Human Computer Interaction

CSE4015

Slot: A1+TA1

Winter Semester 2022-23

A J Component Report

On

Augmented Visual Intelligence

Submitted By:

Mudit Bhatta (20BCE2888)
Sandesh Khatiwada (20BCE2898)
Sarjak Devkota (20BCE2922)
Siddhant Karki (20BCE2936)

Submitted To:

Prof. Dr. Swarnalatha P
Associate Professor Grade 2
SCOPE
VIT, Vellore.

Table of Contents

Declaration

We affirm that we have personally conducted the project work entitled "**Augmented Visual Intelligence**" as a part of the Human Computer Interaction course (CSE4015) at Vellore Institute of Technology, Vellore under the supervision of **Dr. Swarnalatha P.**

Additionally, we confirm that we have not submitted any part or whole of this project for completion of any other project at this institute or any other university.

Sandesh	Sarjak	Mudit	Siddhant
Sandesh Khatiwada	Sarjak Devkota	Mudit Bhatta	Siddhant Karki

Demo Video Link: [HCI project - Google Drive]

Chapter 1 Introduction to Project

1.1 Introduction

The Augmented Visual Intelligence project is aimed at providing an easy-to-use web application to visually impaired students, enabling them to access educational content, play games, take tests, and perform other activities. The web app is designed to be navigated through a search bar or voice commands, making it accessible to those who are visually challenged.

In today's world, the internet has become an essential source of information that should be accessible to everyone, including those with visual impairments. With the help of technologies like speech recognition, websites can be made easily accessible to visually impaired individuals. By providing audio input and receiving speech output, these individuals can navigate the website with ease. This approach allows visually impaired individuals to engage with websites using voice input and simplifies the process of reading content and navigating between links through voice output, thereby addressing the accessibility concerns of the visually impaired.

1.2 Problem Definition

In today's era, the internet has become a crucial resource for people to expand their knowledge and learn new things. However, not everyone has benefited equally from the digital revolution. Unfortunately, individuals with visual impairments are still unable to fully take advantage of the internet due to difficulties in its utilization.

While technology has made life easier for everyone, including blind people, they still face significant challenges when using specific technology, such as browsing a webpage, as they are unable to see the content and interact with it effectively. Additionally, most visually impaired individuals have some degree of vision loss, and the available assistive equipment is often expensive and not easily accessible.

For instance, the majority of blind users cannot afford to install specialised reading aids like screen reader access, which can be both challenging and costly. Thus, it is crucial to design websites with simplicity in mind, making it easy for blind or visually impaired individuals to use and navigate. Technologies like voice recognition can play a vital role in improving accessibility for visually impaired individuals.

1.3 Project Scope

The Augmented Visual Learning project offers a web application specifically designed for visually impaired students. The application provides several features such as playing games, taking voice-assisted tests, reading aloud, and navigating the application through voice commands. Based on the prototype, we can add other functionalities like voice-assisted to-do lists, news, and weather updates. Additionally, we can incorporate voice-activated games, support for multiple languages, and more, as we further explore the concept.

With significant growth, our product could be used in special schools for visually impaired students, simplifying the learning process for both the students and teachers. The potential benefits of this application are immense, and it can have a significant positive impact on the lives of visually impaired students by providing them with a more accessible and interactive learning experience.

1.4 Motivation

Our team has developed a sample website that offers several features specifically designed to assist visually impaired students. These features include object identification, text-to-speech recognition, and voice aid while browsing the website.

Unfortunately, most websites present significant barriers for individuals with partial blindness, but our website aims to reduce these barriers and empower blind people by providing a more accessible and interactive user experience. Through our website, we aim to foster digital literacy among blind people, thereby enabling them to navigate the internet with greater ease.

As we continue to explore and develop our concept, we plan to incorporate more advanced technologies and features to increase accessibility and improve the user experience for those who are blind. Our ultimate goal is to create a more inclusive and equitable digital space that is accessible to everyone, regardless of their visual ability.

1.5 Background Study/Literature Survey

S.NO	TITLE	AUTHOR	LEARNINGS

1	Survey on Text to Speech Synthesis Models and Methods	Pooja A.Gundle, R.K. Chavan	Speech conversion involved text analysis, text normalization, text processing, grapheme-to phoneme conversion and speech synthesis. Process included 1. Text 2. Text analysis/text normalization 3. Phonetic analysis/Grapheme-to-Phoneme conversion 4. Speech synthesis/waveform generation 5. Speech Different type of TTS models are explained: 1. Signal to signal model: In this model the process of converting written signal to spoken signal. Here we directly convert textto speech. 2. Pipelined models: signal to signal model implemented as a pipeline model, theprocess is like the passing representation from one module to another. Each module's job is defined as reading one type of information and producing another 3. Grapheme and phoneme form model: In this model grapheme form of the text inputis -converted into phoneme form of speech synthesis that is exact pronunciation of each word of the input sentences. In this, three main speech synthesis method are explained.

				1. Formant synthesis: It produces quality speech which sounds the unnatural. Formant synthesis adopt model based, acoustic phonetic approach to the synthesis problem.

2. Articulatory speech synthesis: The machine was mechanical device with tubes.The device is of course mimicking vocal tractusing sources and filters.

3. concatenative speech synthesis: Concatenative synthesis produces artificial speech by concatenating the pre-recorded units of speech such as phonemes, diphones, syllables, words or sentences

4. HMM based speech synthesis: In the HMM based speech synthesis the speech parameters of speech unit such as spectrum,fundamental frequency, and phoneme duration are statistically model and generated by using HMMs based on maximum likelihood criterion

2	Text – To – Speech Synthesis (TTS)	Cosmas Ifeanyi Nwakanma, Ikenna Oluigbo , Okpala Izunna	It runs on JAVA platform, and the methodology used was Object Oriented Analysis and Development Methodology; while Expert Systemwas incorporated for the internal operations of the program. This design will be geared towards providing a one-way communication interface whereby the computer communicates with the user by reading out textual document for the purpose of quick assimilation and reading development. A new system was proposed: 1. The new system has a reasoning process. 2. The new system can do text structuring and annotation. 3. The new system's speech rate can be adjusted. 4. The Pitch of the voice can be adjusted. 5. You can select between different voices and can even combine or juxtapose them if you wantto create a dialogue between them 6. It has a user friendly interface so that peoplewith less computer knowledge can easily use it 7. It must be compatible with all the vocal engines 8. It complies with SSML specification The TTS system converts an arbitrary ASCII textto speech. The first step involves extracting the phonetic components of the message, and we obtain a string of symbols representing sound- units (phonemes or allophones), boundaries between words, phrases and sentences along with a set of prosody markers (indicating the speed, the intonation etc.). The second step

			consists of finding the match between the sequence of symbols and appropriate items stored in the phonetic inventory and binding them together to form the acoustic signal for thevoice output device. The speech engine used in this new system wasthe FreeTTS speech engine. FreeTTS was usedbecause it is programmed using JAVA (the backbone programming language of this designed TTS system). It also supports SAPI (Speech Application Programming Interface) which is in synchronism with the JSAPI (Java Speech Application Programming Interface).

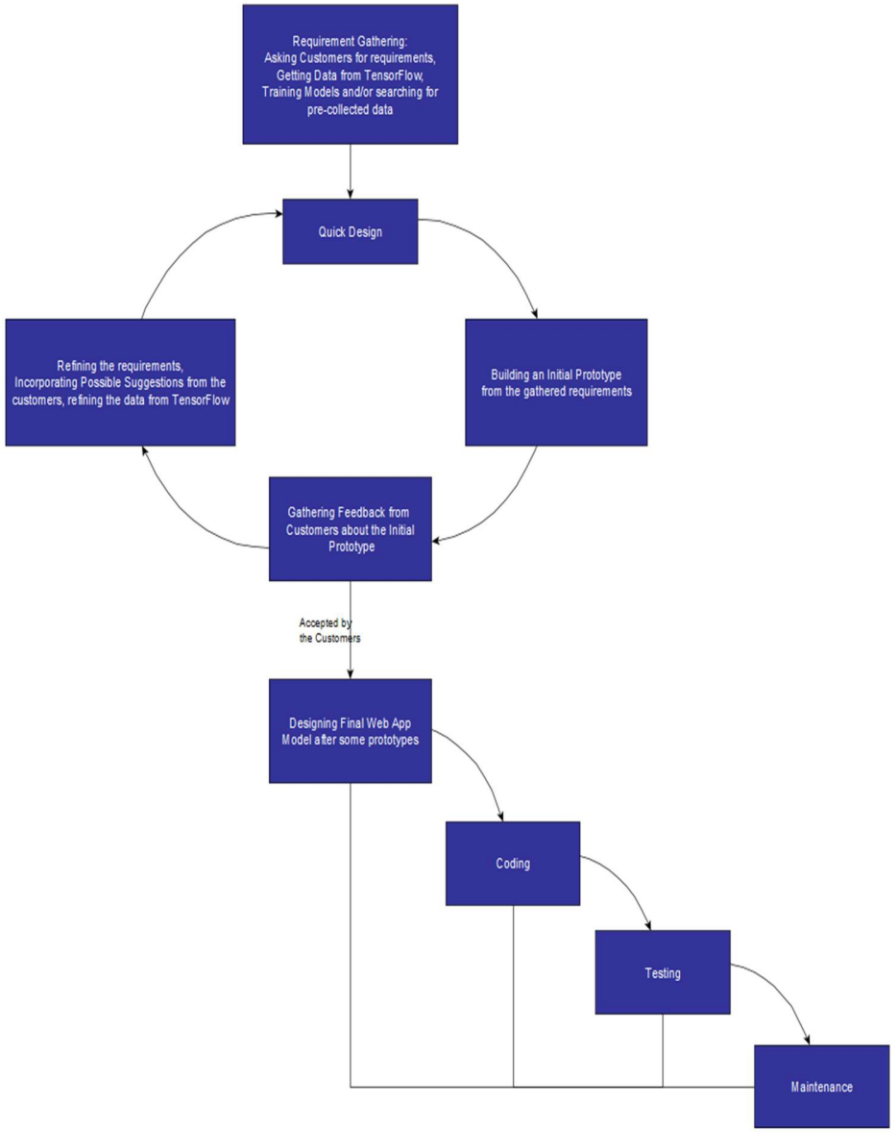
3	VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection	Yin Zhou, Oncel Tuzel	In this work, they proposed the need of manual feature engineering for 3D point clouds and proposed VoxelNet, a generic 3D detection network that unifies feature extraction and bounding box prediction into a single stage, end-to-end trainable deep network. They designed anovel voxel feature encoding (VFE) layer, which enables inter-point interaction within a voxel, by combining point-wise features with a locally aggregated feature. The network learns an effective discriminative representation of objects with various geometries, leading to encouragingresults in 3D detection of pedestrians and cyclists, based on only LiDAR. They have conducted experiments on KITTI benchmark and show that VoxelNet produces state-of-the-art results in LiDAR-based car, pedestrian, and cyclist detection benchmarks.
4	IterDet: Iterative Scheme for Object Detection in Crowded Environments	Danila Rukhovich, Konstantin Sofiiuk, Danil Galeev, Olga Barinova, Anton Konushin	Deep learning-based detectors usually producea redundant set of object bounding boxes including many duplicate detections of the sameobject. In this work we develop an alternative iterative scheme, where a new subset of objectsis detected at each iteration. Detected boxes from the previous iterations are passed to the network at the following iterations to ensure thatthe same object would not be detected twice. They presented an iterative scheme (IterDet) for object detection designed for crowded environments. It can be applied to both two- stage and one-stage object detectors. Experiments on challenging AdaptIS ToyV1 and ToyV2 datasets with multiple overlapping objects demonstrate that IterDet is able to achieve almost perfect detection accuracy. Extensive comparison on CrowdHuman and WiderPerson benchmarks shows that proposed scheme achieves higher accuracy compared to existing works when applied to both two-stage

			Faster RCNN and onestage RetinaNetdetectors.
5	Single-Shot Refinement Neural Network for Object Detection	Shifeng Zhang, Longyin Wen , Xiao Bian, Zhen Lei, Stan Z. Li	They proposed a novel single-shot based detector, called RefineDet, that achieves better accuracy than two-stage methods and maintainscomparable efficiency of one-stage methods. RefineDet consists of two inter-connected modules, namely, the anchor refinement moduleand the object detection module. Specifically, the former aims to (1) filter out negative anchorsto reduce search space for the classifier, and (2)coarsely adjust the locations and sizes of anchors to provide better initialization for the subsequent regressor. The ODM takes the refined anchors as the input from the former ARM to regress the accurate object locations and sizes and predict the corresponding multiclass labels.

