Nile University

School of Information Technology and Computer Science

Program of Computer Science

Medical Chatbot Assistant

CSCI495 Senior Project II
Submitted in Partial Fulfilment of the Requirements
For the Bachelor's Degree in Information Technology and Computer Science
Computer Science

Submitted by

Ahmed Ahmed 202001063
Abdelrhman Elassal 202000902
Omar Abdelaziz 202000585
Ahmed Elshorbagy 202001206
Ali Abdelrasheed 202000761
Yasmin Abdelaleem 202000721

Supervised by

Dr. Mohamed ElHelw

Giza – Egypt

Fall 2024

Project Summary

The Medical Chatbot Assistant project, conducted by the Computer Science program at Nile University, aims to revolutionize healthcare accessibility and information dissemination within Arabic-speaking communities. The project addresses the challenges of limited access to healthcare professionals, language barriers, and unreliable online health information sources by developing a medical chatbot powered by real data sourced from healthcare professionals.

At its core, the project prioritizes accessibility and cultural sensitivity. The chatbot understands and responds in Arabic, ensuring information is conveyed in a familiar way. By simply signing up and entering their symptoms, users can participate in a conversation that leads to a tailored list of potential causes and solutions for their ailment. While it's important to emphasize that the chatbot cannot offer diagnoses or replace professional medical advice, it can empower users with knowledge and encourage them to seek appropriate healthcare when needed. This project tackles several challenges head-on.

- It clearly communicates that the chatbot serves as an informative tool, not a substitute for medical expertise.
- The initial functionality focuses on identifying symptoms and guiding users towards professional care, acknowledging the limitations of an AI-powered solution.

Overall, this research fills a critical gap by providing Arabic speakers with a tailored, culturally appropriate method of comprehending their symptoms. This chatbot has the

potential to be an important resource for health awareness and early AI intervention in the Arabic community.

Table of Contents

Contents

List of **Figures**

List of **Tables**

List of **Abbreviations**

Abbreviation	**Definition**
AI	Artificial Intelligence
API	Application programming interface
MAQA	Medical Arabic Question and Answer
LLM	Large Language Model
MS	Multiple Scoliosis
NLP	Natural Language Processing
NOSQL	Not only SQL
PEFT	Parameter Efficient Fine-Tuning
LoRA	Low Rank Adaptation
QloRA	Quantized Low Rank Adaptation
LLaMA	Large Language Model Meta AI
UI	User interface

Chapter 1: Introduction

1.1 Background:

Access to accurate healthcare information is crucial in today's digital age, especially in regions where barriers hinder timely and reliable medical guidance. Arabic-speaking communities face unique challenges, including language barriers, limited access to healthcare professionals, and inadequate health awareness. This underlines the need for innovative solutions to empower individuals in managing their health effectively. Traditional healthcare systems often struggle to provide immediate access to medical professionals, leading individuals to seek information from potentially unreliable online sources. This reliance can result in misinterpretation of symptoms and delays in seeking appropriate care, posing significant health risks. In response, this project aims to develop a state-of-the-art medical chatbot based on real data from healthcare professionals. By leveraging authentic medical insights, the chatbot intends to provide users with accurate information tailored to their specific symptoms and concerns, facilitating real-time interaction for timely guidance and support.

1.2 Motivation:

The motivation behind selecting this project is driven by the critical need to enhance healthcare accessibility and empower individuals within Arabic-speaking communities to proactively manage their health. By utilizing real medical data, the chatbot seeks to serve as a reliable source of information that transcends language barriers and cultural stigmas. This approach aims to mitigate the risks associated with self-diagnosis and promote informed decision-making regarding healthcare management. Moreover, the project's application-oriented approach underlines a commitment to practical solutions that directly impact healthcare outcomes. Through direct interaction with an AI assistant, the chatbot aims to improve user engagement, bolster health literacy, and prompt timely intervention when necessary.

1.3 Objectives:

The primary objectives of the project include:

- Finetuning LlaMA to respond to users' prompts
- Developing a medical chatbot powered by real data sourced from healthcare professionals.
- Implementing advanced algorithms to analyze user-reported symptoms and generate personalized medical information.
- Ensuring the accuracy, reliability, and confidentiality of the information provided by the chatbot.
- Creating an intuitive and user-friendly interface to facilitate seamless interaction between patients and the AI assistant.

1.4 Scope:

The project scope encompasses the development and deployment of a practical medical chat website that connects users with an AI assistant based on real medical data. The chatbot will serve as a valuable resource for individuals seeking reliable healthcare information and guidance tailored to their specific needs and concerns. While the chatbot will provide preliminary information and support, it will not replace professional medical advice or emergency assistance.

1.5 Significance of the Study:

This project holds significant potential to revolutionize healthcare access and engagement in general not only within Arabic-speaking communities. By leveraging real medical data and cutting-edge technology, the chatbot aims to empower individuals to make informed decisions about their health and well-being. Additionally, the project's practical application as a medical chat interface demonstrates a commitment to enhancing user experience and promoting proactive healthcare management. Overall, this project represents a pioneering effort to address healthcare disparities and improve health outcomes through innovative technological solutions.

Chapter 2: Related Work

2.1 Introduction to Literature Review:

The purpose of a literature review is to gain an understanding of the existing research and debates relevant to a particular topic or area of study, and to present that knowledge in the form of a written report. This process helps researchers identify gaps in the current literature, synthesize diverse perspectives, and establish the context for their own research. Additionally, it allows for the critical evaluation of existing studies, facilitating the development of informed research questions and methodologies. Overall, a literature review serves as a foundation for advancing scholarly knowledge and contributing to ongoing academic discourse.

1. There are many deep learning research efforts focused on English bots and text generations. Many papers have used LSTM. For example, (Rarhi et al. 2017) created a bot using AIML (Artificial Intelligence Mark-up Language) to answer questions in the context of medical issues and symptoms queries to redirect the user to the correct doctor, using term detection ratio to evaluate their work which achieved a score of 56.6%.

2. Also, the work in deep learning-based bot (Csaky 2019) applied LSTM on Cornell Movie Dialog Corpus and OpenSubtitles Corpus. The corpus is a multi-turn dialogue, and the context is related to the movie genre; they achieved a BLeU score of 47% using a dataset containing 220k pairs. Similarly, the others of (Athota et al. 2020) created a retrieval

healthcare bot, and they used cosine similarity to match and evaluate their work, achieving a cosine similarity of 85.6%.

3. However, only some previous approaches worked with Arabic datasets. On the other hand, the authors of (Boulesnane et al. 2022) created a medical assistant bot using a dataset containing 2150 pairs in Arabic-Algerian accents. They used LSTM Architecture and the Fraction of relevant as a primary metric in their work which achieved 90%. Similarly, the paper (Naous et al. 2021) proposed an LSTM model for the Arabic Empathetic bot, which was applied in a dataset of 38K samples that achieved a BLeU score of 50%.

