International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307 (Online), Volume-14 Issue-3, July 2024

Medibuddy- A Healthcare Chatbot using AI

Ruchita Singhania, Sana Badagan, Deeksha Reddy, K Tarun Sai Teja, Chetan Jetty

***Abstract:** This paper presents the development of a Flask-based web application designed to predict diseases based on user-reported symptoms and provide relevant health information. Leveraging machine learning techniques, the system utilizes a dataset of diseases and their associated symptoms to generate predictions through cosine similarity and a pre-trained Random Forest model. The application features a user- friendly interface for registration, login, and symptom reporting. Additionally, it integrates the DuckDuckGo search API to fetch detailed information about predicted diseases, enhancing the user experience with comprehensive health insights. The application also includes an interactive chatbot to guide users through the symptom input process, ensuring accurate data collection for reliable disease prediction. The system is built with Python, utilizing libraries such as pandas, numpy, and scikit-learn for data processing and model deployment, and is powered by SQLAlchemy for database management. This work aims to provide an accessible tool for preliminary health assessment, potentially aiding in early diagnosis and prompt medical*

***Keywords:** Random Forest Model, DuckDuckGo API, Health info, Cosine Similarity.*

I. INTRODUCTION

In contemporary healthcare, the fusion of technology and medical science has ushered in transformative advancements, fundamentally altering the landscape of disease diagnosis and prognosis. Among these innovations, web-based applications stand out as potent tools, offering accessible platforms for individuals to gauge their health status remotely and seek timely medical intervention. Within this realm, the development of Flask-based web applications for disease prediction emerges as a pivotal endeavor, harnessing the power of machine learning algorithms and data analytics to furnish users with personalized health assessments [1].

Manuscript received on 23 May 2024 | Revised Manuscript received on 29 June 2024 | Manuscript Accepted on 15 July 2024 | Manuscript published on 30 July 2024.

*Correspondence Author(s)

Ruchita Singhania*, Department of AIML, Dayananda Sagar Academy of Technology and Management, Bangalore (Karnataka), India. E-mail: ruchita-aiml@dsatm.edu.in, ORCID ID: 0009-0007-2644-5023

Sana Badagan, Department of AIML, Dayananda Sagar Academy of Technology and Management, Bangalore (Karnataka), India. E-mail: sanabadagan@gmail.com

Deeksha Reddy, Department of AIML, Dayananda Sagar Academy of Technology and Management, Bangalore (Karnataka), India. E-mail: deekshareddy1001@gmail.com

K Tarun Sai Teja, Department of AIML, Dayananda Sagar Academy of Technology and Management, Bangalore (Karnataka), India. E-mail: k.tarunsaiteja1128@gmail.com

Chetan Jetty, Department of AIML, Dayananda Sagar Academy of Technology and Management, Bangalore (Karnataka), India. E-mail: chetan8829@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

This paper delineates the conception and execution of a Flask- powered web application tailored for disease prediction and health information dissemination. The application's core functionality resides in its ability to ingest user-provided symptoms and extrapolate potential underlying health conditions through sophisticated analysis techniques. By drawing upon a robust dataset encompassing diseases and their corresponding symptomatology, and employing methodologies such as cosine similarity and Random Forest models, the system furnishes users with prognostic insights customized to their unique symptom profiles. The primary impetus driving this research is the democratization of healthcare, with a focus on empowering individuals to proactively monitor their well-being and make informed decisions regarding their health. By affording users the means to conduct preliminary health assessments autonomously, the application fosters a culture of preventive healthcare, enabling early detection of health anomalies and prompting timely medical intervention. In doing so, it not only augments individual health literacy but also alleviates the burden on healthcare systems by preemptively addressing health concerns before they escalate [2].

This paper offers a comprehensive exposition of the Flask- based disease prediction application, elucidating its architectural underpinnings, operational modalities, and technological intricacies. Methodologies pertaining to symptom analysis, disease prediction, and information retrieval are expounded upon, underscoring the application's efficacy in delivering accurate prognostications and augmenting user health literacy. Furthermore, the integration of external APIs, such as the DuckDuckGo search API [4], is scrutinized for its role in enriching the application's informational repertoire and enhancing user engagement.