1.6 SDLC approach used to develop project

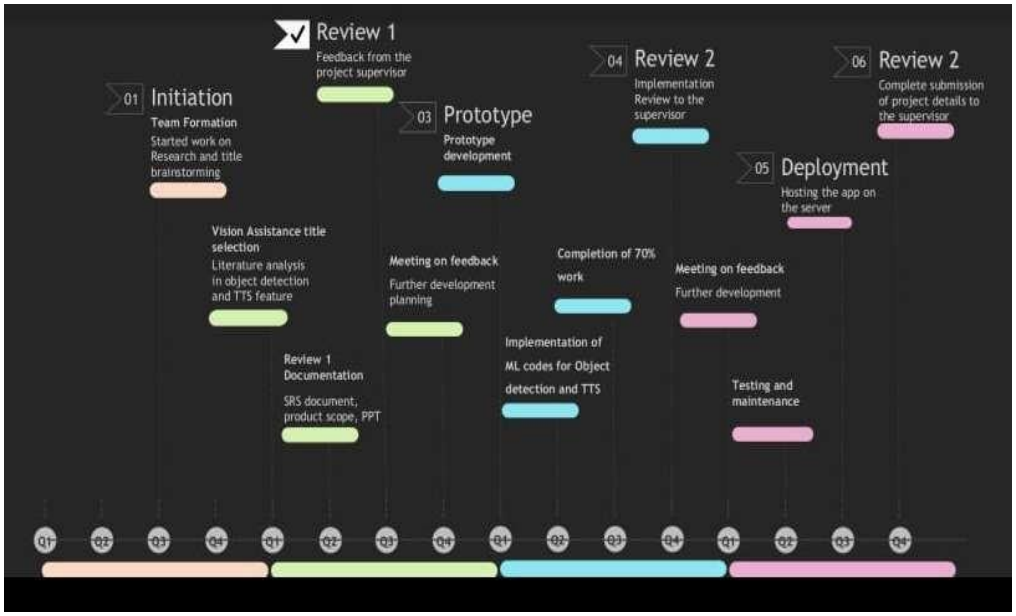
Our project utilises the concept of prototype modelling, whereby a prototype is created, tested, and refined as needed to achieve a satisfactory outcome that can serve as the basis for the entire system or product. Given that not all needs may be fully understood at the outset, our team plans to add additional features as they become necessary and appropriate along the way.

Initially, we will implement the project with a few predetermined basic elements, and then gradually introduce more features and improvements based on user and customer feedback. We will follow a trial-and-error approach, making adjustments based on input from our users and customers, and refining our product until we achieve the desired outcome. Our ultimate goal is to create a solution that meets the needs of visually impaired students, is user-friendly, and enhances their learning experience.


Requirement Gathering:
Asking Customers for requirements,
Getting Data from TensorFlow,
Training Models and/or searching for
pre-collected data
Quick Design
Building an Initial Prototype
from the gathered requirements
Refining the requirements,
Incorporating Possible Suggestions from the
customers, refining the data from TensorFlow
Gathering Feedback from
Customers about the Initial
Prototype
Accepted by
the Customers
Designing Final Web App
Model after some prototypes
Coding
Testing
Maintenance

Chapter 2 Project Planning

2.1 Project Schedule



01 Initiation

Team Formation

Started work on Research and title brainstorming

Review 1

Feedback from the project supervisor

03 Prototype

Prototype development

04 Review 2

Implementation Review to the supervisor

06 Review 2

Complete submission of project details to the supervisor

05 Deployment

Hosting the app on the server

Vision Assistance title selection

Literature analysis in object detection and TTS feature

Meeting on feedback

Further development planning

Completion of 70% work

Meeting on feedback

Further development

Implementation of ML codes for Object detection and TTS

Review 1 Documentation

SRS document, product scope, PPT

Testing and maintenance

2.2 Effort and Resource Estimation

Our project is a collaborative effort that involves the integration of various functionalities to create a seamless user experience for visually impaired students. To ensure that each module is developed to the highest possible standard, we divided the project among ourselves, with each team member responsible for a particular aspect. The front-end team was tasked with creating the user interface, while the back-end team worked on the server-side functionality.

We also developed modules for object detection, note making, and text-to-speech conversion. These modules were integrated with the rest of the project to ensure a smooth flow of information and functionality. The use of Google's search API for text-to-speech conversion enabled us to provide a high-quality voice output that is easy to understand and navigate for visually impaired students.

Throughout the development process, we maintained a high level of collaboration and communication to ensure that the project was delivered on time and to the desired standard. We tested each module thoroughly and refined it as necessary to ensure that it met the needs of our target audience. By dividing the project into modules and integrating them seamlessly, we were able to create a user-friendly solution that enhances the learning experience of visually impaired students.

Chapter 3 Requirement Gathering and Analysis

3.1 SRS

1. Introduction

1.1 Problem Statement

In today's world, the Internet has become an indispensable tool for acquiring knowledge and expertise. However, not everyone has equal access to it, especially those with visual impairments who face difficulty in navigating the web. While technology has made life easier for everyone, including the visually challenged, they still face obstacles when it comes to interacting with specific technologies like browsing a website. Moreover, assistive devices can be expensive and are often inaccessible to most blind users. Hence, creating a website that is easy to navigate and use for visually challenged people requires the incorporation of speech recognition and other assistive technologies.

To address these concerns, the Augmented Visual Learning project has created a web application that enables visually impaired students to play games, take voice-assisted tests, and read material aloud, among other features. The prototype includes object identification, text-to-speech recognition, and voice aid while browsing the website. Such features will reduce the technological barrier and empower the blind by fostering digital literacy. The team plans to add more features along the way and take a trial-and-error approach to adjust and improve the product based on user feedback.

In summary, technology has made life easier for everyone, but it has not yet solved the accessibility issue for visually impaired people on the internet. The Augmented Visual Learning project aims to reduce this gap by creating a website that incorporates assistive technologies like speech recognition. With the trial-and-error approach and user feedback, the project will continue to improve and become more inclusive, making learning simpler for visually challenged students.

1.2 Solution

incredibly beneficial for visually impaired students, including object detection, text-to-speech recognition, and voice assistance for website navigation. Unfortunately, most websites are not designed with partial blindness in mind, making it difficult for the visually challenged to interact with them. Our website eliminates this technological barrier, enabling visually challenged people to interact with websites more easily, empowering them and promoting digital literacy.

As we continue to develop our project, we plan to incorporate even more sophisticated technologies and features to enhance the accessibility of the internet for the visually impaired. Our website represents just the beginning of a broader effort to create a more inclusive online environment that accommodates people with different levels of vision impairment. By continually expanding the scope of our project, we aim to make the internet more accessible to people with visual impairments, enabling them to experience the same benefits and opportunities as those with normal vision.

Overall, our goal is to promote digital literacy and empower visually impaired individuals, enabling them to access and use online resources more effectively. By incorporating innovative technologies and designs that address the specific needs of visually challenged people, we hope to create a more inclusive online environment that benefits everyone.

1.3 Purpose

Additionally, the Augmented Visual Intelligence project has incorporated various features such as object identification, text-to-speech recognition, and voice aid while browsing the website. These features are intended to help the visually impaired students interact with the website more easily and reduce the technological barrier, thereby empowering them and promoting their digital literacy.

As the project progresses, more features such as voice-activated games, support for multiple languages, and other capabilities like news and weather updates can be added based on user feedback and requirements. The ultimate goal of the project is to make learning simpler for both visually impaired students and teachers at special schools, and to increase their accessibility to educational content online.

1.4 Process Model

Prototype modelling is the chosen method for this project, which involves building, testing, and revising a prototype until the final product is obtained. As the team plans to add more features as the project progresses, not all requirements may be known beforehand. To begin, the project will be implemented with predetermined basic functionalities, with additional features and improvements added later. The team will utilize a trial-and-error approach, taking feedback from users and customers to make necessary adjustments.

1.5 Intended Audience

The visually impaired individuals are the primary stakeholders and users of our project, while the owners of special institutes for visually impaired students and their family and friends are the secondary users. The project is also accessible to the entire society, making them tertiary users.

This document serves as a guide for developers and testers of the system as it is subject to approval or disapproval based on user and customer feedback. Additionally, individuals and developers who are interested in computer vision can use our project as a prototype for further development and advancement.

1.6 Product Scope

Our Augmented Visual Intelligence project includes a web application designed for visually challenged students to assist them with activities like playing games, taking voice-assisted tests, reading text with voice assistance, and navigating the web-app through voice command. Additional features such as voice-assisted to-do lists, news and weather updates, and voice-assisted gaming may be added based on the prototype. Further research may also allow us to integrate multiple languages and other advanced features. As the project develops, it could potentially be implemented in special institutes for visually impaired students to make learning more accessible for both students and teachers.

2. Overall Description

2.1 Product Perspective

Augmented Visual Intelligence is a comprehensive solution that aims to make learning smooth and effective for visually impaired students. Our technology enhances various cognitive skills and incorporates teaching through unified experiences. With features such as object detection, voice assistance, text to speech capability, and academic support, our technology is a valuable resource for visually impaired individuals.

2.2 Product Functions

Objection detection: Upon opening the page, the camera automatically turns on and the system begins reading aloud whatever is in front of it.

Text to speech: The voice assistant is capable of translating all instructions and listening to the user's voice commands.

Academic help: The web app includes a feature for entertainment such as games like tic tac toe to provide a fun and engaging experience for users. In addition, the voice assistance feature can also read out any text or document paragraphs for the convenience of visually impaired users.

2.3 Operating Environment

The Augmented Visual Intelligence project is a web application that will be deployed on a cloud Platform as a Service (PaaS), making it easily accessible to visually impaired students and other individuals in need. The web app is compatible with both Windows and macOS operating systems.

2.4 Design and Implementation Constraints

- The visually impaired user of Assisted Vision needs to be a given a proper training about how to use the interface
- Minor problem could arise during object detection due to insufficient brightness of the surrounding environment
- There could be viewpoint variation, deformation---incase the object detector istrained to detect a person only in a particular posture, it might not be able to detect people in any other posture. The objects of interest can be occluded.

Sometimes only a small portion of an object (as little as a few pixels) may be visible.

- Text to speech can also give rise to pronunciation errors as in the user might not understand the pronunciation of the synthetic voice system (when the tts system might not know the correct way to pronounce the text).
- Audio quality of the TTS might be poor and not clear sometimes

2.5 User Documentation

The website can be accessed easily by users from their personal computers, including desktops and laptops.

2.6 Assumptions and Dependencies

The project is designed to operate on personal computers and may not perform optimally on smartphones or tablets. It employs an object detection algorithm that does not authenticate the user's actual identity. Additionally, it is assumed that the user will comprehend the voice assistance's interpretation.

3. External Interface Requirements

3.1 User Interfaces

The app aims to make the life of vision impaired people easier and more productive.To achieve this, we have decided on following UI Interface:

- A simple and clean interface.
- Large area of screen utilized as controls so the user can have a bit ofsense of where to press the button.
- Constant voice feedback to navigate the app.
- Ability to give voice commands to the app.

3.2 Hardware Interfaces

- **Camera:** This system will include a camera attachment to detect the objects and provide other general and security functionalities. The camera we will use will be already available to user in form of web cam in laptop orcamera in phone.

- **Computer/phone:** An end device is necessary to run our proposed application to utilize its features to full extent.

