InANutShell

store.listen.learn.

Project Final Presentation

Instructor: Yan Liu

COEN6313 Programming On Cloud

Akhil Movva – 40106477

Main function of our web service

Store → The source of documents can be personal documents.

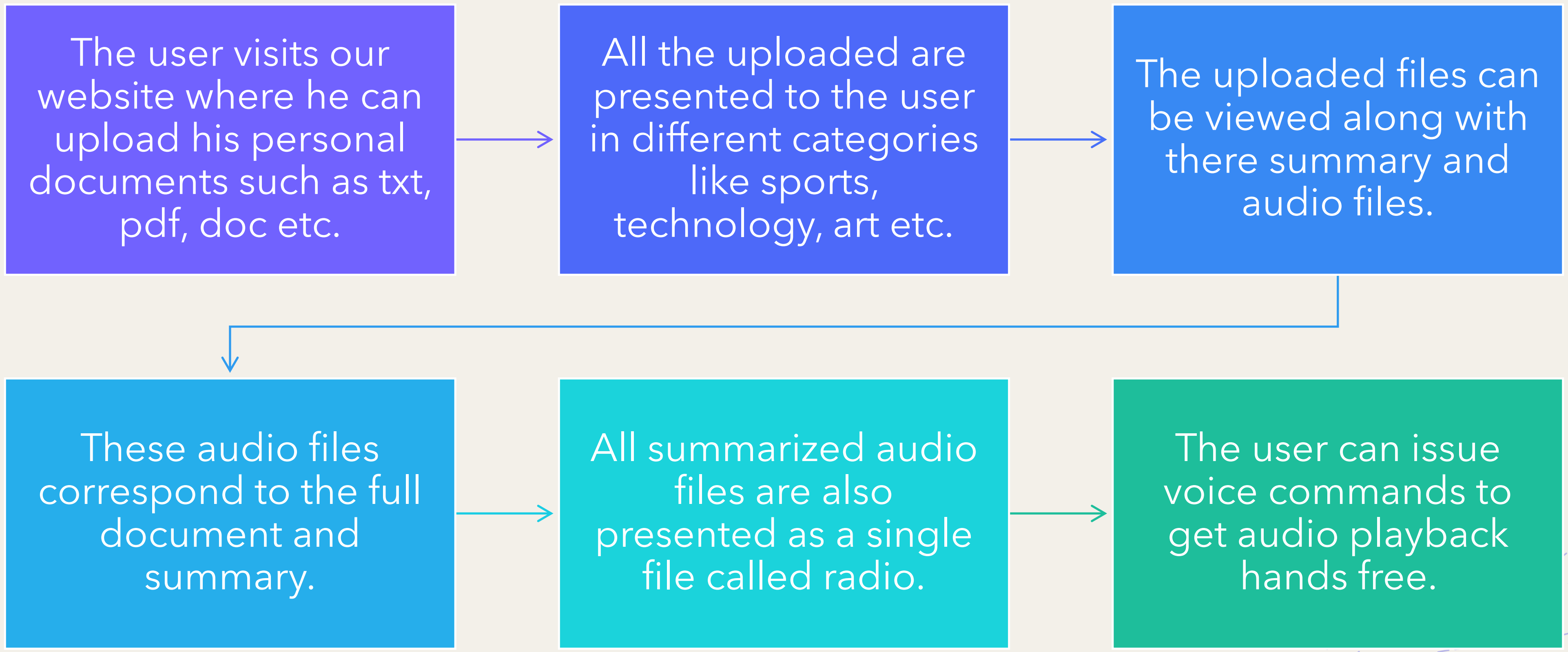
To keep the documents organized the app performs the following: Categorize all the documents in it based on article genres such as news, blogs, magazines, sports etc.

Listen → The application would extract the gist of each article and convert it into an audio clip.

The app also plays a stream of audio gist as a radio with channels being the category of documents.

Learn → The user can get a full audio playback of the chosen article

Requirements & Use Cases



The user visits our website where he can upload his personal documents such as txt, pdf, doc etc.

