

Desenvolvimento de Aplicação para o Ensino da Língua Brasileira de Sinais

Gabriel Celestino, Vinicius Pasa Cervelin,
Wilson Heck Junior, Paula Clarice Santos Grazziotin de Jesus

¹Ciência da Computação – Instituto Federal de Santa Catarina - Lages (IFSC)
Caixa Postal 15.064 – 88506-400 – Santa Catarina – SC – Brasil
gabriel.c31@aluno.ifsc.edu.br, vinicervelin@gmail.com,
wilson.junior@ifsc.com.br, paula.jesus@ifsc.edu.br

Abstract. *The lack of incentive and practicality when it comes to LIBRAS (Brazilian Signs Language) learning is responsible for stabilising a communicational barrier between deaf and hearing people. Therefore, this conclusion paper aims to improve the interaction between both hearing and deaf people through the developing of a application that helps on LIBRAS learning. Based on the projection of a flashcard memorization system into a web architecture adapted for learning the language, the software developed according to the knowledge obtained during the IFSC/Lages-SC Computational Science graduation makes use of platforms, languages and frameworks as PHP, CSS, Bootstrap, MySQL and many others, to provide all who wish to learn and practice LIBRAS with a more gamified way.*

Resumo. *A Falta de incentivo e praticidade no aprendizado de Libras acaba por gerar uma barreira comunicacional. Diante disso, este trabalho de conclusão de curso visa aprimorar a interação entre ouvintes e surdos por meio do desenvolvimento de um aplicativo que auxilie no aprendizado da Língua Brasileira de Sinais (Libras). Tendo como base a técnica de memorização flashcard com arquitetura web adaptada ao ensino da língua, o software desenvolvido em acordo com o conhecimento obtido ao decorrer do curso de Ciência da Computação do IFSC/Lages-SC faz uso de plataformas, linguagens e frameworks como PHP, CSS, Bootstrap, MySQL e vários outros, almejando assim auxiliar todos que desejem aprender e praticar a libras de uma forma mais gamificada.*

Palavras-chave— Flashcard, Gameficação, Ensino, Libras

1. Introdução

De acordo com o Instituto Brasileiro de Geografia e Estatística (IBGE, 2020), o Brasil conta com aproximadamente 10 milhões de deficientes auditivos (IBGE 2010). Movimentos sociais com origem dentro da comunidade surda acarretaram no reconhecimento da Língua Brasileira de Sinais (Libras) como língua oficial de comunicação da comunidade surda com a (Lei 10.436/2002), que foi a mais importante para a comunidade nos últimos tempos, e consequentemente, a implantação do Decreto Nº 5.626, de 22 de Dezembro de 2005 (BRASIL, 2005). Este sendo o mais recente para a comunidade, nele é dito que cada instituição deve dispor de docentes capacitados para o ensino de Libras para crianças no ensino fundamental e adultos surdos no ensino superior. Essa medida foi tomada de forma a melhorar o acesso de pessoas surdas à educação de qualidade, mas

a realidade é que a busca pelo reconhecimento não acabou, ainda faltam estruturas, políticas e ferramentas que ajudem a implementar e reforçar o dispositivo legal. Diante desta realidade é que se desenvolve a presente pesquisa.

O surdo muitas vezes depende de um intérprete para se comunicar, e isso acarreta em problemas como, por exemplo, na área da saúde, onde pode vir a ocorrer uma falha na comunicação entre profissionais da saúde e pessoas surdas ou pessoas com deficiência auditiva (MAGRINI e dos SANTOS, 2014). Este problema é uma consequência, entre outros fatores, da falta de incentivo ao aprendizado de Libras por parte da população ouvinte (ANDREIS-WITKOSKI, 2011). Normalmente, os ouvintes são mais incentivados a aprender libras em cursos de educação superior - principalmente licenciaturas - e, assim, apenas uma pequena parcela da nação tem acesso ao aprendizado de Libras. (LEMOS e CHAVES, 2012).

Tendo como exemplo o próprio IFSC-Lages, o qual em devido momento chegou a ter 4 estudantes surdos matriculados simultaneamente. O campus, no entanto, se dispõe de poucos servidores com o conhecimento em línguas de sinais, o que acarreta em dificuldades de adaptação no ensino desses alunos.

Assim sendo, o projeto tem como objetivo contribuir para a criação uma ponte de informação entre ouvintes e surdos através de um aplicativo interativo para ambos surdos, e ouvintes terem uma ferramenta a mais de aprendizado, logo em sua infância, a Libras. Assim, a pesquisa se propõe a contribuir para promover a aprendizagem da língua brasileira de sinais como forma de promover o diálogo entre ouvintes e surdos, em uma sociedade mais inclusiva. Objetivos específicos: a) consultar especialistas em Libras para definir um conjunto de palavras e sinais; b) obter gravações de sinais e imagens que representam os sinais para serem utilizados no sistema de *Flashcards*; c) especificar como o desenvolvimento do *software* será feito através do levantamento de requisitos, casos de uso e modelagem do banco de dados; d) implementar um *software web* para aprendizagem de Libras através de *Flashcards*;

O *software* proposto atua como um suporte online ao método de ensino conhecido como *Flashcard*, método que coloca em prática uma implementação simples do princípio da repetição espaçada, em que os cartões são revisados em intervalos crescentes de tempo, garantindo que seu conteúdo seja associado aos poucos pelo usuário. O sistema *Flashcard* a ser utilizado neste projeto será adaptado ao ensino de Libras para ouvintes, e surdos.

Durante o processo, educadores de Libras irão colaborar para a criação de um conteúdo simples de forma intuitiva para surdos, deficientes auditivos e os que fazem parte do grupo dos ouvintes, criando assim um aplicativo tutor capaz de acolher à todos. Isso aliado a ajuda e colaboração de surdos e ouvintes, trazendo assim a inclusão não só no produto final, mas por meio de todo o processo de desenvolvimento, o que acarretará em profundo conhecimento a respeito dos surdos não apenas por meio de referências bibliográficas, mas também por meio do contato e convivência.

Além desta seção introdutória, este documento é composto das seguintes seções: A seção 2 apresenta os assuntos relacionados ao entendimento do trabalho; A seção 3 registra o desenvolvimento do trabalho; E, na seção 4, as conclusões sobre o trabalho são apresentadas.

2. Referencial Teórico

Para entender melhor a respeito do proposto, precisa-se conhecer melhor o seu público alvo, suas características culturais, biológicas, sociais e linguísticas, diferenciando assim o surdo do deficiente auditivo. Assim sendo, a seção 2.1 aborda as diferenças entre deficientes auditivos e pessoas com surdez; a seção 2.2 aborda sobre a cultura surda, seguida pela seção 2.3, onde é feito um breve

resumo histórico sobre a Língua Brasileira de Sinais; a seção 2.4 descreve o sistema *Flashcards*, para então falar a respeito das tecnologias de desenvolvimento na seção 2.5 e finalizar com a seção 2.6 onde é explicado sobre banco de dados.

2.1. Deficiência Auditiva vs Surdez

O conceito geral de deficiência auditiva é, segundo a Lei nº 5.296 (2004), “a perda bilateral, parcial ou total, de quarenta e um decibéis (dB) ou mais, aferida por audiograma nas frequências de 500Hz, 1.000Hz, 2.000Hz e 3.000Hz”.

Mas qual a diferença entre um deficiente auditivo e uma pessoa surda?

Pessoa com deficiência auditiva é aquela cuja audição está prejudicada a ponto de **dificultar**, mas não impedir a compreensão da fala. São pessoas que se comunicam de forma oral. Geralmente são pessoas que apresentam uma perda leve ou moderada. Estas pessoas podem fazer bom uso de Aparelhos de Amplificação Sonora Individual e apresentam, na maioria das vezes, uma boa leitura labial, o que facilita seu acesso à língua oral e, em consequência, ao processo de alfabetização e letramento.

Pessoa com Surdez é aquela cuja audição está prejudicada a ponto de **impedir** a compreensão da fala através do sistema auditivo, com ou sem o uso de um Aparelho de Amplificação Sonora Individual. Estes são os usuários da Língua Brasileira de Sinais. Indivíduos que geralmente apresentam uma perda auditiva severa ou profunda. Podem ou não fazer uso da leitura labial e se comunicar pela língua oral, dependendo das intervenções e tratamentos feitos anteriormente (HONORA, 2014, p.25).

O termo “Surdo-Mudo” é errôneo, um vez que é importante notar que estes dois termos não representam a mesma aflição, uma vez que pessoas com surdez, como o próprio nome diz, não escutam, mas conseguem falar, sua oralidade apenas não foi “treinada” devido a falta de entendimento auditivo, porém não apresentam nenhum problema nos órgãos fono articulatórios, assim como pessoas mudas não necessariamente sejam surdas. A utilização de termos como “Surdo-Mudo”, “Mudo”, “Mudinho” pode acabar causando desconforto, seu uso é desaconselhado até que se tenha conhecimento de como a pessoa em questão prefere ou não ser chamada (HONORA, 2014).

Além deste critério clínico, as pessoas surdas também se identificam como tais por sua identidade cultural, ao se reconhecerem como uma comunidade que compartilha uma experiência de vida desconhecida pela maioria ouvinte.

2.2. Cultura Surda

A cultura surda é uma cultura diversa e complexa, dentro da qual se encontram as culturas da libras. Assim como as culturas das línguas orais, como o português a cultura surda tem em si uma diversidade de elementos que agregam os indivíduos que dela fazem parte, por um processo de identificação. A língua é um destes elementos.

A **comunidade surda**, na verdade não é só de surdos, já que tem sujeitos ouvintes junto, que são família, intérpretes, professores, amigos e outros que participam e compartilham os mesmos interesses em comuns em um determinado localização que podem ser as associação de surdos, federações de surdos, igrejas e outros (STROBEL, 2009).

