

QR Code를 이용한 간편 로그인

팀 명 : Security-M
지도교수 : 이 병 천 교수님
팀 장 : 지승우
팀 원 : 이승용
백진이
박종범

2018. 11.
중부대학교 정보보호학과

목 차

1. 서론

1.1 연구 목적 및 주제 선정	1
-------------------------	---

2. 관련연구

2.1. SSO(Single Sign On)	2
2.2. QR Code	3
2.3. BCrypt 암호화	3

3. 본론

3.1 APM 구축	4
3.2 Laravel Framework	7
3.3 Chrome Extension	8
3.4 실행 화면	9

4. 결론	13
-------------	----

5. 별 첨

5.1 프로그램 소스	14
5.2 참고자료	38
5.3 발표 PPT 자료	39

1. 서 론

1.1 연구 목적 및 주제 선정

사람들은 누구나 사용하기 쉽고 간편한 서비스를 제공받기를 원한다. 이러한 니즈를 충족시키기 위해 이전보다 간편하게 웹과 앱을 이용하고 생활에 편리함을 주기 위해 사용되는 것이 QR Code이다.

ID와 PASSWORD에 의한 로그인 방식의 획기적인 변화가 필요하다고 생각했고 문자형 ID와 PASSWORD 대신 QR Code를 활용하기로 했으며 이로 인해 간편한 타겟 인식이 가능하고 인증 방식의 간소화로 다양한 분야에서 활용이 가능하다.

최근에 중국에서는 식당, 노점상, 지하철, 자판기 등 많은 곳에서 결제수단으로 QR Code를 사용하고 있고 상용화 되어있다.

국내에서는 신용카드 및 체크카드의 보급이 잘 정착되어 결제수단으로 사용되지는 않지만 광고 및 홍보 업계에서의 마케팅 수단으로 주로 활용되고 있을 뿐만 아니라, 홍보동영상 및 사진 정보와 할인권 및 입장권 정보, 또 영화관과 경기장 정보 등도 QR Code를 통해 받을 수 있다.

QR Code의 정보제공은 온라인과 오프라인을 넘나들며 이루어지는데, 예를 들어 잡지에 나온 어떤 제품의 QR Code를 인식시키면 해당 제품의 웹사이트로 연결되는 방식이다.

또한 개발자의 입장에서 보았을 때 관리자가 여러 시스템을 관리하게 되는 경우 각각의 ID와 PASSWORD를 필요로 하게 된다 .하지만 이런 경우 어떠한 문제가 생겼을 때 관리자가 즉각적으로 바로 대응할 수 없게되는 문제점이 있다. 그래서 SSO(Single Sign On)방식을 이용해서 통합적인 인증을 통해 서버관리의 불편함을 최소화하고 여러 가지 응용프로그램을 사용함에 있어서 더욱 편리하게 사용 할 수 있게 하였다. 또한 SSO의 단점인 하나의 서버가 침해될 시에 모든 서버의 보안이 침해된다는 문제점을 보완하기 위해 OTP(One Time Password) 방식을 참고해 30초마다 QR Code가 갱신되어 전에 쓰던 QR Code는 폐기되는 것으로 개발을 해 QR Code가 유출되거나 복제되어 오용되는 사고를 방지하고자 하였다.

이외에도 QR Code의 활용방안을 고안해보고자 본 프로젝트를 선정하였다.

