

ARTHUR CORREIA BRÍGIDO
BRUNA LIMA FARIAS
GABRIEL MARCONDES RIBAS

Prancha de Comunicação Alternativa com Predição de Figuras

Versão 2.0

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Cientista da Computação.

Área de concentração: Ciência da Computação

Orientador: Prof. Dr. Percy Nohama

Curitiba
2024

ÍNDICE DETALHADO

PREFÁCIO	4
1. REQUISITOS DO SISTEMA	9
1.1. REQUISITOS FUNCIONAIS	9
1.2. REQUISITOS NÃO-FUNCIONAIS	10
1.3. PROTÓTIPO	10
1.4. FUNCIONALIDADES	12
1.5. PREDIÇÃO POR LOCAL E TEMPO	12
1.6. PREDIÇÃO POR MARKOV	13
1.7. INTERFACE	14
1.8. BACKEND	14
1.9. TESTES	14
1.10. MODELO DE DADOS	16
1.10.1. <i>Modelo Lógico da Base de Dados</i>	16
1.10.2. <i>Criação Física do Modelo de Dados</i>	16
2. IMPLEMENTAÇÃO	17
2.1. MANUAL DE IMPLANTAÇÃO	19
3. MANUAL DO USUÁRIO	20
4. CONCLUSÕES E CONSIDERAÇÕES FINAIS	21
BIBLIOGRAFIA	22

Prefácio

A etapa de prototipação do site da prancha de comunicação é fundamental para o desenvolvimento deste projeto inovador. Neste contexto, vamos mostrar modelos que representam o design e a funcionalidade da prancha, permitindo uma visualização e teste prévio das suas características. Esses protótipos serão essenciais para validar decisões de design, identificar possíveis melhorias e coletar feedback dos usuários. Dessa forma, poderemos garantir que a prancha de comunicação seja intuitiva, eficaz e inclusiva, atendendo às necessidades e expectativas daqueles que a utilizarão.

Estamos comprometidos em realizar uma prototipação detalhada e interativa, visando aprimorar continuamente a experiência do usuário e as funcionalidades da prancha. Vamos embarcar nesta jornada com entusiasmo e determinação, buscando criar uma solução de comunicação verdadeiramente transformadora.

1. Requisitos do Sistema

Os requisitos para a prancha de comunicação incluem tanto requisitos funcionais quanto requisitos não-funcionais. Os requisitos funcionais se referem às funcionalidades específicas que a prancha deve oferecer, como a predição de palavras eficiente e a interface intuitiva. Já os requisitos não-funcionais dizem respeito a características mais amplas do sistema, como compatibilidade com dispositivos, interface intuitiva, velocidade de processamento e eficiência na predição de palavras. Esses requisitos são essenciais para garantir uma experiência de comunicação acessível, eficaz e intuitiva para os usuários da prancha.

Segue o plano de requisitos em porcentagem.

Predição Local e Tempo 25%	Predição Markov 25%	Interface 20%	Backend 15%	Testes 15%	
Desenvolvimento de algoritmo de predição de palavras que levam em consideração o contexto local do usuário, como localização geográfica e horário do dia (25%)	Pesquisa e implementação de modelos de Markov para prever palavras (25%)	Design centrado no usuário para criar uma interface intuitiva e amigável (25%)	Integração do código de predição por local e tempo no site (25%)	Planejamento e de testes de usabilidade com usuários finais para identificar problemas e áreas de melhoria na prancha	Feito
Ajustar a predição com base em padrões de uso específicos do usuário em diferentes momentos e locais (25%)	Desenvolvimento do algoritmo de predição que aprendem e se adaptam ao usuário ao longo do tempo (25%)	Mostrar figuras e adicionar figuras à lista da frase formada (25%)	Integração do código de predição por markov no site (25%)	Coleta de feedback para avaliar a eficácia, a eficiência e a satisfação do usuário	Em parte
Integrar predição com as palavras e figuras (25%)	Melhorar a precisão e a relevância das previsões (25%)	Deletar frase e ultima figura da frase formada, fazer botões para cada ação (25%)	Desenvolvimento da infraestrutura necessária para suportar a funcionalidade da prancha de comunicação (25%)	Realizar ajustes e melhorias na interface e na funcionalidade com base nos feedbacks	Ainda não feito
Melhorar a precisão e a relevância das previsões (25%)	Integrar predição com as palavras e figuras (25%)	Implementação de elementos de design visualmente interessantes e coerentes para melhorar a usabilidade e experiência do usuário (25%)	Implementação de sistema de gerenciamento de dados para armazenar e processar as informações da prancha. (25%)	Testes para ajuste e verificação	

Figura 1.1 - Plano de Requisitos do Projeto

1.1. Requisitos Funcionais

- Seleção de Figuras: Os usuários devem poder selecionar figuras na prancha para

compor suas mensagens ou expressar suas necessidades.