All the uploaded are presented to the user in different categories like sports, technology, art etc.

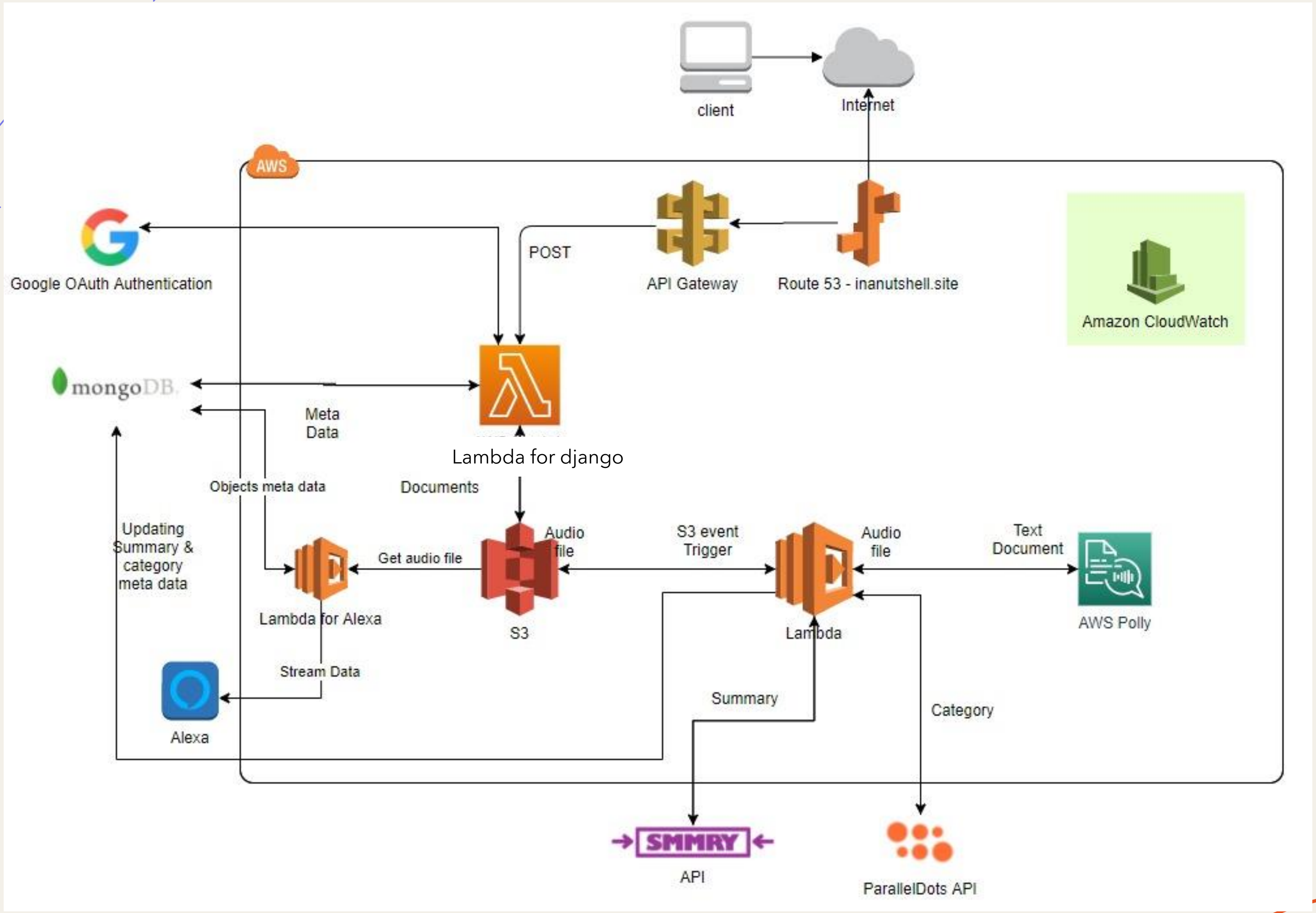
The uploaded files can be viewed along with there summary and audio files.

