ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

NEUMANN JÁNOS
FACULTY OF INFORMATICS

DEGREE PROJECT

THESIS

OE-NIK	Student's name:	**Vania Margareth Tavares Fernandes**
2022	Student's registration number:	**T/007199/FI12904/N**

Óbuda University
Neumann János Faculty of Informatics
Institute of Applied Mathematics

THESIS
WORKSHEET

Student's name: **Vania Margareth Tavares Fernandes**

Student's name: [redacted]
Registration number: T/007199/FI12904/N

Title of study:

Conceptual Design and User Interface Implementation for the CruzR Service Robot

Institutional supervisor: Dr. Galambos Péter

Submission deadline: 2022. May 15.

Subjects of the final examination: Cloud Computing System and Mobile Application I-II

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Neumann János Faculty of Informatics

STUDENT'S DECLARATION

I the undersigned student hereby declare that this degree project/thesis is the result of my own work; I have disclosed the references and tools used in an identifiable manner. The results indicated in my degree project/thesis completed may be used by the university and the institution announcing the task for their own purposes free of charge.

Dated: Budapest, 15 May 2022

Vania Fernandes
....
student's signature

Óbuda University
John von Neumann Faculty of Informatics
Institute of Biomatics and Applied Artificial Intelligence

DIPLOMA THESIS
TOPIC DESCRIPTION

Student name: **Vania Margareth Tavares Fernandez**
Registration number: T/007199/FI12904/N
Neptun's code: [redacted]

Title of diploma thesis:

Conceptual Design and User Interface Implementation for the CruzR Service Robot
Felhasználói felület koncepcionális tervezése és megvalósítása a CruzR szolgáltató robothoz

Internal supervisor: Dr. Péter Galambos
External supervisor

Deadline: 15th of May, 2021

Topic description:

Service robotics is an emerging field with large business potentials. Several manufacturers appear on the market with a wide selection of robots for various purposes. A major class of these robots is the wheeled mobile platforms with a humanoid torso. In late 2019, Óbuda University purchased a UBTech CruzR robot to start education projects and business development involving humanoid service robots. The goal of this thesis project is to design and implement the user interface of a body temperature screening service robot that is also able to interact with people in environments like e.g., hospitals, airport lobbies, etc. The work should be conducted in teamwork cooperating with other students who are working on software logic design and implementation.

The diploma thesis should cover:

- literature review on service robot use cases and software solutions for human-robot interaction,
- a survey on the modern UI prototyping tools,
- precise specification of the body temperature screening application,
- the UI design in form of storyboard, wireframe, and mockup,
- implementation of the UI using web technologies (in accord with the team's decision),
- report of usability testing and demonstration,
- evaluation of the results and future work.

ÓBUDAI EGYETEM Neumann János Informatikai Kar Ph. 1. Budapest

....
Dr. György Eigner
acting head of institute

Term of limitation: **15th of May, 2023.**

The diploma thesis is adequate and can be submitted:

....
external supervisor

....
internal supervisor

CONSULTATION LOG

Student's name: Vania Margareth Tavares Fernandes

Neptun code: [redacted]

Specialty: MSc. Computer Science Engineering

Phone number:

Mailing address (e.g., domicile):

[redacted]

Degree project / thesis[1] title in Hungarian:

Felhasználói felület koncepcionális tervezése és megvalósítása a CruzR szolgáltató robothoz

Degree project / thesis[2] title in English:

Conceptual Design and User Interface Implementation for the CruzR Service Robot

Institutional supervisor: ....Dr. Péter Galambos

External supervisor:

Please write the data in printed capital letters

Occ.	Date	Content	Signature
1.	23 Sep 2021	Feature specifications to be implemented in the final version	
2.	18 Nov 2021	Thesis status update and modification of speech recognition feature to text to speech	
3.	26 Nov 2021	Thesis code update and improvements required	
4.	10 Mar 2021	Thesis essay update and improvements required	

The Consultation log is required to be countersigned by either supervisor on a total of 4 occasions of consultation.

The student has complied with the requirements of the subject titled Degree project I / Degree project II (BSc) or Thesis 1 / Thesis 2 / Thesis 3 / Thesis 4[3], and is allowed to proceed for reporting / defense[4].

Dated: Budapest, 11th May 2022.

....
institutional supervisor

[1] Underline as appropriate
[2] Underline as appropriate
[3] Underline as appropriate
[4] Underline as appropriate

Contents

Chapter 1

Abstract

There is a demanding interest in having service social robots leave the factory and labs to help resolve our society's essential concerns, ranging from entertainment, health care, and education to user support.

These so-called social robots are designed to interact, work, collaborate, and improve people's lives by engaging with humans on an interpersonal and socially adequate level. They usually need to covey some form of human intelligence capability, as well as be able to acquire new relevant skills after deployment.

However, it is quite complex to program these social robots with every human social skill they need to interact and have a valuable role within society. Thus, by taking advantage of Artificial Intelligence and Machine Learning technologies, it becomes possible to succeed in developing robots that can mimic human cognitive and social behaviors.

Key-Words: social robotics, artificial intelligence, convolution neural network, natural language programming, human-robot interaction, social robot user interface experience, AI web application, face detection, speech synthesis, performance evaluation, etc.

Chapter 2

Introduction

This thesis work aims to create and implement the **user interface of a social robot named Cruz Robot** to interact with people in social settings such as universities, hospitals, airports, lobbies, and others. It was conducted in teamwork collaboration with other students, who were also working on other parts of the software logic design and implementation of Cruz.

We will focus specifically on the **Cruz Robot User Interface** implementation design **built on top of web application technologies**. The developed web application will be integrated into Cruz Robot to run the **Face detection** and **Speech synthesis** features.

This project explores the emerging field of social robots by researching all and every surrounding area that contributes to its development to accomplish the goal of creating an AI (Artificial Intelligence) web application for the Obuda University Cruz Social Robot.

This work starts by explaining the concept of social robots in conjunction with a survey overview research of services robots existing nowadays. Some are successfully active social robots that inspire and serve as guidance for the Cruz development process.

Next, it describes the Human-Robot interaction (HRI) field, highlighting the critical features social robots should have to achieve human acceptance. In parallel, it explores the user interface components to develop social robots and how such components were applied in the Cruz web application.

Finally, we dive into the Cruz software development process. It discusses Cruz's Robot web application design, the technologies used in the development phase, and the expected future implication for Cruz Robot within society. Overall questions such as What are social robots? How do we measure human-robot interaction? How do social robots approach humans? What are the goals when designing the Cruz Robot web application? How will the Cruz robot web application impact humans? All these and more questions are examined in this thesis work.

The main goal is to build the web application for Cruz Robot by identifying the benefits and challenges of user interface development for social robots. For Cruz technologies to successfully integrate daily use in social environments is also necessary to gain users acceptance. Thus a positive user interface experience is vital.

Chapter 3

Social Robots

Before we dive into the world of social robots, a general definition of Robots is needed to establish the category where **social robots** can be integrated. According to the Robot Institute of America (RIA), **Robots** are: *"reprogrammable, multipurpose machines developed to move within its target environment. They are trained devices programmed to execute motions for the performance of a variety of tasks"*[1] (H. Allan Hunt, p.8). Simply put, they are machines in motion, programmed to execute various or specific tasks.

Additionally to the RIA, the the International Federation of Robotics (IFR) sets two primary utilities of robots:

1. **To Assist people:** for educational purposes, safeguarding, personal assistant, entertainment, etc.

2. **To serve as equipment:** for assistance in medical, cleaning, maintenance, repair, etc.

With that in mind, by technology standards, nowadays, it is possible to classify robots into two main classes, as we consider their functions and the target market. As such, we can now establish two distinct categories of robots: **social robots** and **industrial robots**.

Disclosing the Robotic Industries Association, definition: "*an industrial robot is a reprogrammable, controlled, multipurpose device in three or more axes that may be fixed in place or mobile for industrial automation applications*" [2](p.41). As we can see, within this context, the robot's purpose is to serve as equipment while controlled by humans(in most cases).

So how about social robots? Well, we can evidently deduce that social robots have the purpose of assisting humans. However, what are social robots exactly, and how do they own the ability to assist humans?

The field of **social robots concentrates on the design and development of robots that possess the ability functions to interact socially with humans**. This social attribute is a vital part of its functionalities to help humans achieve a specific goal.

The development of social robots mainly centers on designing Artificial Intelligence (AI) systems, able to communicate and cooperate with humans by **exchanging social interactions** such as coaching, companionship, motivating, educating, monitoring performance, and providing customer support to improve and facilitate people's life [3]. Furthermore, the unique capability of exchanging social interaction with humans differentiates a social robot from an industrial robot.

Social robots like Pepper, Jibo, Temi, Zenbo, Buddy, Nao, [4] and our own **Cruz Robot** were specifically developed with the intent to interact socially with humans, to elevate human care in a way that can save costs while providing a personalized and satisfying user experience.

The technology behind **Cruz robot contains social and cognitive intelligence capabilities** to interact with its users, which is designed in a way that causes users to recognize Cruz as a social "being" due to its intelligent attributes.

Nowadays, social robots are often integrated into social settings for professional use, therefore is not unlikely that eventually, one will cross paths with these robots at an office, airport, university, hotel, entertainment spaces, hospital, nursing care, therapy, and rehabilitation, etc. Thus the motivation and inspiration behind Cruz's development process.

3.1 Goals Of Social Robots

The goals of a social robot must be directly proportional to its functionality to assess [5]. However, regardless of the robot's specialized task, all social robots should be able to:

- **Interact**

  To interact with humans through speech, gesture, or both. Moreover, in some cases, they can portray enough expressions and emotions to meet human expectations.

  After all, one's social intelligence makes interaction possible. For social robots, this translates into creating algorithms that enable them with social capabilities such as face detection, speech detection, people tracking, object learning, etc. Each algorithm is implemented to execute a specific task.

- **Adapt**

  To portray cognitive intelligence capabilities to adapt behavior patterns within a social setting. Such ability is directly correlated with a social robot's ability to predict and adapt to its user response, translating into skills such as motor skills abilities, face, gesture or speech recognition, etc.

Chapter 4

A Survey On Social Robots Use Cases

Social robots now hold great value and promise for business and society, with competitive benefits regarding efficacy, quality, durability, speed, and cost. The introduction of social robots brings significant and positive changes within the User Interface (UI) area due to their impact gained by exchanging social interactions.

In computing, robotics encompasses different areas that can be addressed for the development process, focusing on the software used (which presents at a higher hierarchy level than the hardware) to how and what its functionalities aim. Some robots can even exist virtually, being made entirely of software.

Today we see a rapid development of Robot concepts modified for professional, entertainment, and consumer use. Robots, especially Social Robots, have traded our daily lives and, with the advance and changes driven by technology, it is expected that more and more people will cross paths with one. Therefore, to succeed in the web application development for the **Cruz robot**, survey research was conducted to find successful use cases of social robots developed through the years. This research aims to present some of the most incredible social robots and their functionalities existing nowadays [6].

This survey is done arbitrarily, as the social robots listed here were selected based on a user interface development perspective while focusing on how they have impacted humans, the culture, and the world of robotics. Below are some of the most amazing social robots developed for the entertainment, health care, consumer industry, education system, and science community.

Creator: Jibo Inc

Country: United States US

Year: 214

Type: UI

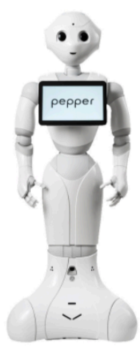
Target Market: Consumer, Family

Software Language: unknow

Figure 4.1: Jibo Robot

- **Description:** Jibo (Fig.4.1) is a friendly, adorable social robot designed to assist and interact with humans in a more personal manner, becoming one of the most successful family friend robots. Equipped with cameras, microphones, and a 3-axis motor system, he can spin 360 degrees, turn his face in any direction, and move freely [7]. This social robot is developed with algorithms that can execute tasks management duty, read the news stories, take photos, check the weather, dance, and make jokes.

- **Abilities:** Jibo (Fig.4.1) can pretty much be used as an intelligent assistant with the ability to integrate and engage in a unique way. Jibo (Fig.4.1), the social robot, is programmed with Natural Language Understanding (NLU) which enables the speech recognition system to recognize its user's speech and respond in a friendly way. It also includes a facial recognition system that provides specific users with notifications or reminders regarding their tasks, appointments, answering calls, answering questions, playing, and controlling users devices.



Creator: SoftBank Robotics

Country: Japan JP

Year: 2014/15

Type: Humanoids, UI and Research, Consumer,

Target Market: Entertainment, Both homes and businesses

Software Language: Python, Java and, JavaScript

Figure 4.2: Pepper Robot

- **Description:** In 2014 this social robot named Pepper (Fig.4.2) was developed in Japan by Aldebaran Robotics, but afterward acquired and sophisticated by Softbank Group

Corp in 2015 [8]. This social robot was designed to provide companionship to family homes and help customers at retail stores.

- **Abilities:** It is reported that Pepper (Fig.4.2) was the world's first social robot programmed with an algorithm able to understand human emotions. Pepper has succeeded tremendously in exchanging social interactions through its ability to have conversations. Its feature with speech recognition and dialogue system available in 15 languages, it is capable of giving directions and even has the ability to dance smoothly [9]. It speaks, gestures, and appears prompt to make everyone smile.

  The showcase of this social robot in 2015 was so successful and well-received by its target market that 12,000 Pepper robots had been sold by May 2018. In which at least 3,000 were integrated into businesses such as coffee shops, clothing stores, restaurants and bars, SoftBank shops, and even airports providing customer service support [10].

Creator: Temi Robotics

Country: United States

Year: 2018

Type: UI ,Telepresence, Consumer

Target Market: Family and Entertainment

Software Language: Kotlin or Java

Figure 4.3: Temi Robot

- **Description:** About 3 feet high, Temi (Fig.4.3) is an intelligent social robot assistant for home and families that respond accordingly with voice commands. It is equipped with an interactive 25 centimeters touchscreen display, real-time mapping, anti-collision system, path deduction, obstacle avoidance, and sensors to achieve autonomous navigation around people's homes and offices. This robot is considered the world's first personal social robot for your home or office [11].

