

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL GOIANO – CAMPUS IPORÁ
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

RELATÓRIO DE ESTÁGIO NÃO CURRICULAR

RELATO DE ESTÁGIO -
MACHINE LEARNING COM AWS

JOHNATHAN RAFAEL SANTOS BORBA

IPORÁ - GO

2024

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL GOIANO – CAMPUS IPORÁ
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

RELATO DE ESTÁGIO -
MACHINE LEARNING COM AWS

JOHNATHAN RAFAEL SANTOS BORBA

PROF. DR. THAMER HORBYLON NASCIMENTO
Orientador

Relatório de Estágio Não Curricular
apresentado ao Instituto Federal Goiano
– *Campus* Iporá, como requisito parcial
para conclusão do Curso Tecnologia em
Análise e Desenvolvimento de Sistemas.

IPORÁ - GO
Outubro/2024

S726r Borba, Johnathan Rafael Santos

Relatório de estágio: machine learning com AWS / Johnathan Rafael Santos Borba; orientador Dr. Thamer Horbylon Nascimento. – Iporá, 2024.

30 f.

Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) – Instituto Federal de Educação, Ciência e Tecnologia Goiano, campus Iporá.

1. Machine Learning. 2. AWS. I. Nascimento, Thamer Horbylon (orient.). II. IFGoiano III. Título.

CDU 004

Responsável: Ítala Moreira Alves (Bibliotecário-documentalista CRB-1 nº 2772)
Sistema Integrado de Bibliotecas – Instituto Federal Goiano



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR PRODUÇÕES TÉCNICO-CIENTÍFICAS NO REPOSITÓRIO INSTITUCIONAL DO IF GOIANO

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia Goiano, a disponibilizar gratuitamente o documento no Repositório Institucional do IF Goiano (RIIF Goiano), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IF Goiano.

Identificação da Produção Técnico-Científica (assinale com X)

☐ Tese

☐ Dissertação

☐ Monografia – Especialização

☐ Artigo - Especialização

☒ TCC - Graduação

☐ Artigo Científico

☐ Capítulo de Livro

☐ Livro

☐ Trabalho Apresentado em Evento

☐ Produção técnica. Qual: _____

Nome Completo do Autor: **Johnathan Rafael Santos Borba**

Matrícula: **2019105210430300**

Título do Trabalho: **RELATO DE ESTÁGIO - MACHINE LEARNING COM AWS**

Restrições de Acesso ao Documento [Preenchimento obrigatório]

Documento confidencial: ☒ Não ☐ Sim, justifique:

Informe a data que poderá ser disponibilizado no RIIF Goiano: **28/10/2024**.

O documento está sujeito a registro de patente? ☐ Sim ☒ Não

O documento pode vir a ser publicado como livro? ☐ Sim ☒ Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

1. O documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
2. Obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia Goiano os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
3. Cumpru quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia Goiano.

Iporá, 28 de outubro de 2024

Johnathan Rafael Santos Borba

Assinado eletronicamente pelo o Autor e/ou Detentor dos Direitos Autorais

Ciente e de acordo:

Thamer Horbylon Nascimento

Assinatura eletrônica do(a) orientador(a)

Documento assinado eletronicamente por:

- Thamer Horbylon Nascimento, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 28/10/2024 10:12:23.
- Johnathan Rafael Santos Borba, 2019105210430300 - Discente, em 28/10/2024 19:30:48.

Este documento foi emitido pelo SUAP em 28/10/2024. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifgoiano.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 646273

Código de Autenticação: ab55ebc38d



INSTITUTO FEDERAL GOIANO

Campus Iporá

Av. Oeste, Parque União, 350, Parque União, IPORA / GO, CEP 76.200-000

(64) 3674-0400



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

Ata nº 87/2024 - GE-IP/CMPIPR/IFGOIANO

ATA DA SESSÃO DE JULGAMENTO DO TRABALHO DE CURSO
DE JOHNATHAN RAFAEL SANTOS BORBA

Aos vinte e três dias do mês de outubro de dois mil e vinte e três, às vinte horas e dez minutos, no Laboratório de Informática III (sala 11 do bloco IV) do Instituto Federal Goiano – Câmpus Iporá, reuniu-se, em sessão pública, a banca examinadora designada na forma regimental pela Coordenação do Curso para julgar o trabalho de curso intitulado **“RELATO DE ESTÁGIO - MACHINE LEARNING COM AWS”**, apresentado pelo acadêmico **Johnathan Rafael Santos Borba** como parte dos requisitos necessários à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas. A banca examinadora foi presidida pelo orientador do trabalho de curso, Professor Doutor Thamer Horbylon Nascimento, tendo como membros a Professora Mestra Luciana Recart Cardoso e o Professor Mestre Wesley Flávio de Miranda. Aberta a sessão, o acadêmico expôs seu trabalho. Em seguida, foi arguido pelos membros da banca e:

(**X**) tendo demonstrado suficiência de conhecimento e capacidade de sistematização do tema de seu trabalho de curso, a banca conclui pela **aprovação** do acadêmico, sem restrições.

() tendo demonstrado suficiência de conhecimento e capacidade de sistematização do tema de seu trabalho de curso, a banca conclui pela **aprovação** do acadêmico, **condicionada a satisfazer as exigências** listadas na Folha de Modificação de Trabalho de Curso anexa à presente ata, no prazo máximo de 60 dias, a contar da presente data, ficando o professor orientador responsável por atestar o cumprimento dessas exigências.

() não tendo demonstrado suficiência de conhecimento e capacidade de sistematização do tema de seu trabalho de curso, a banca conclui pela **reprovação** do acadêmico.

Conforme avaliação individual de cada membro da banca, será atribuída a nota **9,0 (nove)** para fins de registro em histórico acadêmico.

Os trabalhos foram encerrados às vinte e uma horas e quatro minutos. Nos termos do Regulamento do Trabalho de Curso do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal Goiano – Câmpus Iporá, lavrou-se a presente ata que, lida e julgada conforme, segue assinada pelos membros da banca examinadora.

(Assinado Eletronicamente)
Prof. Dr. Thamer Horbylon Nascimento

(Assinado Eletronicamente)
Profª. Ma. Luciana Recart Cardoso

(Assinado Eletronicamente)
Prof. Me. Wesley Flávio de Miranda

Documento assinado eletronicamente por:

- Thamer Horbylon Nascimento, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 23/10/2024 21:06:13.
- Wesley Flavio de Miranda, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 24/10/2024 13:20:52.
- Luciana Recart Cardoso, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 28/10/2024 08:58:46.

Este documento foi emitido pelo SUAP em 23/10/2024. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifgoiano.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 645484

Código de Autenticação: f452032067



INSTITUTO FEDERAL GOIANO

Campus Iporá

Av. Oeste, Parque União, 350, Parque União, IPORA / GO, CEP 76.200-000

(64) 3674-0400

AGRADECIMENTOS

Aos devidos e merecidos agradecimentos, primeiramente a Deus, a Jesus Cristo e ao Arcanjo Miguel. Nos momentos em que estive perdido na depressão, foi na fé que me reencontrei. Toda conquista e cura que tenho hoje, devo à minha fé e à confiança em cada ser divino mencionado.

Também estendo meus sinceros agradecimentos à minha família: minha mãe, Izaulina Divina dos Santos, meu pai, Ivan de Souza Borba, e meu irmão, Danilo Ramom dos Santos Borba, pela força e coragem que me proporcionaram para chegar até aqui. Em grau igual de importância e reconhecimento, agradeço à minha amada companheira, Nayara M. Wachekowski, que esteve ao meu lado nos momentos mais difíceis, muitas vezes me apoiando durante as crises de ansiedade e me ajudando a manter o foco na realização e conclusão do curso e do estágio.

Da mesma forma, expresso minha gratidão aos meus professores e instrutores, que fizeram parte da minha jornada, tanto no curso quanto no estágio. Além do conhecimento transmitido, algumas vezes foram o ombro amigo de que muito precisei, com especial menção ao professor mestre Wesley Flávio Miranda.

Em especial, agradeço ao meu orientador, Dr. Thamer Horbylon Nascimento, que, além de se dedicar ao meu aprendizado, demonstrou paciência e empatia diante da minha realidade. Nada do que conquistei teria sido possível sem sua orientação e sua amizade.