- Predição de Figuras por Markov: A prancha deve oferecer a função de predição de figuras, sugerindo automaticamente figuras com base no contexto e no histórico de uso do usuário, utilizando Markov.
- Predição de Figuras Por Local e Tempo: A prancha deve oferecer a função de predição de figuras, sugerindo as automaticamente com base no horário e no local do usuário.
- PWA: O usuário tem a possibilidade de baixar uma versão do site em aplicativo, podendo ser usado tanto online como offline.

1.2. Requisitos Não-Funcionais

- **Compatibilidade com Dispositivos:** A prancha deve ser projetada para ser acessível em uma ampla gama de dispositivos, incluindo computadores desktop, laptops, tablets e smartphones. Isso significa que a interface da prancha deve ser responsiva e capaz de se adaptar a diferentes tamanhos de tela e sistemas operacionais, garantindo uma experiência consistente e sem problemas em qualquer dispositivo utilizado pelo usuário.
- **Interface Intuitiva:** A interface da prancha deve ser projetada com foco na usabilidade e na acessibilidade. Isso inclui a organização lógica das opções de comunicação, a utilização de ícones e símbolos compreensíveis, além de uma navegação clara e direta. A linguagem utilizada na interface deve ser simples e fácil de entender, facilitando a interação para pessoas com diferentes níveis de habilidade cognitiva e linguística.
- **Velocidade de Processamento:** A prancha deve ser capaz de processar as solicitações do usuário de forma rápida e eficiente. Isso inclui a resposta instantânea ao toque ou clique em uma opção de comunicação, bem como a capacidade de carregar rapidamente novas figuras ou páginas conforme necessário. Uma alta velocidade de processamento é essencial para garantir uma comunicação fluida e sem interrupções.
- **Predição de Palavras Eficientes:** A função de predição de palavras da prancha deve ser precisa e eficaz, antecipando com precisão as palavras que o usuário deseja comunicar com base no contexto e no histórico de uso. Isso requer algoritmos avançados de predição de texto que levem em conta não apenas as palavras individuais, mas também os padrões de linguagem e as intenções de comunicação do usuário.

1.3. Protótipo

A etapa de prototipação do site da prancha de comunicação é fundamental para o desenvolvimento deste projeto inovador. Neste contexto, vamos mostrar a criação de modelos interativos que representam o design e a funcionalidade da prancha, permitindo uma visualização e teste prévio das suas características.

Funcionalidades

1. Predição por Local e Tempo

- **Descrição:**

Implementação de algoritmos de predição de palavras/figuras que levam em consideração o contexto local e temporal do usuário. Utiliza dados de localização geográfica e horário do dia para ajustar dinamicamente as previsões de palavras com base nos padrões de uso específicos do usuário.

- Funcionalidades Principais:

Predição de palavras com base na localização geográfica atual do usuário.

Ajuste da predição de palavras com base no horário do dia.

Personalização das previsões de palavras de acordo com as atividades do usuário em diferentes momentos e locais.

- Tecnologias Utilizadas:

Linguagem de programação: Java Script

Api: Geoapify

- Fluxo de Dados:

Captura dos dados de localização e horário do usuário.

A figura abaixo se refere ao fluxograma relacionado a predição de local e tempo.