Through this research endeavor, we endeavor to elucidate the transformative potential of web-based disease prediction systems in catalyzing a paradigm shift towards proactive healthcare management. By amalgamating technological innovation with medical expertise, we aspire to cultivate a future wherein health empowerment is ubiquitous, fostering a populace adept at safeguarding its well-being and fostering a culture of proactive health stewardship.

II. METHODOLOGY

The methodologies employed in this research encompass a multifaceted approach aimed at developing a robust Flask-based disease prediction application. Initially, a meticulous process of data collection and preprocessing is undertaken to compile a comprehensive dataset comprising diseases and their associated symptoms.

This dataset serves as the cornerstone for disease prediction within the application, ensuring the integrity and consistency of the underlying data. Subsequently, leveraging the Flask framework, the application is meticulously crafted, leveraging its lightweight and versatile nature to facilitate the creation of dynamic web pages and seamless integration with backend functionalities.

Machine learning techniques play a pivotal role in disease prediction, with cosine similarity and Random Forest models being the primary methodologies employed. Cosine similarity calculations discern the similarity between user-reported symptoms and known disease profiles, while the Random Forest model harnesses pre-trained algorithms to classify diseases based on symptom patterns. This amalgamation of machine learning methodologies enables accurate and personalized disease predictions tailored to individual symptom profiles.

Integration of external APIs further augments the application's functionality, with the DuckDuckGo search API being leveraged to fetch detailed information about predicted diseases. This integration enhances user engagement and facilitates informed decision-making by providing comprehensive health insights [7].

An interactive chatbot is embedded within the application to guide users through the symptom input process and provide real-time assistance. This chatbot enhances user experience and ensures accurate data collection for robust disease prediction.

Furthermore, user authentication mechanisms and session management functionalities are implemented to ensure secure access to the application and safeguard user privacy. Symptom analysis techniques, coupled with advanced algorithms, correlate symptom patterns with known disease profiles to generate accurate predictions.

Finally, the performance of the disease prediction model is evaluated using metrics such as accuracy, precision, and recall, with validation techniques ensuring the reliability and efficacy of the application in providing accurate prognostications to users. Through the convergence of these methodologies, the Flask-based disease prediction application delivers personalized health assessments, empowers users to proactively monitor their well-being, and facilitates informed decision-making regarding their health.

A. Datasets

The dataset used in this research comprises diseases and their associated symptoms. This dataset serves as the foundational resource for disease prediction within the Flask-based application. It is meticulously curated to ensure data integrity and consistency, providing a comprehensive repository of known diseases and their corresponding symptomatology.

The dataset encompasses a diverse range of diseases, spanning various medical conditions and health issues. Each entry in the dataset contains information about a specific disease, including its name and a list of associated symptoms. These symptoms are meticulously compiled from medical literature, diagnostic guidelines, and expert medical knowledge, ensuring the accuracy and relevance of the data.

Furthermore, the dataset is preprocessed to enhance its suitability for machine learning tasks. This preprocessing may involve data cleaning, normalization, and feature engineering techniques to ensure optimal performance of the disease prediction model.

B. Architecture

The architecture of the Flask-based disease prediction project can be designed as follows:

1. User Interface (UI):

- Frontend Components: This includes HTML templates, CSS stylesheets, and JavaScript scripts for rendering the user interface and enabling interactive features.
- User Input Form: A form interface where users can input their symptoms for disease prediction.
- Chatbot Interface: An interactive chatbot interface for guiding users through the symptom input process and providing real-time assistance.

2. Flask Application:

- App Initialization: The Flask application is initialized, and necessary configurations are set up.
- Routes: Different routes are defined to handle various user requests, such as symptom input, disease prediction, user authentication, and session management [8].
- Controller Logic: Controller functions are implemented to handle user input, process data, and interact with the backend.