'A vida é a arte do encontro, embora haja tanto desencontro pela vida. '

O Menestrel.

SUMÁRIO

1. INTRODUÇÃO	01
2. CARACTERIZAÇÃO DA EMPRESA E DADOS DO ESTÁGIO	03
3. ATIVIDADES DESENVOLVIDAS	04
4. CONSIDERAÇÕES FINAIS	20
5. REFERÊNCIAS BIBLIOGRÁFICAS	21

ANEXO (S)

ANEXO I - GLOSSÁRIO DE TERMOS TÉCNICOS

1 – INTRODUÇÃO

Este relato tem como propósito apresentar as experiências e atividades desenvolvidas durante o estágio realizado em uma empresa globalmente reconhecida por sua liderança em transformação digital, Inteligência Artificial (IA) e *Machine Learning* (ML). O estágio, com uma carga horária total de 400 horas, foi conduzido entre novembro de 2023 e abril de 2024, proporcionando uma imersão aprofundada no ecossistema de inovação tecnológica da empresa. De acordo com Russell e Norvig (2021), a IA desempenha um papel crucial na transformação digital ao automatizar processos e fornecer insights baseados em grandes volumes de dados, destacando-se como uma das principais tecnologias emergentes no setor de TI.

O programa de bolsas, com foco em *Machine Learning* utilizando AWS, visa capacitar estudantes e profissionais em início de carreira para atuar em áreas de alta demanda tecnológica. Estruturado em módulos teóricos e práticos, o programa oferece uma trilha de aprendizado abrangente, desde os fundamentos de programação e metodologias ágeis até a implementação de soluções avançadas de *machine learning* e inteligência artificial na nuvem. Goodfellow, Bengio e Courville (2016) destacam que o aprendizado profundo, uma subárea do *machine learning*, é responsável por muitos dos avanços recentes em IA, e plataformas de nuvem como a AWS são essenciais para o treinamento e a implementação desses modelos em escala.

A estrutura do estágio foi organizada em *Sprints*, nos quais desafios reais de desenvolvimento de software foram apresentados para serem solucionados de maneira colaborativa, simulando o ambiente de trabalho de alta performance. Cada *Sprint* trouxe novos desafios, e ao longo do estágio, atuei em projetos envolvendo front-end, back-end, integração de APIs e desenvolvimento de sistemas baseados em IA e ML, sempre sob a orientação de instrutores e guiado por meu orientador de estágio. Segundo Schwaber e Sutherland (2020), o uso do *Scrum* e de metodologias ágeis tem mostrado ser uma abordagem eficaz para a entrega contínua de valor em projetos de desenvolvimento de software, promovendo a adaptação e flexibilidade.

A experiência no estágio não se limitou ao aprimoramento de habilidades técnicas. A empresa promove uma cultura colaborativa e de aprendizado contínuo, o que me permitiu desenvolver também habilidades interpessoais essenciais, como trabalho em equipe,

comunicação eficaz e adaptação às rápidas demandas do setor de tecnologia. Senge (2006) argumenta que a aprendizagem organizacional é um componente essencial para o sucesso em longo prazo de qualquer organização, especialmente em um setor tão dinâmico como o da tecnologia. Este relato detalha cada *Sprint* realizada, as ferramentas utilizadas, os desafios enfrentados e as principais lições aprendidas ao longo dessa jornada, que contribuiu significativamente para meu crescimento profissional e pessoal.

2 – CARACTERIZAÇÃO DA EMPRESA E DADOS DO ESTÁGIO

2.1 – Caracterização da empresa

Em respeito às políticas de privacidade e integridade de informações, detalhes específicos sobre o nome, localização e histórico administrativo da empresa serão mantidos em sigilo. Assim, referimo-nos a ela apenas como “empresa” ao longo deste documento. Essa abordagem assegura o cumprimento de diretrizes de confidencialidade, valorizando a transparência e o respeito a informações sensíveis da organização.

A empresa oferece uma ampla gama de serviços tecnológicos e é reconhecida como referência em transformação digital, especialmente por sua expertise em Inteligência Artificial (IA) e *Machine Learning* (ML). Com esse foco, a empresa apoia outras organizações na implementação de soluções de ponta em IA, promovendo inovação e eficiência em diversos setores.

Em 2023, a empresa recebeu a certificação em IA Generativa da AWS, destacando-se como líder em soluções baseadas em IA. Ela integra também um grupo comprometido com a revolução digital por meio de tecnologias avançadas de IA. Além disso, a empresa é certificada como um *Great Place to Work*, reforçando seu compromisso com o bem-estar de sua equipe e a qualidade do ambiente de trabalho, essenciais para o desenvolvimento de soluções de alto nível e o atendimento aos mais elevados padrões de inovação e desempenho.

2.2 - Dados do estágio

Caracterizado como Programa de Bolsas *Machine Learning* com AWS, o estágio ocorreu entre o período de 13/11/2023 até 10/04/2024 e teve duração de 400 horas, dentre estas foram divididas em 4 horas de estudo diárias, fechando 20 horas semanais, onde tive a oportunidade de aprender e exercer conhecimentos variados voltados à IA e serviços da Amazon Web Services (AWS) remotamente. As orientações aconteceram pelo lado do Instituto Federal Goiano Campus Iporá GO pelo Professor Dr. Thamer Horbylon Nascimento e supervisão no ambiente laboral um supervisor foi designado para acompanhamento e orientação do estagiário.

3 – ATIVIDADES DESENVOLVIDAS

Durante o período de estágio, passamos por uma trilha de aprendizado intensiva, com testes técnicos, visando nos preparar para as melhores práticas em programação, versionamento, integração contínua e desenvolvimento ágil, todas aplicadas ao uso de técnicas e ferramentas de Inteligência Artificial (IA) e *Machine Learning* (ML). O foco principal foi o ecossistema de computação em nuvem da AWS, onde aprendemos a utilizar e integrar diferentes abordagens tecnológicas baseadas em IA. Conforme mencionado por Amazon Web Services (2023), a utilização de serviços de nuvem facilita a escalabilidade e a aplicação de soluções de IA, permitindo que as empresas implementem modelos de ML de forma eficiente e segura.

O estágio foi estruturado em *Sprints*, ciclos de desenvolvimento de duas semanas, nos quais enfrentávamos desafios técnicos propostos pelos instrutores. Cada *Sprint* incluía uma trilha de aprendizado detalhada, fornecendo os conhecimentos necessários para superar esses desafios. Trabalhamos em equipes de três a quatro estagiários, o que incentivou a colaboração e a troca de experiências durante o desenvolvimento das soluções. Gonçalves (2021) salienta que o trabalho em equipe é fundamental em ambientes ágeis, pois promove a comunicação efetiva e a resolução rápida de problemas complexos.

As trilhas de aprendizado abrangeram áreas fundamentais da tecnologia, incluindo linguagens de programação como JavaScript e Python, bem como conceitos e técnicas de desenvolvimento ágil, como o método *Kanban*. Exploramos também tópicos cruciais como Infraestrutura como Código (IaC), que nos permitiu automatizar a gestão da infraestrutura de TI, além dos serviços e ferramentas da AWS voltados para IA e ML. Segundo Hüttermann (2012), a IaC melhora a eficiência operacional ao permitir que o código gerencie a infraestrutura de forma automatizada, reduzindo a intervenção manual e os erros associados.

Recebemos também treinamento sobre Programação Orientada a Objetos (POO), práticas de Segurança da Informação e Containerização, tecnologias que facilitam o empacotamento e execução de aplicações de maneira mais eficiente e segura. Esses conhecimentos foram essenciais para moldar o perfil profissional que o programa de bolsas esperava dos participantes.

Ao final de cada *Sprint*, os times apresentavam as soluções desenvolvidas, proporcionando momentos de reflexão sobre os desafios enfrentados e as lições aprendidas ao

longo do processo. Argyris (1999) defende que a reflexão crítica sobre os erros cometidos é fundamental para a aprendizagem organizacional e o aprimoramento contínuo.

A seguir, serão detalhadas cada *Sprint*, destacando as experiências, desafios e aprendizados obtidos em cada uma das etapas.

