A Project Report on

Learning Management System for visually impaired people using Natural Language Processing

Submitted in partial fulfillment of the requirements
for the award of the degree of

Bachelor of Engineering

in

Information Technology

by

Pranav Patil(15104053)

Neha Patil(16204006)

Akshay Jadhav(16204021)

Gaurang Patankar(16204036)

Under the Guidance of

Prof. Apeksha Mohite

Department of Information Technology
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2018-2019

Approval Sheet

This Project Report entitled ***“Learning Management System for visually impaired people using Natural Language Processing”*** Submitted by ***“Pranav Patil”(15104053) Neha Patil”(16204006)”Akshay Jadhav”(16204021)”Gaurang Patankar”(16204036)*** is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

(Prof. Apeksha Mohite)
Guide

Prof. Kiran Deshpande
Head Deartment of Information Technology

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***“Learning Management System for visually impaired people using Natural Language Processing”*** submitted by ***“Pranav Patil” (15104053) ”Neha Patil”(16204006) ”Akshay Jadhav”(16204021)”Gaurang Patankar”(16204036)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology*** ,to the University of Mumbai,is a bonafide work carried out during academic year 2017-2018.

Mrs.Apeksha Mohite
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Declaration

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

____

(Signature)

(Pranav Patil 15104053)

____

(Signature)

(Neha Patil 16204006)

____

(Signature)

(Akshay Jadhav 16204021)

____

(Signature)

(Gaurang Patankar 16204036)

Date:

Abstract

The objective of this paper is to guide visually impaired with smart system. Internet has become an important tool for learners to acquire information and knowledge that encompasses various elements such as text, graphic, numeric, and animation for their learning process. However, the visually impaired learners have no access at all to these tools nor can it be easily taught to them as they are not able to see the links in the web pages. This paper highlights the usage of popular learning management system (LMS) like Moodle with natural language processing (NLP). NLP gives the capability to access the contents of the website only by inputting the voice command to the system and also it enables the LMS to give proper response based on users voice command. As per a survey, there are various LMS available but, none of them provides the functionality of voice in and voice out. So by implementing this functionality, we can make education easy for the needy.This website is deployed using a translator which has the capability to read out content from various learning materials which is available on our website and it also helps to take notes with our voice to text functionality. Finally we can say that our website will help visually impaired engineering students by speeding up their studies.

Contents

1 Introduction 1

1.1 Introduction About The Entire Topic 2

1.2 Proposed System 2

2 Literature Review 4

3 Working Of the Website 6

3.1 Theorotical Part Of The Website 6

4 NLP Technology 8

4.1 Introduction 8

4.2 History 9

4.3 NLP Modules 10

4.3.1 Syntax 10

4.3.2 Semantics 12

4.4 Advantages 13

4.5 List About Each And Every Libraries Used In Website 15

4.6 Working of Each And Every Libraries 15

5 Diagram 16

5.1 Flow Diagram 16

5.1.1 Student Point of View 16

5.2 Teachers Point of View 17

5.3 Administration Point of View 18

5.4 UML Diagram 19

5.4.1 Activity Diagram 19

5.5 Use Case Diagram 20

5.6 Sequence Diagram 20

6 CODING 23

7 SCREENSHOTS 30

7.1 Login 30

7.2 Admin 31

7.2.1 Admin Change Password 31

7.3 Student 32

7.4 Teachers Dashboard 33

7.5 Teachers News 33

7.6 Subjects . . . 33
7.7 Teacher Resources . . . 33
7.8 Teacher Feedback . . . 34

8 Result 39

9 Conclusion And Future Scope 40
9.1 Conclusion . . . 40
9.2 Future Scope . . . 40

Bibliography 41

10 Appendices 42
10.1 Appendix-A: Python Download Installation . . . 42
10.2 Appendix-B: Installing PIP . . . 42

Publication 45

List of Figures

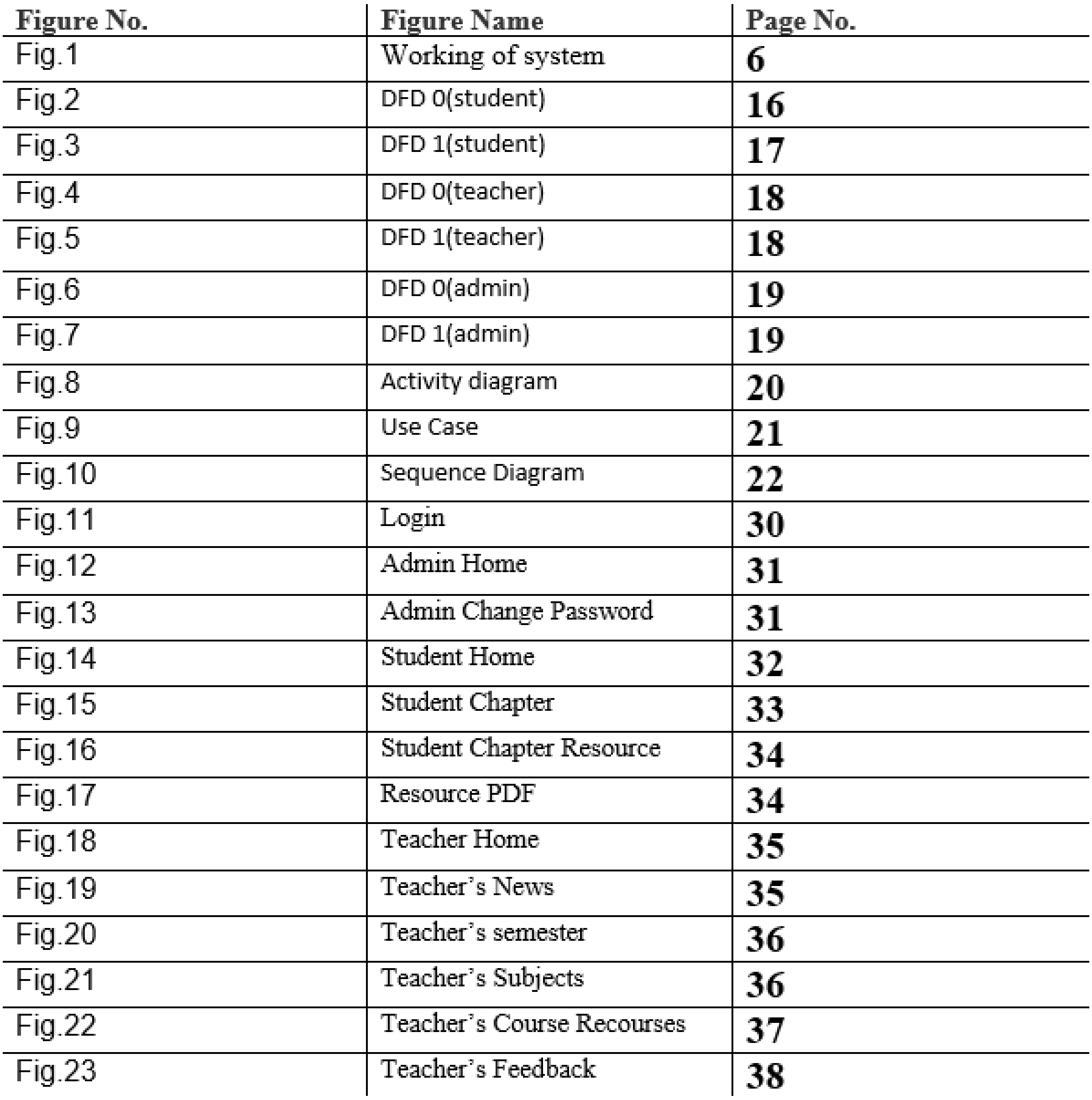


Figure No.	Figure Name	Page No.
Fig.1	Working of system	6
Fig.2	DFD 0(student)	16
Fig.3	DFD 1(student)	17
Fig.4	DFD 0(teacher)	18
Fig.5	DFD 1(teacher)	18
Fig.6	DFD 0(admin)	19
Fig.7	DFD 1(admin)	19
Fig.8	Activity diagram	20
Fig.9	Use Case	21
Fig.10	Sequence Diagram	22
Fig.11	Login	30
Fig.12	Admin Home	31
Fig.13	Admin Change Password	31
Fig.14	Student Home	32
Fig.15	Student Chapter	33
Fig.16	Student Chapter Resource	34
Fig.17	Resource PDF	34
Fig.18	Teacher Home	35
Fig.19	Teacher's News	35
Fig.20	Teacher's semester	36
Fig.21	Teacher's Subjects	36
Fig.22	Teacher's Course Recourses	37
Fig.23	Teacher's Feedback	38

(22).png

List of Abbreviations

NLP:	Natural Language Processing
LMS:	Learning Management System
V/I:	Voice In
V/O:	Voice out
SYS:	System
VIP:	Visually Impaired People
NLPA:	Natural Language Processing Architecture

Chapter 1

Introduction

Contemporary projects aiming at launching learning management systems (LMS) often focus on the introduction of an existing software tool, rather than on an innovation by means of the modern information technologies. In effect, there is almost no original research directed to the complex integration of e-learning systems with the relevant IT such as assistive technologies (dialogue systems, speech recognition and synthesis . . .), knowledge acquisition and knowledge management systems, etc. Among others, the current LMS do not integrate the emerging natural language processing (NLP) applications. The adopted learning management systems often do not support even the basic functionality of a language-oriented search and retrieval of learning objects.

Futhermore, the present-day LMS are not directly linked to the wealth of relevant information and knowledge sources. In the case of the higher education institutions, these sources could comprise standard libraries that provide at least an electronic catalogue of their sources nowadays, local digital libraries that are usually freely available for academics from particular institution, and the access to comprehensive electronic archives or digital libraries that are provided by many publishers and other organizations on a commercial basis. Companies often neglect valuable knowledge sources too. For example, they should consider the integration of their knowledge bases in the form of recorded questions and answers from the call centers. The current e-learning systems do not exploit the potential of available high-level personalization techniques and adaptability of the form, the content and the access to the education. Most of them cannot play the role of a showcase for the modern teaching methods.

The Internet has became an important tool for learners to acquire information and knowledge that encompasses various elements such as text, graphic, numeric, and animation for their learning process. However, the visually impaired learners have no access at all to this tool nor can it be easily taught to them as they are not able to see the links in the web pages. This paper highlights the usage of learning management system through natural language processing that has the capability to access the contents of website and listening to the content of the search only by inputting the voice command to the system. As per the survey there are various E-learning websites available but, none of them provides functionality of voice in - voice out so, by using this functionality we are providing ease of education to the blind user through our website. The website is built with a translator that has the functionality to convert html codes to voice; voice to text again.

1.1 Introduction About The Entire Topic

The basic idea is that we will be creating a plug-in for Moodle where the system will take voice input from user and based on the received input our system will action example checking current subjects through voice command. There is still no system offering easy and intelligent solution for visually impaired but rather a complex solutions. As Moodle is one of the most popularly used open source E-learning website. Hence, we can efficiently use some best features of Moodle in our work.

1.2 Proposed System

There is a growing awareness among parents, teachers, blind youth, and the adult blind community that the education which blind children are receiving is failing them. They are not receiving a quality education which can prepare them to compete in the demanding high tech economy and society of the 21st Century. They are not learning to use and trust the alternative techniques like learning systems which blind persons must adopt with if they are to be successful. They are not developing a positive attitude towards their blindness which is essential to them if they are to become confident, responsible and productive adults. So considering the problem definition above have proposed system which satisfies following objectives:

(1)To make learning easy for the visually impaired with the help of voice in voice out.

(2)Can be used by physically impaired people.

(3)Certain research explained that you gain more knowledge when you listen something than when you read hence our system can help to learn efficiently.

The basic idea is that we will be creating a plug-in for Moodle where the system will take voice input from user and based on the received input our system will action example checking current subjects through voice command. There is still no system offering easy and intelligent solution for visually impaired but rather a complex solutions. As Moodle is one of the most popularly used open source E-learning website. Hence, we can efficiently use some best features of Moodle in our work.
Once the blind person has been entered the eLearning website he/she will tell the system to go to the login page. The system should go to the login page through voice authentication. After entering the login page there should be three types of Login Page

(1) Admin Login

(2) Teacher Login

(3) Student Login

a. Student Login i.e. the Username and password credentials should be done through blind

persons voice authentication. After login, subjects should be displayed based on students department and current year of the student.
After login been done successfully through voice authentication the system should ask the student to choose the subject. After the subject been selected by the student the system will list down the notes available for that particular subject which will be in the pdf format. Once the student has selected a particular topic the system will open that topic which will be in the pdf format and then after opening that pdf the system will read out all the contents available in that pdf. The content should be only in text format.

b. Admin and Teacher Login should be done manually . Admin should have the privileges to add new teacher and new student .Teacher should have the privileges to upload the pdfs i.e. the notes for the blind person.

