# PROFESSIONAL TRAINING REPORT

**at**

**Sathyabama Institute of Science and Technology (Deemed to be University)**

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering

By

## CHILUKURI SRIVARSHINI

**REG. NO. 39110230**

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**SCHOOL OF COMPUTING**

**SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,**

**CHENNAI – 600119, TAMILNADU**

**APRIL2022**

|  |  |  |
| --- | --- | --- |
|  | **SATHYABAMA**  **INSTITUTE OF SCIENCE AND TECHNOLOGY** (DEEMED TO BE UNIVERSITY) **Accredited with Grade “A” by NAAC**  (Established under Section 3 of UGC Act, 1956)  JEPPIAAR NAGAR, RAJIV GANDHI SALAI  CHENNAI– 600119  [**www.sathyabama.ac.in**](http://www.sathyabama.ac.in) |  |

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**BONAFIDE CERTIFICATE**

This is to certify that this Project Report is the bonafide work of **CHILUKURI SRIVARSHINI (Reg. No: 39110230)** who carried out the project entitled “**SPEECH TO TEXT TOOL**” under my supervision from MARCH2022 to APRIL2022

Internal Guide

## Dr. G.MEERA GANDHI ,M.E., Ph.D

## Head of the Department



## Submitted for Viva voce Examination held on

**InternalExaminer ExternalExaminer**

**DECLARATION**

I, **CHILUKURI SRIVARSHINI**hereby declare that the project report entitled PRIDEdone by me under the guidance of **Dr. G.MEERA GANDHI** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering.

## DATE:

**PLACE: SIGNATURE OF THE CANDIDATE**

**ACKNOWLEDGEMENT**

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D**, **Dean**, School of Computing, **Dr. S. Vigneshwari, M.E., Ph.D. and Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department** of **Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. G.MEERA GANDHI, M.E., Ph.D.,** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project

**VOICE TO TEXT USING MACHINE LEARNING**

**ABSTRACT**

Speech recognition is one of the fastest-growing engineering technologies. It has several applications in different areas, and provides many potential benefits. A lot of people are unable to communicate due to language barriers. We aim to reduce this barrier via our project, which was designed and developed to achieve systems in particular cases to provide significant help so people can share information by operating a computer using voice input. This project keeps that factor in mind, and an effort is made to ensure our project is able to recognize speech and convert input audio into text; it also enables a user to perform file operations like Save, Open, or Exit from voice-only input. We design a system that can recognize the human voice as well as audio clips, and translate between English. The output is in text form, and we provide options to convert audio from one language. Going forward, we expect to add functionality that provides dictionary meanings for English words. Neural machine translation is the primary algorithm used in the industry to perform machine translation. This work on speech recognition starts with an introduction to the technology and the applications used in different sectors. Part of the report is based on software developments in speech recognition.

TABLE OF CONTENTS

CHAPTER TITLE NAME PAGE NO

ABSTRACT 6

LIST OF FIGURES 7

LIST OF ABBREVATIONS 8

1. INTRODUCTION 10

LITERATURE REVIEW 11

REAL TEXT TO SPEECH CONVERSION 12

AUTOMATIC SPEECH RECOGNITION 13

SPEECH TO TEXT CONVERSION 13

TEXT TO SPEECH ALGORITHM 13

2. PROPOSAL SYSTEM 14

SYSTEM REQUIREMENTS 15

PYTHON 15

HISTORY OF PYTHON 16

1. MODULE 24

FEATURE EXTRACTION MODULE 25

SPEECH TO TEXT MODULE 26

REQUIREMENT ANALYSIS 26

FUNCTIONAL REQUIREMENTS 26

1. CONCLUSION 43

REFERENCES 46

LIST OF FIGURES

FIGURE NO FIGURE NAME PAGE NO

1. ARCHITECTURE DIAGRAM 15

2. DATA FLOW DIAGRAM 32

3. USE CASE DIAGRAM 35

4. CLASS DIAGRAM 36

5. SEQUENCE DIAGRAM 37

6. SCREENSHOTS 44

LIST OF ABBREVIATIONS

ABBREVIATION EXPANSION

TCS TEXT TO SPEECH CONVERTER

ASR AUTOMATIC SPEECH RECOGNITION

HMM HIDDEN MARKOV MODEL

MFCC MEL FREQUENCY CEPSTRAL COEFFICIENTS

SST SPEECH TO TEXT

**OBJECTIVE:**

The main objective of our project is to encourage the use of our native language and to help illiterate people for the easy typing of the text. The concept involves the recognition of voice through a microphone. The background noise is reduced by using a noise filter cap over the mic. The words are extracted from the input voice by using some artificial intelligence and feature extraction. By using NLTK we convert the word and these words in the uttered speech are identified using word tokenizer. The extracted words are then compared with the pre-trained data set by using data analysis

