



INSTITUTO FEDERAL DO PIAUÍ – IFPI
PRÓ-REITORIA DE ENSINO
DIRETORIA DE ENSINO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS

VOZ DA AVENTURA: HISTÓRIAS INTERATIVAS PARA O SISTEMA AMAZON
ALEXA

JOÃO HENRIQUE ARAÚJO DOS SANTOS

PICOS - PI

2026

JOÃO HENRIQUE ARAÚJO DOS SANTOS

VOZ DA AVENTURA: HISTÓRIAS INTERATIVAS PARA O SISTEMA AMAZON
ALEXA

Relatório Técnico de Software apresentado
como exigência parcial para obtenção do
diploma do Curso de Tecnologia em Análise
e Desenvolvimento de Sistemas do Instituto
Federal de Educação, Ciência e Tecnologia
do Piauí, Campus Picos.

Orientador: Prof. Me. João Paulo Lima do
Nascimento

JOÃO HENRIQUE ARAÚJO DOS SANTOS

VOZ DA AVENTURA: HISTÓRIAS INTERATIVAS PARA O SISTEMA AMAZON
ALEXA

Relatório Técnico de Software apresentado
como exigência parcial para obtenção do
diploma do Curso de Tecnologia em Análise
e Desenvolvimento de Sistemas do Instituto
Federal de Educação, Ciência e Tecnologia
do Piauí, Campus Picos.

Aprovada em: 05/03/2026.

BANCA EXAMINADORA

Prof. Me. João Paulo Lima do Nascimento (Orientador)
Instituto Federal do Piauí (IFPI)

Prof. Me. Jader Anderson Oliveira de Abreu
Instituto Federal do Piauí (IFPI)

Prof. Dr. Rogério Figueredo de Sousa
Instituto Federal do Piauí (IFPI)

RESUMO

O presente Trabalho de Conclusão de Curso (TCC) apresenta o desenvolvimento de uma aplicação para protótipo para a assistente virtual Amazon Alexa denominada Voz da Aventura, focada na narração de histórias interativas. O projeto, de caráter experimental, teve como objetivo principal investigar a viabilidade técnica e as possibilidades narrativas proporcionadas por interfaces de voz, explorando a arquitetura e recursos da Alexa Skills Kit (ASK). A metodologia adotada incluiu o estudo do design conversacional, modelagem de fluxo de diálogos e gerenciamento de estados. A solução é composta por duas aplicações distintas e integradas: um Backoffice para gerenciamento de conteúdo e uma Skill Alexa como interface de voz para o usuário final. Detalhes técnicos sobre as tecnologias empregadas no desenvolvimento do Backoffice (Node.js, Express, Postgres, TypeORM) e da Skill Alexa (ask-sdkcore, ask-sdk-model, Axios) são apresentados. Adicionalmente, foram realizados testes de usabilidade com usuários por meio da escala System Usability Scale (SUS), cujos resultados indicaram uma percepção positiva quanto à usabilidade da aplicação. De forma geral, os resultados obtidos evidenciam a viabilidade da solução proposta.

Palavras chave: Amazon Alexa; Histórias interativas; Interfaces de voz; Design conversacional.

ABSTRACT

This Final Course Project (TCC) presents the development of a prototype application for the Amazon Alexa virtual assistant, named *Voz da Aventura*, focused on interactive story narration. The project, of an experimental nature, aimed to investigate the technical feasibility and the narrative possibilities provided by voice interfaces, exploring the architecture and resources of the Alexa Skills Kit (ASK). The adopted methodology included the study of conversational design, dialogue flow modeling, and state management. The solution is composed of two distinct and integrated applications: a Backoffice for content management and an Alexa Skill as the voice interface for the end user. Technical details regarding the technologies used in the development of the Backoffice (Node.js, Express, Postgres, TypeORM) and the Alexa Skill (ask-sdk-core, ask-sdk-model, Axios) are presented. Additionally, usability tests were conducted with users using the System Usability Scale (SUS), whose results indicated a positive perception regarding the application's usability. Overall, the results demonstrate the feasibility of the proposed solution.

Keywords: Amazon Alexa; Interactive Stories; Voice interfaces; Conversational design; .

SUMÁRIO

1. INTRODUÇÃO	6
1.1. Objetivos	7
1.1.1. Objetivo geral	7
1.1.2. Objetivos específicos	8
2. TECNOLOGIAS ENVOLVIDAS	9
2.1. Skill Alexa	9
2.2. Backoffice	10
3. MODELAGEM DO PROJETO	12
3.1. Levantamento de requisitos	13
3.2. Diagramas de casos de uso	14
3.3. Arquitetura da solução	16
3.4 Diagrama de Entidade-Relacionamento (DER)	18
3.5. Interface	20
4. SOFTWARE	21
4.1. Implantação	22
4.1.1. Backoffice	22
4.1.2. Skill Alexa	23
4.2. Metodologia de Avaliação de Usabilidade	31
4.3. Testes	32
4.4. Resultados e Discussão	33
4.5. Comparação com Aplicações Semelhantes	35
5. CONSIDERAÇÕES FINAIS	38
REFERÊNCIAS	40

1. INTRODUÇÃO

A crescente adoção de tecnologias digitais no cotidiano tem ampliado as possibilidades de acesso à informação, educação e entretenimento (Brynjolfsson; McAfee, 2014). Nesse cenário, as formas de interação entre usuários e sistemas computacionais evoluem continuamente, com destaque para o surgimento de novas abordagens que buscam tornar a experiência mais natural e intuitiva.

A predominância de interfaces tradicionais baseadas em elementos visuais e dispositivos de entrada convencionais pode limitar a exploração de experiências mais interativas e imersivas em conteúdos digitais. Nesse sentido, aplicações baseadas em interação por voz apresentam-se como uma alternativa para o desenvolvimento de narrativas interativas, nas quais o usuário pode participar ativamente do desenrolar da história por meio de comandos de voz (Porcheron; Fischer; Reeves; Sharma, 2018).

Em um contexto global, o avanço das interfaces de voz e a popularização de assistentes virtuais, como a Amazon Alexa, têm transformado a forma como os usuários interagem com sistemas computacionais, permitindo interações mais naturais e baseadas em linguagem natural (Pearl, 2016).

Estudos recentes indicam que assistentes virtuais estão se consolidando como uma importante interface de interação humano-computador, oferecendo novas formas de acesso a serviços, informações e entretenimento por meio de comandos de voz (Hoy, 2018). Apesar desse avanço, ainda existe uma lacuna significativa no desenvolvimento de conteúdos interativos narrativos que explorem plenamente o potencial dessas interfaces, especialmente no que se refere à criação de experiências mais dinâmicas e imersivas (Hoyle et al., 2019; Laugwitz et al., 2020; Porcheron; Fischer; Reeves; Sharma, 2018).

No contexto nacional, essa lacuna torna-se ainda mais evidente no que se refere ao desenvolvimento de aplicações que explorem o potencial de assistentes virtuais para além de usos utilitários. Embora essas tecnologias estejam cada vez mais presentes em dispositivos domésticos e móveis no Brasil — sendo que 91% dos brasileiros já utilizaram assistentes virtuais por voz em seus dispositivos móveis (Ilumeo, 2022) —, o desenvolvimento de aplicações voltadas especificamente para experiências narrativas interativas ainda é limitado.

Diante desse cenário, este Trabalho de Conclusão de Curso documenta o desenvolvimento de uma aplicação protótipo para a assistente virtual Amazon Alexa, com foco na narração de histórias interativas. A aplicação foi concebida como um projeto experimental, com ênfase no estudo e na aplicação de tecnologias emergentes de interfaces de voz, utilizando recursos do Alexa Skills Kit (ASK) para criar uma experiência interativa baseada exclusivamente em comandos de voz.

O sistema foi projetado para permitir que os usuários participem ativamente do desenvolvimento da narrativa, por meio de escolhas que influenciam o desenrolar da história, utilizando conceitos de design conversacional, modelagem de fluxos de diálogo e gerenciamento de estados. O repositório contendo o código-fonte da aplicação desenvolvida encontra-se disponível no Apêndice A deste trabalho.

Como principais contribuições, este projeto estabelece avanços relevantes no campo tecnológico, destacando-se pelos seguintes aspectos:

- Exploração de interfaces conversacionais para narrativa interativa: a solução utiliza interação por voz como principal forma de comunicação com o usuário, possibilitando a construção de experiências narrativas interativas baseadas em comandos de voz.
- Avanço no design conversacional e interação humano-computador: o projeto contribui com boas práticas de modelagem de diálogo, gerenciamento de estados e fluxos narrativos, favorecendo experiências conversacionais mais naturais e eficientes.

1.1. Objetivos

1.1.1. Objetivo geral

Desenvolver um protótipo funcional de uma aplicação de narração interativa baseada na assistente virtual Alexa, permitindo a interação dos usuários com narrativas dinâmicas por meio de comandos de voz.

1.1.2. Objetivos específicos

- **Levantar e analisar os requisitos funcionais e não-funcionais do sistema**, a fim de definir as necessidades para o bom funcionamento das aplicações envolvidas;
- **Modelar o sistema utilizando diagramas UML**, como casos de uso, atividades e arquitetura, além da modelagem de dados utilizando diagrama entidade-relacionamento. Estes documentos detalham o funcionamento e estruturas seguidas no desenvolvimento das aplicações envolvidas;
- **Projetar e implementar uma aplicação de narração interativa baseada em voz**, utilizando a plataforma Alexa, com foco em navegação intuitiva, fornecimento de feedback auditivo adequado e suporte à construção de narrativas interativas baseadas em escolhas do usuário;
- **Definir a arquitetura tecnológica da solução**, composta por uma Skill Alexa executada em AWS Lambda e integrada a um backend responsável pelo gerenciamento das narrativas e pela disponibilização de uma API REST.
- **Implementar a comunicação baseada em API REST** entre a skill Amazon Alexa e a aplicação backend.
- **Avaliar a usabilidade da aplicação**, por meio da realização de testes com usuários e da aplicação de métodos consolidados de avaliação, como o System Usability Scale (SUS), com o objetivo de identificar oportunidades de melhoria e assegurar que a solução desenvolvida atenda adequadamente às necessidades e expectativas do público-alvo.

Este trabalho está organizado em cinco capítulos, incluindo esta introdução. O Capítulo 2 apresenta as tecnologias utilizadas no desenvolvimento da solução. O Capítulo 3 descreve a modelagem do projeto, incluindo requisitos, diagramas e arquitetura. O Capítulo 4 detalha a implementação do software, os testes realizados e a análise dos resultados. Por fim, o Capítulo 5 apresenta as considerações finais, as contribuições do trabalho e possíveis direções para trabalhos futuros.

2. TECNOLOGIAS ENVOLVIDAS

A escolha adequada das tecnologias no desenvolvimento de *software* é fundamental para garantir a qualidade, a escalabilidade e a manutenibilidade do produto. De acordo com Ian Sommerville (2016), decisões arquiteturais e tecnológicas influenciam diretamente atributos de qualidade, como desempenho, segurança e facilidade de evolução do sistema. Tecnologias alinhadas aos requisitos funcionais e não funcionais do projeto reduzem riscos técnicos e contribuem para a construção de soluções mais estáveis e sustentáveis ao longo do tempo (ISO/IEC 25010, 2023; IEEE, 2022; Kruger; Matthes, 2021).