2. 관련연구

2.1. SSO

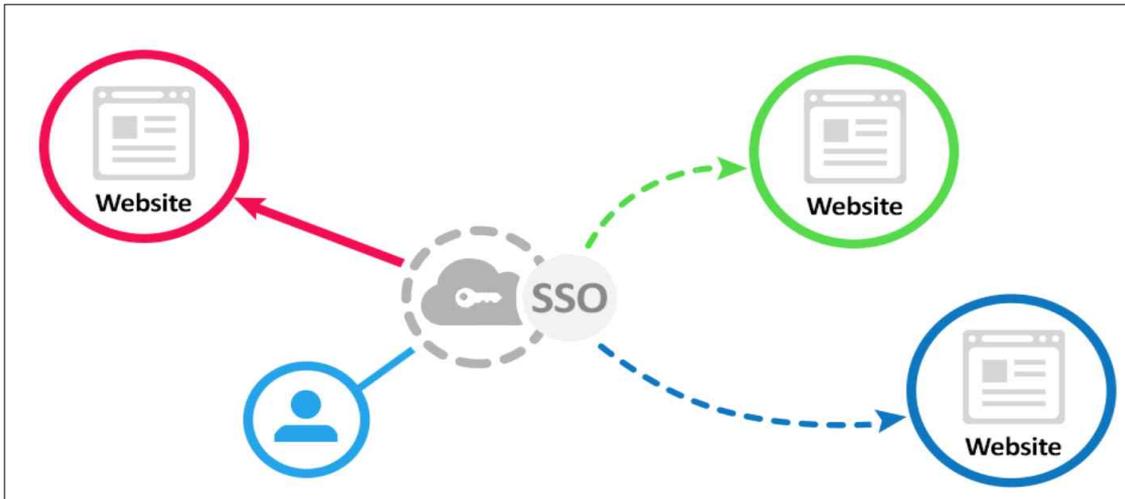
SSO는 Single Sign On의 약자이며, 한 번의 인증 과정으로 여러 사이트나 서비스를 이용할 수 있는 인증 기능으로, 첫 글자만 따서 SSO라고 불린다. 단일 계정 로그인, 단일 인증이라고도 부른다.

예를 들어 어느 컴퓨터에 로그인한 후 그룹웨어 등의 응용 프로그램을 사용할 때에 또 로그인을 하고, 다른 서버상의 응용 프로그램을 사용할 때에도 다시 로그인이 필요한 상황이라면, 사용자는 여러 개의 아이디와 비밀번호를 관리해야 하는 번거로움이 생긴다. 하지만 SSO 인증을 도입한 환경에서 사용자는 하나의 아이디와 비밀번호로 모든 기능을 사용할 수 있다.

여러 개의 사이트를 운영하는 대기업이나 인터넷 관련 기업이 각각의 회원을 통합 관리할 필요성이 생김에 따라 개발된 방식으로, 1997년 IBM이 개발하였으며 우리나라에는 2000년 코리아닷컴이 처음 도입하였다.

SSO의 가장 큰 약점은 일단 최초 인증 과정을 통과하고 나면 모든 서버나 사이트에 접속할 수 있다는 것이다. 이를 Single Point of Failure라 하는데, 이러한 치명적인 약점을 보완하기 위해 중요 정보에 대한 접근 및 동작 시 지속적인 인증(Continuous Authentication)을 하도록 되어 있다.

도입시에는 통합인증 대상 응용 프로그램 변경이 필요한지를 고려해야 한다. 응용 프로그램의 변경이 필요한 경우 도입 비용과 위험이 높아질 수 있다.



[그림 2.1] 싱글사인온 개요

2.2 QR CODE

사각형의 가로세로 격자무늬에 다양한 정보를 담고 있는 2차원(매트릭스) 형식의 코드로, QR은 Quick Response 의 약자이다.

1994년 일본 덴소웨이브사(社)가 개발하였으며, 덴소웨이브사가 특허권을 행사하지 않겠다고 선언하여 다양한 분야에서 널리 활용되고 있다.

기존의 1차원 바코드가 20자 내외의 숫자 정보만 저장할 수 있는 반면 QR코드는 숫자 최대 7,089자, 문자(ASCII) 최대 4,296자, 이진(8비트) 최대 2,953바이트, 한자 최대 1,817자를 저장할 수 있으며, 일반 바코드보다 인식속도와 인식률, 복원력이 뛰어나다. 바코드가 주로 계산이나 재고관리, 상품확인 등을 위해 사용된다면 QR코드는 마케팅이나 홍보, PR 수단으로 많이 사용된다.

QR코드의 정보 제공은 오프라인과 온라인을 넘나들며 이루어진다. 예를 들어 종이잡지에 나온 모 제품의 QR코드에 스마트폰을 갖다 대면 그 제품의 웹사이트로 연결되어 생산·유통·가격 정보 등 각종 상품 정보와 이벤트 정보, 쿠폰 혜택 등을 얻을 수 있고, 스포츠 경기나 영화 포스터의 QR코드를 스캔하면 홍보동영상 및 사진 정보, 할인권 및 입장권 정보, 영화관 또는 경기장 정보 등을 받을 수 있는 식이다.