**INTRODUCTION:**

Speech recognition is the ability of a machine or program to identify words and phrases in spoken language and convert them to a machine-readable format. Many speech recognition applications, such as voice dialing, simple data entry and speech-to-text are in existence today. Automatic speech recognition systems involve numerous separate components drawn from many different disciplines such as statistical pattern recognition, communication theory, signal processing, combinatorial mathematics, and linguistics. Speech recognition is an alternative to traditional methods of interacting with a computer, such as textual input through a keyboard. An effective system can replace, or reduce the reliability on, standard keyboard input Attempts to build automatic speech recognition (ASR) systems were first made in the 1950s. These early speech recognition systems tried to apply a set of grammatical and syntactical rules to identify speech. If the spoken words adhered to a certain rule set, the system could recognize the words. However, human language has numerous exceptions to its own rules. The way words and phrases are spoken can be vastly altered by accents, dialects and mannerisms. Therefore, to achieve ASR we make use of Algorithm.In modern civilized societies for communicationbetween human speeches is one of the commonmethods.Different ideas formed in the mind of thespeaker are communicated by speech in the form ofwords, phrases, and sentences by applying some proper grammatical rules.The speech is primary mode of communication amonghuman being and also the most natural and efficient form ofexchanging information among human in speech. By classifyingthe speech with voiced, unvoiced and silence (VAS/S)an elementary acoustic segmentation of speech which is essential for speech can be considered. Insuccession to individual sounds called phonemes thistechnique can almost be identical to the sounds of eachletter of the alphabet which makes the composition ofhuman speech.The mainpurpose of speech recognition is to convert the acousticsignal obtained from a microphone or a telephone to generate a set of words. In order to extract anddetermine the linguistic information conveyed by aspeech wave we have to employ computers.

**LITERATURE REVIEW:**

Literature survey is the most important step in software developmentprocess. Before developing the tool it is necessary to determine the time factor,economy and company strength. Once these things are satisfied, then the nextstep is to determine which operating system and language can be used fordeveloping the tool. Once the programmers start building the tool theprogrammers need lot of external support. This support can be obtained fromsenior programmers, from book or from websites. Before building the system theabove consideration are taken into account for developing the proposed system.The major part of the project development sector considers and fullysurvey all the required needs for developing the project. For every projectLiterature survey is the most important sector in software development process.Before developing the tools and the associated designing it is necessary todetermine and survey the time factor, resource requirement, man power,economy, and company strength. Once these things are satisfied and fully surveyed, then the next step is to determine about the software specifications inthe respective system such as what type of operating system the project would require, and what are all the necessary software are needed to proceed with thenext step such as developing the tools, and the associated operations.

**Real time text-to speech conversion system for Spanish**

Research aiming to develop a text-to-speech converter (TSC) for Spanish, that accepts a continuous source of alphanumeric characters (up to 250 words per minute) and produces good quality, natural Spanish output, is described. Four sets of problems are considered in this work: the hard-ware structure adopted for real time operation; the complex control software needed to handle the orthographic input and linguistic programs; the linguistic processing rules, and the parametrization of the Spanish language matched to a TSC. Emphasis is made on the problems of adapting a general hardware structure to a specific language.

**Acoustic modeling problem for automatic speech recognition system**

In automatic speech recognition (ASR) systems, the speech signal is captured and parameterized at front end and evaluated at back end using the statistical framework of hidden Markov model (HMM). The performance of these systems depend critically on both the type of models used and the methods adopted for signal analysis. Researchers have proposed a variety of modifications and extensions for HMM based acoustic models to overcome their limitations.

**SOPC-Based Speech-to-Text Conversion**

For the past several decades, designers have processed speech for a wide variety of applications ranging from mobile communications to automatic reading machines. Speech recognition reduces the overhead caused by alternate communication methods. Speech has not been used much in the field of electronics and computers due to the complexity and variety of speech signals and sounds.

**Text-to-speech algorithms based on FFT synthesis**

Present FFT synthesisalgorithms for a French text-to-speech system based on diaphone concatenation. FFT synthesis techniques arecapable of producing high quality prosodic adjustments of natural speech. Several different approaches are formulated toreduce the distortions due to diaphone concatenation.

**Explore Speech-to-Text(STT) conversion using SAPI for Bangla language**

Although achieved performance is promising for STT related studies, they identified several elements to recover the performance and might give better accuracy and assure that the theme of this study will also be helpful for other languages for Speech-to-Text conversion and similar tasks.

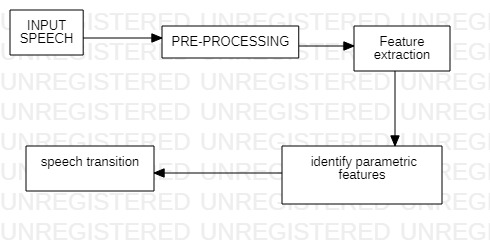
**EXISTING SYSTEM**

* The Java Speech API defines a standard, cross platform software interface to state-of-the-art speech technology. Two core speech technologies are supported through the Java Speech API: speech recognition and speech synthesis. Speech recognition provides computers with the ability to listen to spoken language and to determine what has been said. In other words, it processes audio input containing speech by converting it to text.
* But the main disadvantage is it recognize only the some reserved words only

**PROPOSED SYSTEM**

Speech recognition aids in translating the spoken language into text. We have come up with a Speech Recognition model that converts the speech data given by the user as an input into the text format in his desired language. This model is developed by adding Multilingual features to the existent Google Speech Recognition model based on some of the natural language processing principles.

