Project Requirements Document V1

Project: I 👀 U (I See You)
Company: Teladoc
Team Name: 404: Team Name Not Found
Authors: Ishana Narayanan (Team Lead), Sophie Groenwold (Scribe), Steven Huynh-Tran, Anthony Palomera, Evan Yip, Katelyn Zhang

Introduction

Background

The field of telehealth – where physicians can provide care to patients without an in-person visit – has seen a massive acceleration since March 2020. As providers and consumers sought ways to safely give and access healthcare amidst the uncertainty of the COVID-19 pandemic, the use of telehealth services expanded to 38 times its pre-pandemic use (McKinsey). As a major player in the telehealth industry with 10.5 million virtual visits delivered in 2020, Teladoc plays a key role in ensuring patients' sense of safety and satisfaction.

With this increase in consumers and given the ever-evolving nature of COVID-19, it is vital that patients feel that their consultation history is clear and accessible, and that physicians can efficiently and effectively retrieve patient information. Teladoc's current functionality includes a chatbot that patients can interact with to access the resources or services relevant to their concerns. This use of a chatbot has helped patients navigate Teladoc's platform in an intuitive and natural language-powered format. For this project, we will leverage these benefits by building a pre-consultation interface that conducts screening and symptom checking through a patient-chatbot dialogue. Patients will be able to access their chat history, as well as a summary of their chat, and physicians will be able to view a patient's concerns and symptoms before the start of their appointment.

Problem

There are two issues that we hope to address with this project: first, that patients have a critical need for transparency in their medical visits, and second, that physicians are in demand and thus can benefit from efficient consultations. Transparency in patient-physician communication is a frequently overlooked aspect of medical care, yet is extremely important to a patient's understanding and sense of autonomy (Robbins et. al., 2011). In a telehealth context, these problems with communication are heightened by worries of information security and data privacy (AMA Journal of Ethics). Allowing patients to see records and summaries of their chatbot history will increase transparency on this end; and having a clear record of a patient's questions will help physicians address these concerns.

In addition, pre-consultation screenings and initial symptom identification will lower the workload of a physician during their appointment with a given patient. While similar methods of doing pre-screens are available, they typically take place through deterministic survey software (for instance, Google forms). This restricts the accuracy of an initial diagnosis and can feel impersonal for the patient. Our solution utilizes the power of natural language, as collected by chatbot responses, and thus improves the accuracy of pre-consultation data; in turn, this will increase the efficiency of a physician during their appointments.

Goals & Objectives

Our goal is to create an application which allows patients to complete pre-appointment screens and initial symptom identification to streamline physician consultation time and increase diagnosis transparency. The pre-consultation screens take place in a text interface with a chatbot which asks tailored questions about health and current symptoms. Using NLP and other machine learning methodologies, the application will then summarize and provide recommendations for different courses of action to the physician prior to the appointment.

For patients, this will include an authentication into the website to ensure privacy of appointments and medical data. After logging in, patients will be directed to a home page with a list of their upcoming appointments. For each appointment, the patient can complete their pre-consultation screen by having a conversation with the chatbot pop-up. After completing the screen, patients will be able to view their chat history on a separate page.

For physicians, this will also include authentication into the website to ensure privacy of patient data. After logging in, physicians will be directed to a home page which has a list of appointments for the day. For each patient, physicians can view the pre-consultation screen chat summarization as well as initial diagnosis based on symptoms.

