Computer Science and Engineering, University of Nevada, Reno

ePCR System

Team #27

Kennedy Anukam

Mason Harlan

Yi Jiang

Alec Moore

Devrin Lee

David Feil-Seifer

Bryan Pond

February 12, 2020

Table of Contents

Abstract 3

Project Description 3

- **Main Goals and Objectives** 3
- **Technological Description** 3

Significance 4

Legal and Ethical Aspects 4

Changes and Progress since the Initial Project Concept 5

Project Responsibilities 5

- **Web** 5
- **Tablet App** 7

Project Monitoring and Risks 8

Contributions of Team Members 9

References 9

Abstract

The ePCR system is a patient record tracking and entry system designed for use by the Rescate Rescue ambulance service. The system will allow patient records to be logged and accessed with ease from anywhere through the web. The user interface of the system must be intuitive and support both english and spanish. The database of patient records must be secure. The system could have a big impact on healthcare in lower income regions. This project will be open source in order to benefit as many healthcare providers as possible. This document will outline the significance of the project and how the senior design team will implement the project as a team.

Project Description

Main Goals and Objectives

A primary goal of the ePCR project for the Rescate Rescue team is to improve upon their current method of logging and interacting with patient data. The project seeks to allow the rescue team to log their data on a web application that can store and query patient care records. The project is needed as it solves the problem of storing and creating all of their information by hand. The rescue team currently writes down all their patient care records. One of the main goals is to allow the users to instantly view detailed statistics and reports generated from the patient data stored in the system. This will be a dramatic improvement from paper records since creating statistics analysis on hardcopies can take hours. Written records also come with the challenge of readability and physical storage. With our team's project, patient care reports will not suffer the issue of readability as data will be stored electronically with a database. The physical storage of patient care records will be eliminated with the use of a database for storing records. Another primary goal of the ePCR project is security. We seek to create a web application that has limited to no security flaws and is reliable. This is needed as the information that the rescue team will be logging is sensitive and critical. Security can be achieved by encrypting stored information, limiting access to sensitive information, and authenticating all users. Reliability can be achieved by creating secure backups of databases and deploying in a cloud environment to minimize downtime. An additional goal of the project is to make the website and application fully functional on multiple devices. This is significant as a multi-device web application will increase the level of convenience for the response team. The system will also need to be multiplatform and will support an android implementation. The rescue team has acquired tablets that will allow them to log patient information offline. This offline information will need to sync with the main system when the user enters a wifi zone.

Technological Description

Our technological stack will focus on scalability (in the case aforementioned) and ease of development. Development will be focused on the web platform first to create the base functionality of the platform. Next, development on the android application will begin. This development process will be focused on the rescue teams tablet operating system. Support for other operating systems will be added once the rescue team's version is successful. Our programming language of choice and development will be heavily weighted towards Javascript and notably the markup language HTML. The web app user interface styling will use Bootstrap and the React.JS component Material UI. To overview, our (tentative) frameworks, libraries, and software will be as follows:

Backend - Node.JS
Frontend - React.JS, Bootstrap, Material UI, Figma
Database/Storage - SQL

Source Control - GIT, Github

Significance

An open source ePCR system has the potential to make a big impact in the healthcare records industry. Current healthcare record systems are expensive and are licensed for use. This ePCR system would be free and open source, allowing anyone around the world to use it. This system is especially attractive to healthcare services in lower income communities and could have a positive societal impact. For example, a small ambulance service working in rural Mexico does not have the budget to purchase a license for an expensive ePCR system. Having access to patient charts in a digital format would dramatically decrease the time spent searching through paper records and increase the amount of time providing care. With the team's implementation, the system would be able to retrieve specific data in an instant. The team's implementation would support multiple languages in order to reach as many communities as possible across the globe. SaaS companies would be acting in direct competition with the system. Some of these companies include: ZOLL [zoll.com], imagetrend [imagetrend.com], and CloudPCR [cloudpcr.net]. These other systems are not able to implement specific needs at such a low cost. The team's implementation of an ePCR system would tailor directly to the charts and needs of specific communities. Creating an open source system will allow a community to maintain and improve the system which is something that SaaS providers do not have. More eyes on the system will help identify security issues and make needed changes that might be too small for the main developers. The ePCR system will have little impact on the environment and will be deployed on public servers such as AWS to reduce e-waste and cost.

This project will allow the development team to gain real world experience as software developers. Getting feedback from the rescue team as development continues will teach the developers professionalism and allow them to practice requirements elicitation. This project is a large platform and will expose the developers to new technologies and tools that they have not encountered before. Increased exposure to new practices and development methods will help the developers hone their interests and stay up-to-date on the latest development strategies. This project focuses on developing practical skills and has the potential to make the team better engineers. The development team is excited to contribute to a meaningful open source project that will make a positive impact on healthcare.