**ARCHITECTURE DIAGRAM:**

****

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* System : Pentium Dual Core.
* Hard Disk : 120 GB.
* Monitor : 15’’ LED
* Input Devices : Keyboard, Mouse
* Ram : 4 GB.

**SOFTWARE REQUIREMENTS:**

* Operating system: Windows 7/10.
* Coding Language :Pyt**SOFTWARE ENVIRONMENT**

**Python:**

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

* **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
* **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
* **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

**History of Python:**

**Python Features:**

Python's features include −

* **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
* **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
* **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
* **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
* **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
* **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
* **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
* **Databases** − Python provides interfaces to all major commercial databases.
* **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
* **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below −

* It supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* It provides very high-level dynamic data types and supports dynamic type checking.
* It supports automatic garbage collection.
* It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

**Getting Python:**

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python [https://www.python.org](https://www.python.org/).

Windows Installation

Here are the steps to install Python on Windows machine.

* Open a Web browser and go to <https://www.python.org/downloads/>.
* Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.
* To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
* Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

**First Python Program**

Let us execute programs in different modes of programming.

**Interactive Mode Programming**

Invoking the interpreter without passing a script file as a parameter brings up the following prompt −

$ python

Python2.4.3(#1,Nov112010,13:34:43)

[GCC 4.1.220080704(RedHat4.1.2-48)] on linux2

Type"help","copyright","credits"or"license"for more information.

>>>

Type the following text at the Python prompt and press the Enter −

>>>print"Hello, Python!"

If you are running new version of Python, then you would need to use print statement with parenthesis as in **print ("Hello, Python!");**. However in Python version 2.4.3, this produces the following result −

Hello, Python!

**Script Mode Programming**

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension **.py**. Type the following source code in a test.py file −

print"Hello, Python!"

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows −

$ python test.py

This produces the following result −

Hello, Python!

The following table summarizes different http methods −

|  |  |
| --- | --- |
| **Sr.No** | **Methods & Description** |
| 1 | **GET**  Sends data in unencrypted form to the server. Most common method. |
| 2 | **HEAD**  Same as GET, but without response body |
| 3 | **POST**  Used to send HTML form data to server. Data received by POST method is not cached by server. |
| 4 | **PUT**  Replaces all current representations of the target resource with the uploaded content. |
| 5 | **DELETE**  Removes all current representations of the target resource given by a URL |

By default, the Flask route responds to the **GET** requests. However, this preference can be altered by providing methods argument to **route()** decorator.

In order to demonstrate the use of **POST** method in URL routing, first let us create an HTML form and use the **POST** method to send form data to a URL.

Save the following script as login.html

<html>

<body>

<formaction="http://localhost:5000/login"method="post">

<p>Enter Name:</p>

<p><inputtype="text"name="nm"/></p>

<p><inputtype="submit"value="submit"/></p>

</form>

</body>

</html>

Now enter the following script in Python shell.

from flask importFlask,redirect,url\_for, request

app=Flask(\_\_name\_\_)

@app.route('/success/<name>')

def success(name):

return'welcome %s'% name

@app.route('/login',methods=['POST','GET'])

def login():

ifrequest.method=='POST':

user=request.form['nm']

return redirect(url\_for('success',name= user))

else:

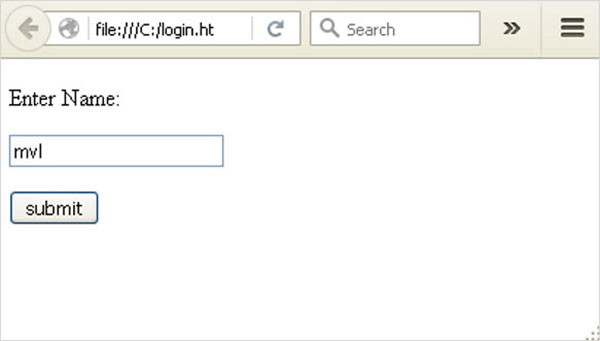
user=request.args.get('nm')

return redirect(url\_for('success',name= user))

if \_\_name\_\_ =='\_\_main\_\_':

app.run(debug =True)

After the development server starts running, open **login.html** in the browser, enter name in the text field and click **Submit**.

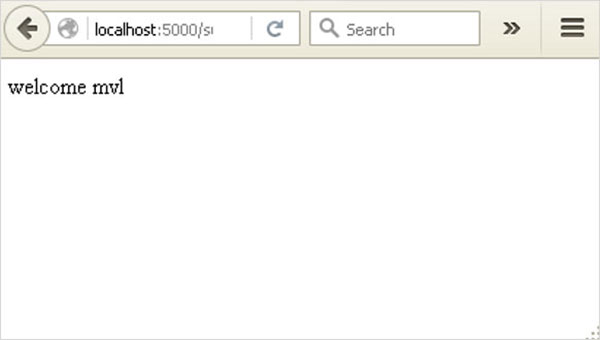


Form data is POSTed to the URL in action clause of form tag.

**http://localhost/login** is mapped to the **login()** function. Since the server has received data by **POST** method, value of ‘nm’ parameter obtained from the form data is obtained by −

user = request.form['nm']

It is passed to **‘/success’** URL as variable part. The browser displays a **welcome** message in the window.