Da mesma forma, Roger S. Pressman (2020) destaca que a qualidade do produto está intimamente relacionada às decisões tomadas nas fases iniciais do desenvolvimento, especialmente na definição de ferramentas, *frameworks* e padrões arquiteturais. Uma escolha tecnológica coerente com o contexto do projeto e com a capacidade da equipe favorece boas práticas de engenharia de *software*, facilita a manutenção e aumenta a competitividade do produto no mercado (Fowler, 2022; Richards; Ford, 2020; Microsoft, 2023).

2.1. Skill Alexa

O desenvolvimento da Skill Alexa foi realizado utilizando tecnologias baseadas no ecossistema JavaScript e bibliotecas fornecidas pela própria plataforma de desenvolvimento da Amazon. Essas tecnologias foram escolhidas por oferecerem suporte direto à integração com a assistente virtual Alexa, além de facilitarem o desenvolvimento e a manutenção da lógica da aplicação.

Node.js (v20.20.0). Trata-se de um ambiente de execução JavaScript baseado no motor V8 do Google, amplamente utilizado para o desenvolvimento de aplicações backend e serviços web (Nodejs, 2024). No presente projeto, Node.js foi utilizado como base para a implementação da lógica da Skill Alexa, permitindo a criação de funções responsáveis pelo processamento das requisições enviadas pela plataforma da assistente virtual.

ask-sdk-core (v2.14.0) e **ask-sdk-model** (v1.86.0). Essas bibliotecas fazem parte do Alexa Skills Kit (ASK) que consiste em um conjunto de ferramentas disponibilizado pela Amazon para o desenvolvimento de aplicações para a assistente Alexa (Amazon, 2024). O SDK simplifica o desenvolvimento de Skills ao fornecer estruturas prontas para manipulação de requisições de voz, gerenciamento de intenções e geração de respostas. No projeto, essas bibliotecas foram utilizadas para estruturar o fluxo de interação entre o usuário e a aplicação, permitindo o processamento das intenções e a geração de respostas narrativas.

Axios (v1.7.9). Trata-se de uma biblioteca cliente HTTP baseada em promessas utilizada para realizar requisições a serviços web e APIs (Axios, 2024). No contexto do projeto, Axios foi utilizado pela Skill Alexa para realizar chamadas à API do backend responsável pelo gerenciamento das histórias, permitindo a recuperação das narrativas e de seus respectivos fluxos interativos

ts-node (v10.9.1). Trata-se de uma ferramenta que permite executar código TypeScript diretamente no ambiente Node.js sem necessidade de compilação prévia (Typescript, 2024). Essa ferramenta foi utilizada durante o desenvolvimento da Skill para facilitar a execução e testes do código, tornando o processo de desenvolvimento mais ágil.

TypeScript (v5.3.3): Trata-se de uma linguagem de programação desenvolvida pela Microsoft que estende o JavaScript adicionando tipagem estática e recursos avançados de desenvolvimento (Microsoft, 2024). No projeto, TypeScript foi utilizado para melhorar a organização do código e reduzir a ocorrência de erros durante o desenvolvimento da lógica da Skill Alexa.

2.2. Backoffice

O backend da aplicação foi desenvolvido com o objetivo de gerenciar as narrativas interativas utilizadas pela Skill Alexa, além de disponibilizar os endpoints responsáveis pela comunicação entre a assistente virtual e o sistema. Além da execução das histórias, o backend também atua como um sistema de gerenciamento de conteúdo, permitindo o armazenamento e a organização das

narrativas utilizadas pela aplicação. Para isso, foram utilizadas tecnologias do ecossistema JavaScript voltadas ao desenvolvimento de APIs, gerenciamento de dependências e persistência de dados.

Node.js (v20.20.0). Node.js é um ambiente de execução JavaScript baseado no motor V8 do Google, amplamente utilizado para o desenvolvimento de aplicações backend e serviços web (Nodejs, 2024). No presente projeto, Node.js foi utilizado como base para a implementação da aplicação backend responsável pelo processamento das requisições provenientes da Skill Alexa e pela execução da lógica de gerenciamento das narrativas interativas.

Express.js (v4.x). Express é um framework minimalista para desenvolvimento de aplicações web e APIs utilizando Node.js, oferecendo mecanismos simplificados para criação de rotas, manipulação de requisições HTTP e organização da estrutura da aplicação (Express, 2024). No projeto, Express foi utilizado como base para a estruturação da API REST responsável por disponibilizar os dados das narrativas e permitir a comunicação entre a Skill Alexa e o backend.

TSOA. TSOA é um framework para desenvolvimento de APIs REST utilizando TypeScript, permitindo a definição de rotas e contratos de API por meio de classes e decorators, além de possibilitar a geração automática de documentação baseada no padrão OpenAPI (Tsoa, 2024). No backend da aplicação, TSOA foi utilizado para organizar os controladores da API e estruturar as rotas responsáveis pelas operações do sistema.

InversifyJS. InversifyJS é uma biblioteca que implementa o padrão de injeção de dependências em aplicações JavaScript e TypeScript, baseada nos princípios de Inversão de Controle (IoC) (Inversify, 2024). No projeto, essa biblioteca foi utilizada para gerenciar a criação e o ciclo de vida dos componentes da aplicação, contribuindo para uma arquitetura mais modular e desacoplada.

TypeScript (v5.3.3). TypeScript é uma linguagem que estende o JavaScript adicionando tipagem estática e recursos avançados de desenvolvimento, contribuindo para maior organização e segurança do código (Microsoft, 2024). No backend do projeto, TypeScript foi utilizado para melhorar a estrutura do código e facilitar a manutenção da aplicação.

No que se refere à camada de persistência de dados, foi utilizado um banco de dados relacional juntamente com ferramentas que facilitam sua integração com aplicações desenvolvidas em Node.js.

PostgreSQL. PostgreSQL é um sistema de gerenciamento de banco de dados relacional de código aberto amplamente utilizado em aplicações que demandam confiabilidade, integridade de dados e alto desempenho (PostgreSQL, 2024). No projeto, PostgreSQL foi utilizado para armazenar as narrativas interativas, seus segmentos e as possíveis ações disponíveis durante a execução das histórias.

TypeORM. TypeORM é um Object-Relational Mapper (ORM) para aplicações Node.js e TypeScript que permite a interação com bancos de dados relacionais por meio de entidades definidas no código da aplicação (Typeorm, 2024). No backend do projeto, TypeORM foi utilizado para mapear as entidades da aplicação às tabelas do banco de dados PostgreSQL, simplificando a manipulação dos dados e reduzindo a necessidade de escrita direta de consultas SQL.

pg. A biblioteca pg é um cliente PostgreSQL para Node.js responsável por estabelecer conexões com o banco de dados e executar operações de consulta e persistência (Node-Postgres, 2024). No projeto, essa biblioteca foi utilizada em conjunto com a ORM para gerenciar a comunicação entre a aplicação backend e o banco de dados.

Por fim, para garantir mecanismos de autenticação no sistema de gerenciamento de conteúdo, foi utilizada uma abordagem baseada em tokens.

JSON Web Token (JWT). JSON Web Token é um padrão aberto para transmissão segura de informações entre partes na forma de objetos JSON assinados digitalmente (Jones; Bradley; Sakimura, 2015). No projeto, JWT foi utilizado para implementar mecanismos de autenticação e controle de acesso no sistema de gerenciamento de conteúdo do backend.

3. MODELAGEM DO PROJETO

A modelagem do projeto é uma etapa crucial para garantir que todos os aspectos do desenvolvimento da aplicação sejam bem planejados e executados. A

seguir, são descritas as principais atividades envolvidas na modelagem do projeto para a criação de histórias interativas acessíveis utilizando a assistente virtual Alexa.

3.1. Levantamento de requisitos

O levantamento de requisitos é uma etapa essencial no desenvolvimento de software, pois permite identificar, analisar e documentar as funcionalidades e restrições do sistema, garantindo que a solução atenda às necessidades dos usuários e aos objetivos do projeto. Além disso, os requisitos são classificados em funcionais e não funcionais, contemplando tanto os comportamentos do sistema quanto seus atributos de qualidade, como usabilidade e desempenho (ISO/IEC/IEEE 29148, 2023; IEEE, 2022; Méndez Fernández; Wagner, 2021).

Para este projeto, os requisitos foram definidos e organizados nos Quadros 1 e 2 que apresentam a identificação, descrição e classificação de cada requisito, servindo como base para o desenvolvimento da aplicação.

Durante o desenvolvimento da solução, buscou-se garantir que os requisitos definidos fossem contemplados pela arquitetura e pelas funcionalidades implementadas. A implementação do sistema, descrita nas seções seguintes deste capítulo, evidencia como esses requisitos foram atendidos por meio da integração entre a Skill Alexa, o backend da aplicação e o banco de dados responsável pela persistência das narrativas e do progresso dos usuários.

Quadro 1: Requisitos funcionais.

ID	Descrição	Prioridade
RF001	O sistema deve solicitar a data de nascimento do usuário no primeiro acesso, a fim de controlar o acesso a conteúdos com restrição etária (por exemplo, histórias com classificação indicativa).	Alta
RF002	Permitir que o administrador cadastre, edite e exclua histórias interativas no Backoffice.	Alta
RF003	A Skill Alexa deve iniciar uma história com base na escolha do usuário.	Alta
RF004	A Skill deve interpretar comandos de voz e redirecionar o usuário conforme a escolha feita.	Alta

RF005	Permitir a visualização e edição dos fluxos narrativos no Backoffice	Média
RF006	Registrar os acessos administrativos com autenticação baseada em token JWT.	Alta
RF007	Exibir logs de operação e falhas no Backoffice.	Média
RF008	Disponibilizar <i>endpoint</i> seguro de API para comunicação entre Backoffice e a Skill.	Alta

Fonte: Acervo do autor

Quadro 2: Requisitos não funcionais.