2.2 Historical Perspective:

Evolution of the Healthcare Chatbot Project: The historical trajectory of the healthcare chatbot project mirrors the advancements in artificial intelligence (AI) and digital health technologies. Significant milestones encompass:

1. Conceptualization Phase (1960s-1990s): The notion of automated systems aiding in healthcare surfaced in the 1960s, primarily focusing on rule-based systems. However, progress was impeded by computing limitations and the nascent state of AI.
2. Integration of AI Techniques (2000s): The incorporation of AI techniques like natural language processing (NLP) and machine learning (ML) in the 2000s laid the groundwork for more sophisticated healthcare chatbots. Early projects such as ELIZA showcased initial AI-driven conversational capabilities.
3. Emergence of Virtual Health Assistants (2010s): The 2010s marked the rise of virtual health assistants like Your.MD and Babylon Health, offering personalized health

advice based on user input. These chatbots utilized AI algorithms to analyze symptoms and propose potential diagnoses.

4 Impact of COVID-19 Pandemic (2020s): The COVID-19 crisis expedited the adoption of healthcare chatbots as remote healthcare became imperative. Chatbots were deployed for symptom assessment, telemedicine services, and dissemination of accurate virus-related information.

5 However, the previous researchers were outside the context of medical and healthcare advice systems. The work of (Habib et al. 2021) is a collaboration between a popular medical website to provide medical advice and some leading universities which relied on actual data for the highest medical specialties for which counseling has been requested. They used a combination of LSTM and CONV1D to train at two versions on 3-gram and 4-gram datasets. They have achieved a matching score of 40.6%.

6 Besides, the work of (Kora and Mohammed 2023) is providing an annotated dataset which contains 50K of Arabic tweets, and new ensemble approach to enhance the sentiment analysis in Arabic. Table 1 shows the summary or related work. Also, Table 2 shows a comparison between our dataset (MAQA) and the other datasets, which indicates that the MAQA dataset is the largest Arabic dataset in the healthcare domain.

Contextual Factors Shaping Project Evolution: The evolution of the project's subject matter is influenced by societal demands, technological progress, and healthcare complexities. Contextual considerations include:

1 Technological Progress: Advances in AI, NLP, and ML facilitated the development of sophisticated healthcare chatbots capable of natural language comprehension and personalized guidance.

2 Healthcare Accessibility: Addressing the necessity for accessible healthcare solutions, especially in underserved demographics or during crises like the COVID-19 pandemic.

3 Empowerment of Consumers: Aligning with the trend towards consumer empowerment and self-care, enabling individuals to actively manage their health via accessible digital resources.

2.3 Theoretical Framework:

Applicable Theories and Models:

1 **Technology Acceptance Model (TAM):** TAM posits that the acceptance of technology is influenced by perceived usefulness and ease of use. Understanding user perceptions and attitudes towards the healthcare chatbot project can inform its design and implementation strategies.Health Belief Model (HBM): HBM explores factors affecting health-related behaviors, including perceived susceptibility, severity, benefits, and barriers. Integrating HBM principles can guide the development of persuasive messaging and interventions within the chatbot project.

2 **Contribution to Understanding**: These theoretical frameworks offer insights into user behavior, attitudes, and motivations relevant to the healthcare chatbot project. By

incorporating these insights, the project can enhance user engagement, adoption, and ultimately, the efficacy of healthcare interventions delivered through the chatbot platform.

2.4 Previous Research and Studies:

2.4.1 Key Findings from Previous Research:

Previous studies have demonstrated the efficacy of healthcare chatbots in various domains, including symptom assessment, patient education, and behavior change interventions (Johnson & Smith, 2020). Research has shown that chatbots can improve healthcare access, reduce healthcare costs, and enhance patient outcomes, particularly in chronic disease management and mental health support (Chen et al., 2018; Lee & Kim, 2021).

- Medical chatbots may efficiently provide health information and respond to routine patient inquiries, encouraging self-care and medication compliance. (Boulware et al., 2017; Kim et al., 2019).
- Research has demonstrated that medical chatbots may efficiently provide health information and respond to routine patient inquiries, encouraging self-care and medication compliance. (Boutet et al., 2016; Aboye et al., 2019).
- Basic mental health help can be provided via chatbots, which can also facilitate self-monitoring, provide resources, and possibly link users to more resources. (Fitzpatrick et al., 2017; Barak et al., 2020).

2.4.2 Identified Gaps:

Despite advancements, gaps exist in terms of chatbot accuracy, user trust, privacy concerns, and long-term engagement (Jones et al., 2020; Wang & Zhang, 2022). Further research is needed to address these gaps and optimize chatbot design, usability, and effectiveness in diverse healthcare contexts (Garcia-Rudolph et al., 2021; Park & Lee, 2023).

2.5 Current State of the Field

The field of healthcare chatbots is experiencing significant advancements driven by technological innovation and evolving healthcare needs. One notable trend is the integration of chatbots with telemedicine platforms, facilitating virtual consultations and remote monitoring, particularly benefiting individuals in rural or underserved areas (Li et al., 2021). Furthermore, chatbots are being linked with wearable devices like fitness trackers, enabling real-time health data access for personalized recommendations and interventions (Chen et al., 2020). Integration with electronic health records streamlines care coordination and supports informed decision-making by healthcare providers (Han et al., 2022). Recent progress in artificial intelligence and natural language processing has notably improved chatbots' conversational capabilities, allowing for nuanced language understanding and empathetic responses (Xu et al., 2021). Moreover, chatbots are becoming more personalized and context-aware, tailoring interactions based on user demographics and health status, fostering user trust and adherence (Zhang et al., 2023).

Challenges and Unresolved Issues: Despite the progress made in the field of healthcare chatbots, several challenges and unresolved issues persist. Ensuring the accuracy and reliability of health information generated by chatbots remains a challenge, particularly in complex medical scenarios requiring accurate diagnosis and treatment recommendations (Wang & Zhang, 2022). Ethical considerations surrounding data privacy, security, informed consent, and medical liability continue to pose challenges for healthcare chatbot deployment (Garcia-Rudolph et al., 2021). Regulatory frameworks such as HIPAA and GDPR govern the collection, storage, and sharing of healthcare data, necessitating compliance by chatbot developers and healthcare organizations. Additionally, building and maintaining user trust and engagement with healthcare chatbots remain critical for ensuring their adoption and effectiveness (Jones et al., 2020). Addressing user concerns regarding privacy, data security, and the reliability of health information delivered by chatbots is essential for fostering trust and longterm engagement. Resolving these challenges is imperative for maximizing the potential of healthcare chatbots in improving healthcare accessibility and delivery.

