

**CENTRO UNIVERSITÁRIO DA SERRA GAÚCHA**  
**CENTRO DE INOVAÇÃO E TECNOLOGIA**  
**CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE**  
**SISTEMAS/SEGURANÇA DA INFORMAÇÃO**

**RAMON CARDOSO DE OLIVEIRA**

**TRANSCRITOR DE PODCAST**

**CAXIAS DO SUL**

**2021**

Ramon Cardoso de Oliveira

## TRANSCRITOR DE PODCAST

Artigo apresentado como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, pelo Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Centro Universitário da Serra Gaúcha – FSG

Orientador(a): Prof. Me. Luís Felipe Schilling

Caxias do sul

2021

## Resumo

Com a crescente expansão do público de *podcast* nos últimos anos, ocorreu uma grande mudança na forma de consumir o conteúdo em áudio, trazendo uma imensa quantidade de tópicos como política, cultura pop, esportes, notícias, áudio dramas, tecnologia e assuntos acadêmicos. No entanto, há muita dificuldade em procurar um conteúdo, que já foi escutado dentro de algum episódio por meio de pesquisa textual. Tendo esse problema em vista foi elaborada uma aplicação *web* que possibilita catalogação de podcast e posteriormente a sua transcrição, assim facilitando a busca do mesmo via texto, por algo que foi mencionado diretamente dentro do áudio do programa em questão.

Palavras-chave: *podcast*, aplicação *web*, transcrição.

### **Abstract**

With the growing expansion of podcast audiences in recent years, there has been a huge shift in the way audio content is consumed, bringing a huge amount of topics such as politics, pop culture, sports, news, audio dramas, technology and academics. However, it is very difficult to search for content that has already been heard within an episode through textual research. With this problem in mind, a web application was created that enables podcast cataloging and later its transcription, thus facilitating the search for it via text, for something that was mentioned directly within the audio of the program in question.

Keywords: podcast, web application, transcription.

**LISTA DE FIGURAS**

Figura 1 - Diagrama de caso de uso: Cadastro de Catálogo de podcast.	17
Figura 2 - Diagrama de caso de uso: Busca de podcast.	18
Figura 3 - Fluxo do cadastro de catálogo de podcast.	19

## LISTA DE QUADROS

Quadro 1 - Requisito número 1: Catalogar episódios de podcast.	15
Quadro 2 - Requisito número 2: Pesquisa de <i>podcast</i> a partir das transcrições catalogadas.	16
Quadro 3 - Requisito número 3: Indexar as transcrições em motores de busca.	16
Quadro 4 - Requisito número 4: Fazer as transcrições dos podcasts automatizadas.	16

## LISTA DE ABREVIATURAS E SIGLAS

ABPOD	Associação Brasileira de Podcasters
RSS	Really Simple Syndication
IBOPE	Instituto Brasileiro de Opinião Pública e Estatística
SEO	Search Engine Optimization
UML	Unified Modeling Language
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
CSS	Cascading Style Sheets
API	Application Programming Interface
REST	Representational State Transfer
IOT	Internet of Things
MVP	Minimum Viable Product
CRUD	Create Read Update Delete
IBM	International Business Machines
ORM	Object Relational Mapper

## SUMÁRIO

<b>Introdução</b>	<b>7</b>
Questão de Pesquisa	7
Objetivos	8
Objetivo Geral	8
Objetivos Específicos	8
<b>Referencial Teórico</b>	<b>9</b>
Diagrama caso de uso UML	9
Engenharia de requisitos	9
Especificação de Caso de Uso	10
Aplicação Web	10
.Net Core	10
Rest API	11
IBM Watson speech to text	11
Microsoft SQL Server	11
Sonic	11
<b>Metodologia</b>	<b>12</b>
Pesquisa de viabilidade	12
Início da pesquisa	12
Pesquisa de serviços speech to text	12
Levantamento de requisitos	13
Diagrama de casos de uso	13
Especificação do caso de uso	13
Tecnologias usadas no desenvolvimento da aplicação web	13
Serviço de pesquisa(Sonic)	14
<b>Resultados e discussões</b>	<b>15</b>
Levantamento de requisitos	15
Diagrama de caso de uso	17
Especificação do caso de uso	18
Aplicação web	18
Arquitetura	19
Fluxos das funcionalidades	19
Limitações e restrições	20
Monetização	20
<b>Conclusões e Considerações finais</b>	<b>21</b>



## 1. Introdução

*Podcasts* ou *Podcasting* são programas de áudio que podem ser baixados da Internet por sites de seus idealizadores, por *feed RSS* a partir de aplicativos como Google Podcast, Castbox, Addict, etc, ou até reproduzidos por meio de serviços de *streaming* como Spotify, Deezer, Amazon Music entre outros. Organizados em uma série de episódios, os podcasts podem abranger uma imensa gama de assuntos, como política, cultura pop, esportes, notícias, áudio dramas, tecnologia, assuntos acadêmicos, enfim uma ampla série de conteúdo. É possível ouvir os episódios individualmente ou assinar o programa por meio de *feed RSS* ou serviços de *streaming*. Em sua grande maioria os *podcast* são disponibilizados gratuitamente.

O primeiro *podcast* criado no Brasil foi o Digital Minds, criado por Danilo Medeiros em 2004, no ano atual(2021) da publicação deste trabalho, não existe uma pesquisa que indica a quantidade total de programas de *podcast* em atividade, porém se estima que existam mais de 2 mil programas ativos no Brasil segundo ABPOD (Associação Brasileira de Podcasters). De acordo com Pod PESQUISA 2020 realizado pela Associação Brasileira de Podcasters (ABPOD), se estima que no Brasil tenha 34,6 milhões de ouvintes de podcast. Ou seja, quase 8% da população brasileira. Em outra pesquisa é informado que quatro em cada dez usuários de internet brasileiros já ouviram algum programa de *podcast* e mais da metade destas 56 milhões de pessoas são ouvintes mensais de *podcasts*, esses são números referentes a pesquisa realizada pelo IBOPE Inteligência em Abril de 2019.

Conforme foi apontado acima, é possível analisar que existe uma grande oferta de conteúdo dentro da mídia *podcast*, possuindo uma vasta diversidade de tópicos que podem ser abordados. Pelo fato de haver inúmeras maneiras de consumir esse conteúdo, por meio de *streaming*, *feed RSS* ou *download*, ocorrendo assim uma descentralização da mídia, assim tornando difícil rastrear os conteúdos, principalmente o que é abordado dentro do áudio do podcast, pois os episódios possui apenas uma breve descrição do que exposto dentro do mesmo.