ID	Descrição	Prioridade
RNF001	A Skill Alexa deve responder às interações do usuário em menos de 3 segundos.	Alta
RNF002	A Skill e o Backoffice devem estar disponíveis 24/7.	Média
RNF003	O Backoffice deve ter autenticação baseada em <i>token</i> JWT.	Alta
RNF004	A arquitetura deve suportar aumento de usuários sem degradação.	Média
RNF005	O código deve ser modular, testável e documentado.	Alta
RNF006	A Skill deve oferecer experiência de voz natural e intuitiva.	Alta
RNF007	A Skill deve funcionar em diversos dispositivos Alexa.	Média
RNF008	A aplicação deve lidar com erros e apresentar mensagens claras.	Alta
RNF009	A Skill deve gerar logs integrados ao Amazon CloudWatch.	Média

Fonte: Acervo do autor

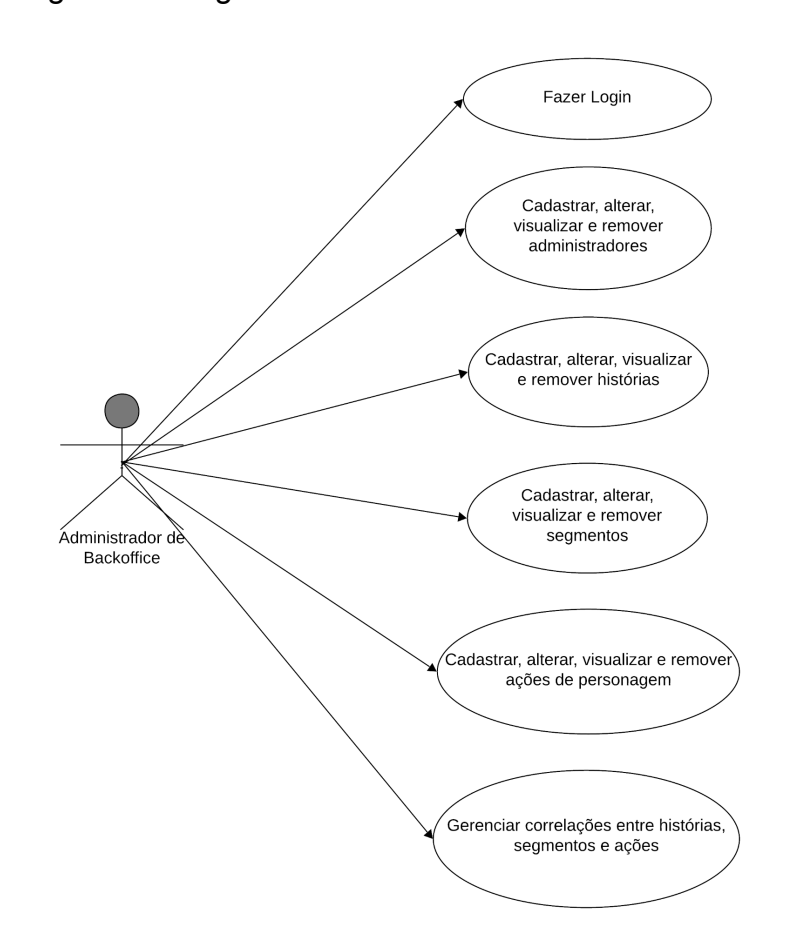
3.2. Diagramas de casos de uso

O diagrama de caso de uso é uma ferramenta amplamente utilizada na engenharia de *software* para representar, de forma visual e simplificada, as interações entre os usuários (atores) e o sistema, permitindo identificar e compreender as funcionalidades que o sistema deve oferecer. Esse tipo de diagrama auxilia na comunicação entre desenvolvedores, *stakeholders* e usuários,

facilitando o entendimento dos requisitos funcionais e contribuindo para a definição clara do escopo do sistema (OMG, 2021; Torres; Martins, 2021).

Além disso, os diagramas de caso de uso, apresentados nas Figuras 1 e 2, são úteis nas fases iniciais do desenvolvimento, pois ajudam a organizar os requisitos e servem como base para atividades posteriores, como modelagem, implementação e testes do *software* (Ameller; Franch, 2022).

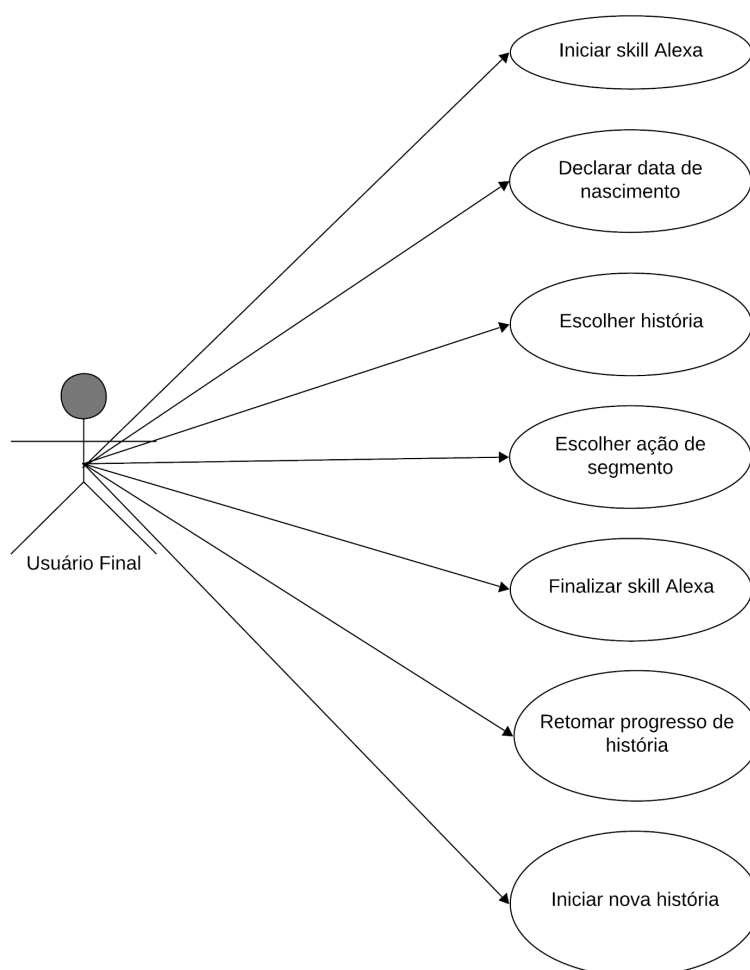
Figura 1 - Diagrama de Casos de uso do Administrador



Fonte: Elaborado pelo autor.

O diagrama apresentado na figura 1 mostra as ações do administrador no gerenciamento do conteúdo disponibilizado. Esse usuário é responsável por gerenciar o cadastro de outros administradores e do conteúdo das histórias (histórias, segmentos e ações). Seu acesso é feito através de um login gerando uma sessão na aplicação.

Figura 2 - Diagrama de Casos de Uso do Usuário Final



Fonte: Elaborado pelo autor.

As ações do usuário final, representadas na Figura 2, são compostas basicamente de ações de consumo das histórias e são feitas por comandos de voz dadas diretamente ao dispositivo Alexa. A este usuário cabe ações como, iniciar nova história, continuar um progresso salvo, escolher as ações do personagem a cada segmento e a finalização da skill.

3.3. Arquitetura da solução

A arquitetura da solução define a organização estrutural do sistema, seus componentes e as interações entre eles, sendo essencial para garantir atributos de

qualidade como escalabilidade, manutenção e desempenho (ISO/IEC/IEEE 42010, 2022).

A arquitetura de *software* estabelece os elementos fundamentais de um sistema e seus relacionamentos, servindo como base para sua compreensão e evolução. Além disso, arquiteturas bem definidas facilitam a integração entre diferentes componentes e tecnologias, especialmente em sistemas distribuídos e baseados em serviços (Kruger; Matthes, 2021; Auer et al., 2021).

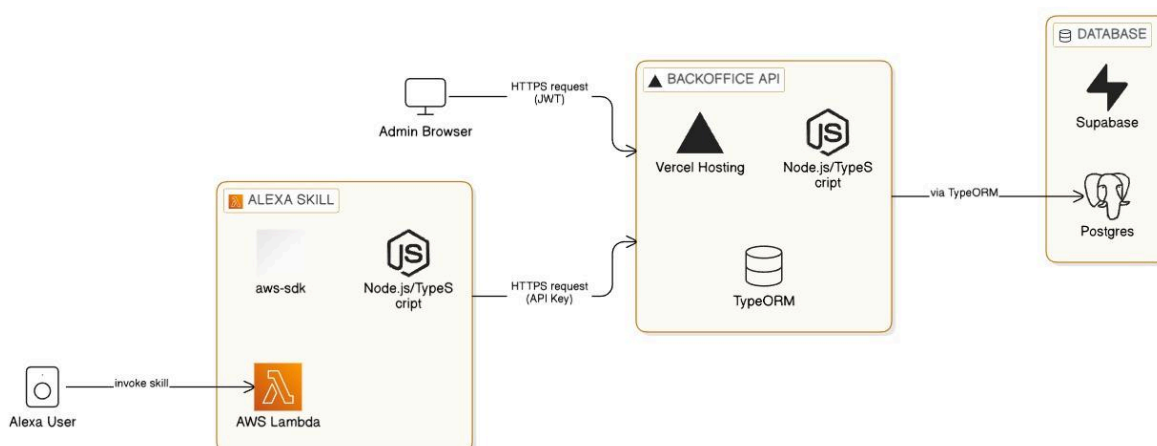
A solução proposta é composta por duas aplicações principais, cada uma com sua própria arquitetura, conjunto de tecnologias e responsabilidades específicas, que se integram para proporcionar a experiência completa de histórias interativas por meio da assistente virtual Alexa. Essa abordagem segue o modelo de separação de responsabilidades, amplamente utilizado para melhorar a organização, escalabilidade e manutenção do sistema. (Richards; Ford, 2020; García-Galán et al., 2022; Microsoft, 2023)

A integração entre o Backoffice e a Skill Alexa constitui um componente essencial para o funcionamento da solução, sendo responsável por viabilizar a comunicação entre o sistema de gerenciamento de conteúdo e a interface de interação por voz. Por meio dessa integração, o Backoffice atua como provedor central das informações narrativas, disponibilizando histórias, segmentos, ações e dados de progresso, enquanto a Skill Alexa é responsável por processar as interações do usuário e solicitar dinamicamente os conteúdos necessários para a continuidade da narrativa.

Esse modelo permite a separação entre a camada de gerenciamento e a camada de interação, promovendo maior organização, escalabilidade e flexibilidade da aplicação. Dessa forma, qualquer atualização realizada no Backoffice é refletida diretamente na experiência do usuário, sem a necessidade de alterações na Skill.

O diagrama de arquitetura do sistema, apresentado na Figura 3, ilustra essa comunicação e integração entre os componentes, evidenciando o fluxo de dados desde o gerenciamento do conteúdo até sua execução na interface de voz.

Figura 3 - Arquitetura do Sistema



Fonte: Acervo do Autor.

O diagrama representado na figura 3 destaca a integração entre a Skill Alexa, executada por meio do AWS Lambda, o Backoffice responsável pela lógica da aplicação e o banco de dados PostgreSQL hospedado no Supabase. O diagrama também ilustra os fluxos de comunicação entre os componentes, incluindo as requisições da Skill e do administrador à API do Backoffice, responsável pelo gerenciamento das histórias interativas e pela persistência dos dados.

A arquitetura adotada foi definida com o objetivo de promover a separação de responsabilidades entre os componentes do sistema, permitindo maior organização e facilidade de manutenção. A utilização de uma função serverless por meio do AWS Lambda para execução da Skill Alexa, em conjunto com uma API REST no Backoffice, possibilita um modelo desacoplado, no qual a interface conversacional e a lógica da aplicação podem evoluir de forma independente. Além disso, essa abordagem favorece a escalabilidade da solução e a flexibilidade no gerenciamento das narrativas interativas.

3.4 Diagrama de Entidade-Relacionamento (DER)

O Diagrama de Entidade-Relacionamento (DER) é uma técnica amplamente utilizada na modelagem conceitual de dados, permitindo representar de forma clara as entidades de um sistema, seus atributos e os relacionamentos entre elas. Esse

modelo foi originalmente proposto por Chen (1976), sendo reconhecido como uma abordagem fundamental para o projeto de bancos de dados, pois facilita a compreensão da estrutura lógica antes da implementação física.