- **Abilities:** Temi (Fig.4.3) exchanges social interactions with humans through speech, facial detection, and recognition, records and plays music, starts video chats, safeguards your home, helps with daily tasks, and controls devices such as the thermostat, turning the lights on and off, etc. This social robot can offer its users an easy way to access any web content, connecting them with families and friends [12].

Creator: Aldebaran Robotics and SoftBank Robotics

Country: France FR

Year: 2008

Type: Humanoids, UI and Research, Education,

Target Market: Entertainment and Healthcare

Software Language: Python, C++, and JavaScript

Figure 4.4: Nao Robot

- **Description:** With 0.57cm in height, NAO (Fig.4.4) social robot weighs around 4.5 Kg and, it was designed with an appealing looking resembling a human toddler, which makes this social robot with the most friendly appearance. This social robot can distinguish faces from objects, and it's embedded with sensors that enables autonomous motions such as dancing, walking, and gesture [13]. This social robot design includes pelvis kinematics that requires one motor, which gives him enough mobility to move naturally, spread his legs, stand on one leg, bend forward, and move most likely anywhere.

  Its most important work is its capability to work with children, specifically autistic kids, to run exercise sessions and care [14].

- **Abilities:**
  - NAO (Fig.4.4) speech software allows him to allocate sounds and execute text to speech tasks in more than 20 languages.
  - This social robot has shape and objects detecting algorithms, which assist him in avoiding falling, and if he does fall, it can get back up quickly.
  - It can react and respond as needed. Can hear, see, and identify both who you are and your expressions/emotions.
  - Its many light-emitting diodes (LEDs) allows him stunning, accurate visual effects and pattern.
  - It can convey accurate gesture motion, such as sign language.

Creator: Aldebaran Robotics and SoftBank Robotics
Country: Japan JP

Year: 2009

Type: Humanoids, UI and Research, Consumer,

Target Market: Entertainment, Both homes and businesses

Software Language: Python, C++, Java and, JavaScript

Figure 4.5: Romeo Robot

- **Description:** Compared to Nao (Fig.4.4) and Pepper (Fig.4.2), the Romeo (Fig.4.5) robot is taller at 1.4cm and weighs about 36 Kg. It was primarily designed for assistance purposes to improve and facilitate people with daily activities such as taking notes of medication, doctor appointments, finding lost objects, and assembling a shopping list.

- **Abilities:** Romeo can walk up and down the stairs, grasp objects, recognize faces, support people that have difficulty walking, or fetch objects in another location by himself. It is programmed to achieve ”harder” tasks such as opening doors in the long term, and it includes vision traits able to detect if an individual has a fall and help them get up or contact the emergency services if needed [15].

  This social robot displays advanced human interaction skills being able to have dialogues, give suggestions, play games, building a real connection with people, especially elders.

Creator: Blue Frog Robotics
Country: France

Year: 2015

Type: UI and Consumer

Target Market: Family, Education and Entertainment

Software Language: TensorFlow & Android Studio

Figure 4.6: Buddy Robot

- **Description:** About 60 cm tall and weighing 8 Kg, Buddy (Fig.4.6) is an endearing, friendly, emotional robot that wins the heart of the whole family, especially children.

  Based on its advanced human social capabilities, this social robot can express and adapt to a range of emotions. Thus, it often serves a variety of applications such as homes, administration, hospitals, office, schools, entertainment, and retail, where it can welcome, educates, and inform guests, patients, or visitors in a polite and amicable manner. Furthermore, among all the social robots mentioned in this survey, Buddy (Fig.4.6) is probably one of the best choices for families with kids [16].

- **Abilities:** Buddy (Fig.4.6) can sense his surroundings, avoid static and moving objects, understands and expresses human-like emotions. This robot can move around and make decisions, which enables him to be autonomous. Among its primary abilities of Buddy are skills such as home security, educating children, executing daily activities of its owner, entertaining people, supporting those with disabilities, or simply being a robot "friend."

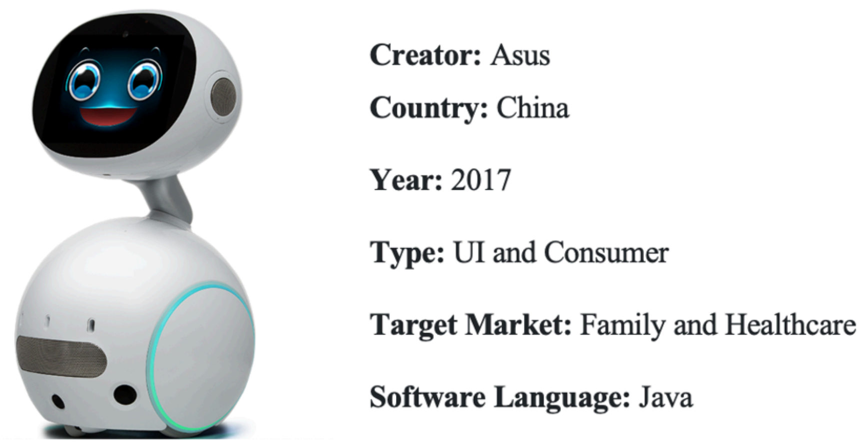


Figure 4.7: Zenbo Robot

- **Description:** Like other social robots, especially Buddy (Fig.4.6), Zenbo (Fig.4.7) is a pleasant home robot that offers families assistance, entertainment, and companionship.

  It helps families safeguard their homes, health, and well-being with real-time video monitoring systems. In case of emergencies, it can notify authorities, and if needed, authorities can remotely take control of this robot to inspect its user's home [17].

- **Abilities:**

  – To move independently around its user home.

– Releases stress by engaging users in various online tasks such as games, singing, dancing, or telling stories.

– Notify users of appointments or medication with audible daily reminders.

– Its embedded camera can recognize faces, take photos, arrange video calls, etc.

– Uses different facial expressions to convey emotions to exchange interactions with people.

Creator: Honda

Country: Japan

Year: 2000

Type: Humanoids , UI and Research

Target Market: Entertainment

Software Language: unknown

Figure 4.8: Asimo Robot

- **Description:** Asimo (Fig.4.8) is a UI, a humanoid social robot designed by Honda engineers to be a helper to people. Since it was introduced in 2000, this social robot has been continuously developed, and nowadays, it is one of the most impressive and advanced social robots on the market.

- **Abilities:** Asimo (Fig.4.8) has the ability to recognize the faces of individuals mapped to its database using its embedded camera eyes, it can sense its surroundings and register objects in static mode. Its senses capabilities enable the awareness of when to avoid obstacles as it moves through its environment, responding to voice commands.

  The combination of physical skills and adaption to its surrounding environment allows Asimo (Fig.4.8) to efficiently execute human-like tasks [18] such as holding or opening bottles and grasping various objects from trays to shopping carts. Beyond that, this social robot's motions are significantly advanced that it can run forwards and backward, hop on one or both legs, turn smoothly to reach for and grasp objects and even use the stairs while maintaining a stable posture.

4.1 Social Robots Use Case SDKs

Software Development Kit (SDK) refers to tools used to develop software applications or hardware devices, usually provided by a manufacturer. Following up on the survey on social robot use cases, below are some of the SDK used to create these social robots:

4.2 Jibo SDK

Jibo (Fig.4.1) SDK, built on Electron (a framework for building desktop applications), consists of speech solutions, animation features, behavior models, and a simulator. Its SDK allows developers to access and integrate functionalities such as: accessing the audio (speech recognition system), managing and manipulating behaviors, and accessing the face structure of this social robot using JavaScript APIs methods.

Behavior Trees (BT) represents the prominent software architecture on which the SDK was based since it holds great tools to generate autonomous behaviors. The Behavior Trees model allowed Jibo (Fig.4.1) developers to implement the functionality of switching tasks quickly and easily [19].

- SDK was built on Atom, an Integrated Developer Environment (IDE) based on electron and produced by GitHub.
- Jibo SDK tools were developed based on a Behavior Trees (BT) structure.
- Its SDK includes visual processing models, audio, speech technology, and cognitive algorithm for interaction and movement.
- Programming Languages: C, Java, JavaScript, Objective-C
- Related APIs: Jibo Indirect API v3.0.7

4.3 Buddy SDK

Buddy (Fig.4.6) was created on top of popular software tools: Unity3D, OpenCV, and Arduino. Currently, it uses an open-source platform running in the Android and Windows environments. Its SDK [20] includes:

- Visual processing tool on top of Unity 3D software.
- Library of animations, speech, and sounds.

- Text-to-speech solutions.
- Virtual simulator in which developers can create animations, behaviors, and timelines for testing without a robot.
- Programming languages supported includes Python, C++, C, C, Java, and JavaScript.

4.4 Zenbo SDK

Zenbo (Fig.4.7) is built on Android, provided by Asus. Its SDK tools include Java files that can be imported and developed using the Android Studio IDE. Using Zenbo (Fig.4.7) SDKs tools [21] includes accessibility to this robot's motion and vision capabilities, as well:

- It includes the complete set of APIs for development.
- NLP: Natural Language Processing.
- DS: Dialogue System.
- CSR: Continuous Speech Recognition.
- **Zenbo SDK architecture includes the following subclass libraries:**
  - Robot dialog, emotional and expression display.
  - Vision models for face detection, recognition, and tracing.
  - Motor skills include body and head movements and remote control.
  - Utility skills, meaning the ability to follow users and execute arm and hand gestures.
  - Simultaneous localization and mapping (Slam) is used for robot navigation purposes.

It is imperative to be aware that most social robot SDK tools are customized for a specific robot software or hardware device, such as those mentioned above (Buddy SDK, Zenbo SK, Jibo SDK). Which makes these social robots SDK somewhat limited to support specific application tasks.

Therefore, with this chapter, we can conclude that, if or whatever SDK tool is chosen for Cruz web application development, it is necessary to apply technologies that are versatile and flexible to provide a wide range of options, such as:

1. **Programming language options**

   Having access to various or commonly used programming languages, for instance: JavaScript, Python, C++, C, and Java.

2. **SDK Flexibility options**

   An adaptable SDK that allows Cruz application performance testing both during and after development.

3. **Integrate different software**

   The chosen SDK technology should easily be transferable to another machine if necessary.

4. **Application Programming Interface (API)**

   Often used in Robotics, the SDK should contain a powerful API in order to interact with a specific hardware or software component, such as:

   - **Sphero API** - interface associated with Sphero toy robots.
   - **Arduino API** - interface associated with Arduino microprocessors commonly used to program the connection of applications or external systems to exchange data.
   - **GoPro API** - generally used to integrate GoPro camera functionalities, providing wired and wireless connection to HTTP and Bluetooth, which can include: tracking camera status, capture media, webcam loading, configuration, etc.
   - **Dronecode API** - can be used to integrate drone functionalities with cargo, surveying applications, and aerial photography.

Chapter 5

Social Robot Essential Features

This section demonstrates an overview of the essential features that all social robots must have to achieve human acceptance. To succeed in the Cruz development process, it is a must to be highly aware of the various aspects that include the building blocks of a social robot. From mechanical engineering, software engineering, and programming languages knowledge to problem-solving skills, all are required in some way or another.

Robotics and AI development require immense research. As such, taking advantage of the knowledge required from the survey on the social robot use cases, it is possible to identify some of the standard and critical characteristics that all social robots should have:

1. **Intelligence**

   Although the intelligence between humans and social robots is still highly complex to replicate, a social robot still needs to be intelligent enough to execute its programmed tasks or receive instructions. When talking about intelligence, two main subclasses of intelligence are taken into account: social intelligence and cognitive intelligence.

   - **Social intelligence:** such skills enable social robots to behave and express themselves accordingly to its user. Social intelligence is directly correlated with the capability to get along with others, understand one's emotions and be effective. Such intelligence requires observation, communication, and sensitivity to interact successfully with humans in any social setting.
   - **Cognitive intelligence:** is associated with the robot's ability to be independent. Autonomous robots can receive and execute given instructions, and through their sensor, they can fetch information autonomously to execute a physical task without human intervention. Such skills allow social robots to improve daily human life by exchanging interactions in a non-intrusive way.

It is important to note that the implementation of some form of intelligence is by far the most crucial feature. Otherwise, social robots will not be accepted. Hence, the acceptance of social robots within society is only possible if they display enough social and cognitive intelligence for: interaction, communication, or collaboration [22].

Thus, it is decided beforehand that, **for Cruz Robot development, intelligence capabilities will be implemented in the form of communication and visual processing**, which will be explained later.

2. **Motor skills (Environment Intelligence)**

   The robot needs to move around according to its social settings and task. From rolling on wheels or walking, being able to move is a must sometimes. Such ability is directly correlated to the robot's task coordination and allocation. Therefore, studying, analyzing, and predicting possible human behaviors is necessary in these cases.

   Embedded sensors should perceive pedestrians movements so that social robots can react accordingly. A popular application could be using sensors embedded in social settings to be used in real-time to support the robot's allocation and coordination. In addition to sensors, a turning algorithm is often implemented to help the robot allocation and stop them from being trapped in corners [23].

   Handling static or dynamic obstacles, avoiding collisions, and moving smoothly is all included in the robot movement skill development, especially in social robots operating in public areas such as Pepper (Fig.4.2), NAO (Fig.4.4), and Asimo (Fig.4.8), as they are in constant contact with humans. Path analysis and planning its definitely an excellent way for social robots to access pre-defined destinations.

   Last but not least, external factors such as safety must also be taken into account to avoid harming pedestrians or damaging their surroundings.

3. **Sensing**

   As mentioned above, sensors are vital devices for robotic motions (movements). For social robots to sense their surrounding environment, a high-level programming sensor is required. A few types of popular and often programmed embedded sensors [24] are light sensors, hearing sensors, touch or pressure sensors, chemical sensors, and taste sensors. All these can give any robot a better sense level and awareness of its environment, defined by a set of connected horizontal and vertical lines.