3.3 Software Interfaces

The development process of the software relies heavily on various tools and components. These can be categorized as follows:

- Frontend Components:o

  HTML/XML

  - CSS/Bootstrap
  - React
- Backend Component:
  - Node/Express js
  - Java script
  - Tensor flow (Object detection)

3.4 Communications Interfaces

Our plan is to host the app on the internet using HTTP protocols, making it accessible to anyone worldwide and assisting many people in their daily life. By utilizing available servers and services, we aim to distribute this facility to everyone.

4. System Features

The primary target audience for this software is individuals with visual impairments, and it offers several key features, including:

4.1 Object Detection:

The software is designed to detect objects using machine learning (ML), object recognition, and image processing. To accomplish this, we will be utilizing Google's Tensor Flow framework, which simplifies the process of acquiring data, training models, serving predictions, and refining future results. Additionally, the software will be capable of supporting face detection and detection of surrounding objects.

The following features are prioritized for implementation in the software:.

4.2 Text-to-Speech:

The following features are considered a high priority for implementation:

- The ability for the user to have the application's contents read out loud by a voice assistant.
- Voice commands can be used for navigation throughout the application.
- The Web Speech API will be utilized to incorporate voice data into the web application.
- The Speech-Synthesis interface of the Web Speech API will serve as the controller interface for the speech service, allowing for the retrieval of information about the available synthesis voices, the start and pause of speech, and other related commands.
- Customized commands can be added based on user needs to enhance the overall user experience.
- Voice commands can also be utilized for emergency purposes such as contacting emergency numbers and activating emergency features.

4.3 Academic Assistance:

The software is designed to provide additional support for visually impaired individuals. The following features have been identified as having medium to high priority:

- Text detection will be implemented, allowing users to read books and written materials through the camera. This will enable visually impaired individuals to participate in classes and other educational activities.
- The software will feature a voice-assisted quiz function, allowing users to take exams and tests using voice commands.
- A to-do list will be included, allowing users to manage their schedule and set tasks using voice commands.

4.4 Entertainment:

- The software may include voice-assisted games such as tic-tac-toe in future versions. Additionally, it may have the capability to play music using voice commands. These features are considered low priority for implementation..

4.5 Other features:

- The software will have multi-language support including English, Hindi, Spanish, and potentially more languages in future versions.
- Additional features like news, weather updates, and horoscopes using the voice assistant may be added in future versions, but these have a low priority to be implemented.
-

5. Other Non-functional Requirements

5.1 Performance Requirements

- Reliability: The software should be able to perform consistently without any downtime or unexpected crashes. This is especially important for visually impaired users who may heavily rely on the software.

- Accessibility: The software should be accessible to all users regardless of their abilities. This means adhering to accessibility guidelines and providing options for customization and personalization.

- Security: The software should have appropriate security measures in place to protect user data and prevent unauthorized access. This includes secure authentication and encryption of sensitive data.

- Scalability: The software should be able to handle a large number of users and requests without compromising on performance. This is important for ensuring that the application remains responsive even under heavy loads.

- Compatibility: The software should be compatible with different platforms, devices and operating systems to ensure that it can be accessed by a wide range of users.

- Usability: The software should be easy to use and navigate for visually impaired users. This includes providing clear and concise instructions, using simple and intuitive interfaces, and providing feedback in a clear and understandable manner.

-

5.2 Safety Requirements

To ensure that the software is reliable and safe to use, the following steps can be taken:

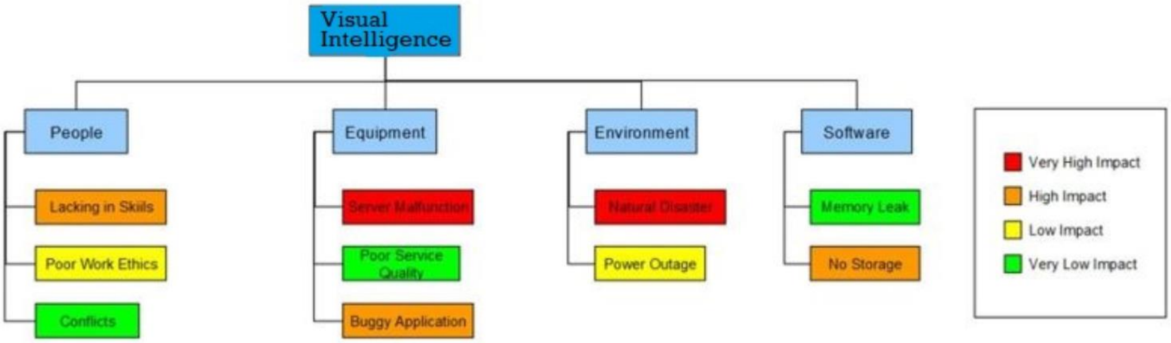
· Thorough testing: The software should undergo extensive testing to detect and fix any potential bugs or errors before it is released to the public.

· Regular maintenance: The software should be regularly updated and maintained to prevent any dormant faults from causing issues.

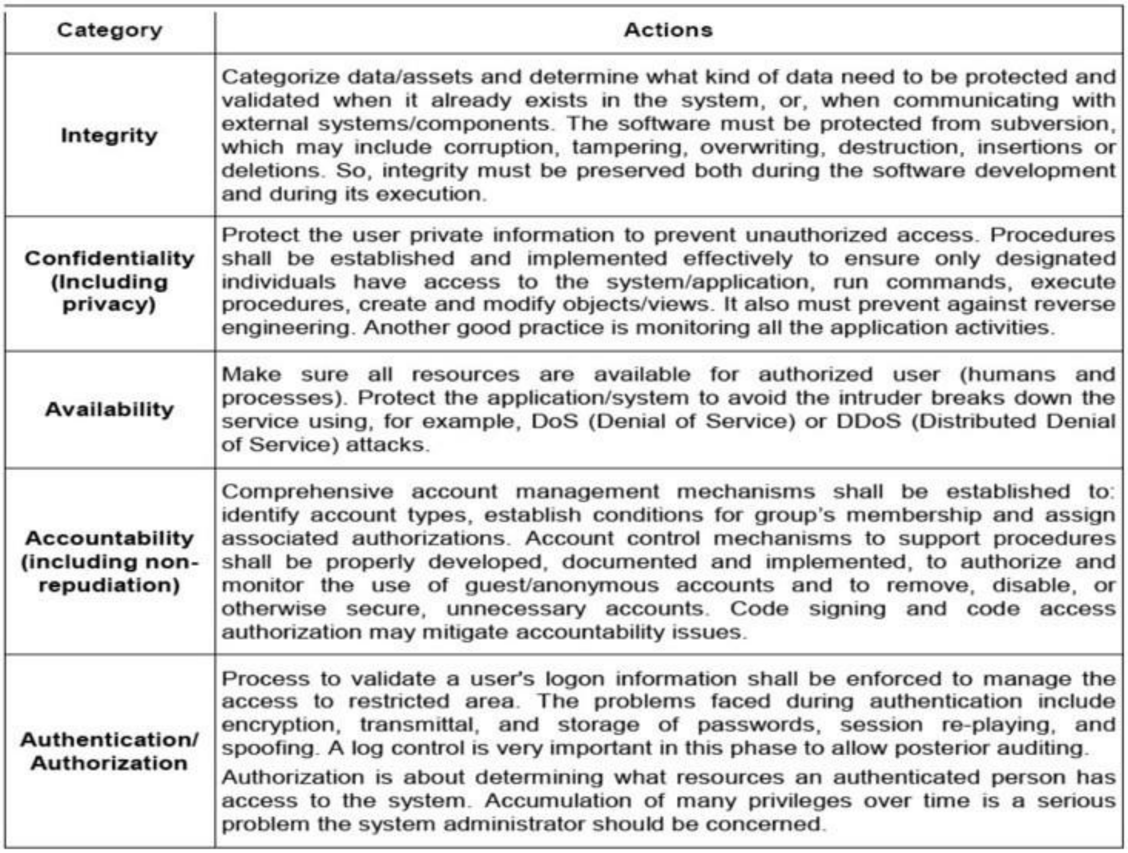
· Proper documentation: The system should have proper documentation and specifications, which will help in preventing specification errors.

· Failure analysis: The system should have failure analysis processes in place that can identify and mitigate risks associated with hardware failures.

· Risk assessment: The system should undergo regular risk assessments to identify and mitigate any potential risks associated with its operation.



5.3 Security Requirements

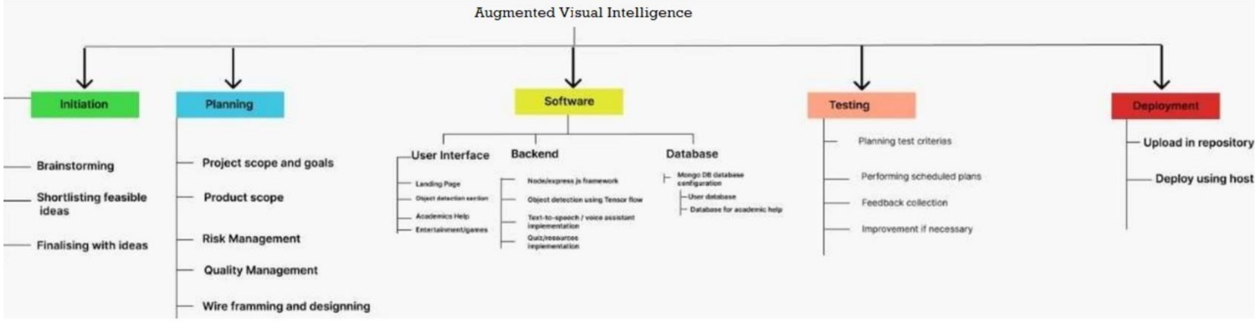


Category	Actions
Integrity	Categorize data/assets and determine what kind of data need to be protected and validated when it already exists in the system, or, when communicating with external systems/components. The software must be protected from subversion, which may include corruption, tampering, overwriting, destruction, insertions or deletions. So, integrity must be preserved both during the software development and during its execution.
Confidentiality (Including privacy)	Protect the user private information to prevent unauthorized access. Procedures shall be established and implemented effectively to ensure only designated individuals have access to the system/application, run commands, execute procedures, create and modify objects/views. It also must prevent against reverse engineering. Another good practice is monitoring all the application activities.
Availability	Make sure all resources are available for authorized user (humans and processes). Protect the application/system to avoid the intruder breaks down the service using, for example, DoS (Denial of Service) or DDoS (Distributed Denial of Service) attacks.
Accountability (including non-repudiation)	Comprehensive account management mechanisms shall be established to: identify account types, establish conditions for group's membership and assign associated authorizations. Account control mechanisms to support procedures shall be properly developed, documented and implemented, to authorize and monitor the use of guest/anonymous accounts and to remove, disable, or otherwise secure, unnecessary accounts. Code signing and code access authorization may mitigate accountability issues.
Authentication/ Authorization	Process to validate a user's logon information shall be enforced to manage the access to restricted area. The problems faced during authentication include encryption, transmittal, and storage of passwords, session re-playing, and spoofing. A log control is very important in this phase to allow posterior auditing. Authorization is about determining what resources an authenticated person has access to the system. Accumulation of many privileges over time is a serious problem the system administrator should be concerned.

5.4 Software Quality Attributes

- **UI/UX:** If user experience is good then more users will come, and user willstay longer. This can be achieved by good user interface.
- **Efficiency:** Efficient use of database, storage and computational power enhances the quality of software. This also reduces cost and increasessecurity.