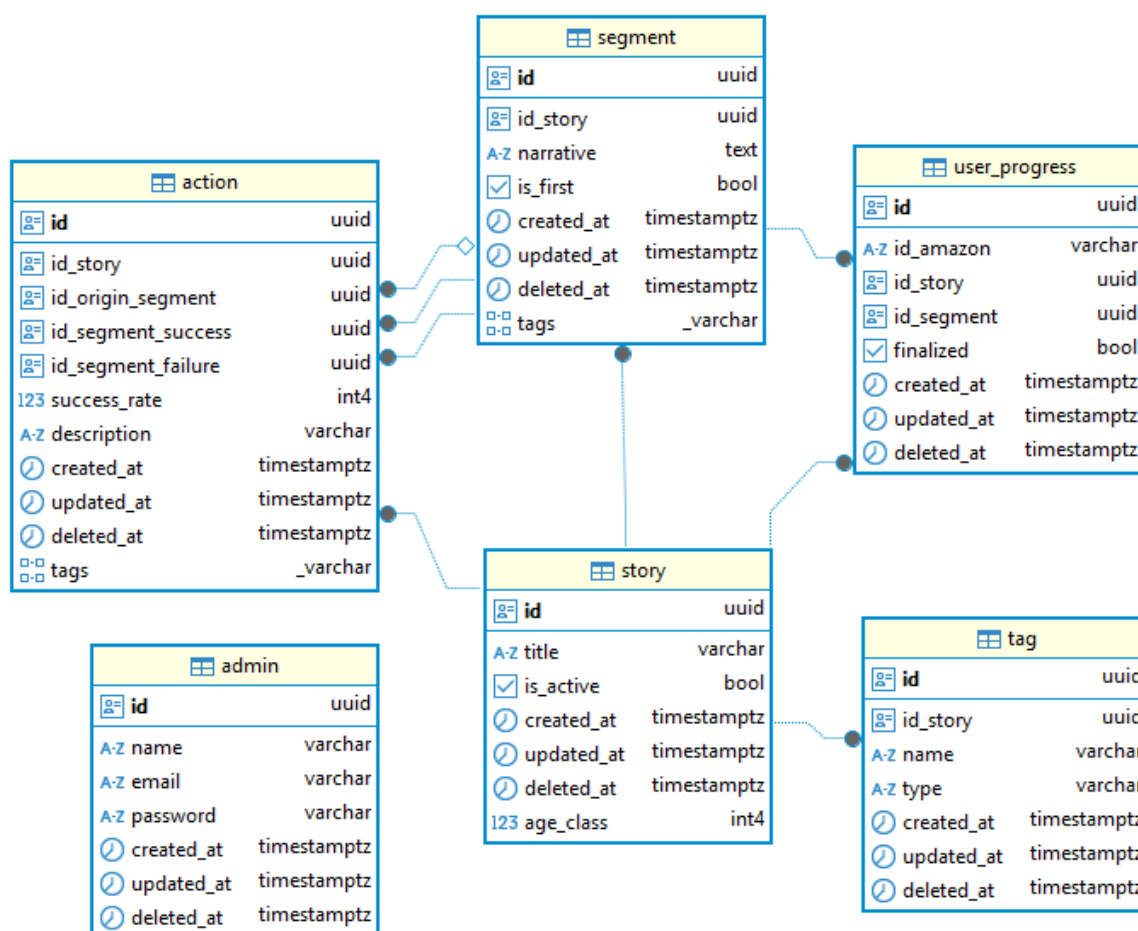
Segundo Elmasri e Navathe (2016), o DER desempenha um papel essencial no desenvolvimento de sistemas baseados em bancos de dados relacionais, contribuindo para a organização, integridade e consistência das informações. Além disso, Silberschatz, Korth e Sudarshan (2019) destacam que a modelagem conceitual por meio do modelo entidade-relacionamento permite abstrair os requisitos de dados do sistema e servir como base para o projeto lógico e físico do banco de dados (ISO/IEC 9075, 2023; Coronel; Morris, 2021; Odeh; Sharaf, 2022)

O DER do projeto (Figura 3), evidencia as principais entidades do sistema, seus atributos e seus relacionamentos. Esse modelo representa a estrutura do banco de dados adotado, servindo como base para o armazenamento e o gerenciamento das informações relacionadas às histórias interativas, seus segmentos, ações narrativas e o progresso dos usuários ao longo da experiência.

O diagrama contempla entidades como *admin*, *story*, *segment*, *action*, *user*, *progress* e *tag*, permitindo representar de forma organizada as relações entre as histórias, as escolhas realizadas pelos usuários e a persistência do estado narrativo.

Dessa forma, o DER apresentado na Figura 4 fornece uma visão estruturada dos dados do sistema, apoiando diretamente sua implementação em um banco de dados relacional e garantindo maior consistência e rastreabilidade das informações.

Figura 4 – Modelo Entidade-Relacionamento do projeto



Fonte: Elaborado pelo autor.

A figura 4 apresenta o DER do sistema, mostrando as entidades responsáveis pelas histórias, segmentos, ações e progresso do usuário, bem como seus relacionamentos. O diagrama representa a estrutura do banco de dados utilizado para gerenciar as narrativas interativas.

3.5. Interface

A definição da interface do sistema foi orientada pelo seu objetivo principal: a disponibilização de uma *skill* para a assistente virtual Amazon Alexa. Nesse contexto, a própria interação por voz constitui a interface do usuário final, ocorrendo por meio dos dispositivos compatíveis com a plataforma Amazon. Assim, a experiência do usuário é mediada pela interface conversacional oferecida pela

Alexa, não havendo a necessidade de uma interface gráfica tradicional para acesso às funcionalidades centrais do sistema.

No que se refere ao backoffice da solução, optou-se por não desenvolver uma interface administrativa própria durante esta etapa do projeto. Essa decisão foi fundamentada em restrições de tempo e escopo, considerando que o foco principal do trabalho era a implementação e o funcionamento da skill. Como o sistema foi concebido prioritariamente para interação por comandos de voz, a criação de um painel administrativo não era essencial para a validação dos objetivos propostos neste projeto.

Além disso, as configurações e testes necessários ao funcionamento da aplicação puderam ser realizados por meio das ferramentas disponibilizadas pela própria plataforma de desenvolvimento da Alexa, como o *console* da Amazon Developer Console. Dessa forma, garantiu-se o gerenciamento e a validação das funcionalidades sem a necessidade de uma interface adicional, mantendo o escopo do projeto alinhado às limitações temporais e aos objetivos acadêmicos estabelecidos.

4. SOFTWARE

Este capítulo apresenta a implementação da solução proposta, detalhando os principais componentes que compõem o sistema, bem como a forma como eles se integram para viabilizar a execução de histórias interativas por meio da Skill Alexa.

São descritos os aspectos estruturais do banco de dados, a arquitetura da Skill de voz, as tecnologias empregadas e os fluxos de funcionamento que regem a interação entre o usuário final e o Backoffice. Além disso, são apresentados os principais fluxogramas que representam o comportamento do sistema, desde o início da interação até a execução da narrativa e o gerenciamento do progresso do usuário.

O código-fonte desenvolvido neste trabalho encontra-se disponível em repositório versionado, cujo endereço é apresentado no **Apêndice A** deste documento. O projeto está dividido em dois componentes principais: a Skill desenvolvida para a plataforma Amazon Alexa e a aplicação backend responsável

pelo gerenciamento de conteúdo e pela disponibilização dos serviços utilizados pela *skill*.

A disponibilização pública da aplicação funcional para testes não foi realizada, pois a publicação de Skills Alexa para acesso aberto requer processo de submissão, revisão e aprovação pela Amazon, além de possíveis custos de infraestrutura associados à hospedagem contínua da aplicação. Dessa forma, os testes do sistema foram realizados em ambiente controlado de desenvolvimento, utilizando dispositivos compatíveis com a assistente Alexa.

4.1. Implantação

A etapa de implementação corresponde ao momento em que os modelos e especificações produzidas nas fases anteriores são transformados em código executável, integrando componentes e viabilizando o funcionamento do sistema.

Conforme descrevem Klaus Pohl, Günter Böckle e Frank J. van der Linden (2005), a implementação materializa as decisões de projeto por meio da construção dos artefatos de *software* e sua integração, permitindo a verificação prática do comportamento do sistema. Dessa forma, essa etapa não se limita à disponibilização, mas envolve a demonstração concreta de como o sistema foi desenvolvido e como suas funcionalidades operam.

4.1.1. Backoffice

O Backoffice é uma aplicação *web* responsável pelo gerenciamento das histórias interativas e pelo fornecimento de conteúdo para a Skill Alexa. Ele atua como o sistema de gerenciamento de conteúdo do projeto, permitindo a criação, edição e organização das histórias, bem como o controle do progresso dos usuários e a adequação do conteúdo à classificação etária. Além dessas funcionalidades administrativas, o Backoffice participa ativamente da lógica de decisão relacionada à escolha das histórias e de ações, considerando a existência de progresso previamente salvo e a compatibilidade do conteúdo com a faixa etária do usuário.

A estrutura do Backoffice é baseada em camadas, seguindo um modelo que promove a separação de responsabilidades e a modularização, conforme descritas a seguir:

- **Controllers:** Esta camada é responsável por receber as requisições HTTP, validar os dados de entrada, orquestrar a chamada aos usecases apropriados e formatar as respostas HTTP. Atua como interface externa da API.
- **Domain/Usecases:** Esta é a camada de lógica de negócio central da aplicação. Cada *usecase* encapsula um objetivo de negócio específico, como criar uma nova história, adicionar um segmento ou autenticar um usuário. Eles coordenam as operações entre os *repositories* e aplicam as regras de negócio.
- **Repositories:** Esta camada abstrai a comunicação direta com o banco de dados. Ela é responsável por realizar as operações de persistência de dados (consultas, inserções, atualizações e exclusões) utilizando o TypeORM. Cada *repository* é específico para uma entidade do domínio.

Para a hospedagem da aplicação Backoffice e seu banco de dados, foram utilizadas plataformas em nuvem que oferecem camadas gratuitas, ideais para projetos de TCC:

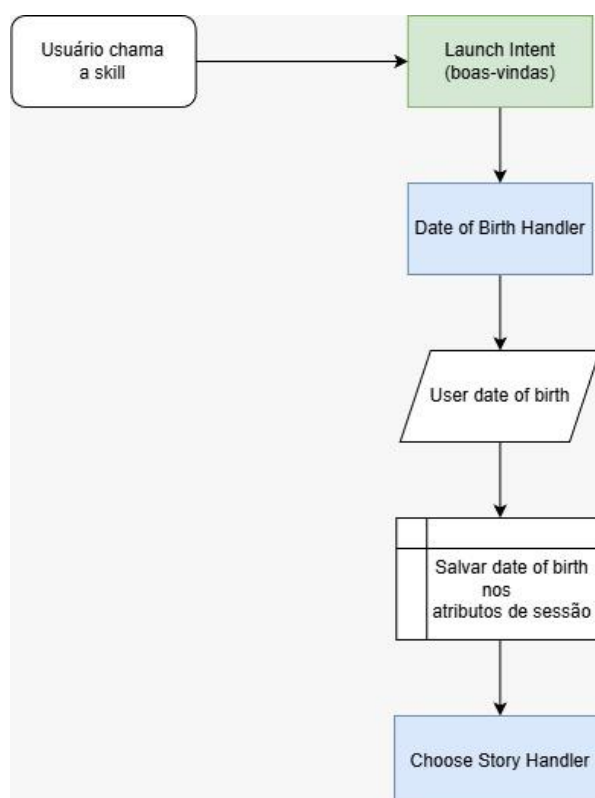
- **Banco de Dados:** Supabase (*Free tier*). O Supabase fornece um banco de dados Postgres gerenciado, além de outras ferramentas para desenvolvimento de *backend*.
- **Aplicação:** Vercel (*Free Tier*). A Vercel é uma plataforma de hospedagem para aplicações *web* e APIs *serverless*, permitindo o *deploy* rápido e escalável do Backoffice.