Se definem por “Surdo” a identidade cultural de grupos de pessoas surdas que se define

como diferente de outros grupos (QUADROS, 2006). O povo surdo é formado por pessoas surdas com seus próprios costumes, história e tradições, não se vendo como pessoas inferiores com relação a outras pessoas com capacidade de ouvir, mas sim como pessoas que constroem uma concepção de mundo por meio da visão (STROBEL, 2009).

A surdez deve ser reconhecida como apenas mais um aspecto das infinitas possibilidades da diversidade humana, pois ser surdo não é melhor ou pior do que ser ouvinte, é apenas diferente (PIMENTA, 2001).

Ainda assim, a ausência do sentido da audição é uma questão que provoca a exclusão de muitos indivíduos, por falta de acessibilidade aos espaços e instituições na sociedade. É neste contexto que este software pretende se inserir, colaborando para o reconhecimento dos direitos destes indivíduos de serem educados, atendidos e amparados por meio de sua língua materna.

2.3. Língua Brasileira de Sinais (Libras)

Tendo como base a Língua de Sinais Francesa (*langue des signes française* ou LSF) e a antiga Língua de Sinais Brasileira, a Língua Brasileira de Sinais presente como conhecemos surgiu em 1857 com a primeira escola de surdos do Brasil - essa escola é conhecida como Instituto Nacional da Educação de Surdos (INES). Em 2002 foi reconhecida como língua oficial das pessoas com surdez pela Lei nº 10.436/2002 (BENASSI, 2014).

É impossível de dizer um número aproximado de línguas de sinais no Brasil pois estas podem ser muito específicas, usadas em algumas regiões, em comunidades surdas específicas, ou em núcleo familiares. LIBRAS é a majoritária, que goza de status de oficialidade por ser adotada pelas políticas do Estado, empregada nos sistemas educacional, judiciário, de saúde, entre outros.

Libras é uma língua visual-espacial, completa com semântica, sintaxe, morfologia, estrutura e gramática única, diferenciando-se do português na questão da língua portuguesa ser uma língua oral-auditiva. Portanto é uma língua totalmente diferente do Português. Além disso, as línguas de sinais não são universais. A libras, por exemplo, é apenas uma das línguas de sinais do Brasil, a majoritária, que goza de status de oficialidade por ser adotada pelas políticas do Estado, empregada nos sistemas educacional, judiciário, de saúde, entre outros.

2.4. Sistema Flashcard

O sistema Leitner é um método amplamente utilizado para o uso eficiente de cartões de memória flash, proposto pelo jornalista científico alemão Sebastian Leitner na década de 1970. Este método funciona de maneira a garantir que o conteúdo passado pelos cartões vai ser fixado e aprendido pelo usuário, e para isto se utiliza de uma estratégia de ensino conhecida, que é realmente potencializada pelo sistema de aprendizado *Flashcards* (TUCKER, 1989).

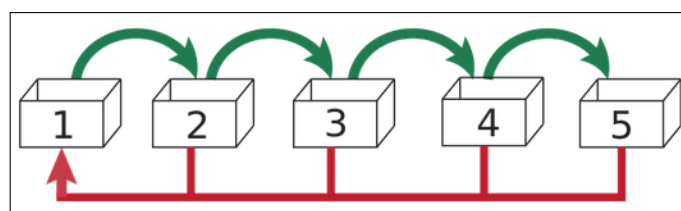


Figura 1. Exemplo de Sistema Flashcard apresentado por (ZIRGUEZI, 2012)

Grande parcela da população já teve contato com o sistema, só não soube identifica-lo. O Sistema Flashcard é utilizado para o aprendizado e retenção eficiente de conteúdos simples, como

frases e palavras em uma língua estrangeira, é o caso do duolingo, que faz uso da metodologia extensivamente em seu método de ensino. Outra área que faz uso extensivo do método é a da matemática, com suas contas e regras mais simples sendo explicadas por meio do sistema em vários jogos na web e até em salas de aula.

Como é possível visualizar na figura 1, neste sistema, o usuário é apresentado a várias cartas, de um lado têm-se perguntas e vídeos e do outro tem-se uma resposta, mudando a estratégia e destino da carta dependendo da resposta do usuário ao lado da pergunta. Uma vez que acerte, o cartão irá passar para a próxima caixa, e conseqüentemente a pergunta se tornará menos propícia a ser repetida. Ao contrário de que o errar acarretaria no desenvolvimento de uma prioridade sobre esta pergunta, garantindo que ela volte repetidas vezes até que o sistema tenha certeza de que seu conteúdo tenha sido fixado pelo usuário.

2.5. Sistemas Web

Sistema Web é um software que tem a internet como área de atuação, usuários cadastrados podem acessa-los por navegador.

Com o passar do tempo, as aplicações Web cada vez mais se destacam por conta de sua facilidade de atualização sendo realizada apenas no servidor e sua abrangência ao conseguir ser acessado por meio de qualquer dispositivo com acesso a internet e a um navegador. A respeito de Sistemas Web, dar-se-ão como informação essencial para o entendimento do trabalho os aspectos e tecnologias descritos a seguir.

2.5.1. Modelo Cliente-Servidor

Modelo Cliente-Servidor é um modelo de aplicação distribuída onde tarefas são divididas entre servidores. Os aplicativos clientes fazem solicitações para um ou mais aplicativos de servidor, que precisam estarem sendo executados para atender a essas solicitações (MARINI, 2012). Na Figura 2 tem-se uma representação visual de um modelo Cliente - Servidor.

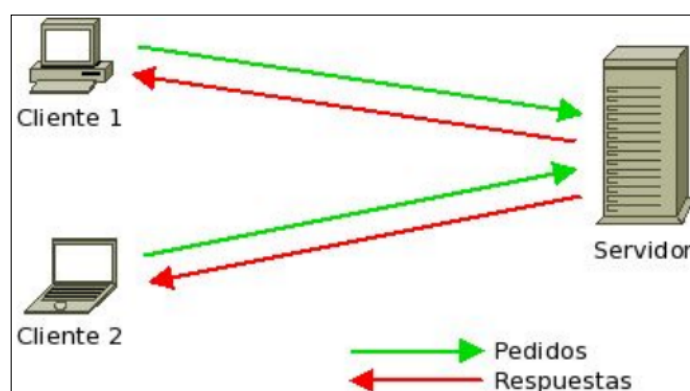


Figura 2. Modelo Cliente-Servidor conforme apresentado por (NINO et al., 2012)

Esse modelo utiliza-se da divisão do trabalho realizado por cada máquina cliente, evitando a sobrecarga de processos numa mesma máquina, tendo maior eficiência e agilidade ao se realizar tarefas complexas, o que o torna um meio facilitador para evitar gastos com máquinas mais potentes, para realizar sozinha um grande processo. O processo é distribuído entre os clientes e os servidores sendo o servidor o gerenciador da distribuição entre os clientes.

O cliente faz uma solicitação ao servidor, que responde com o conteúdo que foi solicitado. Por exemplo, ao entrar em uma página Web, o navegador executa uma solicitação HTTP para o servidor desse endereço, que responde com o código HTML de uma página (NINO et al., 2012).

2.5.2. Hypertext Transfer Protocol (HTTP)

HTTP é o protocolo TCP / IP usado para a transferência de documentos de hipertexto através da internet definindo um conjunto de regras para cada tipo de arquivo enviado (NINO et al., 2012). A porta TCP padrão usada por este protocolo é 80. O cliente faz uma solicitação ao servidor e responde com o conteúdo que foi solicitado. Por exemplo, ao entrar em uma página da Web, o navegador executa uma solicitação HTTP para o servidor desse endereço, que responde com o código HTML da página. Depois que o cliente recebe a mensagem, ele a interpreta e a exibe na tela. É por isso que clientes e servidores HTTP são os componentes básicos da Rede mundial de computadores. Mensagens HTTP são os blocos de dados enviados por aplicativos HTTP. Essas mensagens fluem entre clientes, servidores e proxies.

2.5.3. Hyper Text Transfer Protocol Secure (HTTPS)

O protocolo seguro para transferência de hipertexto (HTTPS) é o intermediador de segurança em transações entre o navegador e o servidor Web, para assim garantir a segurança em sites que requerem informações que de qualquer forma podem ser recebidas, interceptadas e lidas por usuários não autorizados, como por exemplo páginas bancárias, lojas online, e-mails, entre outros.

Para isso, são utilizados mecanismos incluindo a criptografia de comunicações SSL no HTTP, o cliente envia diretamente todas as informações necessárias, incluindo chaves, certificados e códigos de integridade. No entanto, essa troca de informações é feita através do navegador de forma que o usuário não perceba, logo não precisam se preocupar em lidar manualmente com o protocolo, pois o navegador se encarrega desse trabalho (RESCORLA e SCHIFFMAN, 1999).

2.5.4. Hypertext Markup Language (HTML)

HTML é uma linguagem de marcação criada em 1991, para ser um padrão de comunicação e troca de documentos na Web. A linguagem faz uso de conteúdos multimídias (texto, imagens, vídeos e áudios) para criação de páginas Web e comunicar determinada informação (FERREIRA e EIS, 2016).

Baseada na utilização de hipertexto, esta linguagem de marcação possibilita a agregação de blocos de informação, como os conteúdos multimídia, que serão acessados através de *hyperlinks* ou *links*. Este conceito permite que o acesso possa atingir a alta quantidade de informações, que estão conectadas e juntas formam uma grande rede de informação, a internet (FERREIRA e EIS, 2016).