These audio files correspond to the full document and summary.

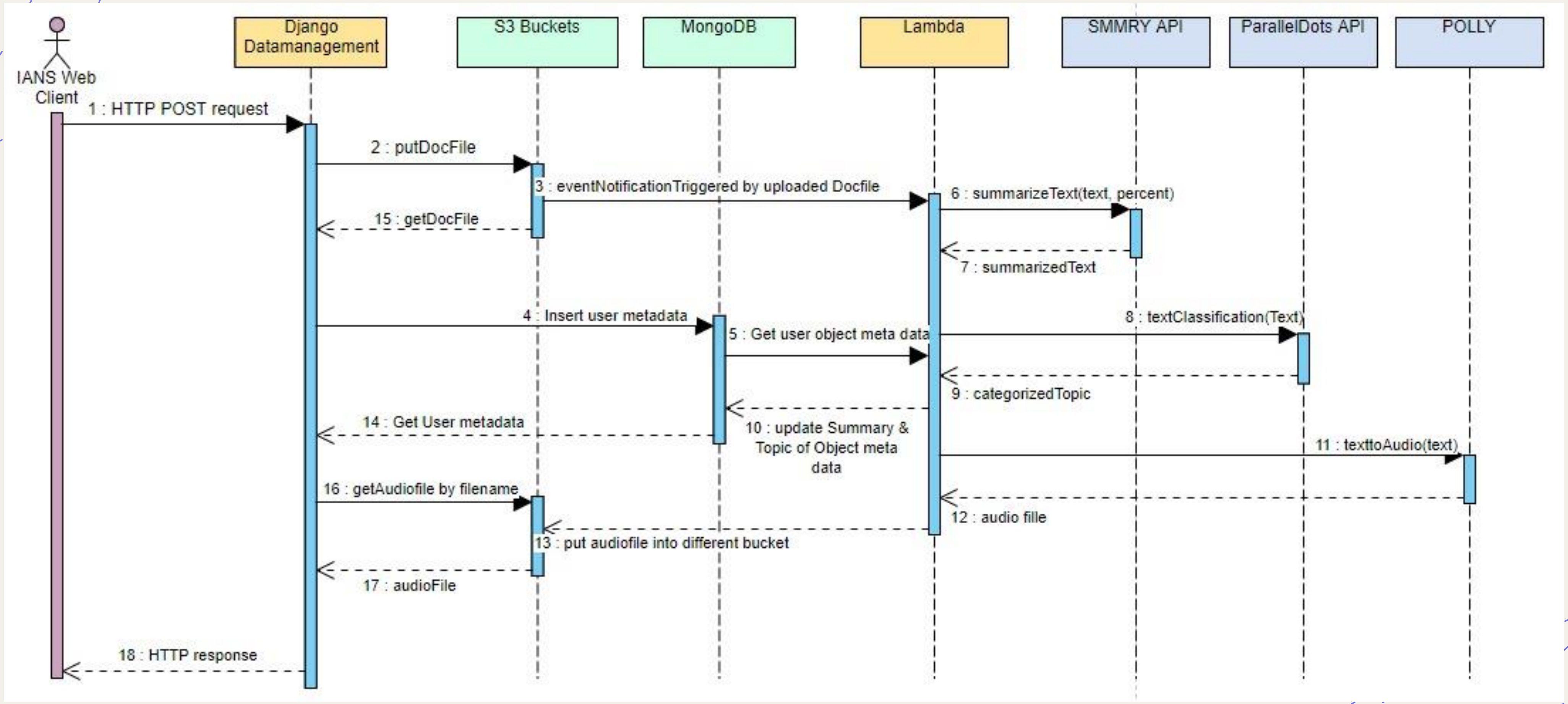
All summarized audio files are also presented as a single file called radio.

The user can issue voice commands to get audio playback hands free.