은행이나 신용카드 회사, 항공사 등은 QR카드를 활용하여 결제서비스를 제공할 수 있고, 백화점이나 할인마트 등 유통업체는 가정으로 배송하던 할인쿠폰 대신 전단지나 신문광고, 홈페이지에 게재한 QR코드를 통해 할인쿠폰은 물론 행사품목이나 프로모션 내용까지 제공할 수 있다. 또한 공인인증서와 같은 중요 인증서들을 옮기는 데도 현재 사용중에 있고 이로 미루어볼 때 QR Code는 활용가치가 높고 잘 사용하면 인증 메커니즘의 변화를 가져올 수 있다고 본다.

또, 비석에 QR코드를 새겨 고인의 사진이나 생전의 기록을 볼 수 있는 모델을 출시한 비석회사도 있을 정도로 QR CODE는 다방면으로 활용이 가능하다.

2.3 BCRYPT 암호화

Bcrypt는 애초부터 패스워드 저장을 목적으로 설계되었다. Niels Provos와 David Mazières가 1999년 발표했고 현재까지 사용되는 가장 강력한 해시 메커니즘 중 하나이다. Bcrypt는 보안에 집착하기로 유명한 OpenBSD에서 기본 암호 인증 메커니즘으로 사용되고 있고 미래에 PBKDF2보다 더 경쟁력이 있다고 여겨진다.

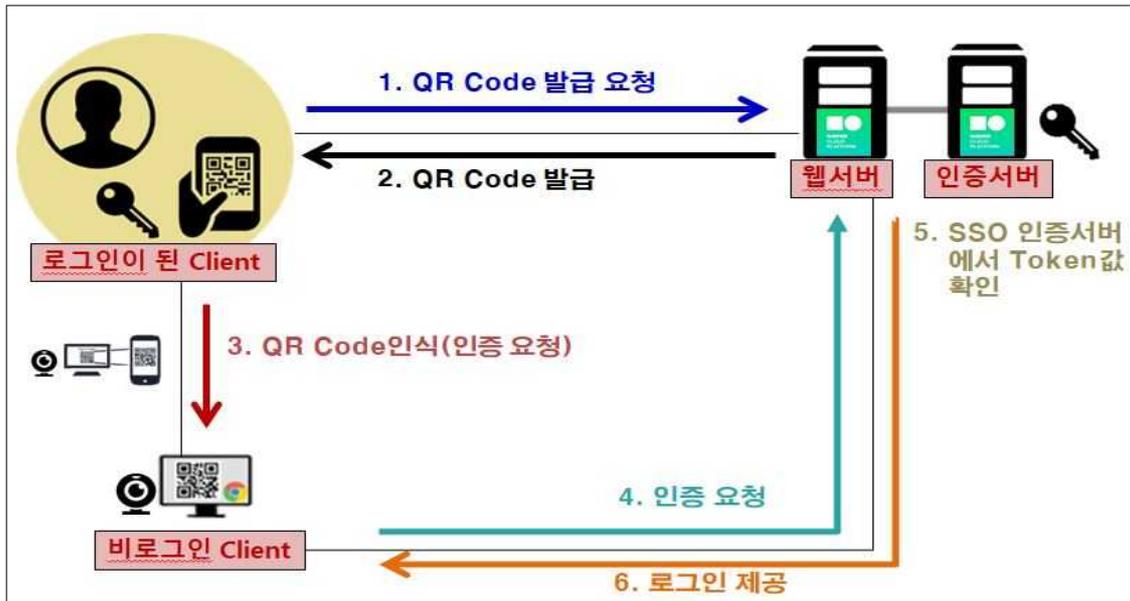
Bcrypt에서 "work factor" 인자는 하나의 해시 다이제스트를 생성하는 데 얼마만큼의 처리 과정을 수행할지 결정한다. "work factor"를 조정하는 것만으로 간단하게 시스템의 보안성을 높일 수 있다.

다만 PBKDF2나 scrypt와는 달리 bcrypt는 입력 값으로 72 bytes character를 사용해야 하는 제약이 있다.