O modo estrutural utilizado pela linguagem são chamados de *tags*, que se utilizam de colchetes angulares como marcador para onde começam e onde terminam seus devidos objetos. Cada página HTML consiste em uma série de *tags* (também chamadas de elementos), que podem ser consideradas os blocos de construção das páginas. Elas criam uma hierarquia que estrutura o conteúdo entre seções, parágrafos, cabeçalhos e outros blocos de conteúdo (HOSTINGER, 2020).

HTML será utilizado neste projeto, mesclado com o PHP na busca do refinamento visual das páginas.

A Figura 3 é uma representação da estrutura básica do HTML.

```
<!DOCTYPE html>
<html>
<head>
  <title>Hello World!</title>
</head>
<body>
  Hello World
</body>
</html>
```

Figura 3. Código de estrutura básica do HTML conforme apresentado por (MORRETO, 2016)

Seguindo o exemplo da Figura 3, segue a explicação sobre as principais *tags*:

“<html>” – É o elemento básico da estrutura do HTML. Uma vez que contém todos os elementos do seu documento e deve iniciar e finalizar todo o documento.

“<head>” – Delimita o cabeçalho do documento. Apesar de não ter nenhum valor visível, é capaz de transmitir ao navegador diversas informações essenciais a um bom documento HTML.

“<body>” – Representa o corpo do documento. É nessa *tag* que todos os elementos visíveis do site devem ser inseridos.(MARQUES, 2015)

2.5.5. Cascading Style Sheets (CSS)

Desenvolvido pelo *World Wide Web Consortium* (W3C) em 1996, o CSS tem por objetivo formatar e estilizar elementos escritos em uma linguagem de marcação como o HTML, de forma a retirar o excesso de informações e o tempo gasto com formatações manuais dentro do próprio HTML. Seu uso basea-se em criar um arquivo separado do HTML contendo apenas os estilos das páginas (tamanho de fonte, cor, efeitos, etc.) e então esses estilos são chamados no arquivo HTML. Esse meio possibilita o reaproveitamento por utilizar a mesma formatação em 10, 50 ou até mesmo 1000 páginas, além de manter o projeto organizado por ter essa separação entre estilos e estrutura da página (LIE e BOS, 1996).

O CSS será utilizado em conjunto com o Bootstrap, almejando o aprimoramento visual das páginas do projeto.

2.5.6. Framework

Framework é uma estrutura de *software* composta de componentes personalizáveis e intercambiáveis para o desenvolvimento de uma aplicação específica. Um *Framework* contém um conjunto de bibliotecas com o objetivo de otimizar e simplificar códigos, por exemplo, com o

Bootstrap pode-se, com um simples comando, criar componentes dinâmicos e responsivos em um site, não precisando assim fazer uso de códigos complexos e que demandariam muito tempo (GUTIÉRREZ, 2014).

2.5.7. Bootstrap

Criado por Mark Otto e Jacob Thornton, o Bootstrap é um *framework front-end* que torna as etapas do desenvolvimento Web mais rápidas, dinâmicas e padronizadas, devido as várias bibliotecas e elementos prontos para serem utilizados (SPURLOCK, 2013).

A Figura 4 representa um código no arquivo HTML utilizando o Bootstrap para a criação de um botão de forma rápida e padronizada. Dentro da *tag* “<a>” há uma classe chamada “*class*” indicando dentro de si estilos próprios do Bootstrap para refinar as propriedades visuais do botão a ser criado no referido trecho de código.

O Bootstrap também trabalha com componentes JS no aspecto de *Plugins JQuery* (cujo trecho de código indicado na Figura 4 mostra uma *tag* “<script>” que carrega um arquivo .JS referente ao jquery) adicionando assim vários outros componentes que ajudam na hora de programar, componentes como carrosséis de imagens, caixas de diálogos, entre muitos outros componentes acompanhados por códigos JS.

```
<body>

  <div class="jumbotron">
    <h1>Hello, world!</h1>
    <p>This is our first sample example that will be more
awesome in the next chapters!</p>
    <a class="btn btn-primary btn-lg" href="#" role="button">
      Bootstrap by Example, Chapter 1
    </a>
  </div>

  <script src="js/jquery-1.11.3.js"></script>
  <script src="js/bootstrap.js"></script>
</body>
```

Figura 4. Código de criação de um botão com Bootstrap apresentado por (MORETO, 2016)

Como saída tem-se, para o usuário, o que é mostrado na Figura 5, que mostra uma criação rápida e padronizada de um botão utilizando do Bootstrap. Para modificar qualquer componente, basta alterar os atributos pré programados pelo Bootstrap, listados nos 3 principais parâmetros citados anteriormente, e também pode-se utilizar de CSS para alterações mais abrangentes e detalhadas se o arquivo CSS do próprio Bootstrap não for totalmente ideal para o programador.

O Bootstrap será usado como estruturação base das páginas, tendo o CSS como complementador/retocador de detalhes.

2.5.8. Java Script

JavaScript (às vezes abreviado para JS) é uma linguagem leve, interpretada e baseada em objetos com funções de primeira classe, mais conhecida como a linguagem de script para páginas

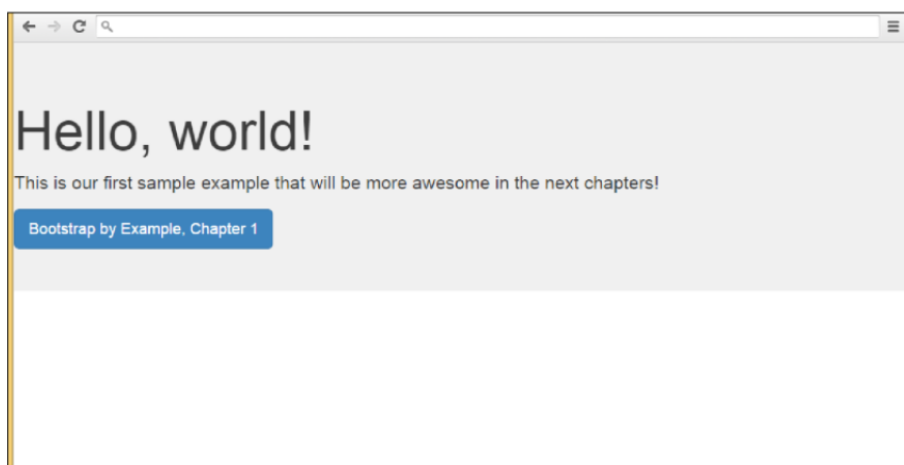


Figura 5. Tela visualização de botão criado pelo Bootstrap apresentado por (MORRETO, 2016)

Web, mas usada também em vários outros ambientes sem *browser*, tais como *node.js*, Apache CouchDB e Adobe Acrobat. O JavaScript é uma linguagem baseada em protótipos, multi-paradigma e dinâmica, suportando estilos de orientação a objetos, imperativos e declarativos (como por exemplo a programação funcional) (MDN, 2019).

O JavaScript se tornou uma característica importante de todos os principais navegadores do mercado. Diferente de antigamente, o JavaScript já não está vinculado somente para validação de dados simples, agora interage com quase todos os aspectos da janela do navegador e seus conteúdos. O JavaScript é reconhecido como uma linguagem de programação completa, capaz de cálculos e interações complexas, incluindo *closures*, funções anônimas (lambda) e até mesmo meta programação. O JavaScript tornou-se uma parte tão importante da Web que até navegadores alternativos, incluindo aqueles em telefones celulares e aqueles projetados para usuários com deficiência o suportam. Mesmo a Microsoft, com sua linguagem de script do lado cliente chamada VBScript, acabou incluindo sua própria implementação de JavaScript no Internet Explorer desde sua versão mais antiga (ZAKAS, 2012).

A utilização do JavaScript dentro do projeto se dá em dois momentos. O primeiro na animação de virada da carta pergunta para a carta resposta no jogo. E o segundo dentro do Banco de Dados, necessário para garantir que administradores possam buscar dado como cartões, usuário e categorias, por nome.

2.5.9. PHP

O desenvolvimento de sistemas para a Web tornou-se uma área de constante crescimento e valorização. A conveniência e disponibilidade desse tipo de sistema motivam as empresas a disponibilizarem suas aplicações via Internet, e para suprir essa demanda, uma série de linguagens e plataformas surgiram nos últimos anos. Dentre elas, o PHP é uma das linguagens mais conhecidas e populares, caracterizada pela sua baixa curva de aprendizagem e facilidade de desenvolvimento (CANTO, 2011).

O PHP é uma linguagem cujo script é *open source* e de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento Web e que pode ser embutida dentro do HTML. Em vez de muitos comandos para mostrar HTML, as páginas PHP contém HTML em código

mesclado (PHP, 2020).

PHP tem suporte a vários protocolos como: IMAP, SNMP, NNTP, POP3, HTTP, LDAP, XML-RPC e SOAP. É possível abrir *sockets* e interagir com outros protocolos. E as bibliotecas de terceiros expandem ainda mais suas funcionalidades (PHP, 2020).

O PHP não se limita a ser apenas uma linguagem embutida no HTML, uma vez que esta pode gerar vários arquivos de mídia em PDF.

É utilizado tanto no servidor e páginas do programa, quanto no Banco de Dados do projeto, onde o PHP conta com cerca de 90% do conteúdo total do mesmo.

2.6. Banco de Dados

Dados são importantes na era onde tudo é informação, desde o controle de gastos do mês, até a maior biblioteca do mundo. A base de dados consiste em qualquer dado necessário para o auxílio das atividades de um indivíduo ou organização, como por exemplo o orçamento do que foi gasto no cartão no fim do Mês, a quantidade exata do tamanho de um terreno, efeitos e contra-efeitos de medicamentos e muito mais. Ela pode ser armazenada por planilhas, arquivos de texto, ou até mesmo uma tabela numa folha de papel. O importante é ter um registro e ter uma visão clara das informações, poder alterá-las e excluí-las, no caso dados de usuários cadastrados dentro do software.

