

Project Requirements Document V2

Project: I 👁️ U (I See You)

Sponsor: 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.

The chatbot conversation will begin by asking the patient for their appointment number. In the backend, the chatbot will verify that this appointment exists for this user and then proceed with symptom screening. Some basic symptoms the chatbot will ask about will include headache, fever, and cough, to name a few. For each symptom, the chatbot will

make a call to Infermedica API to retrieve a more detailed list and then ask the user to identify which case most aligns with their current symptoms. After compiling a list of symptoms, a diagnosis call to Infermedica API will be called with symptom data in addition to general health information about the patient. The initial diagnosis, symptoms, and chat log will then be stored in DynamoDB, indexed by appointment ID.

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. After their consultation with the patient, doctors can then enter their diagnosis into the portal under that particular appointment so that it's available to the patient.

To implement this application, we plan to configure a chatbot using Amazon Lex 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, chatbot, and chat summarization pages. Node.JS will be used on the backend to communicate with our database DynamoDB which will store patient information, appointments, and chat log data. Infermedica API will be used to offer symptom options and an initial diagnosis based on current symptoms and general health of the patient. Amazon Comprehend Medical will be used to summarize the patient conversation which will be available to the physician prior to the appointment.

Regarding the backend, we chose to store the information retrieved from the user via the chatbot in DynamoDB. This is due to the large amounts of data we anticipate will need to be stored, and we chose AWS DynamoDB because much of our tech stack already used AWS, so it was more compatible. Our backend is connected to AWS using secret keys, and is able to read and write data into the database. We want to create endpoints for the frontend to call that will allow us to retrieve necessary information from the database and store necessary information with ease.

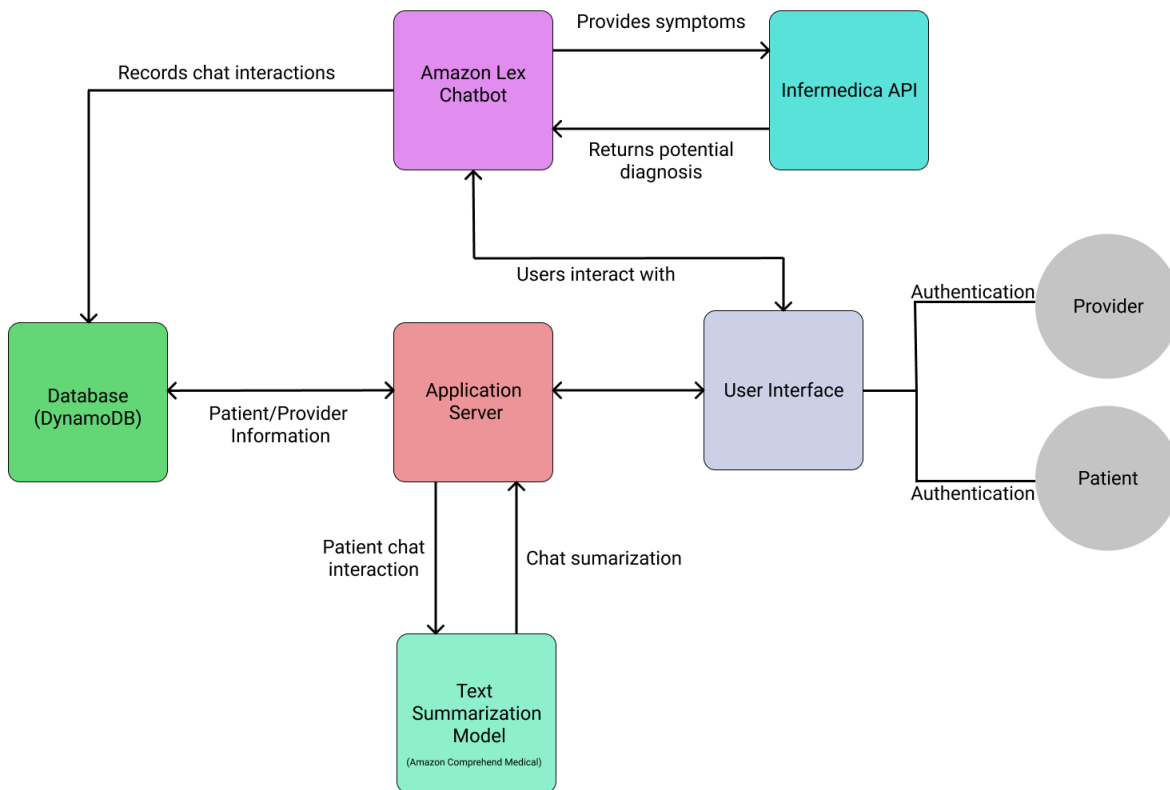
Assumptions

This application will only be available to patients and physicians who are already affiliated with Teladoc. There will be no method to create an account since patients will be using Teladoc's current resources to schedule appointments which will then be populated into our database. Given this assumption, we will assume that the patients table in the backend will be pre-populated with patient information including name, date

of birth, sex, pre-existing conditions, primary care physician, etc. Patients will be able to update some information in our interface such as conditions and physician but not others like date of birth.