Chapter 2

Literature Review

[1] Multilanguage natural user interface to database Ruslan Posevkin Chair of Computation Technologies ITMO University:

Natural language user interface is considered to be a specific intelligence system providing a dialogue between the user and the software system within a definite subject field. Intelligence system of natural language user interface consists of: User interface whereby the user inputs the message and gets an answer from the system .Translator of natural language requests to internal language of queries .Translator of internal language units to natural language .The operation cycle of natural language user interfaces starts with the message input in natural language by entering the text. The next step is creating a formal description of the text. All previous analysis results are used to analyses the following queries. It makes it possible to resolve the issues connected with using the same terms in different subject fields. All components of natural language user interface knowledge machine may be classified into translators and analyzers. Translators translate knowledge from one language to another. For example, they translate the descriptive knowledge of some subject field to a natural language text. The analyzers analyze knowledge units and develop previously unknown facts. For example, it can be the analysis of a users question to find an answer.

[2] Construction of a Voice Driven Life Assistant System for Visually Impaired People ;RunzeChen, Zhanhong Tian, Hailun Liu, Fang Zhao, Shuai Zhang, Haobo Liu ;Beijing, 2017

There are many solutions to simplify the way people interacting with computers. The stability of natural language understanding and voice recognition have developed so well that blind person can also has the chance to use the mobile phone easily. To understand the intention of user and extract key information in the sentences spoken, natural language understanding technology should classify the intent and its content so as to extract the entities from the raw sentence .Microsoft has released LUIS [6], a natural language understanding service which can extract the intent and entities from the sentence. An open-source project named Rasa NLU [7] can also provide support to classify the intent and extract the entities. However, Rasa NLU need to be modified to understand Chinese text. A falling detection system has been designed by Wang Rong [8] et al. has provide a solution to detect elder peoples movement. Also, as a risk warning service, the falling detection system can be used to protect blind and visually impaired people and alarm their family when abnormal event happens. Kaiming He [9] created a method to detect objects in images which extends Fast R-CNN named Mask R-CNN [10].The object detection technology can help blind group to know what appears in front of their walking direction well, so we also integrated the Mask R-CNN algorithm into the system to describe the view in front of the blind user.

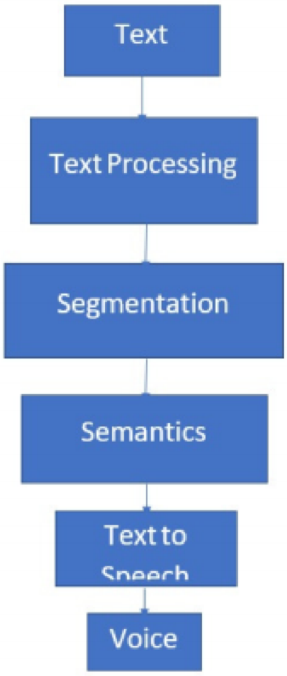
[3] International Journal of Computer Applications (0975 8887) Volume 131 No.11, December2018:

Speech recognition is a vast research field for researchers in modern era. Earlier, the human language was processed by the computer system for speech recognition. Thus, the main objective is to develop recognition system which improves human to human communication by enabling human-machine communication by processing of text or speech. Various applications of speech recognition systems are present and these all includes various research challenges. A6 critical machine learning based review is defined which addresses the various challenging tasks of speech recognition system in NLP. In the existing systems, the recognition rate is very less and the noise ration during the recognition process creates a problem. Thus in this literature review we try to address such kind of challenges and provides a solution to work further in future ..

Chapter 3

Working Of the Website

As the projects core is based on the use of natural language processing(NLP), the main focus will be on Text to Speech Speech to Text. The above architecture diagram start from the Text generated by the text-to-speech module, which is further send for Text Processing.Further Segmentation is performed on the processed text to divide sentences.Semantics is used for grammer utilization for converstion into speech.Finally Speech is the output, the user will receive



3.1 Theorotical Part Of The Website

Once the blind person has been entered the eLearning website he/she will tell the system to go to the login page. The system should go to the login page through voice authentication. After entering the login page there should be three types of Login Page

(1) Student Login.
(2) Admin Login
(3) Teacher Login

a. Student Login i.e. the Username and password credentials should be done through blind persons voice authentication. After login, subjects should be displayed based on students department and current year of the student.
After login been done successfully through voice authentication the system should ask the student to choose the subject. After the subject been selected by the student the system will list down the notes available for that particular subject which will be in the pdf format. Once the student has selected a particular topic the system will open that topic which will be in the pdf format and then after opening that pdf the system will read out all the contents available in that pdf. The content should be only in text format. There is feedback option for the students to give feedbacks to the teachers .

a. Admin login: Admin should have the privileges to add new teacher and new student .Teacher should have the privileges to upload the pdfs i.e. the notes for the blind person . Admin also change passwords.
b. Teacher Login: Teachers can login manually. Teachers can select various courses, semesters and subjects. As one teacher teaches multiple semesters and subjects. Teacher can view the news and also has privilege to add or remove news. Teachers can also add or remove course materials. Teachers can also view and read feedbacks from the students.

Chapter 4

NLP Technology

Natural Language Processing is a technique where machine can become more human and thereby reducing the distance between human being and the machine can be reduced. Therefore in simple sense NLP makes human to communicate with the machine easily. There are many applications developed in past few decades in NLP. Most of these are very useful in everyday life for example a machine that takes instructions by voice. There are lots of research groups working on this topic to develop more practical are useful systems. Natural Language Processing holds great promise for making computer interfaces that are easier to use for people, since people will hopefully be able to talk to the computer in their own language, rather than learn a specialized language of computer commands. For programming, however, the necessity of a formal programming language for communicating with a computer has always been taken for granted. We would like to challenge this assumption. We believe that modern Natural Language Processing techniques can make possible the use of natural language to express programming ideas, thus drastically increasing the accessibility of programming to non-expert users. To demonstrate the feasibility of Natural Language Programming, this paper tackles what are perceived to be some of the hardest cases: steps and loops.

4.1 Introduction

Natural language processing (NLP) is a major area of artificial intelligence research, which in its turn serves as a field of application and interaction of a number of other traditional AI areas. Until recently, the focus in AI applications in NLP was on knowledge representation, logical reasoning, and constraint satisfaction - first applied to semantics and later to the grammar. In the last decade, a dramatic shift in the NLP research has led to the prevalence of very large scale applications of statistical methods, such as machine learning and data mining. Naturally, this also opened the way to the learning and optimization methods that constitute the core of modern AI, most notably genetic algorithms and neural networks. In this paper we give an overview of the current trends in NLP and discuss the possible applications of traditional AI techniques and their combination in this fascinating area.

4.2 History

The history of natural language processing generally started in the 1950s, although work can be found from earlier periods. In 1950, Alan Turing published an article titled "Intelligence" which proposed what is now called the Turing test as a criterion of intelligence. The Georgetown experiment in 1954 involved fully automatic translation of more than sixty Russian sentences into English. The authors claimed that within three or five years, machine translation would be a solved problem. However, real progress was much slower, and after the ALPAC report in 1966, which found that ten-year-long research had failed to fulfill the expectations, funding for machine translation was dramatically reduced. Little further research in machine translation was conducted until the late 1980s, when the first statistical machine translation systems were developed. Some notably successful natural language processing systems developed in the 1960s were SHRDLU, a natural language system working in restricted "blocks worlds" with restricted vocabularies, and ELIZA, a simulation of a Rogerian psychotherapist, written by Joseph Weizenbaum between 1964 and 1966. Using almost no information about human thought or emotion, ELIZA sometimes provided a startlingly human-like interaction. When the "patient" exceeded the very small knowledge base, ELIZA might provide a generic response, for example, responding to "My head hurts" with "Why do you say your head hurts?". During the 1970s, many programmers began to write "conceptual ontologies", which structured real-world information into computer-understandable data. Examples are MARGIE (Schank, 1975), SAM (Cullingford, 1978), PAM (Wilensky, 1978), TaleSpin (Meehan, 1976), QUALM (Lehnert, 1977), Politics (Carbonell, 1979), and Plot Units (Lehnert 1981). During this time, many chatterbots were written including PARRY, Racter, and Jabberwacky. An automated online assistant providing customer service on a web page, an example of an application where natural language processing is a major component.

Up to the 1980s,most natural language processing systems were based on complex sets of hand-written rules. Starting in the late 1980s, however, there was a revolution in natural language processing with the introduction of machine learning algorithms for language processing. This was due to both the steady increase in computational power (see Moore's law) and the gradual lessening of the dominance of Chomskyan theories of linguistics (e.g. transformational grammar), whose theoretical underpinnings discouraged the sort of corpus linguistics that underlies the machine-learning approach to language processing. Some of the earliest-used machine learning algorithms, such as decision trees, produced systems of hard if-then rules similar to existing hand-written rules. However, part-of-speech tagging introduced the use of hidden Markov models to natural language processing, and increasingly, research has focused on statistical models, which make soft, probabilistic decisions based on attaching real-valued weights to the features making up the input data. The cache language models upon which many speech recognitionsystems now rely are examples of such statistical models. Such models are generally more robust when given unfamiliar input, especially input that contains errors (as is very common for real-world data), and produce more reliable results when integrated into a larger system comprising multiple subtasks.

Many of the notable early successes occurred in the field of machine translation, due especially to work at IBM Research, where successively more complicated statistical models were developed. These systems were able to take advantage of existing multilingual textual corpora that had been produced by the Parliament of Canada and the European Union as a result of laws calling for the translation of all governmental proceedings into all official languages of the corresponding systems of government. However, most other systems depended on corpora specifically developed for the tasks implemented by these systems, which was (and often continues to be) a major limitation in the success of these systems. As a result, a great deal of research has gone into methods of more effectively learning from limited amounts of data. Recent research has increasingly focused on unsupervised and semi-supervised learning algorithms. Such algorithms are able to learn from data that has not been hand-annotated with the desired answers, or using a combination of annotated and non-annotated data. Generally, this task is much more dificult than supervised learning, and typically produces less accurate results for a given amount of input data. However, there is an enormous amount of non-annotated data available (including, among other things, the entire content of the World Wide Web), which can often make up for the inferior results if the algorithm used has a low enough time complexity to be practical.

In the 2010s, representation learning and deep neural network-style machine learning methods became widespread in natural language processing, due in part to a flurry of results showing that such techniques[4][5] can achieve state-of-the-art results in many natural language tasks, for example in language modeling,[6] parsing,[7][8] and many others. Popular techniques include the use of word embeddings to capture semantic properties of words, and an increase in end-to-end learning of a higher -level task (e.g., question answering) instead of relying on a pipeline of separate intermediate tasks (e.g., part-of-speech tagging and dependency parsing). In some areas, this shift has entailed substantial changes in how NLP systems are designed, such that deep neural network-based approaches may be viewed as a new paradigm distinct from statistical natural language processing. For instance, the term neural machine translation(NMT) emphasizes the fact that deep learning-based approaches to machine translation directly learn sequenceto-sequence transformations, obviating the need for intermediate steps such as word alignment and language modeling that were used in statistical machine translation (SMT).

4.3 NLP Modules

4.3.1 Syntax

1]Grammar induction
Generate a formal grammar that describes a language's syntax.

2]Lemmatization
The task of removing inflectional endings only and to return the base dictionary form of a word which is also known as a lemma.

3]Morphological segmentation
Separate words into individual morphemes and identify the class of the morphemes. The difficulty of this task depends greatly on the complexity of the morphology (i.e. the structure of words) of the language being considered. English has fairly simple morphology, especially inflectional morphology, and thus it is often possible to ignore this task entirely and simply model all possible forms of a word (e.g. "open, opens, opened, opening") as separate words.

In languages such as Turkish or Meitei,[14] a highly agglutinated Indian language, however,such an approach is not possible, as each dictionary entry has thousands of possible word forms.

4]Part-of-speech tagging
Given a sentence, determine the part of speech for each word. Many words, especially common ones, can serve as multiple parts of speech. For example, "book" can be a noun ("the book on the table") or verb ("to book a flight"); "set" can be a noun, verb or adjective; and "out" can be any of at least five different parts of speech. Some languages have more such ambiguity than others. Languages with little inflectional morphology, such as English,are particularly prone to such ambiguity. Chinese is prone to such ambiguity because it is a tonal language during verbalization. Such inflection is not readily conveyed via the entities employed within the orthography to convey intended meaning.

5]Parsing
Major evaluations and tasks Syntax (see also: Stochastic grammar) Determine the parse tree (grammatical analysis) of a given sentence. The grammar for natural languages is ambiguous and typical sentences have multiple possible analyses. In fact, perhaps surprisingly, for a typical sentence there may be thousands of potential parses (most of which will seem completely nonsensical to a human).
There are two primary types of parsing, Dependency Parsing and Constituency Parsing. Dependency Parsing focuses on the relationships between words in a sentence (marking things like Primary Objects and predicates), whereas Constituency Parsing focuses on building out the Parse Tree using a Probabilistic Context-Free Grammar (PCFG).

6]Sentence breaking (also known as sentence boundary disambiguation)
Given a chunk of text, find the sentence boundaries. Sentence boundaries are often marked by periods or other punctuation marks, but these same characters can serve other purposes(e.g. marking abbreviations).