Analysis Models: Work Break Down Structure

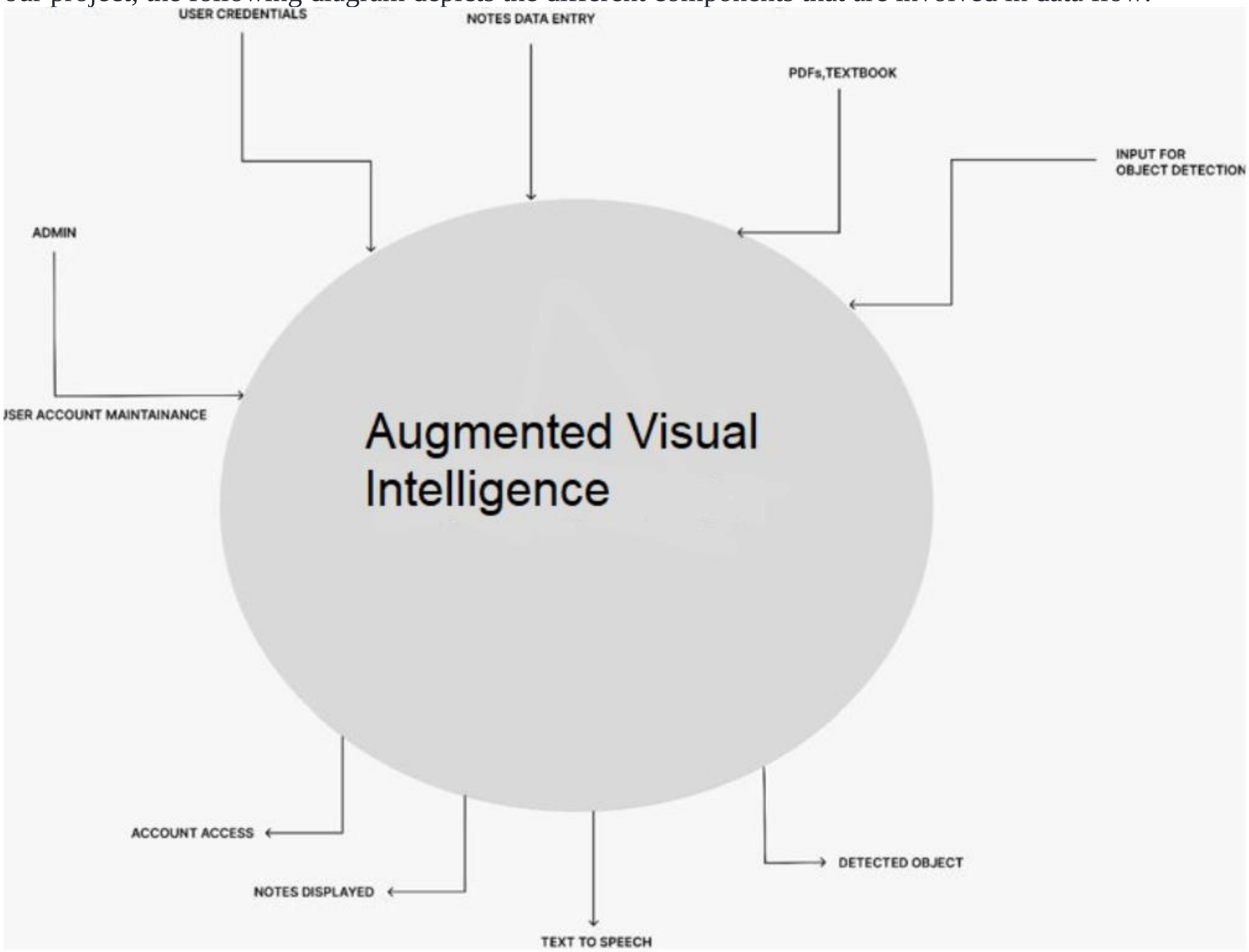


3.2 Data Modelling

A Data Flow Diagram (DFD) is like a map that shows how information flows in a process or system. Different symbols, like boxes, circles, arrows, and words, are used to show where the data comes from, where it goes, and how it moves around.

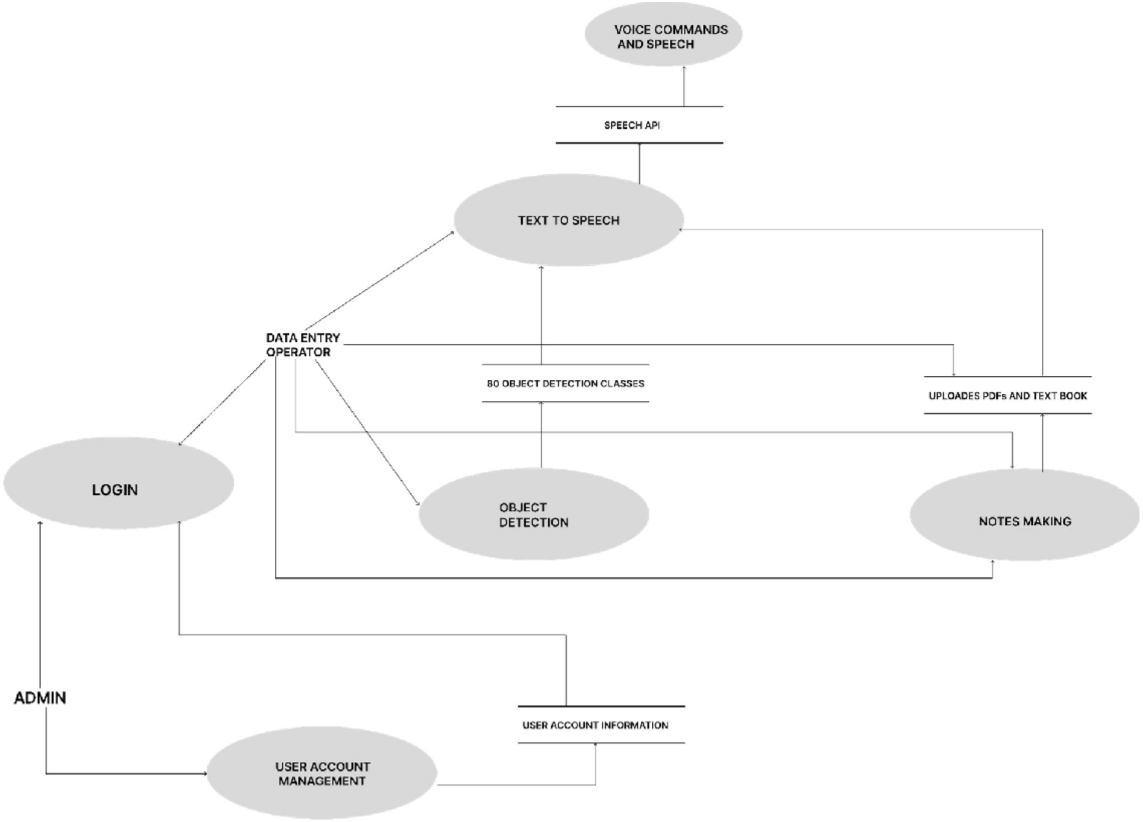
Level 0 DFD

Level 0 of a Data Flow Diagram (DFD) is also known as a Context Diagram. This diagram provides a simple and quick overview of the entire system or process that is being analyzed or modeled. It shows the system as a single, high-level process along with its connections to external entities. In the context of our project, the following diagram depicts the different components that are involved in data flow.



Level 1 DFD

DFD Level 1 is a detailed representation of the high-level processes from the Context Level Diagram. It breaks down each process into sub-processes. For example, in our project, after logging in, the user's credentials are verified, and then they can access modules like object detection, text-to-speech, or note-making through voice commands. These modules have data entry operators that take input data and generate the desired output. The diagram below depicts the flow of data within these processes.