Project Design

High-level System Architecture



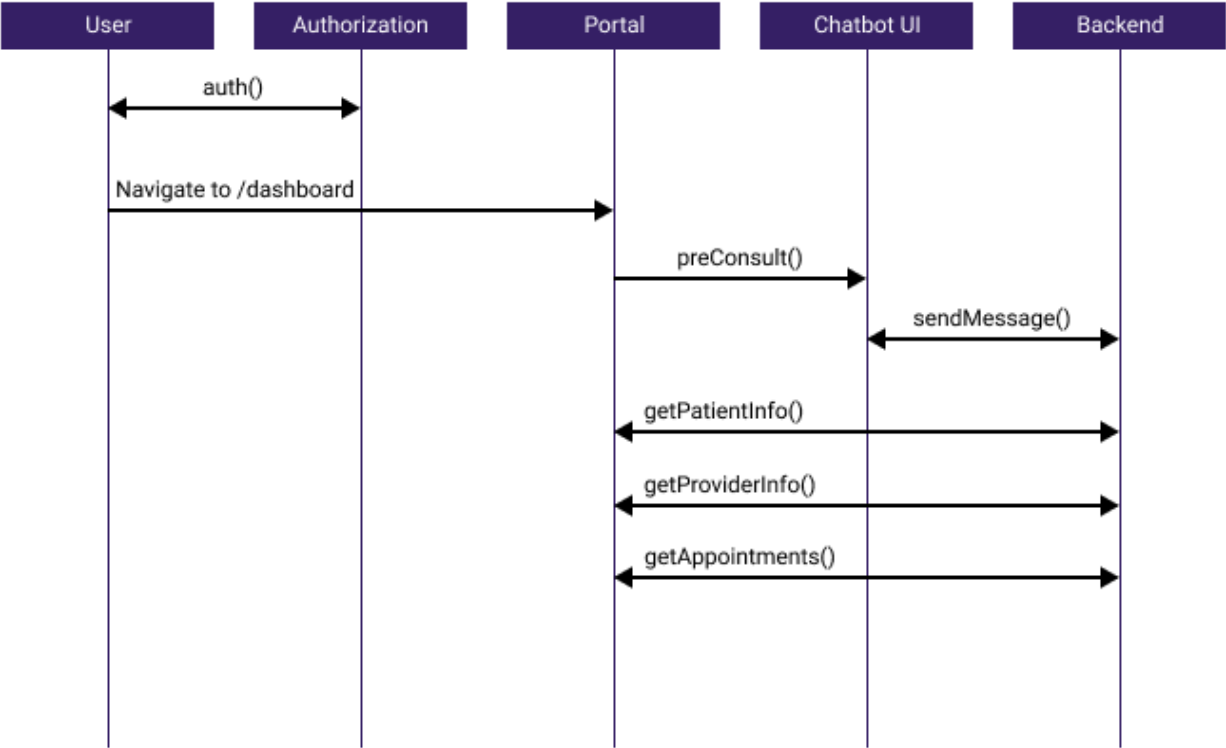
The diagram above represents the different parts of our web application and how they interact with each other. Patients and Providers will be able to access the application through a web user interface. After authentication, the frontend will be able to communicate with our backend REST API. The main 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 through an AWS lambda function to the Text Summarization Model: Amazon Comprehend Medical. All together these components form the high level architecture of our web application.

Sequence Diagrams

The following sequence diagrams depict architecture flows for frontend, backend, and chatbot, as well as user flows for patients and physicians.

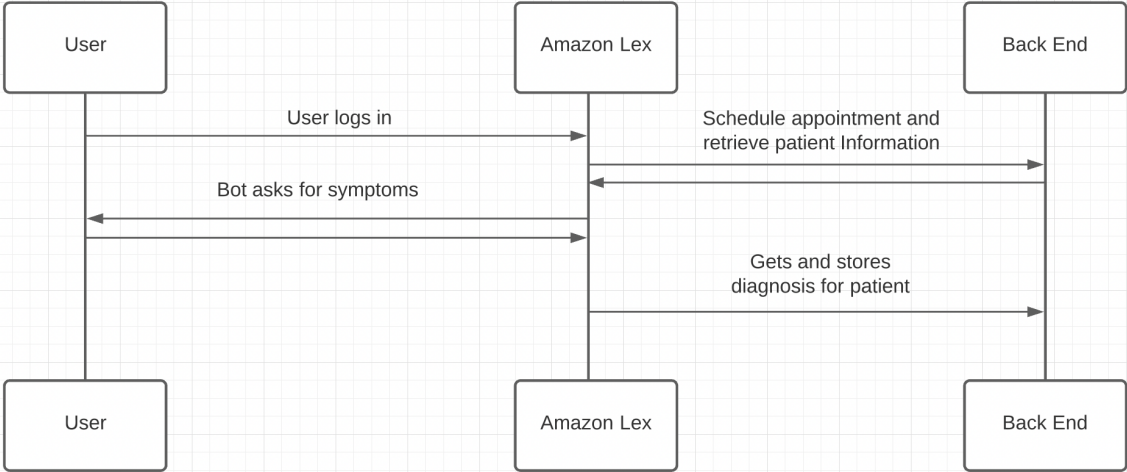
Frontend:

Frontend Sequence Diagram

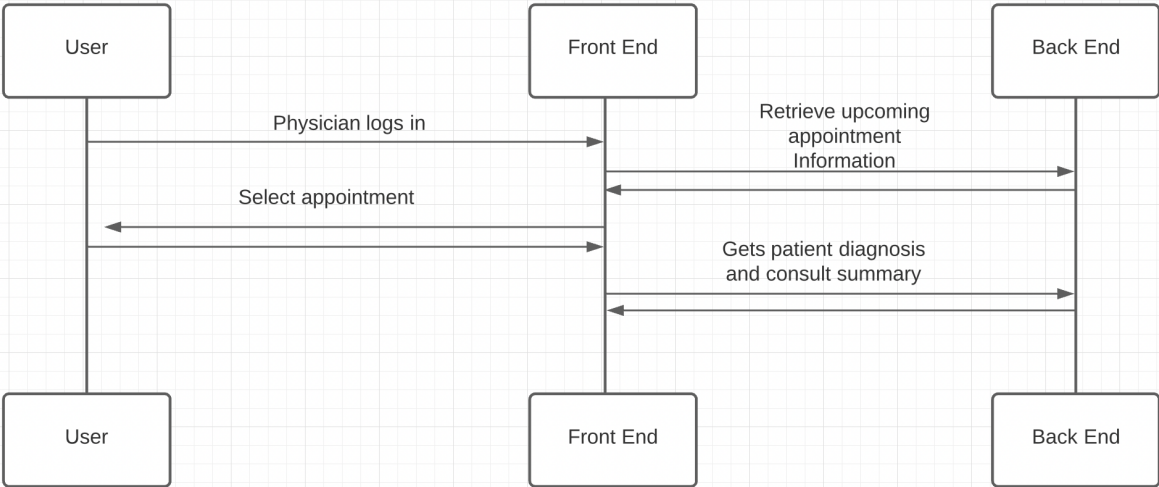


Backend:

Back End (Patient) Sequence

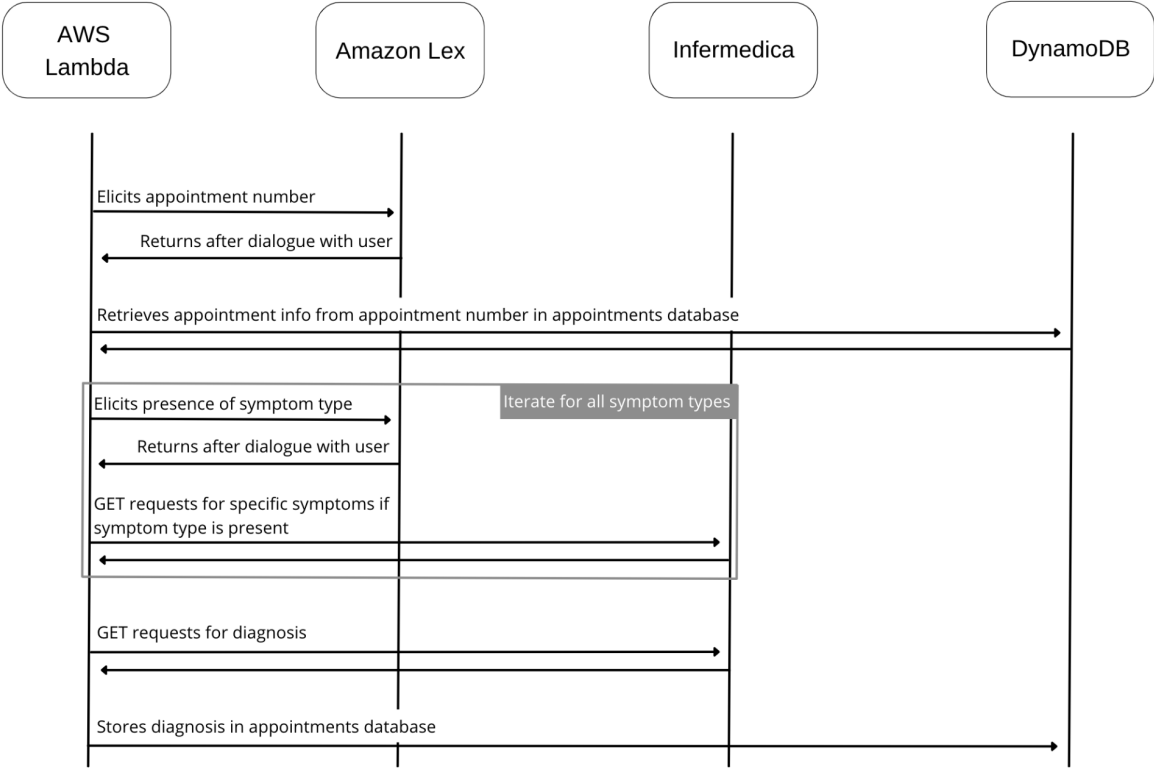


Back End (Physician) Sequence

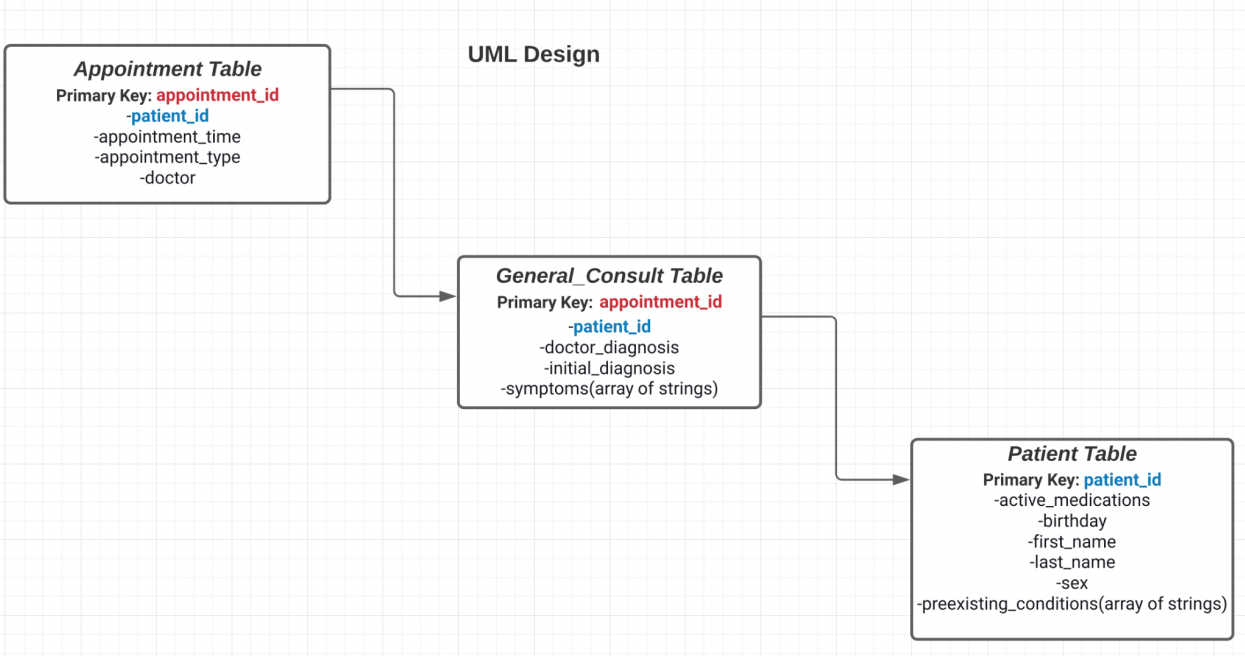


Chatbot:

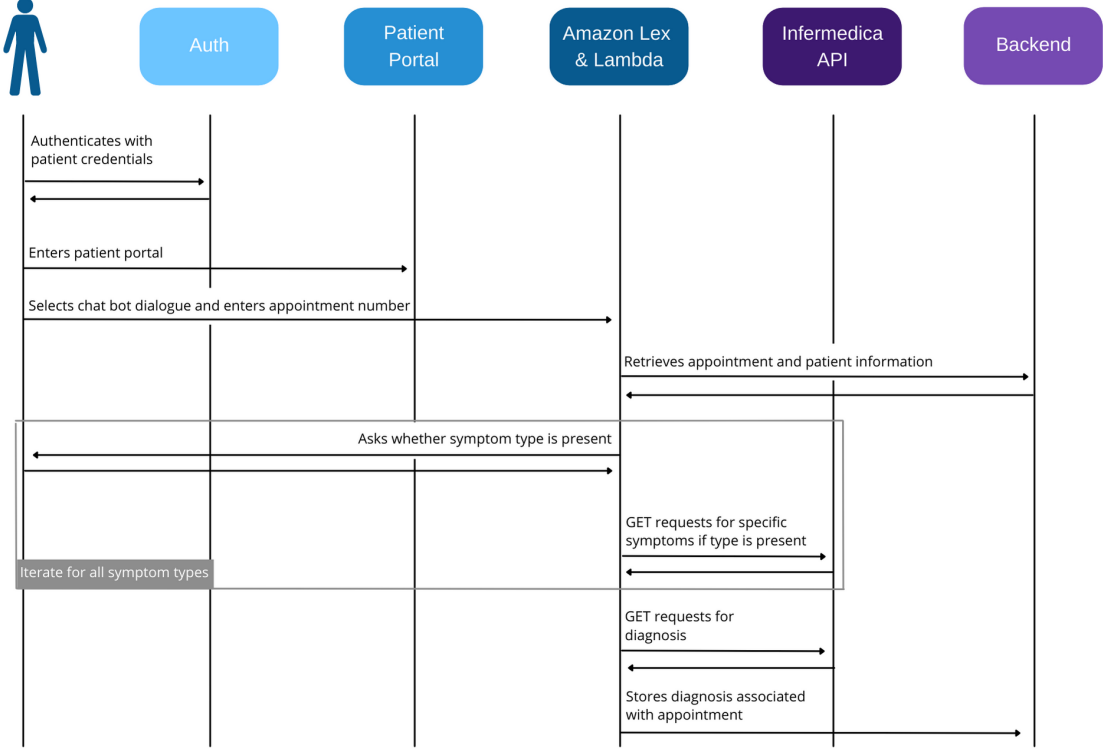
Chatbot Sequence Diagram



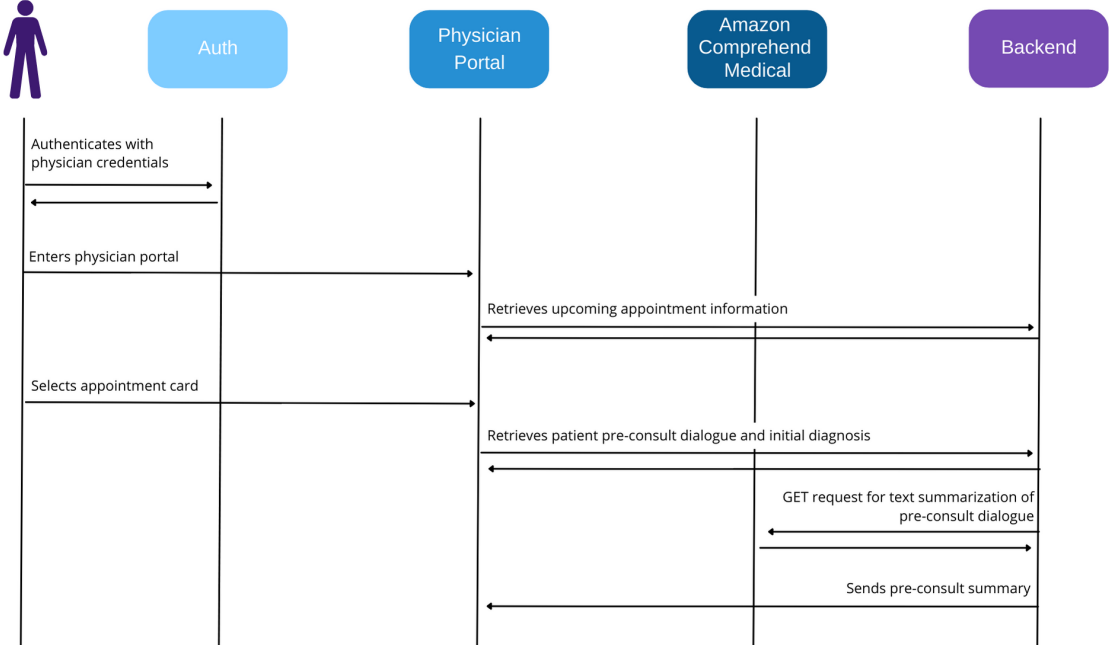
UML:



Patient Diagnosis Sequence Diagram

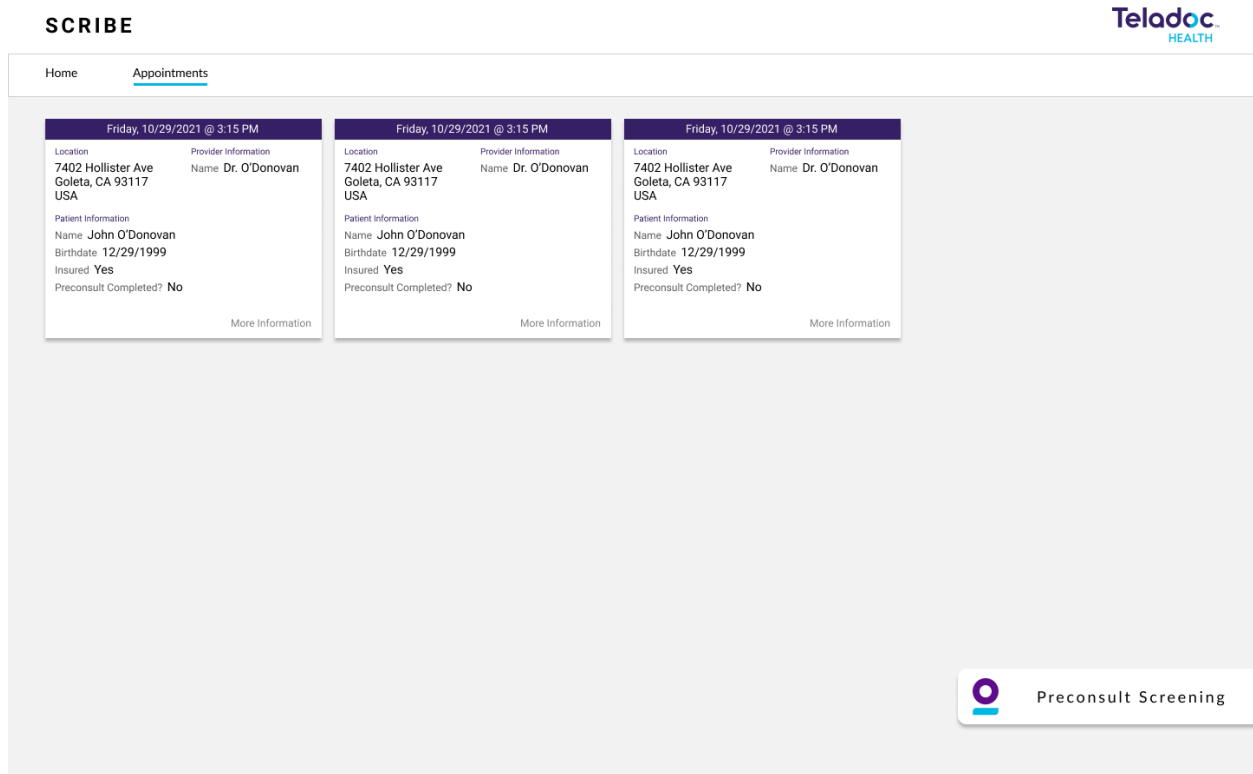


Physician Sequence Diagram



User Interface

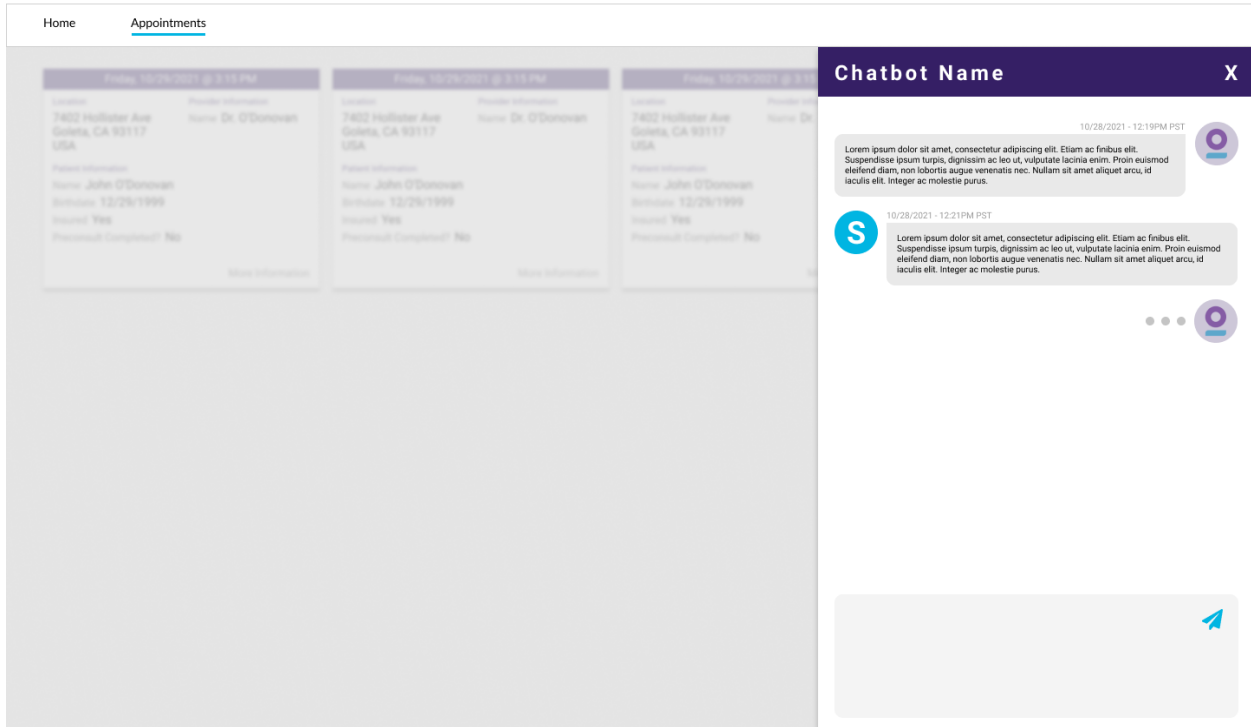
The primary interface for both patients and providers will be through a web application. Below are several screens that a user may see during the pre-consult screening flow.



