FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Computer-Assisted Anamnesis (medical history) - the patient UI

Pedro Alexandre Vieites Mendes

Mestrado em Engenharia Informática e Computação

Supervisor: Prof. Nuno F. Rodrigues

September 12, 2023

Computer-Assisted Anamnesis (medical history) - the patient UI

Pedro Alexandre Vieites Mendes

Mestrado em Engenharia Informática e Computação

September 12, 2023

Resumo

Há uma crescente pressão exercida sobre os departamentos de emergência e os profissionais de saúde devido ao número crescente de pacientes. Isto causa sobrelotação, o que, por sua vez, leva a problemas logísticos e operacionais, períodos de espera mais longos para pacientes, maior probabilidade de erro, etc., que afetam significativamente a qualidade da assistência médica como um todo.
O diagnóstico errôneo, acima de tudo, é responsável pela maioria dos erros que resultam em ferimentos graves ou morte nos departamentos de emergência. A anamnese, também conhecida como entrevista médica, é a forma mais eficiente de recolher dados de qualidade sobre os principais aspectos da condição do paciente, o que é crucial para se chegar a um diagnóstico preciso. Além disso, dados de qualidade formalmente estruturados e padronizados são necessários em quantidades crescentes para alimentar a criação e o desenvolvimento de novas tecnologias de saúde de inteligência artificial, o que é difícil de fornecer, considerando a quantidade cada vez menor de tempo que os médicos podem dedicar ao atendimento de cada paciente.
A solução proposta por meio desta dissertação baseia-se na anamnese computadorizada como meio para ajudar a aliviar a carga de trabalho enfrentada pelos médicos, diminuir a ocorrência geral de erros médicos e fornecer uma fonte de informação médica abundante e de alta qualidade. Mais especificamente, através da introdução de quiosques de autoatendimento na sala de espera do departamento de emergência permitindo que os pacientes utilizem o seu tempo de espera de forma mais produtiva, inserindo as suas próprias informações médicas relevantes no sistema para que possam ser apresentadas ao médico num formato de fácil apreensão.
Do trabalho desenvolvido para esta tese, resultou a implementação da plataforma discutida nesta dissertação que, como mencionado acima, questiona os pacientes sobre as questões relevantes para os médicos antes da consulta e que pode ser melhorada dinamicamente durante a operação para atender melhor às necessidades dos médicos. No entanto, uma implantação de teste da plataforma em um ambiente real de emergência hospitalar acabou por não ser possível dentro do prazo disponível, e portanto a plataforma permanece n ao avaliada até o momento.
Palavras-Chave: cuidado de saúde personalizado, algoritmos médicos, interação computador-humano, sistemas de informação

Abstract

There is an increasingly heavier burden placed on emergency departments and medical healthcare professionals due to the rising number of patients. This causes crowding, which in turn leads to logistical and operational issues, longer waiting periods for patients, higher chance of error and misdiagnosis, etc., which all significantly impact the quality of healthcare as a whole.

Misdiagnosis, most of all, accounts for the majority of errors resulting in serious injury or death in emergency departments. Anamnesis, also known as medical interview and history taking, is the most efficient way to collect quality data about key aspects of the patient's condition, which is crucial to reaching an accurate diagnosis. Additionally, formally structured and standardized quality data is required in growing amounts to fuel the creation and development of novel AI healthcare technologies, which is hard to supply considering the diminishing amount of time doctors can allot to each patient's consultation.

The solution proposed through this dissertation revolves around the use of computerized anamnesis to help alleviate the workload faced by doctors, lower overall occurrence of medical errors and provide a plentiful and high quality source of medical information. More specifically, by introducing self-service kiosks to the emergency department's waiting room and enabling patients to spend their waiting time more productively by entering their own relevant medical information into the system so that it can later be presented to their physician in an easily understood format.

From the work developed for this thesis, resulted an implementation of the platform discussed in this dissertation that, as mentioned above, queries patients on the relevant questions for physicians prior to consultation, and that can be dynamically improved during operation to better meet physicians' requirements. However, a trial deployment of the platform in an actual hospital ED environment ended up not being possible within the time frame available and so the platform remains as of yet unevaluated.

Keywords: personalized healthcare, medical algorithms, human computer interaction, information systems

Acknowledgements

Here I would like to extend my appreciation to all who helped in the development of this project. Namely, my thesis advisors Nuno Rodrigues and Antonio Coelho, Gonçalo Ferreira who worked on the thesis scope complementary to mine, Paulo and Jacinta who laid the predating groundwork for the project, my friends and family for supporting me all throughout, and INESC and FEUP for providing me with this opportunity.

Pedro Mendes

Contents

1 Introdução **1**
1.1 Context . . . 1
1.2 Motivation and Objectives . . . 2
1.3 Document Structure . . . 2

2 State of the art **3**
2.1 Literature Review . . . 3
2.2 Existing similar technologies . . . 4
2.3 Self-service kiosks . . . 6

3 Approach in Computer-Assisted Anamnesis **8**
3.1 Problem Definition . . . 8
3.2 Approach . . . 11
3.3 System Requirements . . . 16
3.3.1 Functional Requirements . . . 16
3.3.2 Quality Requirements . . . 16

4 Implementation **18**
4.1 Database . . . 18
4.1.1 Precursors . . . 18
4.1.2 Feedback . . . 19
4.1.3 Conceptual Data Model . . . 20
4.2 Back end . . . 22
4.3 Patient Application . . . 23
4.3.1 Application Interface . . . 24
4.3.2 Application Data Flow . . . 25
4.3.3 Human Body Model . . . 27
4.3.4 Rephrasing . . . 28

5 Final considerations **29**
5.1 Conclusions . . . 29
5.2 Future Work . . . 30

References **31**

6 System Requirements **35**
6.1 Functional Requirements . . . 35
6.2 Quality Requirements . . . 36
6.3 System Routes . . . 36

6.3.1 User Routes . . . 36
6.3.2 Patient Routes . . . 37
6.3.3 Phycisian Routes . . . 37

List of Figures

2.1 Triage advice result comparison 5

3.1 Age range distribution of physicians in Portugal (2022) 9
3.2 Mechanism for system growth 12
3.3 Mechanism for system growth 15

4.1 System relationships 20
4.2 Conceptual data model 22
4.3 Interface elements 24
4.4 Clickable model of the human body 28

List of Tables

2.1 Kiosk triage results 7

6.1 user routes 37
6.2 patient routes 37
6.3 physician routes 37

Abbreviations

CAA	Computer-Assisted Anamnesis
HCI	Human-Computer Interaction
ED	Emergency Department
CC	Chief Concern
AI	Artificial Intelligence

Chapter 1

Introdução

1.1 Context

Healthcare is a very critical part of society. All issues concerning it can be said to have branching impacts on all other aspects of it and, naturally, on the population's health. As such, it is a meaningful effort to ensure that the healthcare system is functioning as efficiently and optimally as possible across all sectors.

And yet, the number of patients that medical healthcare professionals must deal with each day is ever increasing, which presents an obstacle to our desired ideal of ensuring efficient and optimal healthcare. This is even more apparent in emergency departments, which tend to see the largest influx of patients and require the swiftest response time. However, the percentage of those patients who do not require urgent medical attention and could, thus, ideally be treated via either simple self-care or primary healthcare, is rather large [28]. In Portugal alone, according to a study from 2015, 37% of visits to the emergency department could be deemed as unnecessary [7], a fact which was only exacerbated by the COVID-19 pandemic.

All this contributes to the issue of crowding in Portuguese emergency departments, and in turn, to higher waiting periods for patients and less time dedicated by physicians to each patient [21]. This also induces a rushed mindset in physicians which sees them interrupting patients a little over twice per minute [20] and leaving over a third of all essential history items pertaining to diagnosis and treatment unaddressed [31]. A critical shortcoming, considering misdiagnosis accounts for the majority of medical errors in the emergency department [27], with many proving serious or fatal [12]. To many, anamnesis stands as the best method for gathering medical information and getting a good grasp on the patient's medical condition, above even physical examination or laboratory testing, but these conditions mean that anamnesis is often not conducted properly.

A computer, on the other hand, does not feel the emotional or physical pressures that might lead doctors to rush the process of taking a patient's medical history and would therefore have the potential to be able to conduct a more comprehensive retrieval of all the relevant information. The more complete this retrieval is, the better the understanding the physician will have of the patient's current medical condition and the likelier they will be to come up with the correct diagnosis, effectively contributing towards a downturn in the occurrence of errors and an upturn in the quality of data collected.

Furthermore, being the first step from which the rest of the healthcare process unfolds, anamnesis heavily influences the differential diagnosis and treatment outcome, and as a result, the data collected throughout this whole process - data which is vital for the development of novel AI healthcare technologies and medical tools, as well as help potentiate future medical research, the demand for which has only been growing over the last few years, with the advancement of technology and popularization of artificial intelligence-driven approaches.

Notwithstanding this growing demand, physicians are struggling to keep up, complaining of already spending more time typing than seeing to their patients and with many failing to input quality patient history data. The fast pace at which physicians take medical histories produces data of sub-optimal quality and the subsequent task of documenting all this information in the formally structured, standardized format that computers require can be quite tedious and time-consuming.

1.2 Motivation and Objectives

The motivations behind a computer-assisted approach to anamnesis would then be in dealing with all the aforementioned issues while helping both the patient and physician make better use of their time.

Hopefully, through this approach, patients will be able to make use of their idle time in the waiting room more productively, by leveraging it towards the retrieval of their medical history information prior to being seen by a doctor. This, in turn, will help doctors spend less time performing the anamnesis and writing the medical report, allowing them to allot more of their consultation time towards examining and advising the patient.

Overall, the aim is to streamline and improve the whole medical consultation process to alleviate the workload faced by medical healthcare providers and ease the flow of patients through the emergency department, effectively mitigating ED crowding, while also maintaining an output of good quality data to serve as the foundation for future innovation.

1.3 Document Structure

Beyond the introduction, this document contains 3 more chapters:

Chapter 2, where the background for the study's context and the state of the art are explored.

The review of the system requirements as compared to the previous project implementation and new approach discussion.

The implementation itself, annotated with the different considerations taken into account.

The last considerations, in the final chapter, where the conclusions are stated and the potential avenues of further development in future work discussed.

Chapter 2

State of the art

This chapter is dedicated to discussing the background work done in the scope of computerized medical history taking and the existing technologies of a similar nature, as well as the implications deployment of self-service kiosks in healthcare.

2.1 Literature Review

Due to the extensive and ever growing knowledge base for medical practice, the use of computers as a support tool in helping reduce the involved cognitive load to specialization has been developing itself since the late 40's. The Corner Medical Index [9] was one of the earliest such efforts in the form of a questionnaire to standardize medical data collection and perform it quickly and reliably without the need of the physician's presence. It was instrumental in later setting a precedent showing computers can be as accurate as physicians when dealing with common ailments and making use of self-collected standardized data [10], and proving anamnesis as an empirical, logical endeavor.

