

ONE CARD MOBILE– A MULTIPURPOSE IDENTIFICATION AND ACCESS APP FOR IOS

Tran Ngoc Phuong Dang, Chi Hieu Tran, Huu Tat Dat Hoang

{tdang3618, htran2620, hhoang7975}@conestogac.on.ca

Supervised by: Dalibor Dvorski

School of Engineering and Information Technology

Conestoga College

TECHNICAL REPORT

August 2018

ABSTRACT

ONE Card is a mandatory and essential identification tool at Conestoga College. Everyday, all students and staffs at the college use the ONE Card for plathora of different purposes such as attendance check, onsite purchase, etc. However, the ONE Card department is lacking a useful management tool to help ONE Card users control their ONE Card. That is why we came up with a mobile app to help students and staffs at Conestoga College quickly and conveniently manage their One Card account.

This report will explore the design and purpose of ONE Card mobile – a multipurpose identification and access service app. Additionally, it will cover all major aspects of development, such as methodologies, architectural decisions, and technologies used. It will walk through all important technical details and thoroughly explain the development process. Finally, it will shows how the app and API were created for the iOS platform using the aforementioned technologies.

I. INTRODUCTION

Currently, it takes a lot of unnecessary complications for students and staffs at Conestoga to visit the official website to manage their Print Balance and Condor Cash. Furthermore, the website does not operate very smoothly and leaves much to be desired. On the other hand, there is currently no solution to remotely access ONE Card. All personnel at the institution have to carry their physical cards with them at all time which leads to much hassle.

That is why the students and staffs at Conestoga are in need of a cheap and comprehensive solution that the college administration can quickly deploy for everyone at a moment's notice, and thus, ONE Card Mobile is the key to solving these problems.

ONE Card Mobile is a mobile multipurpose identification and access service that is focused on reducing the overall hassle in ONE Card usage, in addition to enabling fast and easy transaction management for printing and onsite purchase. ONE Card Mobile will also attain a competitive edge by offering services such as virtual card, which have become necessary for any personnel at the school.

ONE Card Mobile intends to establish and operate an iOS app with services costing significantly less than the prices of its competitors while supplying superior quality. Incorporating its website and graphic art services, ONE Card Mobile will enable all educational organizations to reduce their ID and transaction management costs.

II. METHODOLOGIES

The methodology used for this project was based on the Agile Unified Process. At the beginning of the project, an overall plan was created to guide the team throughout the process. Then, a detailed plan was created each sprint, defining which tasks needed to be completed by each member. Each sprint lasts two weeks, and every week a progress meeting was commenced to make sure the project was heading to the right direction.

To control the code during the development process, BitBucket was utilized as the main version control. This helped the team members keep track of each development step and maintain the quality. As for the documentation, the team used Google Docs and Google Sheets as the means of document sharing. They allowed multiple team members to remotely work on the same document.

III. ARCHITECTURE

1. Software-as-a-Service

The first idea of ONE Card mobile used traditional client-server architecture. However, according to the customer's requirements, the app was to be extremely customizable and could, potentially, be reused in various scenarios. To allow this reusability, one major change was required: the server architecture needed

to be decoupled from the client architecture. For this reason, the team decided to completely re-imagine the solution as an API to leverage Software-as-a-Service model.

The key idea was to allow the API to be reused by different businesses. That is why we made use of the organization abstraction. Each institution will run the

service using a unique organization id, which will completely separate the context of the application. This way, our app can be deployed at any college in Canada. The **Figure 1** below shows the deployment diagram describing the main architecture of our entire system.

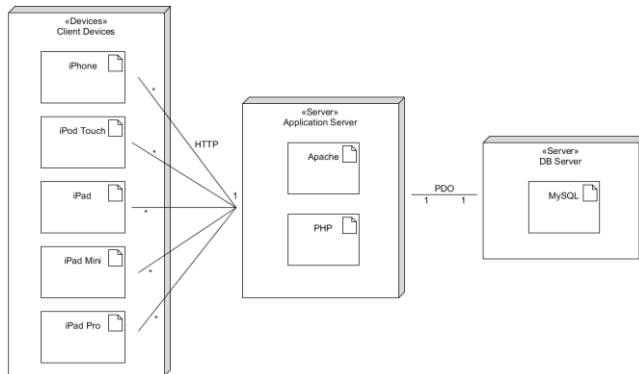


Figure 1: Deployment diagram

2. Software Design Patterns

In this project we made use of the Singleton design pattern for the Session class. The Session class contains token and expired time of session. It has a unique instance which is only created once when the application starts, and provides a global scope for other classes to access it. The **Figure 2** below shows the structure of this Session class.

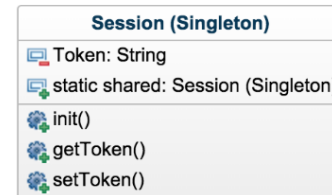


Figure 2: Session class

IV. TECHNOLOGIES

There are several sub-parts contributed to the ONE Card project such as the iOS mobile application and the management website, the Application Program Interface(API), and the database. In general, the iOS application was built on Xcode 10, the management website and API used the PHP framework, and the database was hosted on a MySQL server. **Figure 3** summarizes all the technologies used in this project.