Architecture Design



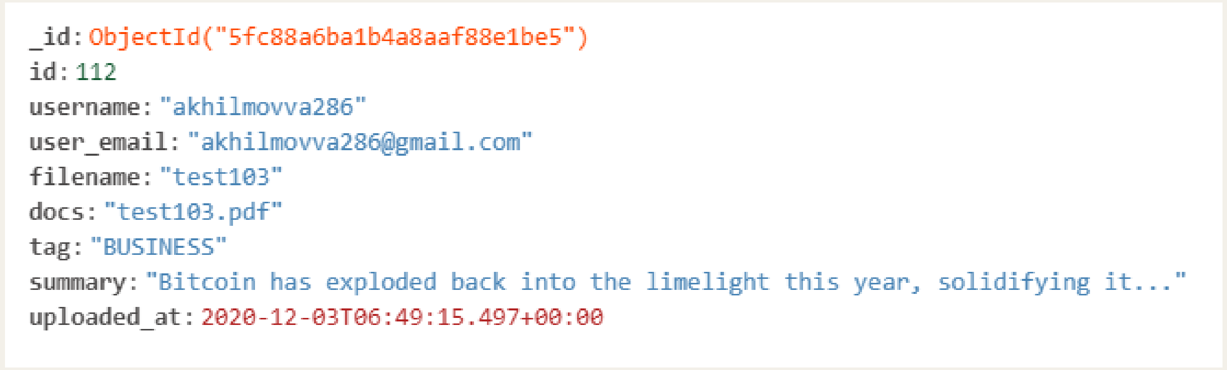
Sequence Diagram



Architecture Design and Technologies

+ Files Data Model :

Key	Value
ID	Integer
Username	String
user_email	String
filename	String
docs	FileField
Tag	String
summary	String
uploaded_at	DateTimeField



```
_id: ObjectId("5fc88a6ba1b4a8aaf88e1be5")
id: 112
username: "akhilmovva286"
user_email: "akhilmovva286@gmail.com"
filename: "test103"
docs: "test103.pdf"
tag: "BUSINESS"
summary: "Bitcoin has exploded back into the limelight this year, solidifying it..."
uploaded_at: 2020-12-03T06:49:15.497+00:00
```

+ Storage : MongoDB Atlas
+ Access to data : NoSQL access to a remote server

Additional Features developed

Integration of web application with Alexa.

An Alexa voice interaction model has been developed for achieving the functionality of using voice commands to access personal documents.

Important functions of the model:

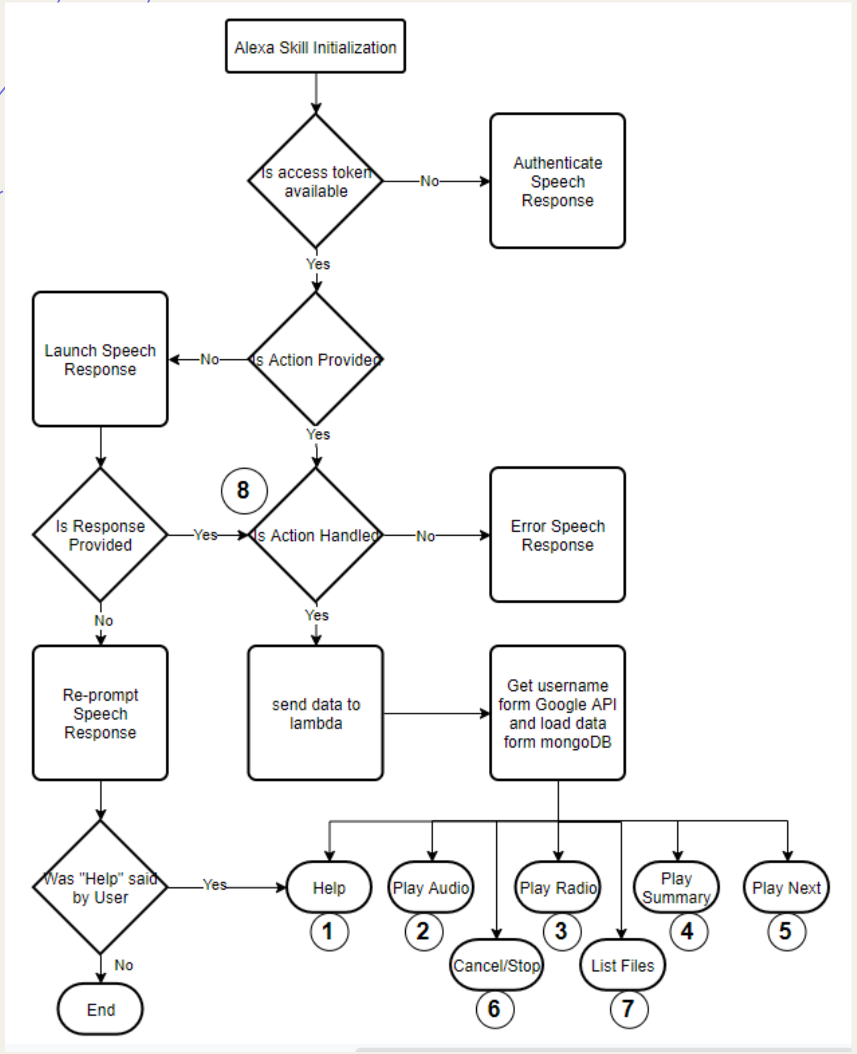
Accessing any file through voice commands