Com o passar do tempo, as bases de dados se tornaram cada vez maiores e isso trouxe problemas como, por exemplo, dificuldade de encontrar os dados, alteração, exclusão e redundância, ou seja, quanto maior a base, mais difícil era encontrar um dado, alterações e exclusões eram trabalhosas e as chances de ter redundância na base de dados era cada vez maior. Assim surgiu o banco de dados computadorizado que consiste em um sistema de computador para armazenamento e manutenção de registros, ou seja, realizar pesquisas, alterações e exclusões de forma mais rápida, e que evita redundâncias (DATE, 2004).

Na Figura 6 observa-se que para um Banco de Dados ser caracterizado como tal, precisa-se de quatro componentes principais: dados indispensáveis para o indivíduo ou organização, um *software* para gerenciá-los, um *hardware*, ou seja, uma máquina capaz de suportar o *software* e com armazenamento suficiente para esses dados, e usuários que farão uso de todas essas informações armazenadas.

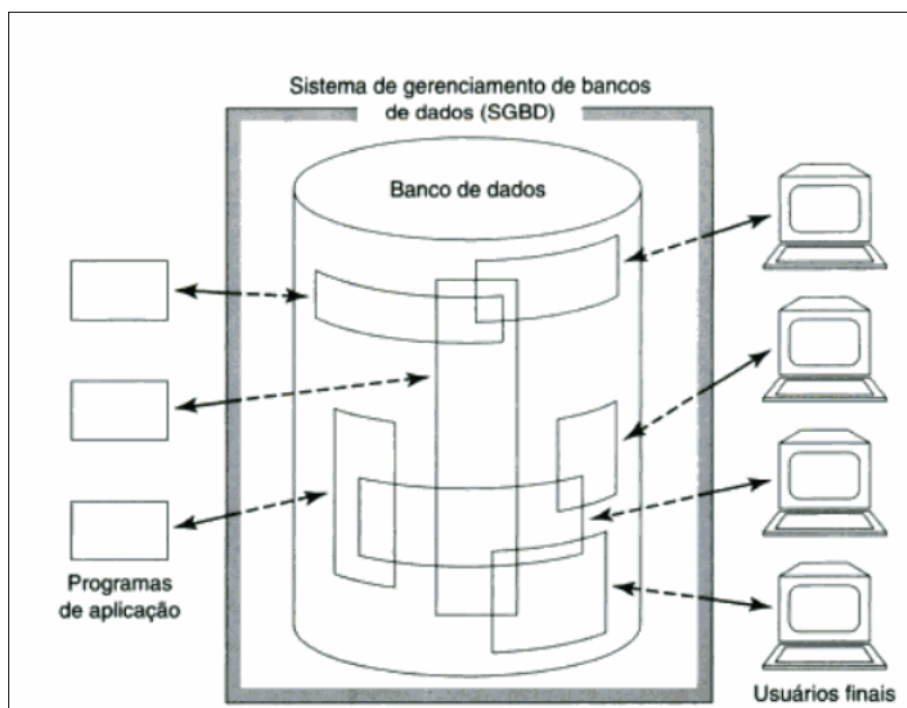


Figura 6. Representação de um Sistema de Banco de Dados apresentado por (DATE, 2004)

Com respeito ao *software*, ele é definido como Sistema Gerenciador de Banco de Dados (SGBD), que é uma coleção de programas que facilitam a construção e manipulação do banco, e desempenha funções citadas anteriormente, como por exemplo, pesquisa dentro do banco para recuperação de dados, atualização desses dados, exclusão e adição, o SGBD também garante proteção do sistema contra falhas tanto no *hardware* quanto no *software*, falhas na segurança (invasões) e também garante o compartilhamento do banco com outros de concorrente, para que assim o banco possa ser usado por vários, por exemplo, a empresa conhecida como Google utiliza um banco de dados onde várias pessoas o acessam de forma concorrente. Esse conjunto de programas também é capaz de se manter por muitos anos, se atualizando (ELMASRI et al., 2005).

Apesar de não ser uma regra, se é muito utilizado programas específicos para atuarem como SGBDs, como por exemplo o MySQL, PostgreSQL, SQL Server, MariaDB, entre outros presentes num grande leque de opções. Pode-se optar pela criação do próprio SGBD.

No caso desse projeto, foi selecionado o SGBD MySQL, uma vez que este foi escolhido pelo campus ao ensinar a matéria de Banco de Dados, e portanto é familiar em comparação as alternativas.

3. Desenvolvimento

Para iniciar o projeto foi necessário, primeiro, validar com especialistas na área de Libras e educação, estes mencionados na seção de 4.1, sobre a real necessidade de um sistema tutor para o aprendizado de Libras, assim como a real necessidade encontrada na população presente no grupo dos ouvintes em aprenderem o idioma. Esta validação foi feita a partir de entrevistas, onde foram discutidos sobre tópicos essenciais sobre aprendizado e cultura surda. Também foi necessário esboçar o projeto idealizado. Assim sendo, a seção 3.1 tem como fim abordar sobre os requisitos; na seção 3.2 é apresentado um Diagrama buscando um auxílio em sua visualização e entendimento do projeto; na seção 3.3 é possível se visualizar o modelo do banco de dados, na seção 3.4 se pode

encontrar protótipos de telas e por fim, na seção 3.6, as explicações referentes ao desenvolvimento, juntamente com as telas da versão final.

3.1. Requisitos

Almejando uma comunicação melhor entre cliente e desenvolvedor, uma vez que o primeiro pode falhar ao entender os termos mais técnicos usados pelo segundo assim como este pode acabar interpretando de maneira errônea os pedidos do cliente. Foi utilizado um método tendo como objetivo auxiliar a comunicação entre as duas partes. Conhecido como tabela de requisito, esta traduz os pedidos do cliente de uma forma clara usando uma base simples. Os requisitos estão representados nas tabelas de 1 a 8 localizadas no apêndice A.

3.2. Diagrama

Para uma melhor visualização de como o sistema funciona, foi desenvolvido um diagrama de caso de uso, conforme observado na figura 7.

Pode ser observado que o sistema funciona por meio de login e senha para que o progresso possa ser salvo na nuvem ao invés de realizar o salvamento via *cookies* do navegador. Para as funções principais, apenas uma conexão com o servidor é necessária como, por exemplo, progredir no jogo de *Flashcards*.

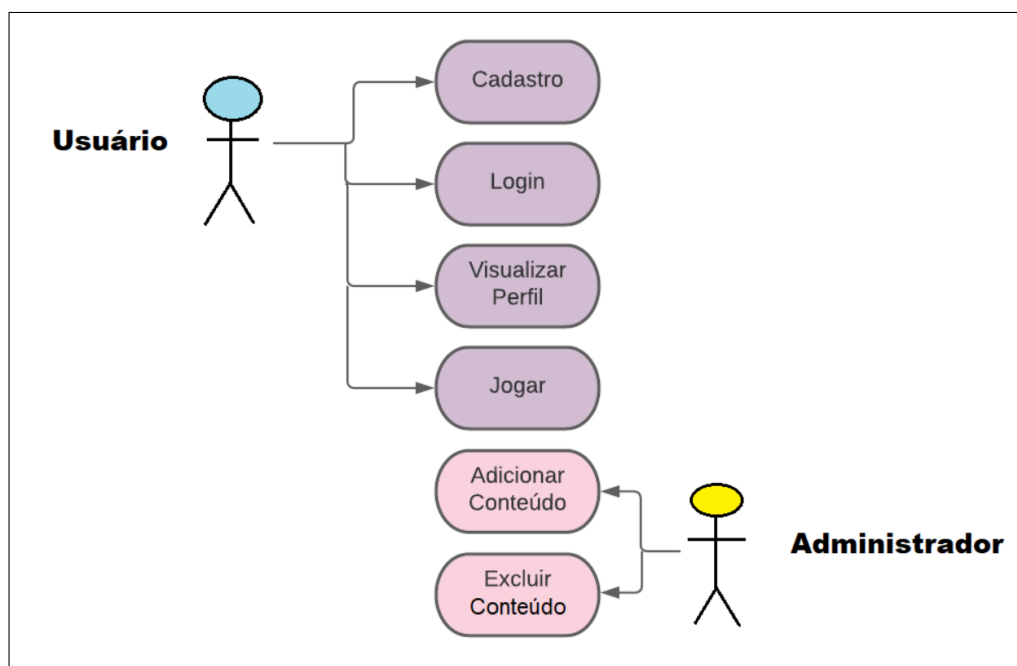


Figura 7. Diagrama de Caso de Uso

3.3. Modelo Banco de Dados

Nesse projeto, há a necessidade de utilização de um banco de dados, sendo que a organização dos caminhos é fundamental ao se trabalhar com um grande número de elementos, que vão de imagens e áudios, até vídeos formato mp4. Logo com finalidade de armazenar estes dados dos usuários e todo o conteúdo referente as atividades de Libras, a figura 8 apresenta o modelo Entidade Relacionamento do Banco de Dados e suas respectivas comunicações entre usuário e servidor de forma a ser visualizada com mais clareza por todos os partidos.