4.1.2. Skill Alexa

A Skill Alexa é a interface de voz principal do projeto, permitindo a interação dos usuários com as histórias interativas por meio de comandos de voz. Seu funcionamento é estruturado em etapas sequenciais que abrangem a inicialização da aplicação, a coleta da data de nascimento para controle de conteúdo por faixa etária, a seleção e progressão das narrativas e o salvamento do progresso, conforme representado no fluxograma a seguir, que detalha as decisões e ações realizadas em cada fase da interação.

A Figura 5 apresenta a primeira etapa do fluxo de execução da Skill, correspondente ao processo de inicialização do sistema. Nessa fase, ocorre a saudação ao usuário, a ativação da aplicação e a coleta de informações básicas, como a data de nascimento. Esses dados são utilizados para personalizar a experiência e aplicar restrições de conteúdo conforme a faixa etária.

Figura 5 - Fluxo de inicialização



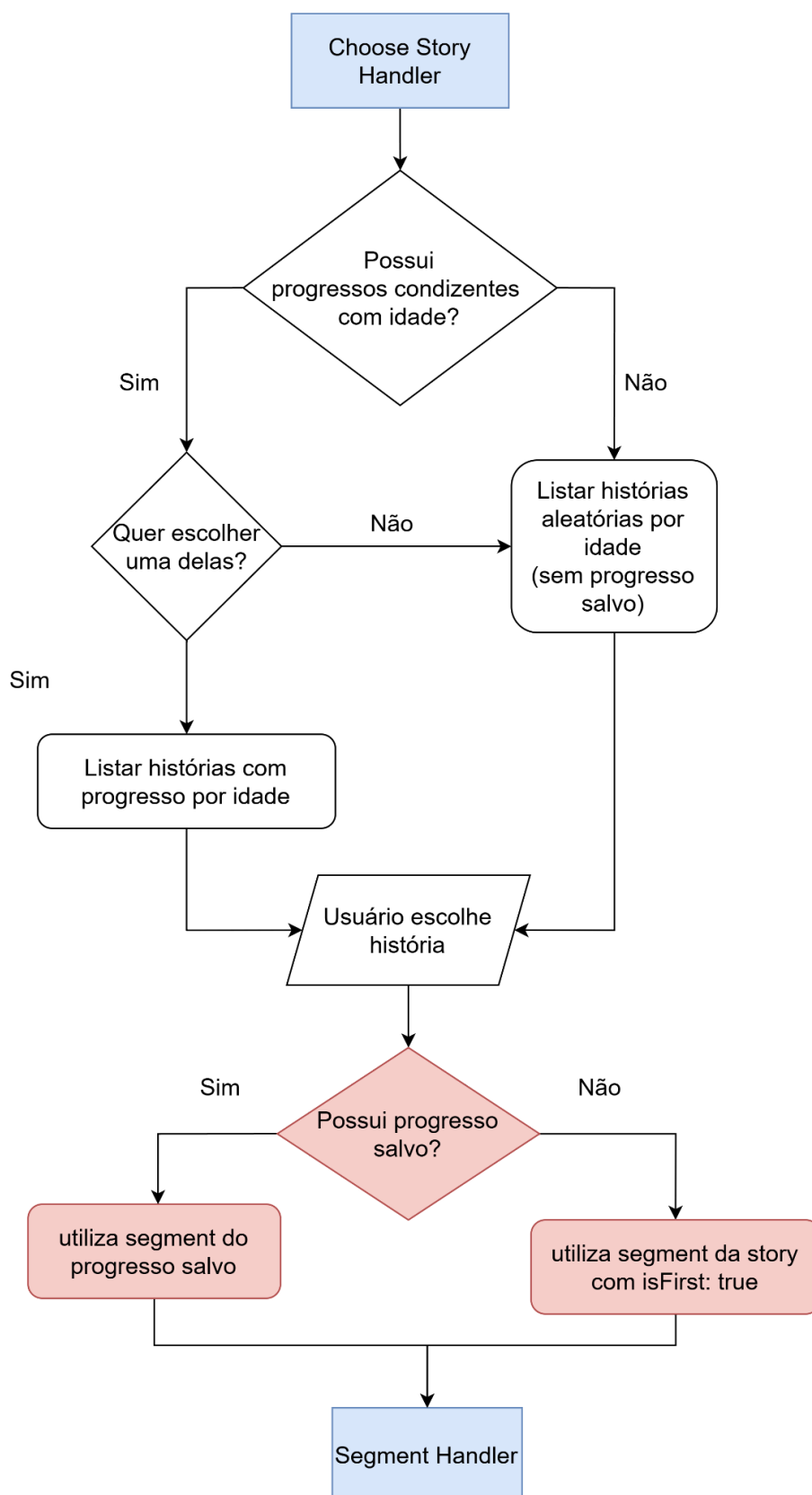
Fonte: Acervo do autor.

A figura 5 mostra o processo inicial da Skill Alexa, incluindo a saudação, a coleta da data de nascimento e o armazenamento dessas informações na sessão. Esses dados são utilizados para personalizar a experiência e permitir a seleção adequada da história.

Essa etapa representa o processo de seleção do conteúdo narrativo na Skill Alexa. Após a validação das informações do usuário, o sistema verifica a existência de progresso previamente salvo e oferece a opção de retomar uma história em andamento ou iniciar uma nova narrativa.

A disponibilização das histórias também considera a classificação etária do usuário, garantindo que o conteúdo apresentado seja adequado ao seu perfil, como ilustrado na figura 6.

Figura 6 - Fluxo de seleção de histórias



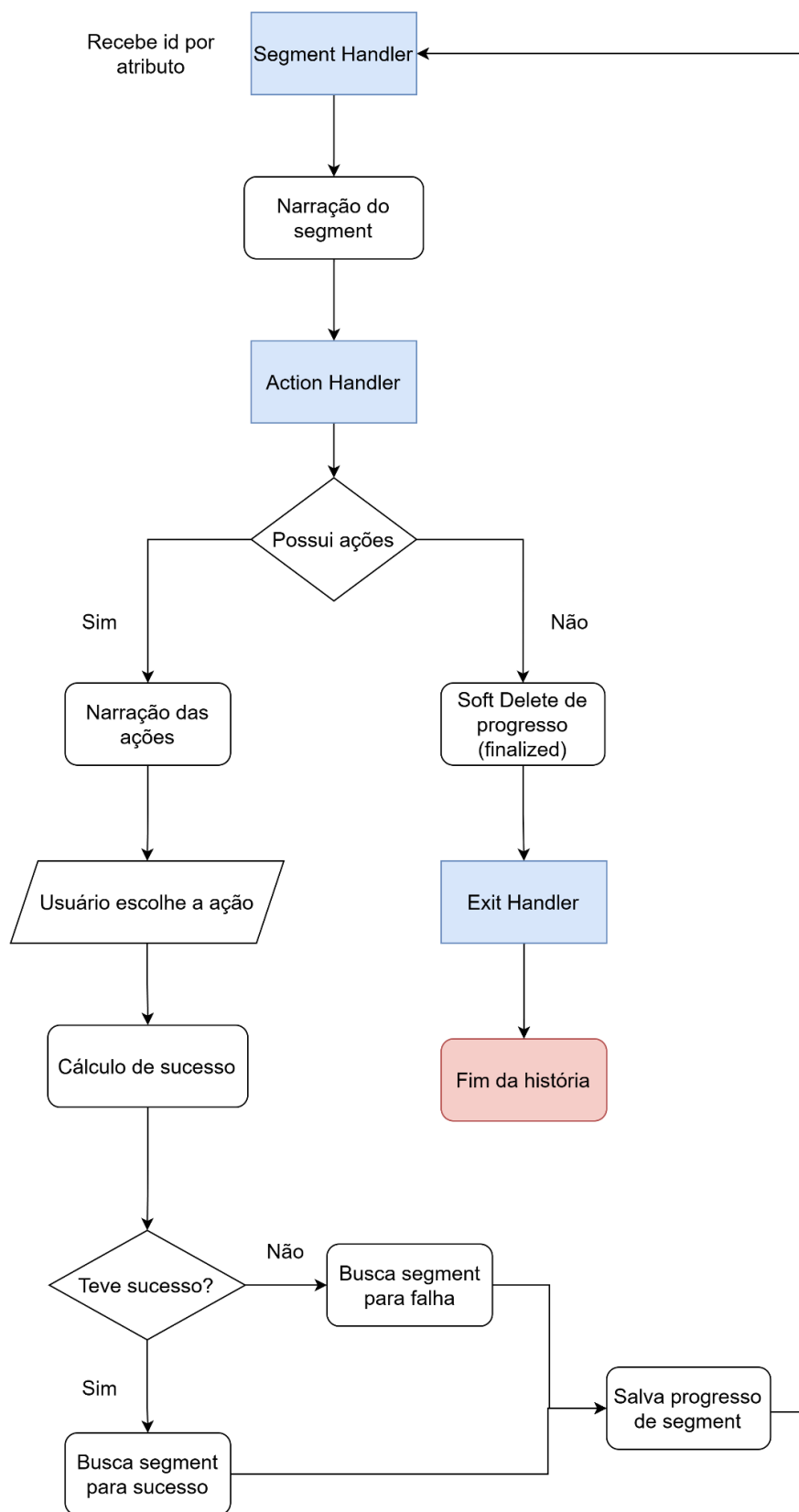
Fonte: Acervo do autor.

O diagrama da Figura 6 mostra o processo de verificação de progresso salvo e a seleção de histórias, permitindo ao usuário retomar ou iniciar uma nova narrativa conforme sua faixa etária.

A etapa seguinte demonstrada na figura 6 representa o processo de progressão da narrativa e o gerenciamento do estado da história. Nesse fluxo, o usuário realiza escolhas que determinam os próximos segmentos do enredo, caracterizando a interatividade da aplicação. O sistema também realiza o salvamento automático do progresso e identifica quando a história é finalizada, garantindo a continuidade ou o encerramento adequado da experiência.

A figura 7 mostra o momento da narração do segmento, seguido da apresentação para escolha das ações pelo usuário. Exibe também o fluxo de decisão do sucesso da ação, salvamento do progresso e finalização da história.