Change the method parameter to **‘GET’** in **login.html** and open it again in the browser. The data received on server is by the **GET** method. The value of ‘nm’ parameter is now obtained by −

User = request.args.get(‘nm’)

Here, **args** is dictionary object containing a list of pairs of form parameter and its corresponding value. The value corresponding to ‘nm’ parameter is passed on to ‘/success’ URL as before

**MODULES:**

* Speech Analysis Module
* Feature Extraction Module
* Speech To Text Module

**MODULE DESCRIPTIONS:**

**Speech Analysis Module**

In Speech analysis technique Speech data contains different types of information that shows a speaker identity. This includes speaker specific information due to vocal tract, excitation source and behavior feature. The physical structure and dimension of vocal tract as well as excitation source are unique for each speaker. This uniqueness is embedded in the speech signal during speech production and can be used for speaker used for speaker recognition.

**Feature Extraction Module**

Feature Extraction is the most important part of speech recognition since it plays an important role to separate one speech from other. Because every speech has different individual characteristics embedded in utterances. These characteristics can be extracted from a wide range of feature extraction techniques proposed and successfully exploited for speech recognition task. But extracted feature should meet some criteria while dealing with the speech signal such as:

a.Easy to measure extracted speech features.

b.It should not be susceptible to mimicry.

c.It should show little fluctuation from one speaking environment to another.

d.It should be stable over time.

E.It should occur frequently and naturally in speech.

**Speech To Text Module:**

A real time speech to text conversion system converts the spoken words into text form exactly in the similar way that the user pronounces. We created a real time speech recognition system that was tested in real time noiseous environment. The purpose of this project was to introduce a new speech recognition system that is computationally simple and more robust to noise than the HMM based speech recognition system.

**REQUIREMENT ANALYSIS**

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analysing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

**FUNCTIONAL REQUIREMENTS**

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

**Usability**

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

**Robustness**

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

**Security**

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

**Reliability**

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and survive and even capable of working around any failure.

**Compatibility**

It is supported by version above all web browsers. Using any web servers like localhost makes the system real-time experience.

**Flexibility**

The flexibility of the project is provided in such a way that is has the ability to run on different environments being executed by different users.

**Safety**

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one’s personal information.

**NON- FUNCTIONAL REQUIREMENTS**

**Portability:**

It is the usability of the same software in different environments. The project can be run in any operating system.

**Performance:**

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

**Accuracy:**

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

**Maintainability:**

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyse, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

**SYSTEM DESIGN AND TESTING PLAN:**

**INPUT DESIGN:**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**SYSTEM STUDY**

**FEASIBILITY STUDY:**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**ECONOMICAL FEASIBILITY:**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### TECHNICAL FEASIBILITY:

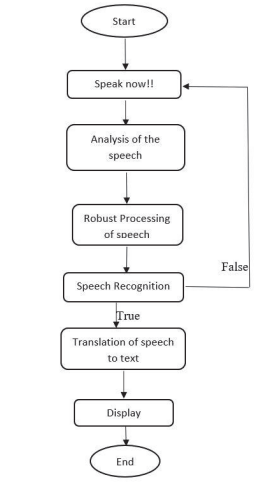
This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**SOCIAL FEASIBILITY:**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity.

**DATA FLOW DIAGRAM:**

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system.
2. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction



**UML DIAGRAMS:**

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

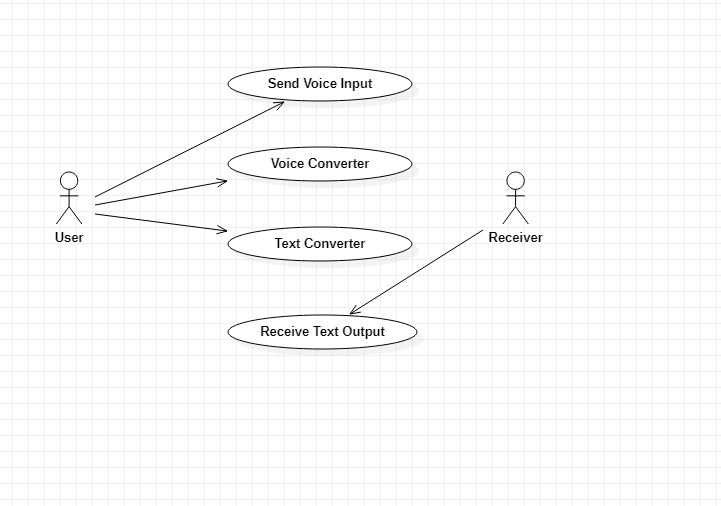
**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