3.1 *Sprint* 1 - Jogo de Adivinhação de PIN

Experiência

Na primeira *Sprint*, fomos desafiados a desenvolver um jogo de adivinhação de PIN usando as tecnologias básicas de front-end: JavaScript, HTML e CSS. Na Figura 1, temos uma representação do resultado final, também é possível observar que o jogo consistia em o usuário tentar adivinhar um PIN gerado aleatoriamente, com um número limitado de tentativas e dicas fornecidas a cada erro. Apesar de ser inspirado em um desafio de curso, o projeto foi expandido para avaliação, com a inclusão de boas práticas de codificação e interfaces mais interativas e amigáveis para o usuário. Segundo Flanagan (2020), o JavaScript é uma das linguagens de programação mais versáteis para desenvolvimento web, especialmente em projetos interativos como jogos, devido à sua capacidade de manipulação dinâmica do modelo de objeto de documento.



Figura 1 - Interface jogo Adivinhação PIN realizado na *Sprint* 1.

Desafios

O maior desafio foi garantir que o número gerado fosse sempre composto de quatro dígitos, o que nos levou a desenvolver funções de validação específicas e a garantir que a geração de números aleatórios seguisse essa especificação. Segundo Crockford (2008), a criação de funções específicas para a validação de dados em JavaScript é fundamental para garantir a integridade e segurança de aplicações interativas. Outro obstáculo foi restringir a entrada do usuário a exatamente quatro dígitos, o que exigiu manipulações do modelo de objeto de documento para prevenir entradas incorretas. Além disso, tivemos dificuldades no controle de versão com o Git, especialmente na nomeação de *commits*. Ao usar o comando *git push --force-with-lease*, aprendemos a resolver conflitos no versionamento de maneira eficiente, um aspecto crucial em equipes de desenvolvimento. Chacon e Straub (2014) destacam que o Git é uma ferramenta indispensável para o controle de versão, principalmente em ambientes de desenvolvimento colaborativo, onde a resolução de conflitos é uma habilidade crítica.

Aprendizado

Esse projeto reforçou minha capacidade de lidar com a validação de dados e a restrição de entradas, habilidades essenciais para a criação de sistemas confiáveis e seguros. Além disso, aprendi sobre a importância de um controle de versão bem organizado, algo indispensável para o trabalho em equipe. A experiência também me proporcionou uma compreensão mais profunda de como pequenos projetos podem ser estruturados para entregar uma experiência de usuário positiva. Norman (2013) ressalta que o design centrado no usuário é um princípio crucial para o desenvolvimento de interfaces que proporcionem interatividade e eficiência ao usuário final.

3.2 Sprint 2 - Spofyteam: Descubra os Hits Mais Tocados

Experiência

Na *Sprint 2*, o desafio era escolher uma API e integrá-la ao nosso sistema. Escolhemos a API do Spotify e criamos o Spofyteam, um sistema que exibia as músicas e artistas mais populares em quatro gêneros musicais. Demonstrada na Figura 2, A interface permitia que o usuário visualizasse as listas de músicas e artistas mais populares e ouvisse as faixas diretamente pelo sistema, sem precisar acessar a plataforma do Spotify. Segundo Fielding (2000), a arquitetura orientada a serviços, especialmente em APIs REST, facilita a comunicação entre sistemas, o que foi essencial para a integração da API do Spotify com o nosso projeto.



Figura 2 – Interface SpofyTeam projeto realizado na *Sprint 2*.

Desafios

A integração com a API do Spotify apresentou diversos desafios. Foi necessário configurar corretamente as rotas e a autenticação, garantindo que as requisições fossem feitas de maneira contínua e eficiente. Conforme Olsson e Engdahl (2021), a autenticação OAuth é amplamente utilizada para proteger dados sensíveis em APIs, e dominar essa técnica foi essencial para garantir que as credenciais e os dados de acesso fossem tratados de forma segura. Otimizar as requisições para garantir um tempo de resposta rápido foi outra tarefa difícil, pois precisávamos assegurar que o usuário tivesse uma experiência agradável ao interagir com o sistema.

Além disso, o tratamento de dados dinâmicos, como a organização das músicas e artistas retornados pela API, exigiu atenção à estruturação e manipulação dos dados para que a interface fosse visualmente atrativa e fácil de usar. Hoffman et al. (2017) afirmam que a manipulação eficiente de grandes volumes de dados em tempo real é uma competência chave para o sucesso de sistemas baseados em APIs dinâmicas. Também tivemos que lidar com questões de segurança, especialmente no tratamento de dados sensíveis, como as chaves de acesso usadas na integração da API.

Aprendizado

Essa *Sprint* ampliou significativamente minha compreensão sobre a integração de APIs externas, especialmente em relação ao gerenciamento de dados dinâmicos e à otimização de performance para melhorar a experiência do usuário. Bosworth, Celko e Wong (2019) apontam que a eficiência no gerenciamento de dados é essencial para a construção de sistemas de alto desempenho, algo que ficou evidente ao desenvolver este projeto. Aprendi a utilizar o Express e o Axios para manipular requisições e respostas de forma eficiente, além de desenvolver uma interface que fosse intuitiva e fácil de usar, de acordo com os princípios de boas práticas de UX.

3.3 *Sprint* 3 - ClimaTempo: Visualização do Clima em Tempo Real

Experiência

Nesta *Sprint*, fomos desafiados a integrar e desenvolver uma API para visualizar o clima em tempo real. Criamos o ClimaTempo, uma aplicação que se integrava à API WeatherStack para fornecer informações atualizadas sobre o clima, como temperatura e condições meteorológicas. Como demonstrado na Figura 3, o usuário podia inserir o nome de uma cidade e, em tempo real, obter os dados climáticos da região. De acordo com Hansen et al. (2020), a integração de APIs meteorológicas é uma prática comum para proporcionar previsões em tempo real, melhorando a precisão das informações meteorológicas acessíveis ao público.

Detalhes do Clima para Diorama, Goiás, Brazil	
Condição atual: Partly cloudy 	Velocidade do vento: 10 km/h
Temperatura: 23°C	Direção do vento: NNW
Sensação térmica: 25°C	Pressão atmosférica: 1012 mb
Umidade: 79%	Índice UV: 6
	Visibilidade: 10 km
	Horário local: 2023-12-23 18:41

© Equipe-1. Todos os direitos reservados.

Figura 3 – Interface do projeto Clima Tempo realizado na *Sprint 3*.

Desafios

Um dos maiores desafios foi configurar o Docker para garantir que o projeto pudesse ser executado tanto no ambiente de desenvolvimento quanto no de produção na AWS. Merkel (2014) aponta que o Docker facilita o processo de containerização, garantindo que o software funcione de forma consistente em diferentes ambientes. Além disso, lidar com erros de entrada do usuário, como nomes de cidades inválidos ou não encontrados na API, foi um problema recorrente.

Para resolver isso, implementamos validações robustas e mensagens de feedback claras, utilizando técnicas como Regex (expressões regulares) para verificar e corrigir as entradas do usuário. Friedl (2006) destaca que o uso de expressões regulares é uma técnica eficaz para lidar com a validação e manipulação de dados textuais de forma precisa.

Aprendizado

Essa *Sprint* foi crucial para o meu aprendizado sobre gestão de ambientes em nuvem e containerização, utilizando Docker e AWS para garantir a escalabilidade do sistema. Segundo Amies, Sadalage e Guo (2018), a utilização de plataformas de nuvem como a AWS aumenta significativamente a flexibilidade e a escalabilidade dos sistemas modernos. Também desenvolvi habilidades na validação de dados e no tratamento de erros, garantindo uma experiência de usuário mais fluida e confiável, conforme recomendado por Norman (2013) em relação à importância de um design que prioriza a usabilidade e a experiência do usuário.

3.4 Sprint 4 - Jokes and Activities

Experiência

O projeto "Jokes and Activities" foi uma aplicação leve e divertida, que consumia APIs públicas para exibir piadas do Chuck Norris e sugestões de atividades aleatórias. O sistema foi desenvolvido com uma interface simples, permitindo ao usuário alternar facilmente entre as funcionalidades de piadas e atividades, exemplo ilustrado na Figura 4. Segundo Fielding (2000), as APIs REST permitem uma comunicação eficiente entre sistemas distribuídos, facilitando o consumo de dados de fontes externas em aplicações leves.

