

Desenvolvimento de uma Skill Alexa para Automação de Comandos por Voz no Robô Temi

Luís Guilherme Gomes Ferreira Rossi¹, Fabiano Costa Teixeira²

¹Aluno de Graduação do Curso de Ciência da Computação da PUC Minas *Campus* de Poços de Caldas – MG, Brasil

² Orientador e Docente do Departamento de Ciência da Computação da PUC Minas *Campus* de Poços de Caldas – MG, Brasil

lggfrossi@gmail.com, teixeira@pucpcaldas.br

Abstract. *This work proposes the development of a Skill for the Alexa virtual assistant, aimed at controlling movement and communication missions of the Temi robot. The initiative seeks to integrate voice commands to facilitate movement operations and message delivery, utilizing external API resources.*

Resumo. *Este trabalho propõe o desenvolvimento de uma Skill para assistente virtual Alexa, destinada ao controle de missões de movimentação e comunicação do robô Temi. A iniciativa busca integrar comandos de voz para facilitar operações de movimentação e envio de mensagens, utilizando recursos de APIs externas.*

1. Introdução

A inteligência artificial (IA) tem revolucionado as interações entre seres humanos e máquinas, especialmente através do desenvolvimento de assistentes de voz inteligentes. Esses sistemas, como a Alexa da Amazon, o Google Assistant e a Siri da Apple, representam a convergência de avanços em processamento de linguagem natural, aprendizado de máquina e computação em nuvem (Amazon Alexa Developer Blog, 2024).

O conceito de assistentes de voz ganhou força a partir da década de 2010, com a popularização de dispositivos capazes de compreender comandos falados e executar ações de forma autônoma. Atualmente, estima-se que existam mais de 8 bilhões de assistentes digitais em uso no mundo, número que ultrapassa a população humana e demonstra o papel central que essas tecnologias assumiram no cotidiano (Statista, 2024).

Em paralelo a isso, o desenvolvimento de robôs móveis para uso em ambientes corporativos, hospitalares e domésticos tornou-se uma tendência crescente. Robôs como o Temi, por exemplo, combinam navegação autônoma e interação por voz, criando possibilidades para automação de tarefas e melhoria da experiência de usuários em diferentes setores (RobotTemi, 2024).

No contexto deste trabalho, buscou-se integrar a tecnologia de assistente de voz da Alexa com o robô Temi, permitindo a criação de funções personalizadas. A partir de comandos de voz simples, o usuário pode controlar ações do robô de forma prática e intuitiva, tornando a interação homem-máquina ainda mais fluida e natural.

Este projeto apresenta o desenvolvimento de uma Skill Alexa personalizada que envia comandos via API ao robô Temi, abrangendo funcionalidades como consulta de status,

verificação e movimentação de pontos pré-definidos no mapa e emissão de mensagens faladas.

1.1 Justificativa

A crescente utilização de robôs em ambientes profissionais e domésticos motiva a criação de meios de interação. Um sistema de controle por voz para o Temi permitirá uma interação eficiente, sem a necessidade de interfaces físicas ou apps dedicados, ampliando o acesso e a inclusão digital.

1.2 Objetivos

Desenvolver uma Skill Alexa que permita ver a disponibilidade do robô e criação e cancelamento de missões, de maneira dinâmica, através de comandos de voz.

1.3 Objetivos Específicos

- Construir e configurar uma Skill Alexa
- Desenvolver Intents personalizadas
- Implementar comunicação com API
- Compatibilidade com dispositivos Alexa físicos
- Diálogo intuitivo homem-máquina

2. Robô Temi

O Temi é um robô pessoal autônomo desenvolvido para atuar como assistente de voz e plataforma de serviços interativos em ambientes domésticos e corporativos. O projeto do Temi foi iniciado em 2015 pela empresa Roboteam, de Israel, e o produto foi oficialmente lançado em 2018, trazendo uma solução de robótica móvel voltada para comunicação, acessibilidade e entretenimento (ROBOTEMI, 2024).

O robô utiliza tecnologias avançadas como o mapeamento simultâneo e localização (SLAM - Simultaneous Localization and Mapping), reconhecimento de voz, controle autônomo de navegação e videoconferência em alta definição. Equipado com integração nativa com assistentes virtuais como Alexa e Google Assistant, o Temi oferece aos usuários a capacidade de comandar o robô por voz para executar tarefas como deslocar-se até locais específicos, realizar chamadas de vídeo, tocar músicas ou fornecer informações (ROBOTEMI, 2024).

