# Image to Speech: Accessibility Helper



## Team Members:

1. Sahil Bagwan(NUID: 001055189)
2. Aditya Salunkhe(NUID: 001348277)
3. Shruti Kasar(NUID: 001306991)

# Overview:

Machine replication of human functions, like reading, is an ancient dream. However, over the last five decades, machine reading has grown from a dream to reality.

Image to speech conversion is a trending aspect of computer technology. It determines an important criterion in which we interact with the system and interfaces across a variety of platforms.

It is a popular feature that lets your computer or phone read images aloud to you and is commonly used as an accessibility feature to help people who have trouble reading on-screen text, but it's also convenient for those who want to be read to.

Though the systems rely on a computerized voice speaking to you, in recent years these voices have become much more natural sounding. Many modern TTS voices are almost indistinguishable from humans, and some even incorporate natural human inflections to make them sound more lifelike.

Similar technology is used by Facebook to help blind or visually impaired users understand what is happening on the News Feed, a description of the photo is generated by Automatic Alt Text using object recognition technology.

# Goal:

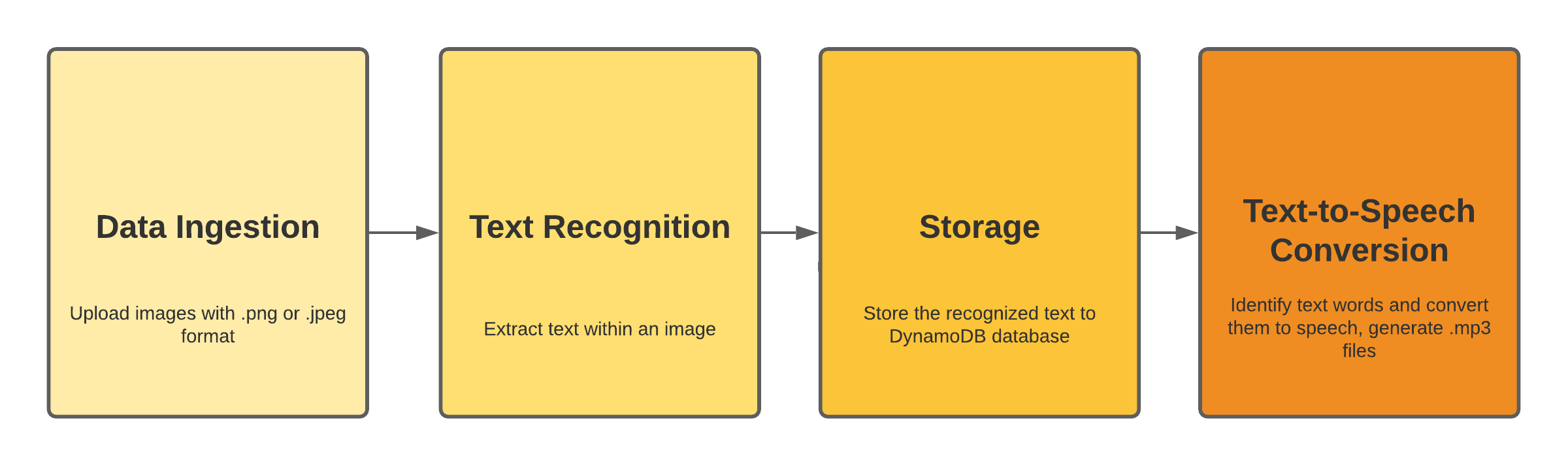


While there are many systems and applications already developed that detect objects, scenes, and faces; extract text; or systems which convert text to speech,a very few systems have been developed to extract textual information which is embedded in an image or scene and convert it to speech.

So with our project we intend to design a model/system to extract textual information from JPG, PNG files and convert it to speech.

The functionality of such a system can help to interpret information within images for people with disabilities.

# Process Outline:



## Part 1:

Here we provide a functionality for the user to upload images (in .png or .jpg format) they want to read out with a webpage hosted on Amazon S3. Those images will be stored in S3 for future reference and information retrieval .

## Part 2:

The extraction of the text from the uploaded images will happen in this part of the pipeline. This process will take place with the help of Amazon Rekognition Image. Rekognition Image uses deep neural network models based on CNNs to detect and label thousands of objects and scenes in your image. We’re using Amazon Rekognition for text detection.

## Part 3:

The extracted text from the previous steps will get stored in DynamoDB to be retrieved for future use as well as to be used as an input to the Text-to-Speech conversion job.