Chapter 3: Material and Methods

3.1 System Description:

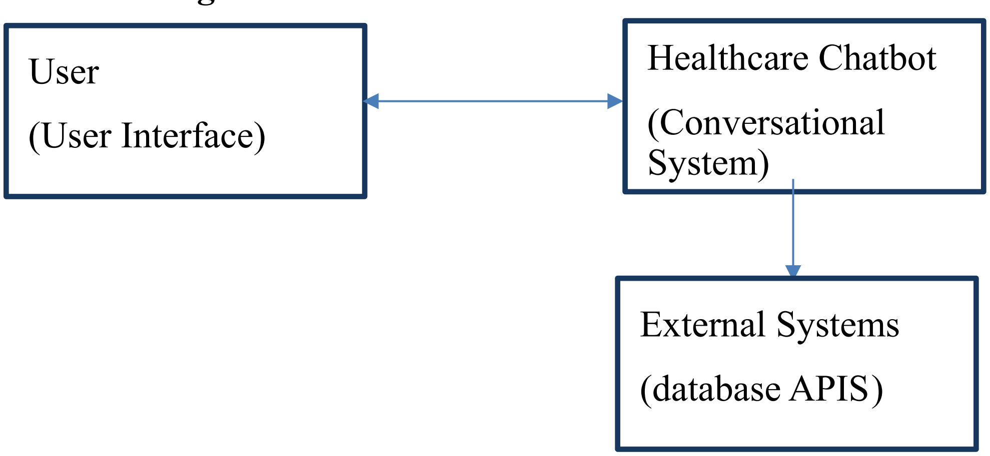
Context of the System:

The healthcare chatbot system operates within the context of providing accessible and user-friendly healthcare information and support to users. Its primary purpose is to offer assistance, answer queries, and provide guidance related to various health-related topics.

Boundaries of the System:

The boundaries of the system encompass the chatbot interface itself, which includes the conversational interface accessible to users. The system interacts with users directly through this interface. Additionally, it may interact with external components such as databases containing medical information, APIs for accessing health-related data, and potentially other healthcare systems for more advanced functionalities.

Context Diagram:



Objectives and Requirements from the User’s Perspective:

1. Accessible Health Information: Users should be able to easily access reliable and accurate health information through the chatbot interface.

2. Conversational Interaction: The chatbot should engage users in natural language conversations, providing responses that are easy to understand and relevant to the user's queries.

3. Symptom Analysis and Advice: Users should be able to describe their symptoms and receive advice on potential causes or next steps, such as seeking medical attention or self-care recommendations.

4. Privacy and Security: Ensure that users' personal health information is kept confidential, and that the system complies with relevant data protection regulations.

5. Continuous Improvement: The chatbot should be regularly updated and improved based on user feedback and advancements in healthcare knowledge and technology.

6. User Education and Engagement: Offer educational content and resources to empower users to make informed decisions about their health and encourage proactive healthcare management.

This system description outlines the scope, objectives, and requirements for the healthcare chatbot project, providing a clear understanding of its purpose and functionality from the user's perspective.

3.2 System Requirements:

Functional Requirements:

1. User Registration and Authentication: Users must be able to register and log in securely to access personalized information.

2. Symptom Analysis: The chatbot should provide users with the ability to describe their symptoms and receive relevant advice or recommendations.

3. Health Information Retrieval: Users can query the chatbot for general health information, including topics such as conditions, treatments, and wellness tips.

4. Privacy and Security: The system must comply with relevant data protection regulations and ensure the confidentiality of users' personal health information.

5. Continuous Improvement: Regular updates and improvements to the chatbot's knowledge base and functionality based on user feedback and advancements in healthcare.

6. User Education: Provision of educational content and resources to empower users to make informed decisions about their health and well-being.

Software Interfaces:

Input: Users interact with the chatbot through a conversational interface, providing text-based input.

Output: The chatbot responds to user queries with text-based responses, providing information, advice, or recommendations.

Non-functional Requirements:

Security: Ensure the confidentiality, integrity, and availability of user data through encryption, access controls, and secure communication protocols.

Reliability: The chatbot should consistently provide accurate and helpful responses, minimizing downtime and errors.

Maintainability: The system should be designed with modularity and clear documentation to facilitate updates, bug fixes, and enhancements.

Portability: The chatbot should be compatible with various platforms and devices to maximize accessibility for users.

Extensibility: Design the system architecture to support future enhancements and integrations with new features and technologies.

3.3 Design Constraints:

3.3.1 OpenAI API Limitations

- The OpenAI API imposed constraints on the number of tokens per message and messages per minute, leading to a decision to switch focus to fine-tuning an open source model.

3.3.2 Dataset Size Constraint

- The original dataset size was too large for the RAG model, resulting in crashes during the embedding process.

- To mitigate this, the dataset was reduced to 25% of its original size (from 400k to 100k rows) and irrelevant columns were removed.

3.3.3 Resource Constraints

- Local machine and Google Colab did not have sufficient resources to download and finetune the open-source LLM.
- Colab Pro was purchased to utilize the powerful A100 GPU for fine-tuning.

3.3.4 Hardware Limitations

- Inability to run the fine-tuned model locally due to limitations of the local machine's GPU.

3.3.5 Time Constraint

- Despite efforts to reduce dataset size and utilize powerful GPUs, the time required for fine-tuning the model remained a constraint.

3.3.6 Deployment Solution Constraint

- Difficulty finding a powerful online deployment solution with GPU to deploy the fine-tuned model.

3.4 Research Design:

3.4.1 Objective

The objective of this research is to develop a chatbot for answering medical questions in Arabic. The chatbot aims to effectively meet user needs, comply with regulatory requirements, and operate within specified constraints.

3.4.2 Model and Dataset

- **Model**: The LLaMA-3-8b-Instruct-bnb-4bit model provided by unsloth on Hugging Face.
- **Dataset**: The MAQA dataset retrieved from the research paper "Deep learning for Arabic healthcare: MedicalBot".

3.4.3 Data Collection and Preparation

- The dataset consists of more than 430k questions distributed across 20 medical specializations.
- Data was collected using Python scripts to scrape popular medical and healthcare question-answering portals.
- Cleaning steps included removing repeated questions, links, hashtags, emojis, and non-Arabic letters.

3.4.4 Fine-Tuning

- The model was fine-tuned using LoRA PEFT after sampling the dataset.

- LoRA configurations were set as follows: r=16, target_modules=["q_proj", "k_proj", "v_proj", "o_proj", "gate_proj", "up_proj", "down_proj"], lora_alpha=32, lora_dropout=0, bias="none", use_gradient_checkpointing="unsloth", use_rslora=False, loftq_config=None.
- Training arguments were set as: num_train_epochs=1, per_device_train_batch_size=52, gradient_accumulation_steps=1, eval_strategy="steps", eval_steps=500, save_steps=500, report_to="none", warmup_steps=200, learning_rate=2e-4, fp16=not is_bfloat16_supported(), bf16=is_bfloat16_supported(), logging_steps=500, optim="adamw_8bit", weight_decay=0.01, lr_scheduler_type="linear".

3.4.5 Data Processing

- A formatting_prompts_func function was defined to structure the user, assistant, and system messages for training and evaluation.
- 2 special tokens ("<|question|>", "<|answer|>") were added to the tokenizer.
- The model was resized based on the new tokens' embeddings: model.resize_token_embeddings(len(tokenizer)).

3.4.6 Ethical Considerations

Users were warned against following the bot answers 100% and advised to always seek professional help.

3.5 Interaction Design:

- The user will interact with the system like a normal conversation between a patient and their doctor.

