

EZ MedChat


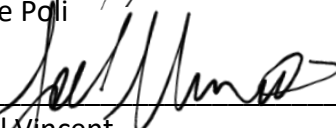


by

Jake Poli, Joel Vincent, and Yisroel Weisberg

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

© Copyright Jake Poli, Joel Vincent, Yisroel Weisberg

The author grants to the School of Information Technology permission
to reproduce and distribute copies of this document in whole or in part.

 _____ Jake Poli	<u>April 12, 2020</u> Date
 _____ Joel Vincent	<u>April 12, 2020</u> Date
 _____ Yisroel Weisberg	<u>April 12, 2020</u> Date
 _____ Abdou Fall, Faculty Advisor	<u>April 12, 2020</u> Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services

April 2021



TABLE OF CONTENTS

ABSTRACT	1
1 INTRODUCTION	2
1.1 Introduction.....	2
1.2 Problem	2
1.3 Solution.....	3
1.4 Goals	4
1.5 Overview	4
2 DISCUSSION.....	5
2.1 Project Concept.....	5
2.2 Design Objectives	5
2.3 Methodology and Technical Approach	6
2.4 User Profiles.....	12
2.5 Use Case Diagrams	14
2.6 Technical Architecture	16
2.7 Testing.....	17
2.8 Budget.....	23
2.9 Problems Encountered and Analysis of Problems Solved.....	23
2.10 Recommendations for Improvement	24
3 CONCLUSION.....	25
3.1 Lessons Learned.....	25
3.2 Abilities and Skills Developed throughout the Project	26
3.3 Completed During Spring	26
4 REFERENCES	27
4.1 References.....	27
5 APPENDIX.....	28
5.1 Appendix A.....	28

LIST OF FIGURES

Figure 1: Home Page	7
Figure 2: Registration page.....	8
Figure 3: login page.....	8
Figure 4: appointment page	9
Figure 5: lobby page	9
Figure 6: lobby page	10
Figure 7: video call page (practitioner).....	10
Figure 8: view patients page	11
Figure 9: lobby page (patients)	11
Figure 10: Appointments page (patients).....	12
Figure 11: video call page (patients).....	12
Figure 12: medical practitioner use case diagram.....	15
Figure 13: Patient use case diagram	16

LIST OF TABLES

Table 1: user profile for medical practitioners.....	13
Table 2: user profile for patients.....	14
Table 3: Budget	23

ABSTRACT

According to the study by *Annals of Family Medicine*, a primary care physician has around 2,300 patients which fills up most of the doctor's typical day. EZ MedChat is a website for doctors, staff, and patients to easily connect and provide exceptional care in a more efficient way. The solution includes creating custom medical chart templates to help streamline each patient visit, providing real-time data to both doctor and patient to improve communication, and creating a virtual waiting room for patients who need quick visits without appointments. EZ MedChat can save precious time for both the doctors and patients and provide better outcomes.

1 INTRODUCTION

1.1 INTRODUCTION

Due to scheduling and other time issues, many people have a difficult time scheduling an appointment with their primary care physician or other medical professionals. There are many online solutions available that offer teleconferencing along with other features, but none that provide a live status of the medical professional that states whether they are free or not. If the medical professional gets free time, they have no way of knowing what patients would like to meet with them immediately. After consulting with a few doctors, we came to discover the problem doctors are facing and developed our unique solution to address it.

1.2 PROBLEM

Getting an appointment with a doctor can be time consuming and immediate appointments may not be available. This is primarily due to the American healthcare system and related to Medicare and Insurance. According to the article in a 2012 issue of the *Annals of Family Medicine*, it was noted that the average primary-care physician has about 2,300 patients under their care. Patients are getting to be older, which means more doctors' visits, and doctors are retiring, causing the doctor-to-patient ratio to expand, stated by AAMC in the article *U.S. Physician shortage growing*. Medicare and Insurance are affecting doctors as well because this adds to the already massive amount of paperwork they have to do. In this climate

with the global pandemic, getting a doctor's appointment can be very difficult. A Patient may not be able to see a doctor in weeks or months.

Traditionally to set up an appointment, the patient would have to call the doctor's office to see what appointment availability there was. After the patient sets up an appointment, there is no easy way to determine if a time slot became available in the near future. When the patient arrives at the doctor's office for the appointment, there is a good chance they will be waiting in the lobby, and the doctor's going to be several minutes late. If the patient took time off from work, this could be a hassle. During the meeting, the patient may feel rushed, and may not feel like they had enough time. The doctor can also feel rushed at times to get to their next appointment, since they have to fill out the medical charts after the appointment.

1.3 SOLUTION

EZ MedChat will provide flexibility and convenience for both doctors and patients. As we stated in the Problem Statement, many doctors have packed schedules every day and there is a possibility that a patient might be required to wait a few weeks before getting an appointment. EZ MedChat will have a waiting room feature that will allow a patient to enter that virtual waiting room and if the doctor has a free block of time, they can choose to see one of the patients in the waiting room before their next scheduled appointment. This will solve the patient's problem of being unable to wait for an official appointment by using the waiting room to see a doctor faster and allow the doctor to make some extra money.

The other unique feature that EZ MedChat will provide will improve a doctor's workflow. EZ MedChat will implement the ability for a doctor to create and modify medical chart templates. This will allow the doctor to narrow down the fields that he actually needs for

each visit. For example, if it is a general checkup the doctor could create a template for a standard visit, which would streamline the visit because only the relevant part of the form is showing. This would work in tandem with the waiting room solution by helping free up more of the doctor's time, which would allow for more patients to be seen overall.

1.4 GOALS

The goals for EZ MedChat are creating an easy-to-use, streamline, medical experience. This will include a web portal for the patients and the doctors. They will be able to meet through an e-visit style of appointment. Using the feature of custom templates that will be designed for EZ MedChat, as well as the ability to use a virtual waiting room, we believe that EZ MedChat will save time for both patients and doctors.

1.5 OVERVIEW

The remainder of this final report outlines in detail how the project was completed. The report includes the following sections: design objectives, methodology, budget, timeline, problems encountered, and future recommendations.