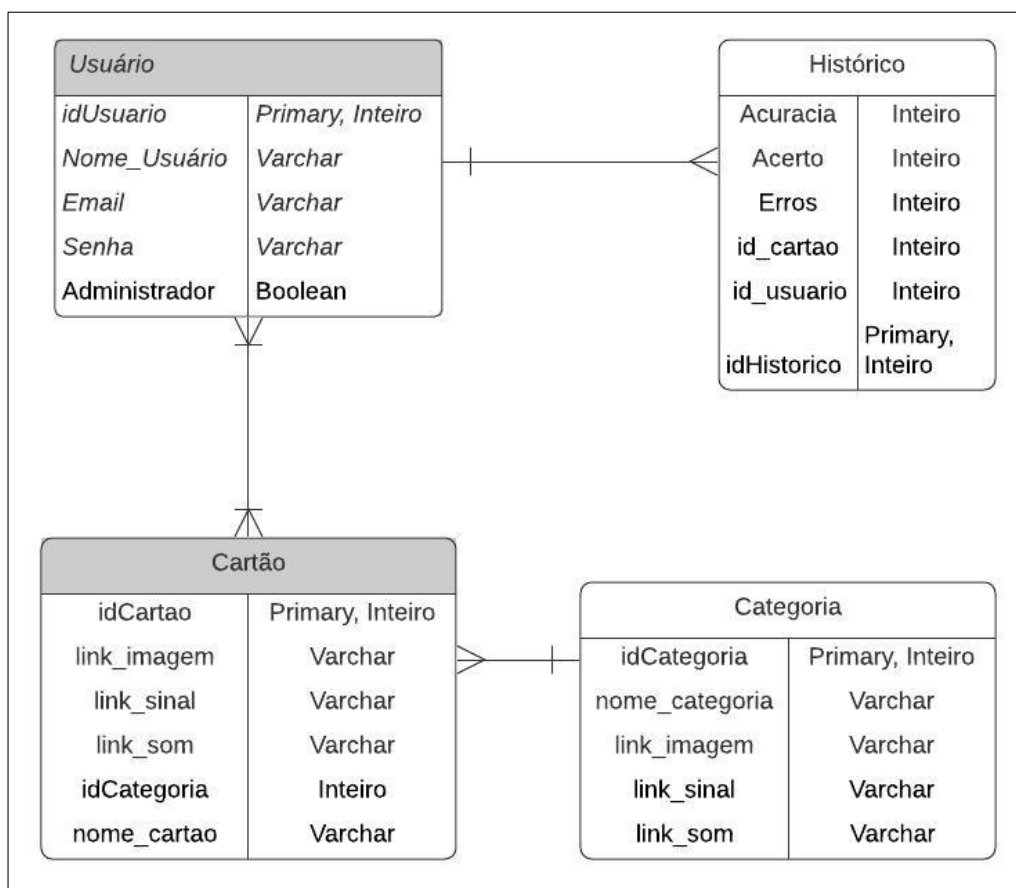


Figura 8. Diagrama Entidade e Relacionamento

3.4. Prototipação

Tendo como objetivo a obtenção de uma orientação focada primariamente em garantir o respeito ao desenvolvimento do trabalho, certificou-se a utilização de protótipos simples, uma vez que estes acarretam em inúmeras vantagens para o desenvolvimento, como por exemplo, uma prototipação requer menos tempo para ser concluída e quaisquer alterações podem ser feitas rapidamente. O protótipo garante que certos problemas de natureza desatenta por parte do programador sejam evitados. Na figura 9 é possível visualizar um protótipo da tela inicial do *software*, onde é possível visualizar as categorias selecionadas pela equipe com a ajuda de profissionais da área para garantir maior efetividade no aprendizado de sinais em Libras.

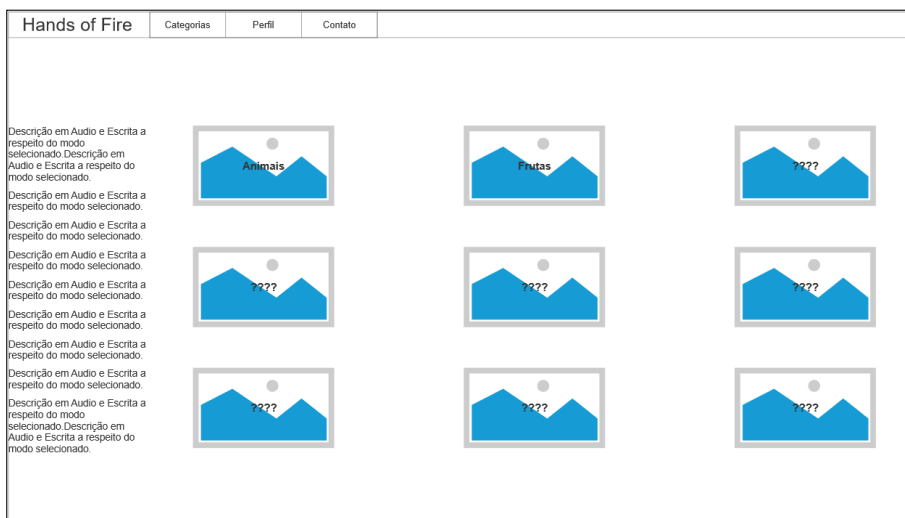


Figura 9. Protótipo Tela Principal

3.5. Tecnologias Utilizadas

Foram utilizadas várias tecnologias ao decorrer do projeto. Sejam para formar a base do software com HTTP, HTTPS e PHP. Para melhorar seu desempenho visual com CSS, HTML e Bootstrap. Ou até para o manuseio do projeto e seu banco com Java Script e MySQL. Estas tecnologias foram mencionadas com maior nível de detalhe na seção 2.5.

3.6. Implementação

A princípio, foi decidido que a maneira correta de alavancar o software seria trabalhar no que anteriormente seria uma tela de seleção de categorias, assim como em um menu na parte superior da página, conforme é mostrado na figura 10.



Figura 10. Tela de Seleção de Categorias

Contudo, foi notado um equívoco quanto à funcionalidade. Devido ao fato de ter sido projetada para computadores, a tela não conseguiria proporcionar uma utilização efetiva em dispositivos móveis. Logo, buscando maximizar o acesso ao aplicativo, grandes alterações foram feitas, assim a reformando tendo como objetivo a criação de um design minimalista, intuitivo e

com um tema escuro, essas alterações podem ser vistas na figura 11 a qual apresenta a tela principal do *software*. É para onde o usuário será primeiramente direcionado após efetuar seu login. Esta é composta por uma aba que permite ao usuário jogar usando como base uma categoria a sua escolha, um botão de "log out" para o encerramento da sessão do usuário, e um botão "Admin", que o redireciona para um ambiente ao qual é possível realizar alterações importantes sobre a aplicação. É importante notar que apenas usuários autorizados o conseguem visualizar. As categorias são mostradas de forma dinâmica conforme são cadastradas no banco de dados.



Figura 11. Tela de Seleção de Categorias - Revisada e Atualizada

Para implementar a função dos botões "log out" e "Admin" mostradas na figura 11, primeiramente foi necessária a implementação do cadastro e o login do usuário, assim como para realizar o log out da aplicação.

A página representada pela figura 12 apresenta a tela de cadastro, onde o usuário poderá criar sua própria conta. A Tela é composta por quatro campos e um link que redireciona o usuário para a tela de login caso já possua uma conta. Seus campos são Usuário, E-mail, Senha e Confirmação de Senha, nesta ordem. Para indicar se o usuário é ou não um administrador, um administrador precisa realizar o cadastro por meio do ambiente de administrador ou alterando os dados do usuário no próprio banco de dados.

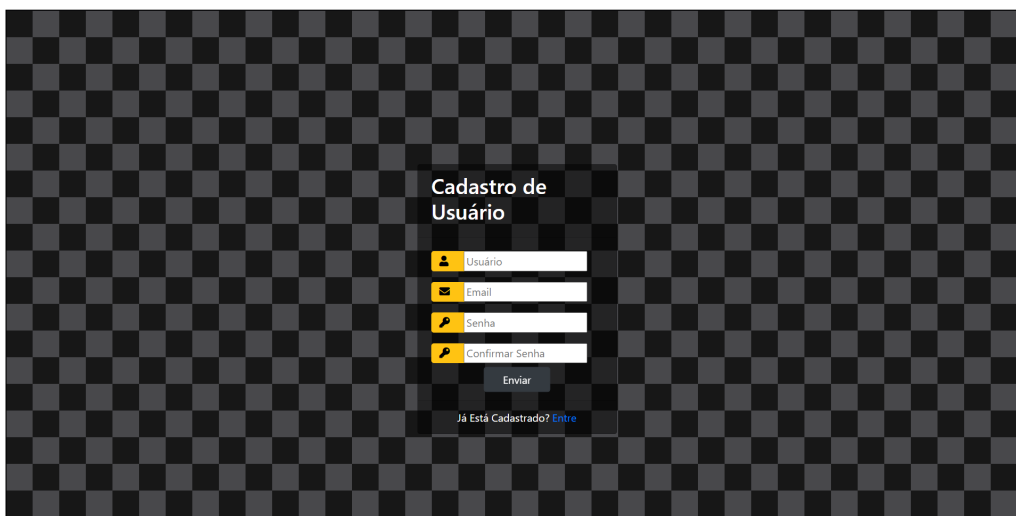


Figura 12. Tela de Cadastro de Usuário

A página representada pela figura 13 apresenta a tela de login. Composta por dois campos e um link que redireciona o usuário para a tela de cadastro caso não possua uma conta. Seus campos são Usuário e Senha. Tem como objetivo secundário verificar se o usuário é administrador ou não, o que vai decidir se esse tem direito de acessar o ambiente de administração. Com o objetivo de evitar que o usuário realize login toda vez que abre a aplicação, foi necessário o uso de um sistema de sessão presente no PHP, onde é armazenado informações do usuário em variáveis globais, as quais são verificadas para ignorar a tela de login e seguir diretamente para página de categorias.