또한 PHP 5.5 이상에서는 기본 내장함수로 사용되어지고 있다.

3. 본론

* 프로젝트 구상도



[그림 2.2.1] SSO(Single Sign On) 구상도

3.1 APM 구축

(1) Apache

아래 경로에서 최신버전 다운로드 및 설치

www.apachelounge.com/download/

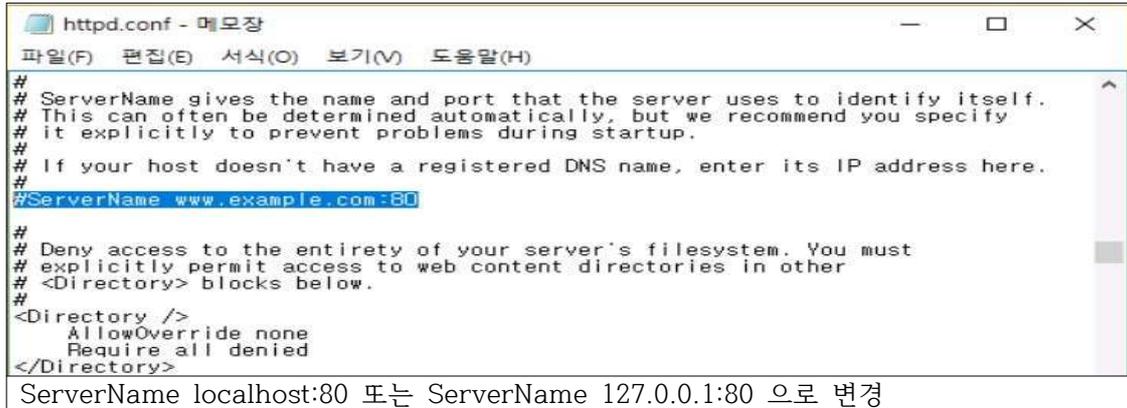
- 설치폴더 / conf 경로에 있는 httpd.conf 파일 수정

① ServerRoot 경로 변경

```

httpd.conf - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
# ServerRoot: The top of the directory tree
# configuration, error, and log files are
#
# Do not add a slash at the end of the dir
# ServerRoot at a non-local disk, be sure
# Mutex directive, if file-based mutexes a
# same ServerRoot for multiple httpd daemc
# least PidFile.
ServerRoot "c:/Apache24"
#
# Mutex: Allows you to set the mutex mecha
# for individual mutexes, or change the gl
ServerROOT " " 안의 부분을 아파치 설치 경로와 일치되게 입력
    
```

② ServerName 변경



```
httpd.conf - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
#
# ServerName gives the name and port that the server uses to identify itself.
# This can often be determined automatically, but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If your host doesn't have a registered DNS name, enter its IP address here.
#
#ServerName www.example.com:80
#
# Deny access to the entirety of your server's filesystem. You must
# explicitly permit access to web content directories in other
# <Directory> blocks below.
#
<Directory />
    AllowOverride none
    Require all denied
</Directory>
```

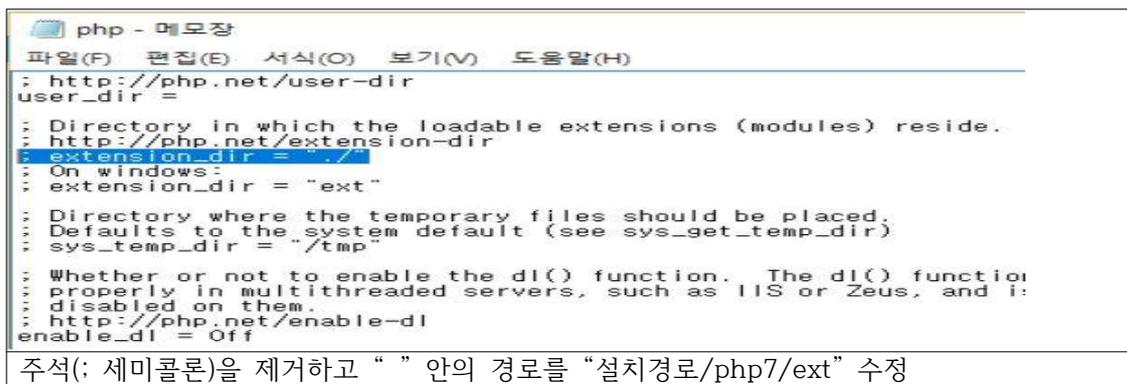
ServerName localhost:80 또는 ServerName 127.0.0.1:80 으로 변경