2 DISCUSSION

2.1 PROJECT CONCEPT

The concept of EZ MedChat was from a second rendition of our original idea. The original idea was to create a Physical Therapist application that would be used between the Physical Therapist and the client. We ended up dropping that idea because it was already an active application but stuck with the medical theme. The team eventually came up with the idea of EZ MedChat after speaking with some doctors. The doctors outlined issues they face with current telehealth applications, specifically highlighting the lack of proper medical chart templating options and the complaints they receive about long waits to get an appointment. We wanted to create a way to help streamline this meeting process and save time for both the doctors and patients. Our solution was through the implementation of form templates for doctors and the waiting room for patients.

2.2 DESIGN OBJECTIVES

Our team's design objectives have evolved throughout the project. Initially, we wanted to separate our overall application into two parts. One, a web application for doctors to use at the office, and two, a mobile application for patients to facilitate ease of use. We also wanted to have our application have the complete functionality of other tele-health applications. i.e., messaging, integration with EMR systems, medication refilling, etc. However, after running into some technical difficulties we realized that the initial scope was too large, and the application must be scaled down to be finished on time. Therefore, the mobile application was dropped, as

well as several non-core features, like the ability to request prescription refills and the integration with Epic.

We wanted to design an easy-to-use application that streamlines doctors workflow and allows them to see more patients which would give patients the ability to see their doctors more easily.

2.3 METHODOLOGY AND TECHNICAL APPROACH

Doctor Side of Web Application

We decided to separate our web application into two parts, one side for doctors and other health professionals, and another side for patients. We decided it would be best to use .NET Core to build the web application due to its cross-platform compatibility, useful built-in functionality like its built-in dependency injection, and powerful libraries like SignalR, which we used for real-time client server communication. To host our application, we chose Azure because it is HIPPA compliant and as another Microsoft platform it is easy to integrate with the application.

- The website includes these main pages:
 - “Home” - A simple welcome screen that includes some details about the application.

Welcome to EZ Medchat

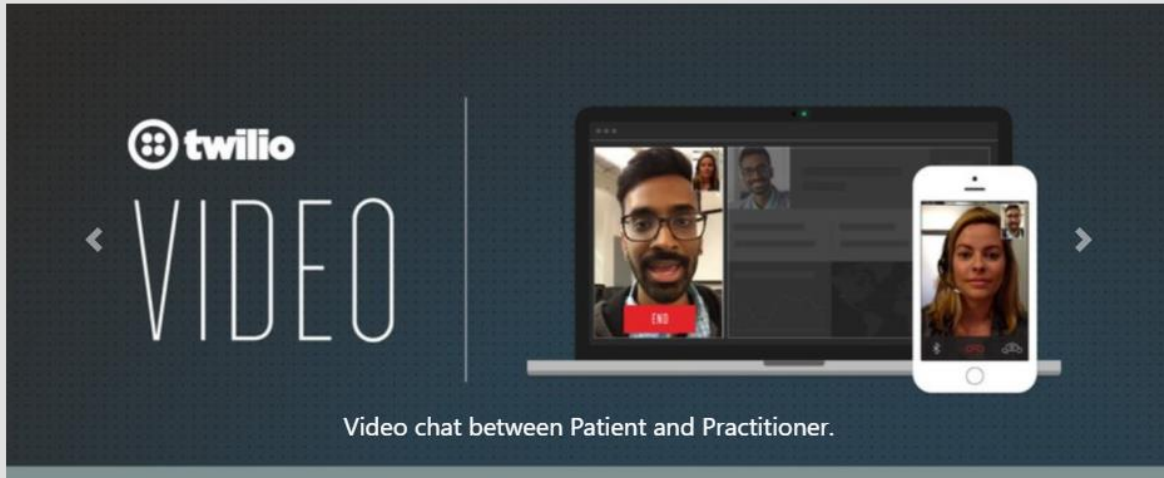


FIGURE 1: HOME PAGE

- “Login / Register” - Healthcare professionals can use this tab to sign into the web app.
 - If they do not have an account, they have the option to sign up.

EZ MedChat Register Login

Register

Create a new account.

Email Address

First Name

Last Name

Password

FIGURE 2: REGISTRATION PAGE

EZ MedChat Register Login

Log In

Use a local account to log in.

Email

Password

[Register as a new user?](#)

FIGURE 3: LOGIN PAGE

- Once Signed in:
 - Appointments - to create or view/edit/delete current appointments.

EZ MedChat Video Call Lobby Template Appointment Patients Hello polijji@gmail.com! Log out

Appointments

Start Time	End Time	Appointment Reason	Appointment Instructions	
2021-03-20 09:35:00 PM	2021-03-20 09:45: PM	BackPain	Join Lobby at 7.30	Edit Details Patient Info. Delete
2021-04-07 10:51:00 PM	2021-04-07 11:51: PM	Regular Visit	Join Lobby at Time	Edit Details Patient Info. Delete
2021-04-10 07:16:00 PM	2021-04-10 07:36: PM	Regular Visit	Join lobby at 7	Edit Details Patient Info. Delete
2021-04-11 07:22:00 PM	2021-04-11 07:53: PM	Regular Visit	Join Lobby at Time	Edit Details Patient Info. Delete

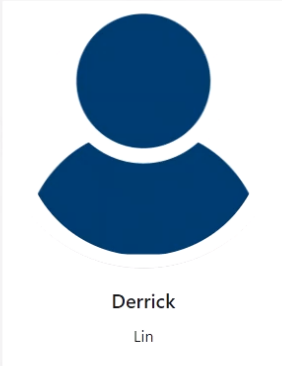
[Create New](#)

FIGURE 4: APPOINTMENT PAGE

- Lobby - to appear online, and view patients that joined the lobby.

EZ MedChat Video Call Lobby Template Appointment Patients Hello jdoe@hotmail.com! Log out

Toggle this switch to go on or off-line



Derrick
Lin

FIGURE 5: LOBBY PAGE

- Templates - to create custom medical templates.

