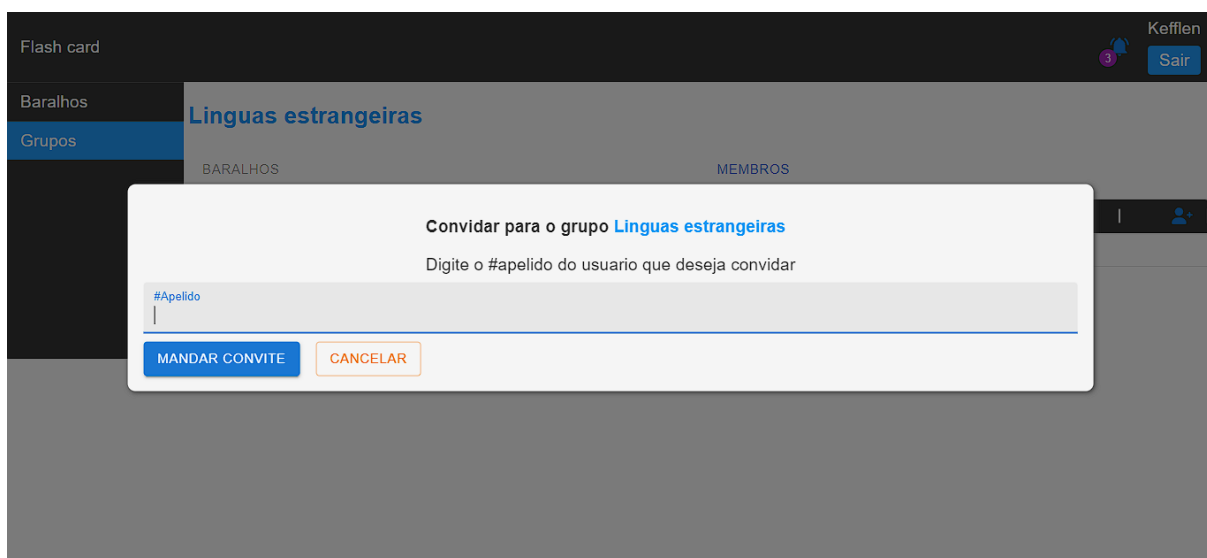
Na tela do grupo, mostrada na Figura 14, observa-se os baralhos do grupo e de seus membros. Esta tela permite enviar convites para outros usuários, como mostrado na Figura 15, assim disponibilizando baralhos para outros membros.

Figura 14 - Tela do grupo



Fonte: Elaborado pelo autor

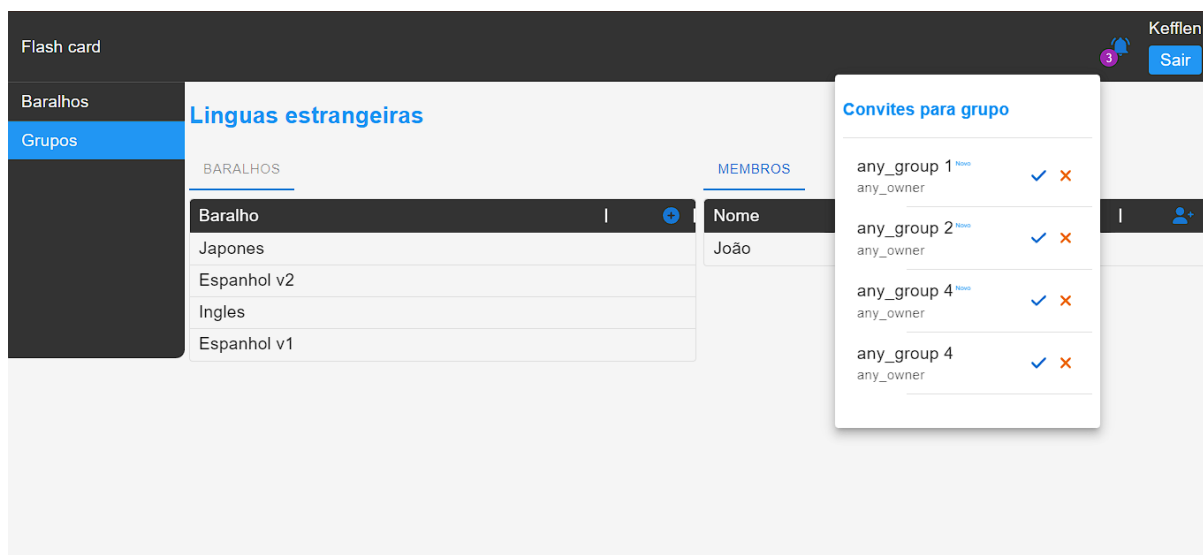
Figura 15 - Convidar membro



Fonte: Elaborado pelo autor

Na tela apresentada na Figura 16, são mostradas para o usuário todas as notificações de convites para grupos que ele tenha recebido. São disponibilizadas as opções de confirmar ou recusar o convite.

Figura 16 - Notificações



Fonte: Elaborado pelo autor

### Considerações finais

A grande dificuldade de auxiliar alunos a memorizar novas palavras de idiomas estrangeiros traz a necessidade de buscar novas soluções, a partir de um sistema que já é usado por políglotas no mundo todo, para aprender novas línguas. Com isso em mente, foi suscitada a possibilidade de desenvolver um sistema que se adeque às necessidades de instrutores dispostos a ensinar grupos de pessoas.

É importante ressaltar que já existem outros sistemas de *flash cards* no mercado, mas o diferencial deste projeto é a possibilidade de criar grupos de estudos e compartilhar *flash cards* que serão gerenciados pelo dono do baralho.

Optou-se inicialmente por implementar uma solução *Web*, mas a ideia é também implementar uma versão *mobile*, pelo fato de ser um dispositivo que costuma estar disponível para o usuário utilizar a qualquer momento.

### Referências

ANKI. **Anki**. Disponível em: <https://apps.ankiweb.net/>. Acesso em: 11 out. 2022.

EDUCA MAIS BRASIL. **Benefícios e desafios da tecnologia na educação**. Disponível em: <https://www.educamaisbrasil.com.br/educacao/escolas/beneficios-e-desafios-da-tecnologia-na-educacao>. Acesso em: 9 out. 2022.

EVANS, Eric. **Domain Driver Design**: atacando as complexidades no coração do *software*. 3. ed. Rio de Janeiro: Alta Books, 2017.

HOLMSTRÖM, Petter. **DDD Part 3**: Domain-Driven Design and the Hexagonal Architecture. 2020. Disponível em: <https://vaadin.com/blog/ddd-part-3-domain-driven-design-and-the-hexagonal-architecture> . Acesso em: 2 out. 2022.

MySQL. **MySQL**. Disponível em: <https://www.mysql.com/>. Acesso em: 8 out. 2022.

NODE.JS. **Node.js**. Disponível em: <https://nodejs.org/en/>. Acesso em: 8 out. 2022.

PRISMA. **Prisma**. Disponível em: <https://www.prisma.io/>. Acesso em: 2 out. 2022.

REACT. **REACT**. Disponível em: <https://pt-br.reactjs.org/>. Acesso em: 2 out. 2022.

RED HAT. **API REST**. 2020. Disponível em: <https://www.redhat.com/pt-br/topics/api/what-is-a-rest-api>. Acesso em: 2 out. 2022.

TYPESCRIPT. **TypeScript**. Disponível em: <https://www.typescriptlang.org/>. Acesso em: 2 out. 2022.