7]Stemming
The process of reducing inflected (or sometimes derived) words to their root form. (e.g."close" will be the root for "closed", "closing", "close", "closer" etc).

8]Word segmentation
Separate a chunk of continuous text into separate words. For a language like English, this is fairly trivial, since words are usually separated by spaces. However, some written languages like Chinese, Japanese and Thai do not mark word boundaries in such a fashion, and in those languages text segmentation is a significant task requiring knowledge of the vocabulary and morphology of words in the language. Sometimes this process is also used in cases like Bag of Words (BOW) creation in data mining.

9]Terminology extraction
The goal of terminology extraction is to automatically extract relevant terms from a given corpus.

4.3.2 Semantics

1]Lexical semantics
What is the computational meaning of individual words in context? Machine translation Automatically translate text from one human language to another. This is one of the most difficult problems, and is a member of a class of problems colloquially termed "AI-complete", i.e. requiring all of the different types of knowledge that humans possess (grammar,semantics, facts about the real world, etc.) in order to solve properly.

2]Named entity recognition (NER)
Given a stream of text, determine which items in the text map to proper names, such as people or places, and what the type of each such name is (e.g. person, location,organization). Note that, although capitalization can aid in recognizing named entities in languages such as English, this information cannot aid in determining the type of named entity, and in any case is often inaccurate or insufficient. For example, the first letter of a sentence is also capitalized, and named entities often span several words, only some of which are capitalized. Furthermore, many other languages in non-Western scripts (e.g. Chinese or Arabic) do not have any capitalization at all, and even languages with capitalization may not consistently use it to distinguish names. For example, German capitalizes all nouns, regardless of whether they are names, and French and Spanish do not capitalize names that serve as adjectives.

3]Natural language generation
Convert information from computer databases or semantic intents into readable human language.

4]Natural language understanding
Convert chunks of text into more formal representations such as first-order logic structures that are easier for computer programs to manipulate. Natural language understanding involves the identification of the intended semantic from the multiple possible semantics which can be derived from a natural language expression which usually takes the form of organized notations of natural language concepts.

5]Introduction and creation of language
metamodel and ontology are efficient however empirical solutions. An explicit formalization Semantics of natural language semantics without confusions with implicit assumptions such as closedworld assumption (CWA) vs. open-world assumption, or subjective Yes/No vs. objective True/False is expected for the construction of a basis of semantics formalization.

6]Optical character recognition (OCR)
Given an image representing printed text, determine the corresponding text.

7]Question answering
Given a human-language question, determine its answer. Typical questions have a specific right answer (such as ”What is the capital of Canada?”), but sometimes open-ended questions are also considered (such as ”What is the meaning of life?”). Recent works have looked at even more complex questions.

8]Recognizing Textual entailment
Given two text fragments, determine if one being true entails the other, entails the other's negation, or allows the other to be either true or false.

9]Relationship extraction
Given a chunk of text, identify the relationships among named entities (e.g. who is married to whom).

10]Sentiment analysis (see also multimodal sentiment analysis)
Extract subjective information usually from a set of documents, often using online reviews to determine ”polarity” about specific objects. It is especially useful for identifying trends of public opinion in the social media, for the purpose of marketing.

11]Topic segmentation and recognition
Given a chunk of text, separate it into segments each of which is devoted to a topic, and identify the topic of the segment.

12]Word sense disambiguation
Many words have more than one meaning; we have to select the meaning which makes the most sense in context. For this problem, we are typically given a list of words and associated word senses, e.g. from a dictionary or from an online resource such as WordNet.

4.4 Advantages

1. The long term perspective:
a. One of the top goals of AI is to build intelligent machines that can understand the world, speak about what they perceive and do, and explain it in a way people can understand - NLP is a critical piece in that.
b. Another related top goal of AI is to achieve true human-machine interaction -machines

we can talk to, teach, and trust the same way we do with humans. NLP is critical here as well.
c. Finally, solving NLP can unwrap some of the big mysteries around language - how does language work? What do words mean, and how do we put words into sentences? How does the brain link language to perception and action? How do we learn new languages?

2. The mid-term perspective: advances in NLP enables a wide range of useful applications:
a. Language understanding - that's a bit broad and vague, and encapsulates multiple areas such as word embeddings, language models, parsing, But at the bottom line think of a chat bot / conversational digital Assistant that can actually have a conversation - understanding semantics, intents, nuances and sentiments (sarcasm, humor, negativity, compliments,)
b. Machine translation -
i. be able to translate complete books, capturing the unique color of the original language, and recreating it in the target language.
ii. Live translation - two or more people having a conversation in multiple different languages but hearing each other in their own language (through ear buds).
c. Sentiment analysis And opinion mining -
i. analysing press coverage and social media to understand sentiments and trends about brands, stocks, celebrities, politicians, celebrities, media, I.e being able to understand complex content that references multiple entities, and uses sarcasm, humor, and context.
ii. Use sentiment analysis to detect personal and interpersonal well-being. Is a person stressed / anxious / angry / depressed How healthy are the conversations between you and your spouse? What about your teenage kids, and family as a who
d. Questions answering -
i. understand complex human questions, be able to identify a credible source for the answer, and figure out the answer. Imagine being able to answer any question about any hotel, restaurant, resort, landmark in the world - ”I am looking for a ski vacation for 6 people. We want a 4 or 5 star hotel, in walking distance to blue slopes. We want high guarantee for snow in late March”. Be able to answer complex questions from your email, photos, documents, Excel sheets, chat messages, location history, etc.
ii. Given some text (and non text) corpus - be able to formalise it as QA
e. Image captioning (in collaborating with visual recognition) - be able to accurately describe the details and meta scene of any visual input. Be able to answer complex QA about it.
i. Be able to find any photo / scene - ”my photo with my son in France when he was 3”, ”the famous dance scene in pulp fiction”, ”the scene from that movie where the guy takes something that looks like a diamond out of his nose”
f. Paraphrasing - accurately summarising articles, papers, books, movies, conversations, meetings, email and social threads, documents, evolving topics, etc. While accurately capturing the most important points, sentiments, opinions, color, etc
g. Content generation - given some instructions / goals / context - generate content such as a story, poem, joke, computer program, article, instructions, documentation, Tweet, play,

4.5 List About Each And Every Libraries Used In Website

i. P5-speech.js
ii. P5-sound.js
iii.Flask
iv. Sqlite3
v. W3-CSS

4.6 Working of Each And Every Libraries

a. P5-speech.js: p5.speech is a JavaScript library that provides simple, clear access to the Web Speech and Speech Recognition APIs, allowing for the easy creation of sketches that can talk and listen.

b. P5-sound.js: p5.sound brings the Processing approach to Web Audio as an addon for p5.js. Functionality includes audio input, playback, manipulation, effects, recording, sequencing, analysis and synthesis. The library is designed to be used in tandem with p5.js.

c. Flask: Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

d. Sqlite3: SQLite is an open-source, embedded, relational database management system, designed circa 2000. It is a lightweight database, with zero configuration, no requirements of a server or installation. Despite its simplicity, it is laden with popular features of database management systems.

e. W3-CSS: W3.CSS is a modern CSS framework with built-in responsiveness. It supports responsive mobile first design by default, and it is smaller and faster than similar CSS frameworks.

Chapter 5

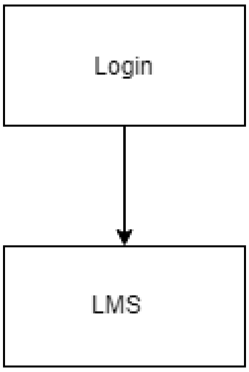
Diagram

5.1 Flow Diagram

A Data Flow Diagram (DFD) is a graphical representation of the flow of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated.

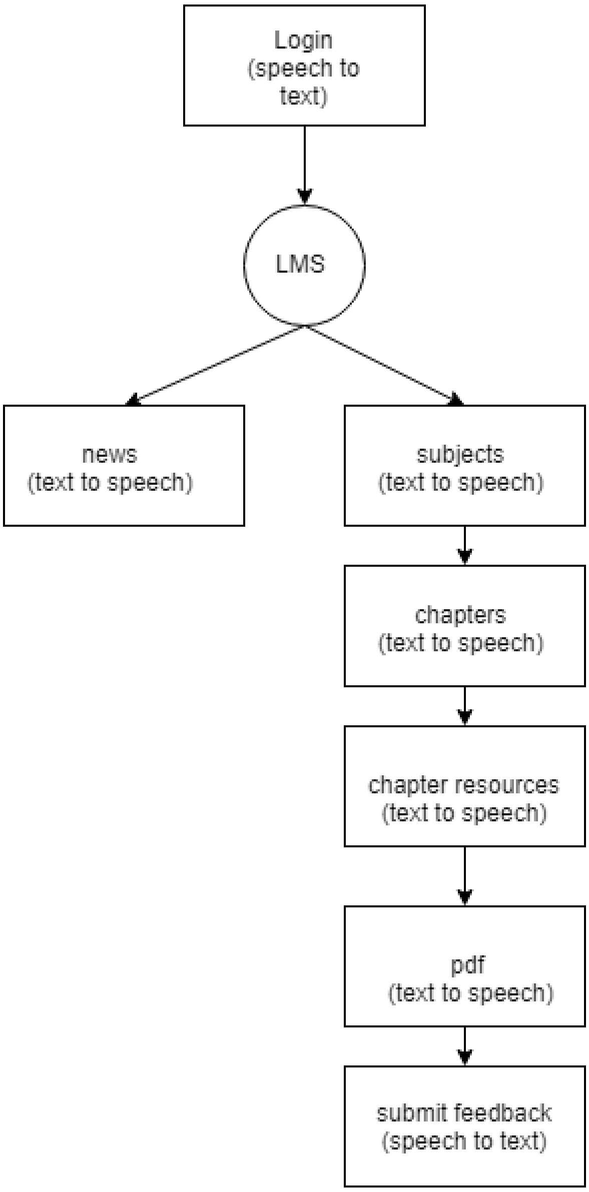
5.1.1 Student Point of View

DFD LEVEL 0:



DFD LEVEL 1:

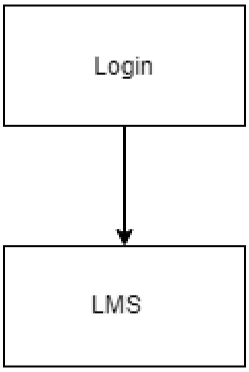
Once the students gets log in into the system successfully through voice authentication. The system will ask the student to select whether he wants to hear news or want subjects. The student needs to speak in the microphone news or subjects to select one thing. If Student speaks News then system responds back through voice the list of news in the news section. The student hears all the list of news and speaks back the news he/she wants to hear. Similarly, if the student says subjects then list of subjects appears the systems again does text to speech and says out all the subjects presented in the particular course, the student simply needs to speak the particular subject he/she wants to hear and study. As the student speaks and selects particular subject , all chapters available for that subjects appears. The system says out all the chapter the teachers have uploaded, accordingly the student can speak out and select the chapter he/she wants to study. After selecting the chapter , resource in the form of PDF gets converted in form of text and then into speech, the system reads out the entire text so that the student listen to the content carefully and study.