To implement this application, we plan to configure a chatbot using either Amazon Lex or Microsoft Azure to communicate with patients and ask tailored questions about their symptoms and health. For the application in general, React will be used to build front-end components including an appointments page, chat box, and chat summarization pages. Node.JS will be used on the backend to communicate with our database MongoDB which will store patient information, appointments, and chat log data. Amazon Comprehend Medical and Infermedica API will be used to offer diagnosis based on initial symptom screening. These recommendations as well as a summary of patient conversation with chatbot will be available to the physician prior to the appointment.
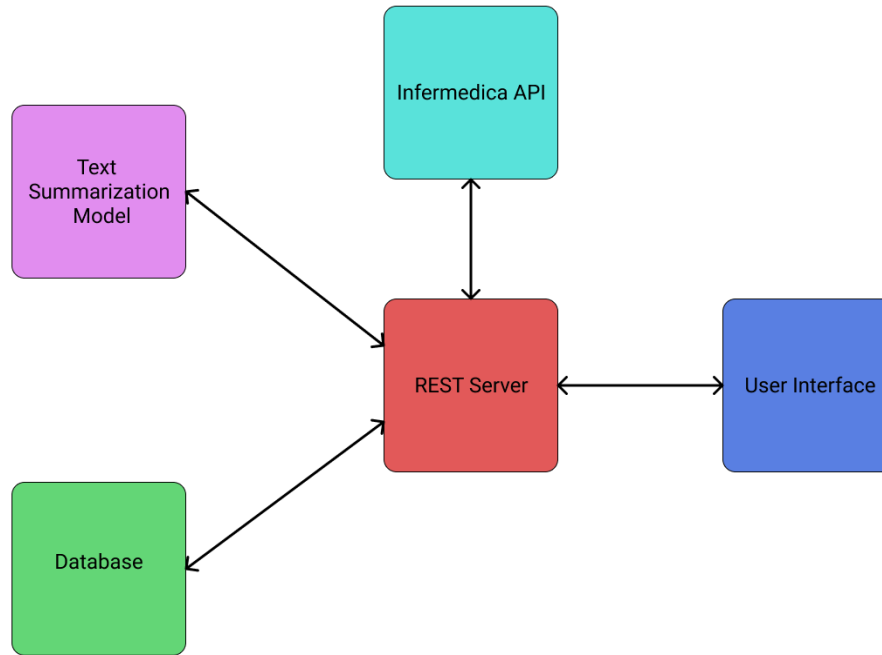
Assumptions

This application will only be available to patients and physicians who are already affiliated with Teladoc. Patients will be using Teladoc's current resources to schedule appointments which will then be populated into our database.
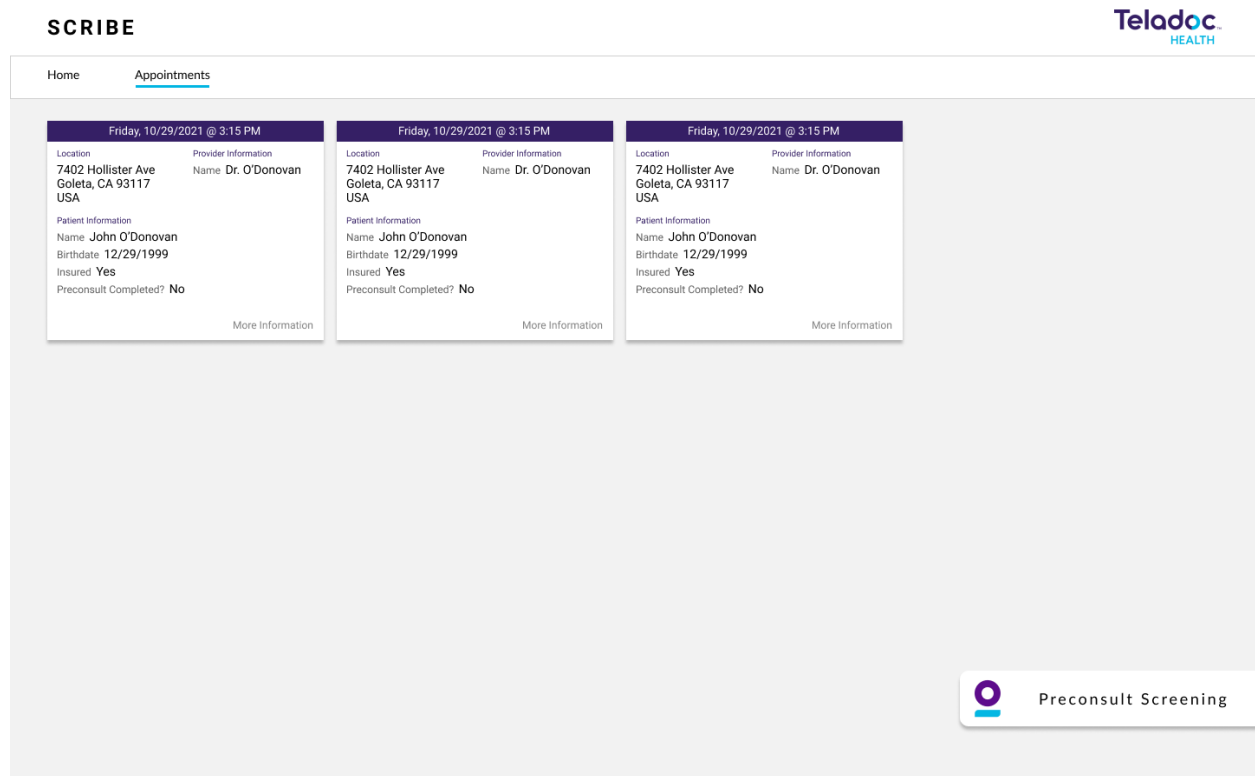
Project Design