### 1.1. Questão de Pesquisa

Com a crescente chegada de novos usuários para essa mídia, e também uma grande quantidade de informação contida dentro dos áudios dos episódios, cada vez mais fica latente a dificuldade de procurar um conteúdo que já foi escutado dentro de algum programa, por

meio de pesquisa textual nos buscadores como Google. **O que poderia ser feito para facilitar a procura desses conteúdos?**

## **1.2. Objetivos**

Desenvolver uma aplicação *web* em *.net Core*, onde será possível catalogar episódios de *podcast* contendo uma transcrição do áudio do mesmo. Assim facilitando a pesquisa sobre assuntos discutidos dentro de episódios específicos.

### **1.2.1. Objetivo Geral**

Implementar uma aplicação *web*, onde será possível a qualquer usuário catalogar ou cadastrar um episódio de *podcast* e anexar um áudio do mesmo. Após o episódio ser cadastrado, a aplicação *web* irá transcrever o áudio por meio de inteligência artificial(IBM Watson speech to text), na conclusão da transcrição será feito o relacionamento entre o texto gerado a partir do áudio e o cadastro do episódio.

### **1.2.2. Objetivos Específicos**

Otimizar o SEO (Search Engine Optimization ou otimização para mecanismos de busca) das páginas da aplicação *web*, para que os buscadores(Google, duckduckgo, Bing etc) indexarem o conteúdo dos podcasts transcritos, de maneira mais rápida. Obtendo um SEO bem feito, também faz com que o ranqueamento nas pesquisas seja mais alto.

## 2. Referencial Teórico

Este capítulo apresenta a revisão bibliográfica para dar apoio a compreensão dos conceitos envolvidos no desenvolvimento desse projeto. Para a produção deste trabalho serão necessários os conhecimentos de engenharia de *software*, desenvolvimento de aplicações *web*, manipulação *api REST*.

### 2.1. Diagrama caso de uso UML

O Diagrama de caso de uso na a linguagem de modelagem unificada ou UML, resume os detalhes dos usuários de um sistema que também são conhecidos como atores, e as interações dos mesmos com o sistema. Esse diagrama documenta o que o sistema faz do ponto de vista do usuário, descrevendo as principais funcionalidades do sistema e a interação dessas funcionalidades com os usuários do mesmo sistema.

Os diagramas de Casos de Uso são representados graficamente por desenhos, sendo uma elipse contendo internamente, o nome do caso de uso, e por sua vez o ator é representado por um boneco palito. Não é obrigatório o diagrama possuir uma fronteira, que determina o sistema, onde os casos de usos estarão representados dentro da fronteira e os atores fora da mesma.

### 2.2. Engenharia de requisitos

A engenharia de requisitos é um processo que integra todas as dinâmicas que apoiam a produção de um documento de requisitos e sua continuidade na manutenção do projeto ou sistema. Segundo Wilson de Pádua Paula Filho esse processo compreende oito atividades de alto nível:

- Revisão do processo de requisitos;
- Revisão das demandas dos usuários;
- Esclarecimento das demandas dos usuários;
- Distribuição das tarefas de requisitos;
- Documentação dos requisitos;
- Revisão dos requisitos;
- Revisão dos requisitos pelos usuários;
- Colocação dos requisitos na linha de base.

Este processo necessita ser implementado depois de um estudo de viabilidade técnica e financeira, com base nas restrições do projeto, comprovando se é viável, prosseguir com o projeto e sua identificação dos requisitos.

### 2.3. Especificação de Caso de Uso

Caso de uso é um procedimento de identificação que descreve um passo a passo de ações que o sistema deve empreender para produzir uma resposta para um ator do sistema. Efetivamente o que ocorre é uma sequência da interação entre caso de uso e ator. O caso de uso detalha o que um sistema deve fazer, descrevendo como uma funcionalidade é usada pelo ator.

A especificação abrange duas partes: o diagrama de caso de uso e o caso de uso propriamente dito. O diagrama de caso de uso UML foi apresentado no item anterior (**2.1 Diagrama caso de uso UML**), enquanto, que o caso de uso consiste de um template (ou modelo), que serve para detalhar uma sucessão de passos de execução do mesmo.

### 2.4. Aplicação Web

Na área da Tecnologia da Informação, a aplicação *web* consiste em sistemas de informática elaborados para uso através de um navegador (*Browser*), por meio da internet (seja intranet ou extranet ) ou aplicativos que utilizam em seu desenvolvimento tecnologias da *web* como: *HTML*, *JavaScript* e *CSS*. Pode ser executado a partir de um servidor *HTTP* ou localmente, no dispositivo do usuário, como por exemplo o programa de computador *Web uTorrent*.

### 2.5. .Net Core

O .NET *Core* é uma plataforma de desenvolvimento de *software* livre gratuita para a criação de vários tipos de aplicativos, como: Aplicativos Web, *APIs* da *Web*, *microservices*, computação em nuvem, *IoT* entre outras opções. É desenvolvida e mantida pela *Microsoft* em conjunto com a comunidade de desenvolvedores.

O .NET dá suporte a três linguagens de programação: *C#*, *F#* e *Visual Basic*. Também possui ambientes de desenvolvimento integrados, são eles: *Visual Studio*, *Visual Studio Code*, *Visual Studio* para *Mac*, *Codespaces* do *GitHub*

## 2.6. Rest API

Para compreender o conceito de *Rest API*, é preciso explicar o que é uma *API*. Refere-se a uma composição de requisições que permite a comunicação de dados entre aplicações diferentes.

*API Rest* utiliza requisições *HTTP* usando seus verbos (*POST*, *GET*, *DELETE*, *PUT*), para executar operações básicas para a manipulação dos dados. As principais requisições são:

- *POST*: inserir dados;
- *GET*: leitura;
- *DELETE*: exclusão;
- *PUT*: atualizações de registros.

## 2.7. IBM Watson speech to text

O *IBM Watson Speech to Text* é um serviço de *cloud* que proporciona a transcrição de áudio por meio de uma API. As Tecnologias para converter áudio em texto são baseados no reconhecimento de fala da *IBM* e podem ser utilizados com diferentes tipos de áudio, como vídeos, arquivos de áudio e *streaming* de áudio, também apresenta modelos que reconhecem e identificam diferentes vozes.