The approach was further developed by programming the computer who performed the medical history taking with logic using branching question trees [36], where a previous answer would lead to a different predetermined line of questioning. This allowed for the expansion of the question pool in comparison to the questionnaires, since only the more relevant questions would need to be answered. Another addition was the one pioneered by Lawrence Weed, who sought to use the quality data collected to aid in diagnosis decision making by comparing it to the recorded items commonly associated with the chief complaint. While outside the scope of this project, it further highlights the importance of the quality of the information gathered through anamnesis.

Through all these early innovations and implementations, it was found that data collected via computer-assisted medical history taking contained more useful clinical data than regular physician collected history data [11]. An expected outcome, considering the part the capacity limits in human cognition played in bringing the interest in the integration of computers in medical practice to the fore in the first place.

And yet, physicians questioned on the value of this approach chalked up the added information collected to being false negative computer outputs or irrelevant [35, 18], and still considered the histories collected by physicians to be more useful for diagnosing than computerized histories even when they did acknowledge them as more comprehensive [30, 26]. Analysis made on the reliability of the information collected via computerized histories being the same when retaken by physicians begs to differ however, showing test-retest replication rates of over 90%. Taking into account that there was also deemed to be more useful information among that collected by the computers when compared to that collected by doctors, an argument can be made on computerized histories being just as, if not more capable as doctors in this data collection capacity [11]. Doctors themselves, in an evaluation of medical histories taken at home prior to consult, also attributed an average score of 7.6 out of 10 to the usefulness of the information collected, further validating this argument [37].

On the patient side, it was found that even since early on when computer integration in society was not as blatant, the level of acceptance was high [36, 35, 29, 38], and has remained so until now [37, 40]. The amount of time patients were willing to invest into recording their own medical history going to upwards of an hour [35, 37], as it pertains to something that directly affects them and their future very vividly.

2.2 Existing similar technologies

In terms of the more recent technologies that could be likened to what has been mentioned so far, the symptom checker applications are one of the prime subjects. They serve as tools to aid in the self-assessment of one's medical condition and based on that provide advice, and their use has become more and more prevalent as of late. In great part, due to the COVID-19 pandemic which saw the development of several apps to help users decide on whether they might have contracted the disease and how they should proceed.

These apps promote the empowering of the user to take charge of their own health care and become self-responsible, independent actors more engaged in their healthcare process. They are also convenient, being able to reach a wide mass of people and much simpler than having to schedule a physical meeting with a physician. Especially in the cases of regions of poorer conditions where access to a physician might not be taken for granted.

They come with their own plethora of shortcomings. The overall outlook on these apps is rife with remarks questioning their accuracy, security, privacy and use of medical jargon [5, 15, 14] . The developers of these apps frequently sell the data to third parties and the need for their apps to include disclaimers stating they are for "entertainment purposes only" or that they do not substitute actual professional medical council hints towards the authority they actually hold in issuing diagnoses [17]. As such, many are of the opinion that these apps could be more useful for self-triage rather than self-diagnosis.

A recent study [33] evaluates just that, by comparing the triage accuracy results of symptom checker apps on case vignettes with a 5 year interval of difference between measurements. They mirrored the approach taken on a past article's analysis and compared those results with those of 2 recent data sets and those returned by a layperson's judgment. The following picture depicts this comparison:

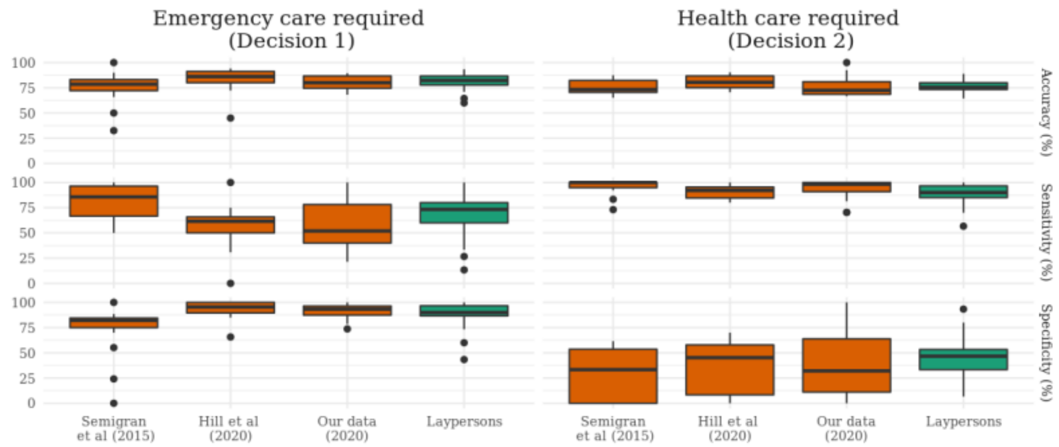

Figure 2.1: Triage advice result comparison

The categorical value representing the urgency of the patient's medical condition was one-hot encoded into two binary classifications that are reflected in the 2 decisions showcased. The overall triage accuracy has not improved in the past 5 years. At least, in respect to the same set of vignettes. At no point was the median accuracy significantly higher than that of a layperson's either. There were a few apps that individually displayed advice more reliable than the average layperson's regarding triage on one of the decisions, but none that did so for both.

Worse yet, risk aversion in triage advice provided actually decreased. This means the amount of false negatives, e.g. emergencies going undetected, and the single most important statistic for measuring a symptom checker's safety, actually increased. All of it combined puts into question the value these apps currently add to a layperson's decision-making - a question whose relevance becomes only more pronounced given that most laypeople's self triage assessment is not meaningfully influenced by internet search [16].

The few apps that can be useful, are usually so for specific questions or for appraising certain types of diseases like in the case of the all the different apps that popped up during the pandemic for symptom tracking, triage, etc [41]. And so, that leaves the users with the extra task of figuring out which app to use in order to get a reliable answer for each question they might have regarding their specific medical condition and circumstances they find themselves in.

Taking all of this into consideration, it strengthens the belief that diagnosis and actual decision-making are better left in the sphere of medicine and to healthcare professionals for now, with our focus better directed towards ensuring the information collected is as comprehensive and of good quality as possible. After all, it only makes sense that before one tries to reach a correct judgment, they first ensure they are asking the correct questions.

2.3 Self-service kiosks

Interactive kiosks are computer terminals featuring specialized hardware and software that provides access to information and applications for communication, commerce, entertainment, or education. Self service kiosks fall in the category of having users perform a task or service that would otherwise require human interaction. One such example are the self-service kiosks installed in fast food chain restaurants, allowing the customer to make their own order.
Deployment of these kiosks into medical environments have been quite successful, with them now being nigh ubiquitous in hospitals and clinics. Most of these serve simply as logistics facilitators, but there have also been cases of these self-service kiosks being used in the collection of patient personal information, triage support and even in the measuring of vital signs. Furthermore, they were found to have been helpful in reducing waiting periods for patients and increasing patient satisfaction, which has been linked to higher quality of care.
The added efficiency they bring to whatever task they embody is something interactive kiosks are known for. This is reflected in several success examples found during literary review displayed in the table below:

All this goes to show that, so long as the quality of user experience is maximized via an intuitive and easy-to-use interface, the benefits are clear. From that point on, it is only a matter of just how much value can be extracted from data collected through this method.

It is also worth noting that all of these implementations made use of static medical algorithms in their questioning components for the most part. If static questioning methods already boast these many benefits, then an even greater potential can be said to lie in a dynamic questioning system with the ability to grow. In [13] in particular, we can see how the growth experienced across 3 different versions of the system affects its overall consistency with nurse triage data further substantiating this notion. From the first test to the third, accuracy rose from 69% to 79%, sensitivity from 66% to 80% and specificity from 69% to 78%, as a result of repetitive implementation of feedback into the system following each test.

Table 2.1: Kiosk triage results

Paper	Context	Demographics	Study dimensions(n)	Notable Results
Sinha, M. et al (2014) [34]	Comparison between Audio-assisted bilingual self-triage kiosk for entering of medical history and nurse-initiated triage in Pediatric ED for inner-city tertiary care hospital.	Predominantly low-socioeconomic, low-literacy, Latino patient population. Patient age range of [0-18] years. Kiosk user distribution: 88.5% parent, 9.5% patient (aged 11-17) and 2% via proxy(sibling or friend). 50.5/49.5 split between english and spanish users	200 instances per group	Mean time spent entering medical history data was considerably lower for the kiosk group than for the nurse group (94.38[38.61] and 126.72[62.61], respectively). Significantly less inaccuracies detected for the kiosk group in regards to medical, medication and immunization histories, and total discrepancy score. User feedback above 90% for the top rating on all usability aspects considered.
Mahmood, A. et al (2020) [19]	Analysis of data collected via National Hospital Ambulatory Medical Care Survey results of 2015-2016 on for various EDs	Age distribution over 50% for ages 25 to 64. 45/55 male/female split. 59.6% non-hispanic white + 21.6% non-hispanic black + 2.9% other non-hispanic population	40 528 visits	Wait times were reportedly 56.8% shorted in EDs with self-check in kiosks as was patient satisfaction increased by 3% to 77% and triage process times lessened for nurses.
Boltin, N. et al (2018) [8]	Kiosk system used to collect data on a simulated chemical mass casualty incident along with a decision support system used to suggest triage results based on said data supervised and evaluated by trained nurses.	95% female 90% aged between 18 and 24 population. 1 male and 12 female nurses aged 30 to 69 with one hour of prior training used for triage.	500 volunteer nurses, staff and first responders recruited for the study. 286 interactions by 296 participants considered. 15 tablets.	97.3%(288/296) of patients were able to successfully use the system either on their own or with an assistance. Enhanced efficiency in 'triaging' patients. Nurse agreement with the decision-support system fed by the self-service kiosk gathered data. 91.6%(268/286) for exposure level classification and 84.3%(241/286) for corresponding action chosen.
Eijik, E.S.V. et al (2015) [13]	Successive computer-assisted self-triage instrument employed in ED of Eye Hospital Rotterdam in the Netherlands.	Average 33.4/66.6 female/male split across tests and age range of 53 years from ages 19 to 89.	144 patients across 3 tests.	Last distribution of the touch-operated application group boasted accuracy, sensitivity and specificity of near 80% in comparison to the triage by triage assistant group.

Chapter 3

Approach in Computer-Assisted Anamnesis

This chapter will once again touch on the thesis' context in order to better explicate the problem definition as well as discuss the reasoning behind the approach taken and its system's requirements.