High-level System Architecture

The diagram above represents the different parts of our web application and how they interact with each other. Patients and Providers will be able to access the application through a web user interface. After authentication, the frontend will be able to communicate with our backend REST API. The main purpose of this backend API is to store user information, chat bot interactions, scheduled appointments, and reports for Doctors. All of this information will be stored on our database. Additionally, the backend will need to be able to communicate with third party APIs. In our case, we need to call the Infermedica API for symptom diagnosis. Finally, the backend will communicate with an AWS lambda function through Amazon Comprehend Medical that will host the Text Summarization Model we use. All together these components form the high level architecture of our web application.
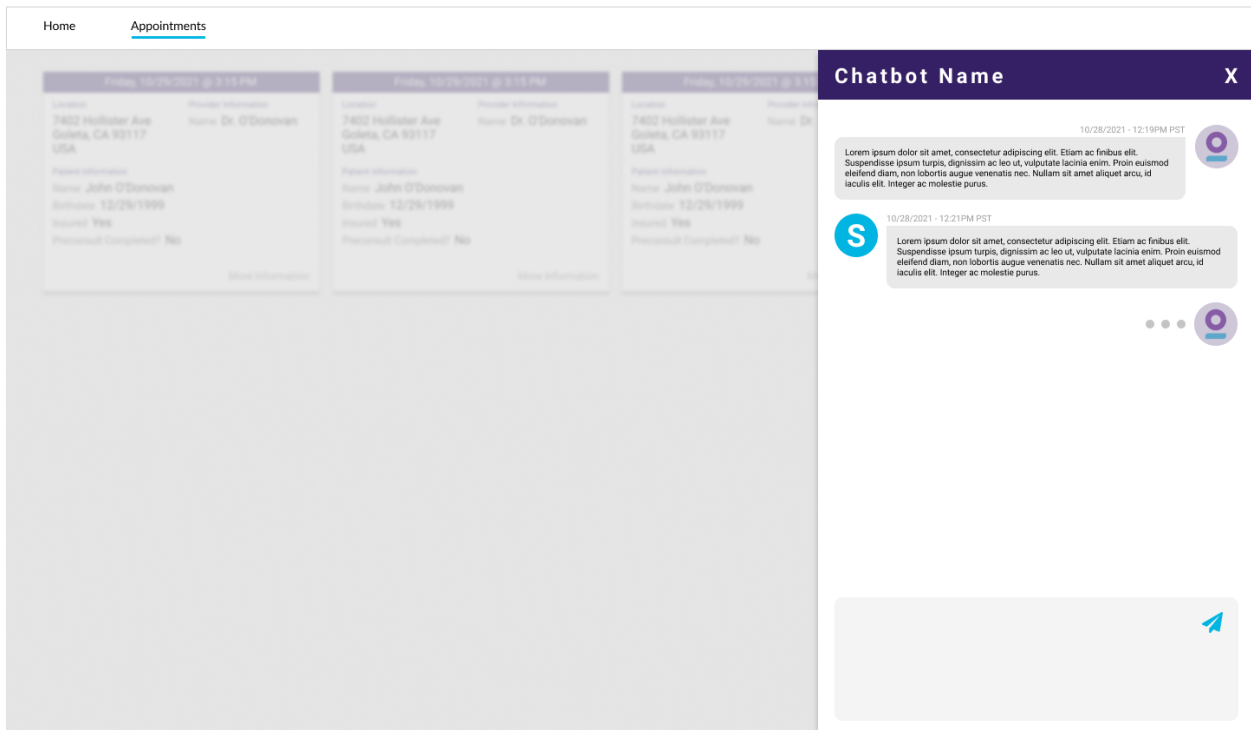
User Interface

The primary interface for both patients and providers will be through a web application. Below are several screens that a user may see during the pre-consult screening flow.