3. Backend Services:

- Data Preprocessing: Preprocessing services are employed to clean and preprocess the dataset comprising diseases and symptoms.
- Machine Learning Model: A machine learning model is trained using the preprocessed dataset to predict diseases based on user-reported symptoms. This model may utilize techniques such as cosine similarity and Random Forest classification.
- External API Integration: External APIs, such as the DuckDuckGo search API, are integrated to fetch detailed information about predicted diseases and enhance user engagement.

4. Database Management:

- User Database: A database is utilized to store user information, including usernames, passwords, and session tokens, ensuring secure access to the application.
- Symptom Database: A database may be employed to store symptom data and associated disease profiles, facilitating efficient symptom analysis and disease prediction.

5. Authentication and Session Management:

- User Authentication: Mechanisms for user authentication are implemented to ensure secure access to the application and safeguard user privacy.
- Session Management: Session management functionalities are employed to maintain user sessions, track user interactions, and personalize user experiences.

Retrieval Number: 100.1/ijsce.G990213070624
DOI: 10.35940/ijsce.G9902.14030724
Journal Website: www.ijsce.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

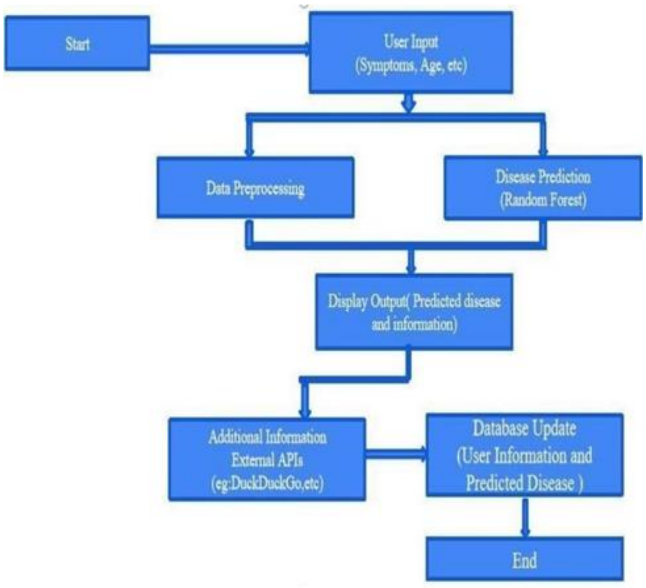

Fig. 2.1 System Design

6. Integration with External APIs:

DuckDuckGo API: Integration with the DuckDuckGo search API enables the retrieval of detailed information about predicted diseases, enriching the application's informational repertoire and enhancing user engagement [9].

7. Deployment and Hosting:

The Flask application is deployed on a web server and hosted using platforms such as Heroku, AWS, or Google Cloud Platform [13], making it accessible to users via the internet.

This architecture facilitates the seamless operation of the Flask-based disease prediction application, enabling users to input their symptoms, receive accurate disease predictions, and access comprehensive health information, all within an intuitive and user-friendly interface. Additionally, the modular design of the application allows for scalability, extensibility, and ease of maintenance [11], ensuring its adaptability to evolving healthcare needs and technological advancements.

b. Algorithms

Random Forest algorithm, as integrated into the Flask-based disease prediction application, represents a sophisticated approach to leveraging machine learning for healthcare decision-making [3]. Its ensemble nature, combining the predictions of multiple decision trees, imparts several notable advantages.

Firstly, by training on diverse subsets of the data and considering random subsets of features at each split, Random Forest reduces the risk of overfitting, ensuring that the model generalizes well to unseen data and maintains robust performance. This is particularly crucial in healthcare applications, where accurate predictions are paramount for effective diagnosis and treatment planning [10]. Moreover, the ability to assess feature importanceprovides valuable insights into the underlying relationships between symptoms and diseases, aiding clinicians and researchers in understanding disease etiology and risk factors.