3.1 Problem Definition

Reiterating, the overcrowding of the ED is a worldwide issue that can impact the hospital in a multitude of ways, such as:

- Over-depletion of resources
- Longer waiting times for patients
- Greater staff allocation to the ED
- Greater chances of error and misdiagnosis

More people coming to the ED implies a greater amount of resources being expended, considering that each person will be using up some resources. Resources, after all, need not only be used to refer to material things. Nurse and overall staff attention and time is expended at the very least in admitting a new patient to the ED and having resources be stretched thin will inevitably correlate to a poorer effectiveness of those resources' use.
Patients waiting longer in the ED will only further contribute to the ED's congestion and in doing so create a vicious cycle. Staff allocation to the ED increasing in order to alleviate the depletion of staff resources implies that the newly allocated staff had to be reallocated from some other section of the hospital. This in turn means that those other sections might be left understaffed because they were deemed to be of a lesser priority in comparison to the ED.
Increased chances of error and misdiagnosis lead to greater rates of morbidity and mortality

[27, 12], a reflection of the worsening quality of the healthcare provided as a result of ED overcrowding. And again, this is also a very relevant issue here in Portugal.
Despite being lauded as one of the countries with one of the higher rates of physicians to inhabitants - currently 5.7 physicians per 1000 inhabitants [1] - that is merely one aspect of the situation. Portugal also possesses one of the world's most aged populations with nearly a quarter over the age of 64 [2], and this is naturally also reflected on the country's physician population as you can see in Figure 3.1 [4], with the greatest section of the graph being occupied by those in that very same age range.

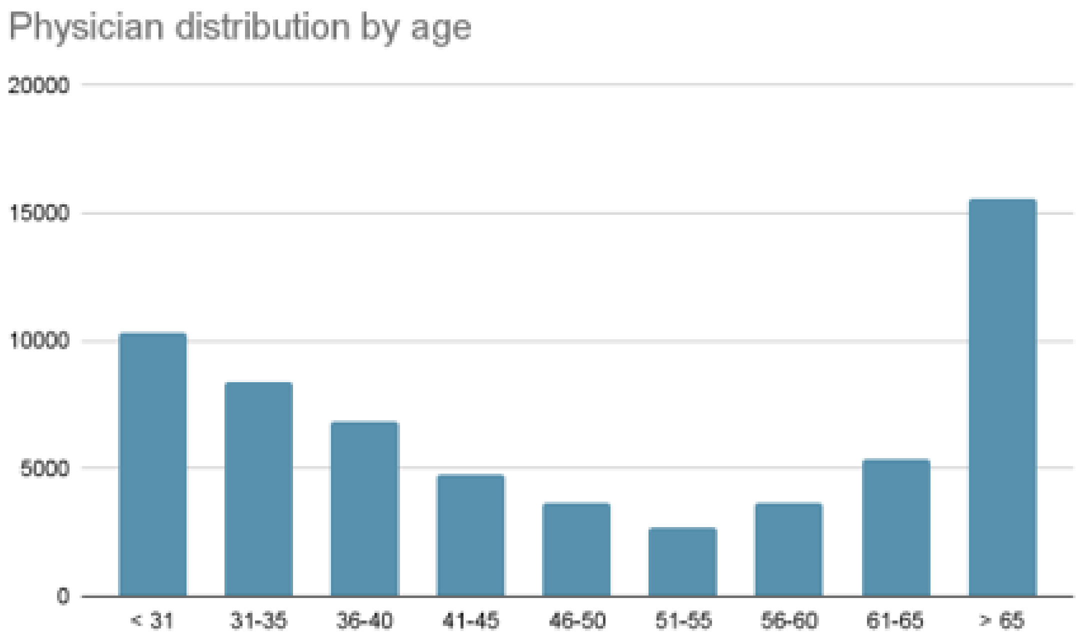

Figure 3.1: Age range distribution of physicians in Portugal (2022)

Even disregarding the fact that those physicians are getting ever closer to the time in which they will eventually retire, this means a significant chunk of the physician population is at or over the age of 50. Considering the ages of 50 and 55 are, respectively, the ages where by law physicians are no longer required to work in the ED at night, or at all, this means that that very same chunk is already no longer active, or at least as active, in the ED and this shortage of physicians manning the ED naturally has an impact on its overcrowding.
Another aspect is that of the increasing number of those without an attributed family physician which has been steeply growing along the past decade. From 2019 alone to now, the number has risen from around 680 thousand to over 1.7 million [22, 3] and this is relevant because this leads to a greater amount of people instead resorting to going to the ED and further worsening the crowding.
According to the current minister of health, this is also an issue that is not expected to be fixed for

at least the next 2 to 3 years [6] and so ED overcrowding can be in turn expected to be an even more prevalent issue for the near foreseeable future.

Given that the shortage of physicians is expected to not be resolved soon, another solution is needed in the interim in order to help keep things in check and a different perspective to directly solving the shortage is to instead help increase what the physicians we do have available can do.

With this intent, looking into what most of the consultation time is taken up by, that would likely be the initial phase where the physician asks the patient questions ***. If this time could be shortened, then it would reduce the average amount of time a physician takes to conduct a consultation and would allow them to see more patients than before in the same time frame.

But they can't also just rush through this process. Again, physicians already often don't ask all the necessary questions [31] and nearly half of severe diagnostic errors results from this very failure to gather and take into consideration certain important data about the patient [27]. So ideally, what is needed is something that would allow the physician to save time in asking questions while still ensuring that all the necessary questions are asked.

This is where the overarching project my thesis was the developed under comes in. Osler is a project being developed at INESCTEC whose main goal is to create a platform that will help combat ED overcrowding by increasing the efficiency with which they conduct their consultations.

When it was first proposed to me, the project was described as consisting of 3 different working parts pertaining to:

1. Computer-Assisted Anamnesis (CAA), the part of the platform which interacts with the patients

2. Data visualization, the part of the platform which interacts with the physicians

3. Trial deployment of the platform at a local hospital

The first regarding CAA was to be the main focus of my thesis. Anamnesis is essentially a term used to refer to that initial question asking phase of a consultation I mentioned previously and the computer-assistance component would come from having a self-service kiosk type of setup in the ED waiting room asking the patients the necessary questions in the physician's stead.

This would help physicians mainly by having the answers provided to them in an intuitive, easily-digestible format prior to consultation that would make it easier for them to more quickly get a grasp of the patient's condition and ultimately helping them save time. Additionally, physicians can also spend a considerable amount of time typing up the reports that would be submitted to their medical unit's EHR(Electronic Health Record) *** and so another expected boon would be using the data collected to create a report template for the physician in order to also save them time in that regard. This visualization of the patient's anamnesis data and related features is what the second working part of the project was set to focus on and was developed in collaboration with my part, but directly implemented by someone else also associated with the project.

Finally, for the third, the project is being developed in collaboration with Porto's Hospital de S.João and so it was expected for there to be a test run of the platform there once a more finalized

version was available.
The specific aim for this thesis was then to explore and find an optimal way to develop the application that would ask the patients the necessary questions in the physicians' stead.
A prior implementation of the platform [24, 25, 32] had also already been developed, but it was a static one in that the set of questions made was always the same and so one of the main goals for the next version was for it to be made dynamic, open to change, so that it could be improved.

3.2 Approach

In order to make the system dynamic and adaptable to the patient's individual case by posing questions in relation to their previous answers, the application of artificial intelligence (AI) and machine learning (ML) first came to mind as a potential solution.
The methods employed by the symptom checker apps I had looked into so far seemed to employ natural language processing (NLP) to understand a patient's free-text symptom description so I surmised that would be the most optimal direction to explore deeper in.
Out of the this exploration, the one approach that interested me the most was actually one that I came across by coincidence in one of the faculty's lectures known as **Semantic Folding** [39].
The way it works is that it segments a *Language Definition Corpus* (LDC) - essentially a set of text documents on the topic of interest - into several snippets representing a single context and then distributes those snippets across a two-dimensional matrix where proximity correlates to context similarity. It then takes every word in the LDC and sets the grid cells where that word appears in to active forming an overall pattern for that word that is referred to as the semantic fingerprint of the word.

These fingerprints can help shed light on the word's different meanings via the different clusters of active cells within them and their overlap with other fingerprints can be used to determine the context similarity between. This works even for sets of multiple words enabling this semantic comparison to be done even between sentences. Semantic folding also lends itself especially well to texts of formal, structured types of data like law code so it would similarly be expected to be useful when applied on medical data.
The foremost way I saw this technique being able to be used for the case of CAA was that of finding new follow-up questions for the patient based on the snippets that scored the highest in semantic fingerprint comparison with the key concepts of a subsets of the questions posed to the patient until that point according to their answers.
Semantic folding comes, however, with the caveat of not having any way to evaluate the sentences in terms of the word order - simply classifying them based on the commutative list of words existing in a given sentence - and so another perspective I considered was that of adding a third dimension to the grid. By having each layer correspond to the given word's order in the snippet, this would hopefully help to provide each semantic fingerprint with a sense of causality and to

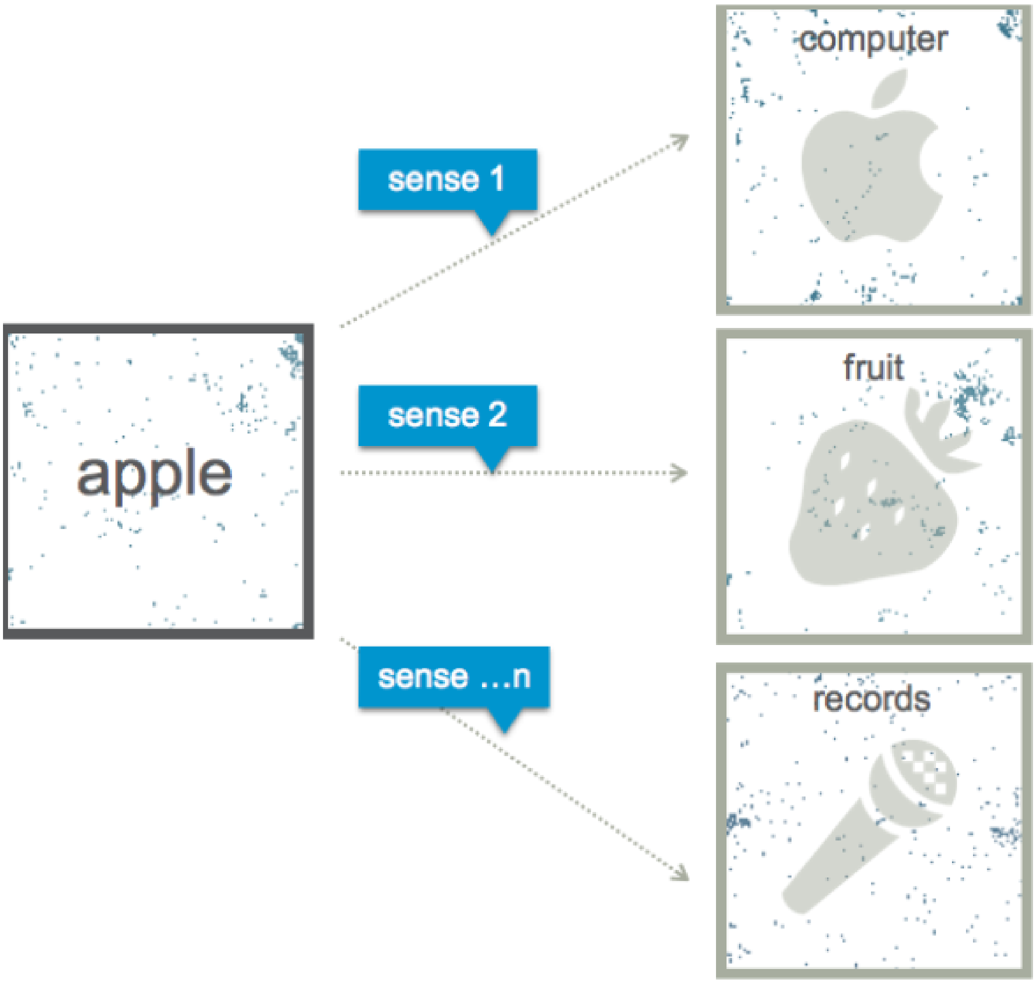

Figure 3.2: Mechanism for system growth

improve the accuracy of the follow-up questions deduced from this method being a natural consequence of the answers to the previous ones.
Finally, the semantic fingerprints resultant of semantic folding are similar independently of the language of the LDC they stemmed from which means there would be potential for a greater flexibility in the acquisition of LDC in regards to the language it was in as well as for future expansions of the platform to other languages.

