**Research Project**

**(IT4010)**

**Group Assessment File**

**Project ID :** 2020-175

**Supervisor:** Prof.Koliya Pulasinghe

**Project Title:**

|  |
| --- |
| Sinhala Conversational Appointment Management System for Medical Domain |

**Group Details:**

|  |  |
| --- | --- |
| **Student ID** | **Student Name** |
| IT16234062 | Rajapakshe D.D.S |
| IT17029278 | U.L.N.P. Uswatte |
| IT17255820 | Kudawithana K.N.B |
| IT17043656 | Nishshanka N.A.B.D |

**Research Project (IT4010)**

**Student Assessment File**

**Project ID :** 2020-175

**Student ID :** IT16234062

**Student Name :** Rajapakshe D.D.S

**Research Domain:** Robotics and Intelligent Systems

**Project Title**

|  |
| --- |
| Sinhala Conversational Appointment Management System for Medical Domain |

**Project Sub Title**

|  |
| --- |
| Understanding user Natural Language. |

**Individual Component Abstract**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Developing a communication interface for patient most important 1st major point Understanding user Natural Language. It is present Sinhala medical termers voice into text. As a research developer most important research area,   * Voice to text : Getting a voice audio and it turn into Sinhala text(Google cloud API voice to text) * Dictionary : Creating a Sinhala Phonetic Dictionary for easy to future process.  |  |  | | --- | --- | | Character | Phoneme | | අ | AH | | ආ | AA | | ඇ | E | | ඈ | EE | | ඉ | IH |  * Converter : Using Dictionary, convert to Sinhala text into English alphabetic text & stored in DB(Hidden Markov Model (HMM)). * User Validation : As a Medical System most important thing is security Because As a person Health information is a personal thing. Therefore, accomplished to user Identification is most important. * Connectivity : connect to the Mobile app, RASA Framework, Flutter Mobile app SDK   User Verification Voice to Text  Converter English Text Database  Dictionary  Key words : RASA, Mobile app, Flutter, Hidden Markov Model (HMM), Google cloud API (voice to text), Python, Elastic Kibana, Machine Learning, Deep Learning, NLP, AI |

**Research Project (IT4010)**

**Student Assessment File**

**Project ID :** 2020-175

**Student ID :** IT17255820

**Student Name :** Kudawithana K.N.B

**Research Domain:** Robotics and Intelligent Systems

**Project Title**

|  |
| --- |
| Sinhala Conversational Appointment Management System for Medical Domain |

**Project Sub Title**

|  |
| --- |
| Natural Language Process |

**Individual Component Abstract**

|  |
| --- |
| 1. The users voice data extracted to Sinhala sentence which is given from voice to text translator are needed to Identify facts for filter those problems and important facts to give solutions. When start the filtering by system simultaneously start java key factor service for authenticate and give permission levels to users and change filtering results in different medical centers. As examples,  * When Patient using system, we installed in Nawaloka Surgical it should be only filtered and search Doctors and channelings in Nawaloka Surgical. Patient’s Search results should be only for location that software installed Channeling center.   After identifying the keywords using algorithm, need to generate query using MySQL or Elastic Kibana with Mongo DB   1. After this there is another thing when patient logging to system, he/she need to have details of their history, so we can create dashboard. After they entering to their profile adding their details, they can see who the doctor they met, what is the date finally they met and all the details of their disease. 2. When patient Make channeling appointment to doctor it should be recorded in database and need to send massage to channeling center counter for mentioning the report and when the channeling date in channeling counter patient should confirm their appointment and if patient need to cancel appointment it should be done before one hour of the channeling time for these we decided to use blockchains,  * A Practical Introduction to **Blockchain** with **Python**. ... As its core, a **blockchain** is a distributed database that allows direct transactions between two parties without the need of a central authority   Have to check whether the request is for diseases identification or doctor channeling in the patient’s question. Because of this have to create API for relevant medical question stream. Necessary medical facts data retrieve form created APIs. Those medical facts used for query generation process. |

**Research Project (IT4010)**

**Student Assessment File**

**Project ID :** 2020-175

**Student ID :** IT17043656

**Student Name :** Nishshanka N.A.B.D

**Research Domain:** Robotics and Intelligent Systems

**Project Title**

|  |
| --- |
| Sinhala Conversational Appointment Management System for Medical Domain |

**Project Sub Title**

|  |
| --- |
| Connecting an NLU into Rasa framework |

**Individual Component Abstract**

|  |
| --- |
| The import facts of the sentence identify and generate query which is given by previous function need to be trained medical terms knowledge. In this phase must Connecting an NLU into Rasa framework. Rasa is a standard infrastructure layer for developers to build, improve, and deploy better AI assistants. In here there are two data trainee types are available.  First one is Rule base type. For that case TensorFlow is the most suitable one. Because in here trained medical terms data already available in our database. Then check the database and generate the reply using that database.  Second one is AI base type. For that case json format is the most suitable format. Received query from the Natural Language Process will be used to identify the relevant response for the medical question.  Have to extract the user voice into text, it shows an Interface. To do, Important to connect a RASA NLU, therefore, have to train data in JSON format. And also received response text from the Dialogue Management Process will be retrieved in interface.  Have to create three SQL database for Doctor Information, Channeling Information and Hospital Information. Then create three Java Services for the relevant SQL databases. These three Java Services are mapped with each other. And these three services are running in update situation. Otherwise want to retrieve the data it is happening using Elastic Kibana.  Elastic Search is an open source, RESTful distributed and scalable search engine. Elastic search is extremely fast in fetching results for simple or complex queries on large amounts of data (Petabytes) because of its simple design and distributed nature. It is also much easier to work with than a conventional database constrained by schemas, tables.  Elastic Search provides a distributed, multitenant-capable full-text search engine with an HTTP web interface and schema-free JSON documents.  Kibana is an open source data exploration and visualization tool built on Elastic Search to help you understand data better. It provides visualization capabilities on top of the content indexed on an Elasticsearch cluster.  Using trained medical terms knowledge used to next phase which generates the dialogue for the medical question that user asked. |

**Research Project (IT4010)**

**Student Assessment File**

**Project ID :** 2020-175

**Student ID :** IT17029278

**Student Name :** U.L.N.P. Uswatte

**Research Domain:** Robotics and Intelligent Systems

**Project Title**

|  |
| --- |
| Sinhala Conversational Appointment Management System for Medical Domain |

**Project Sub Title**

|  |
| --- |
| Dialogue management of the Bot |

**Individual Component Abstract**

|  |
| --- |
| Generating the dialogue for the medical questions that the user asks is done by the knowledge of the trained medical terms. The Sinhala Mozilla Deep Speech model created for an AI Robo is designed in this phase which is an Automatic recognition engine mainly aiming at preparing speech recognition technology and trained models for developers.  Through the Dialog Management process the generated Sinhala text response is sent to the previous Natural Language Understanding Process. The Mozilla TTS which helps to convert text to speech aims at deep learning-based Text to Speech engines consists of two different model implementations which are based on Tacotron and Tacotron2. TTS provides a generic data loader which is easy to be used for new datasets.  Being along with this, you can hear a sample generated voice. In the end of this process a response dialogue will be generated. |