## 2.8. Microsoft SQL Server

Microsoft SQL Server trata-se de um gerenciador de Banco de dados relacional ou SGBD, desenvolvido e distribuído pela Microsoft. Possuindo uma grande integração com o ecossistema da Microsoft

## 2.9. Sonic

*Sonic* é um serviço de pesquisa e indexador, muito parecido com o famoso *Elasticsearch* (outro serviço de pesquisa). É capaz de normalizar consultas de pesquisa em linguagem natural, preenchendo automaticamente uma consulta de pesquisa e fornecendo os resultados mais relevantes para uma consulta. Sonic é um índice de identificador, em vez de um índice de documento; quando consultado, ele retorna *IDs* que podem ser usados para se referir aos documentos correspondentes em um banco de dados externo.

### 3. Metodologia

Este capítulo apresenta as metodologias de desenvolvimento do projeto. Devido ao trabalho se tratar de um produto/*MVP* irá ser descrito o processo de desenvolvimento, desde a pesquisa de viabilidade até a escolha das tecnologias utilizadas para implementar o *MVP*.

#### 3.1. Pesquisa de viabilidade

Pelo fato do projeto possuir apenas duas funcionalidades, sendo em sua grande maioria *crud* simples, por esse motivo não foi realizado uma pesquisa sobre uma tecnologia específica para o desenvolvimento de um *crud*, pois se trata de algo simples.

No entanto, conforme foi descrito no **Objetivo Geral**, haverá uma transcrição de áudio automatizada, para essa funcionalidade foi feita uma pesquisa extensa, tanto com leitura e testes práticos.

##### 3.1.1. Início da pesquisa

No começo da pesquisa foi empenhado um foco específico em encontrar, um serviço de transcrição de áudio ou uma biblioteca de alguma linguagem de programação que possibilitaria a transcrição. Após essa pesquisa não foi identificado nenhuma biblioteca ou linguagem de programação que executasse um processo de transcrição de áudio.

Contudo foi encontrado um termo genérico designado para essa técnica, que é conhecida no mercado como "*speech to text*", que em uma tradução literal tem significado de "fala para texto". A partir dessa descoberta, o foco da pesquisa foi direcionado para encontrar um serviço que possui a funcionalidade *speech to text*.

##### 3.1.2. Pesquisa de serviços *speech to text*

Após procura foi identificado três principais serviços de *speech to text*, que dão suporte a integração a linguagens de programação por meio de *Api*, são elas:

- **IBM Watson *Speech to Text***: Desenvolvida pela IBM, possui preço acessível e uma qualidade de transcrição mediana.
- **Azure *Speech to Text***: Desenvolvida pela Microsoft, possui preço mediano e uma qualidade de transcrição mediana.
- **Google Cloud *Speech-to-Text***: Desenvolvida pela Google, possui preço elevado e uma qualidade de transcrição alta.

Usando esses dois critérios, custo e qualidade, no término da pesquisa foi escolhido o serviço de *speech to text* da IBM, ou seja o **IBM Watson Speech to Text**.

### 3.2. Levantamento de requisitos

Com base no **Objetivo Geral**, foram criados os requisitos do sistema, usando as técnicas da **engenharia de requisitos**.

### 3.3. Diagrama de casos de uso

Os diagramas de caso de uso foram diagramados, usando as normas da **linguagem de modelagem unificada** ou **UML**. Foi usado os requisitos levantados anteriormente para o desenvolvimento dos diagramas.

### 3.4. Especificação do caso de uso

Usando os processos da **especificação dos casos de uso**, desenvolveu-se as documentações das especificações de cada **diagrama de caso de uso**, elaborados anteriormente. Também foram criados os protótipos de tela em paralelo, para anexar os mesmos dentro dos respectivos documentos de especificação.

### 3.5. Tecnologias usadas no desenvolvimento da aplicação web

A partir das especificações de casos de uso, foi elaborada a escolha das respectivas tecnologias para o desenvolvimento da **aplicação web**. A plataforma de desenvolvimento **.Net Core** foi a escolhida para a criação da aplicação, pois o desenvolvedor do projeto/*MVP* possui experiência com essa tecnologia, outro fator é a facilidade, rapidez e segurança que essa plataforma proporciona.

Utilizando o critério de experiência do desenvolvedor, a linguagem de programação designada para o desenvolvimento na plataforma **.Net Core** foi o C#. O banco de dados escolhido foi o Microsoft Sql Server utilizando o modelo relacional tradicional, o fator principal pela sua escolha é a sua fácil integração com a plataforma **.Net Core**. Para executar a tarefa de transcrever o áudio do *podcast* foi escolhido o serviço **IBM Watson Speech to Text**.

Com a finalidade de facilitar a comunicação entre aplicação *web*, e o navegador, foi optado pelo padrão **Rest Api** para requisições *HTTPS*, feitas durante o uso da aplicação no *browser*(navegador) pelo usuário.

### **3.6. Serviço de pesquisa(Sonic)**

Após os testes iniciais do *MVP* foi observado uma deficiência muito latente, na pesquisa dos catálogos de podcast cadastrados. Essa deficiência ocorria pelo fato de ser executada apenas uma consulta simples no banco de dados, assim não era ignorado acentuação, plural, palavras erradas e segmentação de termos para procura, com o intuito de atender essas necessidades de pesquisa foi implementado um serviço de indexação de conteúdo. O escolhido foi o serviço *Sonic*, por se tratar de uma solução de fácil implementação e resultados consistentes.



#### 4. Resultados e discussões

Neste capítulo serão apresentados os resultados do desenvolvimento, da parte de engenharia de *software* que engloba, o levantamento de requisitos, diagrama de casos de uso e especificação do caso de uso. Também serão explanados os resultados do desenvolvimento da aplicação *web*.

##### 4.1. Levantamento de requisitos

Devido a simplicidade do escopo do *MVP* não ocorreu um número grande de requisitos, pois se trata de um projeto simples, onde só possui duas funcionalidades, são elas catalogar *podcast* com sua devida transcrição e pesquisar os mesmos na sua base de dados. Em razão desse fato foi levantado apenas quatro requisitos, são eles:

Quadro 1 - Requisito número 1: Catalogar episódios de podcast.