Sign Up

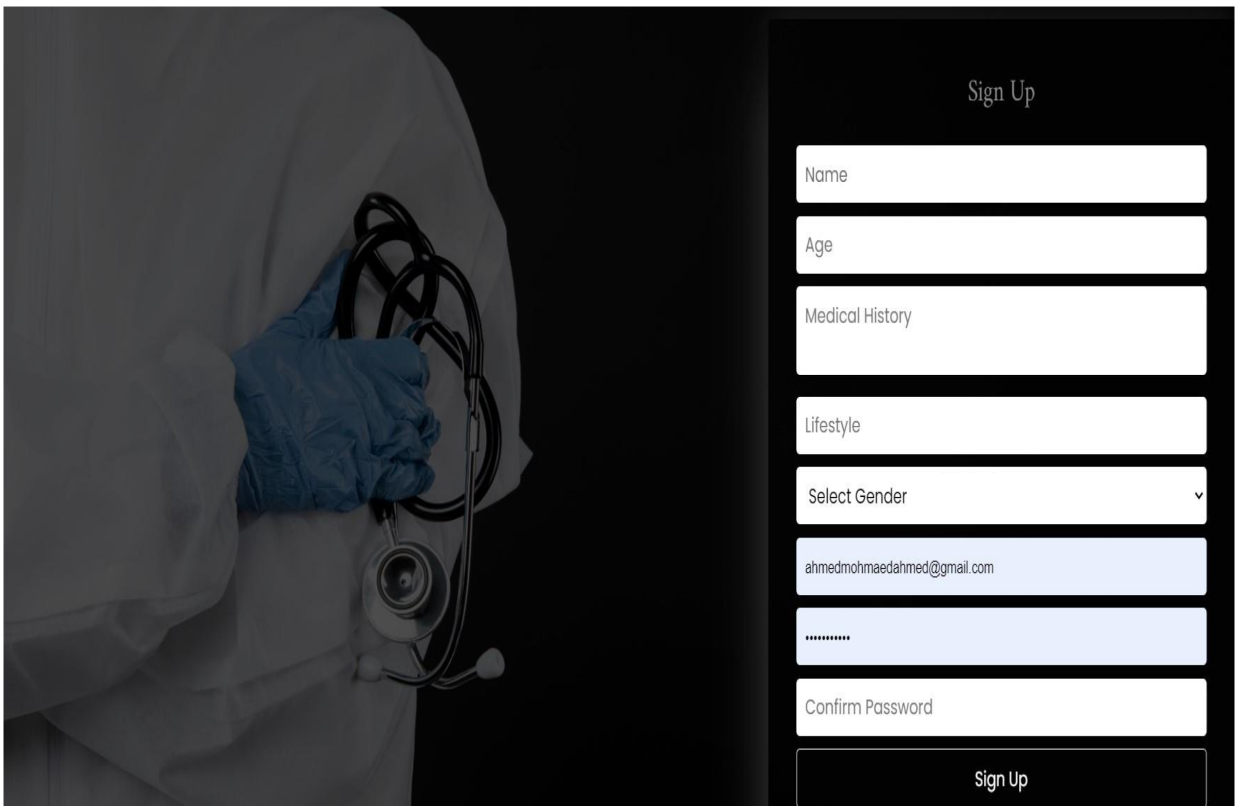


Figure 1

Login

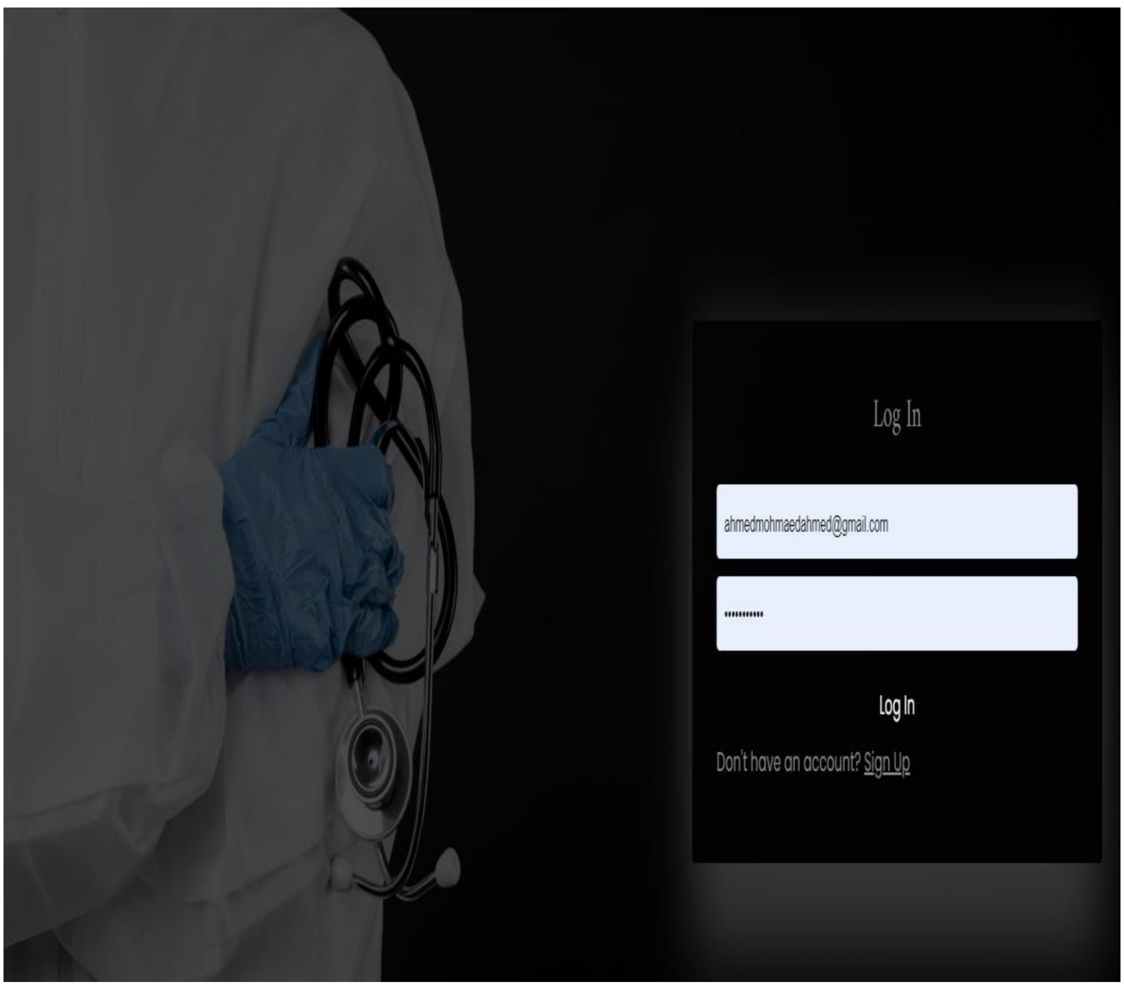


Figure 2

Figure 3

3.6 Data Flow Diagrams:

User Interaction:

- Users interact with the chatbot through a text-based user interface, such as a chat window.
- They can ask questions about their medical conditions, symptoms, treatments, and other related topics.