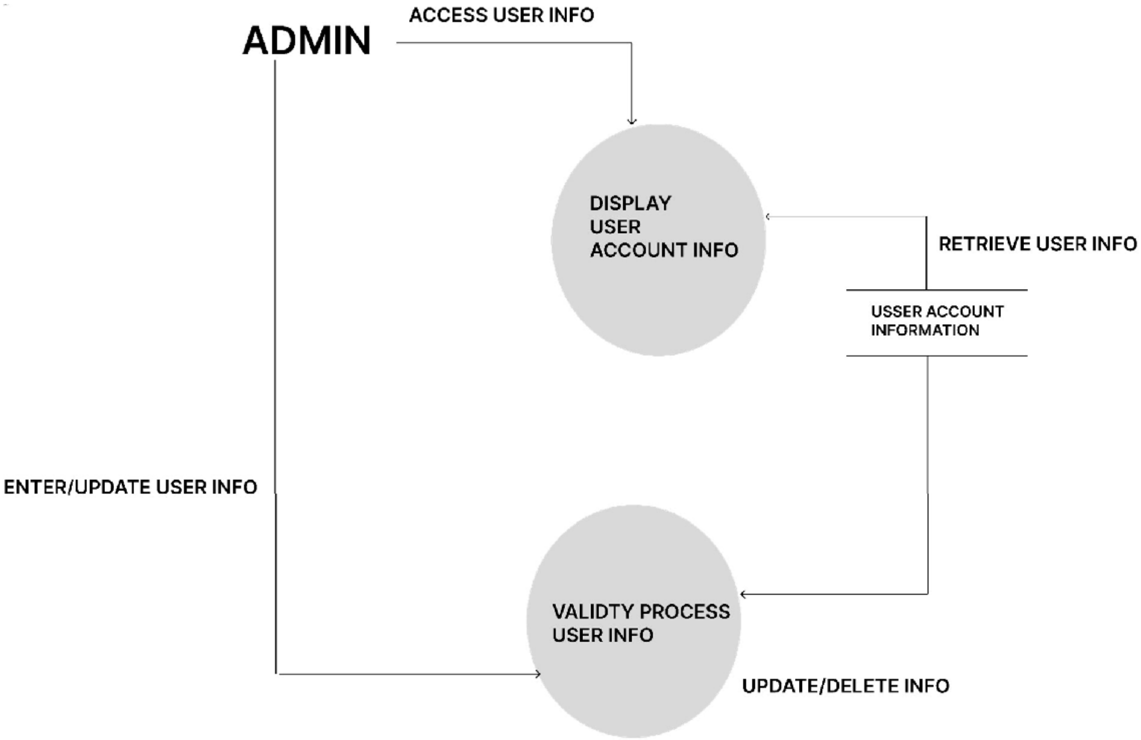


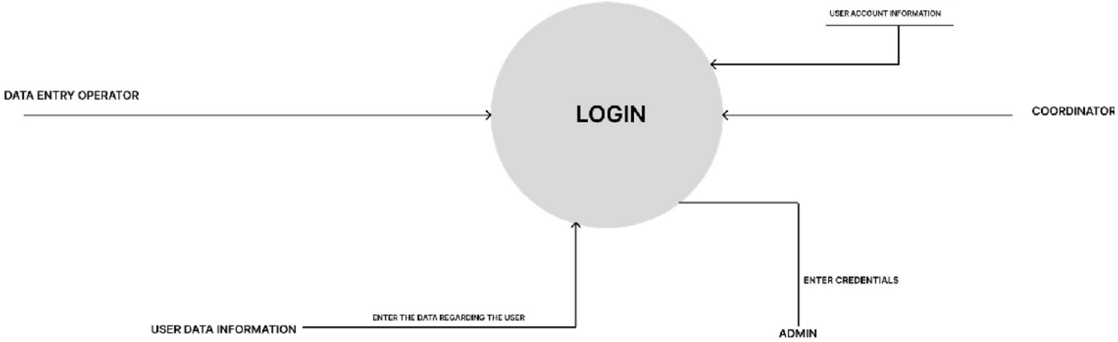
Level 2 DFD

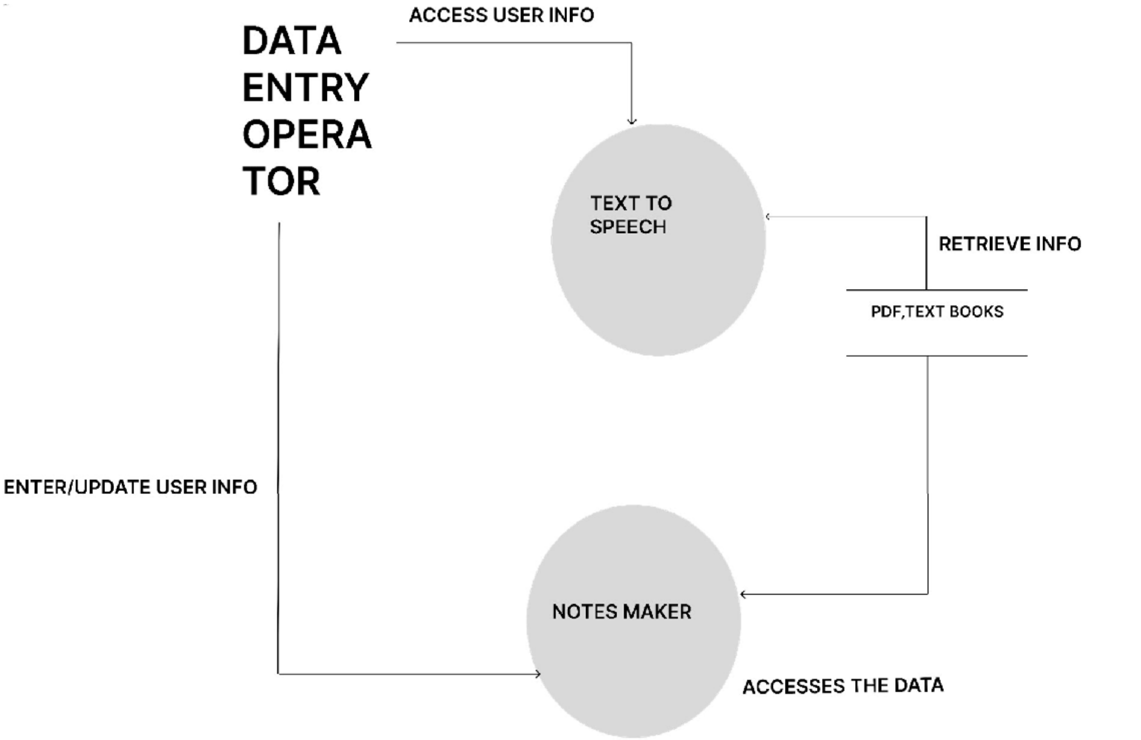
The Level 2 DFD provides an even more detailed breakdown of the Level 1 diagram. It shows the specific processes and data flows involved in the system.

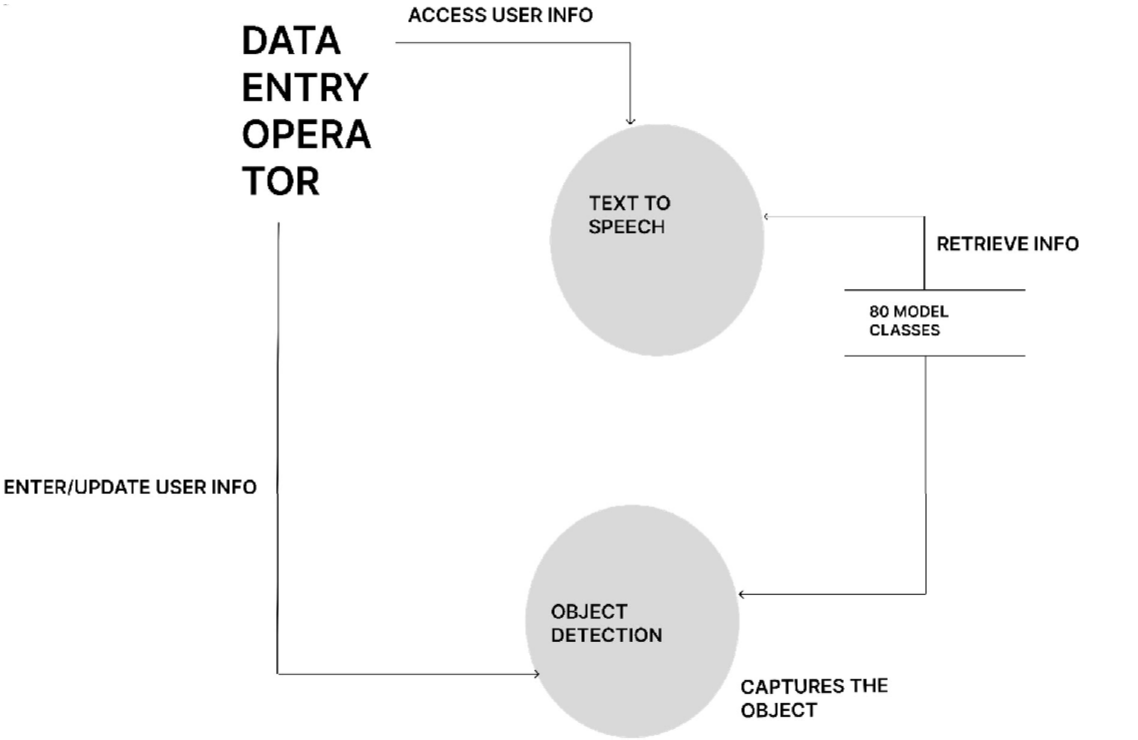
In this diagram, the Admin has the ability to access, enter, delete or update User Information, which is stored in the database. When a User logs in, their credentials are authenticated against the database before they can access the website. The Admin and coordinator can also verify the data in the database.

When a User accesses the text-to-speech module, information on stored pdfs is retrieved from the database. The note maker also accesses the database to store new notes. The text-to-speech module and Object Detection module both retrieve data from the 80 model classes to correctly identify and convert text into speech.


ADMIN
ACCESS USER INFO
DISPLAY
USER
ACCOUNT INFO
RETRIEVE USER INFO
USSER ACCOUNT
INFORMATION
ENTER/UPDATE USER INFO
VALIDTY PROCESS
USER INFO
UPDATE/DELETE INFO


USER ACCOUNT INFORMATION
DATA ENTRY OPERATOR
LOGIN
COORDINATOR
ENTER CREDENTIALS
ENTER THE DATA REGARDING THE USER
USER DATA INFORMATION
ADMIN

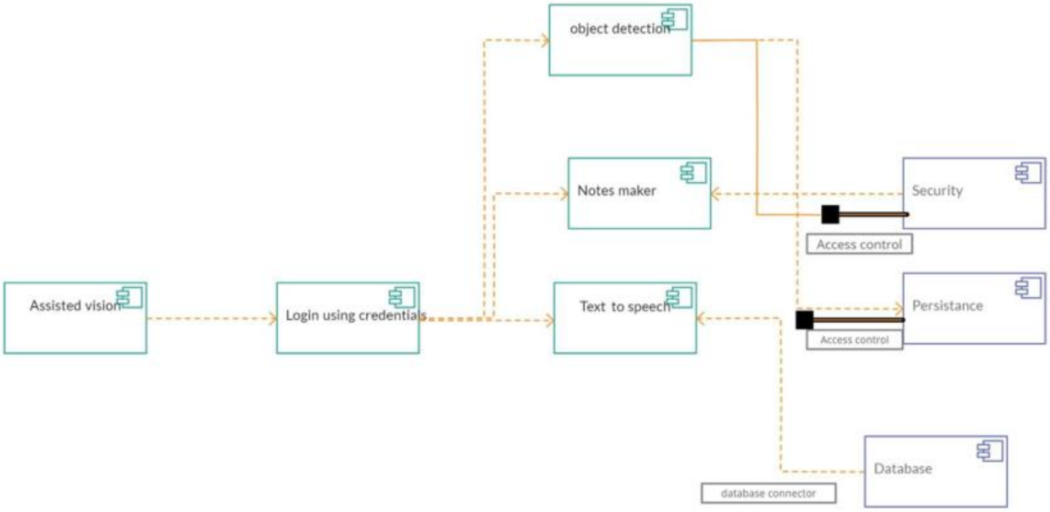

ACCESS USER INFO
DATA ENTRY OPERATOR
TEXT TO SPEECH
RETRIEVE INFO
PDF,TEXT BOOKS
ENTER/UPDATE USER INFO
NOTES MAKER
ACCESSES THE DATA


ACCESS USER INFO
DATA ENTRY OPERATOR
TEXT TO SPEECH
RETRIEVE INFO
80 MODEL CLASSES
ENTER/UPDATE USER INFO
OBJECT DETECTION
CAPTURES THE OBJECT

3.3 Structural Analysis

COMPONENT DIAGRAM

A component diagram is a visual representation of the different parts of an object-oriented system, breaking it down into smaller components for easier management. It depicts the physical view of the system, including executables, files, and libraries, as well as their relationships and organization within the system. In our website, the main components include Object Detection, Note Maker, and Text-to-Speech, with the database connected to TTS. These components are available once the user logs in to the website. The diagram below illustrates the relationships and organization of these components.

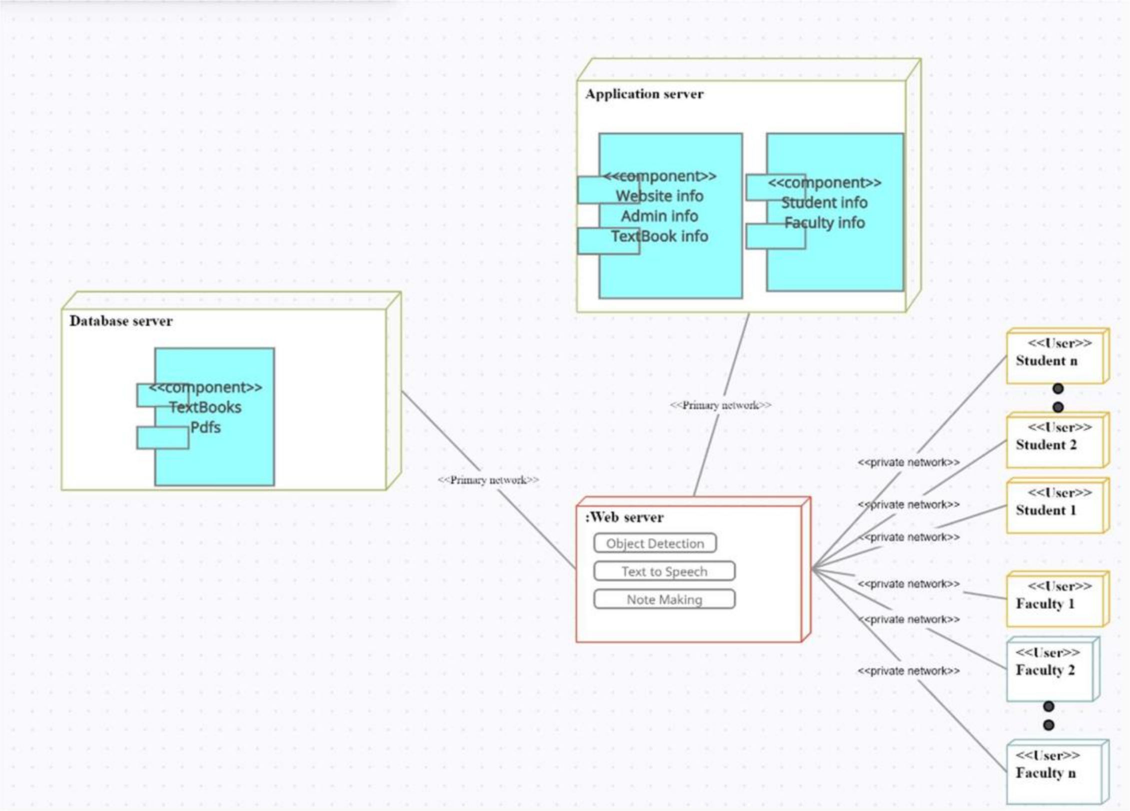


DEPLOYMENT DIAGRAM

A deployment diagram is a type of UML diagram that illustrates the physical architecture of a system, including the hardware or software environments where the system is executed, as well as the middleware that connects them.

The diagram below displays the physical architecture of the system, showing the different nodes, such as hardware or software execution environments, and the connections between them. It provides a visual representation of how the system is deployed in the physical world.

In this deployment diagram, the application server node includes both the business information layer and the physical layer, while the database node is responsible for storing and managing data. The middleware connections between the nodes ensure that they can communicate with each other and function as a cohesive system.