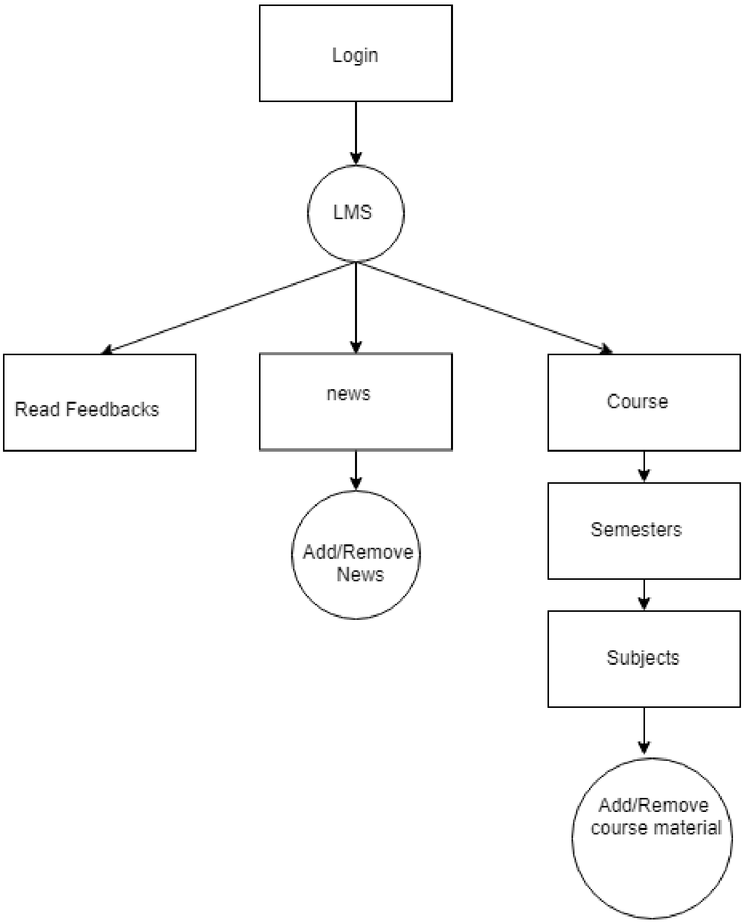
5.2 Teachers Point of View

DFD LEVEL 0: Teachers can manually enter their username and password to get logged in onto the system

DFD LEVEL 1: Once the teacher successfully logs in into the system, the teacher dashboard gets displayed. There are three things teacher can do i.e add/ remove news

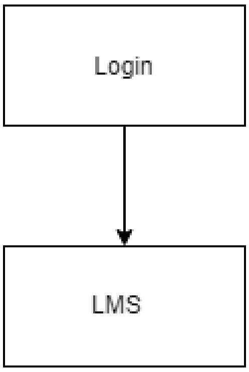


, Read feedbacks from students and select course , semester and subjects to add/remove course resources .Teachers teaches many semester so teacher can select any semester simply by clicking any of the one semester

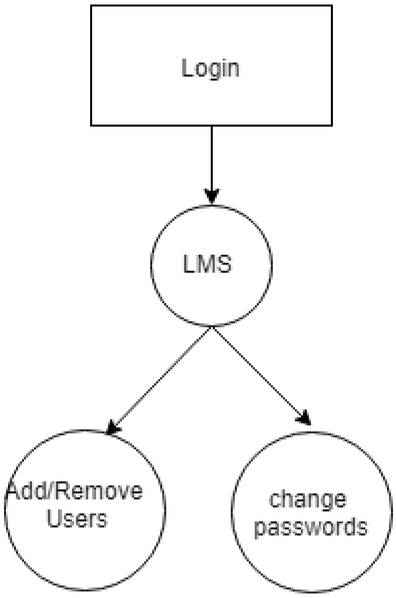


5.3 Administration Point of View

DFD LEVEL 0: Admin can login into the system simply by manually entering username and password



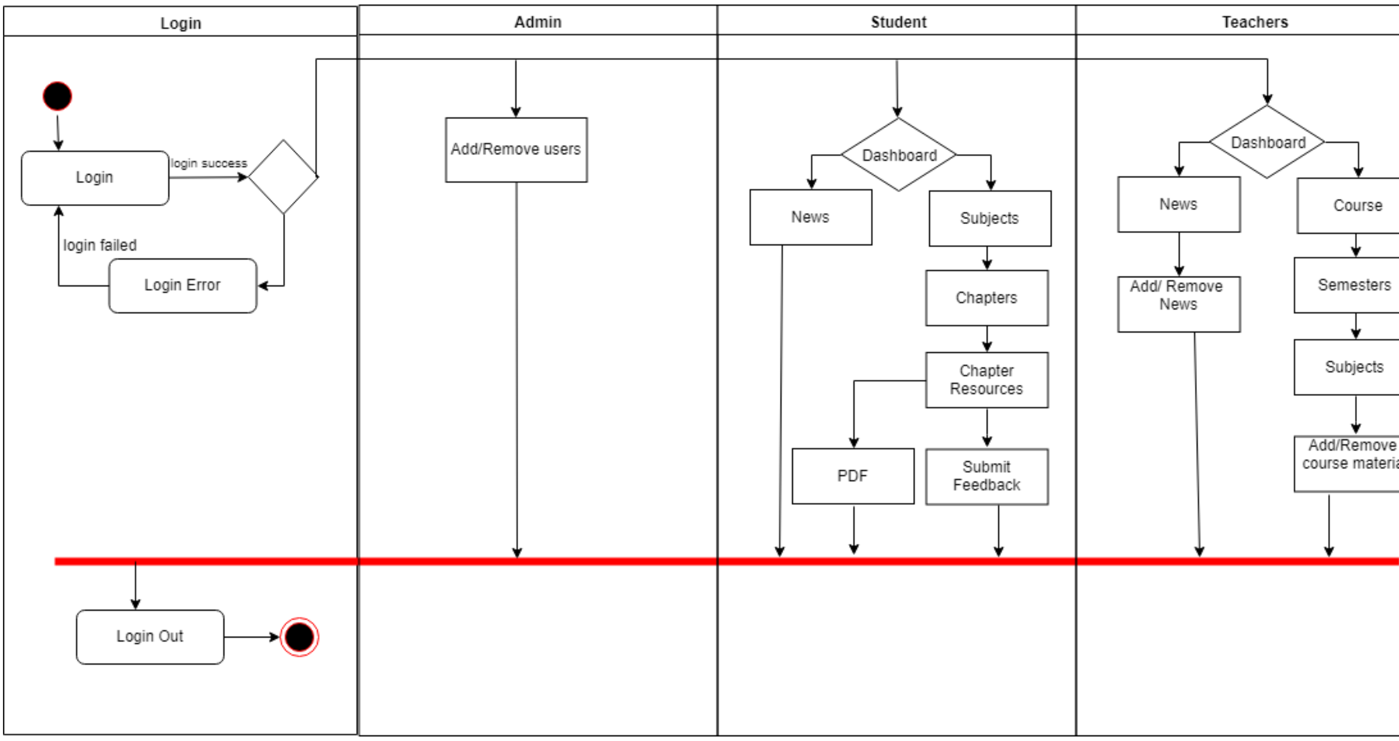
DFD LEVEL 1: As admin logs in into the system successfully the Admin dashboard appears. Admin can add and remove students as well as can add and remove teachers. Admin can also change passwords of the user. While admin adds new students, the admin selects course and semester for the students.



5.4 UML Diagram

5.4.1 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.
In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational process. The below activity diagram explains each activity of the individual user(admin or student or teacher) .

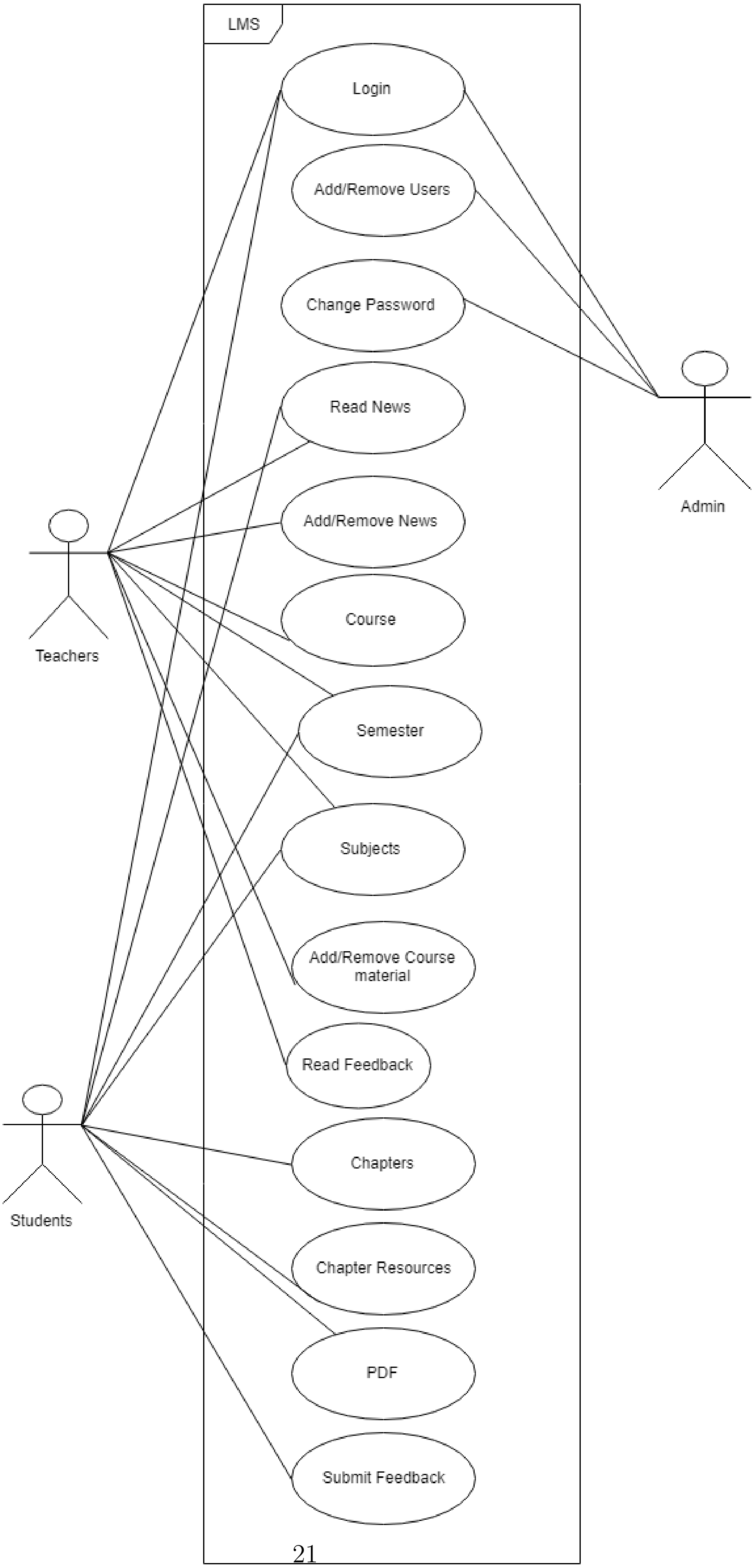


5.5 Use Case Diagram

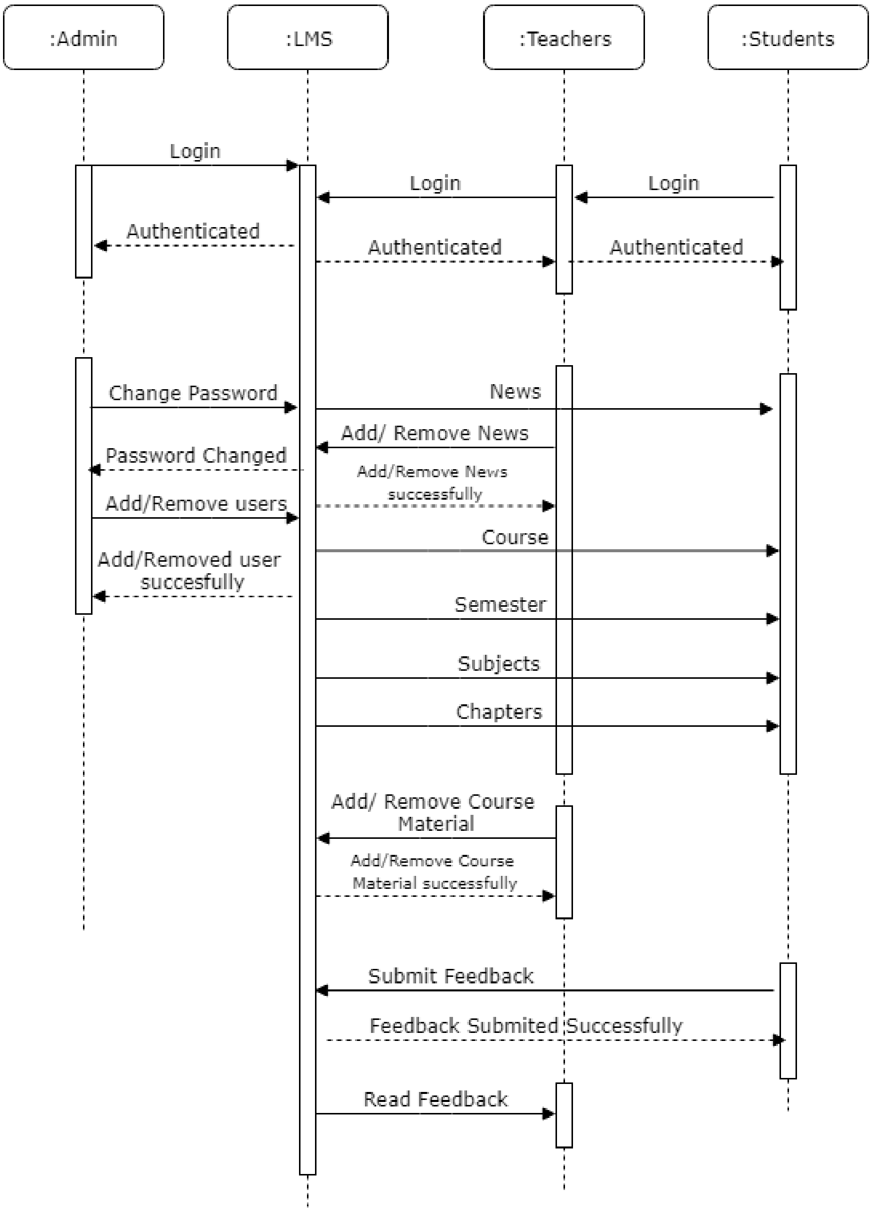
A Use Case diagram is a type of behavioral diagram defined by the UML created from a use case analysis. Its purpose is to prevent a graphical overview of the functionality provided by a system in terms of actors, and their goal represented as use case. It is a type of diagram that shows a set of use cases, actors and their relationship. It should have a distinct name. It commonly contains: 1. Use Cases 2. Actors (Primary and Secondary)
The below use case diagram consists of three actors i.e student, admin and teacher. There are several use cases to explain the static view of each actor

5.6 Sequence Diagram

Sequence diagram is an interaction diagram that emphasizes the time ordering of messages. A sequence diagram is a structured representation of behaviour as a series of sequential steps over time. It is used primarily to show the interactions between objects in the sequential order. The sequence diagram is also called as Message Sequence Chart.
The sequence diagram below explains detail work flow of the entire system.