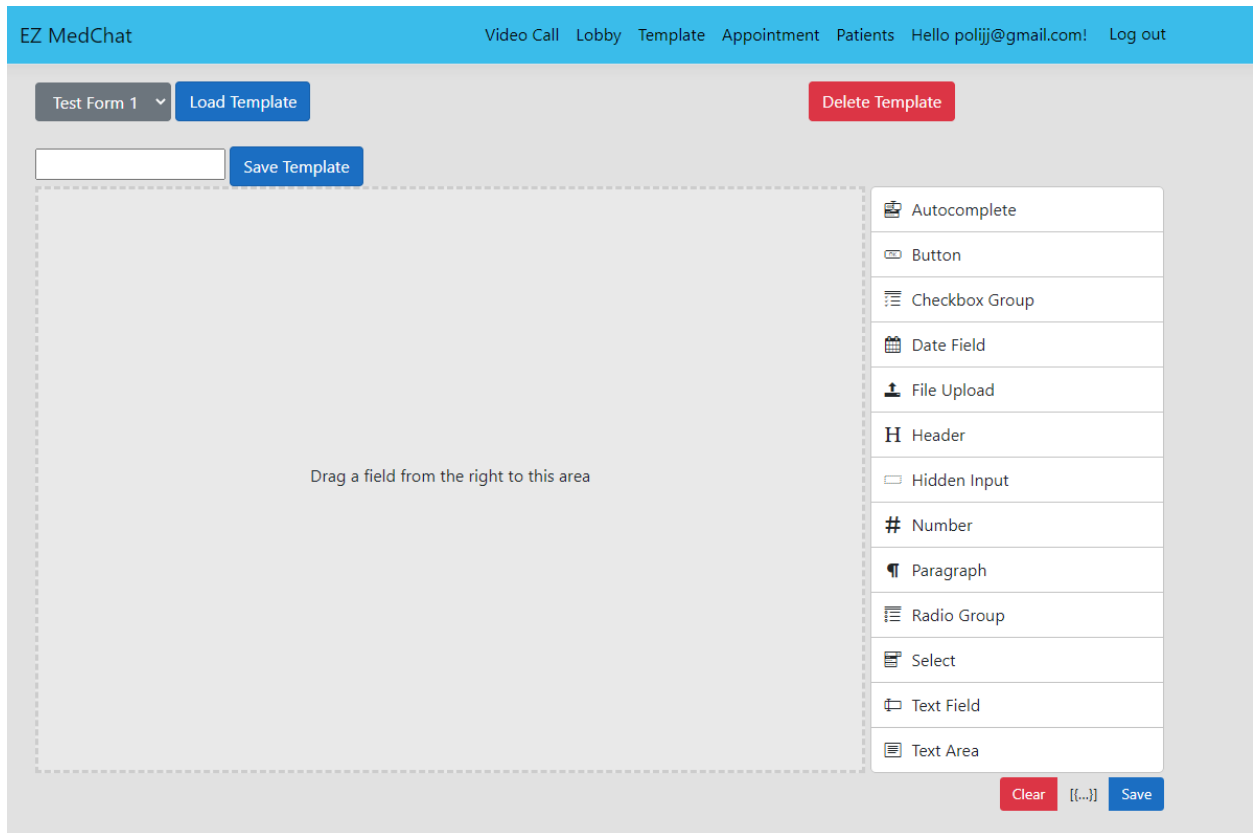


FIGURE 6: LOBBY PAGE

- Video Call - to create a video call with the patients

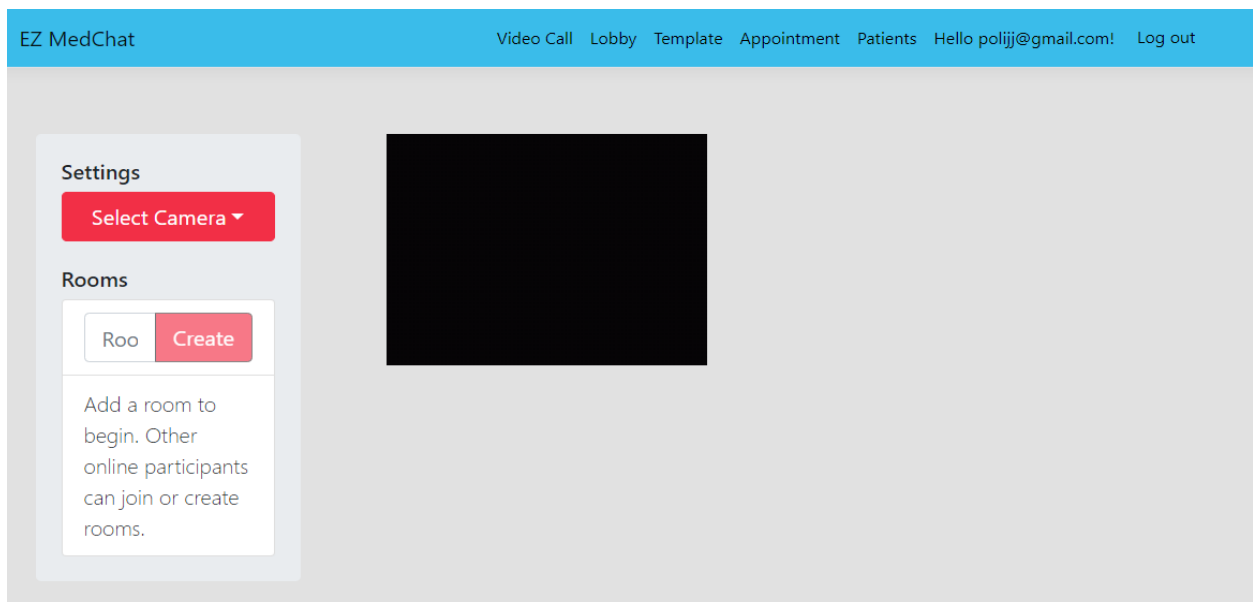


FIGURE 7: VIDEO CALL PAGE (PRACTITIONER)

- Patients - to view patients assigned to the practitioner.

Epicid	FirstName	LastName	EmailAddress
e0w0LEDCYfckT6N.CkJKCw3	Warren	McGinnis	warrenmcg@gmail.com
eh2xYHuzl9nkSFVvV3osUHg3	Olivia	Roberts	oliviareob@gmail.com

FIGURE 8: VIEW PATIENTS PAGE

Patients Side of Web Application

The goal of this part of the web application is to be easy to use for patients to contact, schedule appointments, and teleconference with their medical practitioners. The patients will be using the same application as practitioners, but with different user roles. Below are the main features.

- The Patients will use the following tabs to accomplish their requirements:
 - Waiting room - to have the chance to see a practitioner.