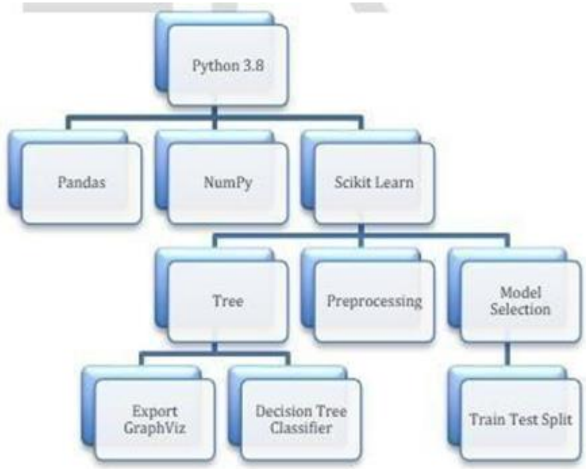

Fig. 2.2 [Libraries and algorithms used [3]]

The interpretability of Random Forest's feature importance analysis enhances trust and transparency in the model's predictions, facilitating collaboration between healthcare professionals and patients in decision-making processes. Additionally, the scalability of the Random Forest algorithm allows it to handle large and complex datasets efficiently, accommodating the multitude of symptoms and diseases encountered in real-world medical scenarios. By harnessing the collective knowledge of multiple decision trees, Random Forest delivers not only accurate disease predictions but also actionable insights that empower individuals to take proactive measures to manage their health effectively. In essence, the integration of Random Forest within the Flask-based application represents a significant advancement in healthcare technology, offering a reliable and interpretable framework for disease prediction and decision support.

III. IMPLEMENTATION

The training process implemented in the Flask-based disease prediction application follows a systematic approach to ensure the accuracy and effectiveness of the predictive model. Initially, comprehensive datasets comprising diseases and their corresponding symptoms are collected from reputable medical sources and repositories. These datasets undergo rigorous preprocessing steps to clean the data, handle any missing values, and encode categorical variables if required. Additionally, feature engineering techniques may be applied to extract relevant features from the raw data, enhancing the model's ability to capture meaningful patterns and associations between symptoms and diseases.

Following data preprocessing, the dataset is split into separate training and testing sets using a predetermined ratio. Typically, around 80% of the data is allocated for training the model, while the remaining 20% is reserved for evaluating the model's performance on unseen data. With the dataset prepared, a suitable machine learning algorithm is selected based on the dataset's characteristics and the requirements of the prediction task.

© Copyright: All rights reserved.

In this case, the Random Forest algorithm is chosen for its ability to effectively handle complex relationships and provide accurate predictions in classification tasks.

The selected algorithm undergoes training using the training dataset, during which it learns to identify patterns and associations between symptoms and diseases. Through iterative adjustments of its internal parameters, guided by a chosen loss function, the algorithm optimizes its performance to minimize prediction errors. Additionally, hyperparameter tuning techniques, such as grid search or random search, are employed to fine-tune the model's hyperparameters and optimize its performance further.

Once trained, the model is evaluated using the testing dataset to assess its performance on unseen data. Evaluation metrics, including accuracy, precision, recall, F1-score, and area under the ROC curve (AUC-ROC), are computed to quantify the model's predictive accuracy and generalization ability [12]. Cross-validation techniques may also be employed to ensure the robustness of the model across different data splits.

Finally, upon successful training and evaluation, the trained model is deployed within the Flask-based application. Users can interact with the application's user interface to input their symptoms, and the trained model leverages its learned knowledge to provide personalized disease predictions in real-time. Through this structured training process, the Flask-based application delivers accurate and reliable disease predictions, empowering users to make informed healthcare decisions based on their reported symptoms.