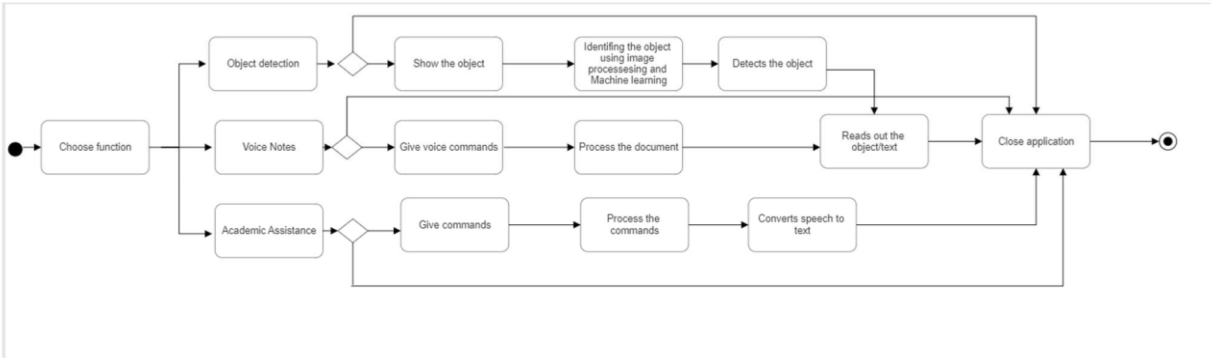
Chapter 4 Designs

4.1 UML Designs

<u>ACTIVITY DIAGRAM</u>

The following diagram shows how a system controls the flow of its activities, both concurrently and sequentially, using an activity diagram. It illustrates the workflow of the system by focusing on the conditions and sequence of its activities. The diagram includes activities and links to represent various functions.

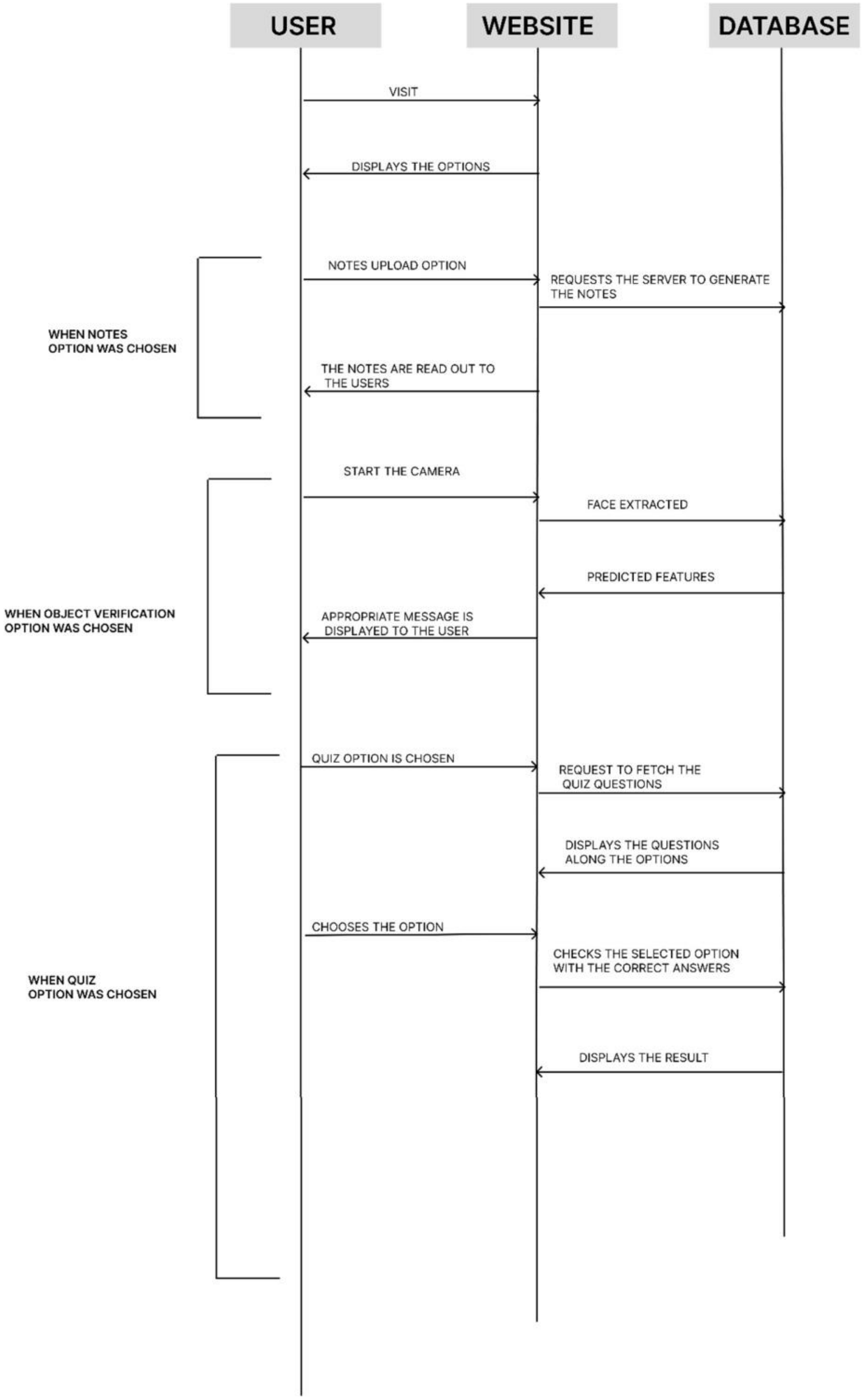
In our project, there are three main functionalities: object detection, voice notes, and academic assistance. The object detection module captures objects, identifies them using image processing and machine learning, and finally announces the name of the identified object using voice commands. The second feature is voice notes, where the user's voice commands are recorded as notes.



SEQUENCE DIAGRAM

The sequence diagram visualizes the communication between lifelines and the flow of messages in a system. It is also known as an event diagram and helps to depict dynamic scenarios. The sequence of events between two lifelines is displayed in a time-ordered manner based on their participation during runtime.

The sequence diagram for our website is shown below. It depicts the sequence of activities when a user interacts with the website and how the website accesses and modifies data in the database.

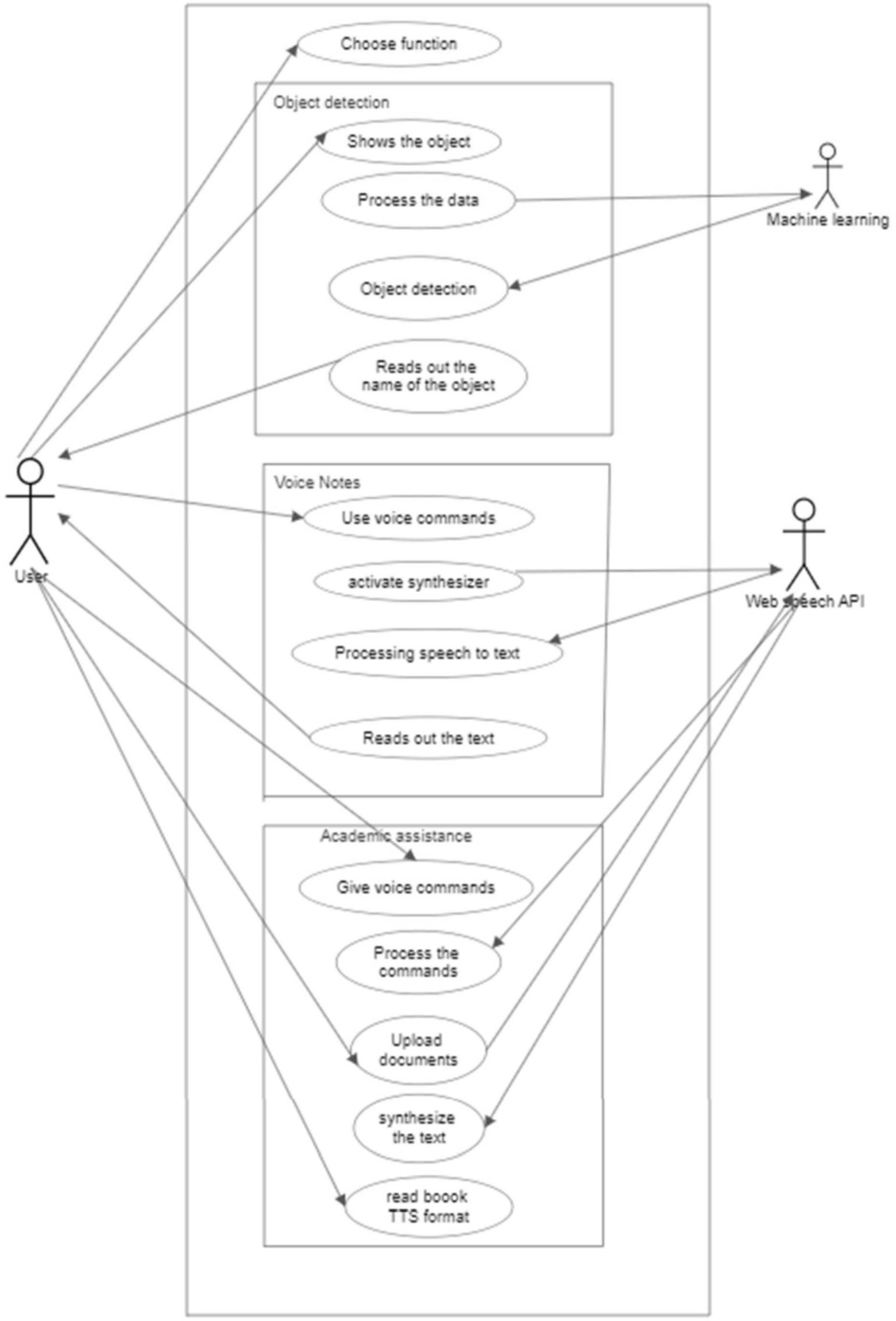

USER
WEBSITE
DATABASE
VISIT
DISPLAYS THE OPTIONS
NOTES UPLOAD OPTION
REQUESTS THE SERVER TO GENERATE THE NOTES
WHEN NOTES OPTION WAS CHOSEN
THE NOTES ARE READ OUT TO THE USERS
START THE CAMERA
FACE EXTRACTED
PREDICTED FEATURES
WHEN OBJECT VERIFICATION OPTION WAS CHOSEN
APPROPRIATE MESSAGE IS DISPLAYED TO THE USER
QUIZ OPTION IS CHOSEN
REQUEST TO FETCH THE QUIZ QUESTIONS
DISPLAYS THE QUESTIONS ALONG THE OPTIONS
CHOOSES THE OPTION
CHECKS THE SELECTED OPTION WITH THE CORRECT ANSWERS
WHEN QUIZ OPTION WAS CHOSEN
DISPLAYS THE RESULT

USE CASE DIAGRAM

The following diagram is a Use case diagram that represents the dynamic behaviour of a system, encapsulating its functionality by using use cases, actors, and their relationships. It specifically shows a Use case diagram for a User who interacts with our website. The User is presented with different options to choose from, such as Object Detection, Voice Notes, and Uploading Documents.

In the Object Detection use case, the User takes a picture of an object, which is then processed by the machine learning module. The name of the object is detected and read out to the user. Similarly, in the Voice Notes use case, the User gives a voice command, which is processed by the Web Speech API. The text output is then read out to the user.

When a User uploads a document, the text in the document is synthesized by the Web Speech API, and the PDF is read out to the user.



Chapter 5 Development

5.1 Tools Description and Development approach used

We have developed the front-end of our website using HTML, CSS, and React. The homepage is divided into three sections:

i) Lander: This section contains the title and a brief description of our project.

ii) Features: Our project offers three features, each of which is represented as a box that can be accessed by clicking on it.

iii) About us: This section provides a more detailed overview of our project's aims and objectives.