## Part 4:

The aim of the final part is to get extracted text converted to speech. The output mp3 will be generated and can be queried, played & downloaded as well via the front-end implementation.

# Services and Technologies Used:

### Amazon Rekognition:

* Rekognition Image is an image recognition service that detects objects, scenes, and faces; extracts text; recognizes celebrities; and identifies inappropriate content in images

### Amazon Polly:

* Amazon Polly is a service that turns text into lifelike speech, allowing you to create applications that talk, and build entirely new categories of speech-enabled products. It uses deep learning to synthesize natural sounding human speech. It also caters additional linguistic functionalities such as accent, voice speed etc.

### Amazon S3:

* Amazon Simple Storage Service is storage for the Internet. It is designed to make web-scale computing easier for developers
* Amazon S3 has a simple web services interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web

### DynamoDB:

* Amazon DynamoDB is a key-value and document database that delivers single-digit millisecond performance at any scale
* It's a fully managed, multi-region, multi-active, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications

### AWS Lambda:

* AWS Lambda is a serverless compute service that runs your code in response to events and automatically manages the underlying compute resources for you
* You can use AWS Lambda to extend other AWS services with custom logic, or create your own back-end services that operate at AWS scale, performance, and security

### Amazon SNS:

* Amazon Simple Notification Service (Amazon SNS) is a managed service that provides message delivery from publishers to subscribers (also known as producers and consumers)
* The feature Application-to-application messaging, supports subscribers such as AWS Lambda functions, Amazon SQS queues, HTTP/S endpoints, and AWS Event Fork Pipelines.

### Amazon API gateway:

* Amazon API Gateway is an AWS service for creating, publishing, maintaining, monitoring, and securing REST, HTTP, and WebSocket APIs at any scale.
* As an API Gateway API developer, you can create APIs for use in your own client applications.