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.

SCRIBE



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
<p>Location 7402 Hollister Ave Goleta, CA 93117 USA</p> <p>Provider Information Name Other Doctor</p> <p>Patient Information Name Patient 1 Birthdate 12/29/1999 Insured Yes Preconsult Completed? Yes</p> <p>Tags Physical New Patient Diabetes</p> <p>See Report</p>	<p>Location 7402 Hollister Ave Goleta, CA 93117 USA</p> <p>Provider Information Name Dr. O'Donovan</p> <p>Patient Information Name Patient 2 Birthdate 12/29/1999 Insured Yes Preconsult Completed? No Send Reminder</p> <p>See Report</p>	<p>Location 7402 Hollister Ave Goleta, CA 93117 USA</p> <p>Provider Information Name Dr. O'Donovan</p> <p>Patient Information Name Patient 3 Birthdate 12/29/1999 Insured Yes Preconsult Completed? No Send Reminder</p> <p>See Report</p>

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.

Appointment #31728541231

Patient Information

Name Patient 1

Birthdate 12/29/1999

Email example@email.com

Phone Number 858-222-2222

New Patient? Yes

Insurance Information

Provider UC SHIP

Group ID 382931231

CHAT LOGS

Chatbot Name 10/29/2021 - 12:19PM PST
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac fribus 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 laculis elit. Integer ac molestie purus.

Steven Huynh-Tran 10/28/2021 - 12:21PM PST
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac fribus 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 laculis elit. Integer ac molestie purus.

Steven Huynh-Tran 10/28/2021 - 12:21PM PST
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac fribus 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 laculis elit. Integer ac molestie purus.

Steven Huynh-Tran 10/28/2021 - 12:21PM PST
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac fribus 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 laculis elit. Integer ac molestie purus.

Chatbot Name 10/29/2021 - 12:19PM PST
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac fribus 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 laculis elit. Integer ac molestie purus.

Steven Huynh-Tran 10/28/2021 - 12:21PM PST
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac fribus 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 laculis elit. Integer ac molestie purus.

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) [HIGH PRIORITY] 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
- **Estimated Time:** 5 hours

2) [HIGH PRIORITY] 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
- **Estimated time:** 5 hours

3) [HIGH PRIORITY] As a Patient, I can see a list of my upcoming appointments on the homepage on the patient portal.

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

- Scenario 1: Patient has logged in and can look through future appointments
 - Patient is on main page which shows appointments in order of most upcoming appointment
- Scenario 2: Patient is not logged in
 - Patient is on the main page which only shows an introduction and a login button
- **Estimated time:** 10 hours

4) [HIGH PRIORITY] 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
- **Estimated time:** 15 hours

5) [MEDIUM PRIORITY] As a Patient, I can select an appointment in the patient portal to view details.

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

- Scenario 1: Patient selects an appointment for which a pre-consult has not been completed.
 - Patient is able to see the time and date of the consultation as well as the name of their consultation physician
- Scenario 2: Patient selects an appointment for which they have already completed the pre-consult.
 - Patient can see the time and data of the consultation as well as the name of their consultation physician
 - Patient can see a log of their pre-consult chatbot dialogue.
 - Patient can see a summary of their pre-consult chatbot dialogue.
- **Estimated time:** 2 hours

6) [HIGH PRIORITY] As a Patient, I can click on the chatbot icon at the bottom of the patient home screen.

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

- Scenario 1: Patient has yet to complete pre-consult and hovers over the bubble icon at the bottom of the screen
 - Clicking on the button opens the chatbot
 - Chatbot greets the user and pre-consult begins
- Scenario 2: Patient has already completed pre-consult and opens chatbot
 - Chatbot greets user
 - Informs user that pre-consult has already been completed
- **Estimated time:** 5 hours

7) [HIGH PRIORITY] As a Patient, I can interact with a chatbot that will ask me questions about my symptoms.

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
- **Estimated time:** 15 hours

8) [LOW PRIORITY] As a Patient, I can dynamically interact with a chatbot that will record and respond based on my concerns.

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

- Scenario 1: Patient does not have any concerns they'd like to share.
 - Chatbot asks the patient if they have any additional concerns about their consultation, the patient responds with "no"
- Scenario 2: Patient elects to alert the chatbot of their concerns.
 - Chatbot asks the patient if they have any additional concerns about their consultation, the patient responds with "yes"
 - Chatbot uses machine learning to dynamically respond to this conversation
 - Chatbot lets the physician know, and when the physician responds to these concerns, the patient will be alerted
- **Estimated time:** 30 hours

9) [MEDIUM PRIORITY] 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
- **Estimated time:** 5 hours

10) [MEDIUM PRIORITY] 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.