LMS
Login
Add/Remove Users
Change Password
Read News
Add/Remove News
Course
Semester
Subjects
Add/Remove Course material
Read Feedback
Chapters
Chapter Resources
PDF
Submit Feedback
Teachers
Students
Admin

case dia.png


:Admin
:LMS
:Teachers
:Students
Login
Login
Login
Authenticated
Authenticated
Authenticated
Change Password
News
Add/ Remove News
Password Changed
Add/Remove News successfully
Add/Remove users
Course
Add/Removed user succesfully
Semester
Subjects
Chapters
Add/ Remove Course Material
Add/Remove Course Material successfully
Submit Feedback
Feedback Submited Successfully
Read Feedback

dia.png

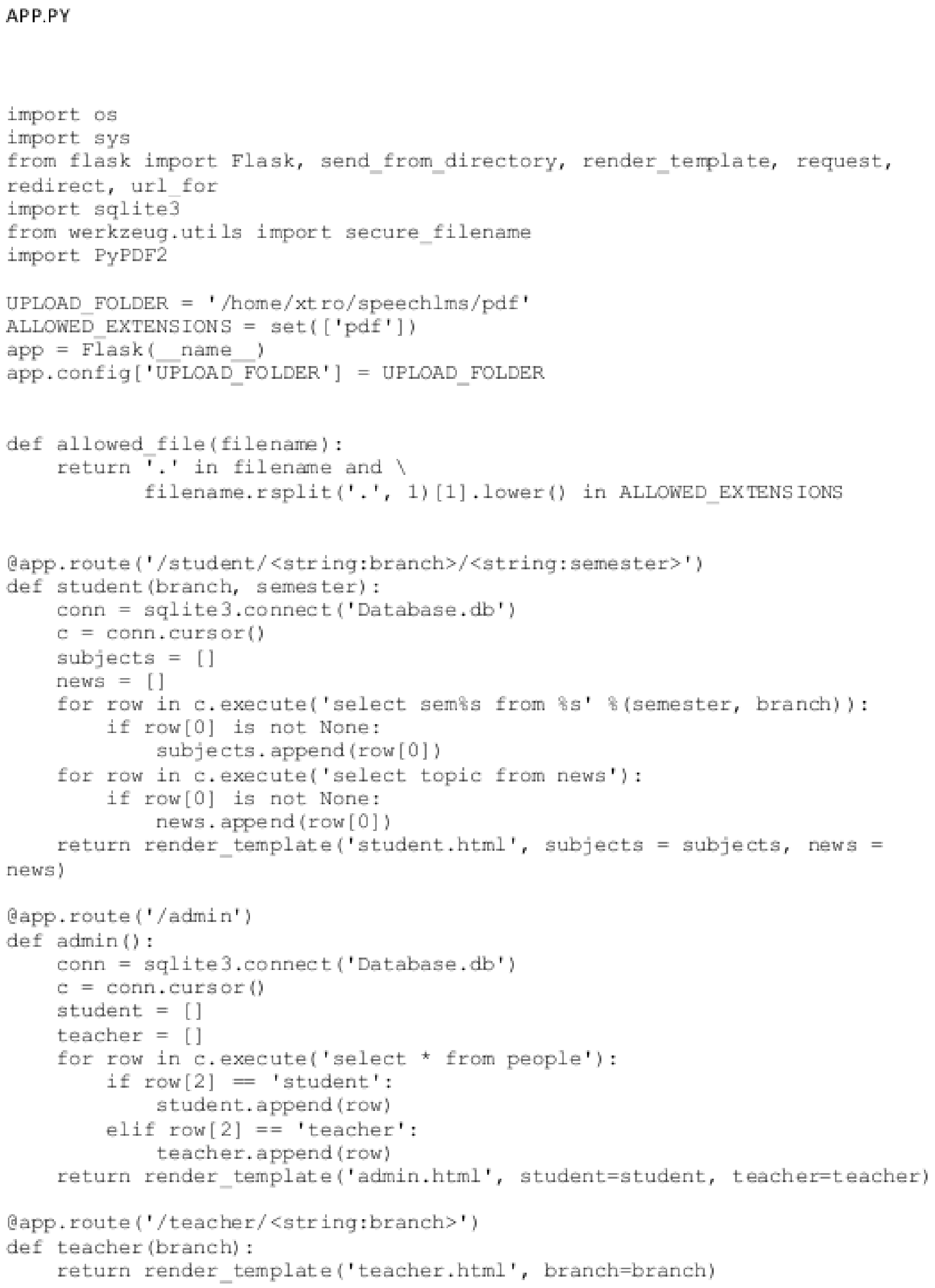
Chapter 6

CODING

CODE :

LOGIN.JS

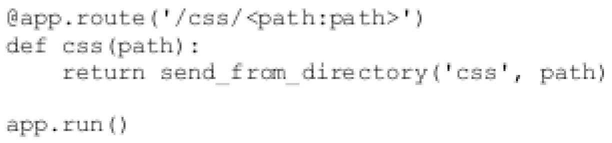
```
function checkFocus() {
  $("#username").focus(function() {
    name = "username";
    speech = new p5.Speech();
    speech.setVoice("Google हिन्दी");
    speech.speak(name);
    document.body.onkeyup = function(e) {
      if (e.keyCode == 17) {
        let speechRec = new p5.SpeechRec("Google हिन्दी", gotSpeech);
        speechRec.start();
        console.log(speechRec);
        function gotSpeech() {
          if (speechRec.resultValue) {
            ans = speechRec.resultString;
            ans = ans.replace(/ /g, "");
            document.getElementById("username").value = ans;
          }
        }
      }
    };
  });

  $("#password").focus(function() {
    name = "password";
    speech = new p5.Speech();
    console.log("run");
    speech.setVoice("Google हिन्दी");
    speech.speak(name);
    document.body.onkeyup = function(e) {
      if (e.keyCode == 17) {
        let speechRec = new p5.SpeechRec("Google हिन्दी", gotSpeech);
        speechRec.start();
        console.log(speechRec);
        function gotSpeech() {
          if (speechRec.resultValue) {
            ans = speechRec.resultString;
            ans = ans.replace(/ /g, "");
            document.getElementById("password").value = ans;
          }
        }
      }
    };
  });
}

window.onload = checkFocus;
```



APP.PY

```
import os
import sys
from flask import Flask, send_from_directory, render_template, request,
redirect, url_for
import sqlite3
from werkzeug.utils import secure_filename
import PyPDF2

UPLOAD_FOLDER = '/home/xtro/speechlms/pdf'
ALLOWED_EXTENSIONS = set(['pdf'])
app = Flask(__name__)
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER


def allowed_file(filename):
    return '.' in filename and \
           filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS


@app.route('/student/<string:branch>/<string:semester>')
def student(branch, semester):
    conn = sqlite3.connect('Database.db')
    c = conn.cursor()
    subjects = []
    news = []
    for row in c.execute('select sem%s from %s' %(semester, branch)):
        if row[0] is not None:
            subjects.append(row[0])
    for row in c.execute('select topic from news'):
        if row[0] is not None:
            news.append(row[0])
    return render_template('student.html', subjects = subjects, news =
news)

@app.route('/admin')
def admin():
    conn = sqlite3.connect('Database.db')
    c = conn.cursor()
    student = []
    teacher = []
    for row in c.execute('select * from people'):
        if row[2] == 'student':
            student.append(row)
        elif row[2] == 'teacher':
            teacher.append(row)
    return render_template('admin.html', student=student, teacher=teacher)

@app.route('/teacher/<string:branch>')
def teacher(branch):
    return render_template('teacher.html', branch=branch)
```

```
@app.route('/course/<string:branch>')
def course(branch):
    return render_template('branch.html', branch=branch)

@app.route('/tsubjects/<string:branch>/<string:semester>')
def tsubjects(branch, semester):
    conn = sqlite3.connect('Database.db')
    c = conn.cursor()
    sub = []
    for row in c.execute('select %s from %s' %(semester, branch)):
        if row[0] is not None:
            sub.append(row[0])
    return render_template('tsubjects.html', subject=sub)

@app.route('/delpdf/<string:name>/<string:subject>')
def delpdf(name, subject):
    conn = sqlite3.connect('Database.db')
    c = conn.cursor()
    c.execute("delete from %s where name = ?" %subject, (name,))
    conn.commit()
    return redirect(url_for('addres', subject=subject))

@app.route('/addres/<string:subject>', methods=['GET', 'POST'])
def addres(subject):
    print('Head', file=sys.stdout)
    conn = sqlite3.connect("Database.db")
    c = conn.cursor()
    if request.method == 'POST':
        if 'file' not in request.files:
            return redirect(request.url)
        file = request.files['file']
        name = request.form.get('name')
        addr = file.filename
        data = (name, addr)
        if file.filename == '':
            return redirect(request.url)
        if file and allowed_file(file.filename):
            filename = secure_filename(file.filename)
            file.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))
            c.execute(' insert into %s values(?,?)' %subject, data)
            conn.commit()
            return redirect(url_for('addres',
                                    message="Uploaded", subject=subject))
    data = []
    for row in c.execute('select * from %s' %subject):
        data.append(row)
    return render_template('addres.html', data=data, subject=subject)

@app.route('/news', methods=['GET', 'POST'])
def news():
    conn = sqlite3.connect('Database.db')
    c = conn.cursor()
    if request.method == 'POST':
        news = request.form.get('news')
        c.execute('insert into news values (?)', (news, ))
        conn.commit()
        cnews = []
        for row in c.execute('select * from news'):
            cnews.append(row[0])
        return render_template('news.html', addmessage = "News Submitted",
news=cnews)
    cnews = []
    for row in c.execute('select * from news'):
        cnews.append(row[0])
    return render_template('news.html', news=cnews)

@app.route('/delnews/<string:news>')
def delnews(news):
    conn = sqlite3.connect('Database.db')
    c = conn.cursor()
    c.execute("delete from news where topic = ?" , (news,))
    conn.commit()
    return redirect(url_for('news'))

@app.route("/subjects/<string:subject>")
def subjects(subject):
    conn = sqlite3.connect('Database.db')
    c = conn.cursor()
    data = []
    say = []
    for row in c.execute('select * from %s' %subject):
        data.append(row)
        say.append(row[0])
    return render_template('subject.html', data=data, say=say)

@app.route("/pdf/<path:path>")
def pdf(path):
    pdfFileObj = open('./pdf/%s' %path, 'rb')
    pdfReader = PyPDF2.PdfFileReader(pdfFileObj)
    n = pdfReader.numPages
    text = ""
    for i in range(n):
        text = text + (pdfReader.getPage(i).extractText())
    pdfFileObj.close()
    return render_template('pdf.html', text=text)

@app.route("/js/<path:path>")
def js(path):
    return send_from_directory('js', path)

@app.route('/', methods=['GET', 'POST'])
def login():
    if request.method == "POST":
        conn = sqlite3.connect('Database.db')
        c = conn.cursor()
        username = request.form.get("username")
        password = request.form.get("password")
        u = 0
        for row in c.execute('select username from PEOPLE'):
            if row[0] == username:
                u = 1
                break
        if u == 1:
            ur = (username,)
            c.execute('select password from PEOPLE where username = ?',
ur)
            pa = c.fetchone()
            if password == pa[0]:
                c.execute('select type from PEOPLE where username =?', ur)
                ty = c.fetchone()
                if ty[0] == 'student':
                    c.execute('select branch from people where username =
?', ur)
                    bra = c.fetchone()
                    c.execute('select semester from people where username
= ?', ur)
                    sem = c.fetchone()
                    return redirect(url_for('student',branch=bra[0],
semester=sem[0]))
                elif ty[0] == 'admin':
                    return redirect(url_for('admin'))
                else:
                    c.execute('select branch from people where username =
?', ur)
                    bra = c.fetchone()
                    return redirect(url_for('teacher', branch=bra[0]))
            else:
                return render_template('login.html', message="Wrong
username or password")
        else:
            return render_template('login.html', message="Wrong username
or password")
    return render_template('login.html', message="")

@app.route('/register', methods=['GET', 'POST'])
def register():
    if request.method == 'POST':
        conn = sqlite3.connect('Database.db')
        c = conn.cursor()
        u = request.form.get('username')
        p = request.form.get('password')
        t = request.form.get('type')
        b = request.form.get('branch')
        s = request.form.get('semester')
        s = str(s)
        data = (u,p,t,b,s)
        c.execute('INSERT INTO PEOPLE VALUES (?,?,?,?,?)', data)
        conn.commit()
        return render_template('register.html', message = "Successfully
Registered")
    return render_template('register.html')
```