That said, from what the aforementioned research I had done into the symptom checker apps, they were, even despite all the recent breakthroughs in AI and machine learning, not yet at the stage where they can be said to be as capable as actual medical professionals, with their conclusions always being considered, at best, secondary to those professionals' opinions, and at worse, merely anecdotal. Additionally, artificial intelligence solutions are heavily reliant upon the data provided to them, regardless of how splendid a model they may employ. Using data external to the project from sources like public medical databases did not seem very sustainable either as it would be a finite amount of data we held no control over when it would get updated. Being a system tied to the physical well-being of all the patients using it, even if indirectly, it would be put under increasing scrutiny as it grew. In order to keep improving the system, we would be forced to keep finding new sources of medical data of sufficient quality to meet the system's growing standards. This would then pose the challenge of homogenizing the information of different sources into a digestible format for the system while retaining enough distinction for useful insight to still be able to be gleaned from it. A challenge only further exacerbated when taking into account that most medical databases only hold data pertaining to a particular subtext of medicine, therefore increasing the amount of required sources.

Ultimately, even under the assumption the data gathered was optimal there will always be a certain degree of uncertainty at play when it comes to results obtained through the deployment of machine learning methods, and as such, along with the suggestion from medical professionals associated with the project that it would be better to go with something with more reliable end results to begin with, I decided to reorganize my approach. It would focus on something that was open to change, unlike its predecessor, and set itself up for having AI techniques employed atop it as support, but that had its main avenues of improvement come from the most reliable source there is for this context: physicians themselves.

To that end, I first considered a simple administration panel where an administrator could submit any needed changes to the system, but this would not be an ideal approach as having the growth mechanism for the system be centralized would mean that there would have to be one or a team of these administrators submitting all the changes themselves which raised several issues:

- Physicians as administrators
- Platform scalability
- Channels of communication

These administrators would most likely have to be physicians themselves, or at least people with some sort of medical knowledge, otherwise they would simply not be qualified for a role where they have to decide what changes make sense or not for a system operating in a medical setting. However, as we have seen, physicians' time is precious and with their current shortage, hospitals do not have the physicians to spare to specifically assign to doing this sort of role for any meaningful amount of time.

Furthermore, as the application grew and more physicians subscribed to the platform, the number of people with this role would also have to grow in order to meet the growing needs of the system and to be able to evaluate all the requested changes the rest of physicians would want. So besides leaving the decision-making for what changes should be made to a specific subset of physicians that may lead to a certain degree of bias, this approach simply does not look to be very sustainable in the long term.

Lastly, setting up the channels of communication for this administrator team and the rest of physicians would be a challenge in an of itself and would most likely be something needed as simply resorting to email and so on would be conducive towards change suggestions getting lost among the rest.

As such, I instead opted for a more decentralized approach where each doctors can directly submit the suggestions they wish to see reflected in the system themselves. Following a consult with a patient, by allowing each physicians to submit change suggestions to anything that they felt was lacking or that they would have liked to see in the interaction data provided by the system, the system would consider all physicians' suggestions.
The quality of the data submitted to the system would be automatically guaranteed as it would be provided by authorities on the field, but to further ensure the quality of the feedback provided though, the system would allow for it to be exposed to scrutiny by the physician's peers before it was accepted. Each suggestion would possess a qualifying score to determine whether or not it would be accepted and could be shown to other physicians when the interaction data they just finished was similar to the one that first prompted the suggestion or just before those physicians submitted a similar suggestion, so that if they were equivalent they could validate it instead of redundantly creating another one. Every consenting opinion would increase a suggestion's score and, conversely, every dissenting opinion would decrease it. Once a suggestion reached a target benchmark value, it would be applied by the system.
With just a fraction of the time being spent by each physician compared to the time an administrator would have to spend, the system would consider all physicians' opinions for improvement while also ensuring that the suggestions applied were of good quality as deemed by the physician user base as a whole.

A simple approach, but all the more effective for it: The backend fetches the question data from the database for the patient-side application where it is shown to the patient. Once the interaction is over, the answers are recorded in the database and the physician-side application will then fetch them for the physician. Again, while the physician side of the platform was developed in collaboration with mine, and its inner-workings are somewhat reflected in the backend and data structure, its implementation was outside of the scope for this thesis, and so for the purpose of this document it is only considered as a black box of sorts in that only its inputs and corresponding outputs are broached.

The only potential point of failure for this approach would be the physicians' unwillingness to provide feedback and discussing this possibility with physicians, the main reason for doing so would again be the time they would otherwise have to expend.
However, taking into account that the platform itself would already save them time by querying the patient on most questions that they would otherwise have to ask themselves, and presenting it to them in a concise, quickly consumable format, all in all it would lead to a net positive gain that rebuts this concern.

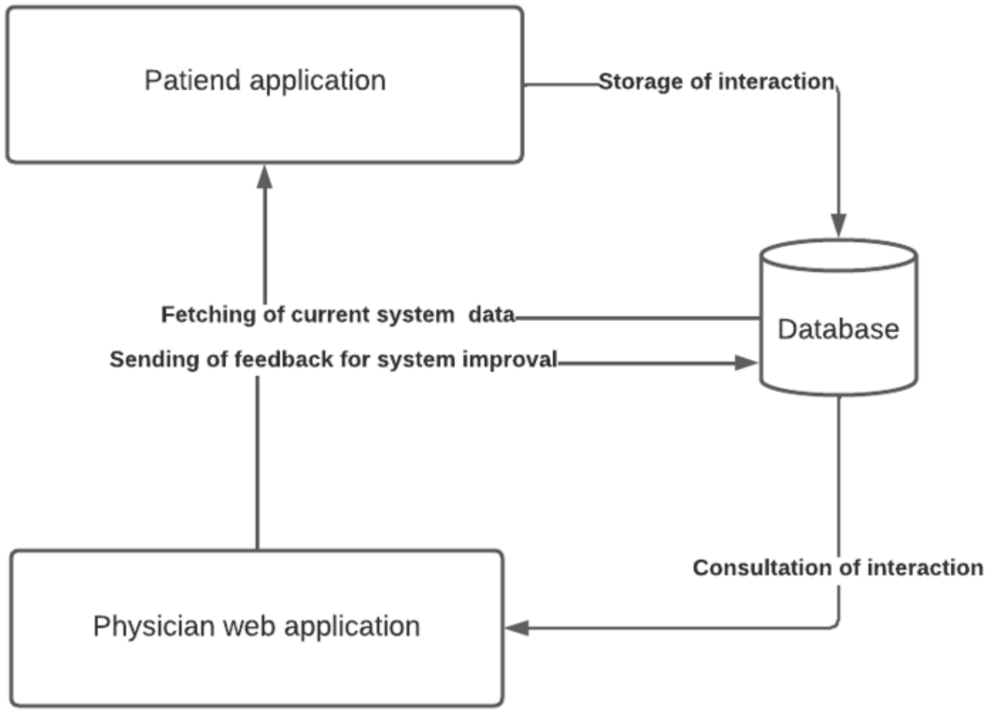

Figure 3.3: Mechanism for system growth

On top of that, considering the increasing amount of consults each physician goes through on an average work day, even just one suggestion being provided every other consult would constitute a significant change and improvement over time, but going even further to using a relatively low benchmark score or even making suggestions be immediately applied by the system could be a potential resort for motivating physicians to submit feedback in the platform's beginning stages by further quickening their return on the time invested. Once enough rapport had been built and the platform gained more users, the qualifying scores could then be gradually raised to meet the platform's similarly growing standards of quality.

Finally, with physicians being able to see, judge and evaluate each other's suggestions, this platform model would have the potential to foment a sense of attachment from the physician. After all, the set of suggestions provided by a physician would be tantamount to an amalgamation of their opinions, and seeing the platform as an extension of themselves would help serve as a powerful motivator towards getting them to contribute to its improvement. Ideally, this sort of approach would help lay the groundwork for a medical social network of sorts that focused on improving the anamnesis process and consultation efficiency altogether while creating a positive feedback loop for the system's growth.

The interaction data accrued in such a way could then serve as further fuel to ML algorithms such as semantic folding to further optimize the platform's feedback mechanism by creating suggestions in the physicians' stead based on the interaction data for example. This would serve as a source of inspiration for physicians when they did not immediately see anything that could have been improved about an interaction and potentially help them save even more time in fully typing up the suggestions themselves.

3.3 System Requirements

This section will discuss the requirements for the system developed under this project. Given that this project has had a prior implementation, the requirements for the current one were based upon a revision of the ones for that previous version. Note that this section is meant solely for the discussion behind them, and so the requirements' actual full explicit declaration is available in the appendix section of this document further below.

3.3.1 Functional Requirements

The original functional requirements will remain mostly the same, with the exception of the gathering of bio-metric data. Despite serving as a powerful indicators, the practical trial showed there were several issues with getting the patients to properly take their own measurements even with staff help. As such, this feature was left to be re-implemented in the future when new, better hardware was available for it.

In regards to newer functional requirements, the system will need to check if there are any "follow-up" questions that must to be posed to the patient as they provide answers and fetch them if so, in order for the system to be able to provide an interaction more tailored to the patient's individual case.

The system should collect at least the basic generally comprehensive data a physician themselves should when performing the anamnesis process.

The system should be open to change so as to be able to grow and improve, to deepen their interaction with the patients and to provide their physicians with a greater ratio of information that is relevant to the patient in question's condition.

One of the major points of criticism on the part of physicians was that there was too much "clutter" provided among the information that was irrelevant towards reaching a diagnosis.

3.3.2 Quality Requirements

The original quality/non-functional requirements will be upheld. In addition to them:

A question must always provide the option to let the patient say they did not understand it. This will not only prevent the patient from being railroaded into giving an incorrect answer, but also help the system find out where it is lacking in clarity so that these faults may be amended. Following that logic, the system should allow the patient to give a reason as to why they could not understand the question. This will make it easier to make the question more clear.

The time it takes for the system to fetch any "follow-up" questions should not be too long so as not to negatively impact the average overall time it takes to perform an interaction, and consequently diminish the return on time provided by the use of the system instead of a normal anamnesis process conducted by a physician.