<b>Requisito N°</b>	1				
<b>Descrição</b>	Catalogar episódios de podcast				
<b>Argumento p/ Inclusão</b>	Catalogar episódios de podcast, por meio de uma transcrição de áudio do episódio, junto com o nome do episódio, nome do podcast e link para o site oficial do mesmo. Armazenar esses catálogo em um base de dados				
<b>Proprietário</b>	Usuário	<b>Fonte</b>	Entrevista		
<b>Prioridade</b>	Alta	<b>Versão</b>	1	<b>Tipo</b>	Produto
<b>Status</b>	Ativo	<b>Data da Conclusão</b>	25/05/2021		

Fonte: Autor, 2021.

Quadro 2 - Requisito número 2: Pesquisa de *podcast* a partir das transcrições catalogadas.

<b>Requisito N°</b>	2				
<b>Descrição</b>	Pesquisa de podcast a partir das transcrições catalogadas				
<b>Argumento p/ Inclusão</b>	Pesquisar podcast a partir das transcrições catalogadas em uma base de dados				
<b>Proprietário</b>	Usuário	<b>Fonte</b>	Entrevista		
<b>Prioridade</b>	Média	<b>Versão</b>	1	<b>Tipo</b>	Produto
<b>Status</b>	Ativo	<b>Data da Conclusão</b>	25/05/2021		

Fonte: Autor, 2021.

Quadro 3 - Requisito número 3: Indexar as transcrições em motores de busca.

<b>Requisito N°</b>	3				
<b>Descrição</b>	Indexar as transcrições em motores de busca				
<b>Argumento p/ Inclusão</b>	Indexar as transcrições em motores de busca como (Google, DuckDuckGo e Bing). Otimizar as pesquisas nos buscadores a partir de uma otimização de SEO				
<b>Proprietário</b>	Usuário	<b>Fonte</b>	Entrevista		
<b>Prioridade</b>	Média	<b>Versão</b>	1	<b>Tipo</b>	Produto
<b>Status</b>	Ativo	<b>Data da Conclusão</b>	25/05/2021		

Fonte: Autor, 2021.

Quadro 4 - Requisito número 4: Fazer as transcrições dos podcasts automatizadas.

<b>Requisito N°</b>	4				
<b>Descrição</b>	Fazer as transcrições dos podcasts automatizadas				
<b>Argumento p/ Inclusão</b>	Fazer as transcrições dos podcasts automatizadas por meio de inteligência artificial, a partir da técnica speech-to-text				
<b>Proprietário</b>	Usuário	<b>Fonte</b>	Entrevista		
<b>Prioridade</b>	Alta	<b>Versão</b>	1	<b>Tipo</b>	Produto
<b>Status</b>	Ativo	<b>Data da Conclusão</b>	25/05/2021		

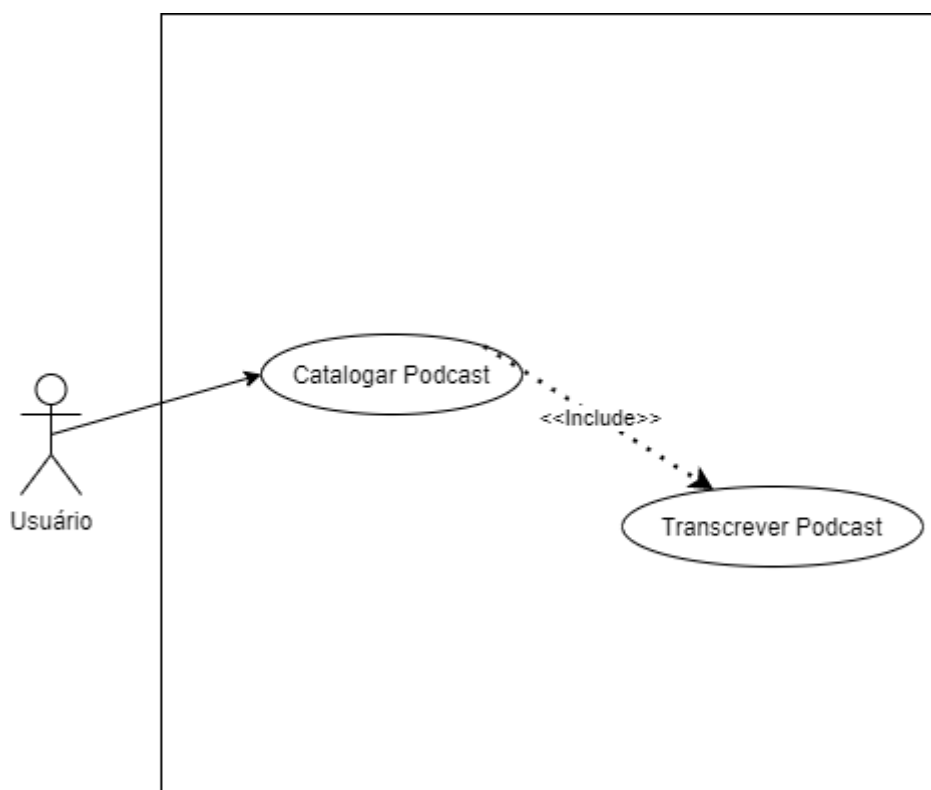
Fonte: Autor, 2021.

Não foi levantado o requisito de registro de usuário, porque o objetivo do catálogo é possibilitar um cadastro simples. Como se trata de algo voluntário, foi elaborada a forma menos burocrática de criar um novo registro, para não desencorajar novos usuários a alimentar a base de dados.