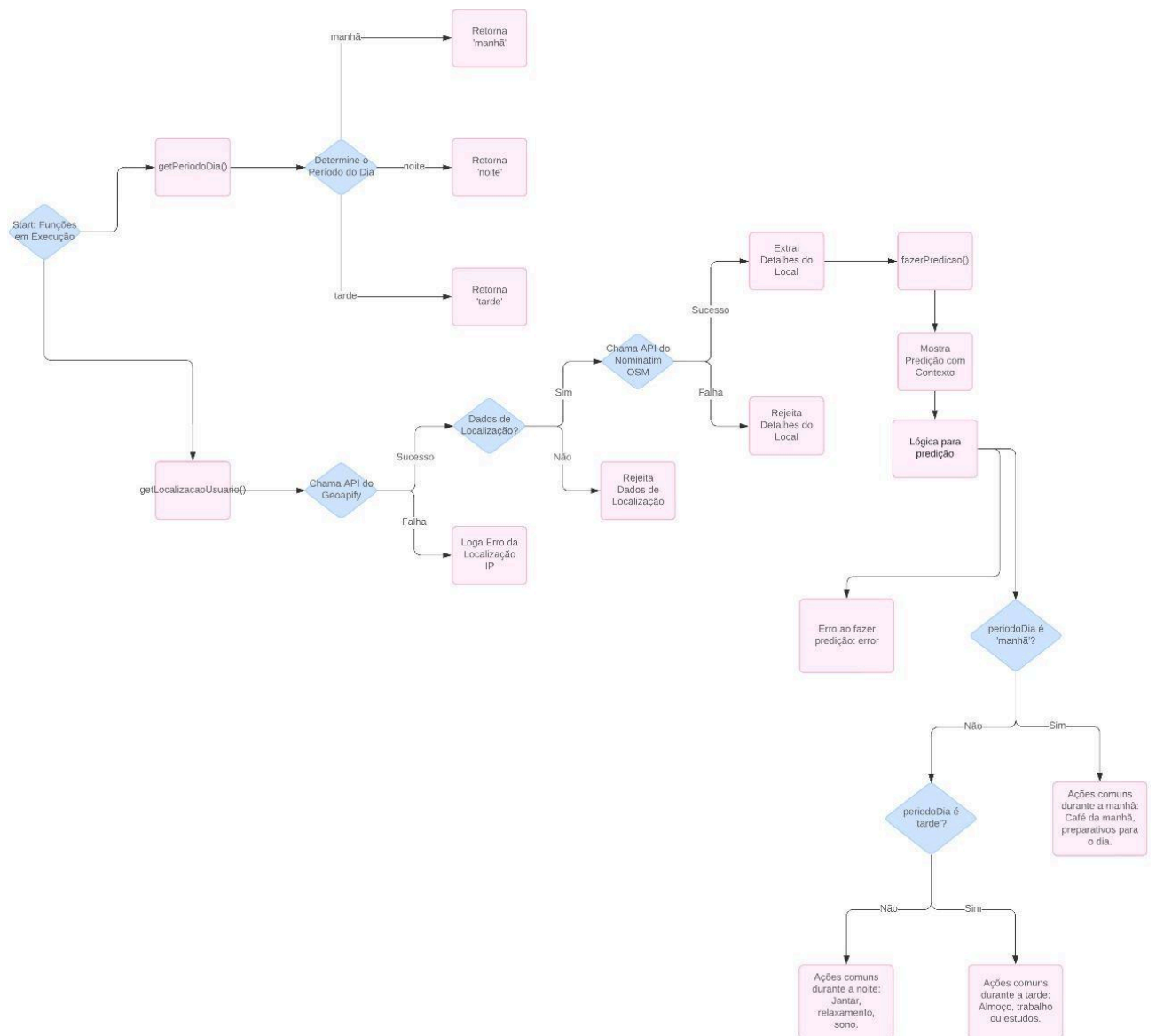


Figura 1.2 - Predição por local e tempo

2. Predição por Markov

- Descrição:

Este módulo implementa modelos de Markov para prever palavras com base em sequências de palavras anteriores. Utiliza técnicas de aprendizado de máquina para aprender e se adaptar ao estilo de linguagem do usuário ao longo do tempo.

- Funcionalidades Principais:

Modelagem de transições de palavras com base em dados de histórico de uso.

Aprendizado automático de padrões de linguagem do usuário.

Atualização contínua do modelo de Markov com novos dados de entrada.

- Tecnologias Utilizadas:

Linguagem de programação: Java Script

Bibliotecas:

- Fluxo de Dados:

Coleta de dados de entrada, como histórico de palavras digitadas.

Treinamento do modelo de Markov com os dados de entrada.

Geração de previsões de palavras com base no modelo treinado.

A figura abaixo se refere ao fluxograma relacionado a predição de Markov.