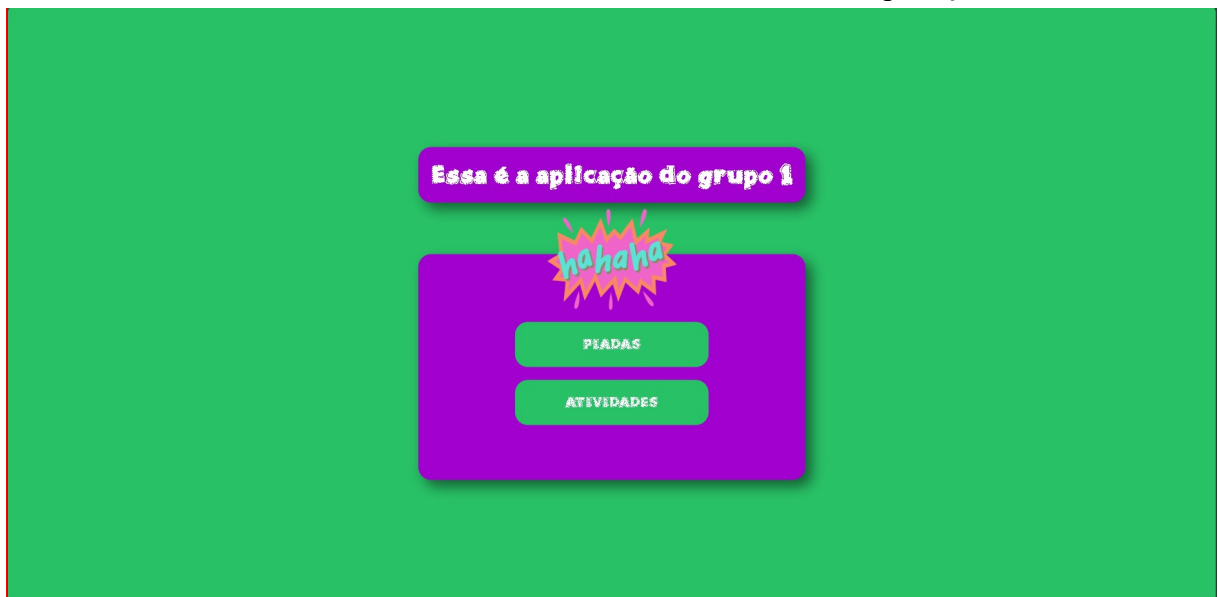


Figura 4 – Interface do projeto Joke and Activities projeto realizado na *Sprint 4*.

Desafios

O principal desafio foi formatar os dados retornados pelas APIs de forma que fossem exibidos de maneira clara e agradável para o usuário. Conforme Bosworth et al. (2019), a formatação adequada dos dados de APIs é essencial para garantir que o conteúdo seja compreendido e utilizado de maneira eficaz pelo sistema e pelos usuários. Outro desafio foi garantir que o backend tratasse as requisições corretamente, lidando com eventuais falhas de comunicação com as APIs sem comprometer a experiência do usuário. Olsson e Engdahl (2021) destacam a importância de prever e tratar falhas no consumo de APIs, uma vez que serviços externos nem sempre garantem a estabilidade contínua.

Aprendizado

Esse projeto me ensinou a importância de formatar corretamente os dados retornados por APIs e a necessidade de garantir que o backend esteja preparado para lidar com falhas nos serviços externos. Também aprendi a dar atenção a detalhes na interface do usuário, mesmo em sistemas simples, visando sempre melhorar a experiência de quem utiliza a aplicação. Norman (2013) ressalta que o design centrado no usuário é um fator-chave, independentemente da complexidade da aplicação, para garantir uma experiência agradável e eficiente.

3.5 Sprint 5 - ReservaRanger: Machine Learning com Amazon SageMaker

Experiência

Nesta *Sprint*, trabalhamos com o Amazon SageMaker para treinar um modelo de aprendizado de máquina utilizando o algoritmo Random Forest. O objetivo do projeto era desenvolver um modelo capaz de classificar e prever dados com base em um conjunto pré-definido. Os resultados do treinamento foram armazenados no Amazon S3 e acessados por meio de uma API, na Figura 5 podemos visualizar como a arquitetura do projeto se relaciona entre as ferramentas, após o processo de containerização.

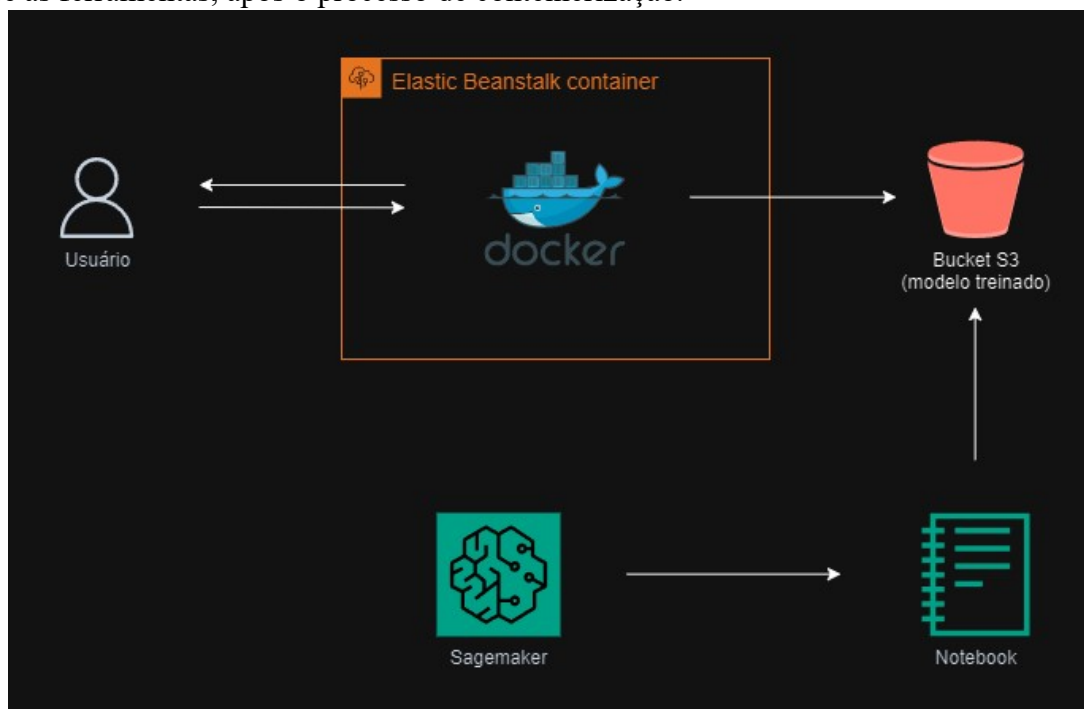


Figura 5 – Diagramação AWS da estrutura do projeto realizado na *Sprint* 5.

Hastie, Tibshirani e Friedman (2009) ressaltam que o Random Forest é amplamente utilizado em tarefas de classificação e predição por sua robustez e capacidade de lidar com grandes volumes de dados, o que o torna ideal para projetos desse tipo.

Desafios

A configuração do ambiente no Amazon SageMaker foi desafiadora, principalmente na integração com o Elastic Beanstalk para garantir que o treinamento do modelo fosse realizado de maneira eficiente. Zaharia et al. (2020) destacam que a integração de plataformas como o SageMaker e o Elastic Beanstalk permite a escalabilidade e a otimização do treinamento de modelos em ambiente de nuvem. Outro desafio foi garantir que os resultados fossem armazenados corretamente no Amazon S3 e pudessem ser acessados e manipulados por meio da API. Amies et al. (2018) apontam que a utilização do Amazon S3 para armazenamento de dados em projetos de aprendizado de máquina garante a durabilidade e a acessibilidade dos resultados.

Aprendizado

Essa *Sprint* proporcionou um aprendizado prático sobre ferramentas de aprendizado de máquina e infraestrutura da AWS. Aprendi a importância de estruturar a arquitetura de soluções escaláveis, capazes de lidar com grandes volumes de dados e que facilitam o desenvolvimento de projetos mais complexos. Goodfellow, Bengio e Courville (2016) mencionam que a escalabilidade e a eficiência são pilares fundamentais no desenvolvimento de sistemas de *machine learning*, especialmente em ambientes de nuvem como a AWS.