The system should be flexible enough to work on as most hardware as possible. The hardware required to emulate a self-service kiosk does not need to match up the its stereotypical look as often seen in malls, fast food chains and the like, but it does need a large tactile screen in order to function as one. This sort of hardware can naturally be a scarcity in hospitals, if they even possess them at all, and therefore it is imperative that the system is flexible enough to run as much hardware variety as possible. An easier time in acquiring the hardware necessary for the operation of the system will result in a greater adoption rate by hospitals and the like.

Hospitals emergency units are open at all times, since there will naturally always be a necessity for them no matter the hour or time of year. The same needs to be reflected in the system's availability in order to mitigate the chances of it going down amidst operation.

Chapter 4

Implementation

This chapter will explain the process undergone for the implementation of the system, explaining the reasoning behind the technologies selected as well as some of the technical decisions made.

4.1 Database

The data format would end up shaping the flow of the patient application so the implementation naturally here. In regards to the database used, I decided to keep it a relational database.
Firstly, because it would make it easier to adapt the previous implementation's model, which was also relational.
Secondly, because as mentioned above, availability is a very relevant requirement. One of the advantages of relational type databases are its ACID properties: atomicity, consistency, isolation and durability. In short, this means that any operation is a single step process in that it either occurs in full or does not at all. This ensures the data stored does not fall into an in-between state and is always in consistent, therefore helping to ensure the platform does not crash during operation. Again, this is vital because ED are expected to remain operational at any and all times.
Likewise, the database management system used previously, MySQL, would still more than suffice for the updated intended purpose of the project.

4.1.1 Precursors

So to go about this approach, first arose the concern of representation. The way the data would be represented would in turn shape the way it is accessed, traversed and seen as a whole. An inefficient representation would be conducive to an inefficient, or even flawed, flow and interpretation of data.
The data would need to be restructured so as to be opened to change, especially for addition, as that is what would provide depth to the system. The foremost question regarding this is then on how to abstract and represent the concept of causality between questions. With the context being that of a conversation akin to an interview, by causality between questions I'm referring to to the

natural succession of one or more questions following the answers provided by previous ones. The relation between the aforementioned "follow up" questions and the respective previous question answers that prompted them.
Dubbed as *Question Precursors*, they serve as the glue connecting all the various questions into a coherent flow. If one were to see the questions and their respective answers as nodes in a graph, then the precursors are the edges connecting them to one another. In my opinion, this would be the most accurate representation that meets the new requirements.
A tree-based representation like the one in the previous implementation of the project does not work because that would mean that once a line of questioning branched out, it could no longer interfere or interact with others. However, an actual conversation is not something that constrained and a conclusion is a result of deliberation based upon several premises which could arise from just about anywhere within the conversation. This in turn means a directed graph would be more suitable to properly capture the crisscrossing relations of causality behind the questions posed in an interview-like context.
Figure 4.1 helps in visualising this relationship.

4.1.2 Feedback

Next came the question of what types of suggestions the system should support. As a base, the system contains these three:

- Addition of a new question
- Rephrasing of a question
- Removal of a question

Adding a new question would be the main propeller for the system's growth, introducing new follow-up questions to expand and deepen the lines of questioning utilized.
A physician can suggest a new question to be added by, selecting the precursors for this new question on the physician-side and providing the original phrasing for the question and the acceptable set of answers for it.
Rephrasing a question would serve as the supporting feature, and as the new usage quality requirements dictate, it would help to ensure that questions were clear and patients understood what they were being asked. However, instead of overriding the original phrasing, a rephrasing will instead be an additional phrasing that can be presented to the patient in accordance with the reason as to why they were unable to properly understand the question.
As for removal, it would serve as more of a refining tool, in order to ensure an outdated line of questioning could be deprecated. I say deprecated rather than deleting because deletion of a question from the database entirely would compromise previous patient-system interaction records. These records are relevant as they constitute important well-structured medical information that

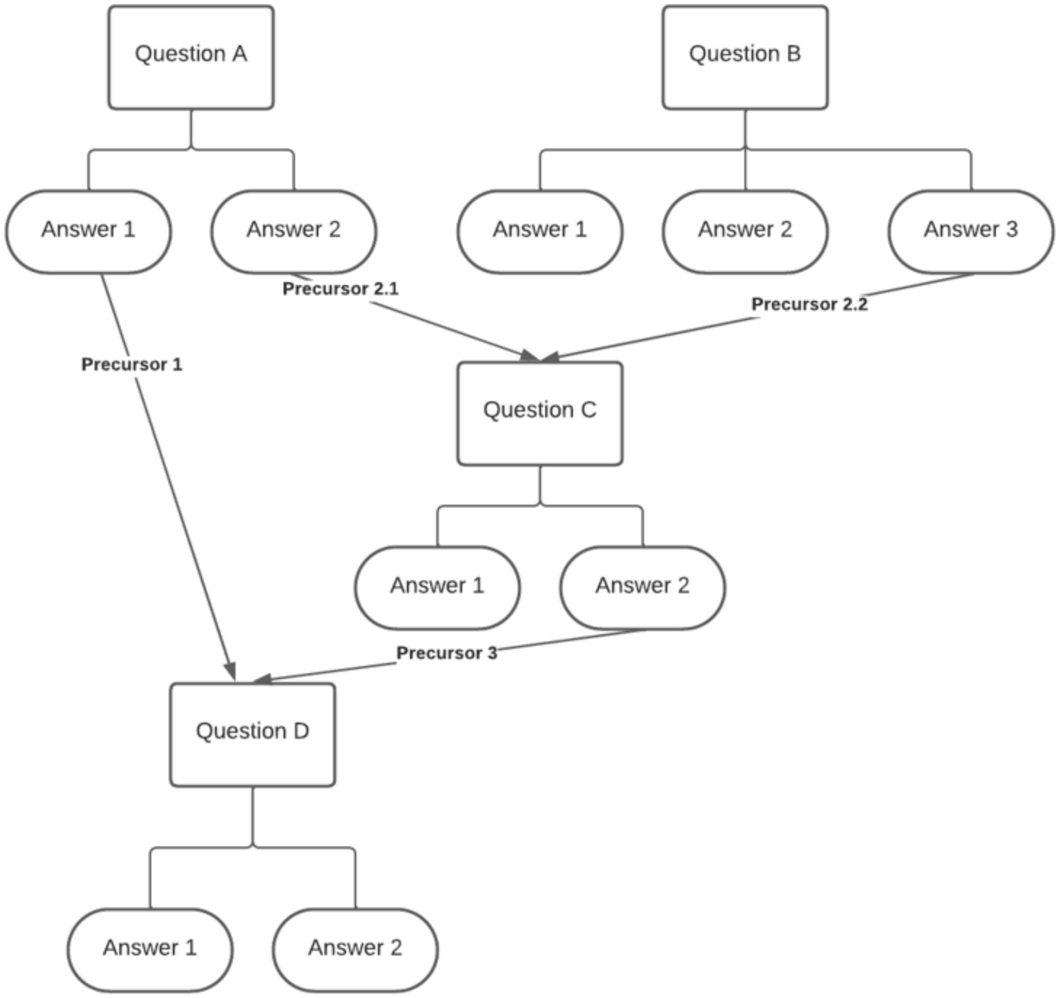

Figure 4.1: System relationships

should be available for later perusal and therefore merely be marked as no longer in use by the system. Soft deletion was useful in achieving this as a widely used pattern in business applications that allows for marking of records as deleted without them actually having been deleted from the database.
Finally, physicians can also provide a reason for justifying submitting their suggestion so as to improve their chances of it being accepted.

4.1.3 Conceptual Data Model

A relational database makes use of tables to store its data. Each table is composed of rows representing instances of said table known as records, and each has a field corresponding to each column of the table according to that column's accepted type and value as stipulated in the table's definition.

A relational database works by linking tables together through the concept of keys. Each record has a unique identifier column field within that table's pool of records that allows it to distinguish itself from the rest known as a primary key. Through this primary key comes the concept of foreign keys, a column field in a table that can hold the value for another table's record's primary key. This connection allows for the establishment of relationships between tables.
As mentioned in section 4.1.1, successor or follow-up questions must be able to arise from the answer provided by the patient in response to any given previous question posed to them. As such both the precursors and feedback models must be able to relate to all other data models that could be seen as questions. The previous conceptual model did not lend itself well to this as the different models used to store the data collected on the patient from the interaction were not unified under a common model. Keeping the models as they were would require the use of several foreign keys in order to establish the relationships. In the future it is possible for new types of questions or information to be collected (e.g. the bio-metric data not being collected in the new system) and so this would prove to be an unsustainable practice in the long term. One alternative considered was the use of polymorphic associations where the *foreign key* column will contain an id value that must exist in one of a set of target tables rather than just a single table. This alternative, however, cannot be modelled using SQL constraints and would simply be a stopgap method of treating the symptoms and not the underlying cause of the problem. In order to tackle the problem at its root, all models that begat an answer from the patient during the interaction were brought under the encompassing *Question* model as sub-classes of it so that all that question related foreign keys would only need to reference the *Questions* table. The final conceptual data model is as defined in the figure 4.2.

The Question Subclass table is a generalization made for more compact viewing, but its instances are:

- ChiefComplaint
- BodyArea
- Descriptive
- Aggravation Factor
- Attenuation Factor
- MedicalHistory
- Medication

An explanation on what each of them pertains to is available in section 4.3.2. The prompt field refers to the prompt shared by all questions of that subclass that frames them as questions. It allows for the questions to be displayed as answers to the question described in the prompt when presented to the patient.