Figura 1: Robô Temi



Fonte: <https://pluginbot.ai/temi/>

3. Alexa

A Alexa é o assistente de voz inteligente desenvolvido pela Amazon, lançado em 2014 juntamente com o primeiro dispositivo da linha Echo (AMAZON, 2024). Baseada em tecnologias de inteligência artificial e processamento de linguagem natural (PLN), a Alexa permite que usuários interajam por meio de comandos de voz para realizar tarefas cotidianas, obter informações, controlar dispositivos e executar aplicações personalizadas, denominadas skills.

3.1. Equipamento

Alexa opera em uma variedade de dispositivos físicos da linha Echo, que incluem alto-falantes inteligentes, displays com tela sensível ao toque e, mais recentemente, equipamentos integrados em automóveis e sistemas residenciais. Esses dispositivos são projetados com múltiplos microfones e algoritmos de beamforming para captar comandos de voz mesmo em ambientes ruidosos (AMAZON DEVELOPER, 2024).

3.2. Inteligência Artificial

A base de funcionamento da Alexa é sustentada por sistemas de inteligência artificial que combinam reconhecimento automático de fala (ASR – Automatic Speech Recognition), entendimento de linguagem natural (NLU – Natural Language Understanding) e motores de decisão para interpretar e executar as solicitações dos usuários. Essa estrutura é continuamente aprimorada através de técnicas de aprendizado de máquina, permitindo adaptações ao idioma, sotaque, contexto e preferências individuais dos usuários (HASSAN et al., 2019).

3.3. Plataforma

Sobre o desenvolvimento, a Amazon disponibiliza gratuitamente o Alexa Skills Kit (ASK), um conjunto de APIs, ferramentas e frameworks que permite a criação de novas interações (Skills) para a Alexa. Essas skills podem ser expostas em nuvem via AWS Lambda e integradas com fontes externas de dados ou APIs, como é o caso da PluginBot para controle do robô Temi.

4. Desenvolvimento da Skill para controle do robô

4.1. Criação da Skill

A criação da Skill foi realizada através do Console de Desenvolvedor Alexa da Amazon. Neste ambiente, foi definido o modelo de interação (Interaction Model), criando intents customizadas que correspondem às funcionalidades desejadas para o robô Temi. Entre as intents desenvolvidas, estão:

- **GetRobotsIntent:** Responsável por consultar a API da PluginBot e listar os robôs registrados para o aplicativo, apresentando ao usuário informações como nome, tipo, código de identificação e status atual de operação de cada robô.
- **GetMapPointsIntent:** Consulta os pontos de movimentação disponíveis no mapa configurado para o robô, retornando ao usuário a lista de nomes dos locais de forma organizada e compreensível.
- **CriarMissaoFalaIntent:** Permite ao usuário criar uma missão de fala, onde o robô pronuncia uma mensagem personalizada fornecida pelo próprio usuário durante a interação.

- **CriarMissaoMovimentacaoIntent:** Destinada à criação de missões de movimentação, esta intent solicita ao usuário a indicação de um dos pontos cadastrados no mapa para onde o robô deve se deslocar.
- **VerMissaoIntent:** Consulta o status da missão ativa vinculada ao robô, informando se a missão está em execução, foi finalizada, foi cancelada ou expirou.
- **CancelarMissaoIntent:** Executa o cancelamento da missão em andamento no robô, removendo o identificador da missão (`mission_id`) armazenado nos atributos persistentes da skill.

Além dessas intents específicas, o projeto também incorporou as intents padrão fornecidas pela plataforma Alexa, que são fundamentais para uma experiência de interação mais natural e robusta:

- **LaunchRequest:** Acionada no momento da abertura da skill, responsável por dar as boas-vindas ao usuário e apresentar as opções iniciais de interação.
- **HelpIntent:** Fornece orientações gerais sobre os comandos possíveis na skill, ajudando o usuário em caso de dúvidas.
- **CancelIntent e StopIntent:** Permitem que o usuário encerre a interação com a skill de forma amigável, utilizando comandos de cancelamento ou parada.
- **FallbackIntent:** Trata requisições que não correspondem a nenhuma intent definida, fornecendo uma resposta alternativa para manter o diálogo.
- **SessionEndedRequest:** Lida com o encerramento natural da sessão da skill, realizando eventuais finalizações necessárias.
- **IntentReflectorHandler:** Utilizado apenas durante a fase de desenvolvimento, esta intent repetia o nome da intent chamada para facilitar os testes e depuração do sistema.
- **CatchAllExceptionHandler:** Responsável por capturar exceções inesperadas durante a execução da skill, garantindo que o usuário receba uma resposta adequada em caso de falhas.