3.6 Sprint 6 - Conversão de Texto para Áudio com AWS Polly e Serverless Framework

Experiência

Nesta *Sprint*, desenvolvi um sistema baseado em arquitetura serverless que converte texto em áudio utilizando a API do AWS Polly. O objetivo era criar uma solução escalável, que pudesse ser utilizada de maneira prática e eficiente para transformar qualquer entrada de texto em arquivos de áudio. Para isso, utilizei o Serverless Framework, que simplificou a implantação do sistema e garantiu sua capacidade de escalar automaticamente conforme a

demanda. Além disso, o DynamoDB foi empregado para armazenar o histórico de conversões realizadas, permitindo que o sistema mantivesse um registro organizado e acessível.

Villamizar et al. (2016) destacam que arquiteturas serverless são ideais para soluções que exigem escalabilidade e baixo custo, uma vez que eliminam a necessidade de gerenciamento manual de servidores. A seguir na Figura 6, é possível visualizar o resultado da interface final do projeto.



Figura 6 – Interface final do projeto AWS Polly realizado na *Sprint 6*.

Desafios

O maior desafio foi garantir que o deploy do sistema ocorresse de forma suave e sem interrupções. Gerenciar as credenciais da AWS de maneira segura foi uma prioridade, especialmente ao integrar serviços críticos como o Polly. Anderson (2018) enfatiza que a segurança na gestão de credenciais é essencial ao utilizar serviços de nuvem, uma vez que o mau gerenciamento pode comprometer a integridade de toda a solução.

Além disso, tivemos que lidar com questões relacionadas ao CORS (Cross-Origin Resource Sharing), ajustando as configurações para garantir que as requisições entre o frontend e a API fossem realizadas de maneira segura e sem comprometer a funcionalidade do sistema. O uso de uma arquitetura serverless também exigiu atenção especial para garantir que o sistema fosse rápido e eficiente. Roberts (2019) destaca que a eficiência e o desempenho são fundamentais em sistemas serverless, especialmente quando há múltiplas integrações de serviços em nuvem.

Aprendizado

Essa *Sprint* foi extremamente valiosa para o meu aprendizado sobre arquiteturas serverless, especialmente no que diz respeito à criação de soluções escaláveis sem a necessidade de gerenciar servidores. Aprendi sobre a importância de gerenciar credenciais de maneira segura ao utilizar serviços de nuvem, e como configurar corretamente o CORS para permitir a comunicação entre diferentes partes do sistema.

A experiência com AWS Polly também expandiu meus conhecimentos sobre APIs voltadas para a conversão de texto em áudio e o impacto que essas soluções podem ter na criação de sistemas mais inclusivos e acessíveis. Turnbull et al. (2016) afirmam que a conversão de texto em áudio é uma tecnologia crucial para tornar as interfaces mais acessíveis, especialmente para usuários com deficiências visuais ou outras necessidades de acessibilidade.

3.7 *Sprint 7* - DengueConsult: Chatbot de Triagem Remota de Dengue

Experiência

Na *Sprint 7*, o desafio foi criar um chatbot para realizar a triagem remota de sintomas de dengue. Desenvolvi o DengueConsult, uma solução integrada com o Amazon Lex, que oferecia uma interface conversacional para os usuários, é possível ser observada sua estrutura através do diagrama AWS demonstrado na Figura 7. A ideia central era permitir que qualquer pessoa pudesse descrever seus sintomas e, com base nessas informações, o chatbot fornecesse orientações sobre a probabilidade de estar infectado com dengue, além de sugerir os próximos passos que deveriam ser tomados, como procurar uma unidade de saúde. Segundo Jurafsky e Martin (2021), os chatbots que utilizam processamento de linguagem natural são ferramentas valiosas para a automatização de tarefas e interação com os usuários de forma mais eficiente.

O sistema foi integrado ao Slack, proporcionando uma experiência de uso prática e direta para o usuário, utilizando a popularidade e acessibilidade dessa plataforma de comunicação. Conforme Zhong (2020), a integração de chatbots em plataformas amplamente utilizadas, como o Slack, melhora a acessibilidade e a interação, permitindo que as informações e serviços cheguem diretamente aos usuários de forma simples e rápida.

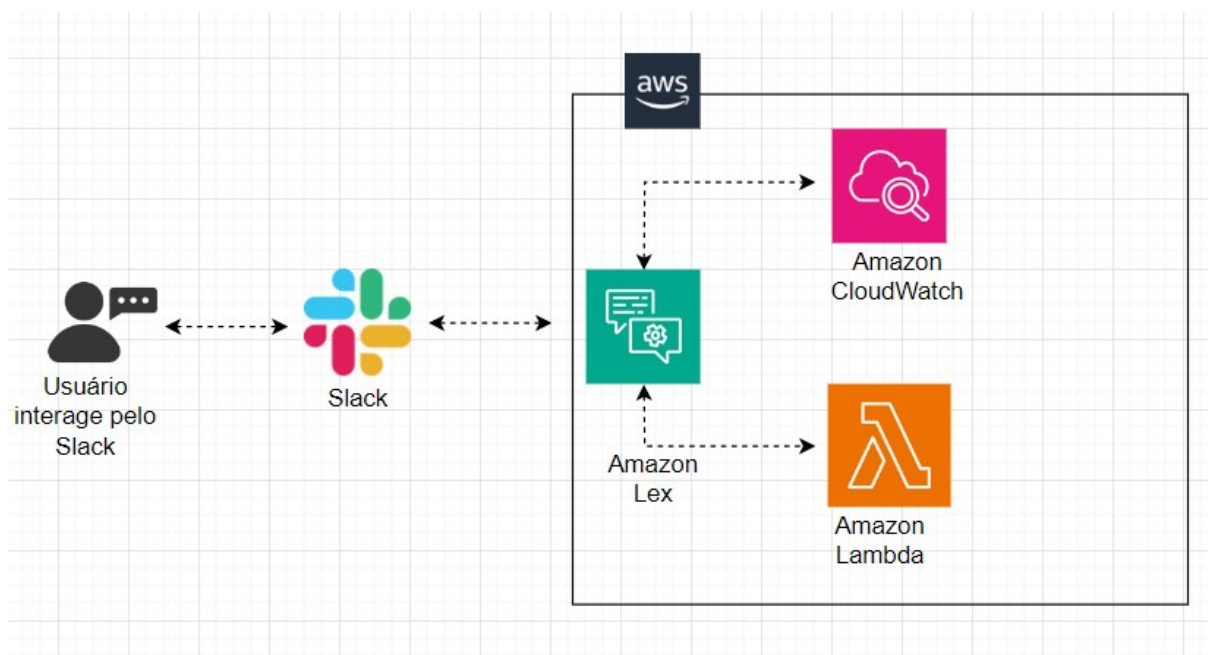


Figura 7 – Diagrama AWS da estrutura do chatbot Dengue Consult realizado na *Sprint 7*.

Desafios

O principal desafio foi configurar corretamente o Amazon Lex para entender e responder às diferentes interações dos usuários. Foi necessário criar intenções (intents) e slots (campos de resposta) que capturassem com precisão as informações fornecidas pelos usuários sobre seus sintomas. Wang e Zheng (2019) enfatizam que o uso de intenções e slots é uma prática fundamental para garantir a precisão na captação de dados em sistemas de conversação baseados em IA.

Além disso, tivemos que garantir que o fluxo de conversação fosse natural e intuitivo, principalmente ao lidar com um tema sensível como saúde pública. Também foi importante garantir que o sistema fosse seguro e cumprisse os requisitos de privacidade de dados, já que lidávamos com informações de saúde dos usuários, conforme discutido por Kumar e Srinivas (2017), que destacam a importância da segurança e privacidade em soluções de saúde digital.