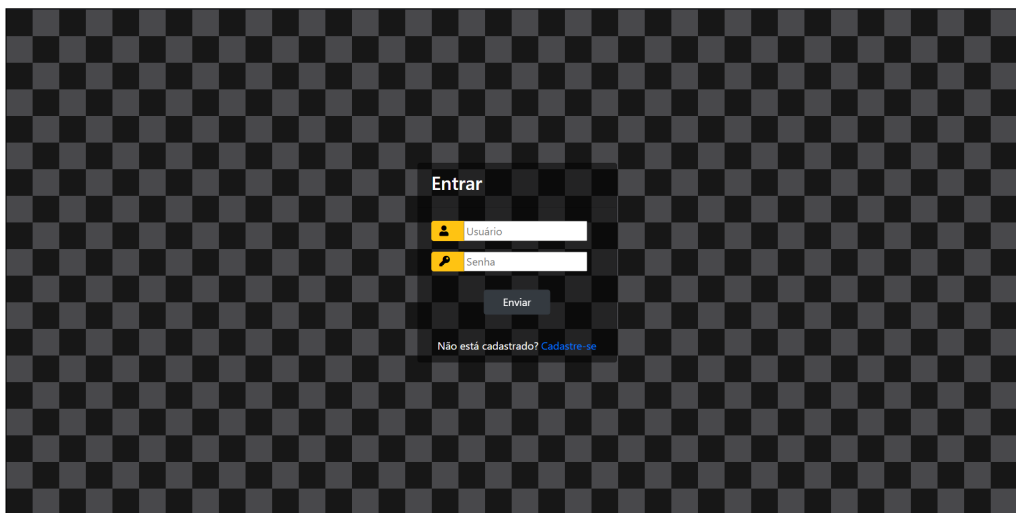


Figura 13. Tela de Login do Usuário

Na figura 14 é possível visualizar o código da página "index.php", de onde é feita a verificação da variável de sessão no início da página no formato php, assim como é possível observar o código responsável pelo direcionamento na parte inferior para a página de seleção de categorias, chamada de "stages.php".


```

<?php
session_start();
if (!isset($_SESSION['usuario'])) {
    $_SESSION['msg'] = "Você precisa estar logado primeiro";
    header('location: login.php');
}
if (isset($_GET['logout'])) {
    session_destroy();
    unset($_SESSION['usuario']);
    header("location: login.php");
}
?>
<!DOCTYPE html>
<html>

<head>
</head>

<body>
    <div class="header">
    </div>
    <div class="content">
        <!-- Mensagem -->
        <?php if (isset($_SESSION['success'])) : ?>
            <div class="error success">
                <h3>
                    <?php
                    echo $_SESSION['success'];
                    unset($_SESSION['success']);
                    ?>
                </h3>
            </div>
        <?php endif ?>
        <!-- Informações do Usuario -->
        <?php if (isset($_SESSION['usuario'])) :
            include 'stages.php';
        ?>
        <?php endif ?>
    </div>
</body>

```

Figura 14. Código index.php

O botão "admin" presente na tela de categorias, visível na figura 11, direciona o usuário para o ambiente do administrador. Conforme mostrado na figura 15, o ambiente é composto por quatro botões: "home" que o redirecionará a página principal de administrador, e três botões que permitem adicionar, procurar ou excluir administradores, categorias e cartões, nesta ordem.

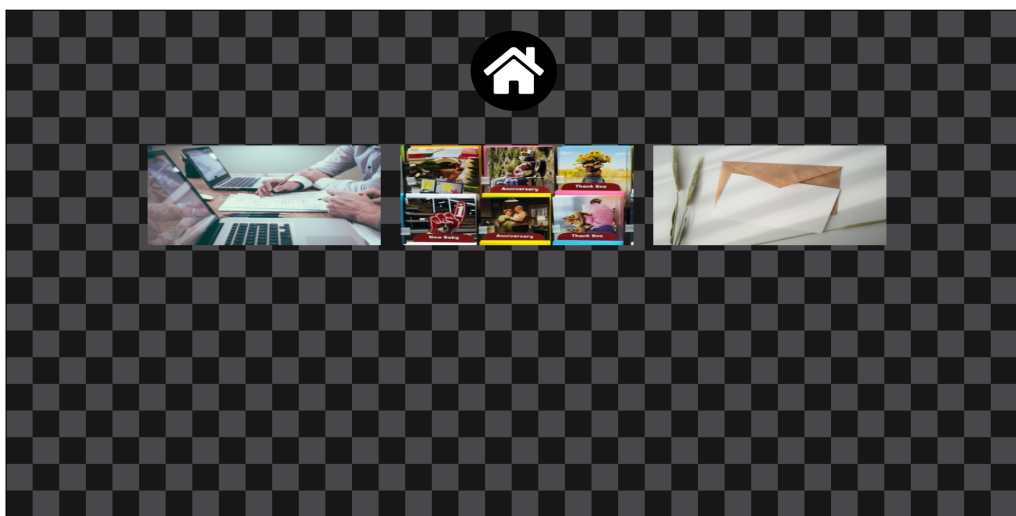


Figura 15. Tela de Ambiente do Administrador

Para evitar o manuseio interno do banco de dados, foram criados vários cadastros almeçando que, futuramente, mesmo um administrador sem conhecimentos de programação ou de banco de dados possa realizar cadastros de administradores, categorias e cartões, garantindo assim um crescimento da aplicação em termos de conteúdo por parte de sócios confiáveis, excluindo necessidade da participação de programadores e consequentemente expandindo o demográfico da equipe.

A página visível na figura 16 apresenta a tela de cadastro de cartas, exclusiva a administradores. Esta é composta por um campo de texto, três botões de envio e uma lista contendo as categorias cadastradas para atribuir o cartão a uma dessas categorias. Seu campo de texto pede o nome do cartão a ser adicionado, enquanto os envios pedem em ordem, imagem, vídeo e som do cartão, e finalizando com a atribuição da categoria.

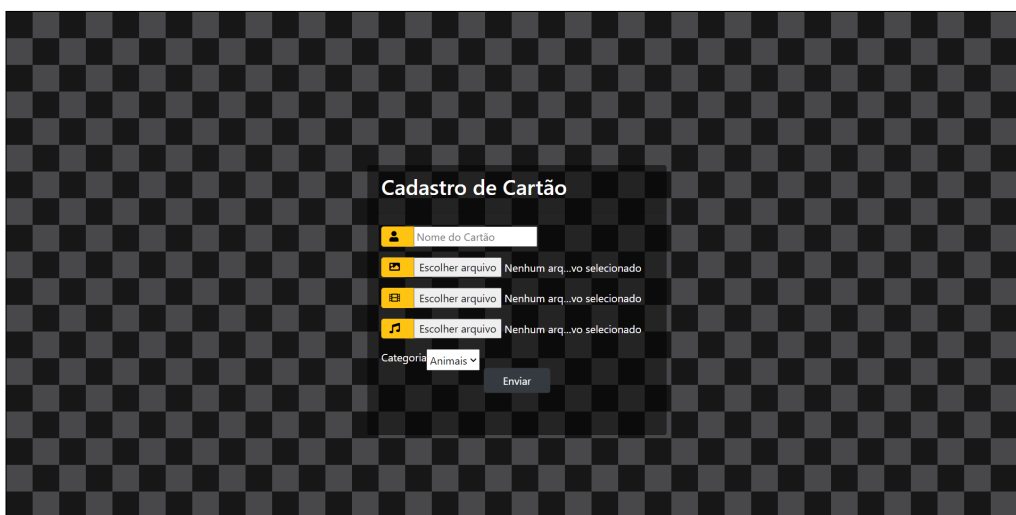


Figura 16. Tela de Cadastro de Cartões

A página visível na figura 17 apresenta a tela de cadastro de categorias. Esta é composta pelos mesmos elementos da tela de cadastro de cartas, excluída a lista contendo as categorias

cadastradas.

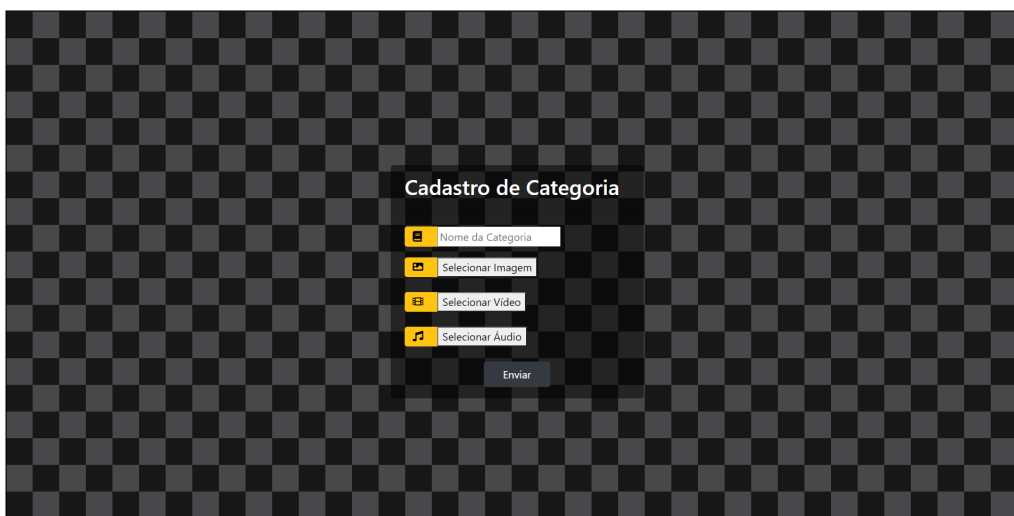


Figura 17. Tela de Cadastro de Categorias

A página visível na figura 18 apresenta a tela de pesquisa e exclusão de informações do banco, exclusiva a administradores. Esta apresenta um campo de pesquisa por nome, uma lista de dados, e um botão de exclusão. Cada dado tem sua própria página de pesquisa e exclusão.