Figura 7 - Progressão de narrativa e escolha de ação

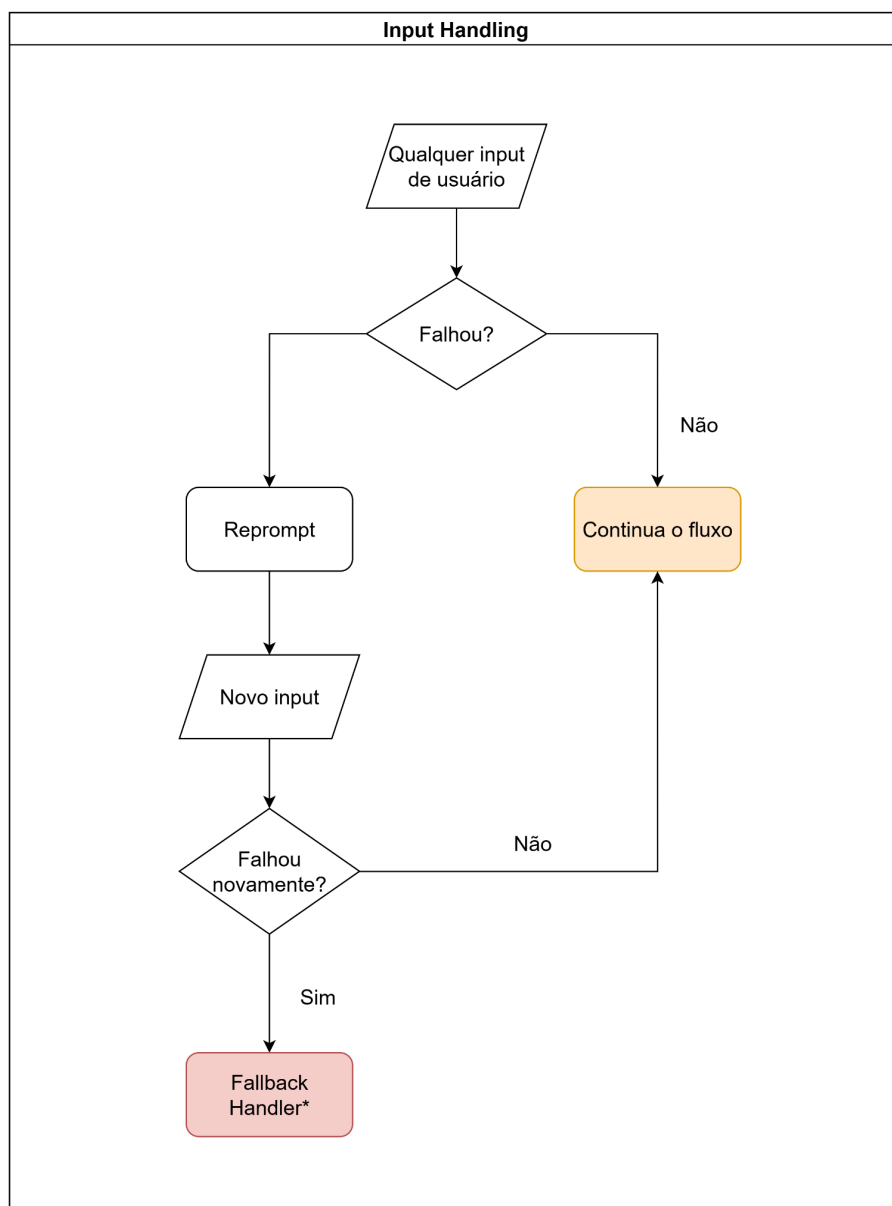


Fonte:Acervo do autor.

Adicionalmente, a Skill é responsável pelo processamento e tratamento das entradas fornecidas pelo usuário durante a interação. Esse processo compreende a validação das *intents* reconhecidas, a solicitação de novas entradas por meio do mecanismo de *reprompt* quando necessário e, em situações de falhas consecutivas ou de não reconhecimento da intenção, o acionamento do *Fallback Handler*.

Esse mecanismo de *reprompt* contribui para aumentar a robustez e a confiabilidade do sistema, garantindo maior tolerância a erros e assegurando a continuidade da interação, conforme apresentado na Figura 8.

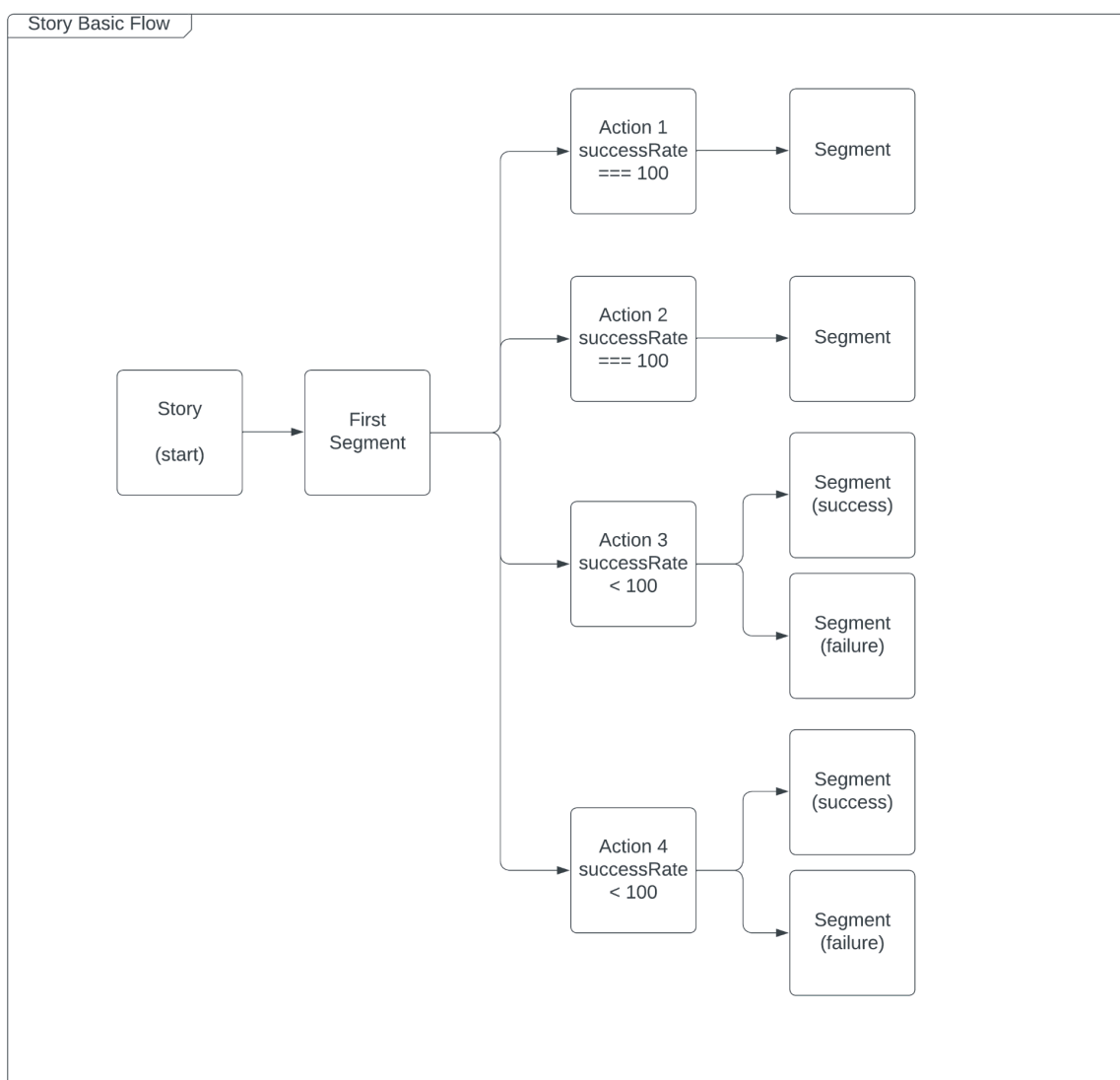
Figura 8 – Fluxo de tratamento de entradas do usuário.



Fonte: Elaborado pelo autor.

O Fluxo Básico da História (*Story Basic Flow*), apresentado na Figura 9, descreve a lógica de navegação e interação do usuário ao longo dos diferentes segmentos da narrativa, evidenciando como as escolhas realizadas influenciam as ramificações do enredo e conduzem a distintos desfechos. Esse modelo representa a estrutura conceitual das histórias interativas, organizando a relação entre histórias, segmentos e ações, bem como a definição de caminhos de sucesso ou falha a partir das decisões do usuário.

Figura 9 – Fluxo simplificado da história



Fonte: Elaborado pelo autor

Quando um usuário interage com a Alexa (por exemplo, “Alexa, abra Voz da Aventura”), o fluxo geral ocorre da seguinte forma:

- A fala do usuário é processada pela Alexa, que a encaminha para a Skill correspondente hospedada de forma *serverless* em Lambda AWS.
- A Skill Alexa, ao receber a requisição, utiliza seus *handlers* para identificar a *intent* do usuário.
- A Skill, através de sua camada API, faz requisições HTTP para o Backoffice para obter os detalhes da história solicitada, incluindo o segmento atual e as opções disponíveis.
- O Backoffice processa a requisição, consulta o banco de dados e retorna os dados relevantes para a Skill Alexa.
- A Skill Alexa narra o trecho da história para o usuário e apresenta as opções de escolha por voz.
- O usuário seleciona uma opção por comando de voz.
- A Skill Alexa processa a nova *intent*, atualiza o estado da interação (qual segmento o usuário está, quais escolhas foram feitas) e busca o próximo segmento da história no Backoffice.
- Este ciclo se repete até que a história seja finalizada ou o usuário decida parar.

4.2. Metodologia de Avaliação de Usabilidade

A avaliação da usabilidade do sistema desenvolvido foi realizada por meio da aplicação da System Usability Scale (SUS), um instrumento amplamente utilizado para mensurar a percepção de usabilidade de sistemas interativos. A escala foi proposta por Brooke (1996) e consiste em um questionário padronizado composto por dez afirmações que avaliam diferentes aspectos da experiência do usuário ao interagir com um sistema.

Cada uma das afirmações presentes no questionário SUS é respondida por meio de uma escala do tipo Likert de cinco pontos, que varia de “discordo totalmente” a “concordo totalmente”. As questões são elaboradas de forma alternada entre afirmações positivas e negativas, com o objetivo de reduzir vieses de resposta

e proporcionar uma avaliação mais equilibrada da percepção do usuário em relação ao sistema.

A pontuação da escala é calculada a partir de um procedimento específico de conversão das respostas. Para as questões ímpares, subtrai-se o valor 1 da resposta fornecida pelo participante. Já para as questões pares, subtrai-se o valor da resposta de 5. Em seguida, os valores resultantes são somados e multiplicados por 2,5, produzindo uma pontuação final que varia de 0 a 100, na qual valores mais elevados indicam melhor percepção de usabilidade.

De acordo com estudos posteriores sobre a interpretação da escala, pontuações superiores a 68 são geralmente consideradas indicativas de usabilidade satisfatória, enquanto valores inferiores podem sugerir a necessidade de melhorias na interface ou na interação do sistema (Bangor; Kortum; Miller, 2009).

A escolha da escala SUS para este trabalho se justifica por sua ampla adoção em pesquisas de usabilidade, pela simplicidade de aplicação e pela capacidade de fornecer uma medida confiável da experiência do usuário mesmo com amostras relativamente pequenas de participantes. Dessa forma, a metodologia permite avaliar de forma objetiva a percepção dos usuários em relação à interação com a aplicação desenvolvida.

As afirmações utilizadas para a avaliação através da escala SUS neste trabalho estão apresentadas no Apêndice B.