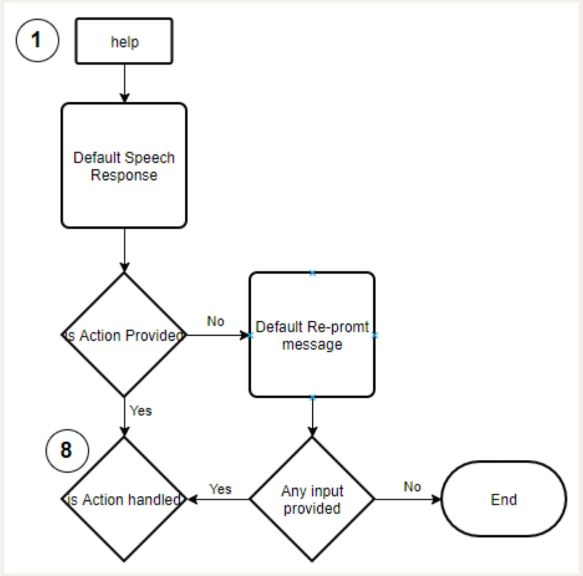
Listen to stream of documents as a radio

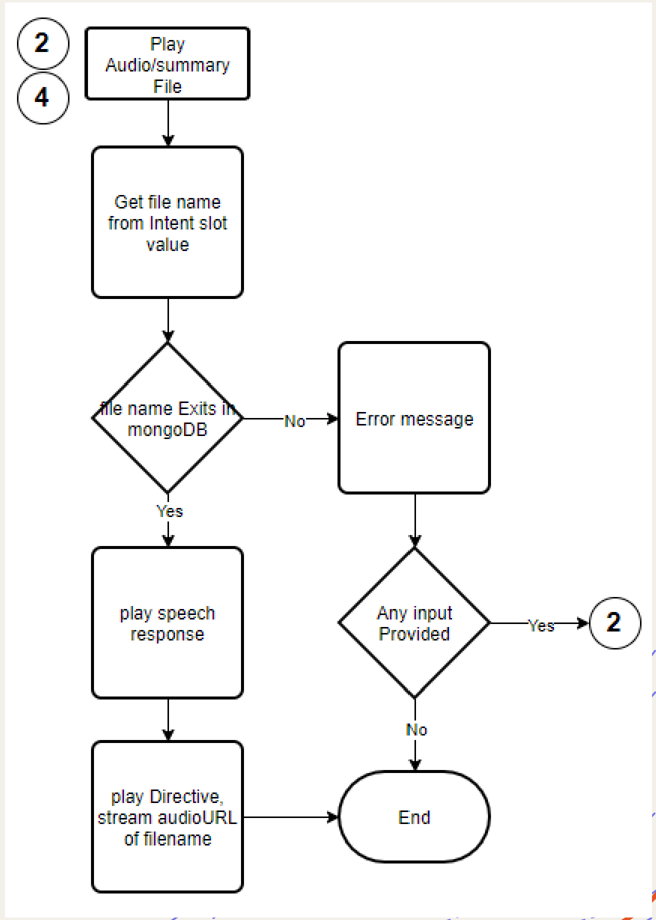
Alexa Design parameters

- **Invocation Name**: “hey cloud”.
- **Built-in Intents**: AMAZON.CancelIntent, AMAZON.HelpIntent, AMAZON.StopIntent, AMAZON.PauseIntent, AMAZON.ResumeIntent, AMAZON.NextIntent.
- **Custom Intents**: GetSmmryAudioIntent, GetAudioIntent, GetRadioIntent, ListFileIntent.
- Audio Interfaces and Account Linking