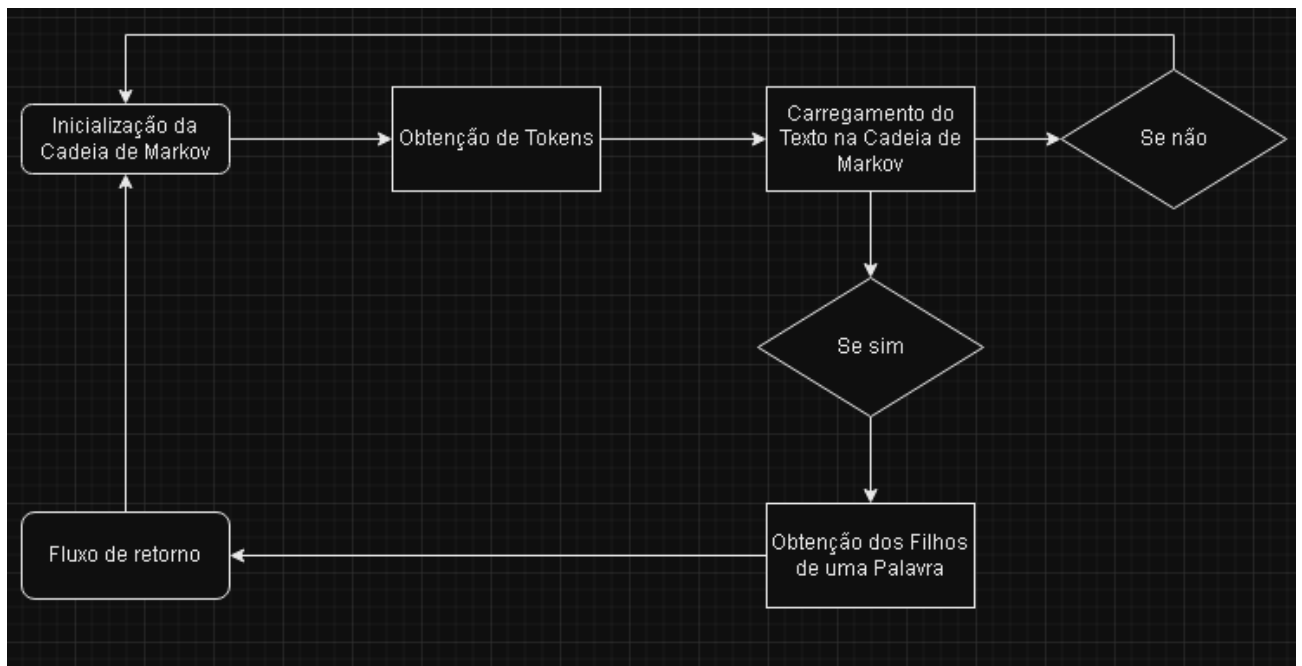


Figura 2 - Predição por cadeia de Markov

3. Interface

- Descrição:

Este módulo implementa a interface do usuário para a prancha de comunicação. É responsável por criar uma experiência de usuário intuitiva e amigável, permitindo que o usuário interaja facilmente com o sistema.

- Funcionalidades Principais:

Design de interface centrado no usuário.

Implementação de elementos visuais atraentes e coerentes.

Personalização da interface para atender às preferências individuais dos usuários.

- Tecnologias Utilizadas:

Linguagem de programação: HTML, CSS, JavaScript

Frameworks: React.js, Bootstrap

- Fluxo de Dados:

Recebimento de entradas do usuário, como toques na tela ou cliques do mouse.

Processamento das entradas para determinar ações correspondentes.

Atualização dinâmica da interface para refletir as interações do usuário.

A figura abaixo se refere ao fluxograma relacionado a interface.

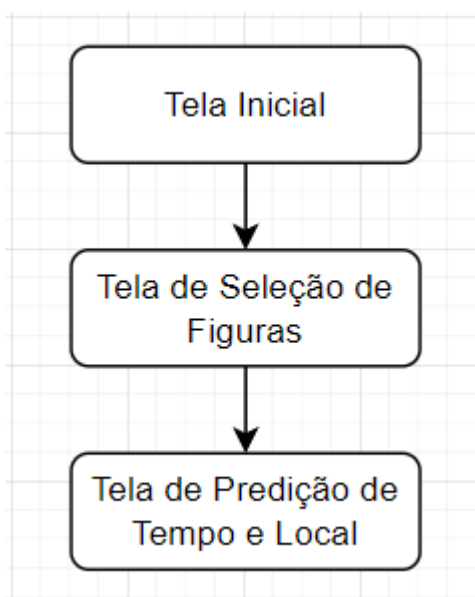


Figura 3 - Fluxograma da Interface do Projeto

4. Backend

- Descrição:

Este módulo implementa a lógica de negócios e a infraestrutura de back-end para suportar o funcionamento da prancha de comunicação. Gerencia o armazenamento de dados, a autenticação de usuários e a comunicação com outros sistemas.

- Funcionalidades Principais:

Desenvolvimento de APIs para interação com o front-end e outros sistemas.

Implementação de sistemas de armazenamento de dados robustos e seguros.

Site com possibilidade de criar arquivo PWA a partir de si mesmo no navegador Chrome.

Integração de serviços externos, quando necessário, para fornecer funcionalidades adicionais.

- Tecnologias Utilizadas:

Linguagem de programação: Python, Node.js

- Fluxo de Dados:

Recebimento de solicitações do front-end por meio de APIs RESTful.

Processamento das solicitações para realizar operações no banco de dados.

Retorno de dados processados para o front-end e outros sistemas.

