Project Requirements Document, Version 1

Project: Provider Augmentation for Behavioral Health Analysis
Team Name: Interlinked
Authors:

- Brian Lim (blim@ucsb.edu)
- Felix Zhao (fzhao@ucsb.edu)
- Diego Perez (diego03@ucsb.edu)
- Gehrig Weber (gehrigweber@ucsb.edu)
- Michael Zhou (yihang_zhou@ucsb.edu)

Intro

Doctors require augmentations/assistance in making accurate behavioral diagnoses, since mental health is complex to analyze. We believe software can enhance doctors' capabilities by providing more information to them. There seems to be a lack of ways for doctors to get extra information about mental health, unlike using lab tests or medical equipment for gathering information on physical health like weight, hormone levels, blood pressure, etc. Additionally, mental health is a fast-growing field that has huge demand and is over-taxing doctors today, leading many experts to call it the largest epidemic affecting humanity. Doctors need ways to handle treatment of the growing number of patients, especially since there are so few specialists in the behavioral health sphere. Remote areas also often lack access to mental health services due to high cost, lack of availability, and stigma against mental health treatment. Therefore there needs to be a tool that allows behavioral health specialists to connect to patients remotely to avoid losing time to travel and also provide additional information on the patient that might not be easy to determine in a conversation across a webcam.

Our objective is to provide that tool that will enhance the diagnostic capabilities of doctors concerning mental health, as well as possibly improving the accessibility of these health services. We are aiming to provide a software tool that streams audio / video between patient and doctor. The software interface would provide results of sentiment analysis of the patient's facial expressions and speech to the doctor. The company we work with, InTouch Health, is providing machine learning models for performing sentiment analysis, but it is possible for us to work on improving the accuracy / diversity of statistics the models provide. Since we are dealing with sensitive patient information, we will also need to secure the data to an appropriate degree. We will also store relevant data to allow doctors to make use of historical data and analysis.

Stretch goals would include enhancements of these features so far discussed, as well as improving the accuracy and modifying the machine learning models that provide sentiment analysis. Adding features or creating models for dealing with nuances in human speech and expressions like sarcasm are also far off possibilities.

We are innovating by creating a communication channel and data analysis for the doctor to better work with their patients. We are increasing the ease of interaction between doctor and patient, first of all. Second, we are providing information that is not often gathered for use in behavioral health analysis using sentiment analysis by machine learning models. Our tool provides a way for doctors to get statistics they do not get in person or normally as well as reanalyze previous sessions with patients so that valuable information is not lost.

We do not expect to require devices more complicated than normal computers with webcams most ideally (that is, no proprietary hardware is expected). We also do not expect this tool to be an extensive tool for data analytics, as it should simply provide information. The information gathered should solely be based off the audio and video provided by the patient. No other physical hardware to gather data like heartbeat or blood pressure would be used. We also assume that data from these sources is suitable enough to assist doctors in treating behavioral health. We assume that doctors want real time and post processing of data. We assume doctors generally should be looking at the sentiment analysis during these sessions, but patients can possibly also be shown the sentiment analysis. Patients should be able to look at some of their own data. We assume that patients can legally agree to all the services that our software should provide.

We have decided on the main technical stack of our application. Our frontend will consist of a ReactJS application using Material UI, because those simple enough to build applications with and are powerful enough to give us agency over how we design the UI to fit the needs of doctors and patients. This ReactJS app will connect to a Golang REST API router designed using the Gin framework. Gin is a lightweight framework designed to create REST endpoints that a ReactJS app can connect to and pull information from. This router will perform the computations associated with sentiment modeling and be able to pull data from a database containing patient and provider data. The REST API will connect to a MongoDB database which will store that data. We opted to use a NoSQL database because the interface does not have an emphasis on the relationship between entities, but rather a need to store data flexibly.