4. **Energy**

   Whether the Robot is powered by solar, battery, or electrically, the way it is charged impacts its functionality and applications. A robot needs electrical power to its embedded systems to work and perform its required task.

5.1 Gesture Synthesis Process

As we know now, the main goal for developers when it comes to designing and developing social robots is to create robots that can socially engage in a way that is appealing and familiar to humans. To achieve this interaction, it is necessary to endow social robots with communicative attributes that include **speech solutions** and **gesture synthesis**. Gesture synthesis enables these social robots to have an expanded degree of flexibility, especially concerning the shape properties of the gesture (each gesture has a form and a meaning) [25].

Although much of the Robotics field research has been intent on gesture synthesis and recognition, only a few approaches within the social robot community incorporate both speech and gesture synthesis. The primary purpose of gesture synthesis is to replicate a particular human gesture (goodbye, hello, hands fist close or open, etc.) and convey that same gesture to a social robot.

The area of gesture synthesis is extensive and deserves a project itself. Therefore, it is beyond this project's scope to explore this topic in detail, as such a solution will not be included in Cruz robot web application development. Nevertheless, it is worthy of mentioning since it is one of the leading solutions that make social robots intractable and successful. Below are two of the most popular ways where gesture synthesis techniques can be implemented:

- **Hand Gesture Recognition**

  Gesture recognition consists of any repetition of static or dynamic movements. In the absence of keyboards and mouse devices, **hand gesture recognition** can increase human-robot interactions. Hand gestures are often implemented to control applications, where systems can be controlled remotely by hand gestures with telerobotic communication (the control robots from a distance, with a wireless network). For instance, some social robots have a 3D Kinematic model implemented to execute the hand gesture actions, precisely **glove technique** where bend sensors can be used to retrieve data which is then combined and processed [26]. The glove technique can accurately compute hand gesture movement or pose by extracting the numerical data of an arm and/or hand mapping and joint angles. Additionally, light and camera angles are added to reproduce these movements accurately.

Hand Gesture recognition based on Artificial Neural Networks (ANN) solutions enables the possibility to generate movement tracking by replicating human movements successfully. A great example of this implementation is the social robot NAO (Fig.4.4) where its gesture recognition algorithm can generate up to 19 gestures [27] (p.3) such as: *hello, me, yes, no, happy, big, kissing, where, hungry, angry, little, giving, coming, waiting, tall, listening, and pointing*. This particular form of gesture recognition in the NAO (Fig.4.4) social robot has shown to be very helpful in improving the lives of children with Autism Spectrum Disorder (ASD), where reciprocal training mirror games are used between NAO (Fig.4.4) and the kids.

- **Speech Gesture Recognition**

  Although this is a new and complex approach, co-speech gesture recognition is slowly gaining momentum within the social robot community. Machine Learning (ML) solutions can be used to develop hand and arm gestures in conjunction with phonetic linguistics data [28].

  Usually, for this approach, a data set of gestures are created and then manipulated with neural network models to achieve the synchronization between these gestures and speech, where each of the gestures is associated with a word and has a meaning [29]. Speech gesture recognition combines a sequence of gestures to equal the speech input and match the robot's actions accurately and simultaneously, making it an arduous task for developers to achieve since it includes: speech recognition, gesture recognition, and robot motions. Although current implementation methods have reached great results, challenges are still notable, specifically regarding speech as accents, phonetics, and word pitch can be challenging to differentiate.

  Each developer has his way of implementing this approach. However, to reduce synchronization issues, one frequent strategy is to independently train the speech and gesture models to merge them later [30], which is a widespread technique adopted by developers.

In summary, the **gesture synthesis process**, whether hand gesture recognition or speech gesture recognition, is the skeleton for generating action from a specific word command to then be translated into robot motions. It allows users to better understand the robot task functionalities and provides better interactions. Gesture synthesis is a powerful technique for increasing social robot reliability, as it provides guarantees to the robot.

Chapter 6

Human - Robot Social Interaction

Human-Robot interaction is the field of study that aims to understand the integration and designing of a social robot, so they can successfully exchange interaction and be accepted by its users (humans).

Pepper (Fig.4.2), NAO (Fig.4.4), and our Cruz robot (Fig.7.1) were all designed to function in social settings. As previously said, they also require cognitive and social intelligence in the form of **speech** (natural language processing), **visualization** (visual processing), or **gesture** (gesture synthesis) to convey verbal or non-verbal communication to its users in a natural (non-machine-like) and easily understood manner [31].

One critical strategy to facilitate human-robot interaction is to focus the development process by attempting to **emulate human social intelligence and communication skills**. However, completing such a task is not simple. Human social intelligence is challenging, to the point where some people avoid interacting at all. Making eye contact, understanding tone of voice, expressing thoughts, and reading facial expressions are all characteristics that can have a favorable or lousy impact on the interaction process, making it very difficult for robots to match the accuracy of such capabilities. But regardless of how difficult it is to match the human capability, the Human-Robot Interaction among social robots involves mainly three components: **motor control** (social setting analysis), **function** (to conduct its required task), and **response** (human behavior understanding) [32].

Therefore, to understand how the human-robot interaction works, we start by categorizing interaction in two ways:

1. **Remote Interaction:** when the robot and the Human do not share the same environment, spatially or temporally (ex: Mars Rovers [33], which is why this robot is not considered a social robot).

2. **Proximate Interaction:** when the robot and the Human share the same environment, been located in the same space simultaneously (ex: social robots).

Luckily for us, nowadays, **Artificial Neural Network Models** [34] have come a long way to try and mimic how people behave, emote, and communicate to produce the same results in social robots. Cruz (Fig.7.1) social robot was implemented with an effective **Neural Network Model** (which will be discussed later in this thesis) to interact with people daily using **visualization** and **speech**.

In conjunction with Artificial Neural Network Models, Human-Robot interaction tries to solve the problem of shaping robot design and development accordingly with its environment settings to interact with one or more humans. Such a problem is solved by understanding areas from cognitive sciences, psychology, linguistics, engineering mathematics, and computer science.

6.1 How Social Robots approach the target person

To approach and get acceptance from its target user, different Human-Robot Interaction techniques can be implemented, such as:

1. **Human-Robot Visual interaction**

   Where to interact with humans, social robots are designed with visual processing capabilities. Computer vision neural network models are an essential part of the visual interaction, allowing social robots to perform tasks like face detection, face recognition, human tracking in a cluttered environment, geometric shape recognition, gesture recognition, and color identification [35]. This implementation has proven successful in the building blocks process of social robot development.

2. **Human - Robot Speech Interaction**

   Communication using speech is the primary way of interaction among humans. Hence, naturally, social robots often include speech synthesis or speech recognition interfaces. A robot with the ability to initiate a conversation, dialogue, or reply to its users using speech provides better services and is more acceptable to humans [36].

3. **Human-Robot Environment interaction**

   Adapting the social robot embodiment and task to behave according to its social setting provides better results. The use of facial expressions to display ”emotions” and respect can be extremely important and coherence with social conventions. Furthermore, regarding its physical embodiment [37], it can be divided into two types:

   - **Zoomorphic social robots:** are designed with an animal ”toy” appearance, such as iCat [38] or Aibo [39].

- **Anthropomorphic robots:** this robot attempts to resemble the human shape, often built to express ”emotional” states, interaction, and desires (social robots).

Chapter 7

Cruz Social Robot

User interface design is a critical focal point for any computer system usability, as it is the source of many user complaints. This chapter introduces the Cruz (Fig.7.1) social robot and analyzes the main aspects of the Cruz user interface development process: from the web application itself to Cruz embodiment, functionalities, and intelligence (cognitive and social). Every aspect that constitutes the building blocks of social robots is now present in Cruz.

This chapter emphasizes the user importance in designing UIs by addressing their expectations and needs when developing the Cruz (Fig.7.1) **Covid-19 survey web application**. User interface development should be user-centered, which means we need to understand the user, the task that Cruz is expected to complete, and the social setting context in which Cruz will be used. These considerations were critical in the interface development procedure to achieve a successful Human-Cruz robot interaction.

Designing robots that can effectively communicate, move, and have a human-like appearance has relatively become possible over the years. These social robots designed to interact with humans, be used by humans, or work alongside humans, now include unique capabilities that enable them to recognize, identify, communicate, and even modify their mannerisms to their users. However, we are still far from attaining and matching the complexity of human-to-human interaction and communication levels. In this regard, when developing the Cruz user interface web application, critical characteristics that social robots should possess to interact with humans for extended periods were considered. By identifying these characteristics, we can then provide suggestions and directions for future developers working in the Cruz web application. Furthermore, trial testing was conducted to measure the impact and effectiveness of the Cruz web application.

Usually, a discrepancy between the user's expectations and the reality of social robots might negatively impact their use and acceptance. Therefore, it is imperative to understand and consider the components that may increase or decrease the acceptance and approval of the Cruz robot. To measure how the Cruz robot will be approved by its users, three main factors were taken into account when designing the interface, such as:

1. The Cruz robot appearance
2. The Cruz robot intelligence
3. The Cruz robot functionalities

Understanding the above factors allowed the development of the Cruz interface to achieve better results, which hopefully will be widely approved and accepted by its future users.

7.1 Cruz Robot Appearance

In contrast to the industrial sector, where robots are designed to perform specific tasks, social robots are designed for everyday life activities to perform a specific task for or instead of humans, such as the **Cruz web application built to perform face detection tasks and conduct a Covid-19 survey using speech**.

Since social robots are created to be used in social settings, they will likely be designed to have human-like features and a pleasing enough appearance to perform their intended function and be approached by humans. This is because users are more inclined to accept Cruz and other social robots if they operate and present human-like embodiment attributes such as a face, arms, legs, and some form of a lower body.

Actually, a survey done by researcher Manja Lohse [40] concluded that people are more inclined to assume less cognitive or social intelligence competence from social robots when their appearance/embodiment is less human-like, especially when it comes to domestic social robots.

Hence, the appearance of the Cruz robot was given the same weight as its behavioral functionality, social intelligence, and application social domain, as its appearance plays an essential part in the expectations and first impressions that people naturally will have of Cruz.

Furthermore, research indicates that social robots require an additional user interface, such as a touchscreen display, to promote engagement, improve communication, and provide users with security if the speech communication system fails [41].

All of these features, as seen in the image below (Fig.7.1), were meticulously thought of and integrated into the **Cruz** (Fig.7.1) **social robot, which has a human-like appearance with features such as a face, arms, a lower body, and a touchscreen**.

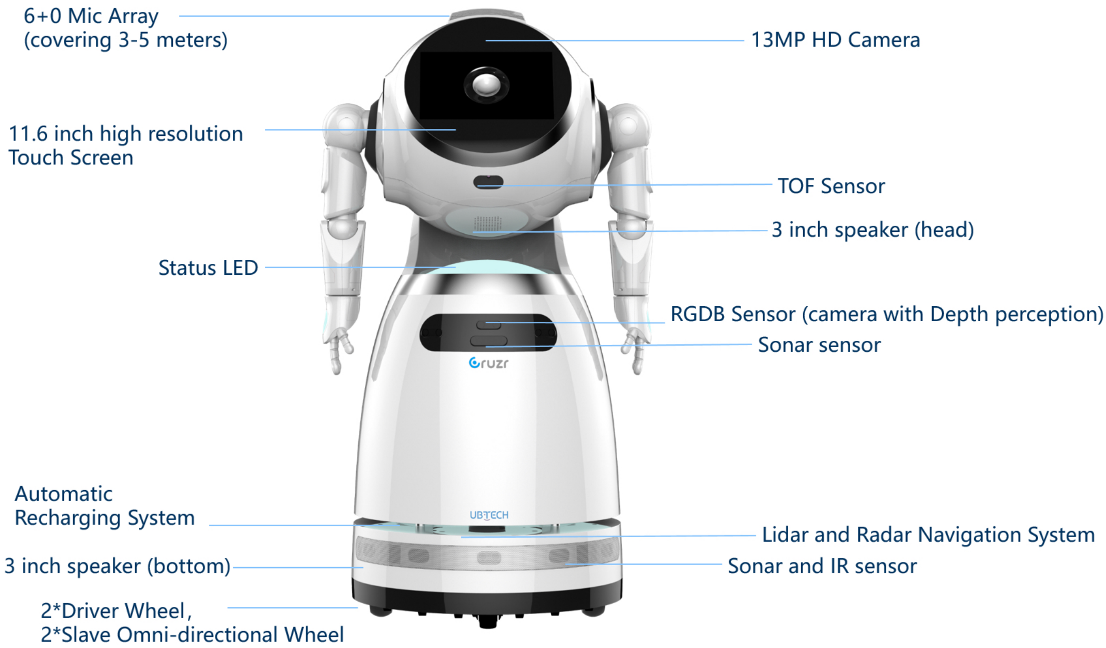


Figure 7.1: Cruz Robot Appearance [42]

7.2 Cruz Robot Intelligence

As we now know, robots must exhibit some form of human-like ”intelligence” in order to interact with and execute jobs for people. Integrating intelligence in social robots transcribes into programs designed to assure social interaction and cognitive skills [43]. These skills are essential, and they can include: emotional expression, social intelligence, conversation, and dialogue speech system, which does impact the user approval of the robot's capabilities.

The development of such capabilities for Cruz (Fig.7.1) translated into **creating algorithms behind the Covid-19 web application to provide the Cruz robot with different social and cognitive abilities such as face detection, face tracking, and speech synthesis with natural language processing**. Such intelligence attributes enable users to subconsciously be more engaged and more accepting of the Cruz (Fig.7.1) robot, thinking of it less as a machine.

In conjunction with these some of the design requirements that future Cruz developers can consider the implementation, could be:

1. Face recognition.
2. Object recognition.
3. Detect the presence of a human through their senses: vision, touch, or sound.
4. Displaying gesture skills to engage in physical acceptance, such as holding and/or touching.
5. Use physical motions.
6. Perceive and/or express emotions.