Todo o processo seguiu as diretrizes descritas na documentação oficial da Alexa Skills Kit (Amazon, 2024).

4.2. Desenvolvimento da Função Lambda

Para implementar a lógica necessária à execução das intents criadas na skill, foi desenvolvido um código Python utilizando o serviço AWS Lambda. O AWS Lambda é uma plataforma de computação serverless da Amazon Web Services que permite executar código em resposta a eventos, como o acionamento de intents na Alexa, sem necessidade de provisionar ou gerenciar servidores. Isso proporciona escalabilidade automática, alta disponibilidade e modelo de cobrança baseado apenas no tempo de execução efetivo do código, características essenciais para aplicações de assistentes virtuais, que podem ter picos de uso variados ao longo do dia (AWS, 2024a).

A função Lambda foi configurada para receber requisições diretamente da skill, processar as informações necessárias (como consultar a lista de robôs, criar missões de movimentação ou fala, consultar e cancelar missões) e responder de forma apropriada para o usuário. Esse processo é realizado por meio do `CustomSkillBuilder`, que organiza os handlers responsáveis por cada intent, garantindo que a comunicação siga o fluxo correto desde a recepção do evento até a resposta final.

Para armazenar informações persistentes entre diferentes sessões da Alexa, como o `mission_id` de uma missão criada, foi utilizado o `S3Adapter`, que permite integrar facilmente a função Lambda com o serviço Amazon S3. Isso é fundamental para garantir continuidade no diálogo com o usuário, por exemplo, quando ele deseja consultar ou cancelar uma missão que foi criada anteriormente.

Durante o desenvolvimento, para atualizar o código da função Lambda, foi seguido o seguinte procedimento:

- Empacotar o projeto em formato `.zip`, contendo o código `.py` e as bibliotecas necessárias, com a linha de comando (dentro do terminal no diretório do arquivo `.py`):
powershell -Command "Compress-Archive -Path * -DestinationPath ../alexa_lambda.zip -Force";
- Realizar o Upload do arquivo compactado dentro da plataforma AWS.

Esse fluxo de atualização permite manter a função sempre sincronizada com a última versão do código-fonte desenvolvido localmente.

A escolha da AWS Lambda como ambiente de execução está alinhada com as melhores práticas para o desenvolvimento de skills da Alexa, recomendadas no Alexa Skills Kit Developer Guide (Amazon, 2024).

4.3. Armazenamento com Bucket S3

O Amazon Simple Storage Service (Amazon S3) é um serviço de armazenamento de objetos oferecido pela Amazon Web Services (AWS), desenvolvido para fornecer escalabilidade, alta disponibilidade e segurança de dados na nuvem (Amazon Web Services, 2023). No presente projeto, o Amazon S3 foi utilizado como a solução para o armazenamento persistente de atributos da skill Alexa.

O uso do armazenamento persistente é importante para manter o estado da aplicação entre sessões, permitindo, por exemplo, que o usuário consulte o andamento de uma missão criada anteriormente. Para essa integração, foi utilizado o `S3Adapter` disponibilizado no Alexa Skills Kit SDK for Python, que facilita a conexão entre a função AWS Lambda da skill e o bucket S3 (Amazon, 2023).

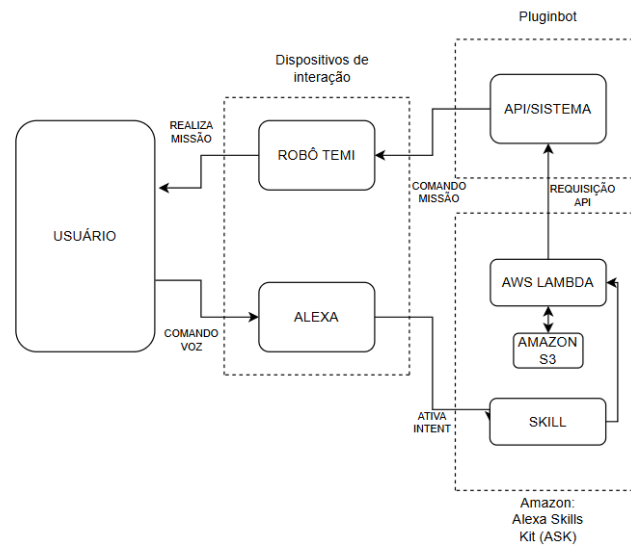
A escolha pelo S3 se deu devido à sua integração nativa com a AWS Lambda, sua alta durabilidade e sua capacidade de suportar aplicações sem a precisar de um servidor dedicado, além de não necessitar de gerenciamento manual de infraestrutura (Amazon Web Services, 2023). Esses fatores garantem que a skill mantenha seus dados de forma confiável, mesmo em cenários de grande volume de usuários ou interações simultâneas.

