Coding Project Final Report Summary

Group 9: Martin Gawron, Maryann Olugbamila, Micah Olugbamilia, Nolan Reilly

Project Overview

PatientCare+ enables users to monitor vital health metrics like heart rate, blood sugar, and blood pressure. The app provides AI-generated health insights, tracks health trends, and allows data sharing with doctors, reducing unnecessary hospital visits. It is designed to be user-friendly, secure, and accessible to patients and doctors, facilitating better health management and encouraging positive lifestyle changes with reminders and notifications.

Credit Where Credit is Due

This project is the creation of Group 2 of the Fall 2022 semester at UIC. The group consisted of members: Saif Alnuaimi, Sultan Alshkeili, Hazaa Alhosani and Sultan Aljneibi. Without the incredible creativity and work that they have put into this project we would not have made this great application, we give them the credit they deserve for creating this great project to work on.

Comparison to Original Project Description

In terms of device compatibility, the original project report given to us had in mind a mobile application. This medium for connecting patients and doctors is most convenient as phones are accessible for most people. However, for the sake of convenience, since we are better versed in web applications, we built the prototype as a web-based application. The same approach can be used to develop a mobile version of the application.

One thing to also note is that the original report had in mind a way for sensors to periodically collect data from the user with no need for the patient to manually input data each time. Instead of this approach, since it would be much more complex to set up automatic sensor reading, we opted for the easier method of manual user data entry. Although the amount of data is not as much as wanted by Group 2 of Fall 2022, the data can still be insightful.

Also, we were able to implement many of the desired use cases with only missing out on two-way communication between the patient and their doctor, as well as doctors generating a patient code for the patient to join the application and connect with their doctor. We also did not have time to implement the feature of patients scheduling appointments with their doctor through the application. On a positive note, we successfully implemented a way to add health data, patients manage doctors in their list, view doctor messages and prescriptions; doctors can also view patients health data, send messages and prescriptions. On top of this, we also have an AI feature for analyzing health data, algorithm generated health scores/statistics; while doctors can have an urgent care list, add to todo list, and track financial performance of practice.

Overall, the original report for this project seemed to have been motivated by the idea of automated health tracking, where patients can track their data with no additional stress with extra insight from their doctors. Even though our approach to building the prototype is different to this end vision of Group 2, Fall 2022, we believe that our project still exemplifies the great bridge between patients and doctors where doctors can gain greater insight into their patients' health.

Testing, Testing, and more Testing…

The goal of this project is to connect patients and their doctors in a more meaningful way that allows for patients to streamline their health goals. It can only be as streamlined as the application, therefore we determined testing of functionality and appearance was crucial. We made sure to streamline the testing of the server functionality by using tools such as Postman to help confirm API calls work as intended, this was on top of rigorous calls to the API to also test response payloads and statuses. For the frontend part of the project, calls to the API were tested for error handling to make sure the user experience is never interrupted and as smooth as possible. Added to this was the testing of any block of code that was performing any operations on data sets to prevent corruption and confirm corrected formatting. To top this all off, our review of the UI design on release 2 triggered a design overhaul to improve the user experience. Even though testing design choice can be nuanced, we thoroughly believe that this overhaul has revamped our application for the better, both visually appealing and easily navigable.

Project Retrospective

As a group, we feel that this project – along with the development project – has taught us all very valuable lessons in software engineering practices. Not only have we learned a lot about the technical aspects of development, programming, and frameworks, we have also taken great strides in understanding group dynamics, time management, and project workflow.

Looking back on our performance and experience, we think that we got many things right when taking on this project. For example, for release 2 we attempted the extreme programming method of peer programming, this was very successful as it increased both productivity and understanding of each others' code. We also appreciate the opportunity to work alongside great programmers/developers, this opportunity gave us room to grow as we challenged each other to keep improving throughout the semester. We are also very pleased with our performance on release 2 and 3, as we believe the client was very pleased with the result and are proud of what we were able to accomplish.

Although there are many things that we plan on continuing to do in terms of future programming practices, we do believe that there are many things that we can improve upon. We have learned the importance of being prepared for each and every meeting, we were not always prepared for each scrum meeting and they did not feel as productive because of that. In terms of agile development, sometimes it took some time for our team to give feedback on an action item which slowed down progress for members at times; we look forward to improving on this for future collaboration as we have learned how vital fast and constructive feedback can be. While learning Jira, we tried our best with adapting to that workflow, reflecting back on our use of that software we have realized that we should be more detailed in Jira stories and tasks for a story. This could have really sped up our development process and kept everyone on track.