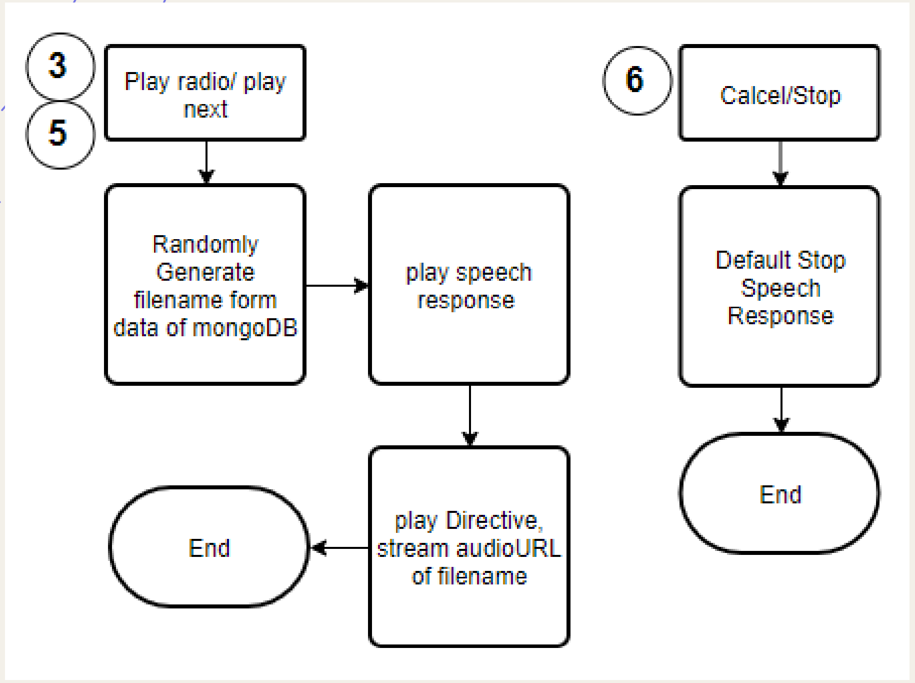
Voice Interaction Model for Alexa Skill

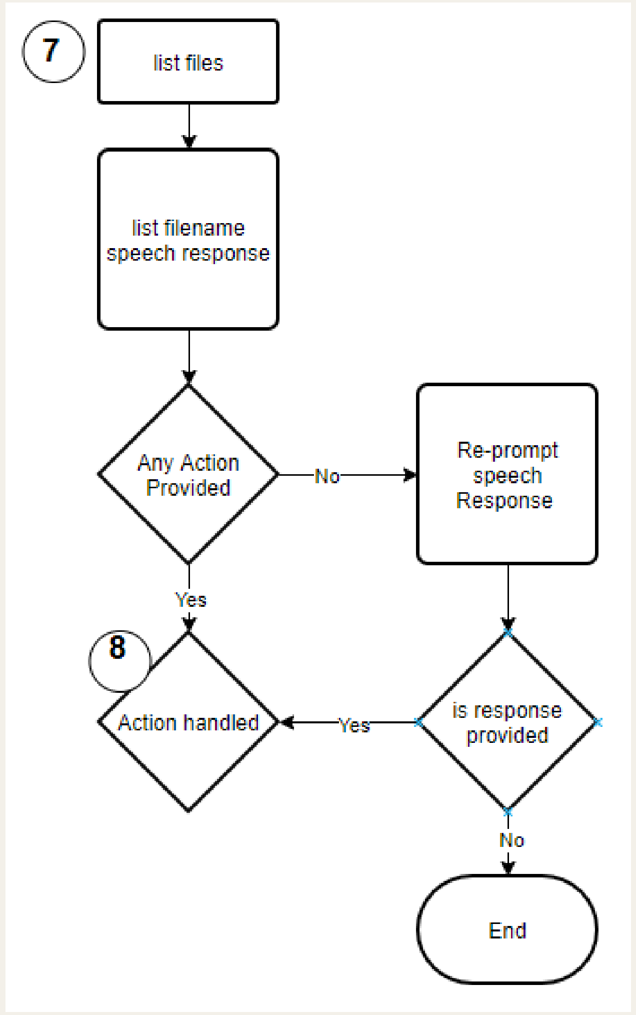





2
4
Play Audio/summary File
Get file name from Intent slot value
file name Exits in mongoDB
No
Error message
Yes
play speech response
Any input Provided
Yes
2
No
play Directive, stream audioURL of filename
End

Interaction Model - cont.


3
5
Play radio/ play next
Randomly Generate filename form data of mongoDB
play speech response
play Directive, stream audioURL of filename
End
6
Calcel/Stop
Default Stop Speech Response
End



Technologies used

For the Front-end we have used Html-CSS-Bootstrap-JavaScript.

For the back-end we have used Django which is python-based web framework. The boto3 SDK is used to communicate Django web application to AWS services.

The web application is made serverless using compute service AWS lambda. To deploy Django on lambda we used a service called 'zappa'.

We have chosen AWS s3 as our default file storage to store data such as user documents and audio files. For storing meta data we are using MongoDB which is NoSQL database.

To convert text documents to audio we are using AWS Polly.

'SMMRY' and 'Parallel dots' API are used for document summarization and classification

Alexa skill is used for enabling voice interaction model in the web application.

What works and what doesn’t work?

- Working functionalities:
  - Authentication for every user
  - Documents categorization
  - Document summarization
  - Audio clips for all the documents and their summaries
  - Voice commands for listening to audios or stream of audios
  - Our application is serverless
- What Doesn’t Work :
  - Support for other types of files such as pptx, docx and web links. Could be solved through containerization using AWS Fargate.