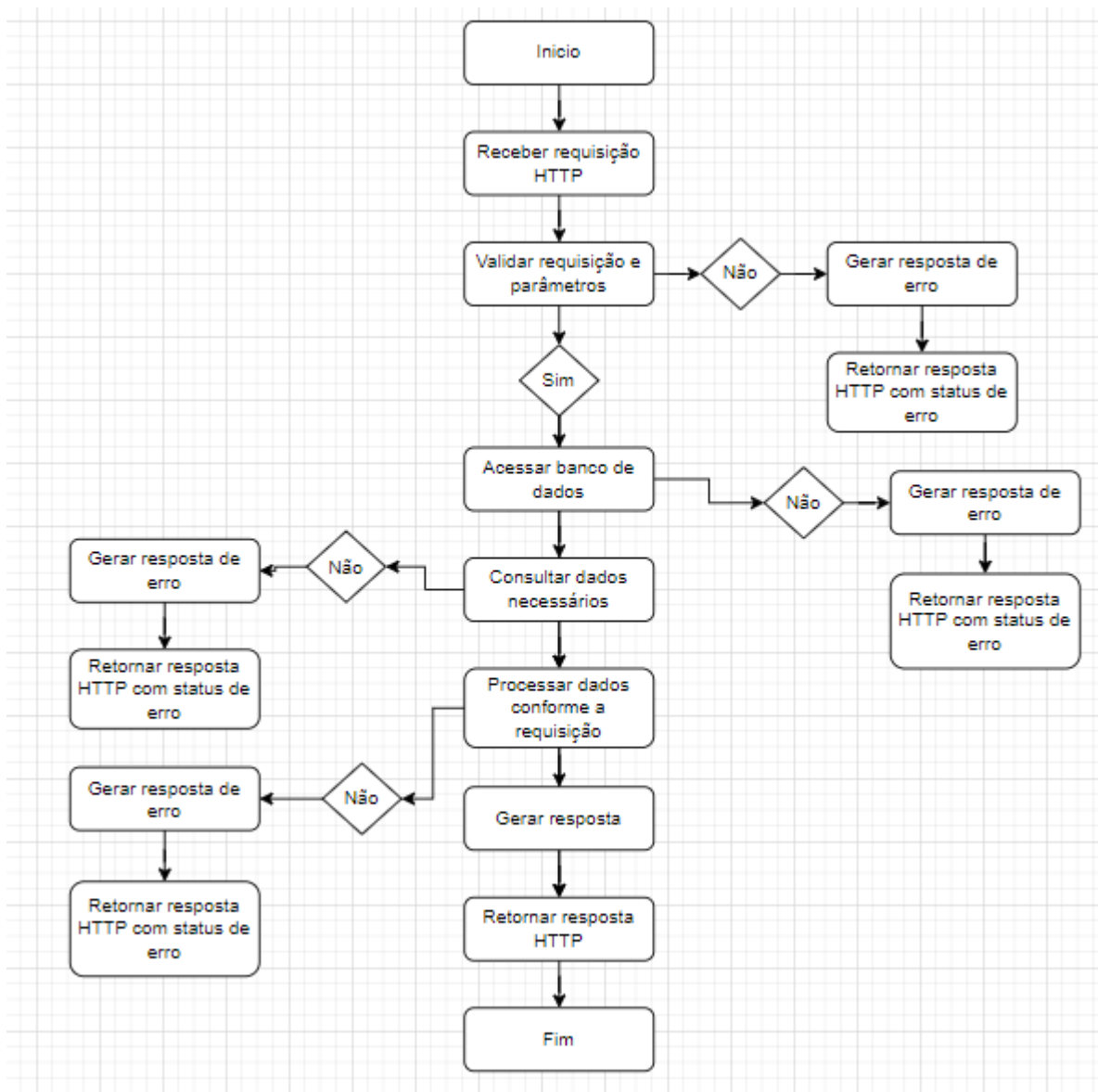


Figura 4 - Fluxograma do Backend do Projeto

5. Testes

- Descrição:

Serão implementados casos de teste automatizados para garantir a qualidade e a robustez do sistema. Os testes serão realizados em diferentes níveis, incluindo testes de unidade e integração para validar o funcionamento correto do sistema em diferentes cenários.

- Funcionalidades Principais:

Criação de casos de teste para cada funcionalidade do sistema.

Execução automatizada dos casos de teste em ambientes controlados.

Avaliação dos resultados dos testes e identificação de possíveis problemas

Detalhamento dos casos de Teste:

- Teste de Unidade (Seleção de Figuras):

Verificar se a funcionalidade de seleção de figuras está funcionando corretamente.