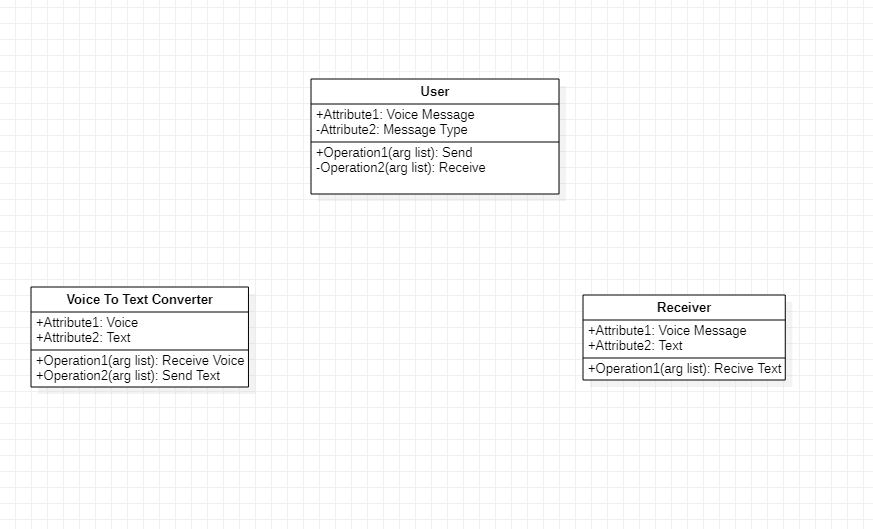
**USE CASE DIAGRAM:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



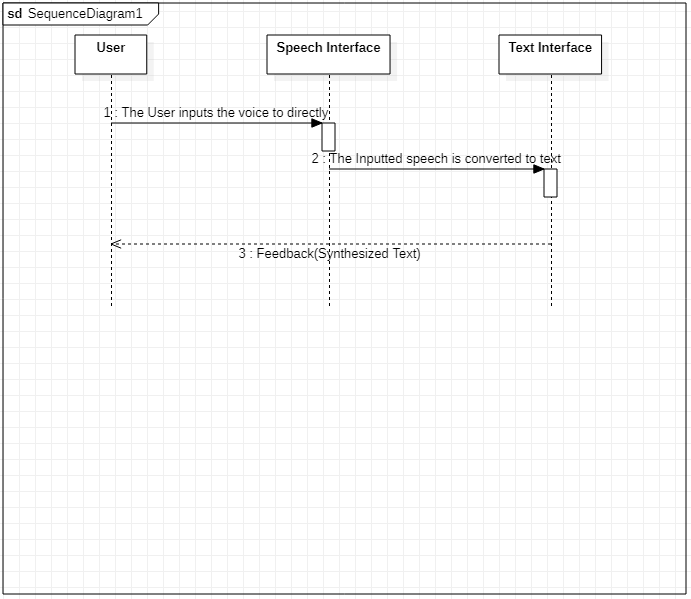
**CLASS DIAGRAM:**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



**SEQUENCE DIAGRAM:**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



### **SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**TYPES OF TESTS**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**6.1 Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

**Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

# 6.2 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**6.3 Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

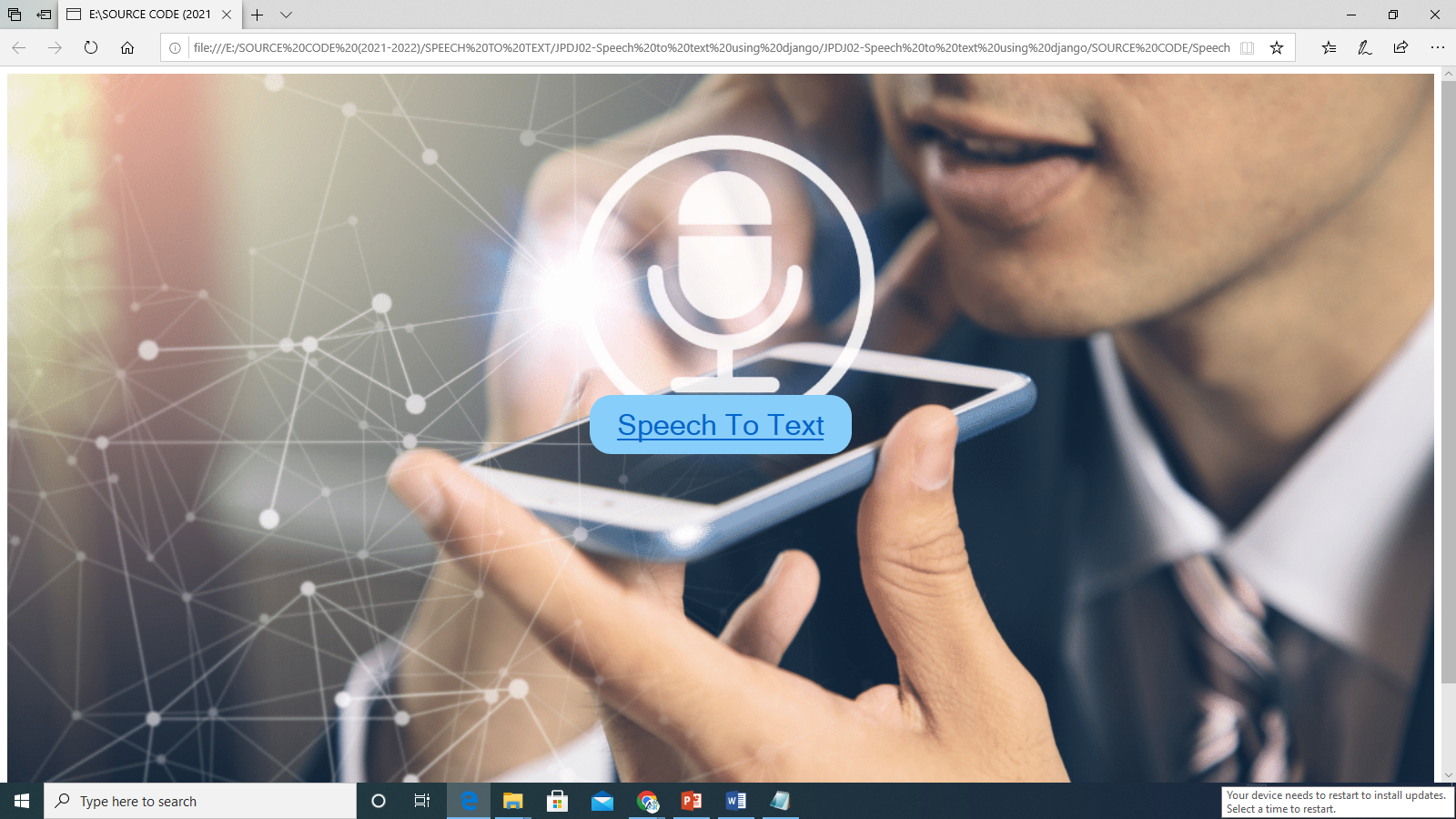
**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