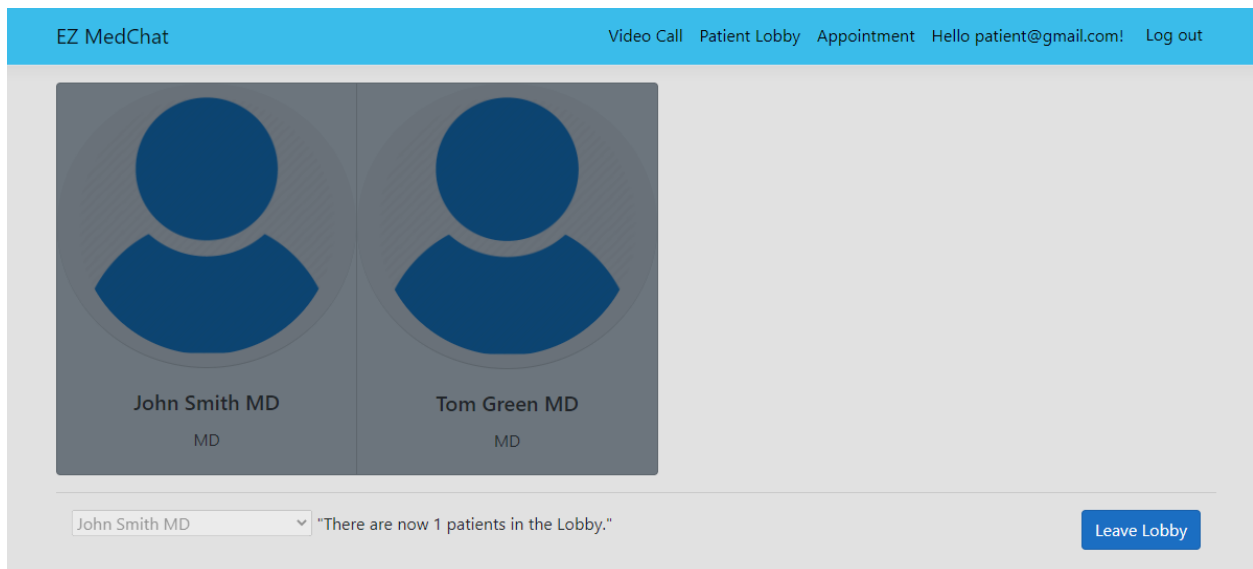


FIGURE 9: LOBBY PAGE (PATIENTS)

- Scheduling - see scheduled appointments and schedule new ones.

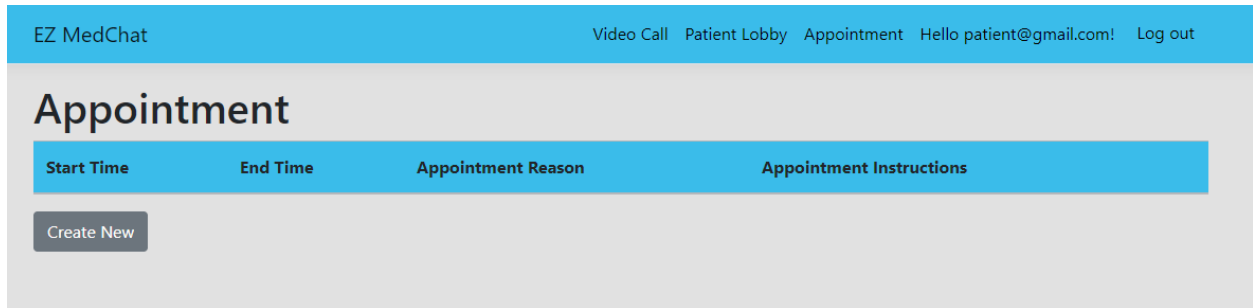


FIGURE 10: APPOINTMENTS PAGE (PATIENTS)

- Video Call - message practitioners through video and chat.

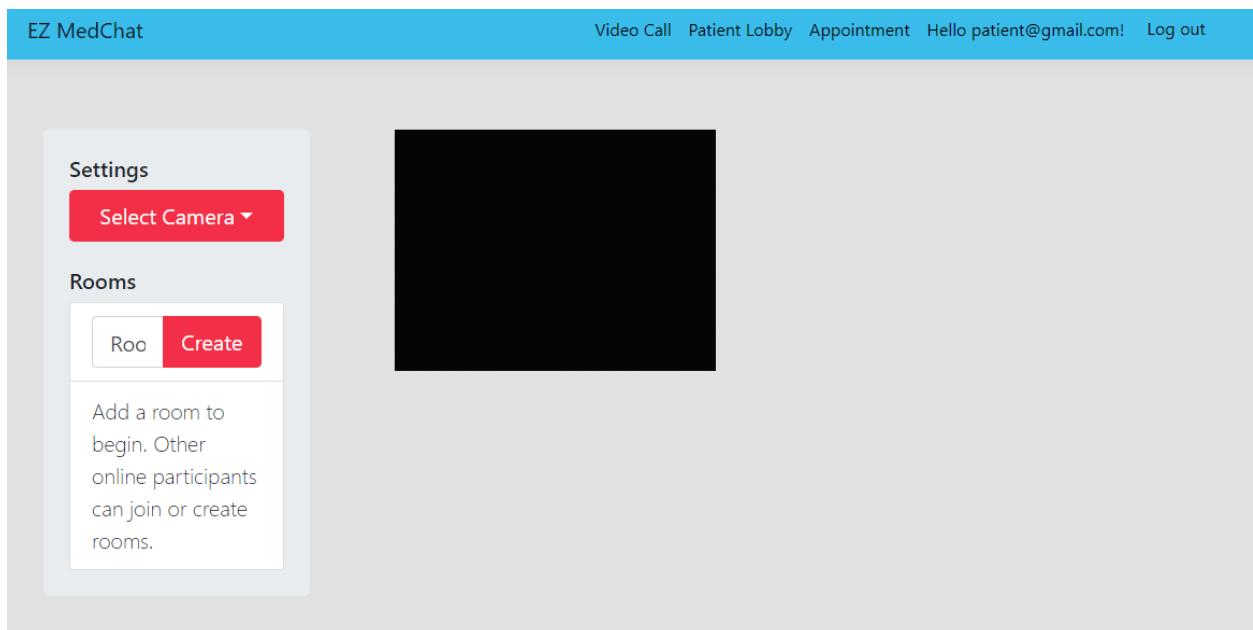


FIGURE 11: VIDEO CALL PAGE (PATIENTS)

2.4 USER PROFILES

The user profiles for EZ MedChat are simple. There are only two main user bases and they are the Medical Practitioners and the Patients. Below are the details of each of the user profiles.