We have implemented object detection using the Tensorflow COCO-SSD model, which is a single shot multi-box detection process that can localize and identify multiple objects within a single image. The model is trained to detect objects from the COCO dataset, which is a large-scale object detection, segmentation, and captioning dataset. The model can detect 80 different classes of objects.

For PDF text extraction, we are using a cross-platform JavaScript module called "pdf-parse." This module is used server-side with the help of Node.js to extract text from PDF files. The extracted text is then passed through a JavaScript speech synthesizer to convert it into TTS (Text-to-Speech) format.

Voice notes are another important feature of our project, for which we have utilized JavaScript's speech synthesizer to read voice commands and take notes on the system.

5.2 Pseudocodes of Important Modules:

Object Detection:

1. Firstly, import necessary packages including react, tensor-flow, coco-ssd, webcam, etc.
2. Asynchronously run coco-ssd for continuous object detection.
3. Get the webcam feed from the device.
4. Set the video height and width to display to the user.
5. Detect objects by reading the video through the function “await net.detect(video);”
6. Finally, draw a box around all the objects displayed on the screen using the canvasRef method and display the detected object's name on the screen.

Pdf-TTS:

1. First, the user uploads the PDF document using the HTML form component.
2. Once the user selects the PDF, the file is retrieved and sent to the server as formData using the POST method. The server processes the data and responds back with the extracted text.
3. On the server side, the "PDF-parse" module performs the text extraction from the PDF document.
4. The available voices in the user's browser are populated in an array called voices[]. The user can select any voice from the options available.
5. Finally, the text received from the server is accessed by the speech synthesizer and converted into TTS (Text-to-Speech) format for the user to hear.

Notetaking-STT:

- To activate the speech-to-text functionality, the user can either give the correct command "start typing" or click on a designated button.

- Once activated, the user can start talking and the software will convert their speech to text format.

- Users can make changes to the text using the keyboard.

- To stop taking notes, the user can give the correct command "stop typing" or click on the designated button.

Chapter 6 Testing

6.1 Test cases for Important Modules

Functional Testing:

1. Check if the latest versions of react, node.js, mongo DB, and other required software are installed.
2. Verify all dependencies are up to date and there are no compatibility issues.
3. Ensure there are no vulnerabilities affecting the application's functionality.
4. Verify that the code is free of bugs and warnings in the terminal.
5. Confirm that the application pops up in the browser with the "npm start" command in the project directory.
6. Verify that clicking the "get started" button on the landing page displays the available features.
7. Check that clicking the "object detection" box opens a new window and starts running the object detection module.
8. Verify that the browser has permission to access the camera and can use it.
9. Check if objects are detected when they are placed in front of the camera.
10. Verify that the text-read server-side code is operational.
11. Verify that speech synthesis voices are loaded onto the select options.
12. Verify that files of all sizes can be uploaded to the server and texts are extracted.
13. Check that the speech synthesis can read all the texts and that the pause and stop buttons work.
14. Verify that the notes component can capture the user's voice and make notes on the screen.

Non-functional testing:

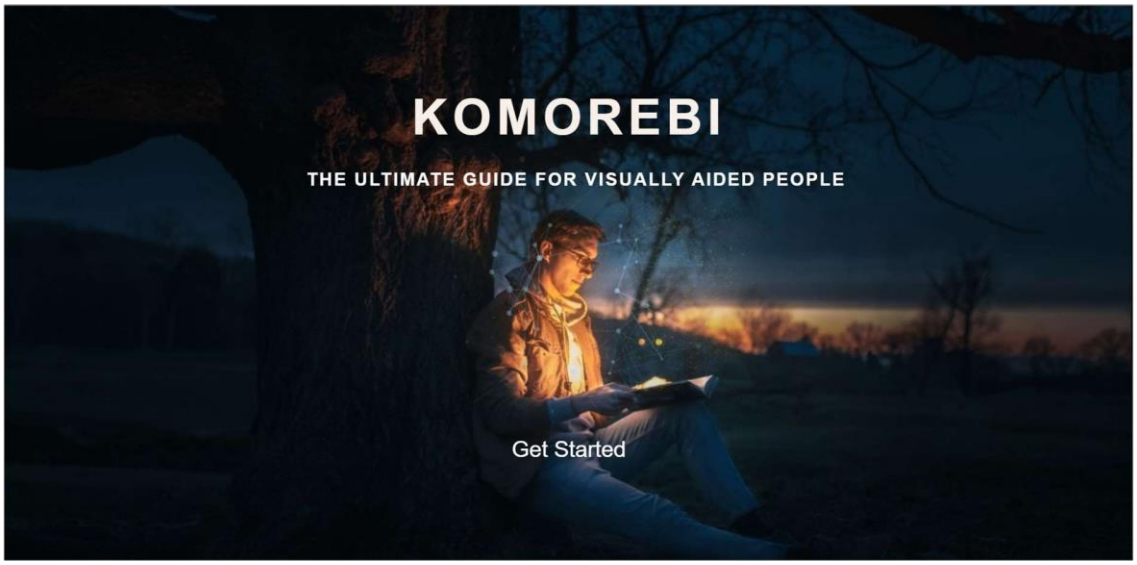
1. Evaluate the accuracy of the object detection model through testing
2. Test the performance of the model by measuring its time and space complexity
3. Gather user feedback on the UI, design, and overall usability of the application
4. Verify the responsiveness of the system by testing it on various screen sizes and devices to ensure proper functionality.

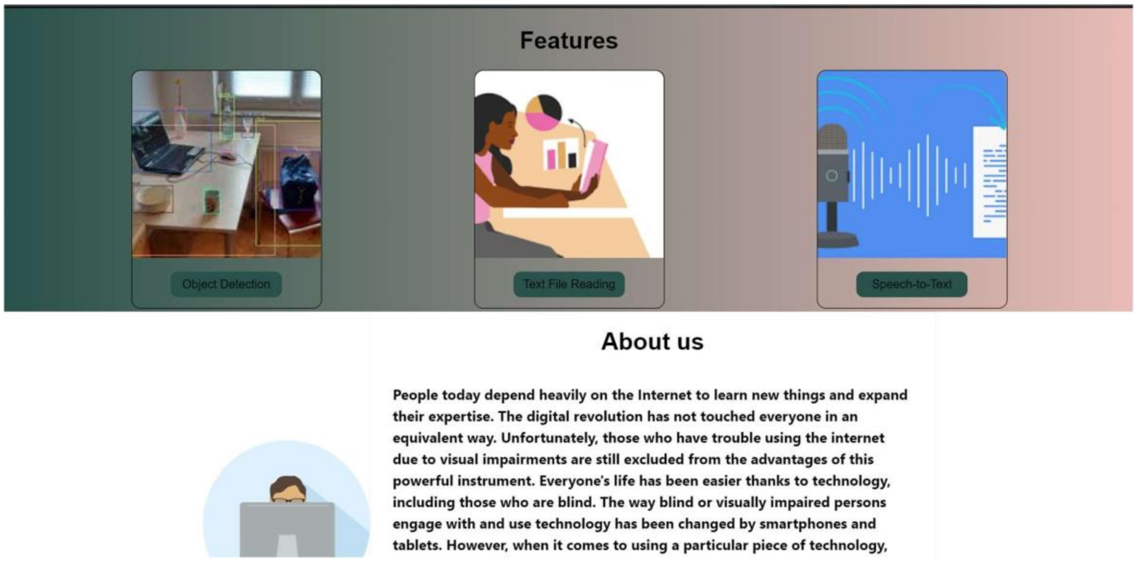
Test Conditions:

Test caseID	Test Case description	Expected Outcome	Actual Outcome	Test Status
1	“npm start” command open application in a browser	The app opens with only the lander page visible	The app was opened in default browser	SUCCESS

			showing the lander page	
2.	Show objects like bottle, phone, person, car to the camera	The cocossd model should detect all the objects	All the objects were detected.	SUCCESS
3.	Upload file on using the html form	The file will be uploaded andthe text from the pdf will be extracted.	The text from the uploaded PDF file was extracted.	SUCCESS
4.	Inside the voice notes module, speak anything	The module should be capable of understandingit and making it as notes	The voices were understood by the system and aformatted voice notes was made	SUCCESS

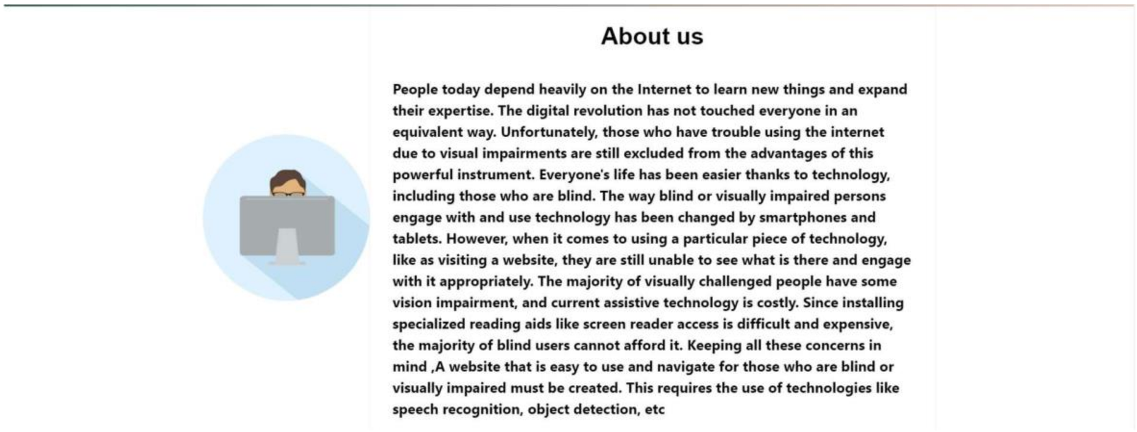
Chapter 7 Screenshots of Developed Product





About us

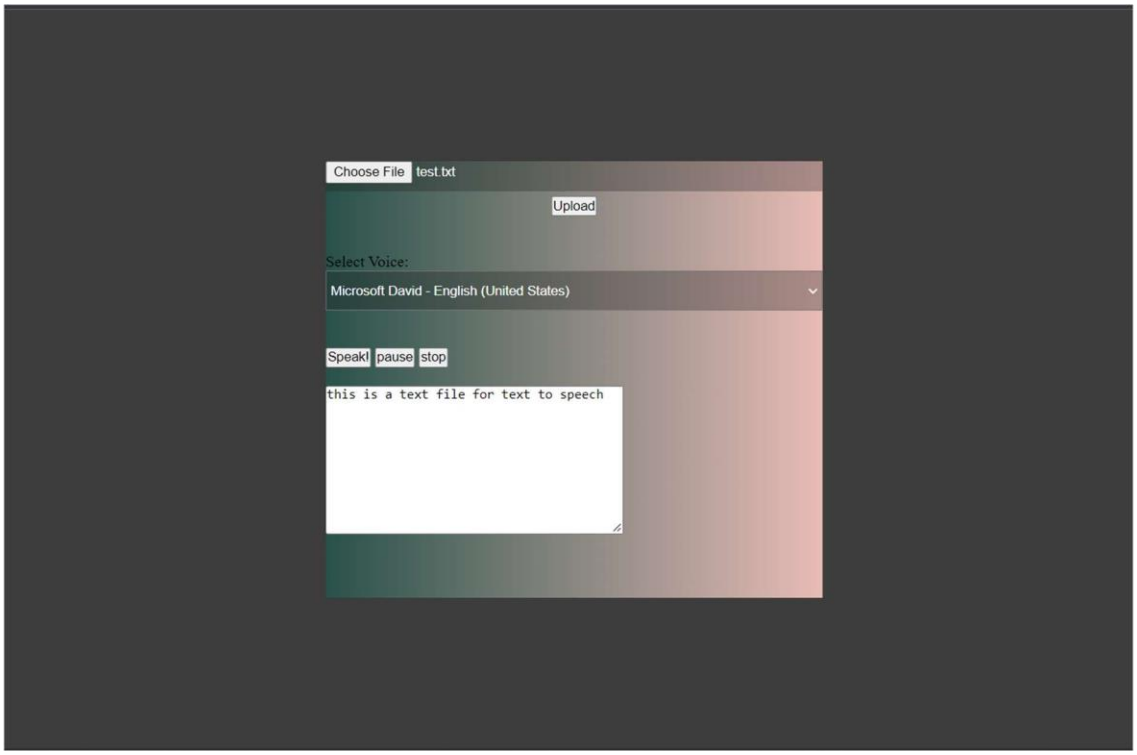
People today depend heavily on the Internet to learn new things and expand their expertise. The digital revolution has not touched everyone in an equivalent way. Unfortunately, those who have trouble using the internet due to visual impairments are still excluded from the advantages of this powerful instrument. Everyone's life has been easier thanks to technology, including those who are blind. The way blind or visually impaired persons engage with and use technology has been changed by smartphones and tablets. However, when it comes to using a particular piece of technology,