O bucket utilizado foi denominado "alexa-temi-armazenamento" e configurado para armazenar atributos como o `mission_id`. Sempre que uma nova missão é criada ou consultada, as informações relevantes são automaticamente gravadas ou lidas desse repositório, permitindo uma experiência contínua e personalizada ao usuário, sem a necessidade de reconectar-se a bancos de dados externos.

5. Análise do Sistema Implementado

A Figura 1 ilustra o fluxo completo de comunicação entre os principais componentes da arquitetura, desde a emissão do comando de voz até a execução da missão pelo robô.

Figura 1: Diagrama de Blocos Funcional



Fonte: O próprio Autor

5.1. Fluxo de Comunicação

O processo tem início com o usuário, que interage verbalmente com a Alexa. A assistente virtual interpreta o comando emitido e ativa a skill. Reconhecendo a intent correspondente, aciona uma função encontrada no código AWS Lambda, a qual executa a lógica necessária. Essa função pode realizar consultas ou atualizações no serviço Amazon S3, utilizado como armazenamento persistente de atributos da aplicação, como por exemplo o identificador de uma missão em andamento (*mission_id*). Com os dados processados, a função Lambda envia uma requisição HTTP à API do sistema PluginBot, responsável pelo controle remoto do robô. A API interpreta os parâmetros da requisição e envia ao robô Temi o comando de missão desejado, que pode envolver movimentação, fala ou outra funcionalidade integrada.

5.2. Interação com o Robô

Ao receber o comando, o robô Temi inicia a execução física da missão, interagindo diretamente com o ambiente e com o usuário. Essa etapa finaliza o ciclo iniciado pelo comando de voz, garantindo uma experiência fluida, natural e automatizada para o usuário.

5.3. Modularidade Arquitetural

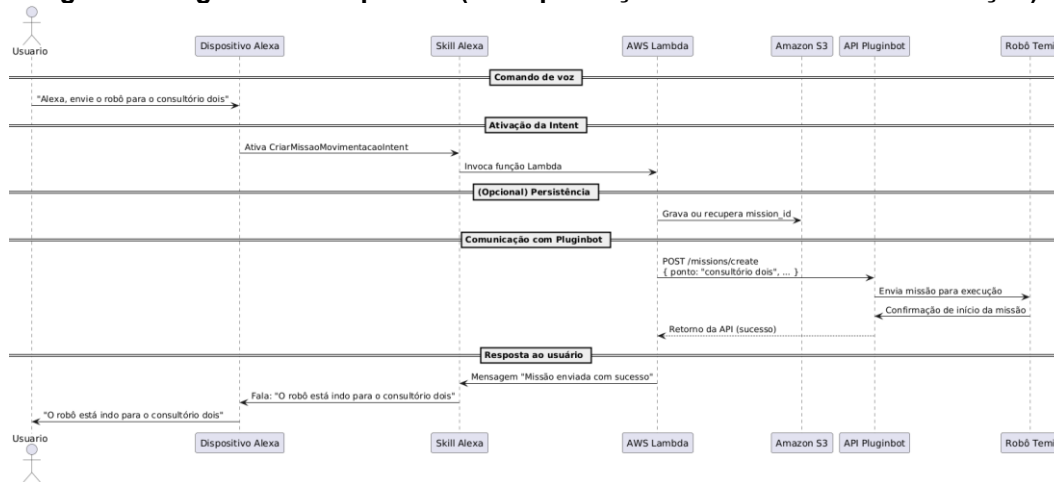
A arquitetura proposta pode ser dividida logicamente em três módulos principais:

- Dispositivos de interação: compostos pela Alexa e pelo robô Temi, formam a camada visível ao usuário;
- Infraestrutura em nuvem (Alexa Skills Kit + AWS Lambda + Amazon S3): executa o processamento da lógica e armazenamento de estado;
- Pluginbot: fornece a interface de controle com o robô, por meio de sua API RESTful.

Essa modularidade permite a atualização independente dos componentes, favorecendo manutenções, melhorias incrementais e reusabilidade da infraestrutura em diferentes projetos com robôs ou assistentes de voz.