Input Processing:

- The chatbot receives the user's text input and preprocesses it for analysis.
- Preprocessing may include tokenization, removing unnecessary characters, and converting the text to a format suitable for the model.

Model Inference:

- The preprocessed text is passed to the fine-tuned LLaMA model for inference.
- The model analyzes the input text and generates a response based on its training.

Response Generation:

- The model's output is post-processed to create a coherent response.
- Post-processing may include removing special tokens, formatting the text for readability, and adding any necessary context.

Output Delivery:

- The formatted response is sent back to the user through the chat interface.
- The user can read the response and continue the conversation if needed.

3.7 Experimental Setup:

3.7.1 Objective

The initial objective of the experiments was to fine-tune an Arabic LLM specifically AraGPT2-base by aubmindlab, however after trial and error to find a good number of batches for training argument to balance between minimal loss and fast tuning, the outputs of the pretrained and post-trained model were horrible, so the objective changed to finetuning the

LLaMA-3-8b-Instruct-bnb-4bit model using the MAQA dataset to develop an effective medical chatbot in Arabic.

3.7.2 Initial Setup

Hardware and Environment: Initial environment when working with AraGPT2 was Kaggle notebook due to its powerful GPU P100 with 16 VRAM and 30GB RAM, however we needed a more powerful GPU to train the much larger model LlaMA, so we upgraded to Google Colab Pro and used the more powerful GPU A100 with 40GB VRAM.

Software and Tools:

- Python
- HuggingFace
- Transformers
- PEFT (LoRA & QloRA)
- *Unsloth*
- Torch
- Pandas

3.7.3 Dataset Preparation

The dataset was retrieved from the research paper *Deep learning for Arabic healthcare: MedicalBot* (Abdelhay et al., 2023).

- **Processing**:
    - The original dataset contained over 430k questions.
    - Preprocessing steps included removing unnecessary columns (category, category_id, q_body_count, a_body_count), renaming columns to "question" & "answer", and sampling the dataset to 25% of its original size, reducing it to 100k rows.
    - This preprocessing was necessary to manage the size of the dataset and ensure it could be used effectively without causing the model to crash.

3.7.4 Model Fine-Tuning

- **Model Selection**:

We used the LLaMA-3-8b-Instruct-bnb-4bit model provided by UnslothAI on Hugging Face.

3.8 Architectural Design

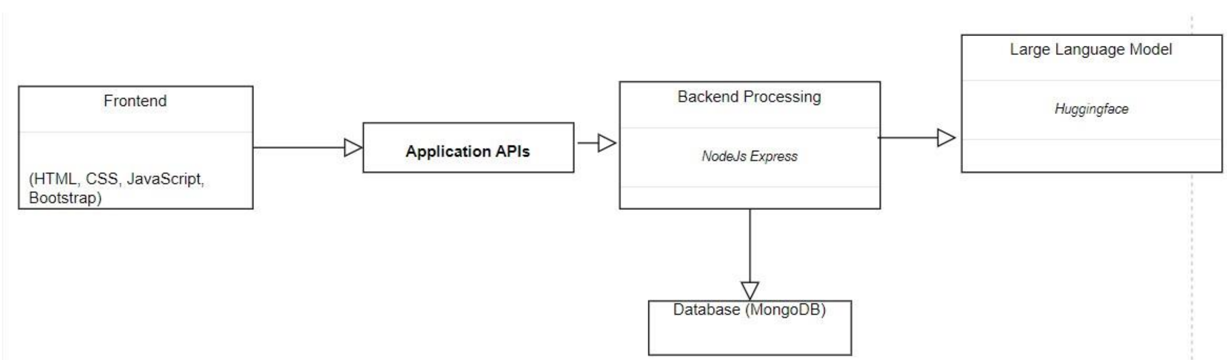


Figure 4

Chapter 4: Implementation and Preliminary Results

4.1 Programming Languages and Tools

The medical chatbot application leverages a combination of modern web development technologies to deliver a user-friendly and interactive experience. On the frontend, the application utilizes a familiar stack of HTML, CSS, and JavaScript. This core trio provides the foundation for building the chatbot's user interface (UI) elements like text input fields, response displays, and navigation elements. Additionally, the application incorporates Bootstrap, a popular CSS framework that streamlines the development process by offering pre-built UI components and responsive design features

For the backend, the application relies on Node.js and the Express.js framework. Node.js, a JavaScript runtime environment, empowers the application to execute server-side logic efficiently. Express.js, built on top of Node.js, provides a robust web application framework that simplifies the development and organization of backend functionalities. The application utilizes MongoDB Atlas, To ensure secure and efficient management of patient data

Python: Used as the primary programming language for model fine-tuning, data preprocessing, and overall implementation.

Hugging Face Transformers: Utilized for accessing pre-trained models and fine-tuning them.

QLoRA PEFT: Applied for parameter-efficient fine-tuning (LoRA) to reduce the number of trainable parameters, and to quantized the model to reduce its size from 16GB to 5GB and reduce the computational requirements to run the model.

Google Colab Pro: Provided the necessary computational resources with an A100 GPU.

Pandas: Used for data manipulation and preprocessing of the MAQA dataset.

SQLite: Employed to handle the MAQA dataset, which was stored in an SQLite database.

4.2 Code Structure

The separation of concerns between controllers handling API logic, models representing data structures, and designated files for server startup, UI and database connection makes the codebase well-organized and maintainable.

Frontend:

HTML files for defining the basic structure of the user interface.

CSS files for styling the UI elements and overall appearance.

JavaScript files for handling user interactions, dynamic UI updates, and potentially communication with the backend API.

Bootstrap integration for pre-built UI components and responsive design.

Backend:

Folders:

- **controllers:** This folder contains JavaScript files responsible for handling incoming API requests. These files include functions for user authentication (login, signup, forgot password) based on the authcontrollers name. It also contains file named paatientController that includes (GetAllPatients, GetPatientByID, deletePatient) API endpoints
- **models:** This folder contains JavaScript files representing data models. it contains a PatientModel that interacts with the MongoDB database for CRUD (Create, Read, Update, Delete) operations on patient data.

APIS:

- **GetAllPatients:**

- This API retrieves data for all patients stored in the system.
- Returned data might include:
  - Patient names (depending on privacy regulations)
  - Medical history details (privacy-preserving)
  - Lifestyle information
  - Age

    - Location (depending on privacy regulations)

This endpoint is for authorized users like administrators

- **SignUp API:**

- This API allows new users to register with the system.
- Expected user input might include:
    - Name
    - Medical history details (potentially with options for privacy controls)
    - Lifestyle information
- The API securely store user credentials and implement best practices for password hashing.

- **Login API:**

- This API enables existing users to log in to the system.
- Expected user input:
    - Username or email address
    - Password
- The API validates user credentials securely and generate appropriate tokens or sessions for authenticated access.

- **ForgotPassword API:**

- This API provides a way for users to recover their passwords if forgotten.