Table 1: User Profile breaks down the medical practitioner user base of the application. This shows how we expect these users to behave as well as what knowledge we assume they have coming in.

User Profile 1: Medical Practitioners
Related Experience: Need to be familiar with how to use a browser.
<p>Experience with similar applications:</p> <p>Most doctors can use an e-visit app successfully, and most of the EZ MedChat’s features will resemble it.</p>
<p>Task Experience:</p> <p>Users need basic web page navigation skills, such as using the mouse to navigate the website, using keyboard to type messages, make notes, etc. and other basic functionality.</p>
<p>Frequency of Use:</p> <p>The user will access the web application on an everyday basis to check up on any updates from their patients, to meet with their patients, create templates/forms and so on.</p>
<p>Key Interface Design Requirements that the Profile Suggests:</p> <p>The web application will need to be HIPPA compliant. The app should be available at all times, with minimal to no downtimes. The user must be able to navigate through the web application with ease, and the app must use words to describe tasks or actions that are not clear. The web app will also provide a clear training video or instruction manual if needed.</p>

TABLE 1: USER PROFILE FOR MEDICAL PRACTITIONERS

Table 2: User Profile breaks down the patient user base of the application. This shows how we expect these users to behave as well as what knowledge we assume they have coming in.

User Profile 2: Patients

<p>Software and Interface Experience:</p> <p>Users should have experience using web browsers.</p>
<p>Experience using web browsers for basic functionality.</p>
<p>Task Experience:</p> <p>Basic app interactions - using a mouse to navigate the web page, log in/out, go into video calls, check availability, join lobby.</p>
<p>Frequency of Use:</p> <p>The user will access this website whenever they feel like they need to get in touch with their primary care physician.</p>
<p>Key Interface Design Requirements that the Profile Suggests:</p> <p>The web application will need to be HIPPA compliant. The website should be available at all times, with minimal to no downtimes. The user must be able to navigate through the web application with ease, and the web application must use words to describe tasks or actions that are not clear.</p>

TABLE 2: USER PROFILE FOR PATIENTS

2.5 USE CASE DIAGRAMS

Figure 1: Medical Practitioner Use Case Diagram *The following diagram, Figure 1, demonstrates the medical practitioner use case for the EZ MedChat web application. The diagram shows the predicted user flow.*

EZ MedChat Practitioner User Flow

Yisroel Weisberg | April 12, 2021

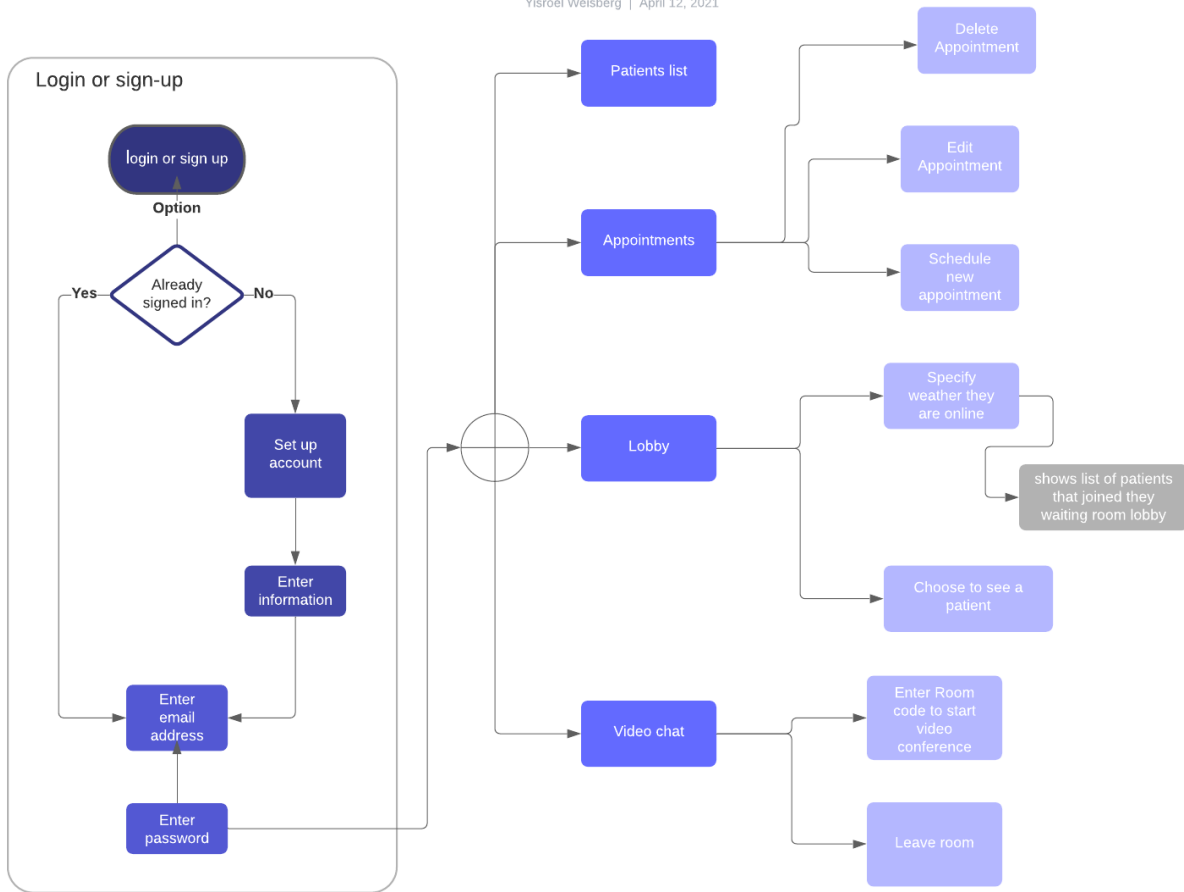


FIGURE 12: MEDICAL PRACTITIONER USE CASE DIAGRAM

Figure 2: Patient Use Case Diagram The following diagram, Figure 1, demonstrates the patient use case for the EZ MedChat mobile application. The diagram shows the predicted user flow.