(2) PHP

아래 경로에서 PHP최신버전 다운로드

www.windows.php.net/download/

① 설치 폴더 내에 php.ini 파일 수정



```
php - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
; http://php.net/user-dir
user_dir =

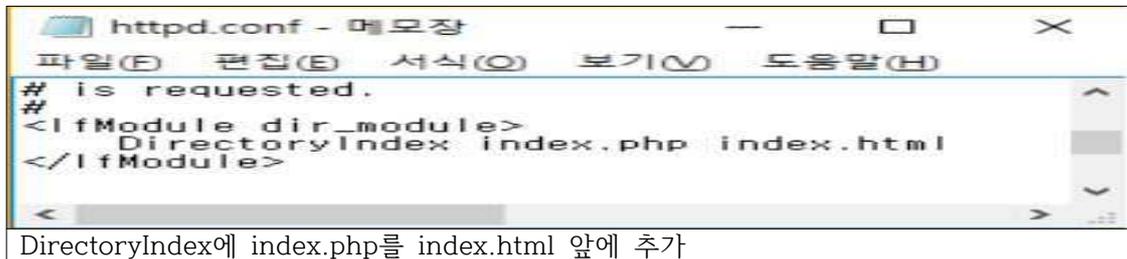
; Directory in which the loadable extensions (modules) reside.
; http://php.net/extension-dir
extension_dir = "./"
; On windows:
; extension_dir = "ext"

; Directory where the temporary files should be placed,
; Defaults to the system default (see sys_get_temp_dir)
; sys_temp_dir = "/tmp"

; Whether or not to enable the dl() function. The dl() function
; properly in multithreaded servers, such as IIS or Zeus, and is
; disabled on them.
; http://php.net/enable-dl
enable_dl = Off
```

주석(: 세미콜론)을 제거하고 “ ” 안의 경로를 “설치경로/php7/ext” 수정

② Apache와 php연동



```
httpd.conf - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
# is requested.
#
<IfModule dir_module>
    DirectoryIndex index.php index.html
</IfModule>
```

DirectoryIndex에 index.php를 index.html 앞에 추가

```

# Secure (SSL/TLS) connections
#Include conf/extra/httpd-ssl.conf
# Note: The following must be present to support
# starting without SSL on platforms with no /dev/random equivalent
# but a statically compiled-in mod_ssl.
#
<IfModule ssl_module>
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
</IfModule>

PHPIniDir "C:/php7"
LoadModule php7_module "C:/php7/php7apache2_4.dll"
AddType application/x-httpd-php .html .php
AddHandler application/x-httpd-php .php