#### 4.2. Diagrama de caso de uso

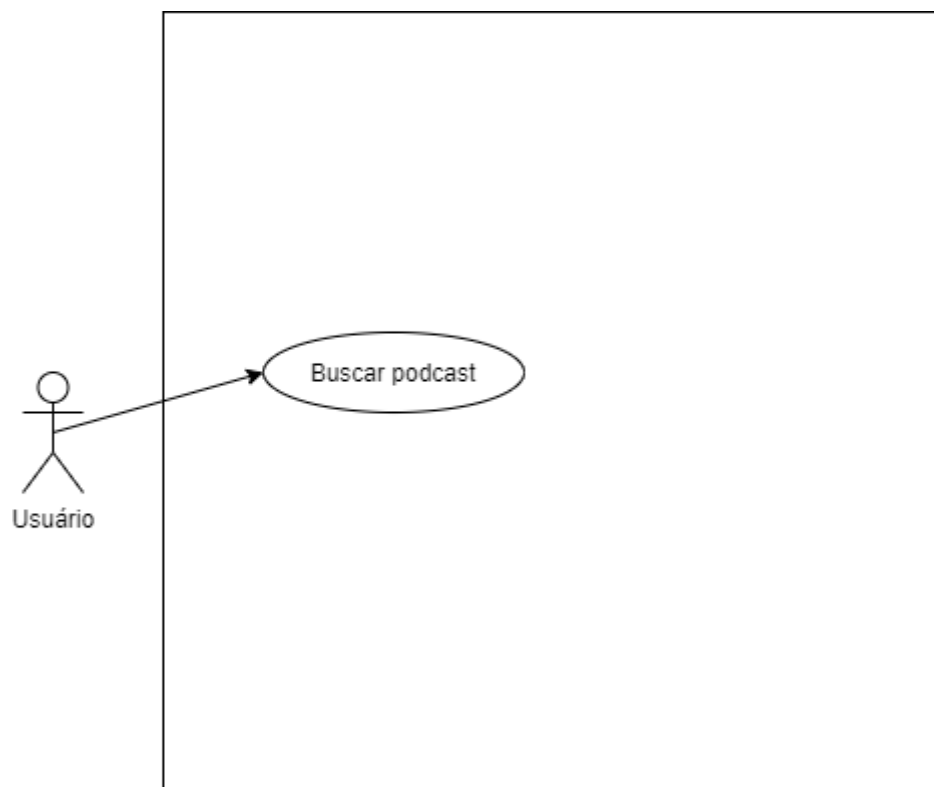
Posteriormente ao levantamento de requisitos foi desenvolvida a diagramação dos diagramas de caso de uso, com base nos requisitos que foram levantados a partir do **Objetivo Geral**.

Figura 1 - Diagrama de caso de uso: Cadastro de Catálogo de podcast



Fonte: Autor, 2021

Figura 2 - Diagrama de caso de uso: Busca de podcast.



Fonte: Autor, 2021

O ator do diagrama de caso de uso que está representando o usuário, refere-se a qualquer pessoa disposta a cadastrar um podcast no catálogo, ou procurar o mesmo na aplicação *web*. Como foi explicado anteriormente, nesta primeira versão do projeto não possui sistema de login.

#### 4.3. Especificação do caso de uso

O processo de criação das especificações dos casos de uso, teve como base para seu desenvolvimento, os requisitos levantados e os diagramas de casos de uso. A partir dos documentos de especificação anexados no APÊNDICE A e APÊNDICE B, será iniciado o desenvolvimento da aplicação *web*, tendo esses documentos como guia para regras de negócios, fluxos, protótipos das telas e exceções.

#### 4.4. Aplicação web

Possuindo toda a documentação de análise de *software* (levantamento de requisitos, diagrama de caso de uso e especificação do caso de uso), foi possível ter suporte necessário para escolher, qual tecnologia usar, e como transformar um objetivo que foi elaborado a partir

de uma pergunta, em uma aplicação *web* funcional, que possibilita uma maior rastreabilidade dos programas/episódios de *podcast* a partir de suas transcrições.

#### 4.4.1. Arquitetura

Como o projeto possui requisitos modestos, não foi necessário criar uma arquitetura com micro serviços, ou desenvolver um sistema de mensageria. Em vista desses fatores, foi optado usar uma arquitetura monolítica, separada em camadas.

Camadas da arquitetura *Web*:

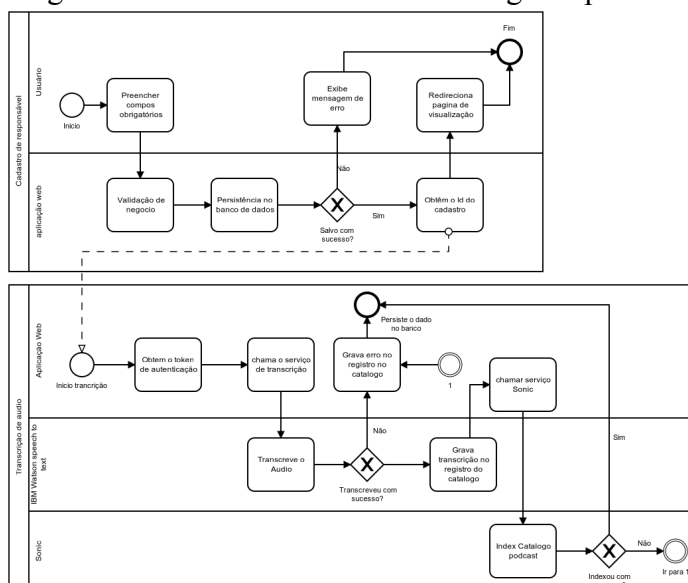
- **Web:** É responsável por servir o html e arquivos estáticos da aplicação.
- **Negócio:** É responsável por centralizar as regras de negócio.
- **Infra:** Fornece a infraestrutura para aplicação.

Por não se tratar de um crud com diversas telas, não foram criadas abstrações para o mesmo, foi apenas utilizado a biblioteca de *ORM* da plataforma *.Net Core*. O código fonte da aplicação *web* e instruções para sua implementação está disponível na plataforma *GitHub*, através da url <https://github.com/rcoliveira2016/TCC-FSG>.

#### 4.4.2. Fluxos das funcionalidades

Com intuito exemplificar melhor o fluxo do cadastro de catálogo de *podcast*, foi criado um diagrama para demonstrar essas operações de forma visual. Nele é demonstrado as interações do sistema *web* com o usuário(navegador), *IBM Watson speech to text* e serviço de pesquisa *Sonic*.

Figura 3 - Fluxo do cadastro de catálogo de podcast.



Fonte: Autor, 2021.

#### 4.5. Limitações e restrições

Por tratar-se de um *MVP*, existem limitações e restrições, quem em uma segunda versão podem ser contempladas. Uma grande limitação, inclusive a maior, é o serviço de transcrição usado(*IBM Watson speech to text*), pelo fato de não possuir uma grande fidelidade nas transcrições, não separar os trechos de texto, quando há duas ou mais pessoas falando. Outro fator é financeiro, pois existe uma quantidade de horas gratuitas limitadas(8 horas e 20 minutos), após essa quantia acabar ocorre uma cobrança, por minutos transcritos.