  - There is a 3000 character limit for audio conversion using Amazon Polly.
  - No custom signup for users.

Quality Attributes achieved

- Our web service is fully serverless.
- Due to deployment of web application on AWS lambda all the quality attributes are based on features of lambda tool.
- Performance features:
  - Process lambda events within seconds so that the web application can be triggered with low latency to the users. The bottlenecks are the external service APIs whose performance depend upon the type of subscription.
- Scalability and Availability:
  - AWS Lambda uses replication and redundancy to provide high availability and horizontal scalability for the Lambda functions ensuring no downtime for the application and the ability to scale to any number of users through concurrent lambda executions.
- Security:
  - All data in transit is HTTPS providing secure connection. The data at rest is at MongoDB and S3. MongoDB provides its own encryption and data in s3 can be protected through different S3 encryption schemes. AWS Lambda stores code in Amazon S3 with encryption. It also performs additional integrity checks while code is in use.

Strengths/Limitations of the cloud technologies

Strengths:

- Pay only for what you use model - Serverless
- Horizontal Scalability
- High Availability
- Infrastructure as a service(IaaS)
- Rapid provisioning of resources(Low Latency)

Limitations:

- Provisioned Concurrency for Lambda limits its performance.
- API Gateway has throttling limits given a burst limit of 5,000 and an account-level rate limit of 10,000 requests per second.
- AWS Polly has 3000 character limit for text to audio conversion.
- SMMRY API free account limits 100 requests/day
- Paralleldots API free monthly trial limits 1000 requests/day

Thank You

Our Live Demo

Any Questions ?