

Software Requirements Specification Document

Team Number	43
Project Title	Tools for Speech Recognition
Team Members	Adhish Singla Nitish Jain Shivam Khandelwal Romil Aggarwal

Brief problem statement

As part of Akshar Speech technology development, who need to build massive training sets with speech recordings. These speech recordings need to be pre- and post-processed, which requires the usage of tools. They have offered development of three tools to us viz. a Web Crawler, a Voice Recorder and ITRANS.

System requirements

Tool 1:

- PHP
- Apache
- MySql
- Python
- NLPK

Tool 2:

- PyQt
- Python
- Sox
- Excode
- Wavesurfer

Tool 3:

- SH file

Users profile

Tool 1 : No user, i.e. Crawler to be run by us (developers).

Tool 2 : Distributed users across the world.

Tool 3 : Akshar Speech users (employees)

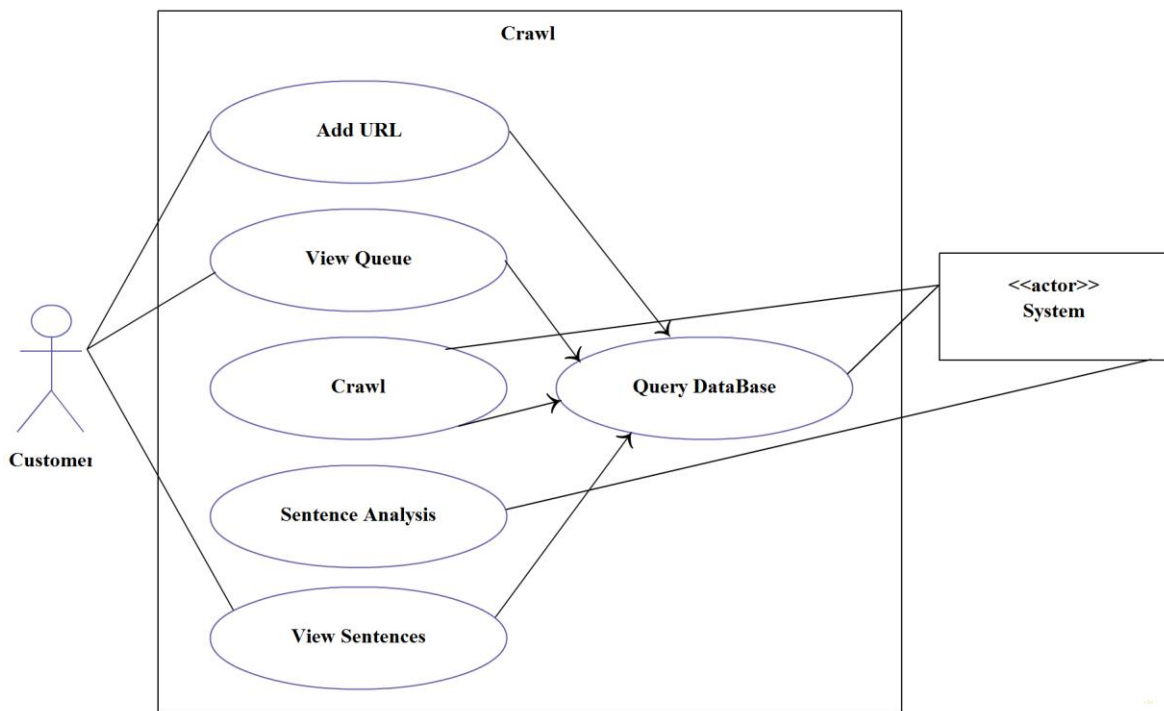
Feature requirements (described using use cases)

No.	User Case Name	Description	Release
1.	UC-10	Enables Customer to add url for crawling.	R1
2.	UC-11	Allows the Customer to view all the first 10 pending URL's in the queue.	R1
3.	UC-12	Crawls 5 URL's from the present URL queue.	R1
4.	UC-13	Saves all the unique words and phenomes of the sentences.	R1
5.	UC-14	Allows the customer to view the sentences and their correspnding phenomes and unique words.	R1
6.	UC-20	Enables the Customer to select an input file.	R1
7.	UC-21	Displays a line on the Interface.	R1
8.	UC-22	Allows the customer to record his voice.	R1
9.	UC-23	Processes the voice recorded by the Customer.	R1
10.	UC-24	Saves the analyzed file.	R1
11.	UC-30	Make .tar.gz file of ITRANS script file, UNICODE script file and Audio files	R2