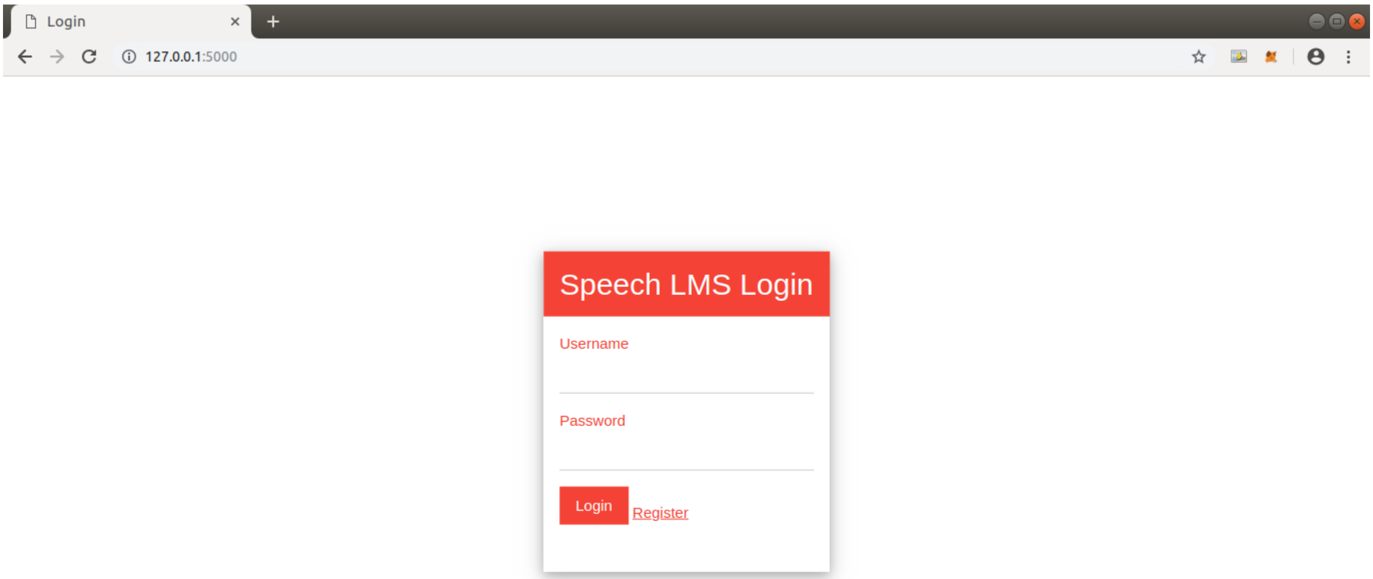
```
@app.route('/css/<path:path>')
def css(path):
    return send_from_directory('css', path)

app.run()
```

Chapter 7

SCREENSHOTS

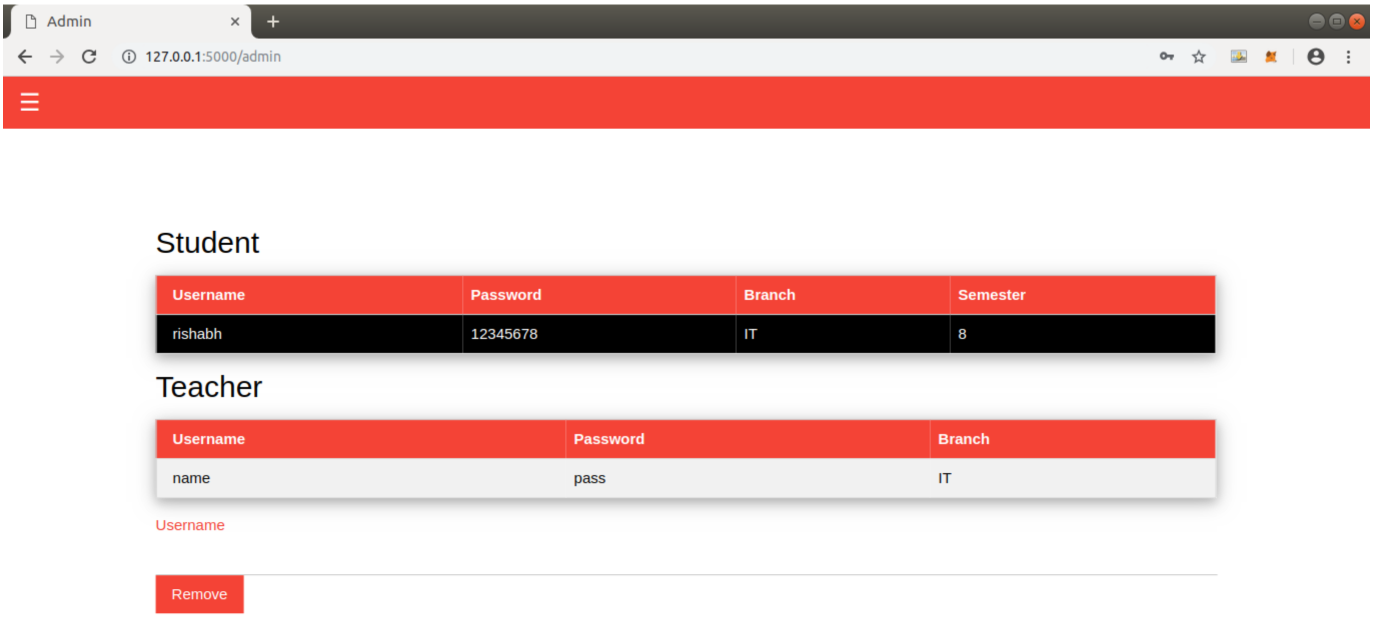
7.1 Login

This system consist of one common Login page.
Once the blind person has been entered the eLearning website he/she will tell the system to go to the login page. The system should go to the login page through voice authentication. After entering the login page there should be three types of Login Page
Admin Login
Teacher Login
Student Login



7.2 Admin

As admin logs in into the system successfully the Admin dashboard appears. Admin can add and remove students as well as can add and remove teachers. Admin can also change passwords of the user. While admin adds new students, the admin selects course and semester for the students. Admin Home:



Student

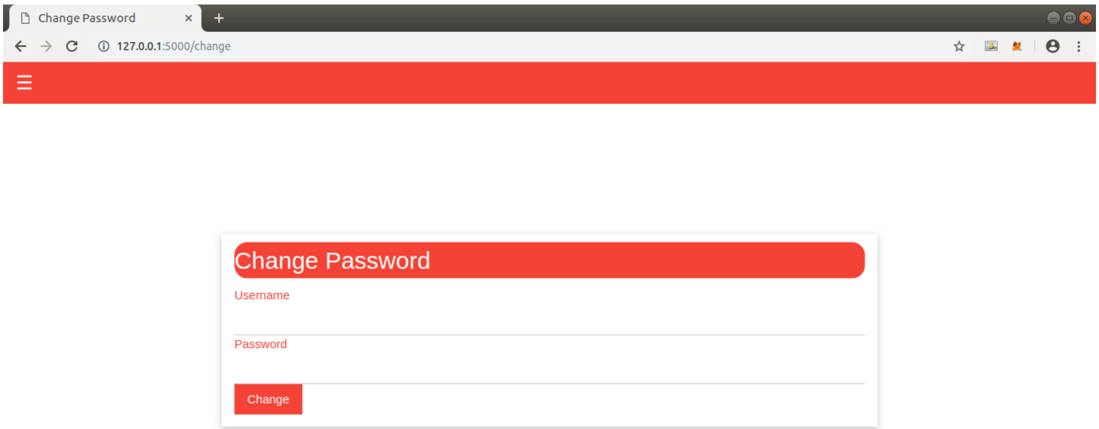
Username	Password	Branch	Semester
rishabh	12345678	IT	8

Teacher

Username	Password	Branch
name	pass	IT

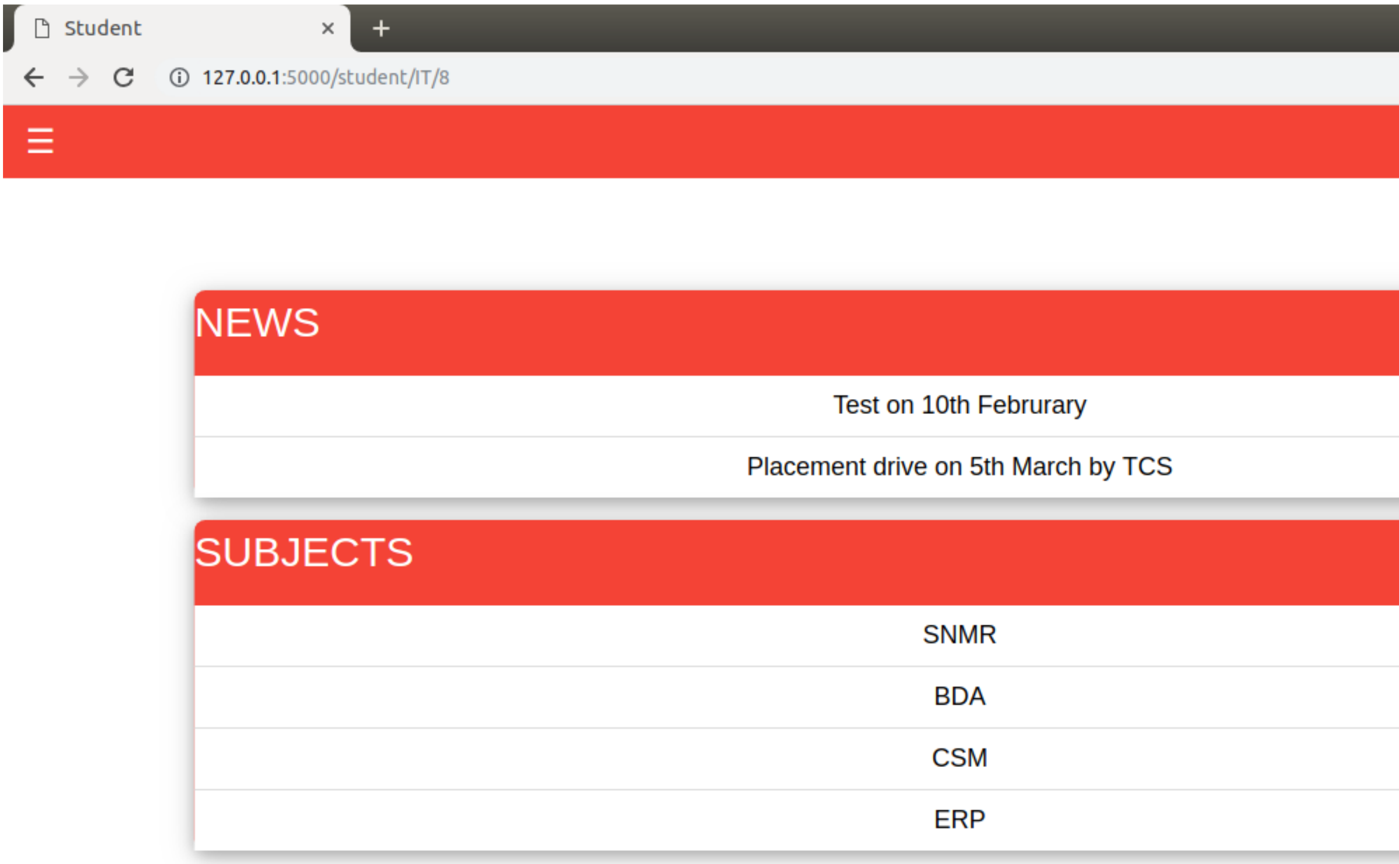
7.2.1 Admin Change Password

The users(student or teacher) username should be provided to change the passwords



7.3 Student

As Student Logs in into the LMS system successfully through voice authentication, the student dashboard appears. As shown in the image the student dashboard consist of two things to access such as news and subjects. So the systems tell the student through text to speech to choose between news or subjects. The student needs to speak in the microphone news or subjects to select one thing. If Student speaks News then system responds back through voice the list of news in the news section. The student hears all the list of news and speaks back the news he/she wants to hear. Similarly, if the student says subjects then list of subjects appears the systems again does text to speech and says out all the subjects presented in the particular course, the student simply needs to speak the particular subject he/she wants to hear and study. As the student speaks and selects particular subject , all chapters available for that subjects appears. The system says out all the chapter the teachers have uploaded, accordingly the student can speak out and select the chapter he/she wants to study. After selecting the chapter , resource in the form of PDF gets converted in form of text and then into speech, the system reads out the entire text so that the student listen to the content carefully and study.