SCRIBE

Home   Appointments

Friday, 10/29/2021 @ 3:15 PM
Location	Provider Information
7402 Hollister Ave Goleta, CA 93117 USA	Name Dr. O'Donovan
Patient Information	
Name John O'Donovan	
Birthdate 12/29/1999	
Insured Yes	
Preconsult Completed? No	
	More Information

Friday, 10/29/2021 @ 3:15 PM
Location	Provider Information
7402 Hollister Ave Goleta, CA 93117 USA	Name Dr. O'Donovan
Patient Information	
Name John O'Donovan	
Birthdate 12/29/1999	
Insured Yes	
Preconsult Completed? No	
	More Information

Friday, 10/29/2021 @ 3:15 PM
Location	Provider Information
7402 Hollister Ave Goleta, CA 93117 USA	Name Dr. O'Donovan
Patient Information	
Name John O'Donovan	
Birthdate 12/29/1999	
Insured Yes	
Preconsult Completed? No	
	More Information

Preconsult Screening

User Interface: Image 1

Image 1 is a mockup of what a patient may see when they successfully log into the patient portal and navigate to their appointments. Here they will see a list of their upcoming appointments (and possibly previous appointments) each with general information regarding that visit. Through this view, a user can conduct their preconsult screening with our chatbot.

User Interface: Image 2

Image 2 shows what a patient would see in the web application when speaking with our chatbot. Through this interface, our system will be able to ask the patient questions regarding their visit in order to analyze their response and compile a summary/report for the provider that will be helping them during their visit.

SCRIBE

Hello, Dr. O'Donovan   Home   Appointments   Sign out

Friday, 10/29/2021 @ 3:15 PM	Friday, 10/29/2021 @ 3:30 PM	Friday, 10/29/2021 @ 3:45 PM
Location / **Provider Information**	**Location** / **Provider Information**	**Location** / **Provider Information**
7402 Hollister Ave / Name Other Doctor	7402 Hollister Ave / Name Dr. O'Donovan	7402 Hollister Ave / Name Dr. O'Donovan
Goleta, CA 93117	Goleta, CA 93117	Goleta, CA 93117
USA	USA	USA
Patient Information	**Patient Information**	**Patient Information**
Name Patient 1	Name Patient 2	Name Patient 3
Birthdate 12/29/1999	Birthdate 12/29/1999	Birthdate 12/29/1999
Insured Yes	Insured Yes	Insured Yes
Preconsult Completed? Yes	Preconsult Completed? No [Send Reminder]	Preconsult Completed? No [Send Reminder]
Tags		
Physical New Patient Diabetes		
See Report	See Report	See Report

User Interface: Image 3

Image 3 shows the view from a provider's perspective. The goal is to help increase provider efficiency which is accomplished by providing easy-to-read notes and summaries at a glance. Here the doctor can see previous and upcoming appointments as well as any analyzed summaries our system has compiled.