Selecionar uma figura da lista de opções.

Verificar se a figura selecionada é exibida corretamente na tela principal.

Critérios de Aceitação: A figura selecionada deve ser exibida corretamente na tela principal após a seleção.

- Teste de Integração (Predição por Markov):

Verificar se a funcionalidade de predição por Markov está integrada corretamente ao sistema.

Inserir uma sequência de palavras anteriores.

Verificar se o sistema gera previsões de palavras com base na sequência inserida.

Critérios de Aceitação: O sistema deve gerar previsões de palavras precisas com base na sequência de palavras inseridas.

- Teste de Aceitação (Interface):

Verificar se a interface do usuário está funcionando conforme o esperado.

Navegar pelas diferentes telas da aplicação.

Interagir com os elementos visuais da interface, como botões e menus.

Critérios de Aceitação: A navegação deve ser fluida e os elementos da interface devem responder corretamente aos cliques e interações do usuário.

- Tecnologias Utilizadas:

Framework de testes: Jest

- Fluxo de Dados:

Definição dos casos de teste e dos critérios de aceitação.

Execução dos testes em diferentes partes do sistema.

Análise dos resultados dos testes e geração de relatórios de cobertura.

1.3.1. Diagrama de Navegação

Este diagrama mostra a sequência básica de navegação entre as telas da aplicação, permitindo ao usuário formar frases utilizando seleções de figuras e palavras previstas.

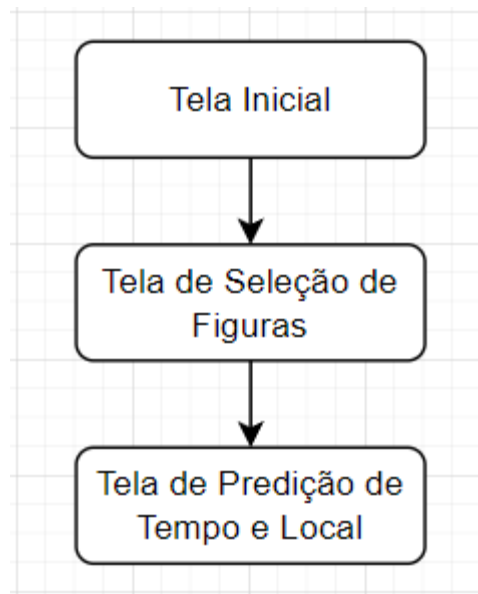


Figura 5 - Navegação pela aplicação

1. Início:

O usuário inicia na tela inicial da aplicação, onde pode ver a frase que está sendo formada e as opções de figuras para seleção.

2. Seleção de Figuras:

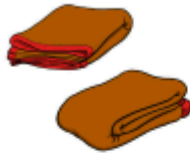
O usuário pode selecionar uma figura da lista de opções disponíveis. Após a seleção, a figura é adicionada à frase em formação na tela inicial.

3. Predição de Tempo e Local:

O usuário pode acessar a função de predição de palavras com base no tempo e local atual. Aqui, o sistema oferece sugestões de palavras que o usuário pode querer usar na sua frase. O usuário pode selecionar uma dessas palavras para adicionar à frase em formação na tela inicial.

A figura abaixo se refere à página web do protótipo do site sobre a Prancha de Comunicação.

Frase Formada:



cama

cobertor



Predição Tempo e Local:

Usuário está a noite.

Ações comuns durante a noite: Jantar, relaxamento, sono.

Fazer Predição

Figura 6 - Interface da Prancha de Comunicação

Categorias:

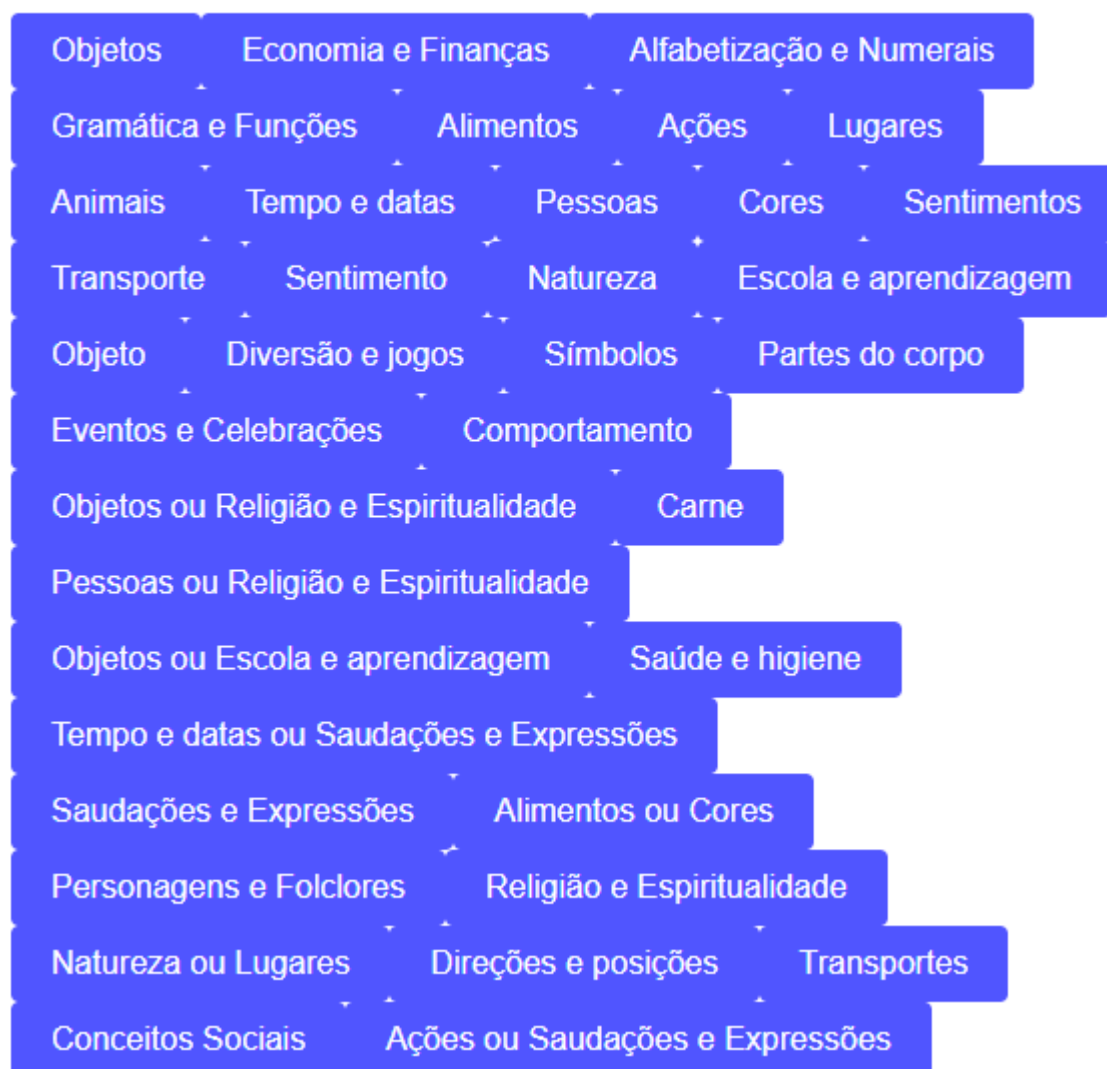


Figura 7 - Interface da Prancha de Comunicação



© 2023 Amplisoft. Todos os direitos reservados.

Figura 8 - Interface da Prancha de Comunicação

1.4. Modelo de Dados

1.4.1. Criação Física do Modelo de Dados

```
DROP DATABASE IF EXISTS prancha_comunicacao;
CREATE DATABASE prancha_comunicacao;
USE prancha_comunicacao;

CREATE TABLE lista_figuras (
    id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    palavra VARCHAR(100) NOT NULL,
    img VARCHAR(255) NOT NULL
);

CREATE TABLE categorias (
    id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    nome VARCHAR(100) NOT NULL
);

CREATE TABLE figura_categoria (
    figura_id INT,
    categoria_id INT,
    PRIMARY KEY (figura_id, categoria_id),
    FOREIGN KEY (figura_id) REFERENCES lista_figuras(id),
    FOREIGN KEY (categoria_id) REFERENCES categorias(id)
);
```

Figura acima se refere ao modelo da base de dados criado para a Prancha de Comunicação.

Versão final usando 2.002 figuras.

2. Implementação

2.1. Manual de Implantação

Este manual fornece instruções passo a passo para a instalação do sistema em um ambiente local.

Requisitos:

Sistema operacional compatível: Windows, Linux ou macOS.

XAMPP e MySQL Server instalados e configurados corretamente.

Passos para Instalação:

- Instalação do XAMPP:

Baixe o instalador do XAMPP para o seu sistema operacional a partir do site oficial: <https://www.apachefriends.org/index.html>.

Execute o instalador baixado e siga as instruções na tela para concluir a instalação.

- Configuração do Apache:

Após a instalação do XAMPP, abra o painel de controle do XAMPP.

Inicie os serviços Apache clicando nos botões "Start" ao lado de cada serviço.