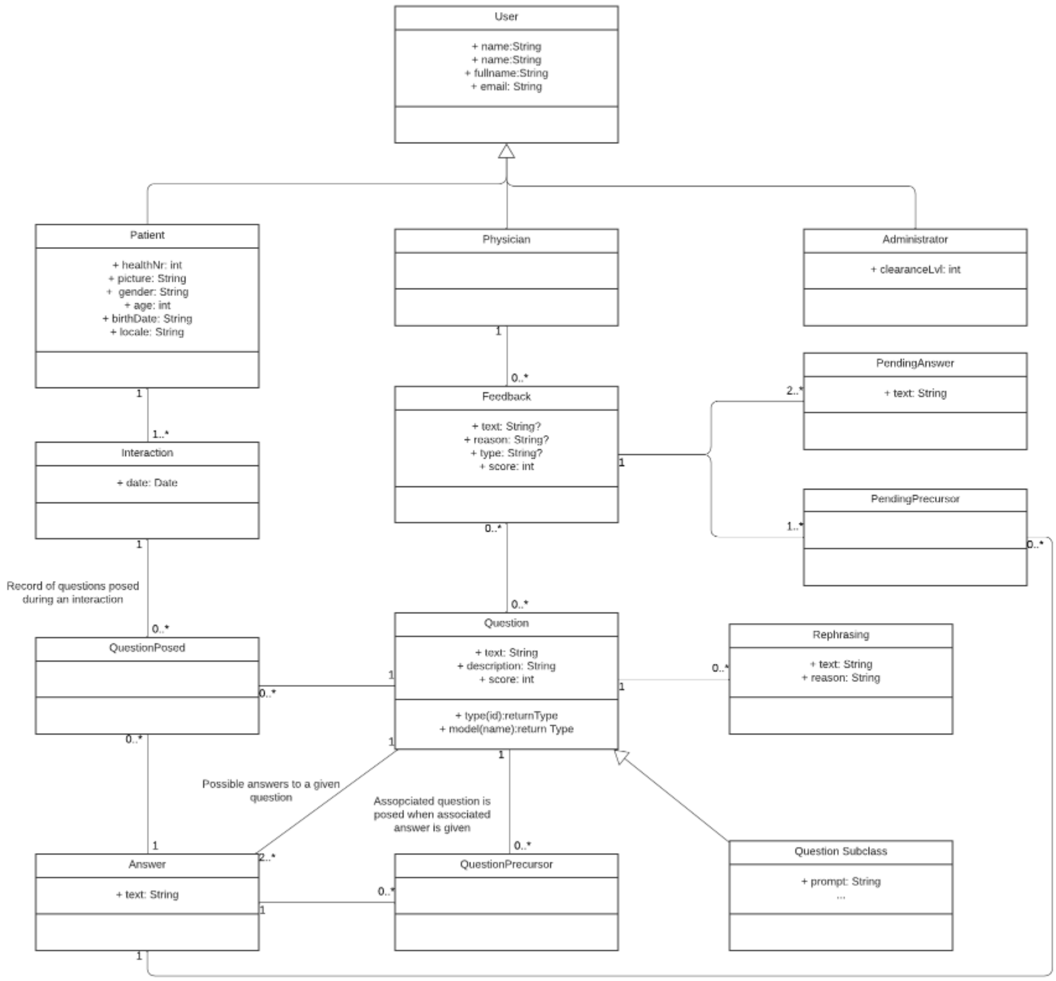

Figure 4.2: Conceptual data model

Further subdivision of the question texts into single phrases separated by AND or OR operators was considered for the sake of reducing question text redundancy in the database and lending itself better to computer analysis, but was ultimately decided against due to the added burden of complexity it would place on physicians in providing feedback due to having to make sure their phrasing was segmented correctly before submission.

4.2 Back end

For the back end, Express was used in conjunction with Sequelize. Express is a popular back end web application framework for constructing RESTful APIs with the use of Node.js while Sequelize is a modern TypeScript and Node.js ORM(Object Relational Mapper) that supports Oracle, Postgres, MySQL, MariaDB, SQLite and SQL Server, and more other such database management systems.

The database definition and population was also implemented using Sequelize as it proved to be much more intuitive and efficient for handling structural changes to the seed data compared to pure SQL scripting. The extra layer of flexibility it provided will also be of value should the project's continuation deem another relational type of database management systems supported by Sequelize to be more apt in the future.

Transactions were used to ensure the system complied with the requirements of availability by ensuring it would not be liable to unexpected errors halting its functions whilst submitting new data to the database and consequently fall into an inconsistent state during operation.

The routing was divided into 3 subsections:

- User routes
- Patient routes
- Physician routes

User routes deal with user registration and authentication. Patient routes with the fetching of questions and the submission of answers as well as fetching of interaction data as a whole. Physician routes with the submission and acceptance of physician feedback. Full list of routes and their use is available in the appendix section.

4.3 Patient Application

The framework employed in the development of the patient application was Flutter. Intuitive and easy to use, Flutter is an open source UI software development kit. Having been developed and supported by Google released in 2017, it has a plethora of quality documentation available online. Better yet, it allows for cross platform development from a single codebase. Be it Android, iOS, Fuchsia, macOS, Windows, Linux or any Web platform, they are all under its purview.
This multi-platform reach made it a superior option in comparison to android NDK(native development kit) used in the prior implementation of the project. Having the system be easily available in both android and iOS format would meet the flexibility requirements for this project.
Additionally, being available in web format is also valuable as hospital emergency units can get quite crowded. When met with a lack of hardware versus a correspondingly greater number of patients, being able to have the patients still access the system via any browser in their personal device would prove to be a powerful resort in helping to supplant that issue.

The architecture for the application was centered around Flutter Provider. Provider is a Flutter

architecture that, as its name suggests, provides the presentation layer with the model that it requires. It does so by having the data models that must be provided extend the *ChangeNotifier* class and then supplying a *ChangeNotifierProvider* widget for that model as the parent widget to the smallest sub-tree of widgets that would require it. Widgets are the central class in the Flutter hierarchy that act as descriptors for the visual components present in the user interface.
When a change occurs in one of the provided data models, they are able to send a notification to all *Consumer* widgets that subscribed to that particular *ChangeNotifier* implementation in the widget sub-tree. This notification lets them know that they need to rebuild their respective code which in turn updates the user interface in a more efficient way than to having to rebuild the widget sub-tree in its entirety.
The leading coordinator model in the application is the *Interaction*. A model containing the patient data was initially considered for this role but deemed redundant since the application does not need to access other past interactions from the patient and rather only needs to focus on conducting a new one. With each interaction being completely independent from the rest and the patient data being immutable throughout the process, it made more sense for the patient data to be encapsulated by the *Interaction* class instead.

4.3.1 Application Interface

Regarding the application interface itself, there is:

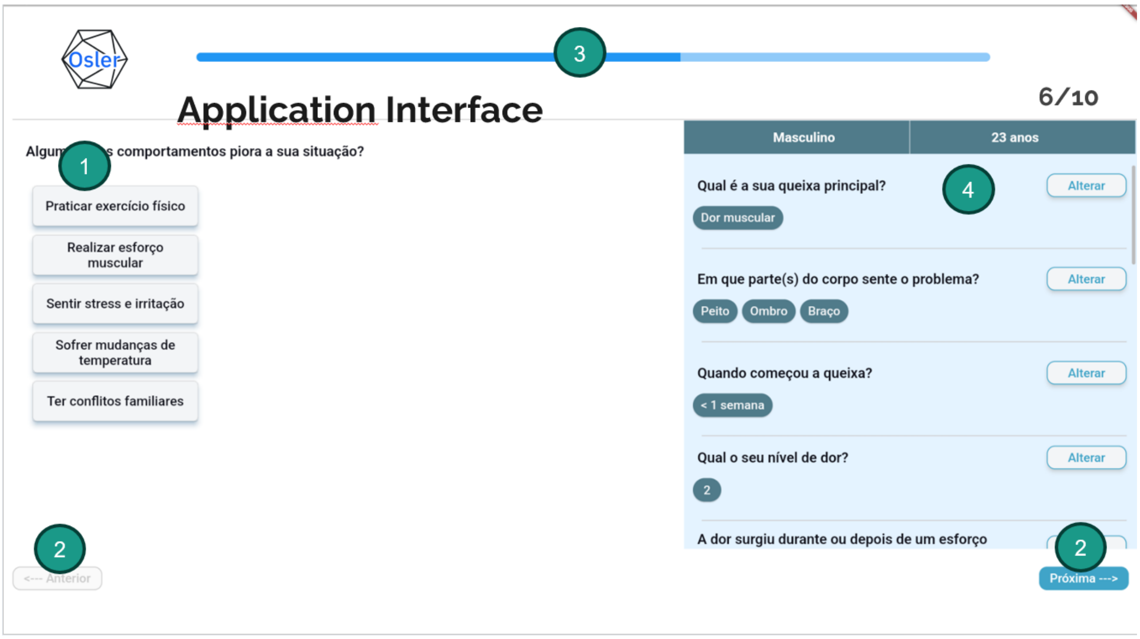

Figure 4.3: Interface elements

1. The display section for questions
2. Basic navigation buttons

3. Progress bar
4. Prior answers tracker

Within the questions' staging area of sorts, questions can be shown in either one-by-one in a sequential format or, like here, in a grid format, for faster answer selection.
The navigation buttons on the bottom allow the patient to go back and forth across different question groups/phases of questioning.
The progress bar on top helps the patient keep track of their completion rate. I decided on a continuous progress bar because as the interaction progresses, follow up questions may be fetched depending on the answers the patient has given causing the progress to have to be recalculated.
Lastly, there is a panel on the right where the patient's previous answers are tracked for easier reviewing and where faster non-sequential navigation is made available.

4.3.2 Application Data Flow

The application data flow, or rather, the data that is effectively being collected by the application is as follows:

1. Patient Data
2. Chief Complaint
3. Body Parts affected (conditional)
4. Descriptive Questions
5. Aggravation Factors
6. Attenuation Factors
7. Symptoms
8. Medical History
9. Medication being taken
10. Remaining 'follow-up' questions

Firstly, the application collects the patient's basic data such as age and gender, along with their name and id number so the system and physicians can differentiate between patient.
Secondly, it queries the patient on their chief complaint, or in other words, the main reason that caused them to seek medical assistance.
In the case this complaint is one where knowing the body areas the patient feels pain or discomfort in is relevant, a model of the human body is presented to the patient so that they may select them.

One such relevant complaint is for example the case of sciatic pain. While this pain often starts in the lower back, it can also irradiate down the leg all the way to the foot and it is important for doctors to know whether or not that is the case because it lets them know more on the extent of the injury and its severity.

Following that, come the questions that help describe the chief complaint. For example, 'when did the complaint start?', 'how would you describe the level of pain you're experiencing?' - in the cases where the patient indeed is experiencing pain- and so on. These questions are naturally always shown in a one-by-one format because they aren't all yes or no questions.

The Aggravation and attenuation factors that come next refer to behaviors, habits or situations that as their names suggest, either worsen or improve the patient's condition. An example of each for the case of sciatic pain would be bending one's back in order to lift something heavy as an aggravation factor, and instead bending your knees and keeping your back straight as you lift it as an attenuation factor. The former is associated with the patient's pain and discomfort rising, while the latter with preventing them.

Symptoms refers to any other secondary complaints the patient might have such as fever, dizziness or vomiting.

Lastly, the application would ask the patient if they have a history of past or chronic medical issues – such as allergies - and what medication they are currently taking as this is something that could have had an impact on their condition and is therefore something that the physician should take into consideration when reaching a diagnosis.

Any remaining follow-up questions that were not able to be presented to the patient yet appear at this last point of the interaction. To further shed some light on this last part, as the patient answers questions, so does the backlog of follow-up questions that are fetched based on those answers build up. If possible, these questions are presented to the patient during their respective phase, but given that the type a follow-up question will belong to is unpredictable, if its type is that of a phase that the patient has already answered, then there needs to be some other phase where this question can be posed and that is where this final 10th phase comes in. Unlike the previous phases, this one is cyclical in that until there are no more follow-up questions the interaction will remain there. The set of questions that it shows in each of its iterations is that of the question type with the largest number of questions that have yet to be posed to the patient. Depending on the amount of follow-up questions of the same type fetched in response to the prior iteration's answers, if any, this means the type of question presented could remain the same. In short, with each iteration of this last phase, the application aims to present as many of the backlogged non-answered questions until non remain.

Question fetching only occurs when moving in between phases so as to avoid having to continuously fetch follow-up questions and potentially delete old ones that no longer apply as the answer selection they are tied to changes within the same phase. This prevents the question fetching from affecting the overall interaction time by forcing the application to wait for it to finish.