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
 - Three options of either seeing the summary, chat log, or initial diagnosis displayed
- **Estimated time:** 10 hours

11) [MEDIUM PRIORITY] As a Physician, I am able to see a summary of the conversation my patient had with the chatbot about their appointment with me.

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

- Scenario 1: Physician clicks on the summary option of the conversation element

- Physician is directed to a new page where the summary of the chatbot conversation is displayed
- Physician can view the summary created from patient's discussion with chatbot
- **Estimated time:** 5 hours

12) [MEDIUM PRIORITY] As a Physician, I am able to see a complete chat log of the conversation my patient had with the chatbot about their appointment with me.

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

- Scenario 1: Physician clicks on the chat log option of the conversation element
 - Physician is directed to a new page where the chat log of the chatbot conversation is displayed
 - Physician can view the chat log of the patient's discussion with chatbot
- **Estimated time:** 5 hours

13) [HIGH PRIORITY] As a Physician, I am able to see an initial diagnosis prescribed by the chatbot based on the user's chat messages.

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

- Scenario 1: If the patient has completed their pre-consult, the physician can view an initial diagnosis on the appointment card associated with the pre-consult.
- Scenario 2: If the patient has not completed their pre-consult, the physician sees that the pre-consult has not yet been completed.
 - The physician can also send a reminder to the corresponding patient's portal to complete the pre-consult before their appointment.
- **Estimated time:** 3 hours

14) [MEDIUM PRIORITY] As a 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: 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
- Scenario 2: If the patient has not completed pre-consult, summary is not displayed.
- **Estimated time:** 12 hours

15) [SUPER LOW PRIORITY] As a Patient, I can complete symptom checking dialogues for different types of consults.

Github Issue: <https://github.com/ishana-narayanan/404-team-name-not-found/issues/11>

- Scenario 1: Patient has an appointment for a general consult.
 - Patient completes a pre-consult dialogue with identification of general symptoms, such as headache, cough, and fever.
- Scenario 2: Patient has an appointment for a mental health consult.
 - Patient completes a pre-consult dialogue with identification of mental health symptoms, such as depression or mood swings.
- Scenario 3: Patient has an appointment for a specialty consult.
 - Patient completes a pre-consult dialogue catered to a specific injury, such as a jammed finger or a muscle sprain.
- **Estimated time:** 7 hours

16) [LOW PRIORITY] As a Patient, I am able to visit a new consult notes page of my previous consultations.

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
- **Estimated time:** 5 hours

17) [LOW PRIORITY] 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 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 2: Patient clicks on the close button
 - Popup closes
 - Patient returns to the consultation notes page
- **Estimated time:** 4 hours

18) [HIGH PRIORITY] As a Patient, I am able to logout and be redirected to the authorization page.

GitHub Issue: <https://github.com/ishana-narayanan/404-team-name-not-found/issues/9>

- Scenario 1: Patient logs out from the patient portal.
 - Patient clicks on log-out button in the main patient page portal once they're done
- Scenario 2: Patient is redirected to the patient portal page and can log in again.
 - Patient can log in again on the patient portal if need be
- **Estimated time:** 5 hours

19) [HIGH PRIORITY] As a Physician, I am able to logout and be redirected to the authorization page.

GitHub Issue: <https://github.com/ishana-narayanan/404-team-name-not-found/issues/9>

- Scenario 1: Physician logs out from the physician portal.
 - Physician clicks on log-out button in the main physician page portal once they're done
- Scenario 2: Physician is redirected to the physician portal page and can log in again.
 - Physician can log in again on the physician portal if need be.
- **Estimated time:** 5 hours

20) [LOW PRIORITY] As a Patient, I can update personal medical information inside of the patient portal.

GitHub Issue: <https://github.com/ishana-narayanan/404-team-name-not-found/issues/8>

- Scenario 1: Patient clicks on Personal Information
 - Redirect to information page
- Scenario 2: Patient updates text boxes per each individual attribute
 - Patient's information is temporarily saved
- Scenario 3: Patient clicks save
 - Information is saved and updated on database
 - New information will be used in future chatbot interactions
- **Estimated time:** 10 hours

Appendix

Tech Stack

- **Amazon Lex:** chatbot implementation
- **React:** frontend elements including webpages and authentication
- **NodeJS:** backend to communicate with database
- **DynamoDB:** to store patient and appointment information, symptom screening/initial diagnosis, and chat logs
- **Infermedica:** symptom tracking and diagnosis recommendation
- **Amazon Comprehend Medical:** chat log text analysis

Links

- **Github:** <https://github.com/ishana-narayanan/404-team-name-not-found>
- **Google Docs:**
<https://drive.google.com/drive/folders/1blQPC1LdTeHdadua9E7ZgSxZ0h4KWTSD?usp=sharing>
- **Trello:** <https://trello.com/404teamnamenotfound7>