7.3 Cruz Robot Functionalities

From healthcare to home-based settings, social robots can provide physical, cognitive, and social support in education, coaching, training, entertaining, monitoring, etc.

Several technology implementations are being used at different stages to create social robots, and during the design process, specific collaborative methodologies are followed. The methodologies design is aimed to take into account the target users and the social settings. A way for these machines to contribute to the decision-making process, thus increasing their acceptance and value.

Their functionalities, tasks, and actions can be a valuable addition to any social context providing support in healthcare to children, adults, the elderly, and even dealing with people's stress management and loneliness. As we have seen during this project, there are some successful examples of social robot functionalities/tasks which can be integrated into domestic, work, and public environment [44]. Long-term studies prove that social robots can adapt to different deployed environments and different types of users according to their functionality.

The primary function of social robots is to support the specific needs of their target users, as is the case with the **Cruz social robot, whose main task is to conduct survey questions related to the SARS-Covid-19 pandemic on a displayed touchscreen** to facilitate communication with its users, minimize failures, and improve safety in the occasion that its speech synthesis system fails.

As this pandemic has affected everyone around the world, the **Cruz robot would be suitable to be deployed in any social setting**, from the healthcare sector to education, offices, lobby, etc., exchanging interactions while conducting the SARS-Covid-19 survey questions to identify possible contaminated users thus, reducing the spread to the virus.

In resume to this chapter, we can conclude that: **a)** social robots need to achieve a human-like appearance to contribute to a satisfying interaction. **b**) in conjunction with the communication process (visual, speech, or gesture) between the robot and its users, is recommended the addition of secondary methods, such as a touchscreen display to facilitate communication and add safety backup in case of Automated Speech (AS) failures, and finally **c)** social robots are overall well-accepted ”beings” due to their functionality in improving human work. Likewise, it can be employed to work alongside humans in very challenging times.

Chapter 8

Cruz Robot Web Application Overview

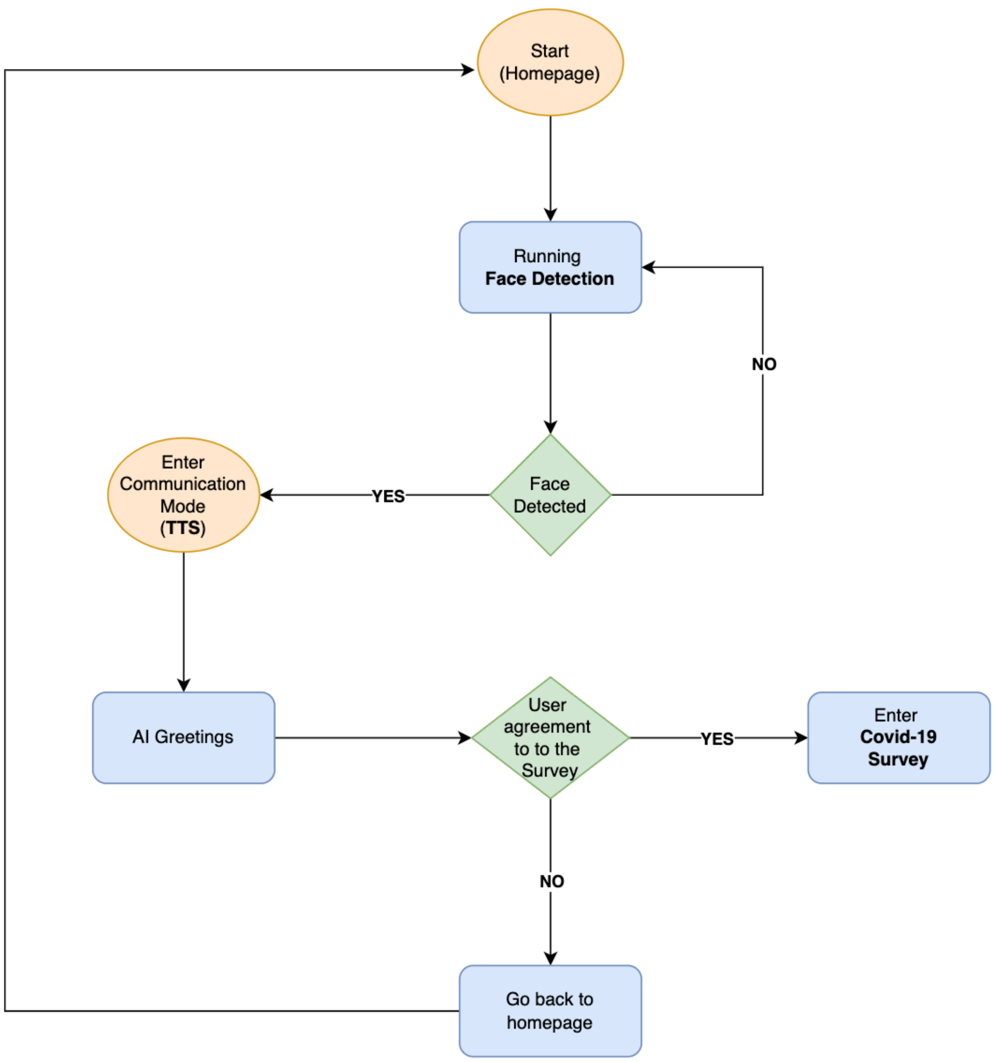


Figure 8.1: Web Application Flowchart

The Cruz robot (Fig.7.1) software was designed as a user-friendly web application called **COVID-19 SURVEY**, developed to join the fight against the virus by detecting possible contaminated users while conducting a set of Covid-19 questions to identify related symptoms of SARS-Covid-19. The AI COVID-19 SURVEY application (Fig.8.1) is built on top of deep learning solutions and web technologies, then deployed to Cruz automation. The Cruz AI web application flowchart, is structured to perform two main functions: **Face Detection** and **Text-to-Speech (TTS)**, as seen in the above image (Fig.8.1).

Like any Web application [45] is a computer program, developed and stored on a remote server (localhost), it can store and/or retrieve data by using server-side languages (in PHP), while the client-side (in HTML5 and JavaScript) displays the essential features for the user interface.

The decision to develop the Cruz user interface into a software web application is because the Cruz robot automation is prepared to support web application deployment, as well as due to the numerous advantages of this development approach. It is versatile, allowing users to perform a set of different functions. It allows the implementation of autonomous functionalities with the addition of artificial intelligence (AI) technologies. Its updates are applied centrally and automatically [46].

8.1 Web Application Key Traits

Some of the key general traits that impacted the decision to develop Cruz AI into a web application software were:

- **Progressiveness:** Web apps are browser and location agnostic, which means they may operate properly and be easily accessible regardless of the user's location or web browser. As a result, whether a person uses Chrome, Firefox, Opera, or another browser, whether they are in Hungary or Brazil, it is accessible. However, it is worth mentioning that when incorporated into a social robot such as Cruz, the user's physical presence is required to trade interactions.
- **Application interface:** the user experience is improved by the web application's simple system style of navigation, workflow, and interaction.
- **Responsiveness:** the media query settings in a web application allow it to accommodate any platform, including desktop, **tablet/touchscreen** (such as the Cruz robot), and mobile.
- **Connectivity independent:** web applications can work remotely, offline, and on a network with low quality.

- **Safe:** since they can be rendered via HTTPS (Hypertext Transfer Protocol Secure), it ensures the security to protect Cruz users' personal health information and reduces the chances of being tampered.
- **Updated:** the service worker update process ensures that web apps are always up to date.

8.2 Artificial Intelligence (AI)

Before we dive into the implementation process of the Cruz web application (Fig.8.1) (Covid-19 Survey), we need to be aware of the essential technologies that made this possible.

Artificial Intelligence (AI) [47] has grown substantially in the twenty-first century, and it is now part of an increasing number of institutions and businesses. Artificial Intelligence has made a significant contribution to the advancement of human society, innovating, upgrading, and assisting with everyday human jobs.

The **term "artificial intelligence" is usually applied to machines, devices, or software applications developed to mimic human cognitive skills**, such as learning, decision-making, and problem-solving. Furthermore, contrary to standards machines, the AI branch aims to construct a society where people and machines can coexist together and interact harmoniously [48].

Virtual assistant systems use artificial intelligence skills to speak and interact with their users. Siri, Alexa, Google Assistant, Lucinda, Cortana, and others, [49] are popular examples of artificial intelligence programs and devices. The application programs behind these virtual assistant devices were created to produce a human-like cognitive clone. As it frequently communicates with consumers via automated speech for customer assistance and information requests, significantly contributing to education, IoT (Internet Of Things), and healthcare systems.

Nowadays, artificial intelligence technologies are divided into several sub-categories (Fig.8.2), including **machine learning**, **deep learning**, **natural language processing**, **game programs**, **computer vision**, and, of course, **robotics**, as in the case of our Cruz, whose artificial intelligence technologies were implemented in the form of a software web application (AI web app) [50]. The sub-classes of AI are extensive, deep, and complex research work that mentioning each one would be extremely long and tedious to break down in this chapter. Therefore our focus is directed only to the Cruz robot web application artificial intelligence features (namely: face detection and speech synthesis) and similar technologies for better understanding and contextualization.

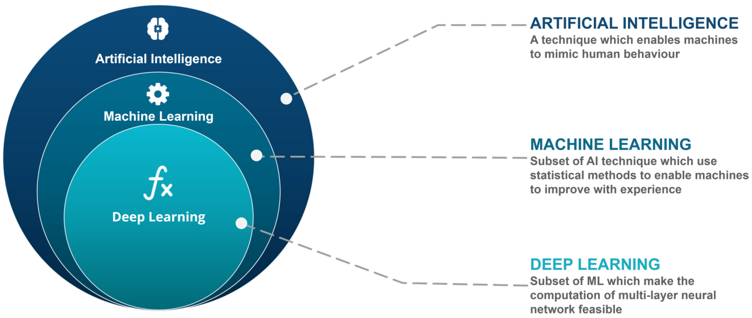


Figure 8.2: Artificial Intelligence field and its Sub-classes [51]

We'll focus in two major, sub-classes: **Computer Vision** and the **Natural Language Understanding (NLU)**, as this two areas of field made it possible the development of Cruz web application.

8.3 Computer Vision

Computer vision is a branch of AI (artificial intelligence), which provides systems and computers with the capabilities to extract data information from videos, digital images, and any other visual data input [52]. If the artificial intelligence field enables systems and computers to "think", its computer vision branch enables computers and devices to process visual data through tracking, seeing, and classifying images.

As we (humans) have the inherent capacity to recognize objects, detect abnormalities in real-time, estimate their distance, whether moving or static and distinguish objects from people in videos or digital images, **computer vision trains machines to perform these identical functions as accurately as possible**. By creating algorithms for image processing, Computer vision enables these tasks to be executed in much less time using cameras and data instead of retinas (tissue layer inside the eye which capture and transmits images to the brain). Because a computer is trained to asset, inspect, and analyze thousands of input data a minute while noticing unapparent flaws or issues, it can quickly surpass human capabilities, which are limited through time [53].

One of the many computer vision applications is the **image classification** in which, when given an input image, its trained models can accurately predict, categorize and identify that image as an object, an animal, or a person's face. **Face detection, the first main feature of the Cruz web application, is one of the many computer vision models built to find faces within a picture or video**.

Computer vision allowed the accessibility of already **machine learning models algorithms constructed to detect if the shape in a video or image matches the human face**

(face detection). Hence, accessing these algorithms makes it possible to build an autonomous application for Cruz to detect faces in real-time automatically. These machine learning models used for image detection, classification, and recognition are called **Convolutional Neural Networks (CNN)** (Fig.8.3) [54], created primarily for processing pixel data. Later on, we will elaborate more on which CNN models were used for Cruz robot face detection.

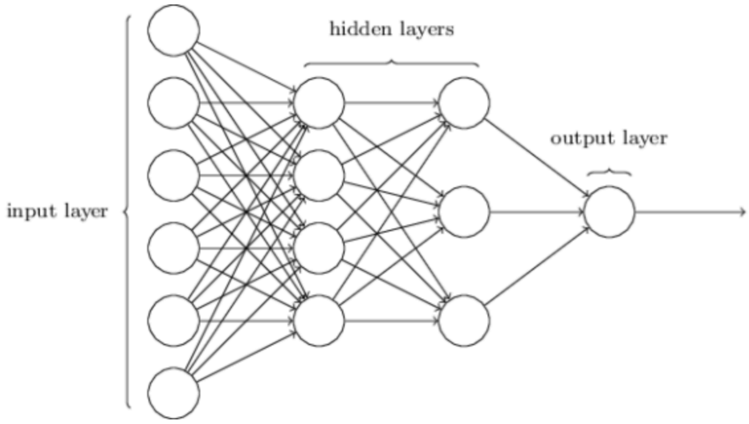


Figure 8.3: Convolutional Neural Network [55]

8.4 Natural Language Processing

Because of the rapid evolution of Natural Language Processing (NLP), some applications now choose speech as the primary form of communication and interaction with its users. The **second feature of Cruz, Text To Speech (TTS), aims to convert natural language into speech**. Such addition was possible due to Natural Language Process understanding (NLP).

Natural language processing (NLP) is a branch of artificial intelligence (AI) that uses linguistic (language) models to facilitate interactions between machines and humans. Models are built and trained to analyze, manipulate, and extract data from natural language speech or text. **NLP is a vital subset of artificial intelligence that studies how machines can analyze and comprehend linguistic, written and speech to associate the correct meaningful definition and perform the desired task**. Typical use cases of natural language processing include speech recognition, speech synthesis, dialogue systems, speech dictation, machine translation, lexical analysis, etc.

However, it is worth mentioning that communication via a Natural Language can require quite complex fundamental knowledge: **producing text** and **understanding text**. To enable the implementation of this branch, neural network algorithms [56] are built to perform this process by **separating human language into fragments so that a text or speech can be grammatically structured as a sentence with meaning in which each word is analyzed**

and then understood within its context. This algorithm allows computers to understand or read spoken or written text similar to humans. Since these NLP neural network methods are so complex, they combine linguistics, statistics, machine learning, and deep learning [57].