Use Cases:

Tool 1:

Use case diagram



Use case description

Use Case Number:	UC-10
Use Case Name:	Add URL
Overview:	The Customer must be able to add url for crawling.
Actors:	Customer
Pre condition:	No Pre condition Required.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none">1. The customer clicks on the add button to open the form.2. The customer adds the url.3. The customer selects the option for recursive call.4. The customer mentions the sentence length.5. The customer clicks on the save button to save the form.
	<p>Alternate Flows:</p> <ol style="list-style-type: none">5. a. If anything is missing from the form, throw an error and ask the customer to fill the form again
Post Condition:	The URL is saved in the queue for being crawled.

Use Case Number:	UC-11
Use Case Name:	View Queue
Overview:	Allows the Customer to view all the first 10 pending URL's in the queue.
Actors:	Customer, System
Pre condition:	The Customer must have entered atleast 1 URL in the queue.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none">1. The Customer clicks on the view button.2. The last 10 URL's in the queue must be displayed.
	<p>Alternate Flows:</p> <ol style="list-style-type: none">2. a. If the Customer has saved less than 10 files show the corresponding no of files.

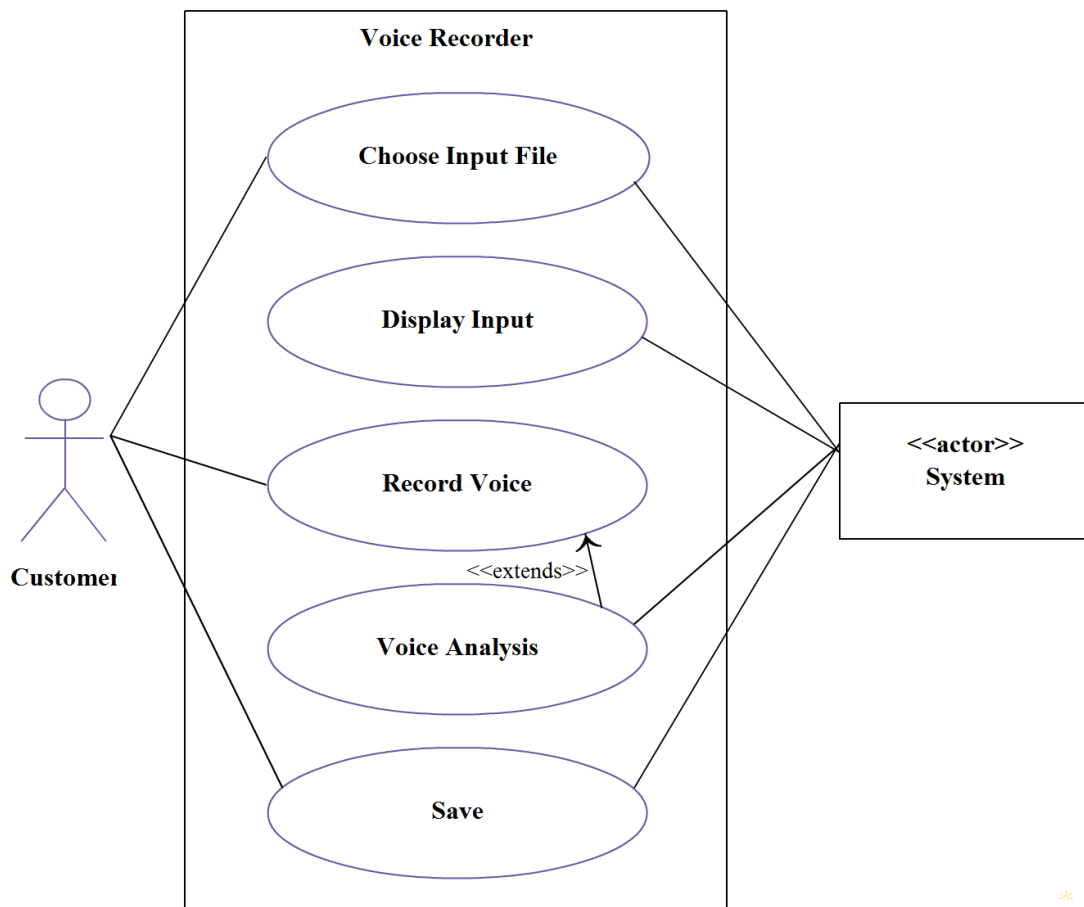
Use Case Number:	UC-12
Use Case Name:	Crawl
Overview:	Crawls 5 URL's from the present URL queue.
Actors:	System.
Pre condition:	There must be a queue of URL's for crawling wit atleast one URL for crawling and 1 min has expired since last crawling.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. System takes 5 or less URL's from the queue. 2. System crawls on the URL's and saves the sentences according to the lengths specified. 3. All the sentences are saved into Sentence Analysis queue.
Post Condition:	Removes the processed URL's from the URL queue.