Aprendizado

Essa *Sprint* foi fundamental para expandir minha compreensão sobre o desenvolvimento de chatbots e interfaces conversacionais. Aprendi a criar fluxos de diálogo claros e eficientes, adaptando as respostas do chatbot de acordo com as necessidades dos

usuários. O trabalho com o Amazon Lex também me permitiu adquirir mais experiência com serviços de Inteligência Artificial voltados para a compreensão de linguagem natural, o que foi crucial para criar uma experiência de usuário envolvente. Russell e Norvig (2021) afirmam que a compreensão de linguagem natural é uma área-chave no desenvolvimento de chatbots avançados, permitindo que esses sistemas interpretem e respondam às necessidades dos usuários de maneira precisa. A integração com o Slack mostrou como chatbots podem ser implementados em plataformas populares, facilitando o acesso a informações e serviços essenciais.

3.8 *Sprint 8* - AWS Rekognition Tools - Análise e Extração de Tags de Imagens

Experiência

Durante esta *Sprint*, tive a oportunidade de trabalhar com o AWS Rekognition, uma ferramenta poderosa para visão computacional. A criação da API foi um desafio técnico interessante, pois precisávamos implementar a detecção de rótulos e análise facial de imagens armazenadas no bucket S3. O uso do framework Serverless facilitou o deploy, e a organização da equipe com o método *Kanban* permitiu uma distribuição clara das tarefas, garantindo um fluxo de trabalho eficiente, como resultado podemos visualizar a interface final na Figura 8.

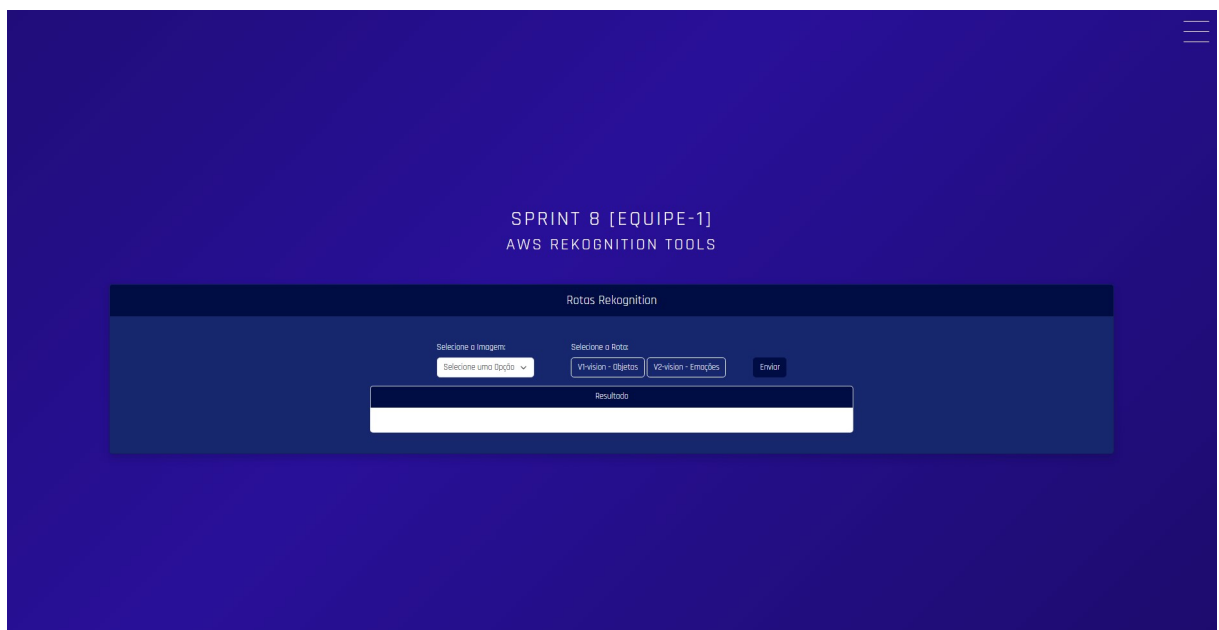


Figura 8 – Interface projeto AWS Rekognition Tools , realizado na *Sprint 8*.

Goodfellow, Bengio e Courville (2016) destacam que ferramentas de IA, como as usadas no AWS Rekognition, desempenham um papel crucial no processamento automatizado de imagens, permitindo soluções escaláveis e eficientes. Além disso, foi muito enriquecedor colaborar diretamente com meus colegas, especialmente no que diz respeito à integração das funcionalidades de IA e ao desenvolvimento de uma solução prática.

Desafios

Os principais desafios enfrentados nesta *Sprint* incluíram a configuração das permissões CORS no arquivo Serverless, que inicialmente bloqueavam o acesso à API por parte dos clientes. Roberts (2019) enfatiza que o gerenciamento adequado de permissões CORS é essencial em arquiteturas serverless para permitir a comunicação segura e eficaz entre os serviços de nuvem e as APIs públicas. Também tivemos que desenvolver uma lógica de comunicação eficiente entre os serviços da AWS e o framework Serverless, o que exigiu um esforço extra para garantir que as respostas fossem precisas e rápidas.

Além disso, a necessidade de alinhar horários entre os membros da equipe foi um obstáculo que demandou organização e flexibilidade para assegurar o progresso contínuo do projeto. Segundo Sommerville (2011), o uso de metodologias ágeis, como o *Kanban*, é fundamental para garantir que o desenvolvimento seja adaptável e flexível frente aos desafios de projetos colaborativos.

Aprendizado

Essa *Sprint* proporcionou um grande aprendizado em integração de serviços em nuvem e uso de ferramentas avançadas da AWS. Aprendi a importância de lidar com configurações de permissões adequadas (CORS), algo crucial em projetos que envolvem APIs públicas. Também adquiri um melhor entendimento sobre como otimizar código para evitar redundâncias e garantir a escalabilidade do projeto.

Amies, Sadalage e Guo (2018) mencionam que a otimização de código e a escalabilidade são elementos-chave para o sucesso de projetos em nuvem. A experiência de trabalhar em equipe, utilizando metodologias ágeis, foi fundamental para aprimorar minhas habilidades de comunicação e colaboração, o que será valioso em projetos futuros.

3.9 Sprints 9 e 10 - Vision Assistant: Chatbot para Pessoas com Deficiência Visual

Experiência

Nas últimas duas *Sprints* do programa, trabalhei no desenvolvimento do Vision Assistant, um chatbot alimentado por Inteligência Artificial (I.A.) voltado para auxiliar pessoas com deficiência visual. O projeto envolveu a utilização de múltiplos serviços da AWS para criar uma solução acessível, capaz de identificar objetos em imagens enviadas pelos usuários e descrevê-los por meio de áudio.

Para isso, utilizamos o AWS Rekognition para a análise das imagens, o AWS Polly para converter a descrição do objeto em áudio, e o Amazon S3 para armazenar as imagens enviadas, podemos observar melhor sua estrutura conforme Figura 9. Russell e Norvig (2021) apontam que a aplicação de I.A. em soluções assistivas, como o Vision Assistant, pode aumentar significativamente a autonomia e a inclusão de pessoas com deficiência.