```

맨 밑에 위와 같은 구문 추가

- 저장하고 나와서 Apache 서비스 재시작

```
httpd -k restart
```

(3) MySQL

아래 경로에서 Mysql 다운로드

```
http://dev.mysql.com/downloads/mysql
```

① 환경변수 편집

새로만들기(N) 버튼 클릭 후 위와 같이 값을 입력

② CMD에서 install

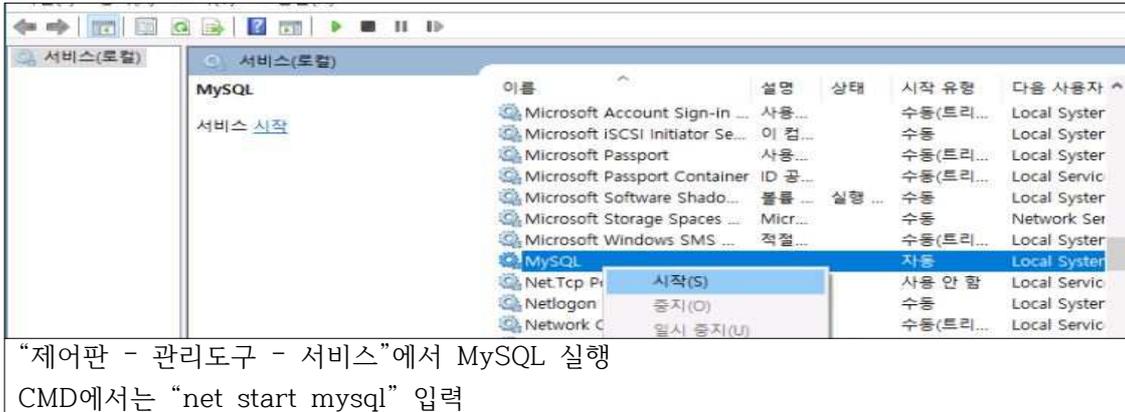
```

C:\WINDOWS\system32>c:\mysql\bin\mysqld.exe --install
Service successfully installed.
C:\WINDOWS\system32>

```

C:/mysql/bin/mysqld.exe -install 입력

③ 실행



3.2 Laravel Framework

(1) Laravel

웹 프레임 워크는 웹 서비스를 개발할 때마다 해야 하는 사용자 인증 및 세션유지, 데이터베이스 접속 템플릿 엔진 등을 미리 잘 포장해놓은 프로젝트의 뼈대 구조이다.

그 중 라라벨은 PHP를 사용할 수 있는 웹 프레임워크를 말하며, PHP로 만들어진 MVC 아키텍처를 지원하는 이 라라벨을 사용하면 웹 개발에서 다른 것보다도 중요한 로직을 구현하는데에 집중할 수 있다.

라라벨(Laravel)은 자유, 오픈 소스 PHP 웹 프레임워크의 하나로, Taylor Otwell 이 개발하였으며 모델-뷰-컨트롤러(MVC) 아키텍처 패턴을 따라 웹 어플리케이션을 개발하기 위해 고안되었다. 라라벨의 기능들 중 일부는 모듈 방식의 패키징 시스템이며, 전용 의존성 관리자, 관계형 데이터베이스에 접근하는 각기 다른 방법, 소프트웨어 전개와 유지보수의 도움을 주는 유틸리티가 포함된다.

라라벨은 Symfony, 젠드, 코드이그나이터, Yii2등과 함께 가장 대중적인 PHP프레임워크 가운데 하나로 간주된다.

(2) Composer 설치



(3) Laravel 설치

```
composer global require "laravel/installer"
```

```
C:\#Users#Sy_patrick>|laravel -v
Laravel Installer 2.0.1

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display this help message
  -q, --quiet               Do not output any message
  -V, --version             Display this application version
  --ansi                    Force ANSI output
  --no-ansi                 Disable ANSI output
  -n, --no-interaction     Do not ask any interactive question
  -v|vv|vvv, --verbose     Increase the verbosity of messages: 1 for normal out
ug
버전 확인
```

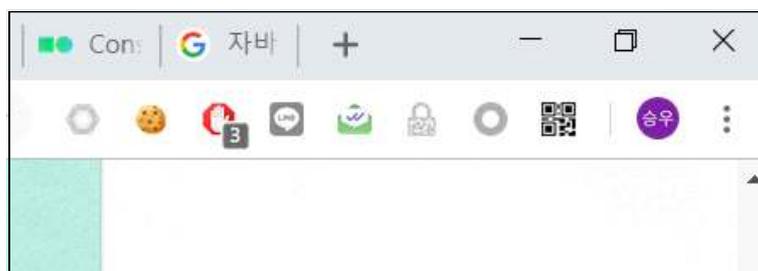
3.3 Chrome Extension

(1) Chrome Extension (크롬 확장프로그램)

SSO의 가장 큰 장점은 간편하게 여러 가지 클라이언트에서 동작한다는 것이다. 그 중 Chrome은 2015년 1월 MS사의 Internet Explorer를 제치고 전 세계 웹 브라우저 시장 1위를 차지했다.

Chrome이 IE를 제칠 수있던 요인 중 하나가 Chrome에서만 제공하는 확장프로그램인데 웹 기술인 HTML, JavaScript, CSS 등을 사용해서 개발할 수 있다.