The ePCR system has the potential for further developments past the time frame of the senior projects course. One core development would be to generalize the system to allow other healthcare providers to use it. Adding support for multiple browsers and operating systems is critical for maintaining and expanding the system. Several features such as rapid deployment and increased security and required to extend the lifetime of the system. Additional features such as annual statistics reports and customizable chart entry fields would help increase the accessibility and usability of the system.

Legal and Ethical Aspects

There are some potential legal issues that may pertain to the project. The use of the developed software will most likely be for the Rescate Rescue team only. A potential legal issue that pertains to the use of the ePCR system is data confidentiality. If an attacker were to compromise the data stored in the database, they would have access to the personal details of the users and patients. Another potential legal issue related to the use of the system is liability. If the user of the system enters inaccurate information, that could potentially place the patient at harm.

To ensure that the product and its related modifications meet the highest professional standards, the team will review the specifications given and the documentations made earlier to avoid straying from the main intent. The team will also frequently test and debug the project to verify that the project meets the

sponsor's expectations and desired goal. The team plans to also develop the project that will effectively protect and respect the privacy of medical information that will be affected by the project. This makes sure that the system maintains the integrity of data and that the team will focus on developing solutions that may counter sensitivity and flawlessness. The team will only utilize information with authorization and not use them in unethical ways.

Changes and Progress since the Initial Project Concept

We have made significant, demonstrable progress. The development has proceeded inline with what we set out to do during the project conception, in that very little major changes have been made to the web application. Our web application is running and being expanded upon to meet all requirements set. As of now, the following notable accomplishments have been achieved:

- Backend and frontend communication
- SQL server operating and being performed upon by backend
- Login, authorization, registration, and admin functionality
- Major frontend development
    - Settings pages
    - Chart pages
    - Patient page
    - Increased dashboard functionality

Many other features have been added but these were of particular note and importance.

Our project has received the major change of our sponsor requesting us to make an application for Android tablets on top of the existing requirement of the web application. This is so users can use it offline, which is of importance as the current users are located in a small town where internet is only available in buildings. This will be marked by us changing development away from React and Node.JS to a to-be-determined Android development kit. The Android development should be expedited by the already made progress on the web application, the backend REST API that has been written, and the React pages which could be transferred to Android.

Project Responsibilities

Web

The web version of the application has subsystems that combine together to create the electronic patient care reporting system. The subsystems include: user authentication/authorization, registration, chart creation, viewing charts, search patients, summary reports creation, trend call analysis, dual languages, dashboard, and settings.

The user authentication system is responsible for verifying the user is allowed to access the website. Authentication is implemented on the frontend and the backend. On the frontend, the user has to go through a login page to enter the website. On the backend, the credentials are verified against the database and a response is sent back. Within the response, the user is assigned a JSON web token if the credentials are valid. The token is used for the client to make requests which serves to authorize the user's request. Requests on the frontend cannot be served by the backend without a valid JSON web token.

The registration system is also implemented on the frontend and backend. On the backend, the credentials are checked to see if the user is allowed to register. The frontend of the registration system provides the interface for the client to enter their credentials.

The chart creation system has implementations from both the frontend and backend. Most of the chart design is largely separated into a few navigable subsections: call information, patient information, assessment, vitals, interventions, etc. For each of these subsections, values are saved to the main chart component and are yet to be submitted without confirmation from the user. The user can freely move between the subsections should they have made any mistakes on the chart filing. Once the user confirms the chart creation, the frontend here is finished and the backend takes over. Each of the subsections' input values will be sent as a JSON string to the database. The backend will first focus on the patient information by grabbing the ID for that particular patient. This ID is used on the chart table so that when a patient is searched or viewed, the associated charts will appear.

The viewing charts component will work with the charts already stored in the database. In a typical patient reporting system, each chart is assigned and filed inside of a patient's folder. This is implemented the same way, as each chart that could be subsequently built for this existing patient could be found under their page. The viewing charts component will also ensure users the ability to add additional notes after the charts have been created. Since the chart's information is locked after creation, the only functionalities enabled will be adding notes, medications, and procedures.

The search patients component is constructed with the backend and the frontend. The backend sends the chart data and the frontend filters the data by the patient being searched.

The summary reports subsystem is mainly implemented on the backend. The system is responsible for generating a summary report based on the results saved in a certain time range. The time range is entered by the user on the frontend. The system summarizes a predefined set of parameters for all the patients in the given time range.

The trend call analysis is implemented on the frontend and the backend. The backend utilizes the database to summarize the call volume in the requested time range. The frontend renders the data that the backend sends back to it.

The dual languages subsystem allows the application to be rendered in Spanish and English. The subsystem allows the user to toggle back and forth between the two languages.

The dashboard subsystem provides the UI layout of the application. The layout is responsive and is rendered differently on different device sizes.

The settings subsystem is implemented on the frontend and the backend. The system allows the user to update their credentials. Admin users have a broader scope of actions that can be performed. Admin users can add, delete, and elevate the privileges of all users in the system.

Table 1 displays the subsystems of the ePCR system alongside the team member responsible for developing it.

Table 1: The subsystems of the ePCR system and the designated team member responsible for it.