To implement the Natural language Processing pipeline tasks, data scientists need at first to define the logic behind a specific human language in a way that the machine can understand. To achieve this, Natural Language Processing includes tasks such as:

1. **Tokenization:** the words that form a sentence should be broken down into smaller semantic units, where each unit is tracked according to its position in a sentence. In this initial stage, words are not categorized grammatically but only separated into a stream of characters such: as letters, words, numbers, hyphens, slashes, and punctuation marks.

2. **Part-of-Speech Tagging:** this is the level where words are grammatically categorized according to the human language rules, where each word is assigned a probability within a context as nouns, verbs, prepositions, adjectives, conjunction, determiner, adverbs, pronouns, etc.

3. **Stemming and lemmatization:** reducing the words into their root form to convert and contextualize the word to its meaningful base form. For instance, plural words are often reduced to their root singular form. (Note that some data scientists may skip this task, as some do not consider as necessary).

4. **Stop word removal:** the gathering of repeated words within a sentence, of most prepositions and articles that usually add no or little information.

Only once these pipelines [58] have been completed can NLP algorithm models be built, in which a structured rule **X → A, B, C...** can generate grammatically correct sentences as a result of many word combinations. Where **X** stands for the sentence and **A, B, C...** stands for the other possible word combinations. The addition of this function translates into speech recognition, speech synthesis, and speech dictation applications.

Most of the time, applications take advantage of automated speech by including both speech synthesis and speech recognition, but the **Cruz web application now features only the speech synthesis artificial production to interact with the users**.

Chapter 9

Cruz Robot Web Application Technologies

9.1 JavaScript

The web browser language, JavaScript, often abbreviated as JS [59], is a programming language that allows developers to create dynamic web applications. A common use of JavaScript is changing the HTML and CSS dynamically to manipulate an application interface through Document Object Model (DOM) API.

JavaScript can pretty much execute everything web application related: manipulation, interaction with the user, and the web server, such as:

- Change or update the existing webs page contents.
- Change or update the existing web page contents.
- Send queries to remote servers across the internet (AJAX and COMET technologies).
- Set and get cookies.
- Local storage (storing data on the client-side).

9.2 Face-api.js

Face-api.js is a JavaScript API [60] used for **face detection or face recognition** in the browser created on top of the tensorflow.js core to run convolutional neural networks (CNNs) Models.

9.3 Tensorflow.js

Tensorflow.js [61] is a library used to execute Machine Learning models in JavaScript to be applied directly in the browser. It allows developers to implement neural network functionalities in web applications. Tensorflow.js has been mainly used to access models for speech recognition, face detection, face recognition, voice commands, object recognition, etc.

9.4 API

Application Programming Interfaces (APIs) are ready code plug-in intermediaries used to connect applications and access data to interact with external software. They fall into two categories:

- **Browser/Web APIs:** Browser/Web APIs: integrated into a web browser, these APIs allow developers to perform complex operations for DOM manipulation, network requests, client-side storage, retrieving device media streams, etc. Some often-used Web APIs are, Fetch API, Canvas API, Fullscreen API, Clipboard API, etc.

- **Third party APIs:** a third-party API is generally stand-alone code provided by companies such as Facebook, Twitter, or Google, allowing you to access its functionality with JavaScript and apply it to your applications.

9.5 Google Speech Synthesis API

This API converts text into speech using a third-party API, Google's AI technologies. Google Speech Synthesis allows developers to synthesize text into speech [62].

9.6 PHP

Hypertext Preprocessor (PHP) [63] is a front-end and back-end language used to develop interactive applications. It is among the first programming languages that can be used in association with HTML, making it simpler to add dynamic functionality to applications.

9.7 phpMyAdmin MySql database

phpMyAdmin is a free software tool built on top of PHP programming language to handle database tasks through MySQL. With phpMyAdmin is possible to execute administration tasks such as creating a database, running queries, and adding user accounts [64]. The Cruz web application takes advantage of the **Structured Query Language** (MySql) to save all the users data. MySQL is an RDBM system, meaning its Relational Database Management System performs fast and easy access to storage and retrieval of significant volumes of data.

Chapter 10

Cruz Robot Face Detection

The **homepage of the Cruz robot web application** *`index.php`* is the entry point to exchange interactions with Cruz. **Cruz web application face detection is based on visual processing technology and can run human face tracking and detection in real-time through Cruz's embedded camera**. As mentioned before, the breakthrough of Convolutional Neural Networks (CNNs) allowed the possibility to accomplish such a task.

Cruz face detection technology [65] uses machine learning algorithms to extract human digital data in videos or images. These digital data images could include structures with and without faces, such as buildings, objects, or human body parts. Its face detection algorithms typically start by looking for human eyes since they are the easiest to detect, followed by the mouth, nose, eyebrows, and iris. When the system recognizes these features, it determines that face landmarks have been discovered.

Cruz web application is developed with **JavaScript Face-api.js** to accomplish the face detection in the browser, as this API is built on top of the **Tensorflow.js core** (machine learning library in JavaScript), which uses **convolutional neural networks (CNNs)** to perform this specific computer vision task. This JavaScript API for face detection, face-api.js, includes the following convolutional neural networks models:

1. **SSD Mobilenet V1 Model**

   The Single Shot Multibox Detector (SSD) is a Convolution Neural Network created on top of an alternative version of MobileNet, MobileNet V1, which includes added traits such as b-box layers to run the face location predictions probability within an image or video.

   This neural network identifies the exact locations of each face in a video streaming or images and returns the bounding boxes (b-box) with the parameters location of each face [66]. Although this face detector model holds the most accuracy in detecting the probability location for facing tracking, the output is not as fast when it comes to its time resolution.

2. **Tiny Face Detector Model**

   An alternative version of Tiny Yolo, Tiny detector Convolution Neural Network employs depthwise separable convolutions to apply a filter to each data input (image, video), which makes this neural network model faster and more flexible to multiple data sizes, but less accurate compared to Single Shot Multibox Detector [67].

   This model can predict an entire cover facial point with its bounding boxes, which returns accurate results in conjunction with **face landmarks detection**.

3. **Multi-task Cascaded Convolutional Neural Networks (MTCNN) Model**

   Lastly, face-api.js includes MTCNN models, a much more lightweight Convolutional Neural Network, and the most popular face detector in the browsers. The MTCNN is popular because it attains the first stage results on a range of benchmark datasets, namely Wider face that includes up to 393 to 703 datasets with a variety of poses, scale and diversity [68](p.5). It can recognize specific facial features called landmarks that correspond to a person's eyes, nose, and mouth. It includes:

   - Faster face detection speed.
   - Allows face alignments, concurrent and precise 5 landmark points.
   - Compared with SSD has a small model size.
   - Allow parameters configuration to increase performance.

   Each stage of MTCNN networks is multi-tasks [69] layer-based (P-Net, R-Net and O-Net), and three crucial processes happen for each layer: face alignment, bounding box regression, and feature tracking.

   - **Face alignment:** extracting the facial landmarks feature points to decide the bounding box. Usually, around 68 landmark points can be computed on a face to gain critical data on how the faces are aligned in a video or image.
   - **Bounding box regression:** the process of surrounding the extracted facial features with a b-box to identify whether it is a human face or not.
   - **Feature tracking:** applying locator face to obtain all the possible face locations within an image or video.

10.1 Face Detection Folder Structure

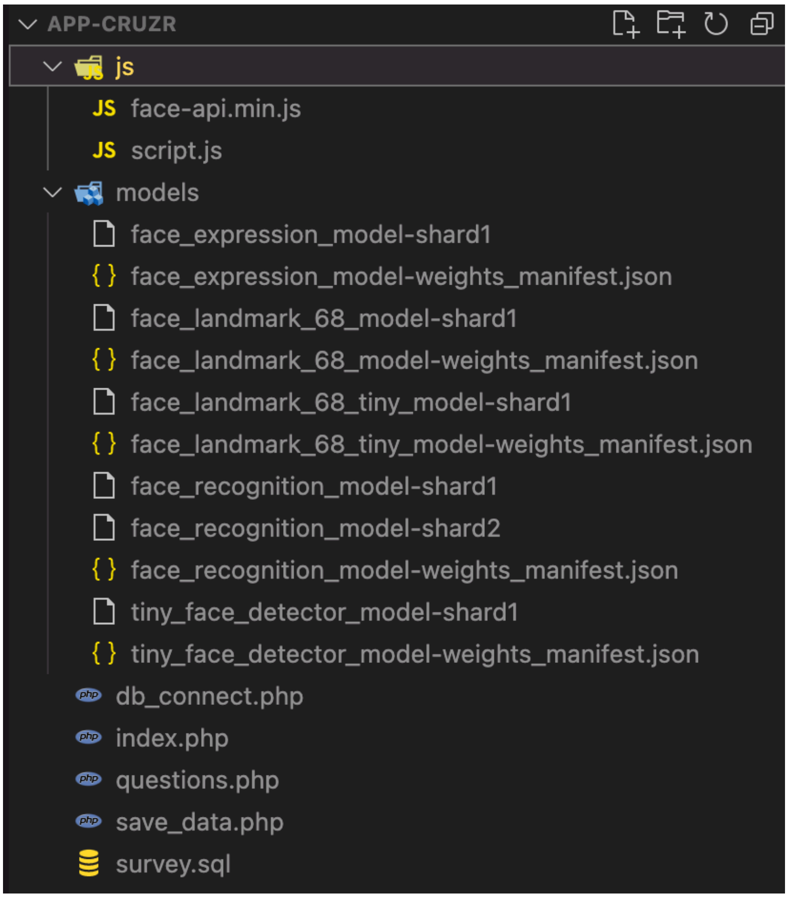


Figure 10.1: Cruz face detection CNN

js: includes all the JavaScript related file.

face-api.min.js: API built on top of JavaScript to execute face detection or recognition running SSD NN and MTCNN.

models: contains the libraries with all CNN models.

face_expression: Model that can detect the facial expression.

face_landmark_68: 68 points landmark face detection.

Tiny_face_detector: Tiny Face Detector CNN models.

10.2 Homepage: index.php

To create the Homepage (Fig.10.4) for the face detection, three main elements were included within the code:

- **Canvas**

  It is used for drawing purposes in the browser using HTML and JavaScript, which enable the drawing the required for the facial landmarks detection and the bounding box for location detector.

- **Loading the models**

  Using the face-api.js library allows us to take advantage of all the pre-trained neural network models (Fig.10.2) mentioned before.

```
const video = document.getElementById('video')

Promise.all([
  faceapi.nets.tinyFaceDetector.loadFromUri('/models'),
  faceapi.nets.faceLandmark68Net.loadFromUri('/models'),
  faceapi.nets.faceRecognitionNet.loadFromUri('/models'),
  faceapi.nets.faceExpressionNet.loadFromUri('/models')
]).then(startVideo)
```

Figure 10.2: Loading the CNN Models

- **Adding Bounding Box**

  After the models are loaded, we then access the video and canvas to run the face locator through to the video stream and pass it via model.

```javascript
// setting an EventListenet when the video start streaming
video.addEventListener('play', () => {

  // to display the api face & landmark detections created from video element
  const canvas = faceapi.createCanvasFromMedia(video)

  // add canvas to the screen
  document.body.append(canvas)
  const displaySize = { width: video.width, height: video.height }

  // match canvas to the display size
  faceapi.matchDimensions(canvas, displaySize)

  // block to be executed repeatedly with a fixed time delay between each call.
  setInterval(async () => {
    const detections = await faceapi.detectAllFaces(video, new faceapi.TinyFaceDetectorOptions()).withFaceLandmarks().withFaceExpressions()

    // displaying canvas inside the video
    const resizedDetections = await faceapi.resizeResults(detections, displaySize)
    await canvas.getContext('2d').clearRect(0, 0, canvas.width, canvas.height)

    // drawing canvas
    await faceapi.draw.drawDetections(canvas, resizedDetections)
    await faceapi.draw.drawFaceLandmarks(canvas, resizedDetections)
    await faceapi.draw.drawFaceExpressions(canvas, resizedDetections)

    if(detections[0]){

      // block to be executed once after the interval of time set
      await setTimeout(
        window.location.href="questions.php",
        1000000
      )
    }

  }, 100)


})
```

Figure 10.3: Code to Run Face Detection & Adding b-box

As such, we can now have a real-time face detection running locally in the browser at `http://localhost/app-cruzr/index.php`

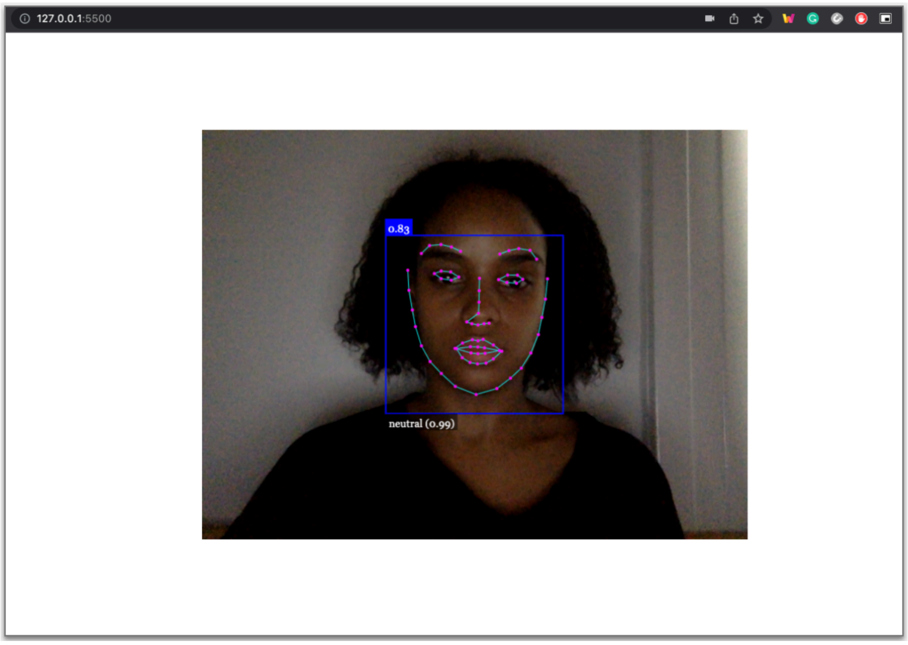


Figure 10.4: Cruz Application Running Face Detection

Chapter 11

Cruz Robot Speech synthesis

Text To Speech takes texts and words as input on a computer or other digital device and converts them into Speech. As we now know, using Natural Language Processing, all kinds of text, files, words, and sentences can be converted into human-like Speech. It is a beneficial feature for writing, editing, and especially for people who struggle with reading or focusing. It works with almost every digital device, including computers, smartphones, and tablets which is the case of Cruz. The voices generated in Text To Speech are computer-generated based on Natural Language Processing, and their reading speed can usually be programmed to the desired output [70].