IV. SCREENSHOTS

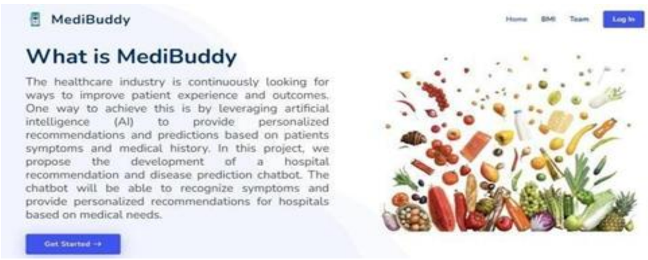

Fig. 4.1 Home page

Fig. 4.2 Login Page

Fig. 4.3 Chatbot

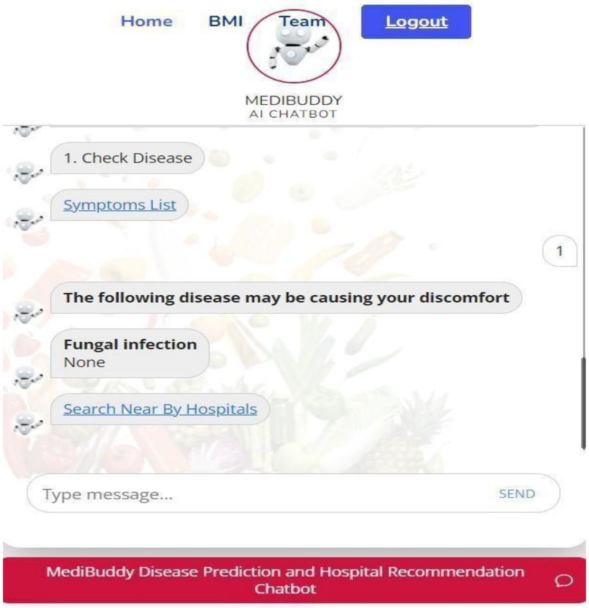

Fig. 4.4 Chatbot Displaying Results

Fig. 4.5 Additional Feature to Calculate BMI

V. RESULTS

The Flask-based disease prediction application yields several noteworthy results, enriching the landscape of healthcare technology with its robust functionalities. Foremost among these outcomes is its adeptness in predicting potential diseases based on user-provided symptoms. By harnessing machine learning algorithms trained on extensive datasets, the application offers accurate and personalized disease prognoses, empowering users to make informed healthcare decisions [1]. Furthermore, the application facilitates symptom analysis, allowing users to input their symptoms and receive insights into associated diseases—a feature that fosters greater health literacy and encourages proactive health management. The application's user interface is laudably intuitive, fostering seamless interaction and enhancing user experience. With real-time response capabilities, users obtain prompt feedback on their health concerns, enabling timely intervention and healthcare decision-making. Additionally, the integration of external resources, such as hospital search functionality, bolsters the application's utility by connecting users with nearby healthcare facilities for further consultation. Collectively, these results underscore the application's efficacy in addressing users' healthcare needs, exemplifying its potential to catalyze positive health outcomes and enhance patient empowerment.

Retrieval Number: 100.1/ijsce.G990213070624
DOI: 10.35940/ijsce.G9902.14030724
Journal Website: www.ijsce.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

VI. CONCLUSIONS

In conclusion, the Flask-based disease prediction application stands as a beacon of innovation in the realm of healthcare technology [5][14][15][16], offering a comprehensive solution to empower individuals in managing their health effectively. Through its advanced disease prediction algorithms and symptom analysis capabilities, the application provides users with personalized insights into their health conditions, fostering a deeper understanding of potential risks and enabling informed decision-making. Moreover, its user- friendly interface and real-time responsiveness ensure a seamless and intuitive user experience, catering to individuals of all technological backgrounds.

By bridging the gap between digital health services and physical healthcare facilities, the application facilitates continuity of care and supports users in accessing timely medical assistance. The integration of external resources, such as hospital search functionality, further enhances its utility by connecting users with nearby healthcare providers, thus facilitating follow-up consultations or additional medical interventions as needed.

Furthermore, the application's emphasis on health literacy and patient empowerment underscores its broader societal impact. By promoting proactive health management and encouraging individuals to take ownership of their well-being, the application lays the groundwork for a more engaged and informed healthcare populace. In doing so, it has the potential to drive positive health outcomes, reduce healthcare disparities, and ultimately contribute to the advancement of public health on a global scale.