SCRIBE

Hello, Dr. O'Donovan   Home   Appointments   Sign out

Appointment #31728541231

Patient Information   **Insurance Information**

Name **Patient 1**   Provider **UC SHIP**

Birthdate **12/29/1999**   Group ID **382931231**

Email **example@email.com**

Phone Number **858-222-2222**

New Patient? **Yes**

> **CHAT LOGS** ›
>
> **Chatbot Name** 10/28/2021 - 12:19PM PST
> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac finibus elit. Suspendisse ipsum turpis, dignissim ac leo ut, vulputate lacinia enim. Proin euismod eleifend diam, non lobortis augue venenatis nec. Nullam sit amet aliquet arcu, id iaculis elit. Integer ac molestie purus.
>
> **Steven Huynh-Tran** 10/28/2021 - 12:21PM PST
> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac finibus elit. Suspendisse ipsum turpis, dignissim ac leo ut, vulputate lacinia enim. Proin euismod eleifend diam, non lobortis augue venenatis nec. Nullam sit amet aliquet arcu, id iaculis elit. Integer ac molestie purus.
>
> **Steven Huynh-Tran** 10/28/2021 - 12:21PM PST
> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac finibus elit. Suspendisse ipsum turpis, dignissim ac leo ut, vulputate lacinia enim. Proin euismod eleifend diam, non lobortis augue venenatis nec. Nullam sit amet aliquet arcu, id iaculis elit. Integer ac molestie purus.
>
> **Steven Huynh-Tran** 10/28/2021 - 12:21PM PST
> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac finibus elit. Suspendisse ipsum turpis, dignissim ac leo ut, vulputate lacinia enim. Proin euismod eleifend diam, non lobortis augue venenatis nec. Nullam sit amet aliquet arcu, id iaculis elit. Integer ac molestie purus.
>
> **Chatbot Name** 10/28/2021 - 12:19PM PST
> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac finibus elit. Suspendisse ipsum turpis, dignissim ac leo ut, vulputate lacinia enim. Proin euismod eleifend diam, non lobortis augue venenatis nec. Nullam sit amet aliquet arcu, id iaculis elit. Integer ac molestie purus.
>
> **Steven Huynh-Tran** 10/28/2021 - 12:21PM PST
> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac finibus elit. Suspendisse ipsum turpis, dignissim ac leo ut, vulputate lacinia enim. Proin euismod eleifend diam, non lobortis augue venenatis nec. Nullam sit amet aliquet arcu, id iaculis elit. Integer ac

User Interface: Image 4

Image 4 is the full report view that is available for each appointment a doctor has. Although seeing summaries of data is useful, oftentimes it is necessary to see the whole picture and this report screen provides doctors with that interface.

There are more pages that need to be added such as the authentication flow, new appointment flow, and others. Furthermore, there are still features to be added to these existing mockups.

Requirements

User Stories

1) As a Physician, I can visit the physician web portal and log in via some authorization so that I can access all my consults.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/1

- Scenario 1: Physician enters the wrong username or password
    - "Incorrect username/password" message will be displayed
    - Users will be prompted to enter their username/password again
- Scenario 2: Physician enters the wrong username and password multiple times over a small time interval
    - "Too many attempts to login" message will be displayed
    - User will be temporarily barred from logging in for a certain time interval
- Scenario 3: Physician enters the correct username/password
    - Account info page will be displayed

2) As a Patient, I can visit the patient web portal and log in via some authorization so that I can access my future appointments with my physician.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/1

- Scenario 1: Patient enters the wrong username or password
    - "Incorrect username/password" message will be displayed
    - User will be prompted to enter their username/password again
- Scenario 2: Patient enters the wrong username and password multiple times over a small time interval
    - "Too many attempts to login" message will be displayed
    - User will be temporarily barred from logging in for a certain time interval
- Scenario 3: Patient enters the correct username/password
    - Account info page will be displayed

3) As a Patient, on the patient portal I can see a list of my upcoming appointments.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/2

- Scenario 1: Patient has logged in and looking through future appointments
    - Patient is on main page which shows appointments in order of most upcoming appointment

- Scenario 2: Patient has clicked on a specific future appointment
  - Patient can see the appointment date as well as the physician's name
  - Patient will then be instructed/guided to click on a button to start their pre-consultation meeting with chatbot

4) As a Physician, on the physician portal I can see a list of my upcoming consultations.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/2

- Scenario 1: Physician has logged in and looking through future consultations
  - Physician is on main page which shows appointments in order of most upcoming consultations
- Scenario 2: Physician has clicked on a specific future consultation
  - Physician can see the appointment date as well as the patient's name
  - Physician will be able to see if patient has completed or has yet to complete their pre-consultation meeting with chatbot

5) As a Patient, I can select an appointment in the patient portal and be directed to a chatbot.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/3

- Scenario 1: Patient hovers over the appointment in the table of upcoming appointments
  - "Open Chatbot" will be displayed as a tooltip
  - The appointment will be highlighted in the table
- Scenario 2: Patient clicks on the appointment in the table of upcoming appointments
  - Chatbot is opened

