HELMO: Acute Concussion Evaluation System – The Cloud Storage and User Authentication System

Shawn Zou,

Department of Electrical and Electronic Engineering,

Imperial College London, SW7 2AZ

Supervisor: Professor Yiannis Demiris

***Abstract* – Helmo is a helmet integrated with an array of sensors and mobile application to aid diagnosis and evaluation of brain injury due to severe impact. The helmet records data that allows visualization of impact force and path to represent how the impact incident occurred in real-time. By incorporating cloud storage capabilities through mobile application, the software will minimize hardware dependency and storage requirements.**

I. INTRODUCTION

As technology advances, the amount of wearable and connected devices is forecasted to reach $25 billion by 2019 [1]. Technology companies started collaborating to challenge and disrupt Life Science [2] and other industries, redefining technology in people's everyday lives by integrating sensors and mobile phones with everything people use. One of the common features on the software user interface side of all the connected devices is the visualization of data produced by the hardware. From allowing users to check their historical record to processing the data to provide some kind of recommendation to the users, secured data storage requirement and user authentication to improve security have become an area that cannot be overlooked [3]. "In healthcare, for example, physicians are tablets to process sensitive data about their patients," said Gartner's research director, Dionisio Zumerie. The increasing needs for robust and secure data storage has raised the standard of database for mobile applications, and lead us to building the database for the application for Helmo in a more rigorous manner – which is instead of storing the data on the data log of the application or in the phone's local memory, all the data collected from Helmo will be stored in cloud server.

In this report, we will discuss backend of how the database will be built and how we will ensure the user information will be stored securely as well as methods to authenticate users who wish to access the data, including patients and medical practitioners.

II. RELATED WORK

There is an unaccountable number of related work in developing internet-based applications from web-based to mobile phone-oriented. With the emergence of various technologies that push researchers to support high density data transmissions in Wide Area Networks [4], data transmission protocols are always updated when new technologies are introduced. As a mobile application developer, we aim to focus on the best practices that are available. Since the iOS development community does not have a standard best practice that it follows, we will aim to work within the guideline provided by the Digital Single Market‘s Digital Economy & Society of European Commission [5].

III. TOOLS AND IMPLEMENTATION

III.A. Web-based development programming – PHP

In order for the application to interact with a cloud storage platform, it needs to go through a "connection file" where the data are received by the server and finally inserted into the database. There is no currently existing solutions that enables direct communication between the mobile application and database. The database will be built in cloud, utilizing web-based services.

As mentioned, in order to communicate with the database, we will need a server that acts as the middleman between the application and the cloud storage unit. There are many web-based programming languages, including HTML, JavaScript, PHP, Python, and so on. We have chosen PHP.

But what is PHP? Used by 75 percent of web servers, PHP is a general-purpose server-side programming language [6]. The main advantages of PHP are attributed to it being an open source language, so it has a large base of established code blocks that can be referenced along with online community support and compatibility with multiple platforms. PHP is generally used for applications that has relatively low traffic requirements, as opposed to Python or Java that is more suitable for applications like National Weather Services or Yahoo! Maps [4].

Another web-based development tool that is strongly comparable with PHP is ASP.net. The latter is a programming framework that uses objective-C. One of the arguable advantages of ASP.net is the use of objective-C, making it a more efficient, scalable, maintainability, and future development onto a larger project. However, it requires more time to develop. In the interest of time for the project, which is one of the largest limiting factor, we have chosen PHP as the server-side scripting language.

III.B. Web-Hosting Services – where you run the Cloud Database

As internet infrastructure capability increases, more and more people are connected via the world-wide web. Whether through web browsers or applications on devices, anytime the information is exchanged through a non-local storage, it is through "the cloud." There are many web-hosting services provider on the market now, some of which, such as GoDaddy and Bluehost, were amongst the oldest and most recognized companies in the field in early 2000's. However, started in 2002, Amazon Web Services (AWS) has risen the rank and captured a significant amount of users in cloud computing. We have chosen AWS to host our database due to its ability to allow us to buy and build customized virtual private server, and according to experienced users, it runs much faster than competitors [7].

However, before simply stating AWS is faster and thus chosen as the hosting service provider, it needs to be noted that due to AWS's flexibility in allowing users to build customized VPS, developers are required to handle almost every single backend details from installing Apache to setting up MySQL to taking care of backups and server updates. Therefore, opting for AWS requires high level of technical competence.

III.C. User Authentication

The Apple iOS development environment have not developed a standard best practice to authenticate user information. Therefore, most of the developers depend on existing protocols and tools of their choice. In order to aid streamlined user experience and provide fuss-free account registration, we have chosen Firebase – it provides tools and infrastructure from backend development to frontend user interface support. We will be using its functions to allow users to register with our app through authentication using external login – such as Facebook, Google+, Github, or Twitter [8]. In general, the flow of the data and authentication should follow OAuth2, an authorization framework that enables applications to obtain limited access to user accounts on an HTTP service [9].

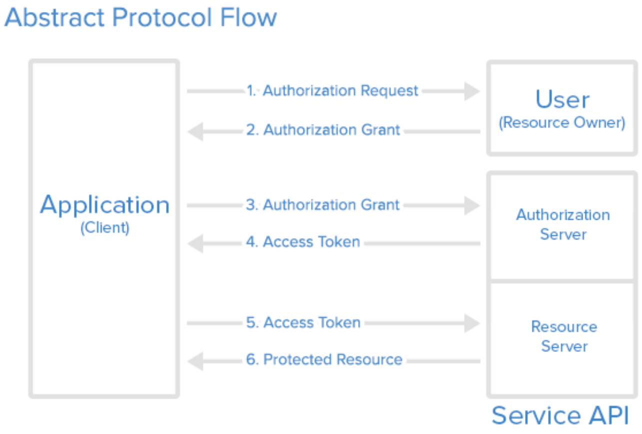

Figure 1. OAuth2 Protocol Flow

The benefits of using external application login include streamlined user experience, easier access, and most importantly, it eliminated the requirement for the team to build the entire user account database from ground-up. However, enabling user registration based on third-party accounts means we will not have control over user account status through third-party; e.g., if the third-party server experiences downtime, our user will not be able to login to the application.