11.1 Speech Synthesis Applications

Text to speech systems has various range of applications. Their original integration was in reading text to voice systems, which were developed for those with vision impairments to read text from books, files, and webpages converted into speech. Since there were hardly any options available for people with these disabilities to access visual information in the past, such a system became a breakthrough within the Artificial Intelligence community.

Of course, with the evolution of technology, much complex and fast Text To Speech technologies have been developed to ease human-computer interaction, not only designed for "normal people" but as well for people with any disability which text to speech technologies quickly helps them to navigate around any computer or device.

In recent years, numerous advances in the quality of the text to speech technologies have been implemented in the real world, with the most common application being automated call-center support, where people can call regarding their bill payments, schedule a doctor appointment, or book a restaurant or travel reservation interacting only with the automatic dialogue system, ensuring accuracy and reducing the workload of employees. In conjunction with these, TTS technologies can be used for weather reports, news stories, GPS requests, etc.

11.2 Speech Synthesis Goals

Building an AI computer system like the **Cruz application aims to achieve a closer form of interaction with this user through speaking**. Thus, the developed text-to-speech system must receive any text input and convey its message. By integrating natural language processing neural network algorithms, it's possible to use human-like voices and language datasets that can be manipulated and trained into speech (audio).

According to this system's set language of choice (English), the **Cruz text to speech can take written input and convert it into speech. The logic behind Cruz's text speech technology is to playback the originally programmed written text message through speech**. One way to achieve the audio/speech playback of text input is to build separate network models of common words and sentences, apply word normalization (NLP pipelines), combine them afterward, and run synthesis, a widely used technique in automated dialogue services.

Through the years, there have been many different approaches of NPL to generate the conversion of text to speech, being the most popular concatenation synthesis and parametric/statistical synthesis [71].

11.3 Concatenation synthesis

In concatenative synthesis, the signal is produced from real segmented speech units using human base linguistics according to the speech production of a specific language to create a new utterance, which is then manipulated to match an actual human speech. Concatenation synthesis is based on a sequence of pre-recorded speech segments. Such segment data can be phonemes, words, diphones, syllables, or entire constructed sentences stored in spectrograms or wave sounds (visual representation of sounds/signals) [72]. And, since Cruz robot web application speech synthesis models were built on this approach, the speech result output of Cruz is more human-like sounding and natural. Just like we listed the types of NN models used for face detection in the previous chapter, we will specify which type of NLP models we are referring to later on.

11.4 Statistical Parametric Synthesis

Statistical Parametric synthesis, also known as Hidden Markov model (HMM), takes advantage of pre-recorded human voices where algorithms are applied to manipulate the audio as parameters to modify the voice. The essential premise is that we can synthesize human speech by manipulating specific word units from a database of pre-recorded speech. This statistical/-

parametric synthesis approach generally includes two stages: **training** and **synthesis**.

The **training** focuses on extracting a set of standard parameters that constitute the pre-recorded speech samples, such as **frequency spectrum**, meaning the vocal trait of the recording, **fundamental frequency** (voice source/pitch), and **time-frequency** (duration) of the recorded speed [73]. Then it uses statistical models to estimate a pattern of those parameters. The **Hidden Markov Model (HMM)** is the model that has proven to execute such tasks and provide the best results.

Hidden Markov Model (HMM) statistical can estimate and generate these parameters (spectrum, frequency and time) from the desire speech sequence and uses the same parameters to synthesize the conversion to speech wave-sounds or spectrograms. Some advantages of this statistical process is: language independence, flexibility in audio characterization and database independent as well. However, in some cases, the synthesized speech is not the best quality, thus the evolution and introduction of **Deep Learning Neural Network models**.

Synthesis is the process to generates the speech given an input. The input which is usually in a form of text it's then merged and processed to construct a phrase.

11.5 Speech Synthesis Models

There are several Text-to-Speech systems models, but this section will be taken into consideration real-life systems applications:

- **Text to language Models:** Text to language Models: In this model, any written text is taken as its linguistic form message, and synthesis is produced on top of it. However, because text production is subjective and imprecise, text normalization NLP pipeline tasks (tokenization, lemmatization, segmentation, stemming) is typically included in conjunction with synthesis. The main idea for this model is that text must be fragmented before it is used as input to the synthesizer. This model follows the Concatenation synthesis approach.

- **Grapheme Models:** this model is very straightforward, the text is input as a grapheme form (the smallest unit in a writing system) and then is converted into full text for synthesis. Although this model is very simple, its approach can only be integrated into languages where the grapheme and the full texts have direct grammar correspondence. Proven to be limited to languages where this direct correspondence does not happen, such as Mandarin or English [74].

- **Common Form Model:** this approach is based on decoding the primary root identity of each word within a text, but some implementations take a step further with linguistic analysis where morphologically analysis (grammar classification) is performed, consequently NLP speech tagging task. This approach is beneficial for finding daily words and associating those words with common daily phrases.

11.6 Communication Process

The general structure of the communication process, which involves spoken and written language, is implemented as a base framework that is analyzed to build text to speech technologies. In order to understand this process better less analyze a basic communication process taking into consideration the following factors:

- **Message**

  In communication, a *message* is defined as information conveyed by words, formed through writing or speech. The transmitter of a message is called the **sender**, and the recipient is called the **receiver** in the communication process.

- **Signal**

  The message constructed to form spoken words and sentences is translates into speech signal. In the case of humans, the speech signal can be associated with the noise produced by speakers, which corresponds to the noise from the mouth or nose. It is considered a one-dimensional process of air pressure within time. In case of computers, microphones can convert the irregular air pressure into electrical signals such as currents or voltages which then forms sounds categorized as speech signals in speech processing. Furthermore, the most critical components related to speech signals are energy, short time zero crossing, and autocorrelation.

- **Meaning**

  Meaning is associated with the human capacity to understand, encode or decode messages. Understanding is the ability of accessing speech learning patterns and thoughts to formulate messages.

Understanding all the above factors combined made Natural Language Processing a reality.

11.7 Google Text-to-Speech API

With Google Text to Speech API, it is possible to synthesize natural speech or written text. The API includes over 100 voices, dialects, and accents available in different languages. Accessing Google's Speech API powerful **DeepMind neural networks** based on **WaveNet Neural Network Models** was possible to generate the text conversion to speech for Cruz Robot to communicate with its users through its load speaker while asking the survey questions. Thus, using this API neural network model, we accomplish Cruz's second form of intelligence: speech.

11.8 WaveNet Model

Based on the Concatenation NLP approach, WaveNet is a neural network model used to train large speech volumes. Was the first model able to successfully manipulate and estimate the raw wave sounds of the audio signal by predicting which speech unit appears to follow each other. The neural networks build a sophisticated speech structure during training, emphasizing tones to generate realistic human-sounding speech.

To generate such precise models, it uses a convolutional neural network associated with dilated convolutions, where each convolutional layer has a dilation factor. Actual human recorded wave sounds it's used during training, and afterward, the final wave sound is output by the neural networks in-unit samples. Thus, when text input is given, WaveNet neural network model can build from scratch its corresponding speech wave sounds on each sample at a time.

WaveNet creates speech which is more human-like, as it includes natural-sounding components such as emotion, accents, intonation, pitch, etc. Although this model is based on the concatenative synthesis approach it solves and surpass its limitations by including aspects left out of earlier of text-to-speech systems such as lip-smacking and breathing rhythms to generate more natural speech with an emphasis on phonemes and syllables [75].

11.9 WaveNet Voices

With Text to Speech raw audio data of natural human language, speech can be created that sounds like a person talking when reading over a text.

The Google Web API Text to Speech has a wide range of WaveNet voices and languages available to use, and these voices can be distinguished by gender, pitch, and accent. For Instance, speech that imitates the female voice is embedded into the API, and it is possible to convert the same speech into a male voice with the same or different accent. WaveNet neural network models are used in conjunction with the WaveNet voice, allowing these voices to be trained to speak as human-like as possible rather than machine-like, as other synthetic voices do, which is one of the many reasons why this API was chosen to construct Cruz speech synthesis feature.

The main difference between WaveNet voice and the WaveNet model is that: WaveNet voice is specifically used to generate a type of voice, as WaveNet models pre-trained these voices as raw audio speech.

11.10 API Speech Interface

The following interfaces and properties can access by this speech API to generate the conversion of Text To Speech (TTS):

- `SpeechSynthesisUtterance:` the API's main interface can be accessed directly in the browser and includes all the necessary speech services (pitch, rate, text, voice, volume, speed) to execute the speech request.
- `SpeechSynthesisUtterance.lang:` sets and get the languages for the speech API.
- `SpeechSynthesisUtterance.pitch:` sets and get the pitch at which the frequency output will be spoken at.
- `SpeechSynthesisUtterance.rate:` sets and get the speed of the speech.
- `SpeechSynthesisUtterance.text:` sets and get the text to be synthesised (converted) into speech.
- `SpeechSynthesisUtterance.voice:` sets and get the voice that will be used to speak.
- `SpeechSynthesisUtterance.volume:` sets and gets volume that will used for the speech.

11.11 Survey Page questions.php

The survey questionnaire consists of **one tab with questions, most of which are mainly related to the SARS-Covid-19 pandemic, asked by the Cruz robot using Text-to-Speech API**, with a set of standard **YES** or **NO input answers** for its users.

The first question (Fig.11.1) addresses the user's availability and agreement to complete the survey, and the remaining 14 questions are Covid-19 related to identifying possible contagious users.