Another factor that works towards the stability and availability requirements for the system is that

it the records for the interaction are only submitted for storage at the end once it is over. This further minimizes the points of failure where the system could be left inconsistent compared to a more incremental approach like submitting the answers as they are given and avoids having to make corrections to the answers previously recorded should the patient change their mind regarding any of them.
Once the patient has finished reviewing their answers and ends the interaction, those answers are submitted and will be later shown to the physician via the data visualization application of the project's second working part.

4.3.3 Human Body Model

The human body model is the only exception to the prior discussed format of either one-by-one or grid-like and it is only presented once if at all, but it is nonetheless a helpful tool for querying the patient on locations. It was created based on an online article [23] by manually delineating where each body part's borders were from a full image of the human body that had been converted into a Scalable Vector Graphics (SVG) image. This resulted in several distinct SVG images pertaining to each outlined body part. This was done using a combination of Figma and Inkscape. With Figma, the original image was converted into an SVG image and divided into different SVG images, while with Inkscape these different SVG images were then translated to where they were supposed to be and combined into a single SVG that image contained the paths tracing all the SVG body part images and the translation coordinates that dictated their placing. These paths and coordinates were then extracted and uploaded into the database's record for each body part.
Using these paths it was possible to upload the SVG images onto the screen and position them properly on top of the original image of the human body in order to create the human body model with clickable sections as shown in Figure 4.4.

The value of this approach comes in the added flexibility to altering the human body sections as compared to a more hard-coded approach of mapping clicks in certain coordinates of the screen to different versions of the original human body image with a corresponding section highlighted, as was in the prior implementation of the platform, for example.
Initially, when clicked, these would open up a selection menu for all the chief complaints associated with the selected body part, but this was forgone for the current use since not all chief complaints are necessarily associated with a body part. However, there is some potential merit in exploring the possibility of giving the patient the option to see the human body model during chief complaint selection. In the case of the patient not knowing what their chief complaint actually is, having them be able to instead first select one or several body parts they associate with their complaint and be shown several possible complaints could prove to be quite helpful.

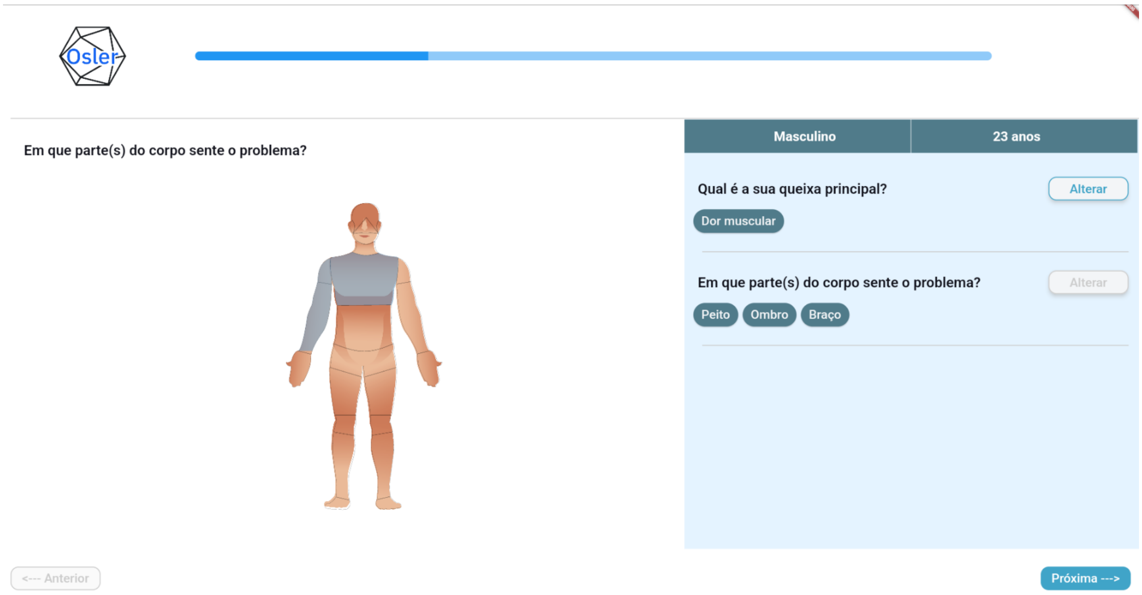

Figure 4.4: Clickable model of the human body

4.3.4 Rephrasing

This is an especially relevant feature because as mentioned before, the current population of Portugal is quite aged overall and so a considerable amount of patients might not be as technologically adept or possess a high level of literacy. To ensure they understand a question's intended meaning, it is imperative that there are some interpretation aids in place.

Aside from the use of descriptions, the platform leverages the rephrasing feedback provided by physicians to provide several alternative phrasings to any more complex question for every potential reason for its misinterpretation.

This is propelled by the *Did not understand* answer available in every descriptive question, the questions more likely to possess a more convoluted phrasing that might lead to misunderstanding. Once they answer a question as not understanding it, then the system will give the patient a list of reasons for that why that could be and on its selection, present another phrasing more suited for the patient in accordance with the reason selected.

Should an appropriate phrasing not yet be available, physicians will see that the patient did not understand the question and will then be able to provide one for the question along with the reason for it as described by the patient. In this way, we can see how the feedback directly provided by the patient can be used to motivate the physicians to contribute to amending the faults in the system and helping to improve it.

Chapter 5

Final considerations

This last section handles a brief summary of the project as a whole along with what could be done in the future to improve it further.

5.1 Conclusions

Even during the development of this project the issue of lack of personnel and overcrowding in hospital EDs has only grown worse and more apparent in Portugal. The project proposed in this document discusses the use of CAA as a potential solution.
Similar approaches are taken into account via the state of the art so as to provide a good perspective on the way such a solution would come about and ensure a solid standpoint for discussing the conjectures behind the optimal approach. Likewise, for the implementation of the underlying system itself, the functional and quality requirements are taken into account to ensure the project's development followed along the its goals.
The database was designed so as to be open to change giving the platform the potential to be improved as it operates and to become a part of the foundation for a new wave of technology-aided medical logistics and processes that would be of great help in overcoming these growing issues.
That said, there are some realizations that have come about through this foray into HCI in CAA and so on. First and foremost, that the human, or rather, physician aspect cannot yet be removed from the development of such applications. To be specific, such applications cannot solely rely on the capabilities of AI and ML techniques alone in order to become viable, as was initially believed when first tackling this project. Which is why although a lot of these applications make sure to highlight the ingenious algorithms behind their product, they also make sure to mention the extensive contribution and collaborative efforts with teams of physicians in its development as a needed seal of validation. Perhaps this will inevitably change in the future as more discoveries are made in this area, but at least for now it is a very staunch requirement and I believe will remain so until the results begin to show otherwise. In that same purview, regrettably, the platform could not be deployed and validated in a real hospital ED environment as expected for the project's third

working part in time for the final deadlines for this thesis. As such, the solution discussed in this document is as of yet unverified in regards to whether or not it fulfills the intended purpose adequately and so will have to stand on its intrinsic merits alone and the reasoning behind them.

5.2 Future Work

There are several worthy avenues of improvement that could be taken from here.

The current system's question precursors are all disjunctive in that they act independently from each other and if one question possesses multiple, it only requires the presence of one to be elicited to the patient. One point of interest that could be explored in the future is to allow to make them conjunctive and for a certain question to only be elicited when all or some of its precursors are observed in the interaction. The combination of applying operators such as these to the system would allow for more complicated interview methods to be faithfully translated into it.

Only textual answers are currently supported, but other formats should be considered. For example, in the case of pain description, there are standards for querying patients on the degree of pain they are experiencing. Namely, through the use of images of expressions rather than textual or numerical description, as they make for a more intuitive and objective measurement.

The human body model interface could be further enhanced by allowing the patient to be more precise in the areas they select or go more into detail as to what exactly it is they feel in each area. Visual methods of recording and displaying data such as this can be very powerful due to their inherently condensed format that allows for a lot of information to be conveyed in a relatively simple way when done well.

Collection of more diverse types of data such as vital signs, patient expression throughout the interaction and so on would provide a more comprehensive picture of the patient's condition.

Lastly, since it was sadly not able to be explored within the deadline for this thesis, it would be very interesting to see how successful the employment of ML techniques as a support for this sort of platform would be.

References

[1] Physicians (per 1,000 people) - portugal. https://data.worldbank.org/indicator/SH.MED.PHYS.ZS?most_recent_value_desc=false&locations=PT. Accessed: 2023-07-15.

[2] Population ages 65 and above (% of total population) - portugal. https://data.worldbank.org/indicator/SP.POP.65UP.TO.ZS?most_recent_value_desc=true&locations=PT. Accessed: 2023-07-15.

[3] Quase metade dos médicos trabalha fora do sns. https://www.dn.pt/sociedade/quase-metade-dos-medicos-trabalha-fora-do-sns-14259825.html, 2021. Accessed: 2023-07-15.

[4] Distribuição de médicos por idade. https://ordemdosmedicos.pt/estatisticas-nacionais/#1505070622822-9e31b1b1-6d1a, 2022. Accessed: 2023-07-15.

[5] Stephanie Aboueid, Samantha Meyer, James R Wallace, Shreya Mahajan, Ashok Chaurasia, et al. Young adults' perspectives on the use of symptom checkers for self-triage and self-diagnosis: Qualitative study. *JMIR public health and surveillance*, 7(1):e22637, 2021.

[6] Rodrigo Antunes. Ministro da saúde admite que falta de médicos vai manter-se nos próximos dois a três anos. https://observador.pt/2022/11/04/ministro-da-saude-admite-que-falta-de-medicos-vai-manter-se-nos-proximos-dois-a-tres-anos/, 2022. Accessed: 2023-07-15.

[7] PP Barros, SR Machado, and JA Sim es. Portugal: health system review. health systems in transition. *European Observatory on Health Systems and Policies*, 9(5).

[8] Nicholas Boltin, Diego Valdes, Joan M Culley, Homayoun Valafar, et al. Mobile decision support tool for emergency departments and mass casualty incidents (edit): initial study. *JMIR mHealth and uHealth*, 6(6):e10727, 2018.

[9] Keeve Brodman, Albert J Erdmann, Irving Lorge, Harold G Wolff, and Todd H Broadbent. The cornell medical index: an adjunct to medical interview. *Journal of the American Medical Association*, 140(6):530–534, 1949.

[10] Keeve Brodman, ANDRIANUS J van WOERKOM, Albert J Erdmann, and Leo S Goldstein. Interpretation of symptoms with a data-processing machine. *AMA Archives of Internal Medicine*, 103(5):776–782, 1959.

[11] Anthony C Carr, Anindya Ghosh, and RJ Ancill. Can a computer take a psychiatric history? *Psychological medicine*, 13(1):151–158, 1983.

[12] Molla S Donaldson, Janet M Corrigan, Linda T Kohn, et al. To err is human: building a safer health system. 2000.