Use Case Number:	UC-13
Use Case Name:	Sentence Analysis
Overview:	Saves all the unique words and phenomes of the sentences.
Actors:	System
Pre condition:	There must be some sentences in the Sentence Analysis queue.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. System takes sentences from the Sentence Analysis queue. 2. System figures out the unique words and phenomes from the selected sentence. 3. System saves the corresponding unique words and phenomes.
Post Condition:	Removes the corresponding sentence from the Sentence Analysis queue.

Use Case Number:	UC-14
Use Case Name:	View Sentences
Overview:	Allows the customer to view the sentences and their corresponding phenomes and unique words.
Actors:	Customer, System.
Pre condition:	There must be atleast one or more sentences and their corresponding phenomes and unique words.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. The customer clicks on the View Sentences Button. 2. The System displays all the Sentences in the Sentence Table along with their phenomes and unique words.
	<p>Alternate Flows:</p> <ol style="list-style-type: none"> 2. a. If there are no sentences in the Sentence Table, throw an error that no sentence present to show.

Tool 2:

Use case diagram



Use case description

Use Case Number:	UC-20
Use Case Name:	Choose Input file.
Overview:	Selecting input file.
Actors:	Customer, System
Pre condition:	No Pre condition required.
Flow:	Main (success) Flow: <ol style="list-style-type: none">1. Customer clicks on the option for selecting file.2. Customer chooses the file to be uploaded.3. Customer uploads the file.
	Alternate Flows: <ol style="list-style-type: none">3. a. If the customer tries to upload a wrong format file throw an error and ask him to upload the file again.

Use Case Number:	UC-21
Use Case Name:	Display Input
Overview:	Displays a line on the Interface.
Actors:	System
Pre condition:	Customer must have selected a valid Input File
Flow:	Main (success) Flow: <ol style="list-style-type: none">1. Keep a count of how many times UC-21 is called for a particular file.2. Checks Line Number Count in the input File.3. Validated line is printed on the interface.
	Alternate Flows: <ol style="list-style-type: none">3. a. If the Input File has no more lines to be printed, it goes to UC-20.
Post Condition:	The text is displayed and the user is ready to Record.