6. Fluxo de Execução: Diagrama de Sequência

Figura 3. Diagrama de sequência (Exemplificação da Missão de Movimentação)



Fonte: O próprio Autor

6.1. Comando de Voz e Ativação da Intent

O processo se inicia quando o usuário pronuncia o comando “Alexa, envie o robô para o consultório dois”. O dispositivo Alexa capta o áudio, converte-o em texto e ativa a intent correspondente, previamente definida no modelo de interação da Skill: CriarMissaoMovimentacaoIntent. A ativação da intent resulta na invocação da função AWS Lambda, responsável por processar a lógica de negócio associada à movimentação solicitada.

6.2. Persistência Opcional

Antes de comunicar a missão à API da PluginBot, a função Lambda pode consultar ou atualizar o armazenamento persistente no Amazon S3, por meio do S3Adapter. Essa etapa é útil para registrar o identificador da missão (mission_id) ou garantir que não exista uma missão anterior ainda ativa.

6.3. Comunicação com a API da PluginBot

Em seguida, a função Lambda estrutura uma requisição HTTP do tipo POST para o endpoint /missions/create, contendo o ponto de destino informado (ex.: “consultório dois”) e demais parâmetros necessários à execução da missão. A API da PluginBot, ao receber essa requisição, envia os dados ao robô Temi, que inicia a execução da missão. O retorno da API, com status de sucesso, confirma que o comando foi recebido corretamente e que a missão está em andamento.

6.4. Resposta ao Usuário

Após o retorno da API, a função Lambda gera uma mensagem de feedback que é encaminhada pela Skill à Alexa. A assistente, então, responde verbalmente ao usuário com uma confirmação, por exemplo: “O robô está indo para o consultório dois”. Essa

resposta finaliza o ciclo da requisição, proporcionando uma experiência interativa, clara e orientada a voz.

7. Conclusões

Este trabalho apresentou o desenvolvimento de uma Skill Alexa para controle de missões de movimentação e comunicação, além de consultas em tempo real do robô Temi. A solução integrou comandos de voz com API externa, permitindo a criação, consulta e cancelamento de missões de forma simples e eficaz.

O projeto atingiu seus objetivos, entregando uma interação fluida entre usuário e robô, com armazenamento persistente e processamento sem uso de servidor, via AWS Lambda e Amazon S3. A arquitetura adotada demonstrou ser eficiente, segura e escalável.

Com isso, a integração entre assistentes de voz e robôs móveis se mostrou viável e promissora, abrindo caminho para futuras melhorias e aplicações em contextos profissionais e residenciais.

Referência

- AMAZON. What is Alexa?. 2024. Disponível em: <https://www.amazon.com/alexa>. Acesso em: 28 abr. 2025.
- AMAZON. Alexa Skills Kit Developer Guide. 2024. Disponível em: <https://developer.amazon.com/en-US/docs/alexa/ask-overviews/what-is-the-alexa-skills-kit.html>. Acesso em: 30 abr. 2025.
- AMAZON. Alexa Skills Kit SDK for Python - Persistence Adapter Classes. 2024. Disponível em: <https://developer.amazon.com/en-US/docs/alexa/alexa-skills-kit-sdk-for-python/persistence.html>. Acesso em: 28 abr. 2025.
- AMAZON DEVELOPER. Alexa Skills Kit Documentation. 2024. Disponível em: <https://developer.amazon.com/en-US/docs/alexa>. Acesso em: 28 abr. 2025.
- AMAZON DEVELOPER. The evolution of voice experiences. 2024. Disponível em: <https://developer.amazon.com/en-US/blogs/alexa>. Acesso em: 28 abr. 2025.
- AMAZON WEB SERVICES. Amazon Simple Storage Service (S3) - Documentation. 2025. Disponível em: <https://docs.aws.amazon.com/s3/index.html>. Acesso em: 28 abr. 2025.
- AMAZON WEB SERVICES. AWS Lambda Documentation. 2025. Disponível em: <https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>. Acesso em: 30 abr. 2025.
- HASSAN, S.; ZHANG, C.; SHOKOUHI, M.; LEVY, A. Understanding the Impact of Voice Assistant Errors on User Behavior. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM). 2019.
- PLUGINBOT. API Documentation - Robot Control and Mission Management. 2024. Disponível em: <https://pluginbot.ai/>. Acesso em: 28 abr. 2025.
- ROBOTEMI. Introducing temi, the personal robot. 2024. Disponível em: <https://www.robotemi.com/>. Acesso em: 28 abr. 2025.
- STATISTA. Number of digital voice assistants in use worldwide. 2024. Disponível em: <https://www.statista.com/statistics/973815/worldwide-digital-voice-assistant-in-use/>. Acesso em: 28 abr. 2025.