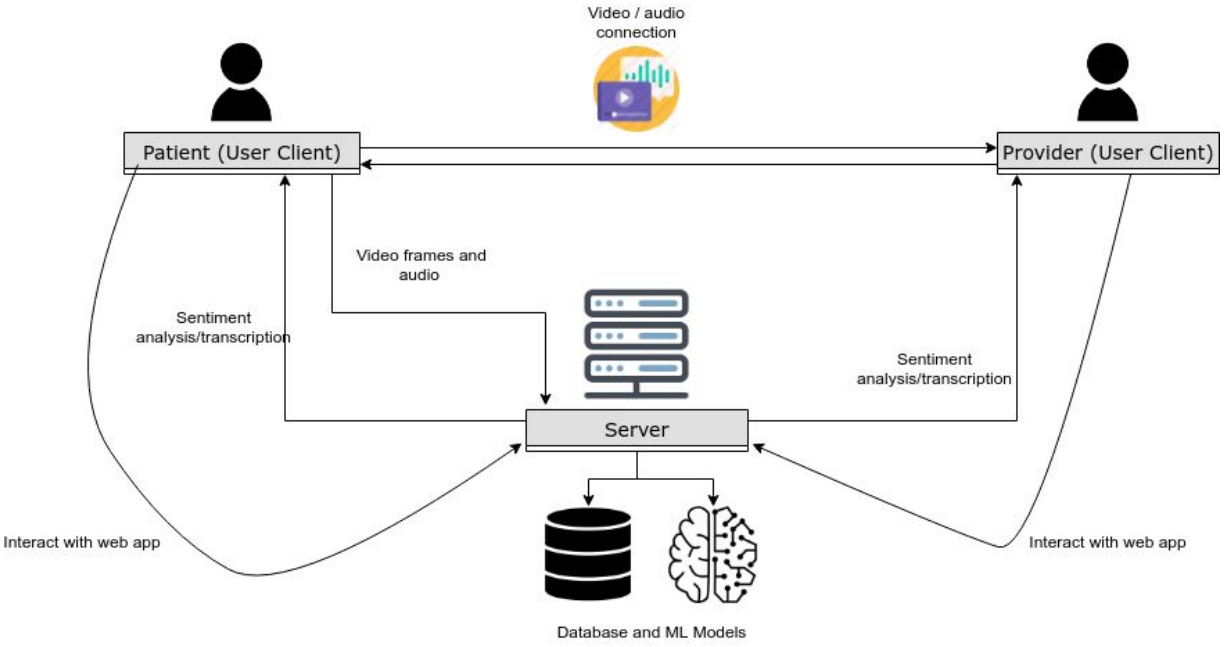
As a team, our main goal is to have a project built from start to finish, from backend to frontend. Along with building a project, we hope to improve each of our interteam communication skills. This project requires skills in front end, back end, and machine learning for sentiment analysis. Provider Augmentation is a project that employs data visualization through getting data using machine learning, so those of us

with a machine learning background will understand and perhaps improve the models that we are provided. The challenge we will be facing is designing a user interface that will be intuitive for providers and patients as they use the system, as it is essential to communicate information to providers clearly for critical mental health analysis.

There are not many other apps in competition with the one we are building. Some provide a platform for therapists to manage their notes on sessions with patients, which is a great idea / extension for our app to implement in the far future. Other apps deal with Telehealth (software for remote access to healthcare) by having secure channels for providers and patients to communicate. None of these apps provide real-time sentiment analysis that we are looking to build in our project. The innovation in our project will be the quality of our machine learning models providing sentiment analysis in real time, and hopefully in post processing as well. We still want to build similar features such as secure remote communication (maybe notes for providers as well), while enhancing these tools with insights of data analysis.

In conclusion, our team is attempting to build an application to increase the accessibility and power of behavioral health services by creating an interface for doctors and patients to interact, and doctors can use this application to view non-obvious signals from a patient.

System Architecture



Requirements

As a user, I can join a session and see a page with UI so that I can interact with another user.

Github links (issues):

https://github.com/BLimmie/intouch-health-capstone-2019/issues/2

https://github.com/BLimmie/intouch-health-capstone-2019/issues/1

- Scenario 1: Given a user clicks a link to join session without another user
    - When user clicks the link and sees the session page
    - Then they will see a black screen until the other user joins
- Scenario 2: Given a user clicks a link to join session with another user present
    - When user clicks the link and sees the session page
    - Then they will see the other user is in the session and extra information available
- Scenario 3: Given the user is not authorized to join
    - When user clicks the link and tries to join session
    - Then the browser displays a connection refused message

As a user, I can see the other user's video feed, so that I can communicate with them effectively.

Github links (issues):

https://github.com/BLimmie/intouch-health-capstone-2019/issues/3

- Scenario 1: Video is successfully received
    - Given the patient and the physician are in a session
    - When a user looks at the region of the screen that should have the other user's video feed
    - Then the user will see the other user's video feed
- Scenario 2: Video is not received successfully
    - Given the patient and the physician are in a session
    - When the user looks at the region of the screen that should have the other user's video feed
    - Then the user will see an error image in that region

As a user, I can hear the other user's audio, so that I can communicate with them effectively.

Github links (issues):

https://github.com/BLimmie/intouch-health-capstone-2019/issues/4

- Scenario 1: Audio is successfully received

    - Given the patient and the physician are in a session
    - When the user attempts to hear the other user
    - Then the user will hear them

As a user, I can quit a session.
Github links (issues):
https://github.com/BLimmie/intouch-health-capstone-2019/issues/5

- Scenario 1: There exists another user in the room once the user leaves
    - Given that a user clicks the quit a session button or exits the page
    - And another user stays on the session
    - When the user sees the other's video feed
    - Then the user should see a black screen a
    - And be notified that the user left
- Scenario 2: No users remain in the room, session is closed
    - Given that both users click the quit a session button or exits the page
    - When the server is notified that both users left
    - Then the session is ended and no user can join that session again

As a physician, I can see the emotion levels of my patient's face at the moment, so that I can more accurately diagnose my patient.
Github links (issues):
https://github.com/BLimmie/intouch-health-capstone-2019/issues/7
https://github.com/BLimmie/intouch-health-capstone-2019/issues/9