Contudo existem pendências mais simples de serem resolvidas, como implementar uma pesquisa mais inteligente, na funcionalidade de “Pesquisa *Podcasts*”. Visto que na atual versão é utilizado um serviço simples de pesquisa(*Sonic*), que pode ser substituído por algum serviço mais completo como *Elasticsearch*.

#### 4.6. Monetização

No que tange a monetização do *MVP* não existe nada concreto, pois não é o foco do estudo. Entretanto observando outros *case*, há poucas formas de monetização, nesse tipo de projeto, pois se trata de um projeto independente, onde não está sendo vendido nada, seja os dados do usuário, um produto ou serviço.

Não obstante existem algumas alternativas:

- Oferecer espaços para anunciantes no site.
- Doação de dinheiro feita por usuários.
- Obter financiamento de empresas.

## 5. Conclusões e Considerações finais

Partindo da questão levantada de como poderia ser facilitada a procura de conteúdo não verbal, contido na mídia *podcast*. Tendo essa pergunta como base, foi delimitado o objetivo da pesquisa, que seria facilitar a busca, a partir de uma aplicação *web*, que iria transcrever o áudio dos episódios, assim facilitando sua pesquisa.

Foram levantados apenas quatro requisitos, dois diagramas de caso e duas especificações de casos de uso, pois se tratava de um objetivo simples. Foi empenhado a maior parte do tempo em pesquisas para encontrar um método para transcrever o áudio. Já o desenvolvimento da aplicação *web* ocorreu de forma simples e fluida, sem nenhum gargalo ou impeditivo, porém foram levantados alguns pontos de melhoria.

Após o período de desenvolvimento da aplicação *web*, iniciou-se os testes de funcionalidades do *MVP*. Tendo os resultados dos testes, observou-se que as transcrições eram confusas, pois nela não existia separação de diálogos paralelos, por exemplo, duas pessoas falando ao mesmo tempo. Outro ponto a ser analisado é a viabilidade comercial, pois o serviço usado para transcrição é pago por demanda, sem contar custos de hospedagens e banco de dados. O projeto não possui uma estratégia para gerar receita, e também não se aventa uma maneira simples de capitalizar com o projeto, pois o mesmo não teria fins lucrativos.

Ponderando todos os pontos levantados neste trabalho, é explícito que existe uma grande demanda para a mídia *podcast*, assim gerando um hipotético interesse pela comunidade de *podcasting*, pela solução gerada a partir do *MVP*. Porém existe um grande gargalo no que tange a viabilidade financeira do projeto, como foi explanado acima.

## REFERÊNCIAS

ALURA. **Engenharia de Requisitos: como levantar, documentar e validar**. Disponível em:

<[https://www.alura.com.br/conteudo/engenharia-requisitos?gclid=Cj0KCCQjw78yFBhCZARIsAOxgSx1WJ6KmsbFL6gLdKtJy4Kc3cPBBu2-c4wH1EoP9oal9MfigdN0rYGwaAlfxEALw\\_wcB](https://www.alura.com.br/conteudo/engenharia-requisitos?gclid=Cj0KCCQjw78yFBhCZARIsAOxgSx1WJ6KmsbFL6gLdKtJy4Kc3cPBBu2-c4wH1EoP9oal9MfigdN0rYGwaAlfxEALw_wcB)>. Acesso em: 30 mai. 2021.

DEMARTINI, Felipe. **Especial | Em alta desde 2019, podcasts multiplicam gêneros e dinheiro no Brasil**. Disponível em:

<<https://canaltech.com.br/entretenimento/especial-em-alta-desde-2019-podcasts-multiplicam-generos-e-dinheiro-no-brasil-175681/>>. Acesso em: 26 mai. 2021.

DEVMEDIA. **Artigo Engenharia de Software 11 - Especificação de Casos de Uso**.

Disponível em:

<<https://www.devmedia.com.br/artigo-engenharia-de-software-11-especificacao-de-casos-de-uso/12210#:~:text=apresentada%20e%20exemplificada.,Especifica%C3%A7%C3%A3o%20de%20Casos%20de%20Uso,caso%20de%20uso%20e%20ator>>. Acesso em: 30 mai. 2021.

LOUBAK, Ana Letícia. **O que é podcast? Saiba tudo sobre os programas de áudio online**.

Disponível em:

<<https://www.techtudo.com.br/noticias/2019/12/o-que-e-podcast-saiba-tudo-sobre-os-programas-de-audio-online.ghhtml>>. Acesso em: 26 mai. 2021.

LOURES, Alexandre. **Nas ondas do áudio: 8% dos brasileiros ouvem podcast e tendência só cresce**. Disponível em:

<<https://exame.com/bussola/nas-ondas-do-audio-8-dos-brasileiros-ouvem-podcast-e-tendencia-so-cresce/>>. Acesso em: 26 mai. 2021.

LUCIDCHART. **Diagrama de caso de uso UML: O que é, como fazer e exemplos**.

Disponível em: <<https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml>>.

Acesso em: 30 mai. 2021.

MICROSOFT. **Introdução ao .NET**. Disponível em:

<<https://docs.microsoft.com/pt-br/dotnet/core/introduction>>. Acesso em: 30 mai. 2021.

PIRES, Jackson. **O que é API? REST e RESTful? Conheça as definições e diferenças!**.

Disponível em: <<https://becode.com.br/o-que-e-api-rest-e-restful/>>. Acesso em: 30 mai. 2021.

PODCASTING. **Wikipedia**. Disponível em: <<https://pt.wikipedia.org/wiki/Podcasting>>.

Acesso em: 26 mai. 2021.

POLITI, Cassio. **CONHEÇA A HISTÓRIA DO PODCAST NO MUNDO**. Disponível em:

<<https://www.comunique-se.com.br/blog/conheca-a-historia-do-podcast-no-mundo/>>. Acesso em: 26 mai. 2021.

SALIOU, Valerian. **Sonic**. Disponível em: <<https://github.com/valeriansaliou/sonic>>. Acesso

em: 19 jun. 2021.



SOUZA, Ivan. **Entenda o que é Rest API e a importância dele para o site da sua empresa.** Disponível em: <<https://rockcontent.com/br/blog/rest-api/>>. Acesso em: 30 mai. 2021.