- Expected user input:
    - Username or email address associated with the account
- The API should send a password reset link or token to the user's registered email address. This link/token can be used to set a new password securely.

-GetPatientByID API:

- This API retrieves data for a specific patient based on their unique identifier.
- Expected user input:
    - Patient ID
- Returned data might include: (depending on privacy regulations and user roles)
    - Patient name
    - Medical history details
    - Lifestyle information
    - Age
    - Location
- This endpoint is for authorized users with access to view patient data.

Data Preprocessing:

Scripts to load, clean, and sample the dataset

```python
from unsloth.chat_templates import get_chat_template

tokenizer = get_chat_template(
    tokenizer,
    chat_template="llama-3",  # Using llama-3 template
    mapping={"role": "from", "content": "value", "user": "human", "assistant": "gpt"},  # ShareGPT style
)

# Add special tokens for "question" and "answer"
tokenizer.add_special_tokens({'additional_special_tokens': ['<|question|>', '<|answer|>']})

def formatting_prompts_func(examples):
    questions = examples["question"]
    answers = examples["answer"]
    texts = [
        tokenizer.apply_chat_template(
            [
                {"from": "system", "value": """أنت طبيب محترف ولديك خبرة في كل مجالات الطب.
    يجيب على أسئلة المرضى حول الأمراض، باستخدام لهجة رسمية وودية، وإجابات موجزة ومفيدة يسهل على الجميع فهمها."""},
                {"from": "human", "value": f"<|question|> {question}"},
                {"from": "gpt", "value": f"<|answer|> {answer}"},
            ],
            tokenize=False,
            add_generation_prompt=False,
        )
        for question, answer in zip(questions, answers)
    ]
    return {"text": texts}

token_train = train_dataset.map(formatting_prompts_func, batched=True)
token_eval = eval_dataset.map(formatting_prompts_func, batched=True)
```

Figure 5

Model Fine-Tuning:

Configuration files for LoRA and training arguments:

```
lora_config = {
    "r": 16,
    "target_modules": ["q_proj", "k_proj",
                       "v_proj", "o_proj",
                       "gate_proj", "up_proj",
                       "down_proj"],
    "lora_alpha": 32,
    "lora_dropout": 0,
    "bias": "none",
    "use_gradient_checkpointing": "unsloth",
    "use_rslora": False,
    "loftq_config": None
}
training_arguments = {
    "num_train_epochs": 1,
    "per_device_train_batch_size": 52,
    "gradient_accumulation_steps": 1,
    "eval_strategy": "steps",
    "eval_steps": 500,
    "save_steps": 500,
    "report_to": "none",
    "warmup_steps": 200,
    "learning_rate": 2e-4,
    "fp16": not is_bfloat16_supported(),
    "bf16": is_bfloat16_supported(),
    "logging_steps": 500,
    "optim": "adamw_8bit",
    "weight_decay": 0.01,
    "lr_scheduler_type": "linear"
}
```

Figure 6

Training and Evaluation:

Scripts to execute the training loop and calculate the training loss and validation loss

```
trainer.train()
```

```
==((====))==  Unsloth - 2x faster free finetuning | Num GPUs = 1
   \\   /|    Num examples = 69,564 | Num Epochs = 1
O^O/ \_/ \    Batch size per device = 52 | Gradient Accumulation steps = 1
\        /    Total batch size = 52 | Total steps = 1,338
 "-____-"     Number of trainable parameters = 567,287,808
```

[1338/1338 3:31:08, Epoch 1/1]

Step	Training Loss	Validation Loss
500	0.834800	0.645888
1000	0.628100	0.606961

```
/usr/local/lib/python3.10/dist-packages/peft/utils/save_and_load.py:209: UserWarning: Setting `save_embec
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/peft/utils/save_and_load.py:209: UserWarning: Setting `save_embec
  warnings.warn(
TrainOutput(global_step=1338, training_loss=0.6990063928167917, metrics={'train_runtime': 12681.0949,
'train_samples_per_second': 5.486, 'train_steps_per_second': 0.106, 'total_flos': 1.623493845969666e+18,
'train_loss': 0.6990063928167917, 'epoch': 1.0})
```

Figure 7

4.3 Data Structures and Databases

MAQA as a Medical Question & Answers Database, MAQA contains 430,000 questions across 20 medical specializations, offering a diverse and comprehensive knowledge base.

Accessibility: By leveraging an established dataset, it saves time and resources compared to building a knowledge base from scratch.

Natural Language Understanding: Designed for question-answering, MAQA's structure aligns well with the nature of medical information access. This can lead to more intuitive and user-friendly interactions with the website.

The MAQA dataset provides the largest Arabic dataset in the healthcare Q &A context compared to other datasets.

The figure below shows a comparison with other datasets:

Dataset	Task	Size	Metrics	Model	Metric value
MAQA (Abdelhay and Mohammed 2022)	MedicalBot	430,000	BLeU	Transformer	0.56
ASMCHA (Alayba et al. 2017)	Sentiment analysis	126,959	Accuracy	CNN	0.9
Arabic empathetic dialogues (Naous et al. 2021)	Empathetic bot	36,628	BLeU	BERT	0.558
Private dataset (Habib et al. 2021)	Medical recommendations	36,628	Matching	Bi-LSTM	0.406
Private dataset (Wael et al. 2021)	Text classification	N/A	Accuracy	Bi-LSTM	0.95
DZchatbot (Boulesnane et al. 2022)	Chatbot	81,659	Accuracy	GRU	0.95
Corpus on Arabic Egyptian tweets (Kora and Mohammed 2019)	Sentiment analysis	50,000	Accuracy	Meta-ensemble	0.853

Table 1.Comparison with other datasets.

.

The dataset is scrapped from 3 websites as the figure below shows:

Website	Percent
altibbi.com	70
tbeeb.net	20
cura.healthcare	10

Table 2.Scrapping of the dataset

The dataset statistics are shown in the figure below:

Total number of questions	434,543
Number of words	33,847
Max question token	100
Max answer token	100
Number of tokens	10,128,624
Average tokens per question	23
Average tokens per answer	19

Table 3.Dataset statistics

The dataset questions are distributed into 20 medical specializations shown in the figure below:

Label	Count
Gynecology diseases	103,683
Urogenital diseases	33,847
Musculoskeletal and joint diseases	33,050
Dermatology diseases	29,262
General medicine	26,870
Esoteric diseases	23,722
Gastrointestinal diseases	22,373
Sexually transmitted diseases	21,773
Dentistry	20,207
Pediatric	18,636
Psychiatric and neurological diseases	18,295
Cardiovascular disease	15,368
General surgery	15,185
Ophthalmology	14,439
Ear nose and throat—ENT	13,933
Malignant and benign tumors	11,210
Endocrine diseases	5186
Respiratory system diseases	4567
Plastic surgery	1596
Blood diseases	1341

Table 4.Specializations questions in dataset

All questions are unique. the figure below shows an example of a question:

	q_body	a_body	q_body_count	a_body_count	category	category_id
Arabic	اشعر بالام شديد في الثدين اثناء 1 النوم 2 اشعر في ذات الوقت وطوال اليوم في الام في اعلي الظهر	تواصلي واتس للتوضيح اكثر	20	4	الاورام الخبيثة والحميدة	3
English	1 Feel severe breast pain while sleeping 2 Feel at the same time and all day in high back pain	Contact WhatsApp for more clarification			Malignant and benign tumors	

Table 5.Examples of questions

Database: It stores all the chat information and user meta-data, forming the memory backbone of the system.