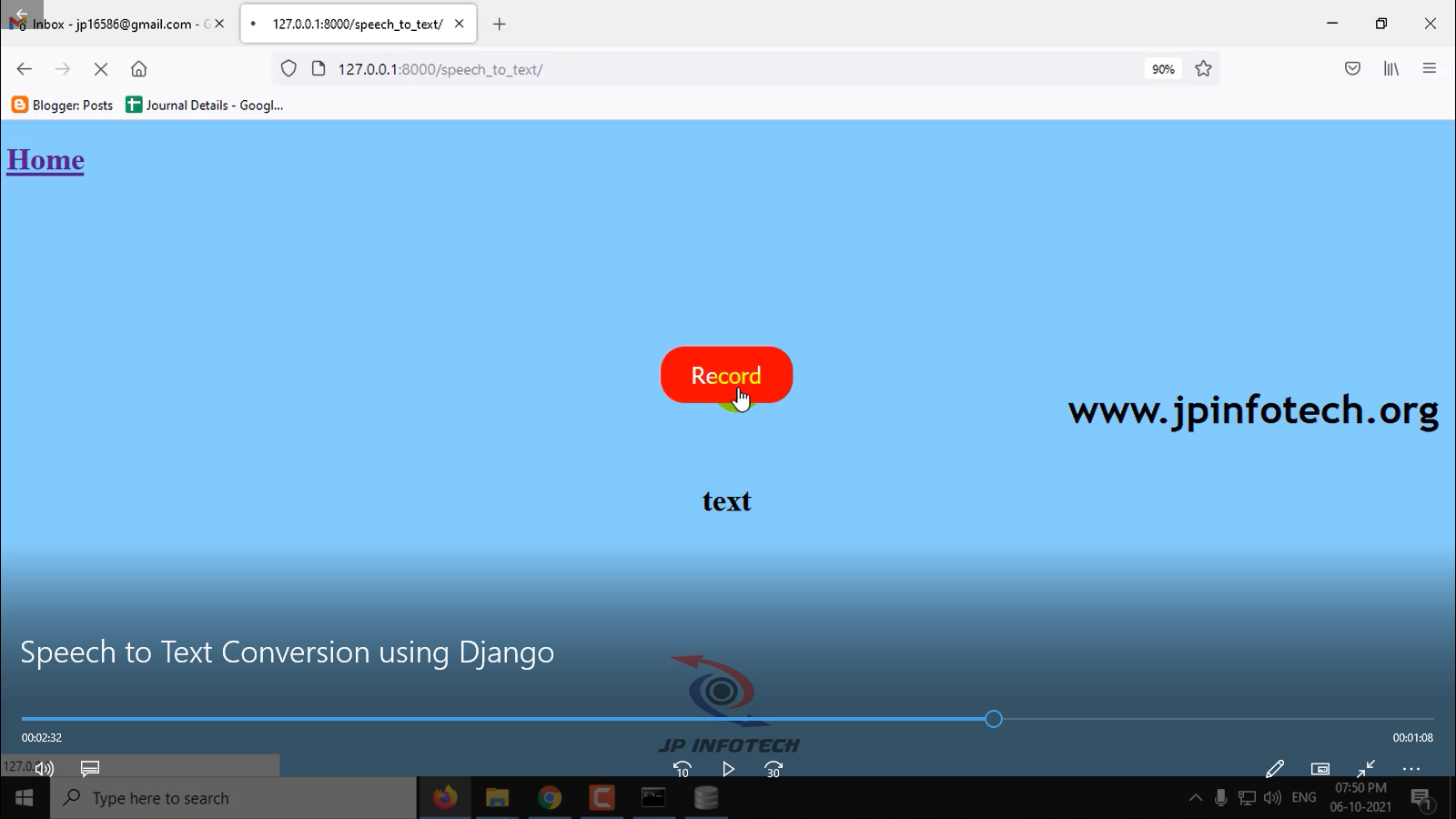
**CONCLUSION**

By implementing this model we have learned how can we use the SpeechRecognition packages to build a Speech Translation model. The more we use these kinds of packages we get more flexibility in the code and output that is to be displayed. This model can be used in any purpose of translation of speech to text. This model has high advantages, one among them is you can survive in unknown places where you don’t know the language to speak but with help of this model you can translate that regional speech to text and it can also be used in areas like telecommunication and multimedia. Moreover, this model is useful for providing effective communication between man and the machine.