4.3. Testes

Após a definição da metodologia de avaliação de usabilidade apresentada na seção anterior, foi conduzido um teste com usuários com o objetivo de avaliar a interação com a aplicação desenvolvida.

Devido às características da plataforma de desenvolvimento da assistente virtual Amazon Alexa, a skill implementada neste trabalho não foi disponibilizada publicamente na loja de aplicações da Amazon. O processo de publicação exige etapas adicionais de submissão e aprovação pela plataforma, além da necessidade

de manutenção contínua da infraestrutura de backend, o que poderia gerar custos financeiros adicionais durante o período de desenvolvimento deste trabalho. Dessa forma, os testes foram realizados utilizando o ambiente de desenvolvimento da própria plataforma.

Para a realização dos testes, foi utilizada uma conta de desenvolvedor previamente configurada para acesso à skill em fase de desenvolvimento. Um dispositivo compatível com a assistente Alexa foi utilizado para executar a aplicação, sendo disponibilizado individualmente para cada participante durante a realização do experimento.

Os testes foram realizados com a participação de 31 usuários, convidados a interagir com a aplicação por meio do dispositivo fornecido, explorando as principais funcionalidades da skill relacionadas à seleção de histórias e à escolha das ações disponíveis em cada segmento narrativo. Durante a interação, os usuários puderam utilizar comandos de voz para navegar pelas opções apresentadas pelo sistema, simulando o uso esperado da aplicação em um cenário real de utilização.

Após a interação com a aplicação, cada participante respondeu ao questionário da System Usability Scale (SUS), apresentado no Apêndice B deste trabalho. O questionário foi utilizado para registrar a percepção dos usuários em relação à facilidade de uso, clareza da interação e consistência do sistema.

As respostas coletadas foram posteriormente utilizadas para o cálculo da pontuação de usabilidade da aplicação, conforme o procedimento descrito na Seção 4.2. Os resultados obtidos a partir dessas respostas são apresentados e analisados na seção seguinte.

4.4. Resultados e Discussão

A partir dos testes de usabilidade descritos na Seção 4.3, foram coletadas as respostas dos participantes ao questionário System Usability Scale (SUS), com o objetivo de avaliar a percepção de usabilidade da aplicação desenvolvida.

Com base nas respostas fornecidas pelos participantes, foi possível calcular as pontuações individuais da escala SUS, seguindo o procedimento de cálculo descrito na Seção 4.2. Cada participante respondeu às dez afirmações do questionário, cujas respostas foram convertidas em valores numéricos que posteriormente permitiram a obtenção da pontuação final de usabilidade.

Os resultados obtidos a partir das respostas dos participantes são apresentados na Tabela 1, que reúne as pontuações individuais calculadas para cada participante, bem como a pontuação média obtida para o sistema avaliado.

Tabela 1: Pontuação SUS geral e por participante

PONTUAÇÃO SUS					
Participante 1	77,50	Participante 12	82,50	Participante 23	77,50
Participante 2	77,50	Participante 13	85,00	Participante 24	77,50
Participante 3	77,50	Participante 14	77,50	Participante 25	80,00
Participante 4	82,50	Participante 15	75,00	Participante 26	77,50
Participante 5	95,00	Participante 16	82,50	Participante 27	80,00
Participante 6	82,50	Participante 17	80,00	Participante 28	80,00
Participante 7	82,50	Participante 18	82,50	Participante 29	80,00
Participante 8	80,00	Participante 19	77,50	Participante 30	85,00
Participante 9	77,50	Participante 20	80,00	Participante 31	85,00
Participante 10	77,50	Participante 21	77,50		
Participante 11	77,50	Participante 22	82,50		
Média total: 80,40					

Fonte: Acervo do autor

Após o cálculo das pontuações individuais, foi obtida uma pontuação média de 80,40 pontos na escala SUS. A escala apresenta valores que variam de 0 a 100, sendo utilizada como um indicador da percepção de usabilidade de um sistema.

De acordo com a interpretação proposta por Bangor, Kortum e Miller (2009), pontuações superiores a 68 pontos indicam um nível de usabilidade considerado acima da média. Dessa forma, o resultado obtido sugere que o sistema desenvolvido apresenta boa usabilidade, de acordo com a percepção dos participantes envolvidos na avaliação.

Os resultados observados através do SUS indicam que a interação por voz utilizada na aplicação foi compreendida pelos usuários, permitindo a navegação pelas narrativas interativas e a seleção das ações disponíveis de forma relativamente intuitiva. Ainda assim, melhorias futuras podem ser consideradas, especialmente no que se refere à ampliação das possibilidades de interação e ao refinamento das respostas fornecidas pelo sistema.

Dessa forma, os resultados obtidos indicam que a solução desenvolvida atende ao objetivo deste trabalho de investigar o uso de interfaces conversacionais como meio de acesso a narrativas interativas. A interação por voz demonstrou-se funcional durante os testes realizados, evidenciando o potencial desse tipo de interface para ampliar as possibilidades de interação com sistemas digitais.

4.5. Comparação com Aplicações Semelhantes

Além da avaliação de usabilidade realizada por meio da escala SUS, é possível discutir o protótipo desenvolvido neste trabalho em relação a outras aplicações de narrativa interativa disponíveis para assistentes virtuais. No ecossistema da Amazon Alexa existem diversas *skills* que exploram o conceito de *storytelling* interativo, entre elas a The Magic Door (The Magic Door, 2026) e a Earplay (Earplay, 2016).

As informações utilizadas na comparação foram obtidas a partir das descrições públicas e da documentação disponível das aplicações analisadas.

Essas aplicações apresentam experiências narrativas baseadas em interação por voz, nas quais o usuário pode influenciar o desenvolvimento da história por meio de escolhas realizadas durante a narrativa. No entanto, essas soluções apresentam algumas diferenças em relação ao protótipo desenvolvido neste trabalho,

especialmente no que se refere à estrutura de modelagem narrativa e às funcionalidades disponíveis para gerenciamento de conteúdo.

No protótipo Voz da Aventura, a narrativa é estruturada em segmentos, cada um representando uma parte da história apresentada ao usuário. Para cada segmento são definidas ações específicas, que correspondem às escolhas disponíveis naquele ponto da narrativa. Cada ação pode possuir probabilidade de sucesso ou falha, sendo que cada resultado pode conduzir o fluxo narrativo para diferentes segmentos subsequentes. Essa estrutura permite a construção de múltiplos caminhos narrativos e amplia significativamente as possibilidades de modelagem de histórias interativas.

Além disso, o sistema desenvolvido conta com um *backoffice* administrativo, responsável pelo gerenciamento das histórias armazenadas na aplicação. Por meio desse componente é possível cadastrar diversas narrativas independentes, cada uma composta por uma quantidade potencialmente indefinida de segmentos e ações, permitindo que o sistema disponibilize múltiplas histórias para escolha do usuário.

Por outro lado, diferentemente de algumas aplicações existentes, o protótipo desenvolvido não implementa um sistema de inventário de itens, no qual o usuário possa coletar objetos ao longo da narrativa e utilizá-los posteriormente. Esse tipo de mecânica está presente em algumas experiências narrativas interativas, mas não foi contemplado no escopo da aplicação desenvolvida neste trabalho.

O Quadro 3 apresenta uma comparação entre algumas características das aplicações analisadas e o protótipo desenvolvido neste projeto.

Quadro 3: Comparação entre aplicações de narrativa interativa para assistentes virtuais

Característica	Voz da Aventura	The Magic Door	Earplay
Plataforma	Alexa	Alexa	Alexa
Idioma principal	Português	Inglês	Inglês
Narrativa interativa	Sim	Sim	Sim

Escolhas por comandos de voz	Sim	Sim	Sim
Persistência do Progresso	Sim	Sim	Sim
Sistema de ações com sucesso ou falha	Sim	Não identificado	Não identificado
Estrutura narrativa baseada em escolhas	Sim	Sim	Sim
Sistema de inventário de itens	Não	Sim	Não identificado
Múltiplas histórias disponíveis	Sim	Sim	Sim
Validação de faixa etária	Sim	Não identificado	Não identificado

Fonte: elaborado pelo autor com base em The Magic door (2026) e Earplay (2016).

A partir da comparação apresentada, observa-se que as aplicações analisadas e o protótipo desenvolvido neste trabalho compartilham características relacionadas à narrativa interativa baseada em escolhas realizadas por comandos de voz.

No entanto, o sistema proposto adota uma estrutura de modelagem narrativa baseada em segmentos e ações, na qual cada ação pode possuir probabilidade de sucesso ou falha, podendo direcionar a narrativa para diferentes segmentos subsequentes.

Além disso, o protótipo utiliza uma arquitetura composta por API REST e sistema de *backoffice*, responsáveis pelo gerenciamento das histórias armazenadas na aplicação. Essa estrutura permite o cadastro e a organização de múltiplas narrativas independentes, contribuindo para maior flexibilidade na criação e manutenção de histórias interativas.

5. CONSIDERAÇÕES FINAIS

Como resultado do desenvolvimento e da avaliação do protótipo, observa-se que os objetivos propostos para este trabalho foram alcançados, conforme evidenciado pela implementação da aplicação baseada em interação por voz e pelos resultados obtidos na avaliação de usabilidade realizada com usuários.

No que se refere ao uso de interfaces conversacionais, a implementação de uma interface baseada exclusivamente em interação por voz demonstrou-se funcional para permitir a navegação pelo conteúdo narrativo sem dependência direta de elementos visuais. Esse resultado evidencia o potencial das tecnologias conversacionais como alternativa para o desenvolvimento de aplicações interativas baseadas em voz.

A arquitetura desenvolvida possibilitou a navegação autônoma pelo conteúdo narrativo, validando a proposta de oferecer uma experiência de storytelling interativo mediada por comandos de voz. Paralelamente, no âmbito do design conversacional e da interação humano-computador, o sistema implementado demonstrou a aplicação efetiva de práticas estruturadas de modelagem de *intents*, gerenciamento de estados e controle do fluxo de diálogo, resultando em uma interação consistente, coerente e tecnicamente estável.

Os resultados obtidos indicam que a abordagem adotada é adequada para o desenvolvimento de aplicações baseadas em voz que exigem controle de contexto e interatividade narrativa. A avaliação de usabilidade realizada por meio da System Usability Scale (SUS) apresentou uma pontuação média classificada como elevada, reforçando a percepção positiva dos usuários em relação à experiência de uso da aplicação.

No que se refere aos objetivos definidos neste trabalho, o objetivo geral — consistente no desenvolvimento e validação de um protótipo funcional de storytelling interativo utilizando interface de voz — foi alcançado por meio da implementação completa da skill, da integração com o sistema de backoffice e da validação funcional do fluxo narrativo interativo. Da mesma forma, os objetivos específicos foram atendidos, incluindo a concepção de uma arquitetura modular e distribuída, o desenvolvimento do sistema de gerenciamento de conteúdo narrativo, a

implementação dos mecanismos de interação conversacional e a realização de testes de usabilidade em ambiente controlado.

Dessa forma, conclui-se que o projeto contribui para a investigação e aplicação de interfaces conversacionais como meio viável para a implementação de experiências interativas baseadas em voz, demonstrando o potencial desse tipo de tecnologia para aplicações que exploram interação narrativa mediada por comandos de voz.