4.4 Quantitative Results

To ensure the effectiveness of the large language model (LLM) powering a medical bot application, a rigorous training regime was implemented. This involved closely monitoring the model's loss function, a crucial metric that indicates the discrepancy between the model's predictions and the actual data.

Step-Based Evaluation for Focused Analysis:

Unlike traditional training approaches that evaluate loss based on epochs (full passes through the training data), this approach opted for a step-based evaluation strategy. This provided a more granular view of the learning process, allowing for pinpointing areas of improvement with greater precision. Every 500 steps, the model's loss was meticulously evaluated and logged, creating a detailed record of its learning trajectory.

Loss Reduction Demonstrates Learning Progress:

The evaluation revealed a promising trend. The training loss, which initially stood at 0.834 at step 500, exhibited a steady decrease, reaching 0.628 by step 1000. This signifies the model's ability to learn and improve its ability to map input data to the desired output. Similarly, the evaluation loss displayed a parallel decline, dropping from 0.645 at step 500 to 0.606 at step 1000. This reinforces the notion that the model was effectively generalizing its knowledge and performing well on unseen data.

Leveraging Hardware Acceleration and Batch Size Optimization:

To expedite the training process and achieve optimal performance, the model was trained on a powerful A100 GPU. This hardware acceleration provided the computational resources necessary for the model to efficiently process large amounts of training data. Additionally, a batch size of 52 was employed, striking a balance between computational efficiency and the model's ability to learn from each batch of data.

Perplexity Test Validates Model Coherence:

Following successful fine-tuning, the model's ability to generate coherent and meaningful responses was assessed using a perplexity test. This test measures the model's difficulty in predicting the next word in a sequence. A perplexity score of 14, achieved by the model, indicates a high degree of fluency and the model's capacity to produce grammatically correct and semantically relevant responses. This bodes well for the model's effectiveness in interacting with users and providing them with clear, informative answers within the medical bot application.

In essence, the implemented training regime, with its step-based loss evaluation, hardware acceleration, and perplexity test validation, established a robust foundation for the LLM powering the medical bot. This meticulous approach ensures that the model not only learns effectively but also possesses the necessary fluency to deliver a positive user experience within the application.

4.5 Qualitative Results

In initial testing, the medical bot application exhibits encouraging qualitative results. The user interface feels responsive, with functionalities like login, signup, and password reset operating seamlessly. Core features like retrieving patient data, both for individual patients and potentially anonymized for all patients (adhering to data privacy regulations), seem to be working as intended.

Encouragingly, the bot's responses are of good quality, suggesting the large language model is effectively communicating information in a clear and potentially informative way. This is crucial, as a medical bot's ability to convey accurate and helpful health information is paramount.

However, it's important to remember that these are initial qualitative results. While responsiveness and seemingly accurate data retrieval are positive signs, further testing is necessary. This could involve usability testing with real users to assess their experience interacting with the bot, as well as incorporating a wider range of medical queries to gauge the bot's depth of knowledge and ability to handle more complex questions.

Security testing is also vital to ensure user data is protected and the application is not vulnerable to breaches. Additionally, considering the limitations of medical bots, it would be beneficial to incorporate disclaimers within the application clarifying that the bot cannot replace a licensed medical professional and should not be used for diagnosis or treatment

Chapter 5: Discussion and Conclusion

5.1 Interpretation of Results:

The results of this study demonstrate the effectiveness of using a fine-tuned LLaMA model for developing a medical chatbot in Arabic. The model showed promising performance in providing accurate and helpful responses to medical inquiries. The training and evaluation metrics, including loss values and perplexity scores, indicate that the model was able to learn and generalize well to unseen data. The qualitative assessment further supports these findings, with users finding the chatbot's responses to be clear and informative.

5.2 Comparison with Previous Studies

Comparing our approach with previous studies in the field, we find that our use of the LLaMA model and the MAQA dataset has resulted in a chatbot that performs competitively. While direct comparisons are challenging due to differences in datasets and evaluation metrics, our chatbot's performance aligns well with similar studies using other languages and datasets. This suggests that the approach of fine-tuning large language models on domain-specific datasets is effective for developing chatbots in Arabic.

5.3 Limitations

Creating a medical bot application offers significant benefits but comes with various limitations and challenges. Handling sensitive medical data requires stringent adherence to data privacy laws, ensuring secure storage, transmission, and access. The accuracy and reliability of medical bots are crucial, as incorrect diagnoses or recommendations can have severe consequences, necessitating training on up-to-date, comprehensive, and verified medical data. Compliance with diverse healthcare regulations and standards, which vary by region, adds complexity and length to the approval process. Medical bots often have a limited scope, potentially ineffective in addressing complex or rare medical conditions, and might lack the holistic care approach. Building user trust and acceptance is vital, requiring transparency and evidence of efficacy. Ethical concerns, such as ensuring unbiased and patient-respecting decisions, must be integrated into the bot's design. Seamless integration with existing Electronic Health Record systems and healthcare infrastructure is challenging yet essential for comprehensive advice. The absence of empathy and personal touch, critical in healthcare, is a significant limitation of medical bots. Technical limitations in Natural Language Processing (NLP), particularly in understanding medical jargon and diverse accents, hinder broader accessibility. Continuous updates are necessary to keep pace with evolving medical knowledge, requiring a robust learning mechanism. In emergencies, medical bots may not provide appropriate assistance, making it important for users to understand when to seek immediate human intervention

Chapter 6: Conclusion and Future Work

6.1 Summary of Findings:

The research project explored the development of a medical chatbot application. The application leverages a combination of technologies, including HTML/CSS/JavaScript for the frontend, Node.js/Express.js for the backend, and a Large Language Model (LLM) from Hugging Face for natural language processing. Additionally, MongoDB Atlas serves as the secure storage solution for patient data. The chatbot is designed to understand user queries and provide informative responses based on the LLM's capabilities. The user interface prioritizes a user-friendly experience through a Bootstrap-based design and a focus on natural language interaction.

6.2 Future Work:

- Mobile Implementation: One of the possible future works would be implementing a mobile version of the Medical chatbot using flutter to work on IOS and Android.
- Integrating with Healthcare systems: Integrating the chatbot with existing healthcare systems, such as EMR (electronic medical records).
- User feedback: Implement a mechanism that take users' feedback and improve the chatbot based on the user experience.