TIGRE, Rodrigo. **PODCAST ADVERTISING.** Disponível em: <[https://iabbrasil.com.br/wp-content/uploads/2019/08/Guia-IAB-Podcast\\_v2.pdf](https://iabbrasil.com.br/wp-content/uploads/2019/08/Guia-IAB-Podcast_v2.pdf)>. Acesso em: 26 mai. 2021.

WIKIPÉDIA. **Aplicação web.** Disponível em: <[https://pt.wikipedia.org/wiki/Aplica%C3%A7%C3%A3o\\_web](https://pt.wikipedia.org/wiki/Aplica%C3%A7%C3%A3o_web)>. Acesso em: 30 mai. 2021.

WIKIPÉDIA. **Caso de uso.** Disponível em: <[https://pt.wikipedia.org/wiki/Caso\\_de\\_uso](https://pt.wikipedia.org/wiki/Caso_de_uso)>. Acesso em: 30 mai. 2021.

WIKIPÉDIA. **Engenharia de requisitos.** Disponível em: <[https://pt.wikipedia.org/wiki/Engenharia\\_de\\_requisitos#cite\\_note-1](https://pt.wikipedia.org/wiki/Engenharia_de_requisitos#cite_note-1)>. Acesso em: 30 mai. 2021.

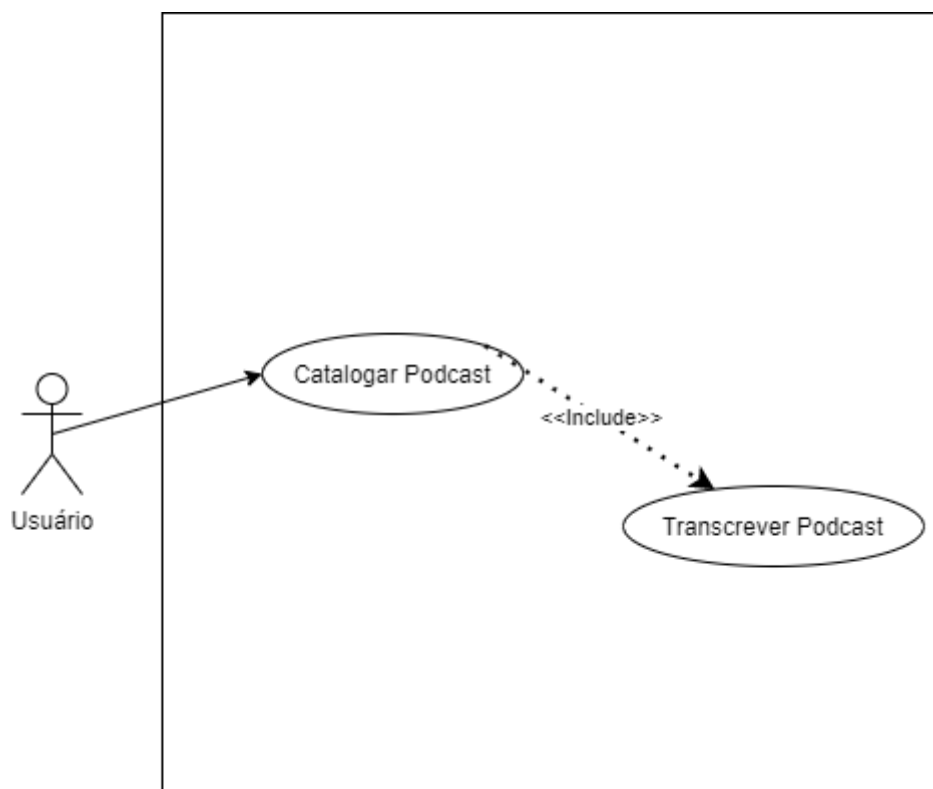
WIKIPÉDIA. **Microsoft SQL Server.** Disponível em: <<https://www.microsoft.com/pt-br/sql-server/sql-server-downloads>>. Acesso em: 30 mai. 2021.

**APÊNDICE A - Especificação de Casos de Uso - [1] Catálogo Podcast****Índice Analítico**

1.	Diagrama do Caso de Uso	4
2.	Nível	4
3.	Ator Principal	4
4.	Requisito Funcional	4
5.	Interessados e interesses	4
6.	Pré-condições	4
7.	Garantia de Sucesso (Pós-condições)	4
8.	Cenário de Sucesso (Fluxo Básico)	5
9.	Pontos de Extensão (Fluxos Alternativos)	5
10.	Requisitos Especiais	6
11.	Lista de Variantes Tecnológicas e de Dados	6
12.	Diversos	7

## Especificação de Caso de Uso: 2 / Catálogo Podcast

### 1. Diagrama do Caso de Uso



### 2. Nível

Permitir catalogar episódios de podcast, por meio de uma transcrição de áudio do episódio, junto com o nome do episódio, nome do podcast e link para o site oficial do mesmo.  
Armazenar esses catálogo em um base de dados

### 3. Ator Principal

Usuário

### 4. Requisito Funcional

Catalogar episódios de podcast

### 5. Interessados e interesses

**Usuário:** Catalogar episódios de podcast, transcrevendo o seu áudio, a partir de um

arquivo mp3. Também será necessário incluir o nome do episódio, nome do podcast e link para o site oficial da publicação

## **6. Pré-condições**

Cadastrar um catálogo de podcast e transcrever o áudio do mesmo

## **7. Garantia de Sucesso (Pós-condições)**

Validação dos dados inseridos, arquivo mp3, nome do episódio, nome do podcast e link para o site oficial da publicação

## **8. Cenário de Sucesso (Fluxo Básico)**

1. Inserir nome do podcast.
2. Inserir nome do episódio.
3. Inserir link ou url do episódio do podcast
4. Fazer upload do arquivo de áudio
5. Salvar catalogo.
6. Iniciar transcrição do áudio do podcast catalogado em segundo plano.
7. Transcrever podcast por meio da Api do IBM Watson speech to text.
8. Após a conclusão da transcrição do áudio em segundo plano, irá ser atualizado o registro do catálogo com a transcrição do áudio.

## **9. Pontos de Extensão (Fluxos Alternativos)**

- 1.1. Inserir o nome do episódio e o nome do podcast, iguais a um já cadastrado.;
- 1.2. Sistema aponta erro de dado duplicado
- 1.3 Sistema e solicita a alteração do Nome do podcast ou episódio;
  
- 2.1. Deixar em branco o campo: nome do podcast;
- 2.2. Sistema aponta erro de campo obrigatório;
- 2.3. O sistema retorna para o início do campo.
  