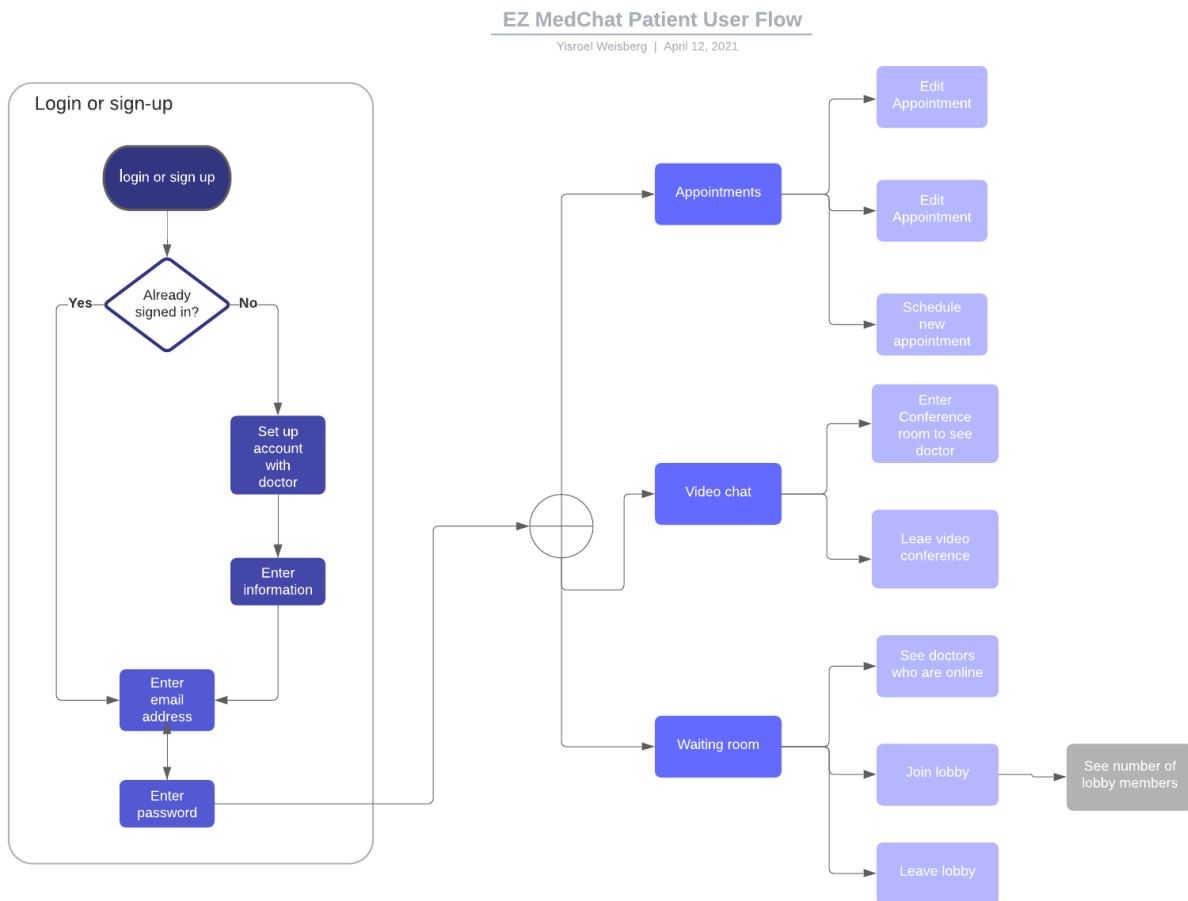


FIGURE 13: PATIENT USE CASE DIAGRAM

2.6 TECHNICAL ARCHITECTURE

Our team chose the technologies we would use for our project. These technologies cover the Web Application, Mobile Application, Database, and Security architecture. All required time to learn, as our team had never used them before. Though it was worth it, to allow for the creation of a well-made final product.

Web Application

The architecture of the web app is built using asp.NET Core. Prior to this, we were considering using .NET framework, but it will not be supported much longer, and did not have many of the functionality required to integrate it with the mobile app. For example, the most updated version of SignalR, as it has not released an update for .NET framework for about a year. SignalR is a free and open-source real time server communication notification to client-side web apps. Another feature the app will have is a way for users to create custom templates, save them, and be able to retrieve them.

Azure Cloud Services

Azure was used as the platform for hosting our database and web app. This allows for complete scalability of the web app and database depending on the traffic. So, if the application gets busy it can be upgraded to a higher performance tier. Another perk to using Azure is the built-in security features that will be used to protect our database and web app and keep everything HIPAA compliant.

2.7 TESTING

Testing Methodology and Approach

Methodology has been broken down into two parts, functional and non-functional methods.

For the functional testing, we concentrated on two specific methods: unit testing and integration testing. Unit testing was chosen because our application is composed of many individual components like video calls, templates, lobbies, and other features. That made it important to ensure each separate piece is working correctly. Similarly, integration testing was

chosen because our application is composed of many pieces. While each may function alone, we need to ensure all the individual components work together.

For the non-functional testing, we concentrated on performance testing and usability testing. Performance testing was selected because in theory, our application is supposed to service an undetermined number of medical professionals and patients at any given time. Therefore, ensuring our application functions properly under a heavy load is imperative.

Usability testing was selected due to the fact our project is a healthcare application. Since our users include patients, our users could have any number of disabilities. Therefore, making sure our application is as accessible as possible is paramount.

Test Plan

Web Application - Practitioner & Patient

- 1A - Register an account using new credentials.
- 1B - Logout of the application. Only have access to sign in/register/home pages.
- 1C - Login with existing credentials and be redirected to the home page. Get authorization to other pages.

Web Application - Practitioner

- 2A - Access lobby page - set online status.
- 2B - View list of patients that joined the lobby.
- 2C - Leave lobby (destroy lobby).

- 3A - Database functionality.
- 4A - Access appointments page - view list of appointments.
- 4B - Create appointments.
- 4C - Edit Appointments.
- 4D - Delete Appointments.
- 5A - Meet (video call) with patients.
- 6A - View list of patients assigned to practitioners.
- 7A - Access template page.
- 7B - Create/Edit Template.
- 7C - Save Template.
- 7D - Load + Render template.
- 7E - Modify rendered template + save it.
- 7F - Delete template.

Web Application - Patient

- 8A - Access lobby page - view online practitioners.
- 8B - Join lobby of an online practitioner.
- 8C - Leave lobby.
- 9A - Access appointments page - view list of appointments.
- 9B - Create appointments.
- 9C - Edit appointments.
- 9D - Delete appointments.

- 10A - Access video call page & turn on camera.
- 10B - Join call.
- 10C - Leave call.