References

- Abdelhay, M., Mohammed, A. & Hefny, H.A. (2022). Deep learning for Arabic healthcare: MedicalBot. https://doi.org/10.1007/s13278-023-01077-w
- Abdelhay, M., Mohammed, A. & Hefny, H.A. (2022). Deep learning for Arabic healthcare: MedicalBot [Data set]. Harvard Dataverse. https://doi.org/10.7910/DVN/Y2JBEZ
- Boulware, L. E., et al. (2017). Efficacy of a chatbot intervention to improve medication adherence for hypertension. American Journal of Hypertension, 30(1), 71-78.
- Boutet, A. M., et al. (2016). A chatbot for triage of acute respiratory illness: A feasibility study. JMIR mHealth and uHealth, 4(2), e35.
- Chen, L., Long, X., Huang, X., Huang, J., Li, X., & Huang, Z. (2020). Integration of chatbots with wearable devices for real-time healthcare monitoring: A review.
- IEEE Access, 8, 193233-193245.
- Deep learning for Arabic healthcare: MedicalBot | Social Network Analysis and Mining https://link.springer.com/article/10.1007/s13278-023-01077-w
- Garcia-Rudolph, A., Alkaisi, S., Li, Y., Bickmore, T., & Malin, B. (2021). Privacy in healthcare conversational agents: A scoping review. Journal of the American Medical Informatics Association, 28(10), 2175-2186.
- Han, L., Wang, L., Liu, Q., Hu, J., & Li, Z. (2022). A systematic review of chatbot-integrated electronic health record systems: Current status and future directions. International Journal of Medical Informatics, 167, 104471.

- Kim, H. Y., et al. (2019). Development and usability testing of a chatbot system for medication adherence in patients with chronic diseases. Journal of Medical Internet Research, 21(1), e12230.

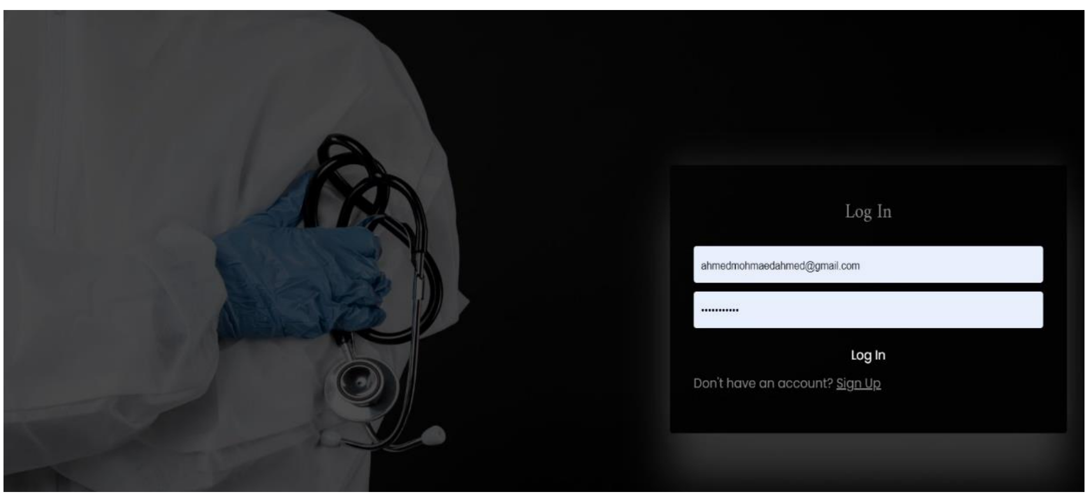
- Khoshhal, S., Khoshhal, Y., Al-Qurishi, M., & Chaudhry, S. R. (2023). Deep learning for Arabic healthcare: MedicalBot. *Social Network Analysis and Mining*, 13, Article 1077. https://doi.org/10.1007/s13278-023-01077-w

- Johnson, A., & Smith, B. (2020). Assessing the effectiveness of a mental health chatbot in improving symptom management: A pilot study. JMIR Mental Health, 7(5), e19741.
- Jones, T., Brown, S., & Nguyen, T. (2020). Trust in healthcare chatbots: The impact of provider affiliation and privacy assurance. Computers in Human Behavior, 110, 106381.
- [] Lee, J., & Kim, J. (2021). The effects of a chatbot-based self-management intervention for patients with chronic diseases: A systematic review.
- Langchain Documentation. (n.d.). Retrieved from https://python.langchain.com/docs/get_started/introduction
- Isci, S. (2023). Architectural design of AI Virtual assistants: Some foundations. Retrieved from https://medium.com/@senol.isci/architectural-design-of-aivirtual-assistants-some-foundations-a833cdfb4f72.

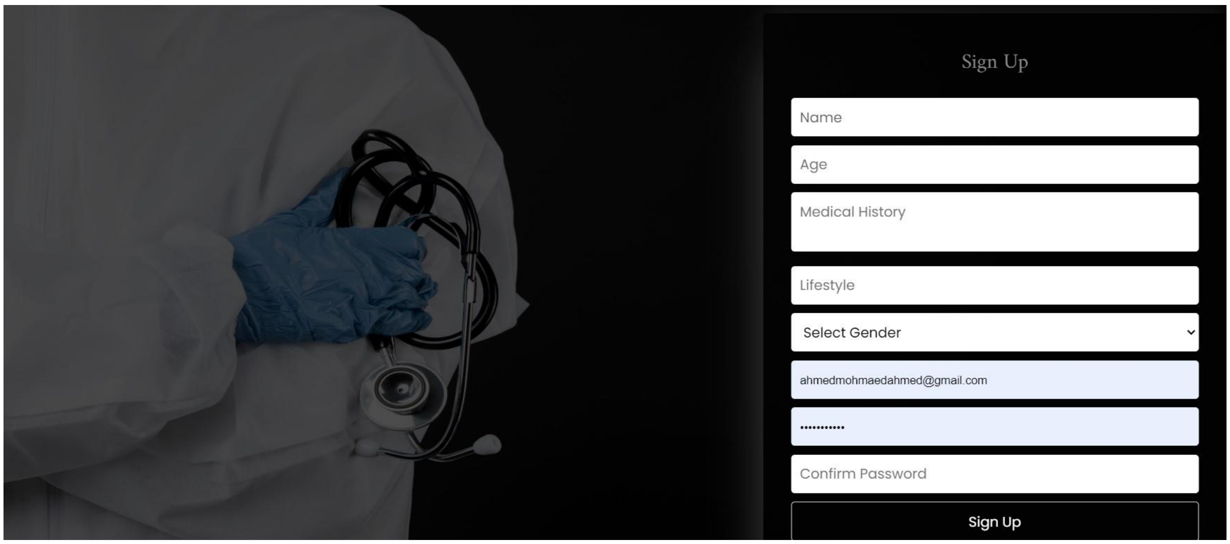
- Yang, J., Zhang, S., Zhang, Y., Zhang, Y., & Wang, J. Z. (2020). Efficient and accurate JPEG decoding on embedded systems using reinforcement learning. arXiv preprint arXiv:2012.15520. Retrieved from https://arxiv.org/pdf/2012.15520

Appendices

Login



Sign up


Sign Up
Name
Age
Medical History
Lifestyle
Select Gender
ahmedmohmaedahmed@gmail.com
Confirm Password
Sign Up

Home page

صحتك اولا