6) As a Patient, I can interact with a chatbot that will ask me questions about my symptoms and concerns.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/3

- Scenario 1: Patient gives a list of symptoms they have
  - Chatbot asks if they have other specific symptoms it thinks the patient may have based on what the symptoms indicate the source of the symptoms is
  - Chatbot saves this information to be displayed to the physician later
- Scenario 2: Patient says they have no symptoms
  - Chatbot saves this information to be displayed to the physician later
- Scenario 3: Patient tells chatbot that they have some concerns about the appointment
  - Chatbot lets the physician know, and when the physician responds to these concerns, the patient will be alerted

7) As a Patient, I can see a list of all my previous conversations with the chatbot, labeled with a summary and the data of the conversation.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/4

- Scenario 1: Patient goes to the conversation history page
  - Patient sees a table of conversations they had with the chatbot
  - Patient sees that each conversation has a data and a short summary of the main topics of the conversation
  - Patient sees the ability to sort the conversation by oldest or most recent chats
- Scenario 2: Patient hovers over the conversation element in the table
  - "Open Conversation" will be displayed as a tooltip
  - The conversation will be highlighted in the table
- Scenario 3: Patient clicks on the conversation element in the table
  - The complete log of the conversation the patient had with the chatbot is displayed

8) As a Physician, I can see a list of all my patients' previous conversations with the chatbot relating to an appointment they had with me, labeled with a summary and the data of the conversation.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/4

- Scenario 1: Physician goes to the conversation history page
  - Physician sees a table of conversations their patients had with the chatbot regarding appointments made with them
  - Physician sees that each conversation has a data and a short summary of the main topics of the conversation
  - Physician sees the ability to sort the conversation by oldest or most recent chats
- Scenario 2: Physician hovers over the conversation element in the table
  - "Open Conversation" will be displayed as a tooltip
  - The conversation will be highlighted in the table
- Scenario 3: Physician clicks on the conversation element in the table
  - The complete log of the conversation the patient had with the chatbot is displayed

9) As a User(Physician/Patient), I'm able to view a summary of the patient's interaction with the chatbot.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/5

- Scenario 1: Physician is able to view each of their patient's summaries from their discussion with the chat box
  - Physician can click on a respective patient's pre-consultation
  - Physician can view the summary created from patient's discussion with chatbot
- Scenario 2: Patient is able to view each of their own summaries from their discussions with the chatbot from any given pre-consultation
  - Patient can click on a pre-consultation they've had from any of their appointments
  - Patient can view the summary created from their discussion with chatbot from that respective pre-consultation

10) As a Physician, I can see the initial diagnosis based on the chatbot's symptom checking.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/6

- Scenario 1: Physician is able to view diagnosis of patient
  - Physician can analyze the initial diagnosis the chatbot created from its pre-consult with patient
- Scenario 2: Physician is able to view symptoms that led to diagnosis from chatbot
  - Physician can observe the symptoms the chatbot was able to gather from pre-consult meeting with patient and confirm if initial diagnosis was on par with what they expect the patient to be experiencing

11) As a Patient, I am able to see what my physician wrote after reviewing the consultation notes.

Github Issue: https://github.com/ishana-narayanan/404-team-name-not-found/issues/7

- Scenario 1: Patient goes to the consultation notes page
  - Patient sees a list of all notes physician posted on the portal labeled by topic and date
  - Patient sees ability to sort by the oldest and most recent notes
- Scenario 2: Patient hovers over a note
  - "Open Note" is displayed as a tool tip
  - The note is highlighted on the page
- Scenario 3: Patient clicks on a note
  - Patient sees consultation notes displayed in a clean format as a popup
  - Patient sees a close popup button to close the popup
- Scenario 4: Patient clicks on the close button
  - Popup closes
  - Patient returns to the consultation notes page

Appendix

Tech Stack

- **Amazon Lex/Microsoft Azure:** chatbot implementation
- **React:** frontend elements including webpages and authentication
- **NodeJS:** backend to communicate with database
- **MongoDB:** to store chat log information
- **Amazon Comprehend Medical/Infermedica:** chat log text analysis, symptom tracking, and diagnosis recommendation