Test Results for Practitioner User Testing

Record #	Test Case #	User + Role	Expected Output	Actual Output	Pass/Fail	Reason for Success/Failure	Date
1	1A	Angel Vincent + Tester	Account registered and & Redirected to home Page	Account registered and & Redirected to home Page	Pass	Requirements matched	4/1/2021
2	1B	Angel Vincent + Tester	Home Page, but logged out	Home Page, but logged out	Pass	User logged out	4/1/2021
3	1C	Angel Vincent + Tester	Home Page	Home Page	Pass	credentials correct	4/1/2021
4	1C	Jake Poli + Tester	Home Page	Invalid Attempt	Pass	Incorrect Credentials	3/21/2021
5	2A	Jake Poli + Tester	Lobby Page	Redirected to Login	Pass	not authorized	3/21/2021
6	2A	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021

7	2B	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
8	2C	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
9	4A	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
10	4B	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
11	4C	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
12	4D	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
13	6A	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
14	7A	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
15	7B	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
16	7C	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
17	7D	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021
18	7E	Angel Vincent +	No Messages	No Messages	Pass	User Authorized	4/1/2021

		Tester				d	
19	7F	Angel Vincent + Tester	No Messages	No Messages	Pass	User Authorized	4/1/2021

Test Results for Patients User Testing

Record #	Test Case #	User + Role	Expected Output	Actual Output	Pass/Fail	Reason for Success/Failure	Date
1	8A	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
2	8B	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
3	8C	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
4	9A	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
5	9B	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
6	9C	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
7	9D	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
8	10A	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021

						d	
9	10B	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021
10	10C	Austyn + Tester	No Message	No Message	Pass	User Authorized	4/2/2021

2.8 BUDGET

The budget for EZ MedChat is \$39,330 for the 36 weeks our team has been working on this application. As the application gets more traffic the Azure price will be increased due to the increased usage of Key Vault, the need for increased processing on the application, and a higher tier Database. Below is the table that shows all the details of the budget.

Table 3: Project Budget The table shows the projected budget for our project. It is an estimation of real-world costs. The budget represents what it would cost to host and run the application with the technologies we chose.

EZ MedChat Budget					
No.	Item	Unit	Time	Unit Price	Total
Software					
1	Azure	1		\$200/month	\$1,800
2	Twilio	1		\$270/month	\$2,430
Labor					
1	Developers/DBA	3	468 Hours	\$25/hour	\$35,100
Total					\$39,330

TABLE 3: BUDGET

2.9 PROBLEMS ENCOUNTERED AND ANALYSIS OF PROBLEMS SOLVED

One consistent problem we ran into was choosing the correct platforms. We spent a lot of time going between .NET Framework and Core before choosing one. We also wasted time looking into Azure FHIR requests when it was something we didn't need for the project.

We also changed our project several weeks in, causing our project to run behind. We had to come up with a new development plan, including what platforms to use, how to host the project server and database, and how to structure the project.

In the spring semester, there were more issues that were encountered, like using multiple Object Relational Mapping (ORM) libraries. In the fall semester, the team used Entity Framework to set up the authentication and authorization of the application. In spring, the team decided to use Dapper ORM due to the ease of usage, and its speed. So instead of having a mixup of ORMs, it was decided to completely get rid of Entity Framework, and use Dapper to accomplish all the tasks, including authentication.

One main issue we faced during the project was having to drop the mobile application. Since we weren't familiar with some of these technologies, we followed along with online tutorials to configure it within our database, without making it unique to our application. Because of these learning curves, we ended up being tight on time, and rushing to get all our features done. Due to these constraints we decided to drop the Xamarin mobile application and apply our focus on the web application.

2.10 RECOMMENDATIONS FOR IMPROVEMENT

We could have improved our project planning. From the start, we were not certain about what our project was going to be and changed around the idea. From there, we did not plan out the building of the application because we were not too familiar with the technology.

So, adding that learning phase to the plan/schedule would have helped out, although it wouldn't help when it comes to the timeline for the class.

We should have given our primary features the importance first. We are building our application from scratch, and at first, we focused on the supporting features such as login functionality, navigation, etc. before focusing on the application's core features. Also, we planned on having too many features while building out almost two different applications, which limited time and resources. Because of this, there were a lot of time constraints and rush to get everything done. If we started by focusing on our application's core features, we could have created a better, more polished final product.

3 CONCLUSION

3.1 LESSONS LEARNED

We have learned many things working on this project. Since this is a software project, most of the things we learn revolve around building an enterprise application that supports different types of users on different platforms. We learned that breaking parts of the project into smaller pieces is easier to comprehend and make progress. During each week, we tried to tackle several problems at once, when we should have tried to focus on smaller details. Everything takes longer than expected, so breaking each unit into even smaller pieces and having an estimate on those would have made it easier. Another important lesson we learned was when we were picking out a platform to build a project, we had to ensure that all the features the project needed can be implemented using that platform. Along with that, research

to select the best tool to build it. For example, the web app was built using the Visual Studio IDE. Later, the team learned IntelliJ IDE was a better tool to build it.

3.2 ABILITIES AND SKILLS DEVELOPED THROUGHOUT THE PROJECT

Several abilities and skills were picked up or improved upon while working on this project. Firstly, we learned a great deal about enterprise application development, because this was the first time we worked on a project this large in school. Though it was scrapped, we originally had a mobile component to the application, and in the process of working on it, we learned a great deal about mobile development and Xamarin, which was our chosen mobile framework. The team used .NET Core to develop the web application. This required learning and understanding how it worked and its many quirks and features. Since both Xamarin and .NET Core are C# frameworks, working on them allowed the team to improve our knowledge of the language. Since we decided to host the application in Azure having no prior experience with it, we had to understand the fundamentals of the platform. Furthermore, Azure Key Vault had to be learned since we decided to use it to store our applications secrets, as well as Azure Active Directory, and Azure permissions to enable our application to seamlessly pull and use the secrets.

3.3 COMPLETED DURING SPRING

The team's focus was to complete the website and get it hosted in Azure. Due to time constraints, the team had to drop the mobile application part of the project. Most importantly, the features that make our application unique has been completed. Instead, new user roles were created that shared the web application. Along with that, the team also completed the test cases for the application.