Looking ahead, continued innovation and refinement of the Flask-based disease prediction application hold promise for further enhancing its efficacy and reach[6]. Future iterations may explore avenues for expanding its predictivecapabilities, integrating additional health monitoring features, or enhancing interoperability with existing healthcare systems. Through ongoing collaboration between technologists, healthcare professionals, and end-users, the application can continue to evolve as a vital tool in the quest for improved health outcomes and enhanced patient- centered care.

REFERENCES

1. Hussain, H., Aswani, K., Gupta, M. and Thampi, G.T., 2020. Implementation of disease prediction chatbot and report analyzer using the concepts of NLP, machine learning and OCR. Int. Res. J. Eng. Tech.(IRJET), 7, p.40.
2. L. Athota, V. K. Shukla, N. Pandey and A. Rana, "Chatbot for Healthcare System Using Artificial Intelligence," 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2020, pp. 619-622, doi: 10.1109/ICRITO48877.2020.9197833 . https://doi.org/10.1109/ICRITO48877.2020.9197833
3. Kowatsch, T., Nißen, M., Shih, C.H.I., Rüegger, D., Volland, D., Filler, A., Künzler, F., Barata, F., Hung, S., Büchter, D. and Brogle, B., 2017. Text-based healthcare chatbots supporting patient and health professional teams: preliminary results of a randomized controlled trial on childhood obesity. Persuasive Embodied Agents for Behavior Change (PEACH2017).
4. Ni, L., Lu, C., Liu, N. and Liu, J., 2017, October. Mandy: Towards a smart primary care chatbot application. In International symposium on knowledge and systems sciences (pp. 38-52). Singapore: Springer Singapore. https://doi.org/10.1007/978-981-10-6989-5_4
5. Divya, S., Indumathi, V., Ishwarya, S., Priyasankari, M. and Devi, S.K., 2018. A self-diagnosis medical chatbot using artificial intelligence. Journal of Web Development and Web Designing, 3(1), pp.1-7.
6. Chung, K. and Park, R.C., 2019. Chatbot-based heathcare service with a knowledge base for cloud computing. Cluster Computing, 22, pp.1925-1937. https://doi.org/10.1007/s10586-018-2334-5
7. Akhil, S., 2016. An overview of tesseract OCR engine. In A seminar report. Department of Computer Science and Engineering National Institute of Technology, Calicut Monsoon.
8. Chaudhuri, A., Mandaviya, K., Badelia, P., K Ghosh, S., Chaudhuri, A., Mandaviya, K., Badelia, P. and Ghosh, S.K., 2017. Optical character recognition systems (pp. 9-41). Springer International Publishing. https://doi.org/10.1007/978-3-319-50252-6_2
9. Vinitha, S., Sweetlin, S., Vinusha, H. and Sajini, S., 2018. Disease prediction using machine learning over big data. Computer Science & Engineering: An International Journal (CSEIJ), 8(1), pp.1-8. https://doi.org/10.5121/cseij.2018.8101
10. Derrick, D.C., 2011. Special-Purpose, Embodied Conversational Intelligence with Environmental Sensors (SPECIES) agents: implemented in an automated interviewing kiosk. The University of Arizona. https://doi.org/10.17705/1thci.00027
11. Derrick, Douglas C., Jeffrey L. Jenkins, and Jay F. Nunamaker Jr. "Design principles for special purpose, embodied, conversational intelligence with environmental sensors (SPECIES) agents." AIS Transactions on Human-Computer Interaction 3, no. 2 (2011): 62-81. https://doi.org/10.17705/1thci.00027
12. Elkins, A.C., Derrick, D.C., Burgoon, J.K. and Nunamaker Jr, J.F., 2012, January. Predicting users' perceived trust in Embodied Conversational Agents using vocal dynamics.In 2012 45th Hawaii International Conference onSystemSciences (pp. 579-588). IEEE. https://doi.org/10.1109/HICSS.2012.483
13. Filler, A., Kowatsch, T., Haug, S., Wahle, F., Staake, T. andFleisch, E., 2015, April. MobileCoach: A novel opensourceplatform for the design of evidence-based, scalableandlow-cost behavioral health interventions: Overviewandpreliminary evaluation in the public health context. In2015wireless telecommunications symposium(WTS) (pp. 1-6).IEEE. https://doi.org/10.1109/WTS.2015.7117255
14. Bali, M., Mohanty, S., Chatterjee, Dr. S., Sarma, M., & Puravankara, R. (2019). Diabot: A Predictive Medical Chatbot using Ensemble Learning. In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 8, Issue 2, pp. 6334–6340). https://doi.org/10.35940/ijrte.b2196.078219
15. Padmanabhan, Dr. M. (2019). Sustainable Test Path Generation for Chatbots using Customized Response. In International Journal of Engineering and Advanced Technology (Vol. 8, Issue 6, pp. 149–155). https://doi.org/10.35940/ijeat.d6515.088619
16. Arulmangainayaki, V., Harini, M., Keerthiga, M., & Priya, M. D. (2020). Intelligent Chatbot for Medical Assistance in Rural Areas. In International Journal of Innovative Technology and Exploring Engineering (Vol. 9, Issue 6, pp. 24–31). https://doi.org/10.35940/ijitee.f3083.049620