## Programming Languages:

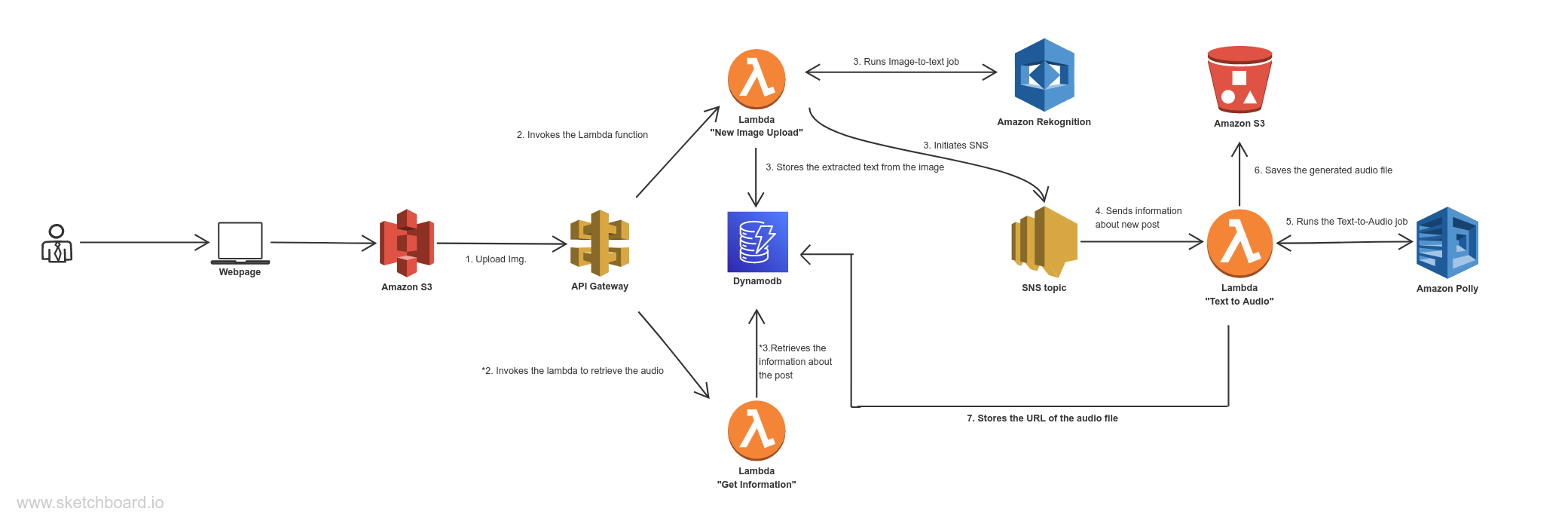
### Python

### JavaScript

### HTML

### CSS

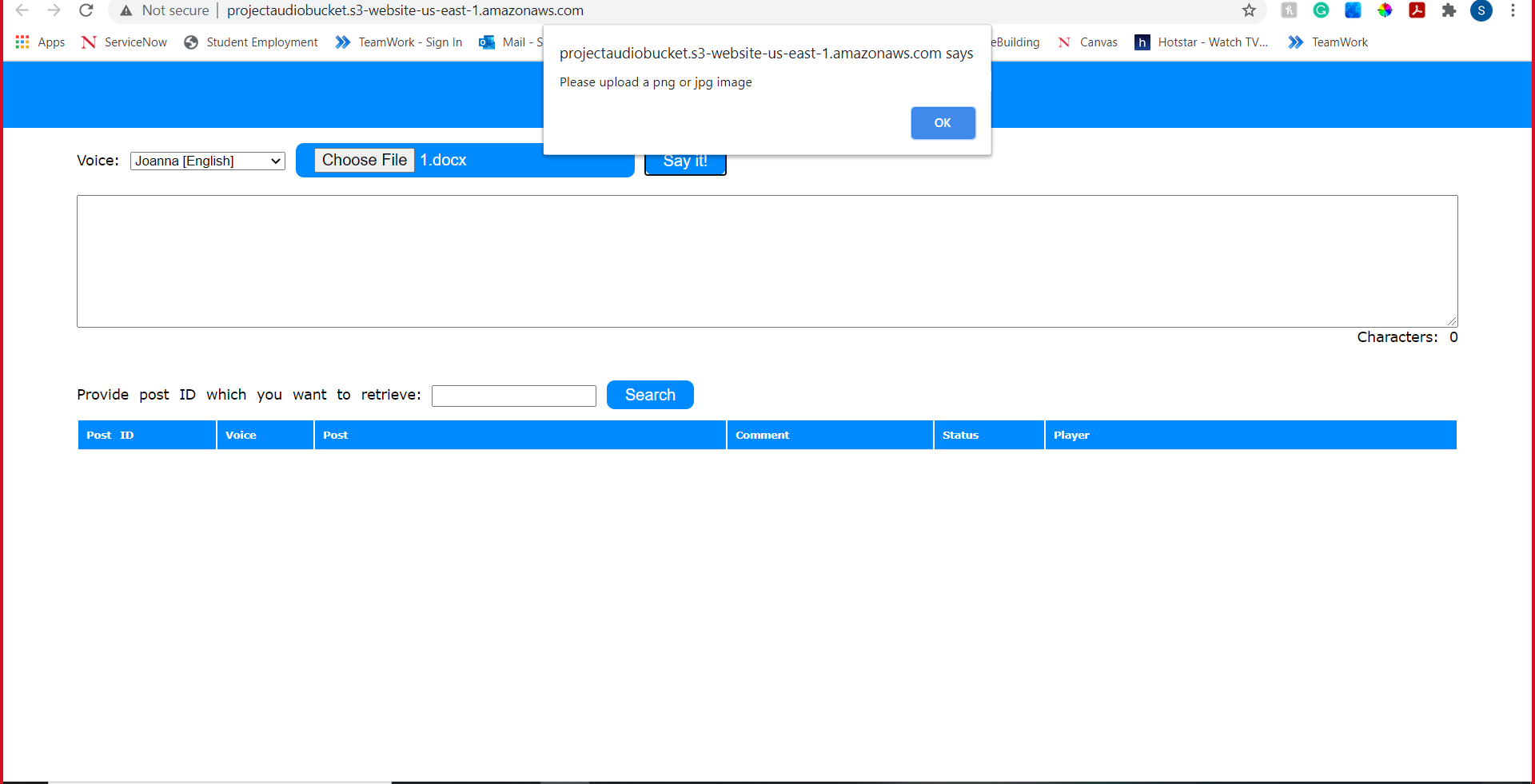
# System Architecture:



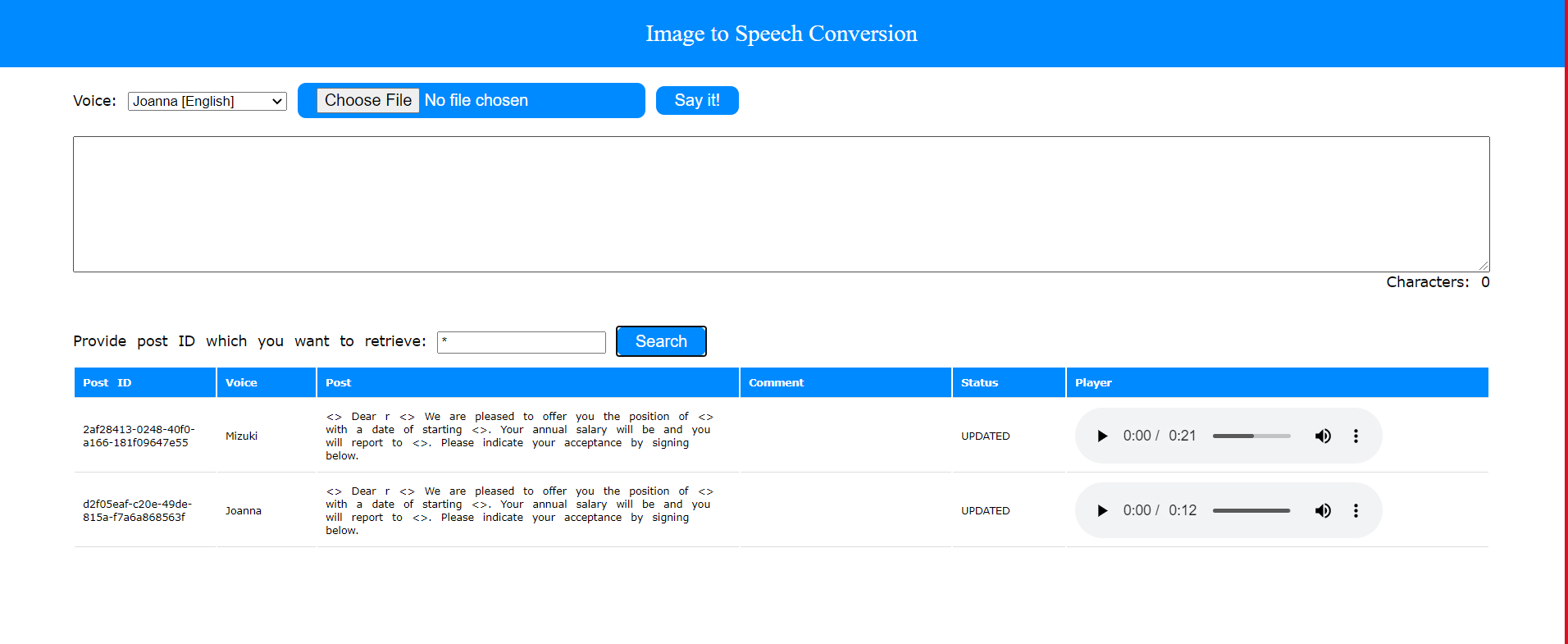
# 

# Test Cases:

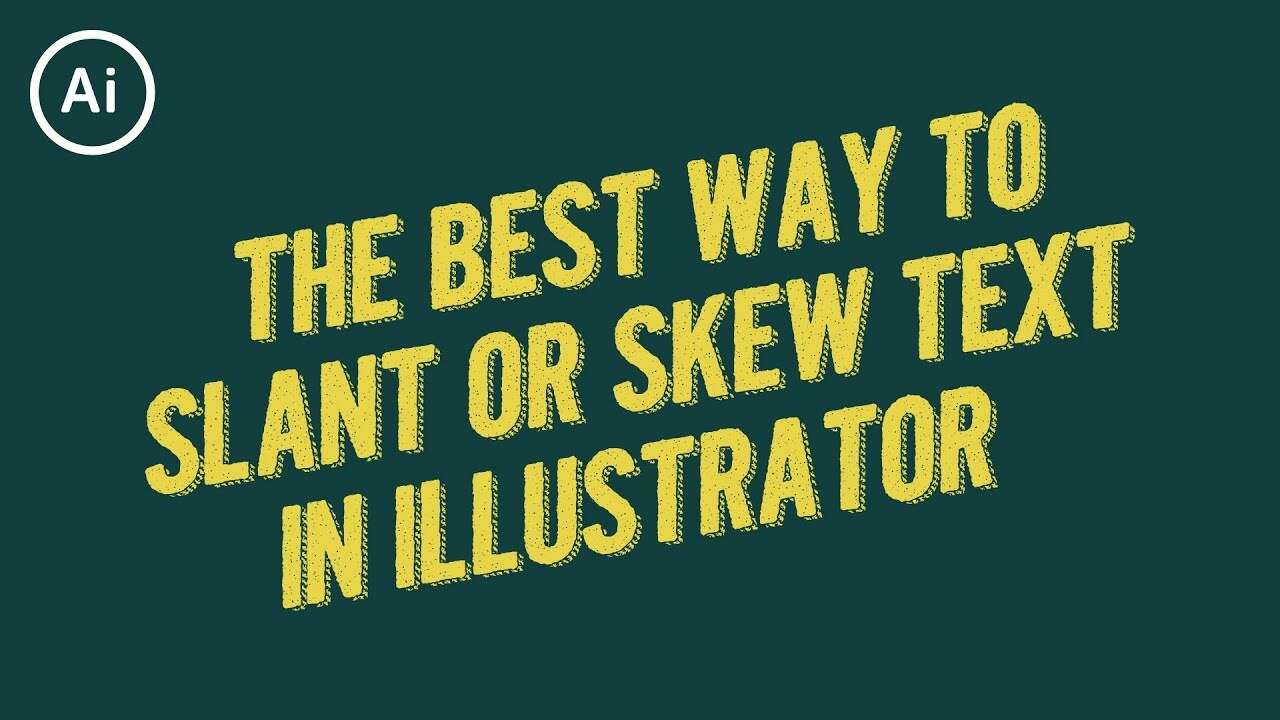
1. File format other than .png or .jpg

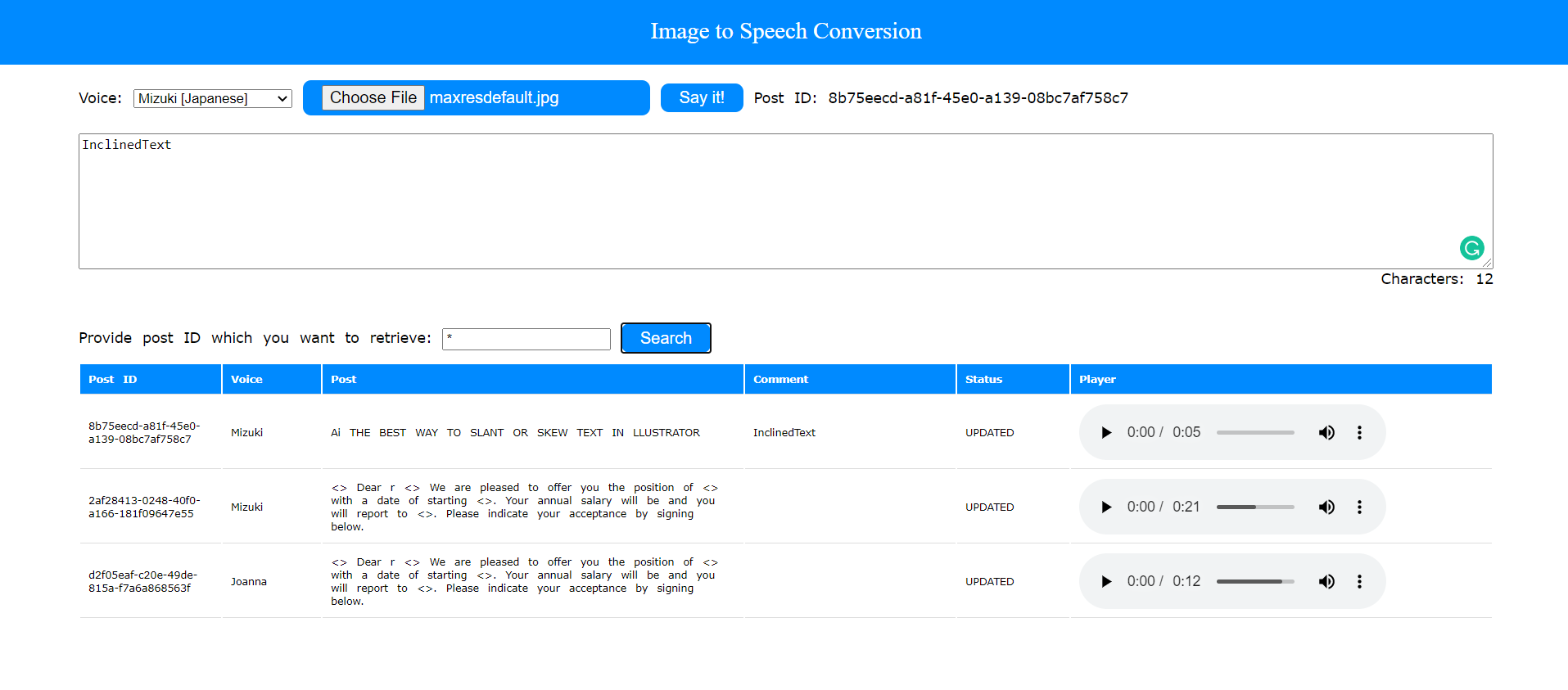


1. Voice configuration for same image with multiple voices



1. Images with inclined text-on-images





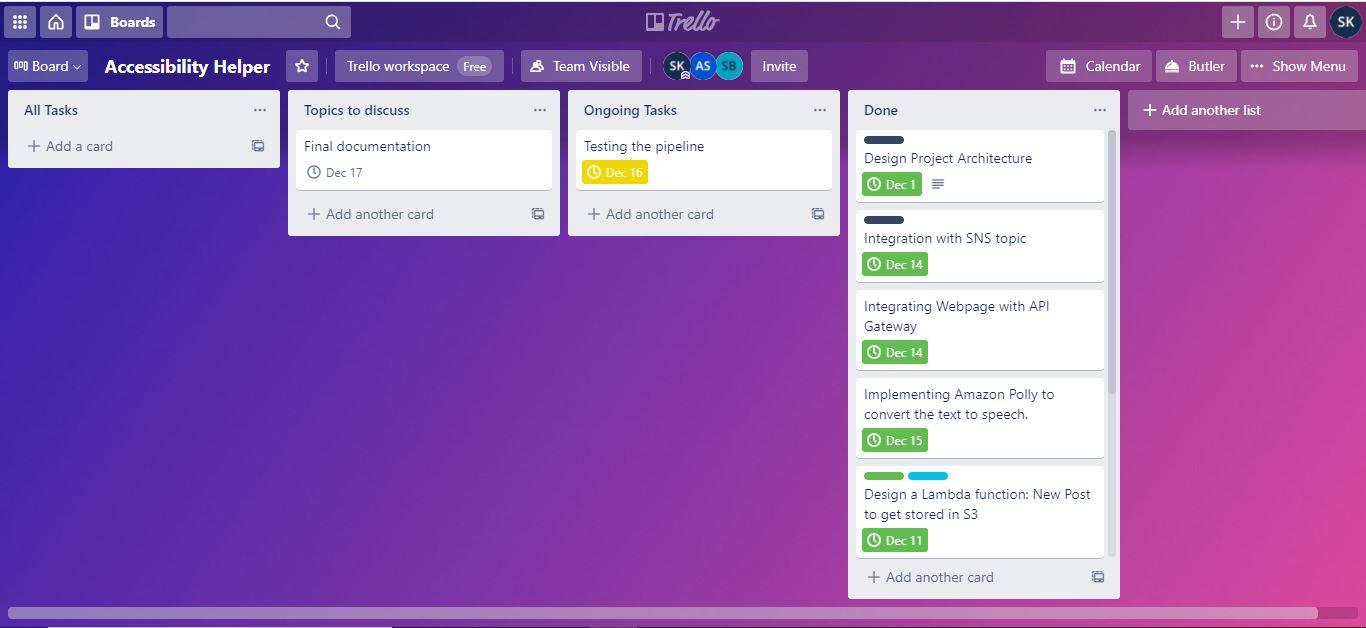
# Future Scope:

1. File formats other than .png & .jpg can be included to widen the scope of the project
2. Sentiment Analysis to predict the general notion behind the text contained in the image can be implemented

# Project Timeline:

|  |  |
| --- | --- |
| **Timeframe** | **Delivery** |
| Day 1-2 | Designing a webpage |
| Day 3-5 | Integrating the web page with APIGateway |
| Day 6-7 | Designing a Lambda function for new post and store it into Dynamodb |
| Day 7-10 | Deploying Amazon Rekognition and integrating it with SNS topic. |
| Day 10-12 | Implementing Amazon Polly to convert the text to speech. |
| Day 12-14 | Testing the pipeline & documentation |

We used Project management tool **Trello** for time management and to distribute the tasks.



# References:

<https://docs.aws.amazon.com/rekognition/latest/dg/text-detecting-text-procedure.html>

<https://docs.aws.amazon.com/rekognition/latest/dg/stored-video-lambda.html>