4 REFERENCES

4.1 REFERENCES

Altschuler, J., Margolius, D., Bodenheimer, T., & Grumbach, K. (2012). Estimating a Reasonable Patient Panel Size for Primary Care Physicians With Team-Based Task Delegation. *The Annals of Family Medicine*, 10(5), 396-400. doi:10.1370/afm.1400

Boyle, P. (2020, June 26). U.S. physician shortage growing. Retrieved December 5, 2020, from <https://www.aamc.org/news-insights/us-physician-shortage-growing>

Lee, B. Y. (2016, September 11). Doctors Wasting Over Two-Thirds Of Their Time Doing Paperwork. Retrieved December 5, 2020, from <https://www.forbes.com/sites/brucelee/2016/09/07/doctors-wasting-over-two-thirds-of-their-time-doing-paperwork/?sh=2dc5016f5d7b>

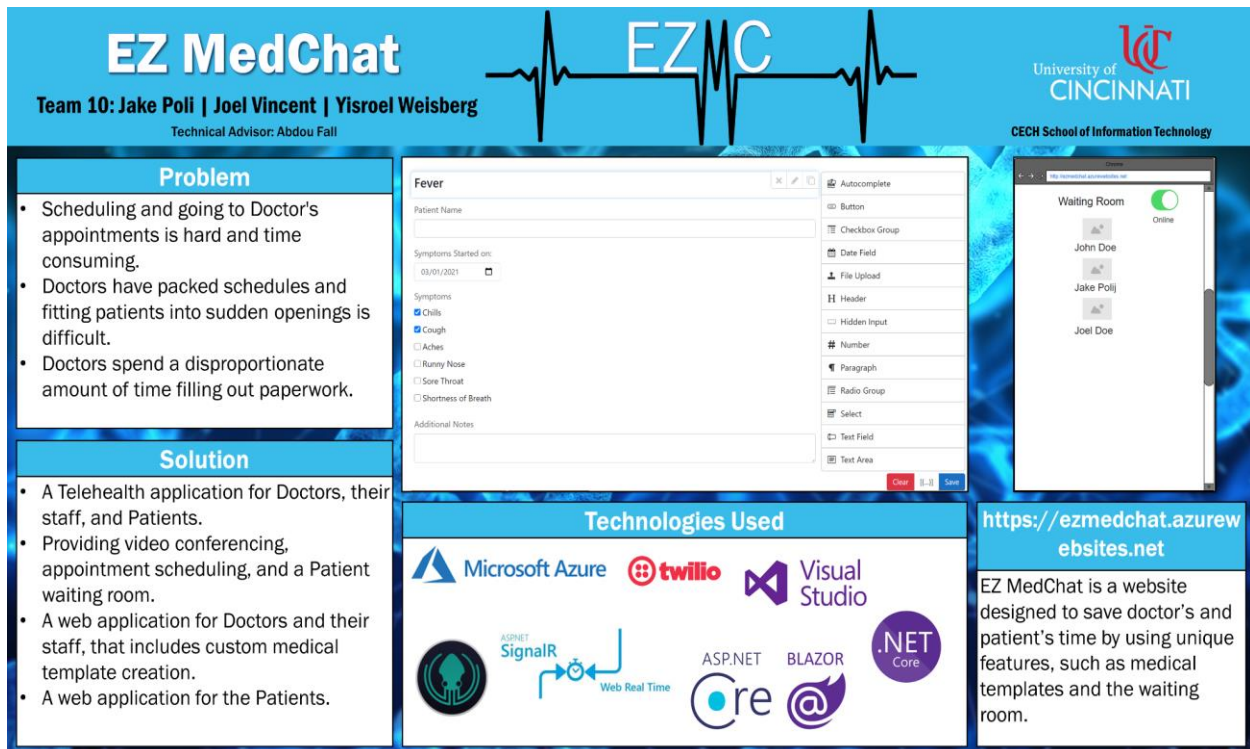
Neighmond, P. (2016, March 07). Can't Get In To See Your Doctor? Many Patients Turn To Urgent Care. Retrieved December 5, 2020, from <https://www.npr.org/sections/health-shots/2016/03/07/469196691/cant-get-in-to-see-your-doctor-many-patients-turn-to-urgent-care>

Torrey, T. (2020, March 02). Trying to Find a Primary Care Doctor. Retrieved December 5, 2020, from <https://www.verywellhealth.com/why-is-it-difficult-to-find-a-primary-care-physician-2615101>

5 APPENDIX

5.1 APPENDIX A: IT TECH EXPO POSTER

Appendix A: IT Tech Expo Poster The following figure shows our final poster that was presented at Tech Expo 2021. It displays an overview of our entire project.



The poster features a blue background with a white heartbeat line. At the top left, the text reads "EZ MedChat" in large white font, followed by "Team 10: Jake Poli | Joel Vincent | Yisroel Weisberg" and "Technical Advisor: Abdou Fall". At the top right is the University of Cincinnati logo and "CECH School of Information Technology".

The poster is divided into several sections:

- Problem:** A box with a blue header containing three bullet points:
 - Scheduling and going to Doctor's appointments is hard and time consuming.
 - Doctors have packed schedules and fitting patients into sudden openings is difficult.
 - Doctors spend a disproportionate amount of time filling out paperwork.
- Solution:** A box with a blue header containing three bullet points:
 - A Telehealth application for Doctors, their staff, and Patients.
 - Providing video conferencing, appointment scheduling, and a Patient waiting room.
 - A web application for Doctors and their staff, that includes custom medical template creation.
 - A web application for the Patients.
- Technologies Used:** A box with a blue header containing logos for Microsoft Azure, Twilio, Visual Studio, ASP.NET SignalR, Web Real Time, ASP.NET Core, Blazor, and .NET Core.
- Medical Form:** A screenshot of a form titled "Fever" with fields for Patient Name, Symptoms Started on (03/09/2021), Symptoms (Chills, Cough, Aches, Runny Nose, Sore Throat, Shortness of Breath), and Additional Notes. A right-hand sidebar lists UI controls like Button, Checkbox Group, Date Field, File Upload, Header, Hidden Input, Number, Paragraph, Radio Group, Select, Text Field, and Text Area.
- Waiting Room:** A screenshot of a "Waiting Room" interface showing a list of names: John Doe, Jake Poli, and Joel Doe, with an "Online" indicator.
- URL:** A box with a blue header containing the URL <https://ezmedchat.azurewebsites.net>.
- Description:** A box with a blue header containing the text: "EZ MedChat is a website designed to save doctor's and patient's time by using unique features, such as medical templates and the waiting room."