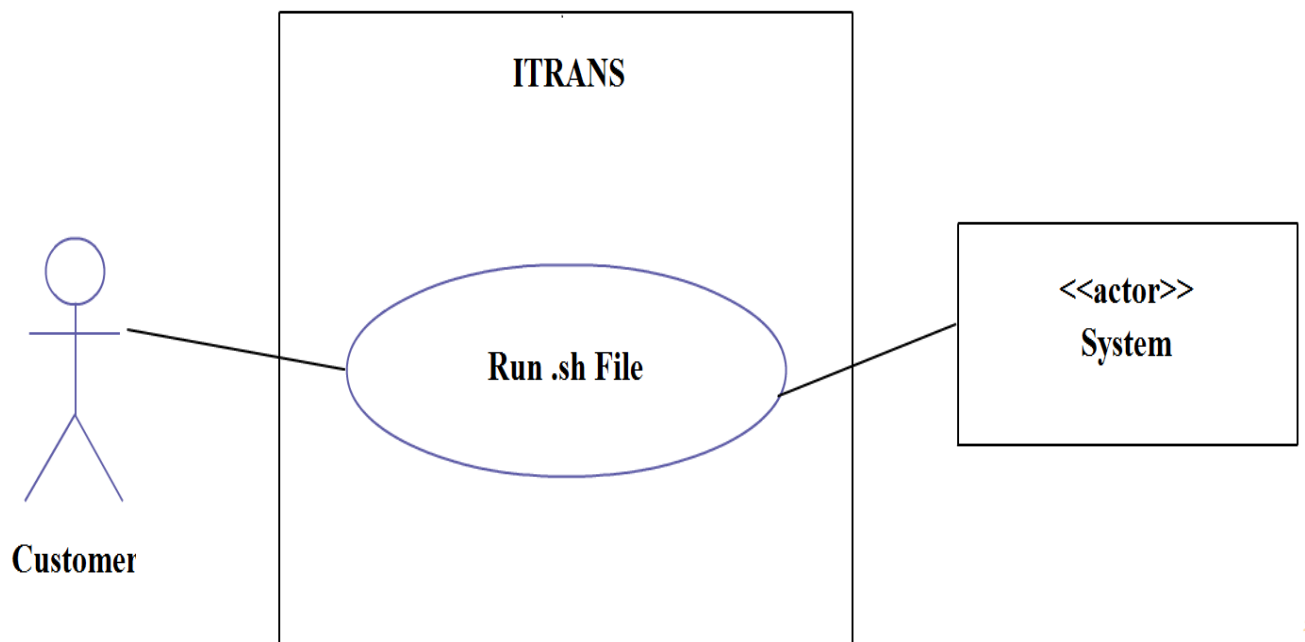
Use Case Number:	UC-22
Use Case Name:	Record Voice
Overview:	Enables Customer to record his voice.
Actors:	Customer
Pre condition:	The Customer must have selected some input file.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. The Customer Clicks on the record Button. 2. The Customer Records his voice. 3. The Customer Clicks on the Stop Button to stop recording.
Post Condition:	Save the recording made by the Customer.

Use Case Number:	UC-23
Use Case Name:	Voice Analysis
Overview:	Processing of the recorded voice.
Actors:	System
Pre condition:	The Customer must have recorded something
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. The system removes any available noise from the file 2. The system checks for silence in the end of file. 3. The manipulated file is saved.
	<p>Alternate Flows:</p> <ol style="list-style-type: none"> 2. a. If there is no silence in the end of the file, throw an error and ask the customer to record again.
Post Condition:	The original recorded file must be replaced by the filtered and manipulated file.

Use Case Number:	UC-24
Use Case Name:	Save
Overview:	Saves the analyzed file.
Actors:	System, Customer
Pre condition:	Customer must have recorded some voice.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none"> 1. Customer Browse Location to Store Analyzed Voice. 2. Analyzed Voice is Stored at the Particular Location. 3. Goes back to UC-21.
Post Condition:	Customer is ready to Record next line of the Input File.

Tool 3:

Use case diagram



Use case description

Use Case Number:	UC-30
Use Case Name:	Run .sh file
Overview:	Make .tar.gz file of ITRANS script file, UNICODE script file and Audio files.
Actors:	Customer, System
Pre condition:	No Pre condition Required.
Flow:	<p>Main (success) Flow:</p> <ol style="list-style-type: none">1. The customer enters relative/absolute path of the folder containing the files2. The audio files present in that directory would be normalized.3. Individual .tar.gz file is created for each audio file(eg. 1.itrans, 1.unicode, 1.wav will be saved as 1.tar.gz)
	<p>Alternate Flows:</p> <ol style="list-style-type: none">1. a. If the relative or absolute path doesn't exist, throw an error and ask for a new path. b. If all the itrans, unicode and wave files are not present, throw an error entioning to check for all the files.
Post Condition:	The created .tar.gz file is saved.