- Instalação do MySQL Server:

Baixe o instalador do MySQL Server para o seu sistema operacional a partir do site oficial da MySQL: <https://dev.mysql.com/downloads/mysql/>.

Execute o instalador baixado e siga as instruções na tela para concluir a instalação.

- Configuração do Banco de Dados:

Após a instalação do MySQL Server, abra o prompt de comando ou terminal e inicie o serviço MySQL.

Faça login no MySQL Server como usuário root ou com privilégios administrativos.

Use o Prancha/src/script.sql do projeto como base para o banco.

Baixe os arquivos do sistema e extraia-os para o diretório htdocs dentro da pasta de instalação do XAMPP. Este é o diretório onde os arquivos PHP são servidos pelo Apache.

Abra um navegador da web e digite o seguinte endereço na barra de endereços: http://localhost/nome_do_diretorio_do_sistema. Substitua nome_do_diretorio_do_sistema pelo nome do diretório onde os arquivos do sistema foram extraídos.

Siga as instruções na tela para concluir a instalação do sistema. Durante o processo de instalação, você será solicitado a fornecer as informações de conexão com o banco de dados que configurou anteriormente.

- Conclusão:

Após a conclusão da instalação, o sistema estará disponível para uso. Você poderá acessá-lo através do navegador da web digitando o endereço `http://localhost/nome_do_diretorio_do_sistema`.

3. Conclusões e Considerações Finais

Nesta pesquisa, exploramos o desenvolvimento e prototipação de um site para a prancha de comunicação com predição de figuras destinada a auxiliar pessoas com dificuldades na fala na comunicação. Durante o desenvolvimento do projeto, várias lições importantes foram aprendidas, incluindo:

A importância de uma colaboração eficaz entre membros da equipe para alcançar os objetivos do projeto.

A necessidade de adaptabilidade e flexibilidade ao lidar com desafios técnicos e de design inesperados.

A valorização do feedback dos usuários como uma ferramenta essencial para melhorar continuamente a experiência do usuário.

Melhorias Futuras:

Algumas melhorias que podem ser consideradas para futuras iterações do projeto incluem:

Implementação de um sistema de personalização mais avançado, permitindo que os usuários personalizem completamente a interface de acordo com suas preferências.

Integração de técnicas de aprendizado de máquina mais avançadas para melhorar ainda mais a precisão da predição de palavras.

Expansão da base de dados de figuras para incluir uma variedade maior de opções, tornando a prancha de comunicação ainda mais abrangente e útil para os usuários.

Bibliografia

ALMEIDA, I., MOREIRA, A., RIBEIRO, J. High-Tech Augmentative and Alternative Communication Devices: Observing Children's Need for Help and Interaction with Caregivers.

AMPLISOFT – Projeto Amplisoft de software livre. Disponível em:
<<https://amplisoft.azurewebsites.net>> Acesso em 04 abr. 2024.

JORDAN, M. Predição de palavras baseada em modelos ocultos de Markov. Curitiba: Universidade Tecnológica Federal do Paraná, 2007. 188p.

MANZINI E. J., DELIBERATO D. Recursos para comunicação alternativa. DF, 2004, 3 p. ISBN 85-86738-26-3

RABINER, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. IEEE, 1989, 286p.

ROTH, V. Prancha de comunicação alternativa por imagens empregando predição de palavras. Curitiba: Universidade Tecnológica Federal do Paraná, 2019. 92p.

STELLA, G. Deficiências Cognitivas e Uso dos Procedimentos Informatizados. In: TUPY, T.M, PRAVETTONI, D.G. ...e se Falta a Palavra, qual a comunicação, Qual Linguagem? São Paulo: Memnon, 1999.

YOUNG, S., EVERMANN, G., GALES, M., HAIN, T., KERSHAW, D., LIU, X., ... & WOODLAND, P. The HTK book (for HTK version 3.4). Cambridge University Engineering Department, 2006.

ZIBETTI, A. Probabilidade e Estatística para Engenharias utilizando o R (RStudio). Universidade Federal de Santa Catarina, 2024. Disponível em:
<<https://www.inf.ufsc.br/~andre.zibetti/probabilidade/cadeias-de-markov.html>> Acesso em: 04 abr. 2024.

MCNAUGHTON, D., & LIGHT, J. The iPad and mobile technology revolution: Benefits and challenges for individuals who require augmentative and alternative communication, 2013. 29(2), 107-116.

BINGER, C., & LIGHT, J. The effect of aided AAC modeling on the expression of multi-symbol messages by preschoolers who use AAC. Augmentative and Alternative Communication, 2007. 23(1), 30-43.