About us

People today depend heavily on the Internet to learn new things and expand their expertise. The digital revolution has not touched everyone in an equivalent way. Unfortunately, those who have trouble using the internet due to visual impairments are still excluded from the advantages of this powerful instrument. Everyone's life has been easier thanks to technology, including those who are blind. The way blind or visually impaired persons engage with and use technology has been changed by smartphones and tablets. However, when it comes to using a particular piece of technology, like as visiting a website, they are still unable to see what is there and engage with it appropriately. The majority of visually challenged people have some vision impairment, and current assistive technology is costly. Since installing specialized reading aids like screen reader access is difficult and expensive, the majority of blind users cannot afford it. Keeping all these concerns in mind ,A website that is easy to use and navigate for those who are blind or visually impaired must be created. This requires the use of technologies like speech recognition, object detection, etc

Text to Speech



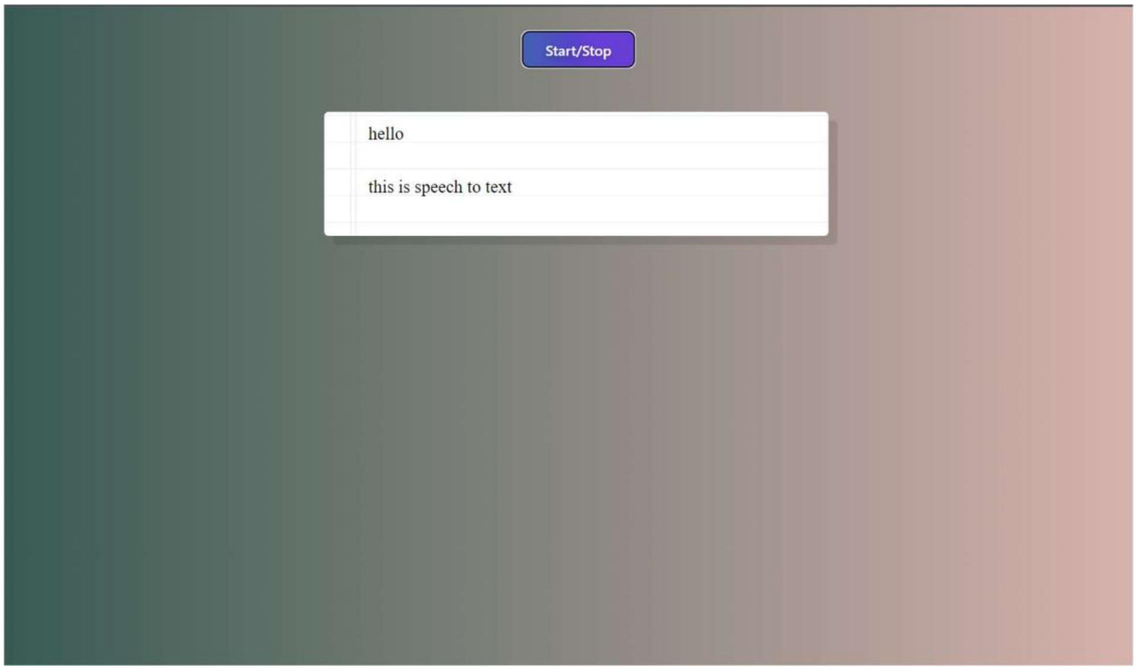
test.txt

Select Voice:

Microsoft David - English (United States)

this is a text file for text to speech

Speech to text note taking



hello

this is speech to text

Object Detection

bottle

scissors

Chapter 8 Implementation

Object Detection:

```
import React, { useRef, useState, useEffect } from"react";

import * as tf from "@tensorflow/tfjs";

import * as cocossd from "@tensorflow-models/coco-ssd";import

'@tensorflow/tfjs';

import Webcam from "react-webcam";import

"./App.css";

import { drawRect } from './utilities';

// import {Helmet} from 'react-helmet';function Odetect()

{

   const webcamRef = useRef(null);const

   canvasRef = useRef(null);




   const runCoco = async () => {


      const net = await cocossd.load();

      //   Loop and detect hands

      setInterval(() => {

         detect(net);
}, 10);
```

```
};


const detect = async (net) => {
  // Check data is availableif (
    typeof webcamRef.current !== "undefined" &&
    webcamRef.current !== null &&
    webcamRef.current.video.readyState === 4
  ) {
        // Get Video Properties
    const video = webcamRef.current.video;const videoWidth =
  webcamRef.current.video.videoWidth;
    const videoHeight = webcamRef.current.video.videoHeight;


    // Set video width webcamRef.current.video.width = videoWidth;
    webcamRef.current.video.height = videoHeight;


    // Set canvas height and width canvasRef.current.width = videoWidth;
    canvasRef.current.height = videoHeight;


    // 4. TODO - Make Detections
    // e.g. const obj = await net.detect(video);
```

```
        const obj = await net.detect(video);console.log(obj);

        // Draw mesh
        const ctx = canvasRef.current.getContext("2d");

        // 5. TODO - Update drawing utility
        // drawSomething(obj, ctx)
        drawRect(obj, ctx);
    }

};

useEffect(() => { runCoco() }, []);

return (
    <div>
        <header  className="App-header">
            <Webcam ref={webcamRef}
                muted={true} style={{
                    position:  "absolute",marginLeft:
                    "auto",
                    marginRight:  "auto",left: 0,
                    right: 0,
                    textAlign:  "center",zindex: 9,
                    width: 640,
                    height: 480,
                }}
            />

            <canvas ref={canvasRef}
                style={{
                    position:  "absolute",marginLeft:
                    "auto", marginRight: "auto",
                    left: 0,
                    right: 0,
                    textAlign:  "center",zindex: 8,
                    width: 640,
                    height: 480,
                }}
            />
        </header>
```

```
    </div>
  );
}

export default Odetect;
```

Notes Module:

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Speech Detection</title>
    </head>
    <body>
        <button class="button-36" role="button"
onclick="btn_talk()">Start/Stop</button>

        <div class="words" contenteditable>
        </div>

        <script>
            var isType = false;

            window.SpeechRecognition =window.SpeechRecognition ||
window.webkitSpeechRecognition;
```

```
          const recognition = new SpeechRecognition();

          recognition.interimResults = true; recognition.lang = 'en-US';


          let p = document.createElement('p');const words =
        document.querySelector('.words');

              words.appendChild(p);


          recognition.addEventListener('result',  e  =>  {var transcript =

              Array.from(e.results)

              .map(result => result[0])

              .map(result  =>  result.transcript)

              .join('');


              if(transcript.includes("start  typing"))  {

                  console.log("start " + transcript + "
" + isType);

                  isType = true;

                  // speak("voice to text started");

              }

              else  if(transcript.includes("stop
typing")) {

                  console.log("stop " + transcript + "
" + isType);

                  isType = false;

                  // speak("voice to text closed");
```

```
            }

            else if(transcript.includes("go back") &&
!isType) {

                  history.back();

            }


            if(isType) {

                const text = transcript.replace(/start  typing|stop
typing/gi,  '');

                p.textContent  =  transcript;


                if  (e.results[0].isFinal)  {

                    p  =  document.createElement('p');words.appendChild(p);

                }

            }

        });

        recognition.addEventListener('end',recognition.start);

        recognition.start(); function

        speak(instruction)  {

            const text_speak = new
SpeechSynthesisUtterance(instruction);

            text_speak.rate = 1;

            text_speak.pitch = 1; window.speechSynthesis.speak(text_speak);

        }
```

```
            function btn_talk() {isType =
                !isType;
            }
        </script>
      <style>
            html {
                        font-size: 10px;
            }
            body {
                /* background-color: rgb(41, 47, 85); */
       /*  background-image:  linear-gradient(to  left,#2c3e50, #4c75af); */
                /* background-image: url(bg.jpg); */font-family:
                'helvetica neue';
                font-weight:  200;font-
                size: 20px;
                background-image:   linear-gradient(toright, #29524A,
#E9BCB7);
            }
            .words {
                 max-width: 500px; margin:
                 50px auto;background:
                 white;border-radius:  5px;
                 box-shadow: 10px 10px 0 rgba(0,0,0,0.1);padding: 1rem
                 2rem 1rem 5rem;
```

```
                        background:  -webkit-gradient(linear,  0
0, 0 100%, from(#d9eaf3), color-stop(4%, #fff)) 04px;

                        background-size: 100% 3rem;position:

                        relative;

                        line-height:  3rem;

              }

              p {

                    margin: 0 0 3rem;

              }

              .words:before {

                        content: ''; position:

                        absolute;width: 4px;

                        top: 0; left:

                        30px;bottom: 0;

                        border: 1px solid;

                    border-color:  transparent  #efe4e4;

              }


              .button-36 {

                      margin-left:  46%;margin-

                      top:  20px;

              }
```

```
/* #listening { margin-top:

    10px;height: 20px;

    width: 20px;

} */


.button-36 {

    background-image: linear- gradient(92.88deg, #455EB5
9.16%, #5643CC 43.89%, #673FD7
64.72%);

    border-radius: 8px; border-style:

    none; box-sizing:  border-box;

    color:  #FFFFFF; cursor: pointer;

    flex-shrink: 0;

    font-family: "Inter UI","SF ProDisplay",-
apple-system,BlinkMacSystemFont,"Segoe
UI",Roboto,Oxygen,Ubuntu,Cantarell,"Open Sans","HelveticaNeue",sans-serif;

    font-size: 16px; font-

    weight: 500; height: 4rem;

    padding: 0 1.6rem;text-

    align:  center;

    text-shadow: rgba(0, 0, 0, 0.25) 0 3px
8px;

    transition: all .5s;
```

```
                user-select:  none;

                -webkit-user-select: none;touch-action:

                manipulation;

            }


            .button-36:hover  {

                box-shadow: rgba(80, 63, 205, 0.5) 0 1px
30px;

                    transition-duration:  .1s;

            }


            @media (min-width: 768px) {

                    .button-36 { padding: 0

                    2.6rem;

                }

            }


        </style>

    </body>

</html>
```

Text to Speech

```
const express = require("express");

const  fileUpload  =  require("express-fileupload");const pdfParse =
require("pdf-parse");
```

```
const app = express();


app.use("/",   express.static("public"));app.use(fileUpload());


app.post("/extract-text", (req, res)=>{ if(!req.files  &&
    !req.files.pdfFile){
        res.status(400);res.end();
    }


    pdfParse(req.files.pdfFile).then(result =>{res.send(result.text);
    });
});
app.listen(5000);
```

Conclusion:

Visually impaired individuals often face a wide range of difficulties due to their lack of vision. In response to this, our project, called "Augmented Visual Intelligence," aims to provide software that can offer academic assistance, object detection, and text-to-speech capabilities to help improve their daily lives. By using this software, visually impaired individuals will be able to access tools that can make their lives easier and more convenient.

Our project aims to fill a gap in the market by providing a comprehensive solution to assist visually impaired individuals. We understand that these individuals face significant challenges and barriers that can limit their access to various tools and services. Through our project, we hope to empower these individuals and provide them with the necessary tools to help them overcome their challenges and improve their quality of life.