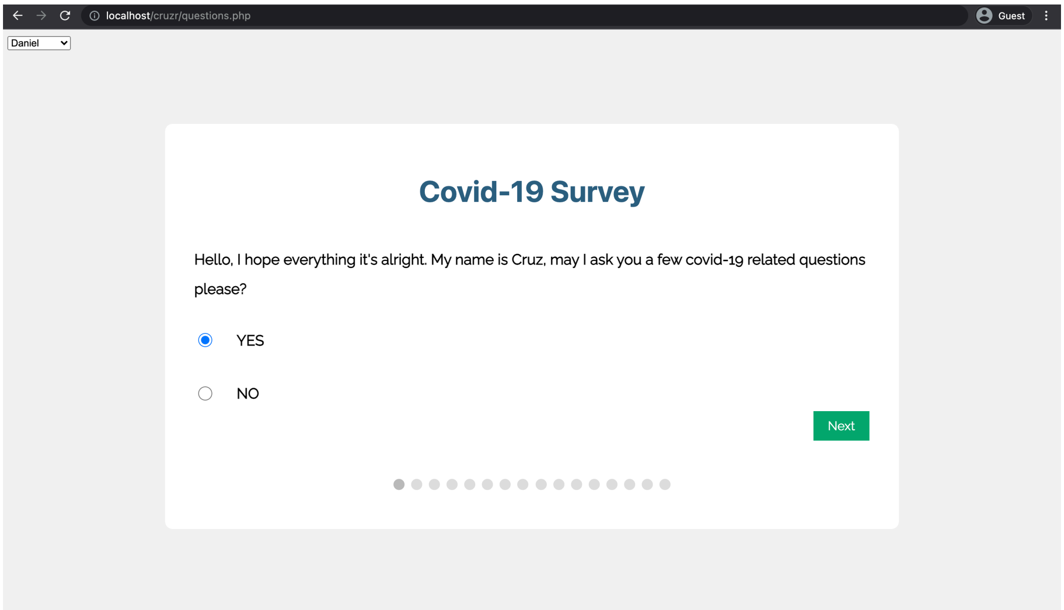


Figure 11.1: Cruz Survey Question 1 Layout

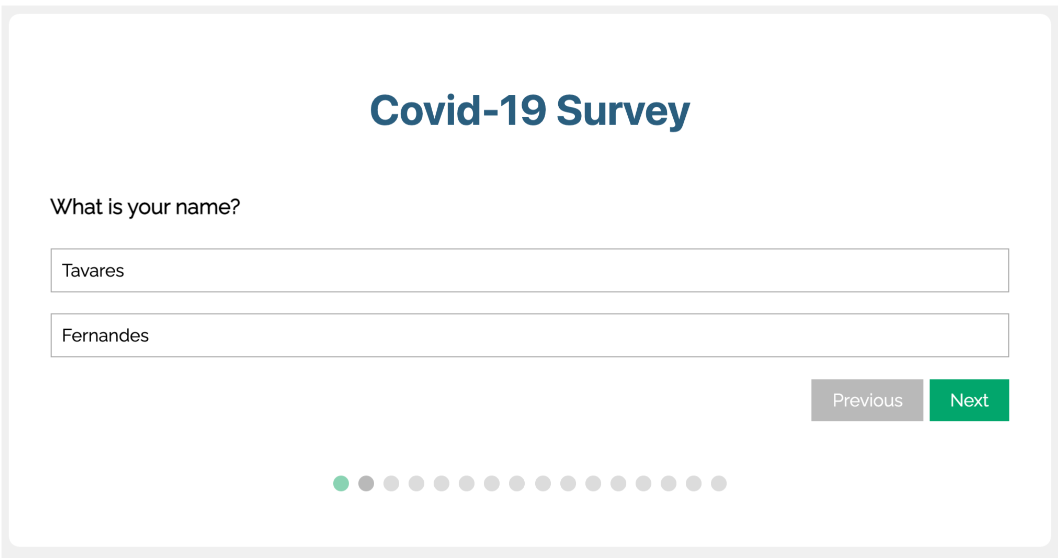


Figure 11.2: Cruz Survey Question 2 Layout

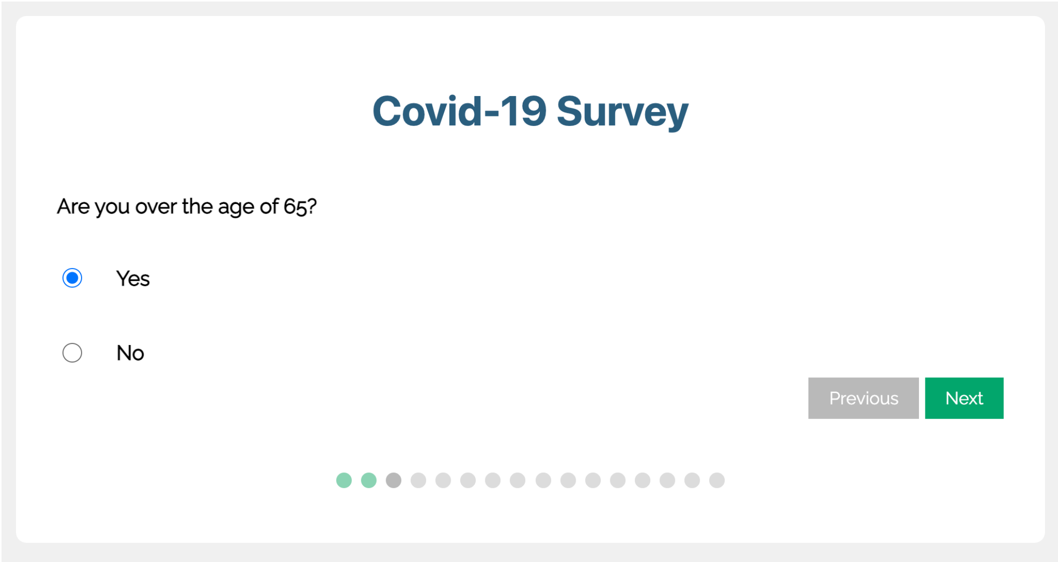


Figure 11.3: Cruz Survey Question 3 Layout

Each question is constructed with its unique id: `id="texti2"` and a specific name attribute: `name="ask_age"` (Fig.11.4) which is then manipulated within the synthesis function to convert each text (question) into speech. Therefore, if this set of questions needs to be changed in the future, it would be only necessary to change its text paragraph instead of its function. As well, new questions can always be added or removed.

```
<div class="tab">
      <div id="textai2">
        Are you over the age of 65?
      </div>
      <input type="radio" id="html" oninput="this.className = ''" name="ask_age" value="Yes" checked>
      <label for="yes">Yes</label><br>
      <input type="radio" id="css" oninput="this.className = ''" name="ask_age" value="No">
      <label for="no">No</label><br>
    </div>
```

Figure 11.4: Code for each Survey Question id name

The **SpeechSynthesisUtterance** (Fig.11.5), object allowed the access of the neural network voices and models to manipulate each text (question) with the speech services that enable Cruz to speak clearly and convey his questions in a human-like manner.

```
<script>
    // Initialize new SpeechSynthesisUtterance object
    let speech = new SpeechSynthesisUtterance();

    // Set Speech Language
    speech.lang = "en";

    let voices = []; // global array of available voices
    window.speechSynthesis.onvoiceschanged = () => {
      // Get List of Voices
      voices = window.speechSynthesis.getVoices();

      // Initially set the First Voice in the Array.
      speech.voice = voices[0];
      // speech.voice = voices.length;

      // Set the Voice Select List. (Set the Index as the value, which we'll use later when the user updates the Voice using the Select Menu.
      let voiceSelect = document.querySelector("#voices");
      voices.forEach((voice, i) => (voiceSelect.options[i] = new Option(voice.name, i)));
```

Figure 11.5: Speech Synthesis Function

```
var rad = document.getElementsByName('ask_age');
var prev = null;
for (var i = 0; i < rad.length; i++) {
    rad[i].addEventListener('change', function() {
        (prev) ? console.log(prev.value): null;
        if (this !== prev) {
            prev = this;
        }
        console.log(this.value)
        if (this.value == '') {
          //alert('Thank You');
          document.getElementById("nextBtn").style.display = "none";
        }else{
          document.getElementById("nextBtn").style.display = "inline";
        }
    });
}
```

Figure 11.6: Sample Code to Synthesize Question 3 into Speech

All the survey questions are grouped in one tab layout format, and Cruz is set to speak and ask questions according to the survey current tab question.

```javascript
function showTab(n) {
  // This function will display the specified tab of the form...
  var x = document.getElementsByClassName("tab");
  x[n].style.display = "block";
  //... and fix the Previous/Next buttons:
  if (n == 0) {
    document.getElementById("prevBtn").style.display = "none";
  } else {
    document.getElementById("prevBtn").style.display = "inline";
  }
  if (n == (x.length - 1)) {
    document.getElementById("nextBtn").innerHTML = "Submit";
  } else {
    document.getElementById("nextBtn").innerHTML = "Next";
  }
  //... and run a function that will display the correct step indicator:
  fixStepIndicator(n)
}
```

Figure 11.7: Code for Current Tab Question

```javascript
function nextPrev(n) {
  // This function will figure out which tab to display
  var x = document.getElementsByClassName("tab");
  // Exit the function if any field in the current tab is invalid:
  if (n == 1 && !validateForm()) return false;
  // Hide the current tab:
  x[currentTab].style.display = "none";
  // Increase or decrease the current tab by 1:
  currentTab = currentTab + n;
  // if you have reached the end of the form...
  if (currentTab >= x.length) {
    // ... the form gets submitted:
    document.getElementById("regForm").submit();
    return false;
  }
  // Otherwise, display the correct tab:
  showTab(currentTab);
  console.log(currentTab);
  console.log(x.length);
  for (var i = 0; i >= x.length; i++) {

  }
```

Figure 11.8: Code for the Next Tab Question

```
if (currentTab == 0) {
        speech.text = document.getElementById("textai0").innerHTML;
        // Start Speaking
        window.speechSynthesis.speak(speech);
      } else if (currentTab == 1) {
        speech.text = document.getElementById("textai1").innerHTML;
```

Figure 11.9: Cruz Application Speaking in Current Tab Question

11.12 Survey Questions Table

Below is the survey table of all the questions asked by the Cruz using Text to Speech communication in its interaction with the users.

Covid-19 Survey Questions:	**User Input Response:**
Hello, I hope everything is all right. My name is Cruz Robot. May I ask you a few Covid-19 question?	YES/NO
What's your name?	YES/NO
Are you over the age of 65?	YES/NO
Are you currently experiencing, any fever?	YES/NO
Are you currently experiencing, any difficulty breathing?	YES/NO
Are you currently experiencing, any cough?	YES/NO
Are you currently experiencing, any sore throat?	YES/NO
Are you currently experiencing, any loss of smell?	YES/NO
Are you currently experiencing, any chills?	YES/NO
Do you have any headache?	YES/NO
Do you have any nausea?	YES/NO
Have you been near anyone who was experiencing any of the prior symptoms mentioned?	YES/NO
Have you been near anyone who has tested positive for Covid?	YES/NO
Have you been waiting to receive a Covid-19 test result?	YES/NO
Have you traveled recently?	YES/NO
Your Covid-19 survey has been successfully completed, please click the *submit* button to save your answers. Thank you so much for your time, have a good day.	Submit

When the survey is completed, the user can submit its answer, which will then be stored in **phpmyadmin MySql Database**.

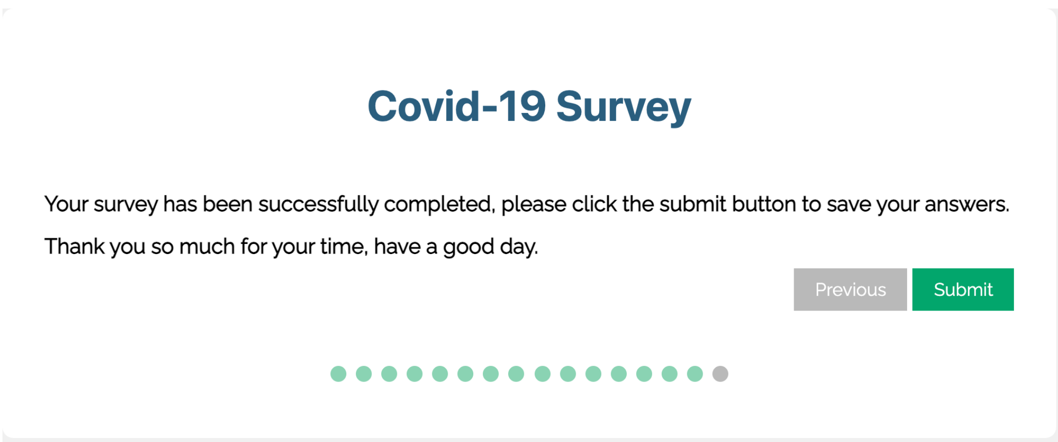


Figure 11.10: Cruz Survey Submission Layout

11.13 Database

With MySQL as a Relational Database Management System (RDBMS), all the Cruz user's answers to the covid-19 survey are saved and stored with the possibility to access and retrieve data if needed. Future Cruz developers can submit queries to the database via PHP to manipulate this information into or from the database.

Bellow (Fig.11.11) is the covid-19 **survey Cruz database storing all the test users answers data:**

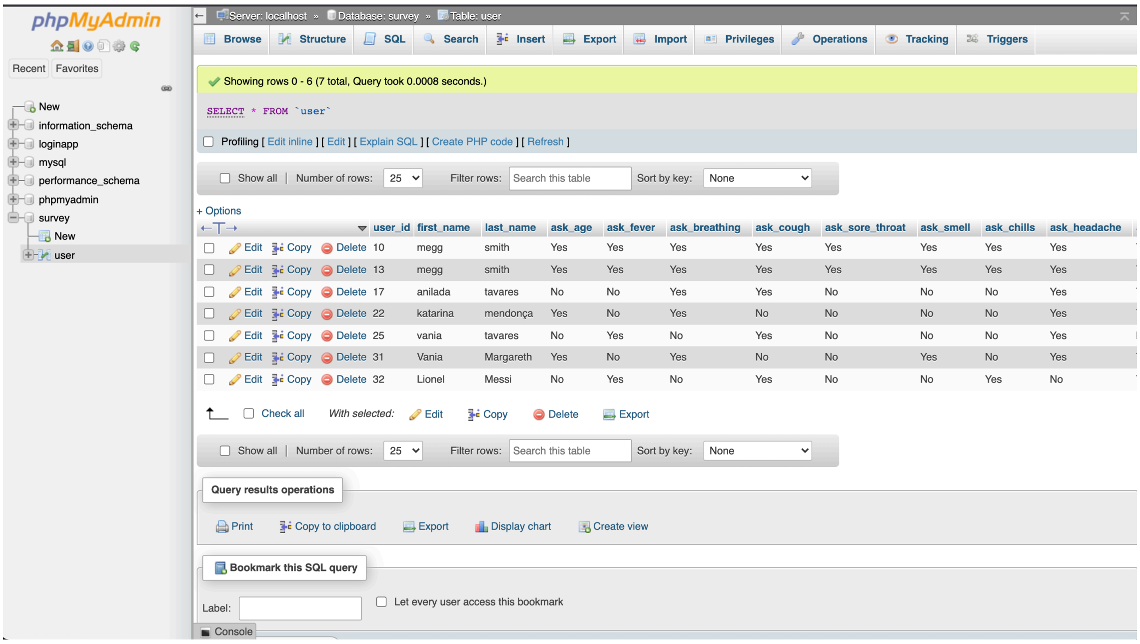


Figure 11.11: Survey Database

Chapter 12

Cruz Robot Social Implications

With the mandatory physical distancing and isolation dictated by the world's various governments to reduce the spread of the SARS-COVID-19 virus, Cruz Social Service Robot demonstrates a precious contribution to society.

Currently, there is no effective cure for SARS-COVID-19 diseases, and isolation can have an extreme impact on psychological well-being. Furthermore, unfortunately, physical and social distancing is not always possible, especially for healthcare professionals on the front line, who are at higher risk of exposure to SARS-COVID-19. Service Social robots such as Cruz are/could be vital to solving these dilemmas.

Identifying new vaccines and drug compounds is required to cure the COVID-19 virus. However, until an effective vaccine is developed, screening, as it is now, is providing the most accurate results in identifying molecules that contain the virus. Hence, the need and development of user-friendly Social Robots such as the Cruz AI Covid-19 Survey web application, which demonstrates the intelligent capability to interact with people by asking questions about SARS-COVID-19 to identify possible corona cases, will significantly contribute to the social distance measures to reduce the virus's spread.

Cruz Social Robot can interact with humans, possessing inherent immunity, making him the best suitable worker to intervene in a pandemic situation such as this. Instead of human-to-human proximity to identify the symptoms of COVID-19, Human-Cruz interaction will take place in approaching people to retrieve and analyze the user health state information to select possible infected individuals.

It is expected that users will positively accept Cruz social robot since its interface features meet all user's expectations: appearance, cognitive intelligence, social intelligence, and visual and communication skills. Furthermore, alongside others successfully introduced social robots in the market, Cruz is set to make a social impact by providing support, improvement, and protection to human life.

Chapter 13

Conclusion

The long-term goal of constructing social robots that can assist people is a challenging task, but this project has demonstrated that it is doable. Taking advantage of Artificial Intelligence Models, which replicate the human cognitive skills such as communicating using the Natural Language Processing Model and visualization using Convolution Neural Network Models, allowed the creation of an AI web application for Cruz social Robot.