NEWS

Test on 10th Februrary

Placement drive on 5th March by TCS

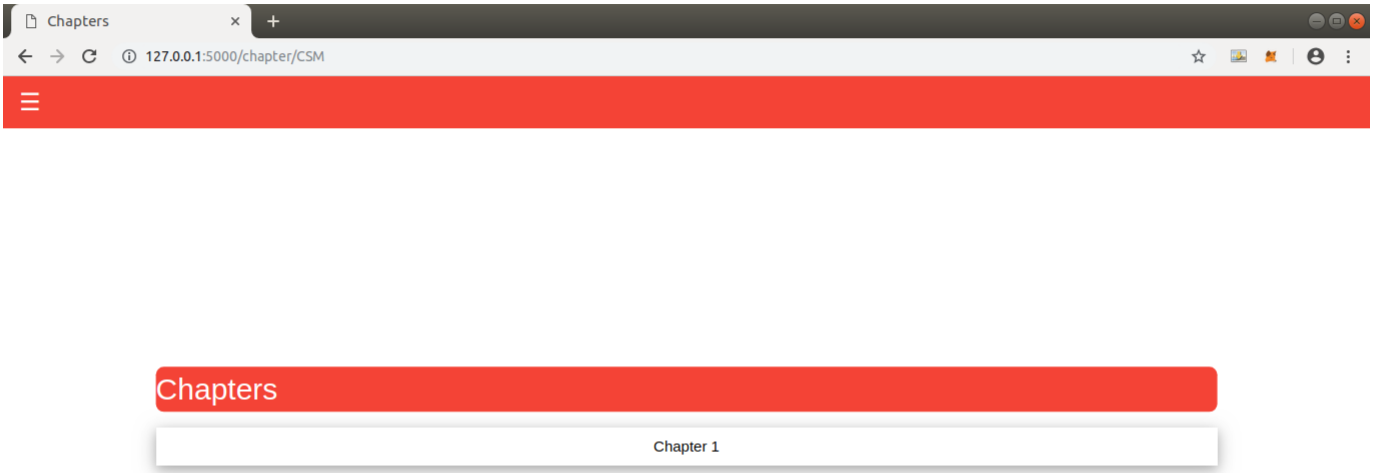
SUBJECTS

SNMR

BDA

CSM

ERP



7.4 Teachers Dashboard

Teacher can login into the system manually. As the teacher logs in into the system successfully, the teacher can get three options i.e News, Feedbacks and Courses. Teachers Home:

7.5 Teachers News

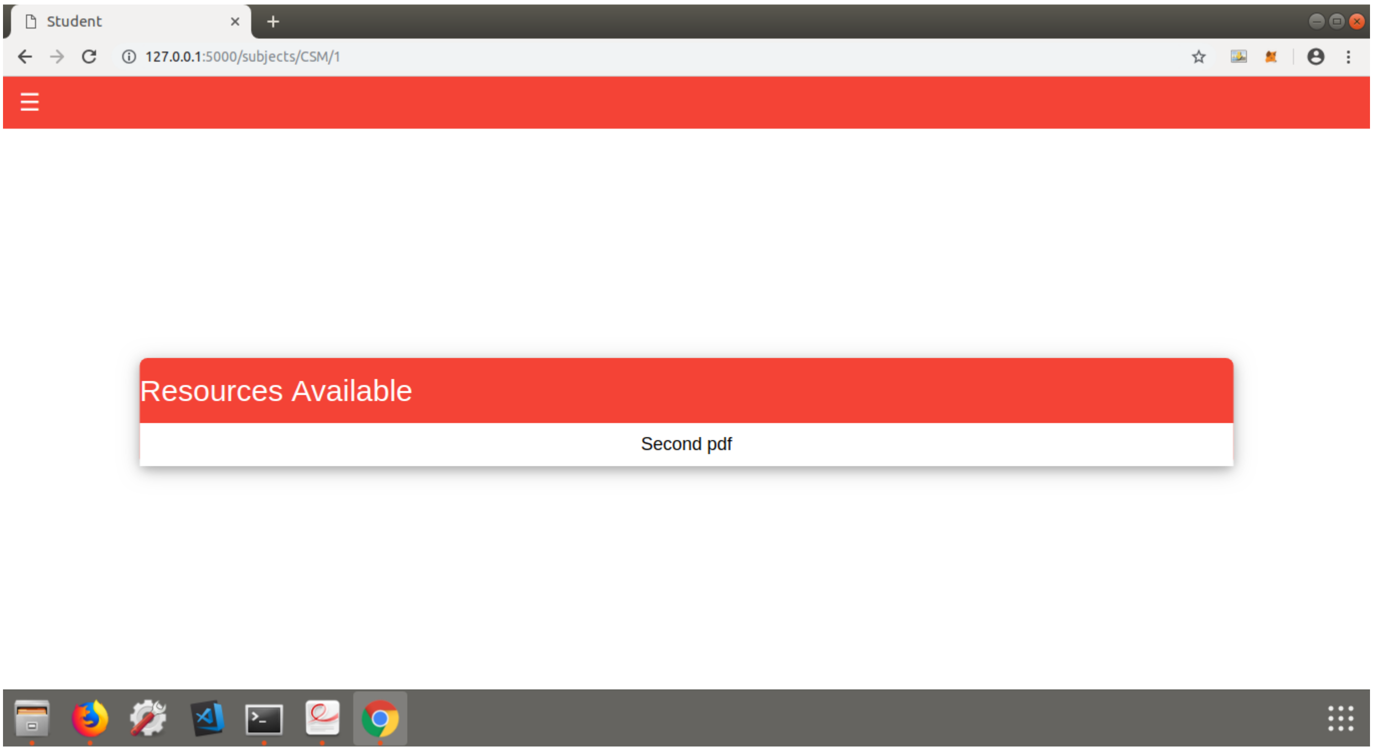
Teacher can Add and Remove News. Teacher should add news manually.

7.6 Subjects

As the teacher selects particular semester, all the subjects related to that semester will display

7.7 Teacher Resources

Teachers can upload course resources by simply uploading PDF document by clicking choose file button and then select respective pdf they want to upload. Once the pdf is uploaded the students can get those resources and they can access that course resource. Teachers can also remove particular course resource by simply clicking on the cross displayed after the course resource name.

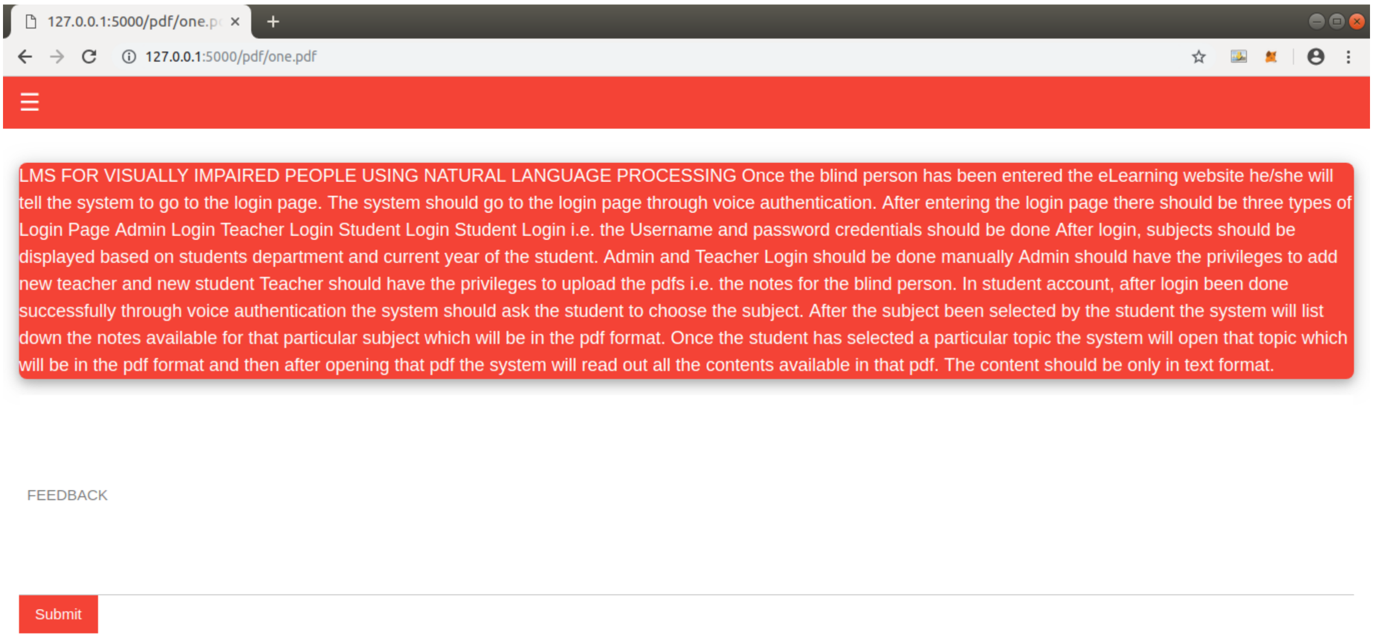


Resources Available

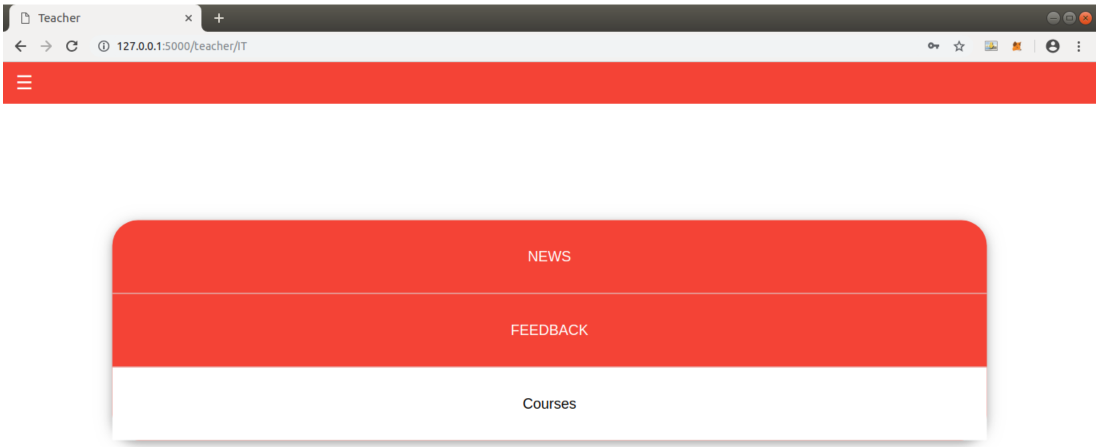
Second pdf

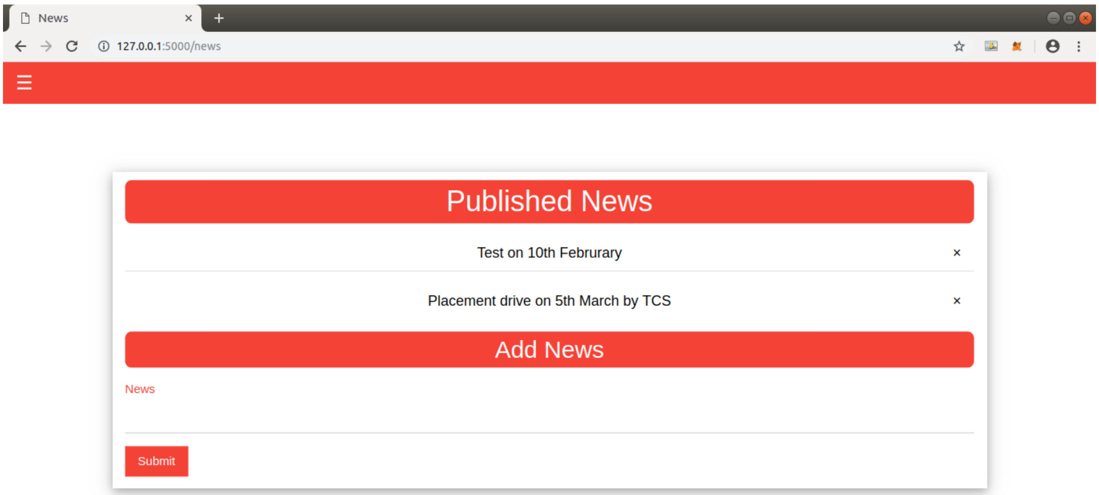
7.8 Teacher Feedback

Teachers can only view feedbacks given from the students.



LMS FOR VISUALLY IMPAIRED PEOPLE USING NATURAL LANGUAGE PROCESSING Once the blind person has been entered the eLearning website he/she will tell the system to go to the login page. The system should go to the login page through voice authentication. After entering the login page there should be three types of Login Page Admin Login Teacher Login Student Login Student Login i.e. the Username and password credentials should be done After login, subjects should be displayed based on students department and current year of the student. Admin and Teacher Login should be done manually Admin should have the privileges to add new teacher and new student Teacher should have the privileges to upload the pdfs i.e. the notes for the blind person. In student account, after login been done successfully through voice authentication the system should ask the student to choose the subject. After the subject been selected by the student the system will list down the notes available for that particular subject which will be in the pdf format. Once the student has selected a particular topic the system will open that topic which will be in the pdf format and then after opening that pdf the system will read out all the contents available in that pdf. The content should be only in text format.