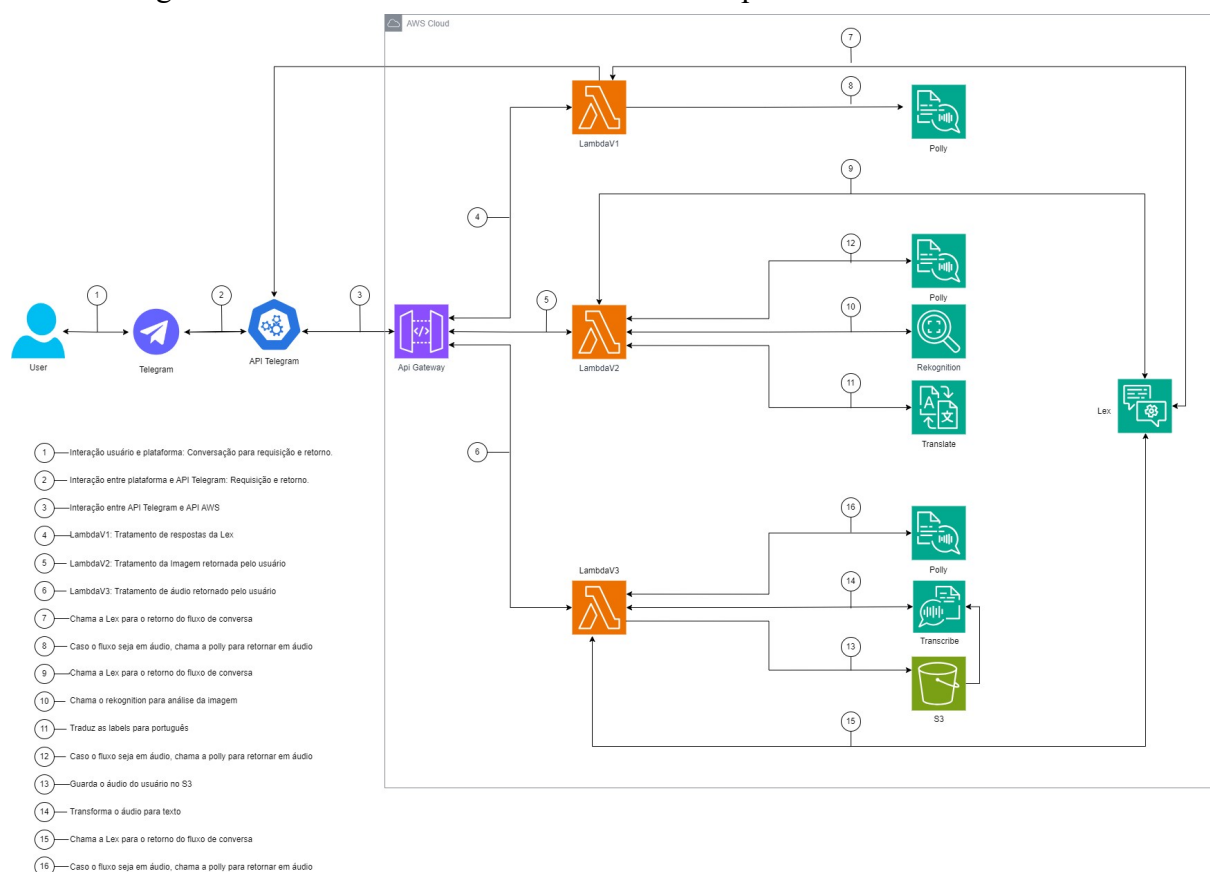


Figura 9 – Diagramação AWS do projeto Vision Assistant realizado na *Sprint* 9.

A funcionalidade principal do Vision Assistant era permitir que um usuário com deficiência visual pudesse enviar uma imagem e, em poucos segundos, receber uma descrição

detalhada do que estava na foto, tudo por meio de áudio, garantindo uma maior autonomia para essas pessoas em suas atividades diárias. Turnbull et al. (2016) destacam que o uso de ferramentas de conversão de texto em áudio é uma tecnologia crucial para tornar as interfaces acessíveis a pessoas com deficiência visual.

Desafios

O maior desafio técnico foi garantir a integração eficiente de diversos serviços da AWS, como Rekognition, Polly, S3 e Transcribe. Cada serviço desempenhava um papel crucial no funcionamento do sistema, e qualquer falha de comunicação entre eles poderia comprometer a experiência do usuário. Segundo Amies, Sadalage e Guo (2018), a integração de serviços em nuvem exige uma coordenação precisa para garantir que todos os componentes funcionem em harmonia, especialmente em sistemas que demandam respostas em tempo real. Além disso, era fundamental que a resposta do sistema fosse em tempo real, garantindo que o usuário recebesse a descrição da imagem de maneira rápida e precisa.

Outro desafio significativo foi garantir que a aplicação fosse verdadeiramente acessível, o que exigiu cuidados especiais no design da interface e na resposta dos serviços. Tivemos que assegurar que o sistema não apenas cumprisse seu propósito técnico, mas também fosse fácil de usar por pessoas com deficiência visual, oferecendo uma experiência inclusiva e eficaz. Norman (2013) enfatiza que o design centrado no usuário é essencial, especialmente em soluções que envolvem a acessibilidade para pessoas com deficiência.

Aprendizado

Este projeto consolidou meu entendimento sobre a integração de múltiplos serviços AWS, além de destacar a importância de criar soluções tecnológicas voltadas para a inclusão social. Aprendi sobre como os serviços de Inteligência Artificial da AWS podem ser combinados para resolver problemas reais e fornecer acessibilidade a pessoas com necessidades especiais. O Vision Assistant reforçou a importância de projetar soluções que tenham um impacto positivo na vida das pessoas, demonstrando como a tecnologia pode ser utilizada para transformar o cotidiano de indivíduos com deficiência visual.

4 – CONSIDERAÇÕES FINAIS

A experiência de estágio foi um marco importante no meu desenvolvimento profissional e técnico. Ao longo das *Sprints*, consegui consolidar conhecimentos essenciais em áreas como desenvolvimento de software, integração contínua, computação em nuvem e inteligência artificial, sempre aplicado em um ambiente colaborativo e ágil. Cada desafio enfrentado foi uma oportunidade de crescimento, tanto em termos técnicos quanto em habilidades interpessoais, como trabalho em equipe e resolução de problemas.

O estágio também me permitiu entender mais profundamente as necessidades do mercado de tecnologia atual, especialmente no que diz respeito à criação de soluções escaláveis e eficientes para empresas de grande porte. Trabalhar com o ecossistema AWS, me deixou claro como as tecnologias de IA podem ser aplicadas para desenvolver soluções inovadoras e acessíveis, com impacto direto no cotidiano das pessoas. Projetos como o Vision Assistant, que utilizou IA para auxiliar pessoas com deficiência visual, reforçaram minha visão de que a tecnologia tem um papel transformador e inclusivo.

Como fruto dessa experiência, obtive minha certificação internacional AWS CLF-C02, que valida meu conhecimento em computação em nuvem, e como feliz resultado de meus esforços, fui contratado pela empresa como Desenvolvedor AI/ML, o que marca o início de uma nova etapa na minha carreira, com a oportunidade de continuar aplicando e expandindo o que aprendi durante o estágio.

Essa combinação de experiência prática com IA e a certificação AWS abriram portas importantes na minha carreira, consolidando meu papel como desenvolvedor de soluções tecnológicas inovadoras e escaláveis, utilizando Inteligência Artificial para gerar impacto positivo e socialmente relevante.

5 – REFERÊNCIAS

- AMAZON. web services (AWS). *Amazon Polly Documentation*. Disponível em: <https://docs.aws.amazon.com/polly/>. Acesso em: 15 de dezembro de 2023.
- AMAZON. web services (AWS). *Amazon Rekognition Documentation*. Disponível em: <https://docs.aws.amazon.com/rekognition/>. Acesso em: 20 de dezembro de 2023.
- AMAZON. web services (AWS). *Amazon S3 Documentation*. Disponível em: <https://docs.aws.amazon.com/s3/>. Acesso em: 10 de janeiro de 2024.
- AMAZON. web services (AWS). *Amazon SageMaker Documentation*. Disponível em: <https://docs.aws.amazon.com/sagemaker/>. Acesso em: 10 de dezembro de 2023.
- AMAZON. web services (AWS). *AWS Certified Cloud Practitioner – Preparation Guide*. Disponível em: <https://aws.amazon.com/certification/certified-cloud-practitioner/>. Acesso em: 05 de dezembro de 2023.
- AMAZON. web services (AWS). *Machine Learning on AWS: The Essentials*. Amazon, 2023. Disponível em: <https://aws.amazon.com/machine-learning>. Acesso em: 10 out. 2024.
- ANDERSON, Paul. *Security in the Cloud: Managing Credentials and API Access in AWS*. 2. ed. San Francisco: Pearson, 2018. 200p.
- ARGYRIS, Chris. *On Organizational Learning*. 2. ed. Oxford: Blackwell, 1999. 400p.
- AWS. *Amazon CloudFormation: Model and manage your AWS resources with templates*. Disponível em: <https://aws.amazon.com/cloudformation/>. Acesso em: 10 out. 2024.
- AWS. *Amazon Lex: Build conversational interfaces into any application*. Disponível em: <https://aws.amazon.com/lex/>. Acesso em: 10 out. 2024.
- BOSWORTH, Adam; CELKO, Joe; WONG, John C. *APIs and Data Management for High-Performance Systems*. 3. ed. San Francisco: Morgan Kaufmann, 2019. 560p.
- CHACON, Scott; STRAUB, Ben. *Pro Git*. 2. ed. Berkeley: Apress, 2014. 458p. Disponível em: <https://git-scm.com/book/en/v2>. Acesso em: 10 out. 2024.
- CROCKFORD, Douglas. *JavaScript: The Good Parts*. 1. ed. Sebastopol: O'Reilly Media, 2008. 172p.
- DOCKER. *Docker Overview: Introduction to Docker and containerization*. Disponível em: <https://docs.docker.com/get-started/>. Acesso em: 10 out. 2024.
- FIELDING, Roy Thomas. *Architectural Styles and the Design of Network-based Software Architectures*. Irvine: University of California, 2000. 162p.
- FLANAGAN, David. *JavaScript: The Definitive Guide*. 7. ed. Sebastopol: O'Reilly Media, 2020. 1096p.