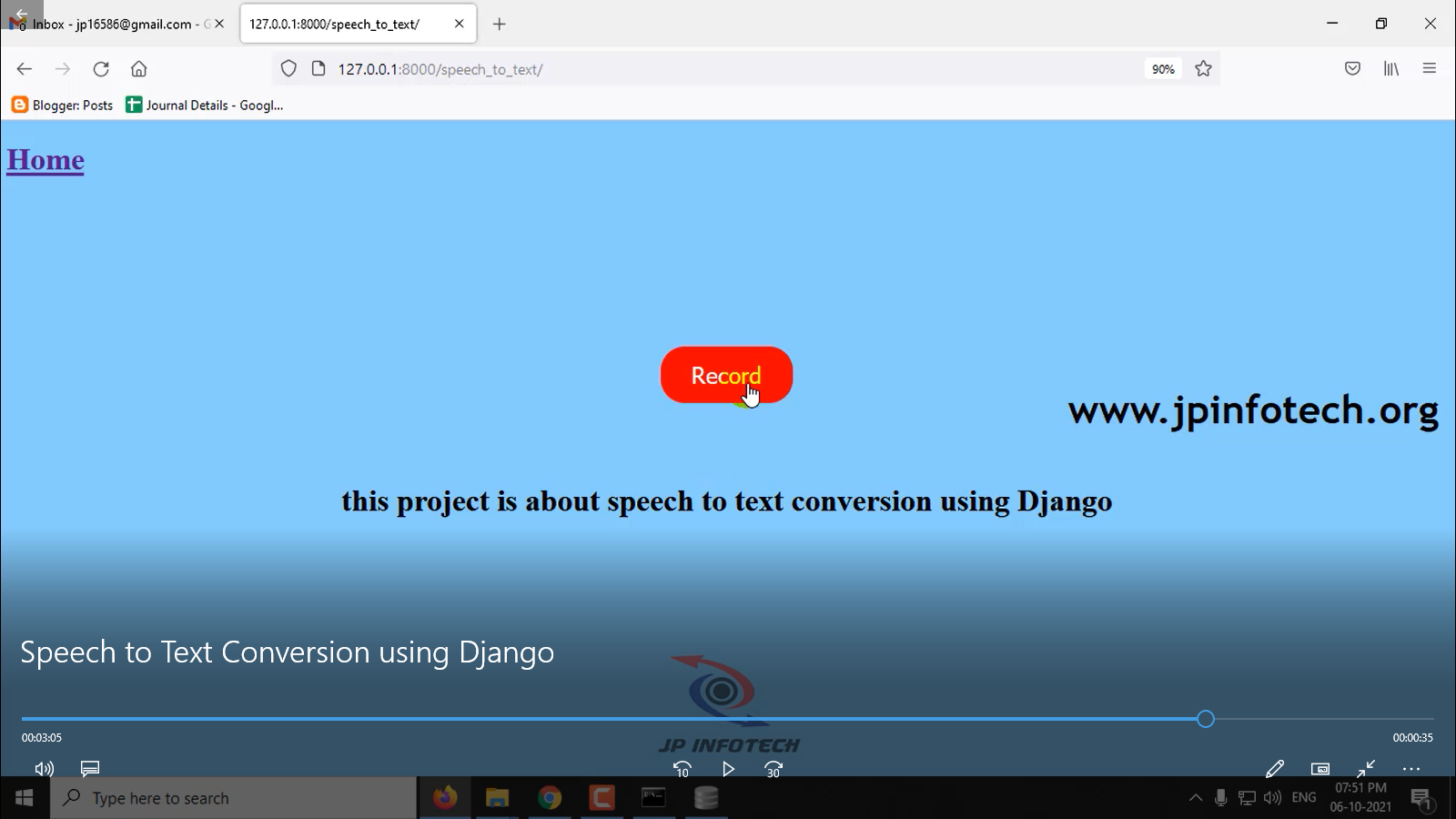
**SCREENSHOTS**

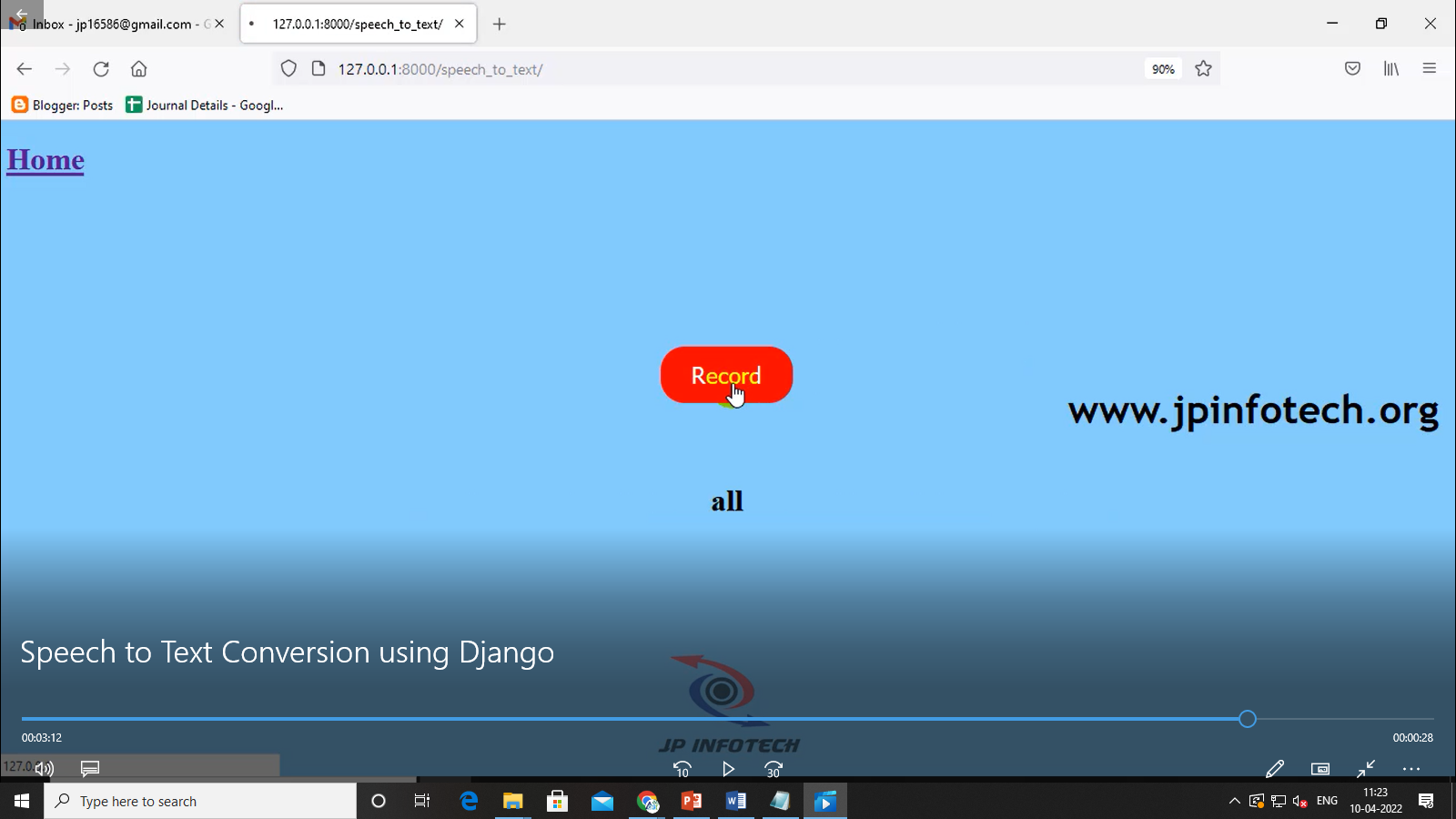
**Home Page**

****



**Speech to Text Page**





**REFERENCES**

[1] MrinaliniKet al: Hindi-English Speech-to-Speech Translation System for Travel Expressions, 2015 International Conference On Computation Of Power, Energy, Information And Communication.

[2] Development and Application of Multilingual Speech Translation Satoshi Nakamura', Spoken Language Communication Research Group Project, National Institute of Information and Communications Technology, Japan.

[3] Speech-to-Speech Translation: A Review, MahakDureja Department of CSE The NorthCap University, Gurgaon SumanlataGautam Department of CSE The NorthCap University, Gurgaon. International Journal of Computer Applications (0975 – 8887) Volume 129 – No.13, November2015.

[4] Sequence-to-Sequence Models for Emphasis Speech Translation. Quoc Truong Do,SkrianiSakti; SakrianiSakti; Satoshi Nakamura, 2018 IEEE/ACM.