Through this work, it became clear that understanding human intelligence and behavior, such as physicality, effectiveness, sociability, and intelligence is vital in designing social robots that can successfully be accepted and have a beneficial role within society.

The Cruz robot user interface development can be considered a positive achievement for the social robot community, especially during this time of the sars-covid-19 pandemic, where Cruz will undoubtedly make a difference in minimizing the spread of the virus and protecting human life.

Another point of interest in the Cruz user interface is its accessible Text to Speech algorithm of the sars-covid-19 survey, which can be easily manipulated with changes in the future. Whether it is through adding new survey questions or entirely rewriting the survey to match a different context for analysis.

Face detection features are becoming increasingly popular, and it's not going anywhere anytime soon. As such, the face detection feature of Cruz can always be used as its primary entry point of interaction with users. Furthermore, it can be upgraded by adding a face recognition feature on top to identify specific people in the future. Of course, such an addition would require a database to match the face detected to a specific individual, but it is always an open possibility.

Bibliography

[1] H Allan Hunt and Timothy L Hunt. The robots are coming. 1983.

[2] Vijay Kumar, George Bekey, Yuan Zheng, R Ambrose, A Sanderson, and B Wilcox. Industrial, personal, and service robots. DRAFT REPORT, 41, 2005.

[3] Frank Hegel, Claudia Muhl, Britta Wrede, Martina Hielscher-Fastabend, and Gerhard Sagerer. Understanding social robots. In 2009 Second International Conferences on Advances in Computer-Human Interactions, pages 169–174. IEEE, 2009.

[4] Desirée Gómez. Personal and social robots in education, 2018.

[5] Salvatore M Anzalone, Sofiane Boucenna, Serena Ivaldi, and Mohamed Chetouani. Evaluating the engagement with social robots. International Journal of Social Robotics, 7(4):465–478, 2015.

[6] Andreas Butz. Automated robotic concierge: User experience study.

[7] Gary Yeung, Alison L Bailey, Amber Afshan, Morgan Tinkler, Marlen Q Pérez, Alejandra Martin, Anahit A Pogossian, Samuel Spaulding, Hae Won Park, Manushaqe Muco, et al. A robotic interface for the administration of language, literacy, and speech pathology assessments for children. In SLaTE, pages 41–42, 2019.

[8] Michiel De Jong, Kevin Zhang, Aaron M Roth, Travers Rhodes, Robin Schmucker, Chenghui Zhou, Sofia Ferreira, João Cartucho, and Manuela Veloso. Towards a robust interactive and learning social robot. In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, pages 883–891, 2018.

[9] Amit Kumar Pandey, Rodolphe Gelin, and AMPSH Robot. Pepper: The first machine of its kind. IEEE Robotics & Automation Magazine, 25(3):40–48, 2018.

[10] Leendert Pieter van der Plas. Informatica en economie.

[11] Guy Zilbershtein. Designing a personal robot, 2018.

[12] American Luxury Staff. Personal Assistant, DJ, Media Hub: The Temi Personal Robot Does It All, 2017.

[13] Syamimi Shamsuddin, Luthffi Idzhar Ismail, Hanafiah Yussof, Nur Ismarrubie Zahari, Saiful Bahari, Hafizan Hashim, and Ahmed Jaffar. Humanoid robot nao: Review of control and motion exploration. In 2011 IEEE international conference on Control System, Computing and Engineering, pages 511–516. IEEE, 2011.

[14] Syamimi Shamsuddin, Hanafiah Yussof, Luthffi Ismail, Fazah Akhtar Hanapiah, Salina Mohamed, Hanizah Ali Piah, and Nur Ismarrubie Zahari. Initial response of autistic children in human-robot interaction therapy with humanoid robot nao. In 2012 IEEE 8th International Colloquium on Signal Processing and its Applications, pages 188–193. IEEE, 2012.

[15] A Kalouguine, V De-León-Gómez, C Chevallereau, S Dalibard, and Yannick Aoustin. A new human-like walking for the humanoid robot romeo. Multibody System Dynamics, 53(4):411–434, 2021.

[16] BLUE FROG ROBOTICS BUDDY. DEDICATED EMOTIONAL ROBOT SOLUTIONS, 2018.

[17] Beyin Chen, Gwo-Haur Hwang, and Shen-Hua Wang. Gender differences in cognitive load when applying game-based learning with intelligent robots. Educational Technology & Society, 24(3):102–115, 2021.

[18] Satoshi Shigemi, Ambarish Goswami, and Prahlad Vadakkepat. Asimo and humanoid robot research at honda. Humanoid robotics: A reference, pages 55–90, 2018.

[19] Cynthia Breazeal. Social robots: from research to commercialization. In Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, pages 1–1, 2017.

[20] Steve Crowe. Buddy Companion Robot SDK: Hands On, 2016.

[21] Supanut Konngern, Naphssorn Kaibutr, Nawaporn Konru, Thitinan Tantidham, Chih-Lin Hu, Tipajin Thaipisutikul, Timothy K Shih, and Pattanasak Mongkolwat. Assistive robot with action planner and schedule for family. In 2019 Twelfth International Conference on Ubi-Media Computing (Ubi-Media), pages 171–176. IEEE, 2019.

[22] Mary-Anne Williams. Robot social intelligence. In International Conference on Social Robotics, pages 45–55. Springer, 2012.

[23] Dylan F Glas, Satoru Satake, Florent Ferreri, Takayuki Kanda, Norihiro Hagita, and Hiroshi Ishiguro. The network robot system: enabling social human-robot interaction in public spaces. Journal of Human-Robot Interaction, 1(2):5–32, 2012.

[24] Ignazio Infantino, Agnese Augello, Umberto Maniscalto, Giovanni Pilato, and Filippo Vella. A cognitive architecture for social robots. In 2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI), pages 1–5. IEEE, 2018.

[25] Sushmita Mitra and Tinku Acharya. Gesture recognition: A survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 37(3):311–324, 2007.

[26] Jagdish Lal Raheja, Radhey Shyam, G Arun Rajsekhar, and P Bhanu Prasad. Real-time robotic hand control using hand gestures. Robotic Systems–Applications, Control and Programming, pages 413–428, 2012.

[27] Alessia Silvia Ivani, Alice Giubergia, Laura Santos, Alice Geminiani, Silvia Annunziata, Arianna Caglio, Ivana Olivieri, and Alessandra Pedrocchi. A gesture recognition algorithm in a robot therapy for asd children. Biomedical Signal Processing and Control, 74:103512, 2022.

[28] Laura Pérez-Mayos, Mireia Farrús, and Jordi Adell. Part-of-speech and prosody-based approaches for robot speech and gesture synchronization. Journal of Intelligent & Robotic Systems, 99(2):277–287, 2020.

[29] Siyang Wang, Simon Alexanderson, Joakim Gustafson, Jonas Beskow, Gustav Eje Henter, and Éva Székely. Integrated speech and gesture synthesis. In Proceedings of the 2021 International Conference on Multimodal Interaction, pages 177–185, 2021.

[30] Shenhan Qian, Zhi Tu, Yihao Zhi, Wen Liu, and Shenghua Gao. Speech drives templates: Co-speech gesture synthesis with learned templates. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 11077–11086, 2021.

[31] Michael A Goodrich and Alan C Schultz. Human-robot interaction: a survey. Now Publishers Inc, 2008.

[32] Haibin Yan, Marcelo H Ang, and Aun Neow Poo. A survey on perception methods for human–robot interaction in social robots. International Journal of Social Robotics, 6(1):85–119, 2014.

[33] Max Bajracharya, Mark W Maimone, and Daniel Helmick. Autonomy for mars rovers: Past, present, and future. Computer, 41(12):44–50, 2008.

[34] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Nachaat AbdElatif Mohamed, and Humaira Arshad. State-of-the-art in artificial neural network applications: A survey. Heliyon, 4(11):e00938, 2018.

[35] Maria Pateraki, Markos Sigalas, Georgios Chliveros, and Panos Trahanias. Visual human-robot communication in social settings. In Proceedings of ICRA Workshop on Semantics, Identification and Control of Robot-Human-Environment Interaction, 2013.

[36] Stefan Heinrich and Stefan Wermter. Towards robust speech recognition for human-robot interaction. In Proceedings of the IROS2011 Workshop on Cognitive Neuroscience Robotics (CNR), pages 29–34, 2011.

[37] Eric Deng, Bilge Mutlu, and Maja Mataric. Embodiment in socially interactive robots. arXiv preprint arXiv:1912.00312, 2019.

[38] Albert van Breemen, Xue Yan, and Bernt Meerbeek. icat: an animated user-interface robot with personality. In Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, pages 143–144, 2005.

[39] Elena Knox and Katsumi Watanabe. Aibo robot mortuary rites in the japanese cultural context. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 2020–2025. IEEE, 2018.

[40] Manja Lohse, Frank Hegel, Agnes Swadzba, Katharina Rohlfing, Sven Wachsmuth, and Britta Wrede. What can i do for you? appearance and application of robots. In Proceedings of AISB, volume 7, pages 121–126. Citeseer, 2007.

[41] Ibrahim A Hameed, Zheng-Hua Tan, Nicolai B Thomsen, and Xiaodong Duan. User acceptance of social robots. In Proceedings of the Ninth international conference on advances in computer-human interactions (ACHI 2016), Venice, Italy, pages 274–279, 2016.

[42] Ubtech Robotics Robotiklab. Cruzr Robot, a New Generation of Intelligent and Autonomous Service Robot, 2020.

[43] Terrence Fong, Illah Nourbakhsh, and Kerstin Dautenhahn. A survey of socially interactive robots. Robotics and autonomous systems, 42(3-4):143–166, 2003.

[44] Yadong Pan, Haruka Okada, Toshiaki Uchiyama, and Kenji Suzuki. Direct and indirect social robot interactions in a hotel public space. In 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO), pages 1881–1886. IEEE, 2013.

[45] Mehdi Jazayeri. Some trends in web application development. In Future of Software Engineering (FOSE'07), pages 199–213. IEEE, 2007.

[46] Gavin Bell. Building social Web applications: Establishing community at the heart of your site. " O'Reilly Media, Inc.", 2009.

[47] Margaret A Boden. Artificial intelligence. Elsevier, 1996.

[48] Jahanzaib Shabbir and Tarique Anwer. Artificial intelligence and its role in near future. arXiv preprint arXiv:1804.01396, 2018.

[49] Han LJ van der Maas, Lukas Snoek, and Claire E Stevenson. How much intelligence is there in artificial intelligence? a 2020 update. Intelligence, 87:101548, 2021.

[50] Hadi Salehi and Rigoberto Burgueño. Emerging artificial intelligence methods in structural engineering. Engineering structures, 171:170–189, 2018.

[51] Anam Mahmood. Tackling bias in machine learning models, 2022.

[52] Richard Szeliski. Computer vision: algorithms and applications. Springer Science & Business Media, 2010.

[53] Jim R Parker. Algorithms for image processing and computer vision. John Wiley & Sons, 2010.

[54] Andrej Karpathy et al. Convolutional neural networks for visual recognition. cs231n. github. io, 2018.

[55] Anindya Goswami, Sharan Rajani, and Atharva Tanksale. Data-driven option pricing using single and multi-asset supervised learning. International Journal of Financial Engineering, 8(02):2141001, 2021.

[56] Jacob Eisenstein. Introduction to natural language processing. MIT press, 2019.

[57] Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. Natural language processing: an introduction. Journal of the American Medical Informatics Association, 18(5):544–551, 2011.

[58] R Kibble. Introduction to natural language processing. London: University of London, 2013.

[59] Douglas Crockford. JavaScript: The Good Parts: The Good Parts. " O'Reilly Media, Inc.", 2008.

[60] Chenyang Li and Chunfang Li. Web front-end realtime face recognition based on tfjs. In 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), pages 1–5. IEEE, 2019.

[61] JDDS Rivera. Practical tensorflow. js. 2020.

[62] Julius Adorf. Web speech api. KTH Royal Institute of Technology, pages 1–11, 2013.

[63] Rasmus Lerdorf, Kevin Tatroe, Bob Kaehms, and Ric McGredy. Programming Php. " O'Reilly Media, Inc.", 2002.

[64] Tim Converse, Joyce Park, and Clark Morgan. PHP5 and MySQL bible, volume 147. John Wiley & Sons, 2004.

[65] Ashu Kumar, Amandeep Kaur, and Munish Kumar. Face detection techniques: a review. Artificial Intelligence Review, 52(2):927–948, 2019.

[66] Chengcheng Ning, Huajun Zhou, Yan Song, and Jinhui Tang. Inception single shot multibox detector for object detection. In 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), pages 549–554. IEEE, 2017.

[67] Feng Ye, Mingxu Ding, Enlai Gong, Xingwen Zhao, and Lijun Hang. Tiny face detection based on deep learning. In 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), pages 407–412. IEEE, 2019.

[68] Xiaochao Li, Zhenjie Yang, and Hongwei Wu. Face detection based on receptive field enhanced multi-task cascaded convolutional neural networks. IEEE Access, 8:174922–174930, 2020.

[69] Juan Du. High-precision portrait classification based on mtcnn and its application on similarity judgement. In Journal of Physics: Conference Series, volume 1518, page 012066. IOP Publishing, 2020.

[70] Paul Taylor. Text-to-speech synthesis. Cambridge university press, 2009.

[71] Thierry Dutoit. An introduction to text-to-speech synthesis, volume 3. Springer Science & Business Media, 1997.

[72] David T Chappell and John HL Hansen. A comparison of spectral smoothing methods for segment concatenation based speech synthesis. Speech Communication, 36(3-4):343–373, 2002.

[73] Simon King. An introduction to statistical parametric speech synthesis. Sadhana, 36(5):837–852, 2011.

[74] Aliya Deri and Kevin Knight. Grapheme-to-phoneme models for (almost) any language. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 399–408, 2016.

[75] Aäron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. SSW, 125:2, 2016.