- Scenario 1: Emotion levels are accurate
    - Given that the physician and patient are in a session
    - And the video can be seen
    - When the patient changes expression
    - Then the physician will be able to see the patient's emotion levels, based on their appearance

As a physician, I can see a transcription of my patient's speech, color coded according to sentiment, so that I can more accurately diagnose my patient.
Github link (prototype):
https://github.com/BLimmie/intouch-health-capstone-2019/commit/8904bb11c72f2a59f16d13bbe77e5ed2b5de006b
Github link (tests):
https://github.com/BLimmie/intouch-health-capstone-2019/commit/cd7d0932b22d77fbca57db3e24da2e96347d1977

- Scenario 1: Audio transcription and color-coding are accurate
    - Given that the physician and patient are in a session
    - And the audio can be heard
    - And the patient has said something since the session started
    - When the patient says something
    - Then the physician will be able to see an accurate transcription of what the patient just said, with key words indicating sentiment color-coded according to which emotion they indicate

<u>As a physician, I can create and log into account with the web app, so that I can start or join sessions with my patients (without creating all the requests manually).</u>

- Scenario 1: No logged in account, no account.
    - Given that the physician hasn't logged into any account and has no account.
    - When the physician tries to log in.
    - A login interface is shown to the physician. It also shows a portal to where the physician is able to create an account. The physician goes through the portal.
    - And the physician starts to use the account creation interface, insert and submit information.
    - And the physician has an account after inserting valid and complete information. They are then shown the login interface to login again.
    - Same as Scenario 2.
- Scenario 2: No logged in account, has account.
    - Given that the physician hasn't logged into any account but has at least one account.
    - When the physician tries to log in.
    - The physician is shown the login interface.
    - Then the physician inputs valid credentials. Server checks the credentials. If credentials good the interface for starting/joining a session is created.
    - Same as Scenario 3, skipping validation of credentials just cached.
- Scenario 3: Logged in 1 account.
    - Given that the physician has logged into one account.
    - When the physician tries to log in.
    - An interface allowing the physician to start/join a session is shown.
    - When the physician chooses to start a session, start a session. When they choose to join a session, show an error when there is no session. Join the session when there is one.
    - An interface of an ongoing session is shown.

As a patient, I can create and log into an account, so that I can have sessions with my physicians, along with accessing information on past sessions.

- Scenario 1: Patient does not have account
    - Given the patient does not already have an account
    - When the patient enters their information and clicks on the “Log In” button
    - Then they will not be able to log in, because they don’t have an account
    - When the patient clicks on the “Create Account” button
    - Then they will be taken to a screen where they can enter their information to create an account
    - And they will be able to log into this account in the future
- Scenario 2: Patient does have account
    - Given the patient already has an account
    - When the patient enters their information and clicks on the “Log In” button
    - Then they will be able to log in to their account
    - And they will be able to join a session with a physician
    - As well as access information on past sessions

As a physician, I can access any and all information (except audio and video) from my past sessions with my patients after logging into my account, so that I can review a patient’s history to assist with diagnoses and future sessions.

- Scenario 1: Physician requests the data of a patient in the system
    - Given the physician wants the personal data collected on a patient in the system
    - And they are asking for the data of that patient
    - When they request that data
    - Then they get a formatted document of that patient’s past sessions and personal information
- Scenario 2: Physician requests the data of a patient not in the system
    - Given the physician wants the personal data collected on a patient not in the system
    - And they are asking for the data of that patient
    - When they request that data
    - Then they will get an error message
- Scenario 3: Physician requests the data of a patient that isn’t linked to them
    - Given the physician wants the personal data of a person that is not their patient
    - And they are asking for the data of that patient

    - When they request the data
    - Then they get an error message
    - And not get the data

As a physician, I can request that a patient's account be linked with my account, so I can properly store future sessions as well as retrieve their past sessions and available medical records.

- Scenario 1: Physician and patient accounts are not linked
    - Given the physician and patient accounts exist
    - And they are not already linked
    - When the physician enters a patient's information
    - And clicks the "Link Account" button
    - Then the two accounts will be linked
    - And the physician can properly store future sessions
    - And retrieve the patient's past sessions and available medical records
- Scenario 2: Physician and patient accounts are already linked
    - Given the physician and patient accounts exist
    - And they are already linked
    - When the physician enters a patient's information
    - And clicks the "Link Account" button
    - Then an error message will appear
    - And the two accounts will still be linked
- Scenario 3: Patient account does not exist
    - Given the physician's account exists
    - And the patient's account does not exist
    - When the physician enters a patient's information
    - And clicks the "Link Account" button
    - Then an error message will appear
    - And the physician's account will not be linked to any new accounts

Appendices

Technologies Employed

React: Frontend elements, including video streaming capabilities, powered by Twilio
Gin / Golang: Backend REST API router
Material-UI: Frontend elements, excluding audiovisual capabilities
MongoDB: NoSQL Database for storing user info and sessions

OpenFace: Library that extracts information from faces
TypeScript/JavaScript: Frontend elements, basis for React and Material-UI