Figure 1: Technologies in use

1. ONE Card application

The application was built for the iOS platform. It was implemented using the Xcode 10 beta, Swift 4.2 and

the beta SDKs for iOS 12. With the Xcode 10 and Swift 4.2, the application was published with markedly improving performance. Using Xcode 10, we also took advantage of the support for developers and worked collaboratively with each team members by utilizing an array of collaborative platforms such as GitHub, Bitbucket and GitLab [1].

2. Management Website

For the back-end side, it was developed using the PHP (Hypertext Preprocessor) language version 7.8.2 from Zend Technologies Ltd. In comparison to the previous versions, PHP 7 has a significant increase in performance, which leads to the website can run faster [2].

For the front-end side, HTML5 (Hyper Text Markup Language) and CSS3 (Cascading Style Sheets) were involved. By using a wide range of new features, we managed to make the website look professional.

Moreover, JQuery 3.2.1 is also utilized to implement the client sides of the website for easier usage.

3. Application Program Interface (API)

The API is a set of clearly defined rules which was implemented using the PHP framework. It is used as the back-end service to support the ONE Card mobile application to access its database.

4. Database

We used MySQL Community Server version 5.6.39 to construct the database. MySQL is well-known as an open-resource relational database management system (RDBMS). Also, MySQL allows for quick and reliable database connections. As a result, it contributes to boosting the application performance to meet the requirements [3].

V. OTHER TECHNOLOGIES

1. Data Interchange Format

JSON (JavaScript Object Notation) is the data-exchange format that was used in the project. The biggest advantage of JSON is that it uses a small size of data when transferring between servers and clients, which results in faster responses. JSON also supports a variety of data structures such as booleans, numbers, or strings.

2. JSON Web Tokens (JWT)

The project applied the JWT as the standard of tokens (RFC 7519). It was generated and managed by the server for authorization purposes and to secure the connection between the mobile application and the server [4].

3. Face ID, Touch ID

The project used Face ID and Touch ID for authorization in order to bring a sense of convenience to user. To be more specific, after the first successful login to the application, users may use either the Face ID or Touch ID for next time login without typing network ID and password again. The **Figure 4** below shows an example of such authentication.

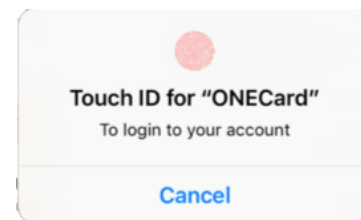


Figure 4: Touch ID authentication

VI. CLASS DESIGN AND DATABASE

1. Design Model

The solution comprises of 22 classes and partially implemented the Model-View-Controller (MVC) architectural pattern. The following diagram (**Figure 5**.) showcases the class diagram of our project.