III.D. Database and Data Transfer

The MySQL database will be created on the chosen hosting service provider, AWS. Using Amazon Simple Storage Service (S3), we would be able to maintain secure, durable and highly scalable object storage in the cloud [10].

Since user account information will be taken care of via external provider, i.e. Facebook, Google, or the service of choice of the user, our MySQL database will only contain the core data of Helmo, which is the information representing the users' fall or incident at the time of impact.

Each user account will generate corresponding code that is stored in a table containing a list of information that the application allowed the user to input, e.g. patients' height, age, previous medical records that could aid medical diagnosis, as well as other sensor data generated from the helmet.

III.E Data Sharing in Closed Network

In order for the patient to share their information with the doctor, a few methods were discussed; exact implementation method will be chosen later in the project stage.

a) Social-Network Mechanism

The idea of social-network approach to sharing user information is similar to the "friends" feature of many social-network-sites, such as Facebook, MySpace, etc [11]; where the patients' information is only available to a designated doctor (say doctor A) if their user identity is connected to Doctor A. The downside of this would be that the process of information sharing should be autonomous for the patient, because when the incident occurs, it is more than likely that the patient will not be able to actively "add" a "friend" to enable doctors to view their information. Another disadvantage of this implementation for this project is that it will create a network that is more complicated than necessary for this application [12].

b) Private-Public Information Pool

In this Private-Public pool, patients' information will only be available to all the doctors. The network topology will be similar to that of a start network [13]. In order for the doctors to see the patients' data, the doctor will just need to identify patients' name or identity tag on the mobile application. Each individual patient will only be able to check their own data, and their data is securely stored within their block.

c) Open Subscription

Open subscription setup will be very similar to the private-public information pool configuration. The difference will be that once the doctor has chosen to subscribe to the patient, the doctor will be able to view patients' status at all times. The main advantage of this method is allowing the medical practitioners to monitor patients in real-time. Whereas in the previous implementation, doctors will have to search for the patient every time he or she wishes to examine patients' status.

IV. CONCLUSION

We have now laid out how the database will be built, evaluated options of tools to use based on time-efficiency in adopting them, scalability, maintainability, performance, and financial costs. In each of the chosen implementation methods, I have also discussed their downsides and potential difficulties we may face. One of the potential changes in tools will be that, I am using Firebase for account registration and authentication, while Firebase also provides web hosting services and other full-range infrastructure products similar to that provided by AWS. While AWS also provides easy-to-use third-party account registration methods, Firebase provides more clarity in implementation. Therefore there could be a change in choice of web-based service provider depending on implementation.

References

[1] CSS Insight, "Wearables Market to Exceed $25 Billion," CSS Insight, 2017.

[2] A. Frappe, "Tech Giants Collaborate and Disrupt in Life Science Industry," PharmExec.com, 2015.

[3] WeLiveSecurity.com, "Why Mobile Security is More Important than Ever Before," WeLiveSecurity.com, 2015.

[4] D. H. D.V. Bernardo, "Securing Data Transfer in the Cloud Through Introducing Identification Packet and UDT-Authentication option Field: A Characterization," 2013.

[5] European Commision, "Summary Report on the Public Consultation on the Green Paper," 2017.

[6] Y. Shiotsu, "Web Development 101: Top Web Development Languages in 2014," 2014. [Online]. Available: https://www.upwork.com/blog/2014/03/webdevelopment-101-top-web-development-languages-. [Accessed 23 02 2017].

[7] B. Edward, "Freewheelings.co," 22 03 2016. [Online]. Available: https://www.freewheelings.com/firing-bluehosthow-i-doubled-my-site-speed-using-aws-and-otherfree-services/. [Accessed 23 02 2017].

[8] "Firebase," 23 02 2017. [Online]. Available: https://firebase.google.com/docs/auth/ios/facebook-login. [Accessed 23 02 2017].

[9] M. Anicas, "DigitalOcean.com," 21 04 2014. [Online]. Available: https://www.digitalocean.com/community/tutorials/an-introduction-to-oauth-2. [Accessed 23 02 2017].

[10] Amazon , "Amazon Web Services," [Online]. Available: http://docs.aws.amazon.com/mobile/sdkforios/developerguide/s3transfermanager.html. [Accessed 23 02 2017].

[11] P. J. M. M. A. P. Amanda L. Traud, "Social Structure of Facebook Networks," *Social Structure of Facebook Networks,* no. arXiv:1102.2166 [cs.SI], 2011.

[12] C. J. S. A. D. Bernadette C. M. van Wijk, "Comparing Brain Networks of Different Size and Connectivity Density Using Graph Theory," 2010.

[13] S. B. Chi-Hsiang Yeh Univ. of California, "Macro-Star Networks: Efficient Low-Degree Alternatives to Star Graphs," *IEEE Transactions on Parallel and Distributed Systems,* vol. 9, no. 10, pp. 987-1003, 1998.

[14] C. J. S. A. D. Bernadette C. M. van Wijk, "Comparing Brain Networks of Different Size and Connectivity Density Using Graph Theory," 2010.