Published News

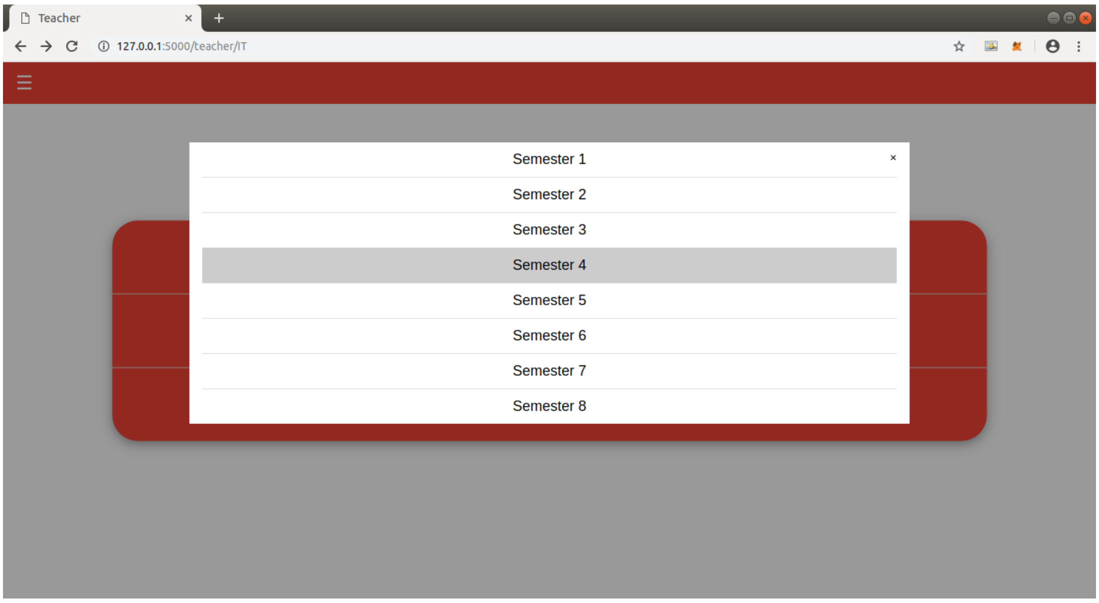
Test on 10th Februrary ×

Placement drive on 5th March by TCS ×

Add News

News

Submit



Semester 1 ×

Semester 2

Semester 3

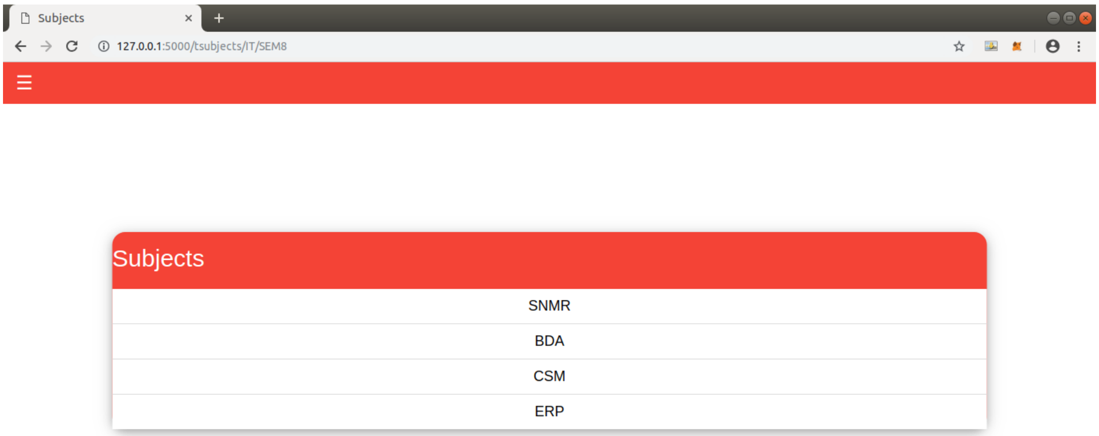
Semester 4

Semester 5

Semester 6

Semester 7

Semester 8



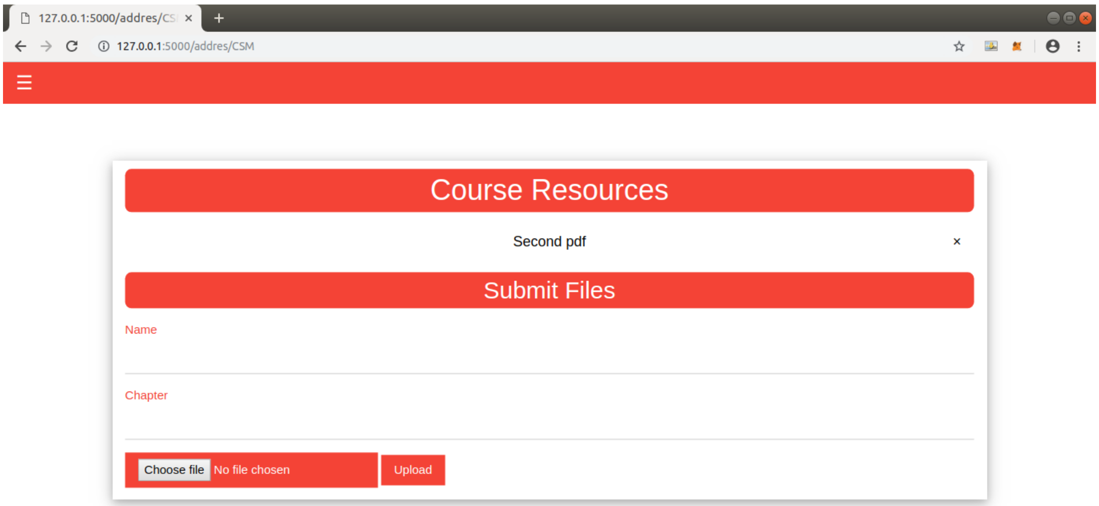
Subjects

SNMR

BDA

CSM

ERP



Course Resources

Second pdf ×

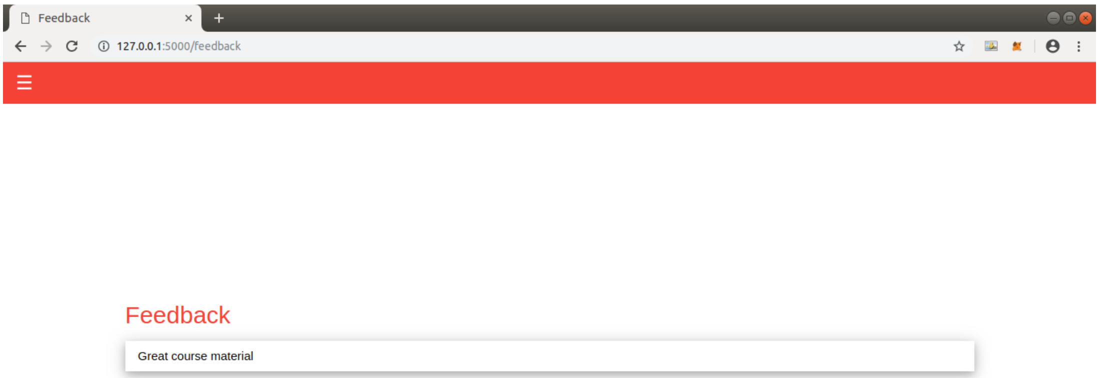
Submit Files

Name

Chapter

Choose file No file chosen

Upload



Feedback

Great course material

Chapter 8

Result

There is a growing awareness among parents, teachers, blind youth, and the adult blind community that the education which blind children are receiving is failing them. They are not receiving a quality education which can prepare them to compete in the demanding high tech economy and society of the 21st Century. They are not learning to use and trust the alternative techniques like learning systems which blind persons must adopt with if they are to be successful. They are not developing a positive attitude towards their blindness which is essential to them if they are to become confident, responsible and productive adults. Hence Considering the existing above problem, we developed a learning management system for blind youth and adult blind community.

The Results as far as now are quite good, students are liking the website and they are easily able to learn and adapt this new technology through which they can study their subjects easily. The system is made very user friendly in order to be understood easily by the teachers and students.

The voice authentication for login works well, the student is able to clearly listen and understand what system is asking for as well the student can respond to the system through Microphone. In student dashboard student listens to all the option the system says and then accordingly the student speaks and responds through microphone to the system back. The communication between the system and student happens efficiently and effectively, there is no problem students are facing as far as now. The system is very user friendly and student are loving to learn their subjects in a new fun way.

Chapter 9

Conclusion And Future Scope

9.1 Conclusion

By practically implementing website for VIP it provides a low expensive learning management system to get study material for engineering. It is the effective way to know the latest updates and report. Using voice application the normal human can reveal themselves in various domains and so breadth of application will be an impressive tool in a Ubiquitous environment. The work presented in this report is related to learning management system using Natural Language Processing. In this project, we have made a website for engineering students who are visually impaired .Blind people can interact with the website through Text-to-Speech (TTS) module. The Blind user will interact with website through voice as an input and the website will respond through text as an output. Particular Website is made for the Engineering students. This will help the Blind People to get the notes through website .It can also be used by the physically impaired people, making learning easy to them.

9.2 Future Scope

We will try to create application for blind people so that they can learn like normal people.The Blind people can hear and learn the notes from the website and also from the application.Blind people can hear and learn the notes from mobile devices.

Bibliography

[1] Multilanguage natural user interface to database Rus-lan Posevkin Chair of Computation Technologies ITMO University Saint Petersburg, Russia rusposevkin@mail.ru2017

[2] Construction of a Voice Driven Life Assistant System for Visually Impaired People ;RunzeChen, Zhanhong Tian, Hailun Liu, Fang Zhao, Shuai Zhang, Haobo Liu ;Beijing, China 2017

[3] Linda A. Winters-Miner PhD, ... Gary D. Miner PhD, in Practical Predictive Analytics and Decisioning Systems for Medicine, 2015

[4] Ira J. Kalet, in Principles of Biomedical Informatics (Second Edition), 2014

[5] International Journal of Computer Applications (09758887) Volume 131 No.11, December2018

[6] Sagar V. Ramani and Yagnik N. Tank, Indoor Navigation on Google Maps and Indoor Localization Using RSS Fingerprinting, International Journal of Engineering Trends and Technology (IJETT), Volume 11 Number 4, May 2014.

[7] David Zhou, Hanbing Yan and Yonggao Yang, A smart virtual eye mobile system for the visually impaired, IEEE Potentials, 2016

[8] Mohd Ibrahim, Rodina Ahmad Department of Software Engineering University of Malaya Kuala Lumpur, Malaysia 2017

[9] Jin-hee Lee, Sang-Chul Lee, Kyeongyul Kim, and ByeongSeok Shin, Smart Backpack for Visually Impaired Person, International Conference on ICT for Smart Society-IEEE, June 201

Chapter 10

Appendices

10.1 Appendix-A: Python Download Installation

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales.

- Download and execute the latest Python installation package from Python.org. For Liquid Web servers, youll most likely want the 64-bit version of the installer, but you may want to discuss software requirements with your developer.
- Choose the recommended installation options for the simplest installation experience (You can also choose Customize Installation if you need to adjust locations or features, but this may require additional configuration. See Python.org for further instructions on custom installation options).
- Check the box for Add Python 3.7 to PATH. This will adjust your System Environment Variables automatically so that Python can be launched from any command prompt.
- Verify a successful installation by opening a command prompt window in your Python installation directory (probably C:yourusername*37 if youve installed the latest available version). You should receive a message similar to what is shown below. If you selected Add Python 3.7 to PATH, you can verify the installation from any command prompt window.

https://www.overleaf.com/project/5c946fa9fc580e021702550f
https://www.overleaf.com/project/5c946fa9fc580e021702550f

10.2 Appendix-B: Installing PIP

One of the best tools to install and manage Python packages is called Pip. Pip has earned its fame by the number of applications using this tool. Used for its capabilities in handling binary packages over the easy installed packaged manager, Pip enables 3rd party package installations.

- Download get-pip.py to a folder on your computer.

- Open a command prompt and navigate to the folder containing get-pip.py.
- Run the following command:
  python get-pip.py
- Pip is now installed.
- You can verify that Pip was installed correctly by opening a command prompt and entering the following command: pip -V
- You should see output similar to the following: pip 18.0 from c:37-packages
  (python 3.7)

Acknowledgement

I have great pleasure in presenting the report on **LMS for visually impaired people using NLP** I take this opportunity to express my sincere thanks towards my guide **Prof. Apeksha Mohite** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. I would like to express my gratitude towards his constant encouragement, support and guidance through the development of project.

I thank **Prof. Kiran B. Deshpande** Head of Department,IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

I thank **Prof. Vishal S. Badgujar** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

I also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. I wish to express my deep gratitude towards all my colleagues of APSIT for their encouragement.

Student Name:

Student ID:

Publication

Paper entitled **“LMS for visually impaired people using NLP”** is presented at **“International Conference/Journal Name”** by **“Author Name”.**