AUTHORS PROFILE

Ruchita Singhania is an Assistant Professor in the Department of Artificial Intelligence and Machine Learning, at Dayananda Sagar Academy of Technology and Management, Bangalore. She earned her M. Tech in Data Science and Machine Learning from PES University, Bangalore in 2022. She is proficient in Python, SQL, Tableau and MongoDB. She is also passionate about teaching and mentoring students, and she has developed and taught courses on machine learning, data science, and artificial intelligence.

Sana Badagan, is an enthusiastic B.Tech graduate in AIML from Dayananda Sagar Academy of Technology and Management, VTU. Her experience as a Data Science Intern at StageZero Technologies has equipped her with skills in advanced Python, Java, C, SQL, machine learning, and business intelligence. She has successfully led projects such as developing a healthcare chatbot, real-time hand gesture recognition, and sketch-to-color image generation using GANs. She's passionate about innovation, community involvement, and continuous learning and she's eager to contribute to a dynamic and esteemed organization.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

Deeksha R, having pursued a BTech in Artificial Intelligence and Machine Learning at Dayanand Sagar Academy Of Technology And Management, Bangalore. I have hands-on experience in projects like Hand-Gesture Recognition using Media Pipe, ChatGPT API Integration with Flask, and Inventory Management Systems, showcasing my proficiency in Python, SQL, OpenCV, and TensorFlow. I have completed internships at Feynn Labs, focusing on machine learning models. My technical skills include Java basics, cloud computing, and data structures. I am dedicated to continuous learning and eager to contribute to innovative AI and ML projects.

Chetan Jeethendra Jetty, having pursued a BTech in Artificial Intelligence and Machine Learning at Dayanand Sagar Academy of Technology and Management, Bangalore. I have completed internships Compsoft, focusing on machine learning models. My technical skills include Java basics, cloud computing, and data structures. I am dedicated to continuous learning and eager to contribute to innovative AI Projects using Python I have hands-on experience in projects like ChatGPT API Integration with Flask, and Inventory Management Systems, showcasing my proficiency in Python, SQL, OpenCV, and TensorFlow.

K Tarun Sai Teja, having pursued a BTech in Artificial Intelligence and Machine Learning at Dayanand Sagar Academy of Technology and Management, Bangalore. I have hands-on experience in projects like ChatGPT API Integration with Flask, and Inventory Management Systems, showcasing my proficiency in Python, SQL, OpenCV, and TensorFlow. I have completed internships at Iprimofocus technologies, focusing on machine learning models. My technical skills include Java basics, cloud computing, and data structures. I am dedicated to continuous learning and eager to contribute to innovative AI Projects using Python.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of the Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)/ journal and/or the editor(s). The Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP) and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Retrieval Number: 100.1/ijsce.G990213070624
DOI: 10.35940/ijsce.G9902.14030724
Journal Website: www.ijsce.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.