[5] Olabe, J. C.; Santos, A.; Martinez, R.; Munoz, E.; Martinez, M.; Quilis, A.; Bernstein, J., “Real time text-to speech conversion system for spanish," Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84. , vol.9, no., pp.85,87, Mar 1984.

[6] Kavaler, R. et al., “A Dynamic Time Warp Integrated Circuit for a 1000-Word Recognition System”, IEEE Journal of Solid-State Circuits, vol SC-22, NO 1, February 1987, pp 3-14.

[7] Aggarwal, R. K. and Dave, M., “Acoustic modeling problem for automatic speech recognition system: advances and refinements (Part II)”, International Journal of Speech Technology (2011) 14:309–320.

[8] Ostendorf, M., Digalakis, V., & Kimball, O. A. (1996). “From HMM’s to segment models: a unified view of stochastic modeling for speech recognition”. IEEE Transactions on Speech and Audio Processing, 4(5), 360– 378.

[9] Yasuhisa Fujii, Y., Yamamoto, K., Nakagawa, S., “AUTOMATIC SPEECH RECOGNITION USING HIDDEN CONDITIONAL NEURAL FIELDS”, ICASSP 2011: P-5036-5039.

[10] Mohamed, A. R., Dahl, G. E., and Hinton, G., “Acoustic Modelling using Deep Belief Networks”, submitted to IEEE TRANS. On audio, speech, and language processing, 2010.