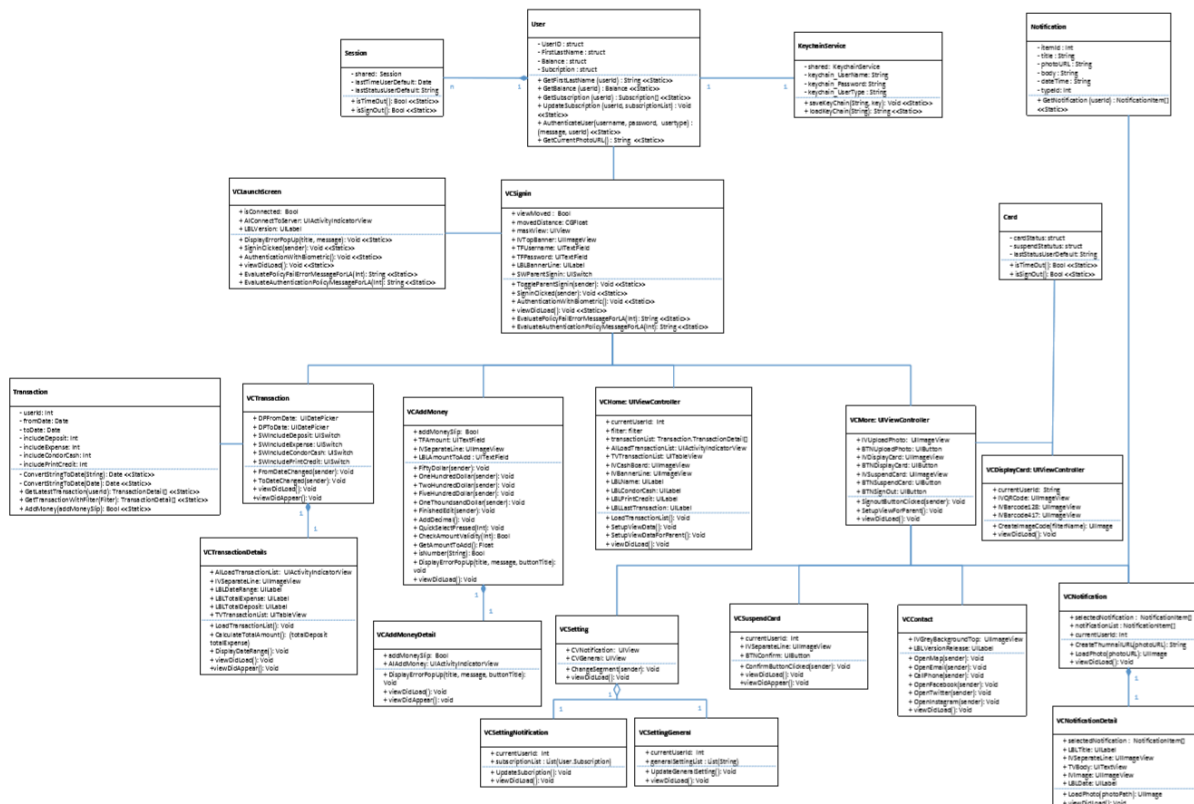


Figure 5: Class diagram

2. Entity Relationship Model

Based on the class diagram generated above, the ERD was created (**Figure 6**). This diagram was used to construct our database.

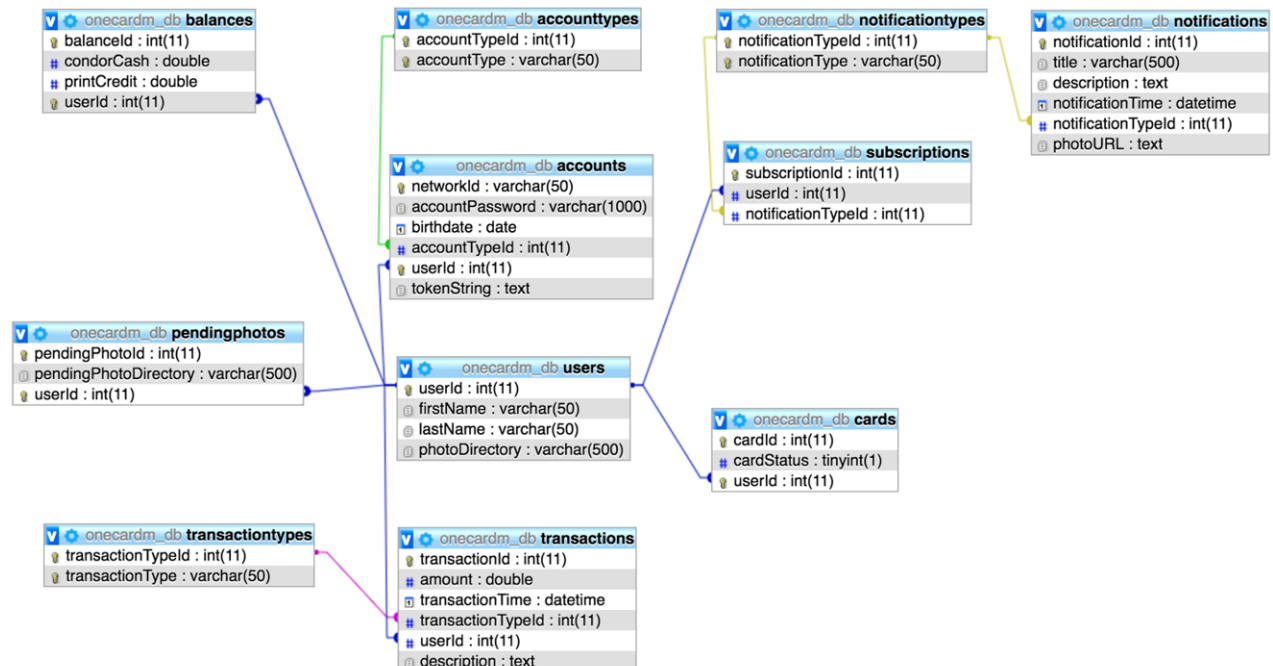


Figure 6: Entity Relationship Diagram

VII. CONCLUSION

This technical report shows how a combination of a variety of different technologies can lead to an innovative solution that solves common problems. Adding a new organization abstraction to the ONE Card Mobile allowed the app to become reusable by different institutions. Also, the Software-as-a-Service model used is a trending model and it is redefining the web. Therefore, we are confident that we have utilized the best approach for our project.

ONE Card mobile was a big project. And with lots of constraints, the solution was not perfectly developed, thus there are many ways to improve it in the future. Nevertheless, working side-by-side with the clients and our supervisor to receive constant feedback during the development process was a valuable tool to move the project forward. It was a great learning experience and a fruitful project. In the future, we will add many more features and develop this app to be the best one for its purpose.

REFERENCES

- [1] "Apple Developer," Apple Inc, 2018. [Online]. Available: <https://developer.apple.com/xcode/whats-new/>. [Accessed 13 August 2018].
- [2] "Zend Technologies," 2018. [Online]. Available: <http://www.zend.com/en/resources/php-7>. [Accessed 13 August 2018].
- [3] "MySQL Developer," [Online]. Available: <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>. [Accessed 13 August 2018].
- [4] "JWT," 2018. [Online]. Available: <https://jwt.io/introduction/>. [Accessed 13 August 2018].