Como perspectivas para trabalhos futuros, sugere-se a incorporação de técnicas de Inteligência Artificial generativa para a criação dinâmica de conteúdos narrativos. A utilização de modelos de linguagem capazes de gerar segmentos de história em tempo real pode ampliar significativamente o nível de interatividade e personalização da experiência, permitindo que a narrativa se adapte às escolhas do usuário de maneira mais complexa e imprevisível. Além disso, a integração de mecanismos de geração automática de enredos pode reduzir a dependência de roteiros previamente estruturados, tornando o sistema mais escalável e potencialmente mais imersivo.

REFERÊNCIAS

AMAZON. *Alexa Skills Kit Documentation*. Disponível em: <https://developer.amazon.com/alexa>. Acesso em: 11 mar. 2026.

AMAZON. Earplay: an interactive audio storyteller made even better by Alexa. Disponível em: <https://developer.amazon.com/en-US/blogs/alexa/post/Tx2TM2QAL89ND1Q/earplay-an-interactive-audio-only-storyteller-made-even-better-by-alex>. Acesso em: 16 mar. 2026.

AMELLER, David; FRANCH, Xavier. How do software architects consider non-functional requirements: an exploratory study. *Journal of Systems and Software*, v. 190, 2022. Disponível em: <https://doi.org/10.1016/j.jss.2022.111328>. Acesso em: 11 mar. 2026.

AUER, F.; LENARDUZZI, V.; FELDERER, M.; TAIBI, D. From monolithic systems to microservices: an assessment framework. *Information and Software Technology*, v. 137, 2021. Disponível em: <https://doi.org/10.1016/j.infsof.2021.106600>. Acesso em: 11 mar. 2026.

AXIOS. *Axios HTTP Client Documentation*. Disponível em: <https://axios-http.com>. Acesso em: 11 mar. 2026.

BANGOR, Aaron; KORTUM, Philip T.; MILLER, James T. Determining what individual SUS scores mean: adding an adjective rating scale. *Journal of Usability Studies*, v. 4, n. 3, p. 114–123, 2009.

BROOKE, John. SUS: a “quick and dirty” usability scale. In: JORDAN, Patrick W. et al. (ed.). *Usability Evaluation in Industry*. London: Taylor & Francis, 1996. p. 189–194.

BRYNJOLFSSON, Erik; MCAFEE, Andrew. *The second machine age: work, progress, and prosperity in a time of brilliant technologies*. New York: W. W. Norton & Company, 2014.

CHEN, Peter Pin-Shan. The entity-relationship model: toward a unified view of data. *ACM Transactions on Database Systems*, v. 1, n. 1, p. 9–36, 1976. Disponível em: <https://doi.org/10.1145/320434.320440>. Acesso em: 11 mar. 2026.

CORONEL, Carlos; MORRIS, Steven. *Database systems: design, implementation, and management*. 13. ed. Boston: Cengage Learning, 2021.

EARPLAY. Alexa Earplay skill on Amazon Echo devices. Disponível em: <https://medium.com/earplay/alexa-earplay-skill-on-amazon-echo-devices-8607e78f1519>. Acesso em: 16 mar. 2026.

ELMASRI, Ramez; NAVATHE, Shamkant B. *Fundamentals of database systems*. 7. ed. Hoboken: Pearson, 2016.

EXPRESS. *Express.js Documentation*. Disponível em: <https://expressjs.com>. Acesso em: 11 mar. 2026.

FOWLER, Martin. *Refactoring: improving the design of existing code*. 2. ed. Boston: Addison-Wesley, 2022.

HOY, Matthew B. Intelligent personal assistants: a survey of popular intelligent personal assistants. *Medical Reference Services Quarterly*, v. 37, n. 1, p. 81–88, 2018.

HOYLE, Roberto et al. Privacy and smart speakers: a multi-user perspective on Amazon Echo. *Proceedings of the ACM on Human-Computer Interaction*, v. 3, 2019. Disponível em: <https://doi.org/10.1145/3359253>. Acesso em: 11 mar. 2026.

IBGE. *Pesquisa Nacional por Amostra de Domicílios Contínua: acesso à internet e à televisão e posse de telefone móvel celular para uso pessoal 2023*. Rio de Janeiro: IBGE, 2023.

IEEE. *IEEE 830-2022: Recommended Practice for Software Requirements Specifications*. New York: IEEE, 2022.

IEEE. *IEEE 42010-2022: Systems and software engineering — architecture description*. New York: IEEE, 2022.

ILUMEO DATA SCIENCE COMPANY. Pesquisa sobre uso de assistentes virtuais por voz no Brasil: 91 % dos brasileiros já usaram assistentes por voz em dispositivos. *IPNews*, 2022. Disponível em: <https://ipnews.com.br/apenas-9-dos-brasileiros-nunca-usaram-assistentes-virtuais-por-voz/>. Acesso em: 11 mar. 2026.

INVERSIFYJS. *InversifyJS Documentation*. Disponível em: <https://inversify.io>. Acesso em: 11 mar. 2026.

ISO/IEC. *ISO/IEC 25010:2023 — Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE)*. Geneva: ISO, 2023.

ISO/IEC. *ISO/IEC 9075:2023 — Information technology — Database languages — SQL*. Geneva: ISO, 2023.

ISO/IEC/IEEE. *ISO/IEC/IEEE 29148:2023 — Systems and software engineering — Life cycle processes — Requirements engineering*. Geneva: ISO, 2023.

JONES, Michael; BRADLEY, John; SAKIMURA, Nat. *JSON Web Token (JWT)*. RFC 7519. Internet Engineering Task Force, 2015. Disponível em: <https://datatracker.ietf.org/doc/html/rfc7519>. Acesso em: 11 mar. 2026.

KRUGER, Stefan; MATTHES, Florian. Software architecture decision-making: a systematic literature review. *Journal of Systems and Software*, v. 181, 2021. Disponível em: <https://doi.org/10.1016/j.jss.2021.111036>. Acesso em: 11 mar. 2026.

LAUGWITZ, Bettina et al. Intelligent personal assistants: a systematic literature review and research agenda. *AIS Transactions on Human-Computer Interaction*, v. 12, n. 4, p. 183–212, 2020.

MARTIN, Robert C. *Clean architecture: a craftsman's guide to software structure and design*. Boston: Prentice Hall, 2017.

MÉNDEZ FERNÁNDEZ, Daniel; WAGNER, Stefan. Naming the pain in requirements engineering. *Empirical Software Engineering*, v. 26, 2021. Disponível em: <https://doi.org/10.1007/s10664-020-09915-6>. Acesso em: 11 mar. 2026.

MICROSOFT. *Azure Architecture Center: software architecture fundamentals and best practices*. 2023. Disponível em: <https://learn.microsoft.com>. Acesso em: 11 mar. 2026.

MICROSOFT. *TypeScript Documentation*. Disponível em: <https://www.typescriptlang.org>. Acesso em: 11 mar. 2026.

NODEJS. *Node.js Documentation*. Disponível em: <https://nodejs.org>. Acesso em: 11 mar. 2026.

NODE-POSTGRES. *node-postgres Documentation*. Disponível em: <https://node-postgres.com>. Acesso em: 11 mar. 2026.

OBJECT MANAGEMENT GROUP. *Unified Modeling Language (UML) Specification Version 2.5.1*. Needham, 2021. Disponível em: <https://www.omg.org/spec/UML>. Acesso em: 11 mar. 2026.

ODEH, Mohammad; SHARAF, Qutaibah. Conceptual data modeling and database design. *IEEE Access*, v. 10, 2022. Disponível em: <https://doi.org/10.1109/ACCESS.2022.3152345>. Acesso em: 11 mar. 2026.

PEARL, Cathy. *Voice user interface design*. Sebastopol: O'Reilly Media, 2016.

POHL, Klaus; BÖCKLE, Günter; VAN DER LINDEN, Frank J. *Software product line engineering: foundations, principles and techniques*. Berlin: Springer, 2005.

PORCHERON, Martin et al. Voice interfaces in everyday life. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 2018. Disponível em: <https://doi.org/10.1145/3173574.3174214>. Acesso em: 11 mar. 2026.

POSTGRESQL GLOBAL DEVELOPMENT GROUP. *PostgreSQL Documentation*. Disponível em: <https://www.postgresql.org/docs>. Acesso em: 11 mar. 2026.

PRESSMAN, Roger S.; MAXIM, Bruce R. *Engenharia de software: uma abordagem profissional*. 8. ed. Porto Alegre: AMGH, 2020.

RICHARDS, Mark; FORD, Neal. *Fundamentals of software architecture: an engineering approach*. Sebastopol: O'Reilly Media, 2020.

SANTOS, Paloma Dias; CARVALHO, José Oscar Fontanini. Assistentes virtuais e acessibilidade digital. *Revista Tecnologia e Sociedade*, v. 17, n. 48, p. 1–15, 2021.

SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. *Database system concepts*. 7. ed. New York: McGraw-Hill Education, 2019.

SOMMERVILLE, Ian. *Engenharia de software*. 10. ed. São Paulo: Pearson, 2016. Disponível em: <https://doi.org/10.5965/2316796313262024069>. Acesso em: 11 mar. 2026.

THE MAGIC DOOR. The Magic Door: Interactive adventure game for Alexa. Disponível em: <https://www.themagicdoor.org>. Acesso em: 16 mar. 2026.

THE MAGIC DOOR. Frequently Asked Questions (FAQ). Disponível em: <https://www.themagicdoor.org/faq>. Acesso em: 16 mar. 2026.

TORRES, Aline; MARTINS, Guilherme. Uso da UML como ferramenta de apoio à engenharia de requisitos. *Revista Brasileira de Computação Aplicada*, v. 13, n. 2, 2021.

TSOA. *TSOA Documentation*. Disponível em: <https://tsoa-community.github.io>. Acesso em: 11 mar. 2026.

TYPEORM. *TypeORM Documentation*. Disponível em: <https://typeorm.io>. Acesso em: 11 mar. 2026.

TYPESCRIPT. *ts-node Documentation*. Disponível em: <https://typestrong.org/ts-node>. Acesso em: 11 mar. 2026.

APÊNDICE A - Repositório do Protótipo

O código-fonte do protótipo desenvolvido neste trabalho está disponível no repositório público do GitHub, acessível pelo link:

Disponível em: <https://github.com/JoaoHAraujo/alexa-rpg>

APÊNDICE B - Questionário SUS utilizado na avaliação

O questionário abaixo corresponde às afirmações da System Usability Scale (SUS) utilizadas para avaliar a usabilidade da aplicação desenvolvida. Cada afirmação foi respondida pelos participantes utilizando uma escala do tipo Likert de cinco pontos, variando de “discordo totalmente” a “concordo totalmente”.

1. Eu acho que gostaria de usar esse sistema com frequência.
2. Eu acho o sistema desnecessariamente complexo.
3. Eu achei o sistema fácil de usar.
4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema.
5. Eu acho que as várias funções do sistema estão muito bem integradas.
6. Eu acho que o sistema apresenta muita inconsistência.
7. Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.
8. Eu achei o sistema atrapalhado de usar.
9. Eu me senti confiante ao usar o sistema.
10. Eu precisei aprender várias coisas novas antes de conseguir usar o sistema.