[13] Eva SV Eijk, Marijke Wefers Bettink-Remeijer, Reinier Timman, and Jan JV Busschbach. From pen-and-paper questionnaire to a computer-assisted instrument for self-triage in the ophthalmic emergency department: process and validation. *Computers in Biology and Medicine*, 66:258–262, 2015.

[14] Healthwatch Enfield. Using technology to ease the burden on primary care, 2019.

[15] Sari Kujala, Iiris Hörhammer, Riitta Hänninen-Ervasti, and Tarja Heponiemi. Health professionals' experiences of the benefits and challenges of online symptom checkers. In *Digital Personalized Health and Medicine*, pages 966–970. IOS Press, 2020.

[16] David M Levine and Ateev Mehrotra. Assessment of diagnosis and triage in validated case vignettes among nonphysicians before and after internet search. *JAMA network open*, 4(3):e213287–e213287, 2021.

[17] Deborah Lupton and Annemarie Jutel. 'it's like having a physician in your pocket!'a critical analysis of self-diagnosis smartphone apps. *Social Science & Medicine*, 133:128–135, 2015.

[18] Rachel E Lutner, Michael F Roizen, Carol B Stocking, Ronald A Thisted, Steve Kim, Peter C Duke, Peter Pompel, and Christine K Cassel. The automated interview versus the personal interview: do patient responses to preoperative health questions differ? *The Journal of the American Society of Anesthesiologists*, 75(3):394–400, 1991.

[19] Asos Mahmood, David K Wyant, Satish Kedia, SangNam Ahn, M Paige Powell, Yu Jiang, and Soumitra S Bhuyan. Self-check-in kiosks utilization and their association with wait times in emergency departments in the united states. *The Journal of emergency medicine*, 58(5):829–840, 2020.

[20] M Kim Marvel, Ronald M Epstein, Kristine Flowers, and Howard B Beckman. Soliciting the patient's agenda: have we improved? *Jama*, 281(3):283–287, 1999.

[21] Peter McKenna, Samita M Heslin, Peter Viccellio, William K Mallon, Cristina Hernandez, and Eric J Morley. Emergency department and hospital crowding: causes, consequences, and cures. *Clinical and experimental emergency medicine*, 6(3):189, 2019.

[22] MCP. Be diz que quase 1,7 milhões de portugueses estão sem médico de família. governo reconhece "situação difícil", mas responde com abertura de vagas a concurso. `https://cnnportugal.iol.pt/be/saude/be-diz-que-quase-1-7-milhoes-de-portugueses-estao-sem-medico-de-familia-governo-responde-com-vagas/20230510/645bc9f8d34ea91b0aac5c87`, 2023. Accessed: 2023-07-15.

[23] Joseph Muller. Adding interactivity to an image in flutter. `https://jtmuller5-98869.medium.com/interacting-with-svg-images-in-flutter-f9d204ab446d`, 2020. Accessed: 2022-12-20.

[24] Paulo Pacheco, Fátima Santos, João Coimbra, Eva Oliveira, and Nuno F Rodrigues. Designing effective user interface experiences for a self-service kiosk to reduce emergency department crowding. In *2020 IEEE 8th International Conference on Serious Games and Applications for Health (SeGAH)*, pages 1–8. IEEE, 2020.

[25] Paulo Alexandre Gonçalves Pacheco. *Self-service kiosk-based anamnesis system for emergency departments*. PhD thesis, 2021.

[26] D. Ramanee Peiris, Peter Gregor, and Norman Alm. The effects of simulating human conversational style in a computer-based interview. *Interacting with Computers*, 12(6):635–650, 2000.

[27] Karell G Pellé, Clotilde Rambaud-Althaus, Valérie D'Acremont, Gretchen Moran, Rangarajan Sampath, Zachary Katz, Francis G Moussy, Garrett Livingston Mehl, and Sabine Dittrich. Electronic clinical decision support algorithms incorporating point-of-care diagnostic tests in low-resource settings: a target product profile. *BMJ global health*, 5(2):e002067, 2020.

[28] Jesse M Pines, Joshua A Hilton, Ellen J Weber, Annechien J Alkemade, Hasan Al Shabanah, Philip D Anderson, Michael Bernhard, Alessio Bertini, André Gries, Santiago Ferrandiz, et al. International perspectives on emergency department crowding. *Academic Emergency Medicine*, 18(12):1358–1370, 2011.

[29] Martien J Quaak, R Frans Westerman, Jan A Schouten, Arie Hasman, and Jan H Van Bemmel. Patient appreciations of computerized medical interviews. *Medical Informatics*, 11(4):339–350, 1986.

[30] Martien J Quaak, R Frans Westerman, and Jan H van Bemmel. Comparisons between written and computerised patient histories. *Br Med J (Clin Res Ed)*, 295(6591):184–190, 1987.

[31] Paul G Ramsey, J Randall Curtis, Douglas S Paauw, and Marjorie D Wenrich. History-taking and preventive medicine skills among primary care physicians: an assessment using standardized patients. *The American journal of medicine*, 104(2):152–158, 1998.

[32] Fátima Jacinta Pereira dos Santos. *E-anamnesis: a clinical observation electronic platform for emergency departments*. PhD thesis, 2021.

[33] Malte L Schmieding, Marvin Kopka, Konrad Schmidt, Sven Schulz-Niethammer, Felix Balzer, and Markus A Feufel. Triage accuracy of symptom checker apps: 5-year follow-up evaluation. *Journal of Medical Internet Research*, 24(5):e31810, 2022.

[34] Madhumita Sinha, Kai-Ning Khor, Ashish Amresh, David Drachman, and Alan Frechette. The use of a kiosk-model bilingual self-triage system in the pediatric emergency department. *Pediatric emergency care*, 30(1):63–68, 2014.

[35] Warner V Slack. A history of computerized medical interviews. In *Buying Equipment and Programs for Home or Office*, pages 138–145. Springer, 1987.

[36] Warner V Slack, Phillip Hicks, Charles E Reed, and Lawrence J Van Cura. A computer-based medical-history system. *New England Journal of Medicine*, 274(4):194–198, 1966.

[37] Warner V Slack, Hollis B Kowaloff, Roger B Davis, Tom Delbanco, Steven E Locke, Charles Safran, and Howard L Bleich. Evaluation of computer-based medical histories taken by patients at home. *Journal of the American Medical Informatics Association*, 19(4):545–548, 2012.

[38] Warner V Slack and Lawrence J Van Cura. Patient reaction to computer-based medical interviewing. *Computers and Biomedical Research*, 1(5):527–531, 1968.

[39] Francisco De Sousa Webber. Semantic folding theory and its application in semantic fingerprinting, 2016.

[40] David Zakim, Niko Braun, Peter Fritz, and Mark Dominik Alscher. Underutilization of information and knowledge in everyday medical practice: Evaluation of a computer-based solution. *BMC Medical Informatics and Decision Making*, 8(1):1–12, 2008.

[41] Haipeng Zhang, Dimitar Dimitrov, Lynn Simpson, Nina Plaks, Balaji Singh, Stephen Penney, Jo Charles, Rosemary Sheehan, Steven Flammini, Shawn Murphy, et al. A web-based, mobile-responsive application to screen health care workers for covid-19 symptoms: rapid design, deployment, and usage. *JMIR Formative Research*, 4(10):e19533, 2020.

Chapter 6

System Requirements

Presenting here the full list of requirements for the system

6.1 Functional Requirements

Requirement #1: The system must perform user registration
Reasoning: User data must be stored to allow the system to distinguish between different users and for future reference should the patient use the system again.

Requirement #2: The system must collect the patient's chief complaint
Reasoning: As the leading reason behind the patient's visit to the ED, it will play the largest role in their diagnosis.

Requirement #3: The system must collect the patient's medical history
Reasoning: Patient medical history may be a factor in the patient's condition and can therefore have an influence on the differential diagnosis reached by the doctor.

Requirement #4: The system must collect the medication being currently taken by the patient
Reasoning: Medication can affect the patient's condition so it's important for it to be recorded.

Requirement #5: Collection of base questions related to chief complaint
Reasoning: The system must at the very least query the patient on the basic questions present in the last implementation so as to ensure the physician has a good description of the chief complaint's details.

Requirement #6: System must be open to change
Reasoning: So that it can grow and improve upon itself as time goes on.

Requirement #7: Fetching of subsequent questions
Reasoning: To provide a more personalized interaction experience to the patient.

6.2 Quality Requirements

Requirement #8: The system must be able to accommodate users of lower levels of literacy and/or digital competency
Requirement type: Usability
Reasoning: The user experience and satisfaction from using the system will be tightly tied to its ease of use.

Requirement #9: The patient must track the questions the patient did not understand and why.
Requirement type: Usability
Reasoning: So that the patient is not forced into answering a question they did not understand and so that the system is more aware of how it should be improved.

Requirement #10: The system must not take too long in fetching
Requirement type: Efficiency/Performance
Reasoning: Too long and the time required to perform an interaction will be impacted, and consequently, the system's ability to make the anamnesis process faster and more efficient will be put into question.

Requirement #11: The system must be flexible in regards to the hardware it can operate on
Requirement type: Flexibility
Reasoning: In order to facilitate the acquisition of compatible hardware for the hospitals.

Requirement #12: The system must always be available
Requirement type: Availability
Reasoning: EDs are open at all times, and so the same should be expected of the system.

6.3 System Routes

Presenting here the full list of routes utilized by the server-side of the platform.

6.3.1 User Routes

URI prepended by namespace: *'/users'*

Table 6.1: user routes

URI	Request Type	Use
'/register'	POST	Registers a new user into the system
'/login/patient'	POST	Authenticates a patient
'/login/physician'	POST	Authenticates a physician

6.3.2 Patient Routes

URI prepended by namespace: *'/patients'*

Table 6.2: patient routes

URI	Request Type	Use
'/interaction/new'	POST	Begins a new patient interaction
'/questions/core'	GET	Fetched the system's base or core questions (those without precursors)
'/questions/answer'	POST	Records the answers for the associated questions sent
'/questions/:precursor'	GET	Fetches all questions who have the answer of id *:precursor* as a precursor
'/questions/:precursor/model/:model'	GET	Fetches all questions who have the answer of id *:precursor* as a precursor belonging to subclass *:model*
'/questions/:precursor/model/:model'	GET	Fetches all questions who have the answer of id *:precursor* as a precursor belonging to the sub-classes of *:model*
'/latestInteractions'	GET	Fetches all the basic data of the latest interactions within a certain time frame
'/interaction/:id/flow'	GET	Fetches all the data on the interaction of id *:id*

6.3.3 Phycisian Routes

URI prepended by namespace: *'/phycisians'*

Table 6.3: physician routes

URI	Request Type	Use
'/additionFeedback'	POST	Creates physician feedback record for the addition of a new question
'/deletionFeedback'	POST	Creates physician feedback record for the deletion of a new question
'/alterationFeedback'	POST	Creates physician feedback record for the addition of an alternative phrasing to a question
'/feedback'	GET	Fetches all unaccepted physician feedback provided so far
'/acceptFeedback'	POST	Accepts and applies the provided physician feedback
'/rejectFeedback'	POST	Rejects and deletes the provided physician