GONÇALVES, Marcelo. *Scrum e Kanban: Colaboração e Produtividade em Equipes de Desenvolvimento de Software*. São Paulo: Novatec, 2021. 220p.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. *Deep Learning*. Cambridge: MIT Press, 2016. 775p.

HANSEN, James; FOSTER, Helen; NELSON, Travis. *Weather Data APIs for Real-Time Meteorological Information*. 2. ed. Cambridge: Cambridge University Press, 2020. 380p.

HASTIE, Trevor; TIBSHIRANI, Robert; FRIEDMAN, Jerome. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2. ed. New York: Springer, 2009. 745p.
HÜTTERMANN, Michael. *DevOps for Developers*. Berkeley: Apress, 2012. 184p.

JURAFSKY, Daniel; MARTIN, James H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 3. ed. Upper Saddle River: Pearson, 2021. 1024p.

KUMAR, Sandeep; SRINIVAS, Vivek. *Security and Privacy in Digital Health: A Practical Guide*. 1. ed. London: CRC Press, 2017. 268p.

MERKEL, Dirk. *Docker: Lightweight Linux Containers for Consistent Development and Deployment*. 1. ed. Berkeley: Apress, 2014. 98p.

MOZILLA DEVELOPER NETWORK. *Document Object Model (DOM)*. Disponível em: https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model. Acesso em: 10 out. 2024.

NORMAN, Donald A. *The Design of Everyday Things*. Revised and Expanded ed. New York: Basic Books, 2013. 368p.

OLSSON, Anders; ENGDAHL, Viktor. *Secure API Development and OAuth 2.0*. Estocolmo: API Academy, 2021. Disponível em: <https://apiacademy.se/secure-api-development>. Acesso em: 10 out. 2024.

ORACLE. *The Java™ Tutorials*. Disponível em: <https://docs.oracle.com/javase/tutorial/>. Acesso em: 10 de março de 2024.

PYTHON SOFTWARE FOUNDATION. *Python Programming Language*. Disponível em: <https://docs.python.org/3/>. Acesso em: 10 out. 2024.

ROBERTS, Mike. *Serverless Architectures on AWS: With Examples Using AWS Lambda*. 1. ed. Sebastopol: O'Reilly Media, 2019. 320p.

RUSSELL, Stuart; NORVIG, Peter. *Artificial Intelligence: A Modern Approach*. 4. ed. New Jersey: Prentice Hall, 2021. 1132p.

SCHWABER, Ken; SUTHERLAND, Jeff. *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. 2020. Disponível em: <https://scrumguides.org/scrum-guide.html>. Acesso em: 10 out. 2024.

SENGE, Peter M. *A Quinta Disciplina: Arte e Prática da Organização que Aprende*. 14. ed. São Paulo: BestSeller, 2006. 462p.

SERVERLESS FRAMEWORK. *Serverless Framework Documentation*. Disponível em: <https://www.serverless.com/framework/docs/>. Acesso em: 20 de janeiro de 2024.

SOMMERVILLE, Ian. *Software Engineering*. 9. ed. Boston: Addison-Wesley, 2011. 792p.
SPOTIFY. *Spotify for Developers: Spotify API documentation*. Disponível em: <https://developer.spotify.com/documentation>. Acesso em: 10 out. 2024.

TURNBULL, James; HIGGINBOTHAM, Rachel; BOWMAN, Scott. *Designing Accessible APIs and Applications*. 1. ed. Cambridge: Cambridge University Press, 2016. 310p.

VILLAMIZAR, Mario et al. *Evaluating the Scalability of Serverless Computing for Cloud-Based Applications*. 3. ed. New York: Springer, 2016. 365p.

WANG, Xiaojin; ZHENG, Yuan. *Conversational AI: Real-World Use Cases and Open Problems*. 1. ed. New York: Springer, 2019. 450p.

ZAHARIA, Matei et al. *Learning Spark: Lightning-Fast Data Analytics*. 2. ed. Sebastopol: O'Reilly Media, 2020. 350p.

ANEXO I

GLOSSÁRIO DE TERMOS TÉCNICOS

API (Application Programming Interface): Interface de Programação de Aplicações. Conjunto de rotinas e padrões de programação que permitem a comunicação entre diferentes sistemas de software.

AWS (Amazon Web Services): Plataforma de serviços em nuvem oferecida pela Amazon, que fornece infraestrutura escalável, como computação, armazenamento e ferramentas de machine learning.

AWS Rekognition: Serviço da AWS de visão computacional que permite a análise de imagens e vídeos, incluindo reconhecimento facial e detecção de objetos.

AWS Polly: Serviço de conversão de texto em fala da AWS, que transforma texto em arquivos de áudio de alta qualidade.

Amazon S3 (Simple Storage Service): Serviço de armazenamento escalável da AWS, usado para armazenar e recuperar grandes quantidades de dados na nuvem.

Amazon SageMaker: Serviço da AWS que permite construir, treinar e implantar modelos de machine learning com facilidade.

CORS (Cross-Origin Resource Sharing): Mecanismo de controle de acesso que define como os recursos de uma aplicação web podem ser requisitados a partir de um domínio diferente.

DynamoDB: Banco de dados NoSQL da AWS, com capacidade de escalar automaticamente e fornecer baixa latência.

Docker: Plataforma de software que permite a criação e gerenciamento de containers, facilitando o desenvolvimento e a execução de aplicações em ambientes isolados.

Elastic Beanstalk: Serviço da AWS que facilita o gerenciamento e implantação de aplicações web, utilizando serviços em nuvem como balanceamento de carga, escalabilidade e monitoramento.

Express.js: Framework web para Node.js que simplifica o desenvolvimento de aplicações e APIs de back-end, gerenciando requisições HTTP e middlewares.

Framework Serverless: Ferramenta para o desenvolvimento de aplicações que utilizam a arquitetura serverless, facilitando o gerenciamento e a implantação de funções sem a necessidade de configurar servidores.

Git: Sistema de controle de versão distribuído usado para gerenciar e versionar o código-fonte, permitindo o trabalho colaborativo em projetos de desenvolvimento de software.

Kanban: Metodologia ágil de gerenciamento de projetos que utiliza um sistema visual para organizar tarefas, maximizando a eficiência e flexibilidade.

Machine Learning (ML): Aprendizado de máquina. Subcampo da inteligência artificial que permite que sistemas aprendam a partir de dados e tomem decisões sem serem explicitamente programados para isso.

OAuth (Open Authorization): Padrão aberto para autorização, permitindo o compartilhamento seguro de recursos entre aplicações sem expor credenciais do usuário.

Random Forest: Algoritmo de aprendizado supervisionado utilizado em tarefas de classificação e regressão, que cria múltiplas árvores de decisão e as combina para melhorar a precisão.

Regex (Regular Expressions): Expressões regulares. Sequências de caracteres que formam um padrão de busca, utilizadas para validação e manipulação de strings em linguagens de programação.

Serverless: Arquitetura de computação em nuvem na qual a alocação de servidores é gerenciada automaticamente, permitindo que os desenvolvedores se concentrem na lógica de negócios, sem a necessidade de gerenciar infraestrutura.

Slack: Plataforma de comunicação em equipe que permite troca de mensagens, arquivos e integração com ferramentas de automação e bots.

Transcribe: Serviço da AWS que converte áudio em texto, usado para criar transcrições automáticas de conversas ou gravações.

Visão Computacional: Campo da inteligência artificial que permite que sistemas automatizados interpretem e compreendam o conteúdo de imagens e vídeos.