- 3.1. Deixar em branco o campo: nome do episódio;
- 3.2. Sistema aponta erro de campo obrigatório;
- 3.3. O sistema retorna para o início do campo.
  
- 4.1. Não enviar o áudio(fazer upload do arquivo de áudio) do podcast.
- 4.2. Sistema aponta erro de campo obrigatório;
- 4.3. O sistema retorna para o início do campo.

## **10. Requisitos Especiais**

Integração com a Api do IBM Watson speech to text.

## 11. Lista de Variantes Tecnológicas e de Dados

Integração com o serviço da IBM Cloud(Api do IBM Watson speech to text.)

## 12. Diversos

A Web Page

https://

Importar podcast

Nome Podcast

Nome do episódio

Url do site Podcast(Opcional)

Arquivo do podcast

Escolha um Arquivo

Salvar

A Web Page

https://

Importar podcast

Nome Podcast

Nome do episódio

Url do site Podcast(Opcional)

Arquivo do podcast

Escolha um Arquivo

Sucesso

www.exemplo.com.br/podcast-importado/afgtr-6512-7as9a . obs.: a transcrição pode demorar alguns minutos

Salvar

A Web Page

https://

Visualizar podcast

Nome Podcast


Nome do episodio

Url do site Podcast(Opcional)

Arquivo do podcast

Baixar arquivo dom podcast

O áudio ainda está sendo transcrito. Tente mais tarde



A Web Page

https://

Visualizar podcast

Nome Podcast

Nome do episodio

Url do site Podcast(Opcional)

Arquivo do podcast

Baixar arquivo dom podcast

Transcrição

...

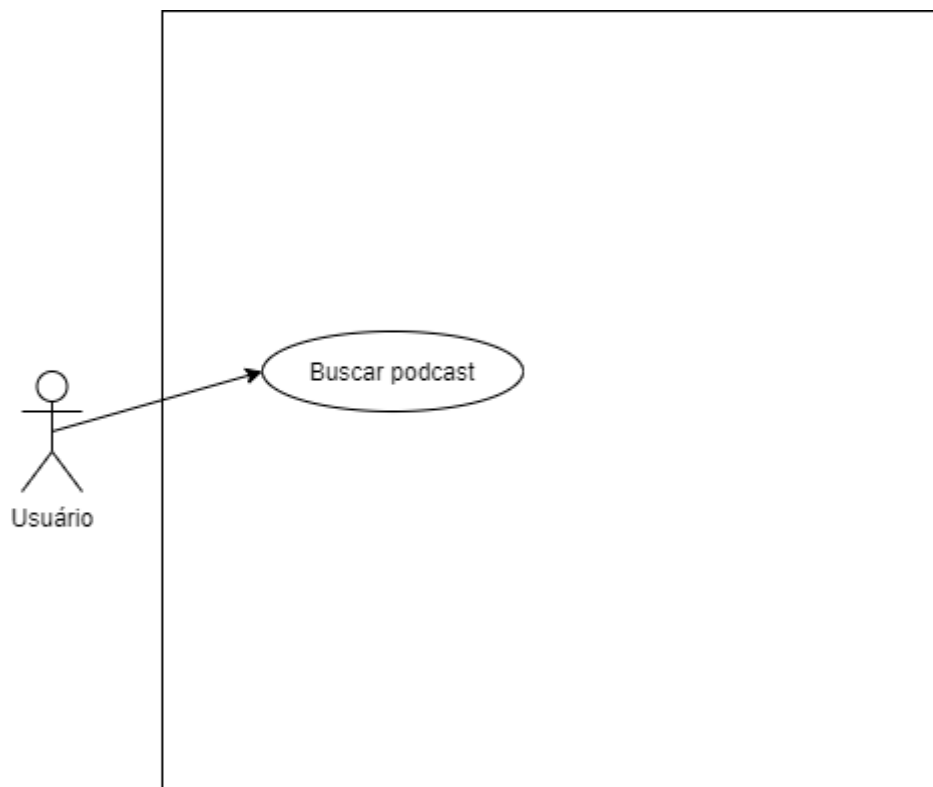
**APÊNDICE B - Especificação de Casos de Uso - [2] Pesquisa de podcast****Índice Analítico**

1. Diagrama do Caso de Uso	4
2. Nível	4
3. Ator Principal	4
4. Requisito Funcional	4
5. Interessados e interesses	4
6. Pré-condições	4
7. Garantia de Sucesso (Pós-condições)	4
8. Cenário de Sucesso (Fluxo Básico)	5
9. Pontos de Extensão (Fluxos Alternativos)	5
10. Requisitos Especiais	6
11. Lista de Variantes Tecnológicas e de Dados	6
12. Diversos	7



## Especificação de Caso de Uso: 2 / Pesquisa de podcast

### 1. Diagrama do Caso de Uso



### 2. Nível

Permite encontrar episódios de podcast catalogados no sistema, por meio de uma pesquisa textual, onde será procurado o termo da pesquisa pelo nome do episódio, ou um trecho contido na transcrição do podcast. Também é possível clicar no botão( 🔍 ), onde será redirecionado para tela de visualizar podcast.


### 3. Ator Principal

Usuário

### 4. Requisito Funcional

Pesquisa de podcast a partir das transcrições catalogadas

## 5. Interessados e interesses

**Usuário:** Permite encontrar episódios de podcast catalogados no sistema, por meio de uma pesquisa textual, onde será procurado o termo da pesquisa pelo nome do episódio, ou um trecho contido na pesquisa. Também é possível clicar no botão(  ), onde será redirecionado para tela de visualizar podcast.


## 6. Pré-condições

Existir dados cadastrados de episódios de podcast

## 7. Garantia de Sucesso (Pós-condições)

Retorno dos dados de acordo com a pesquisa feita por texto.

## 8. Cenário de Sucesso (Fluxo Básico)

1. Inserir busca por algum termo.
2. Clicar na lupa(  ).
3. Fazer busca.

## 9. Pontos de Extensão (Fluxos Alternativos)

## 10. Requisitos Especiais

## 11. Lista de Variantes Tecnológicas e de Dados

## 12. Diversos