Subsystem	Team Member Responsible
1. User Authentication/Authorization 2. Backend: Login 3. Summary Reports/Trend Call Analysis	Alec
1. Registration 2. Frontend: Login 3. Settings	Kennedy
1. Chart Creation 2. Viewing Charts	Yi
1. Search Patients 2. Dual Languages 3. Dashboard	Mason

Tablet App

The tablet app is a simplified version of the web application. The app is composed of 4 main subsystems. The subsystems are: login/logout, create chart, post chart, and view chart.

The login/logout subsystem is responsible for granting and taking away access to the ePCR system. Registration will not be available through the tablet application.

The create chart subsystem will allow the user to create a patient chart even when offline. The act of being able to utilize the system without an internet connection offers a greater level of flexibility.

The post chart subsystem allows the user to post the patient data when they are around an internet connection. The user will be notified if the post was successful or not.

The view chart subsystem allows the user to view a patient's chart. The data will be rendered to fit the tablet application or other mobile devices.

Table 2 shows the subsystems of the ePCR tablet application alongside the team member responsible for developing it.

Table 2: The subsystems of the tablet ePCR system and the designated team member responsible for it.

Subsystem	Team Member Responsible
1. Post Chart	Alec
1. Login/Logout	Kennedy
1. Create Chart	Yi
1. View Chart	Mason

Project Monitoring and Risks

The team plans to monitor the project's progress through weekly meetings. The team has agreed to use Mondays at 3:15 PM for the specific day and time of the week. For each meeting, a team member will possibly list out the responsibilities, progress results, and completions with any components of the project subsystem. The team may suggest feedback on what could be improved or implemented in a more efficient manner. The team will outline the major components needed to complete the desired project and check off each to ensure that requirements are met. This way, the team has a clear vision of where things are moving and what has been accomplished before the final deadline.

Figure 1 shows the Risk Register table pointing out possible risks and potential consequences that the team may face. The "likelihood" and "impact" sections use the scale from 1 to 5, with 1 being the lowest and 5 being the highest. The "severity" is calculated based on "likelihood" multiplied by "impact". If the values fall between 1 and 5, the severity of that risk on the project is low. Likewise, a range of 6 to 12 indicates a medium risk. Any values higher than that are considered at a high risk. The team has given some mitigation strategies that can be utilized to reduce the risks.

Risk Register												
Risk ID	Risks	Potential Consequences	Current Risk			Status	Owner	Mitigation Strategies	Residual Risk			
			Likelihood	Impact	Severity				Likelihood	Impact	Severity	
Team risk												
RP01	Low and poor productivity	Slow down progress or may delay meeting project deadline	4	4	16	Open	Team	Assign tasks to members of the right specialization and set clear goals to act as incentives	2	3	6	
RP02	Poor participation during live meetings or sessions	Limited understanding of the project and may not achieve the desired results	3	2	6	Open	Team	Prepare with questions faced during design and ask to clarify possible confusions	2	1	2	
RP03	Conflict among team members	Inconsistencies and variances in project and may bring in disagreements in each other's work	2	2	4	Open	Team	Utilize project manager skills and appoint a team leader who can resolve all types of potential conflicts	1	2	2	
Design risk												
RP04	Adding new ideas and functionalities to project	Inflate project timeline or inability to finish project on time	4	4	16	Open	Team	Restrict to investor proposals and only finish the required components	3	3	9	
RP05	Lack of acceptance from investor approval	Project might not be going the direction it needs to be or team may suspend activities	2	3	6	Open	Investor	Research existing ePCR designs and make alternative solutions to the specific subsystem needing approval	2	2	4	
RP06	Unclear instructions or lack of information from investor	More time needed to verify errors or more research may be needed to design the correct work	2	5	10	Open	Investor	Reach out to investor frequently and or schedule to meet with investor to strengthen the understanding between design team and investor team	1	3	3	
RP07	Project unintentionally crashes or breaks	Refactoring/redesigning/retesting of project or project stops	3	5	15	Open	Team	Frequently test for errors and look ahead of implementing ways to attack possible design problems that may occur	2	4	8	
Time risk												
RP08	Acceptance of unrealistic deadlines	Failure to meet deadlines and or result in low quality of project	2	3	6	Open	Team	Set clear goals for each week and break deadlines into fair intervals with at least some progress completion	2	2	4	

Figure 1: The Risk Register table shows the possible risks that could occur within the team and or during the development and design stages.

Contributions of Team Members

Table 3: The hours and contributions of each member in the team.

Member	Hours	Contributions
Kennedy Anukam	2	1. Legal and Ethical Issues 2. Project Responsibilities
Mason Harlan	2	3. Revised Abstract 4. Revised Project Description 5. Significance/References
Yi Jiang	2	1. Legal and Ethical Issues 2. Project Responsibilities 3. Project Monitoring and Risks
Alexander Moore	2	1. Project Responsibilities 2. Changes and Progress

References

https://www.cloudpcr.net/

https://www.imagetrend.com/

https://www.zoll.com/