Id	Cartão	Categoria	Operações
21	Cão	Animais	Excluir
22	Abelha	Animais	Excluir
23	Cavalo	Animais	Excluir
24	Cobra	Animais	Excluir
25	Abacaxi	Frutas	Excluir
26	Abacate	Frutas	Excluir
27	Laranja	Frutas	Excluir
28	Maçã	Frutas	Excluir
29	Morango	Frutas	Excluir
31	Galinha	Animais	Excluir

Figura 18. Tela de Pesquisa

A página apresentada pela figura 19 mostra a tela do jogo. Composta por um botão "home", um placar onde é possível visualizar a pontuação com o objetivo de facilitar a percepção de progresso, uma tela que tem como objetivo mostrar a pergunta, esta rotaciona uma vez que o usuário selecionar sua resposta, assim revelando a resposta correta. O painel de baixo contém quatro opções, dos quais o usuário deve escolher a resposta correta. Existem 2 formatos de jogo alternados, essa alternativa está indicada na figura 20.

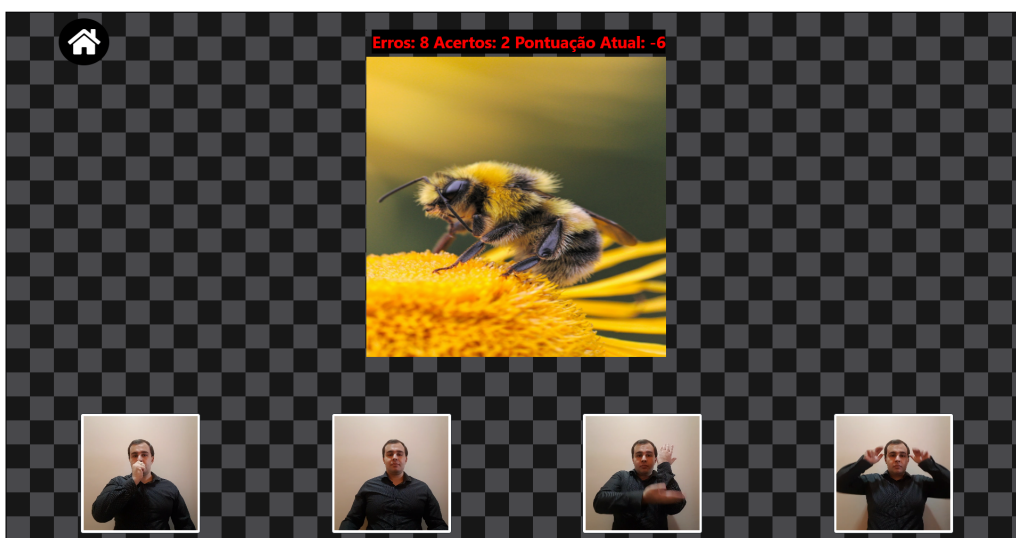


Figura 19. Tela de Jogo (Imagem - Sinal)

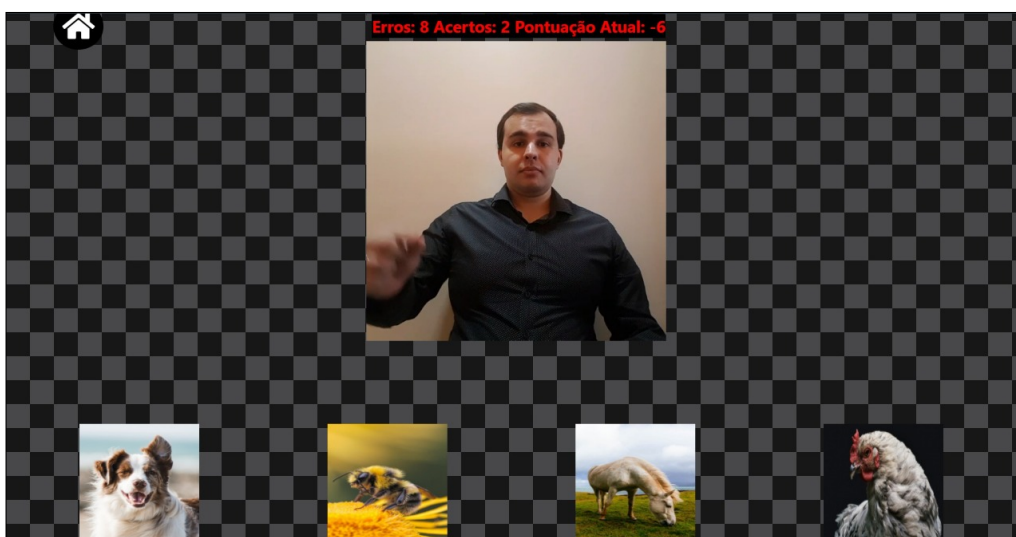


Figura 20. Tela de Jogo (Sinal - Imagem)

3.7. Metodologia

No início a equipe consultou professoras experientes em ensinar alunos surdos, citadas na seção 4.1 de agradecimento, de tal maneira a diminuir a lacuna da diferença cultural e de conhecimentos necessários para a adaptação a necessidade de alunos surdos.

Almejando consistência e eficiência, a equipe optou pela adoção de encontros semanais, uma vez que estes permitiram um desenvolvimento ágil e organizado para uma equipe de dois desenvolvedores, com o objetivo de discutir a direção e o progresso do projeto, e desenvolver atenciosas metas e objetivos a serem atingidos em um prazo selecionado pelos orientadores em conjunto dos desenvolvedores. A busca por consistência foi importante, visto que em decorrência da pandemia, foi necessária a adaptação dos encontros para um meio online. Meio este que pede esforços redobrados na busca por pontualidade.

4. Conclusão

Analisando a dificuldade de comunicação entre surdos e ouvintes, o trabalho de conclusão de curso tratou de buscar um meio que melhore a interação destes dois grupos, usando a tecnologia como meio para compartilhar a informação e aprendizado de modo de fácil absorção a partir da ludificação do conteúdo.

Visando a adaptação do sistema flashcard, juntou-se a sua natureza de aprendizado, com uma língua de sinais como Libras. Visando um público-alvo com pouco a nenhum conhecimento de Libras, o desenvolvimento foi focado em categorias primárias ao aprendizado, elementos que consigam prender a atenção facilmente, assim otimizando seu aprendizado e consequentemente atingindo o objetivo final do TCC de auxiliar o contato entre surdos e ouvintes por meio de uma facilitação na acessibilidade do conteúdo.

A cultura surda era completamente desconhecida por parte dos desenvolvedores. Logo, foram contatados vários especialistas da área, buscando assim compreender melhor toda esta cultura que aos poucos foi se tornando mais familiar, ajudando a equipe a se colocar no lugar de alguém diferente, com necessidades diferentes, facilitando a criação de um software web adaptado para tanto ouvintes como surdos igualmente.

Porém, naturalmente, houve dificuldades encontradas pela equipe. Originalmente não existia um conhecimento apropriado da Libras, e isso se provou um verdadeiro desafio para a dupla. Isto que coincidentemente se encaixou nos interesses da mesma, vendo que ambos desejavam se aprofundar na língua desde antes do projeto. Se utilizando do ensino a distancia proporcionado pelo IFSC-Florianópolis, o conhecimento da língua foi aprofundado, assim possibilitando a equipe o fomento do banco de dados sem a dependência de terceiros, em um momento que reunir uma equipe era inviável devido a contaminação do covid-19. Importante notar que isto acarretou em uma complicação ao encontrar referências, visto que a equipe não teve acesso à biblioteca.

É importante pontuar que nem todos os objetivos listado no início do projeto foram implementados, seja por problemas técnicos como a inabilidade de concluir o desenvolvimento da funcionalidade de recuperação de senha via e-mail, uma adaptação completa para dispositivos mobile, assim como uma melhora no tratamento de verificação de resolução de imagem e vídeos cadastrados, por conta de suas complexidades em comparação ao tempo disponível. Ou por uma mudança de prioridades, como a aparente necessidade inicial de permitir que o usuário tivesse a habilidade de visualizar e alterar seu perfil, algo que com uma visão mais madura e realista do projeto, foi chegada a conclusão de que não haveria a necessidade para tal.

Notando a existência de vários obstáculos, destaca-se o fato da equipe originalmente desconhecer as tecnologias e ferramentas web utilizadas neste projeto. Um obstáculo, que, como os outros se tornou passado ao se utilizar de cursos online, pesquisas intensivas e, principalmente, do apoio de toda uma comunidade. Permitindo assim que os objetivos para este projeto fossem atingidos. Uma vez que o software se concretizou em facilitar o aprendizado de forma eficiente para qualquer pessoa em busca de expandir seus horizontes.

Porém é necessário deixar claro que a aplicação ainda tem espaço para expansão, uma vez que a equipe focou em deixar o software acessível para quem queira trabalhar em cima do tal. Com a adição de administradores, o projeto pode ser adotado por uma equipe de pesquisa confiável, com o objetivo de expandir o banco de dados com infinitas possibilidades de cartões e categorias. Assim garantindo que o projeto não vá perder sua relevância com o tempo.

Além da questão da fomentação do banco, para trabalhos futuros a equipe propõe a inclusão de algumas funcionalidades que não foram implementadas, como a recuperação de senha via e-mail, a possibilidade de editar informações pelos administradores, um tratamento de

verificação mais rigoroso uma vez que o atual só filtra vídeos e imagens por tamanho total do arquivo e seu formato. Assim como possíveis novos métodos de ludificação do conteúdo e uma adaptação completa para dispositivos mobile.

4.1. Agradecimentos

Ao orientador Wilson Heck Junior, que apesar da intensa rotina de sua vida acadêmica aceitou nos orientar neste projeto. Seus conselhos esclareceram nossas maiores dúvidas no decorrer do trabalho, sempre transmitindo seu saber com muito profissionalismo.

À coorientadora Paula Clarice Santos Grazziotin de Jesus pela sua paciência e dedicação durante o projeto. Seus conhecimentos tanto na área de libras quanto na de português fizeram grande diferença no resultado final deste trabalho.

Gratidão pela participação das professoras e especialistas em libras Talita Milani Cordeiro e Greice Franklin da Silveira Lima, cujo apoio e atenção foram essenciais para a criação do escopo inicial para o projeto, assim como possibilitar uma visão mais madura sobre o ensino e aprendizado da língua.

Referências

- ANDREIS-WITKOSKI, S. (2011). *Educação de surdos e preconceito: bilinguismo na vitrine e bimodalismo precário no estoque. 2011, 255 f.* PhD thesis, Tese (Doutorado em Educação), Universidade Federal do Paraná, Curitiba.
- BENASSI, C. A. (2014). Da lei n. 10.436 de 24 de abril de 2002. *Revista Diálogos*, pages 22–25.
- BRASIL, Ministério da Educação, S. d. E. E. (2005). Decreto n 5.626, de 22 de dezembro de 2005. regulamenta a lei n 10.436, de 24 de abril de 2002.
- CANTO, F. H. (2011). Vulnerabilidades da linguagem php.
- DATE, C. J. (2004). *Introdução a sistemas de bancos de dados*. Elsevier Brasil.
- ELMASRI, R., Navathe, S. B., Pinheiro, M. G., et al. (2005). *Sistemas de banco de dados*, volume 6. Pearson Addison Wesley São Paulo.
- FERREIRA e EIS (2016). *W3C escritorio Brasil Curso HTML*.
- GUTIÉRREZ, J. J. (2014). ¿ qué es un framework web. Available in: http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf Accessed August, 12.
- HONORA, M. (2014). *Inclusão educacional de alunos com surdez: concepção e alfabetização*. Cortez Editora.
- HOSTINGER (2020). O que é html? guia básico para iniciantes. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-html-conceitos-basicos/>. Acesso em: 03 Dezembro 2020.
- LEMOIS, A. M. e CHAVES, E. P. (2012). A disciplina de libras no ensino superior: da proposição à prática de ensino como segunda língua. *XVI ENDIPE-Encontro Nacional de Didática e Práticas de Ensino–Campinas: UNICAMP*.
- LIE, H. W. e BOS, B. (1996). Cascading style sheets, level 1.
- MAGRINI, A. M. e dos SANTOS, T. M. M. (2014). Comunicação entre funcionários de uma unidade de saúde e pacientes surdos: um problema? *Distúrbios da Comunicação*, 26(3).
- MARINI, E. (2012). El modelo cliente/servidor. *Recuperado el*, 5.
- MARQUES, r. (2015). *Tags HTML*.
- MDN (2019). *Tecnologia Web JavaScript*.
- MORETO, S. (2016). *Bootstrap 4 By Example*. Packt Publishing Ltd.
- NINO, E. et al. (2012). Modelo cliente servidor.
- PHP (2020). O que é o php? Disponível em: https://www.php.net/manual/pt_BR/intro-what-is.php. Acesso em: 04 Junho 2020.

- PIMENTA, N. (2001). Oficina-palestra de cultura e diversidade. *Anais do Seminário do INES*, 19:24.
- QUADROS, R. M. d. (2006). Políticas linguísticas e educação de surdos em Santa Catarina: espaço de negociações. *Cadernos CEDES*, 26:141 – 161.
- RESCORLA, E. e SCHIFFMAN, A. (1999). The secure hypertext transfer protocol. *IETF Request for Comments, RFC*, 2660.
- SPURLOCK, J. (2013). *Bootstrap: Responsive Web Development*. "O'Reilly Media, Inc."
- STROBEL, K. (2009). História da educação de surdos. *Florianópolis: UFSC*.
- TUCKER, J. A. (1989). Basic flashcard technique when vocabulary is the goal. *Unpublished teaching materials, School of Education, University of Chattanooga. Chattanooga, TN: Author*.
- ZAKAS, N. (2012). *Professional JavaScript for Web Developers*. Wrox Press.
- ZIRGUEZI (2012). A graphical description of the leitner system.

A. Requisitos

RF1	Estabelecer conexão
Descrição	O sistema deve enviar a solicitação de conexão e obter a chave de acesso para a aplicação a partir do servidor Hands of Fire.
Prioridade	Alta
Entradas	Conexão com a internet.
Saída	Uma chave de acesso.
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema envia a solicitação do acesso ao Hands of Fire. 3. O sistema recebe uma chave de acesso do Hands of Fire. 4. O sistema armazena a chave de acesso.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa ao usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 3, o sistema não recebe a chave de acesso. Caso esse problema ocorra, é solicitado que o usuário verifique as configurações de sua conta.

Tabela 1. Tabela de Requisitos Nº1- Estabelecer conexão

RF2	Logar Usuário
Descrição	O sistema deve efetuar o login do usuário no sistema web Hands of Fire. Para efetuar o login é necessário que a aplicação tenha estabelecido a conexão com o Hands of Fire.
Prioridade	Alta
Entradas	Conexão com a internet, chave de acesso, login e senha do usuário.
Saída	O usuário logado no sistema
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema recebe o login e senha do usuário. 3. O sistema envia os dados para o Hands of Fire e espera a configuração. 4. O sistema recebe e armazena o id do usuário.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 3, o sistema informa que os dados do usuário não estão corretos. A aplicação solicita que o usuário verifique os dados e tente novamente.

Tabela 2. Tabela de Requisitos N°2- Logar Usuário

RF3	Armazenar Progresso
Descrição	A aplicação deve salvar o progresso do usuário.
Prioridade	Alta
Entradas	Conexão com a internet, progresso por parte do usuário.
Saída	O progresso será salvo.
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema apresenta a aplicação ao usuário. 3. O sistema vai moldar a experiência para cada usuário diferentemente. 4. O sistema recebe e armazena o progresso e peculiaridades do usuário.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 3, o sistema informa que os dados do usuário não estão corretos. A aplicação solicita que o usuário verifique os dados e tente novamente.

Tabela 3. Tabela de Requisitos N°3- Armazenar Progresso

RF4	Apresentar Pontuação
Descrição	O sistema deve receber uma pontuação baseada nos acertos do usuário referente aos sinais e apresentá-la.
Prioridade	Alta
Entradas	Conexão com a internet, progresso no aplicativo.
Saída	Pontuação representada.
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema envia a solicitação do acesso ao Hands of Fire. 3. O sistema recebe a pontuação do usuário. 4. O sistema mostra a pontuação para o usuário.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa ao usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 3, caso o sistema não receba a pontuação do Hands of Fire, a pontuação não será representada e a mensagem de aviso será enviada.

Tabela 4. Tabela de Requisitos N°4- Apresentar Pontuação

RF5	Processar Pontuação
Descrição	O sistema deve receber uma pontuação baseada nos acertos do usuário referente aos sinais e reagir de acordo.
Prioridade	Alta
Entradas	Conexão com a internet, progresso no aplicativo.
Saída	Sistema Flashcard.
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema envia a solicitação do acesso ao Hands of Fire. 3. O sistema recebe a pontuação do usuário. 4. O sistema calcula o grau de memorização de cada sinal. 5. O sistema requisita sinais de acordo com a pontuação.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa ao usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 3, caso o sistema não receba a pontuação do Hands of Fire o serviço será temporariamente terminado.

Tabela 5. Tabela de Requisitos N°5- Processar Pontuação

RF6	Personalização de Categorias
Descrição	O sistema deve receber as categorias escolhidas pelo usuário com fim de criar uma personalização pessoal para o usuário.
Prioridade	Média
Entradas	Conexão com a internet, escolha de categorias.
Saída	Cards customizados para cada usuário.
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema envia a solicitação do acesso ao Hands of Fire. 3. O sistema recebe a escolha das categorias. 4. O sistema seleciona as categorias no banco de dados.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa ao usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 3, caso o sistema não receba a escolha do usuário, a personalização não será completada.
Fluxo Secundário 3	No fluxo principal 4, caso não seja possível acessar o banco de dados do servidor, o sistema informa ao usuário sobre o problema e solicita que ele tente novamente mais tarde.

Tabela 6. Tabela de Requisitos N°6- Personalização de Categorias

RF7	Perda de Senha
Descrição	Caso o usuário esqueça sua senha, o sistema deve garantir que o usuário consiga alterá-la .
Prioridade	Média
Entradas	Conexão com a internet, pedido de troca de senha, senha nova.
Saída	O usuário recupera sua conta com uma nova senha.
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema envia a solicitação do acesso ao Hands of Fire. 3. O sistema recebe o pedido de troca de senha pelo usuário. 4. O sistema pede email da conta. 5. O sistema envia o pedido de troca por email. 6. O sistema recebe e atualiza o servidor com a nova senha.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa ao usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 4, caso o usuário não saiba o seu email a operação será cancelada.
Fluxo Secundário 3	No fluxo principal 5, o sistema proporciona um botão de reenvio para o caso de o email não ser entregue.

Tabela 7. Tabela de Requisitos N°7- Perda de Senha

RF8	Modo Noturno
Descrição	Botão que varia a página entre seu design normal e um mais escuro e gentil aos olhos.
Prioridade	Média
Entradas	Conexão com a internet, ativação do botão modo noturno.
Saída	O software entra/sai do modo noturno.
Fluxo de Eventos	
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema verifica a conexão com a internet. 2. O sistema envia a solicitação do acesso ao Hands of Fire. 3. O sistema pede a troca de design. 4. O sistema troca de design.
Fluxo Secundário 1	No fluxo principal 1, caso não seja possível acessar a internet, o sistema informa ao usuário sobre o problema e solicita que ele tente novamente mais tarde.
Fluxo Secundário 2	No fluxo principal 4, caso o design não seja trocado, o sistema envia uma mensagem de erro.

Tabela 8. Tabela de Requisitos N°8- Modo Noturno