특징적으론 Google에서 제공하는 chrome전용 google API를 활용할 수있다는 점인데 이미 많은 확장프로그램이 상용화되어 많은 유저들에게 사용되고 있다.



[그림 3.3.1] 크롬 상단 확장프로그램 UI

```

chrome extension의 manifest.json 설정
<> qrcode.html  manifest.json x
1  {
2  "update_url": "https://clients2.google.com/service/update2/crx"
3
4  "manifest_version": 2,
5  "name": "__MSG_application_title__",
6  "version": "1.0",
7  "default_locale": "ko",
8  "description": "login quickly using qr",
9  "icons": {
10   "32": "images/qr_logo_32x32_on.png",
11   "48": "images/qr_logo_48x48_on.png",
12   "96": "images/qr_logo_96x96_on.png"
13 },
14 "browser_action": {
15   "default_icon": "images/qr_logo_16x16.png",
16   "default_title": "__MSG_application_title__",
17   "default_popup": "popup.html"
18 },
19 "background": {
20   "scripts": ["js/md5.min.js", "js/jquery.min.js", "js/propertie
21 ],
22 "permissions": ["gcm", "storage", "notifications", "tabs", "htt
23

```

3.4 실행화면

(1) Laravel기반 client



[그림 3.4.1] 구동시 화면

구동 시 [그림 3.4.1]에서 로그인, QR 로그인, 등록등을 할 수 있으며 처음 등록 할 때 기입된 등록정보로 QR code가 발급이 된다. 발급 후에는 QR code로만으로도 로그인이 바로 가능하다.



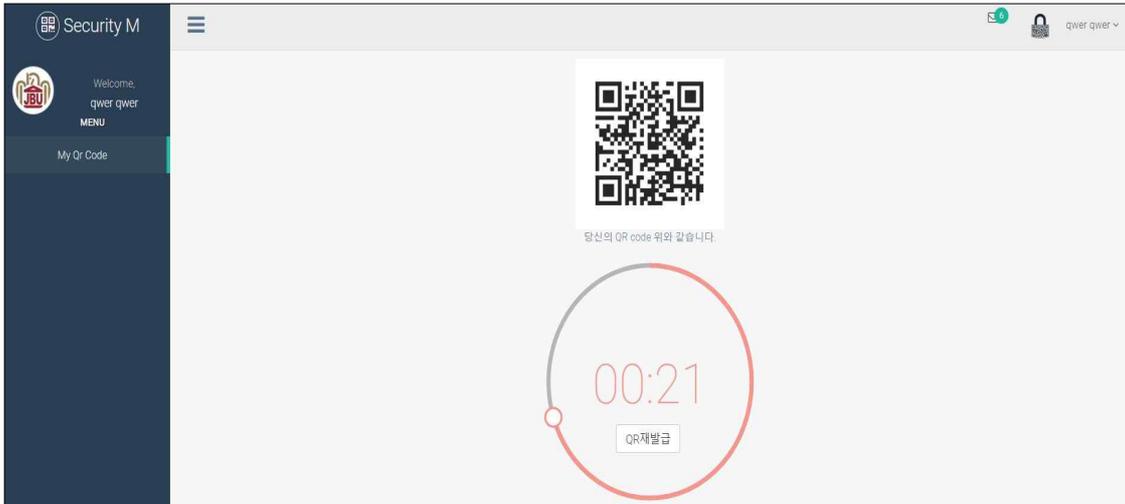
[그림 3.4.2] QR 로그인 시도



[그림 3.4.3] 로그인 된 화면

[그림 3.4.2], [그림 3.4.3] 와 같이 ID, PASSWD나 QR code로 로그인시 동일하게 로그인이 가능하다.

데이터베이스에는 Bcrypt로 암호화가 되어 저장된다. 이 암호화문자열을 가지고 QR code가 생성이 되는데 암호화가 되었다고는 하지만 유출되었을 시 피해를 막기가 힘든 것이 당연하다. 이와 관련해서 고민을 하다가 QR code가 복제, 유출될 위험을 피하기 위해 OTP(One Time Password)와 같이 QR code가 갱신되는 기능을 추가 했다.



[그림 3.4.4] QR code 갱신

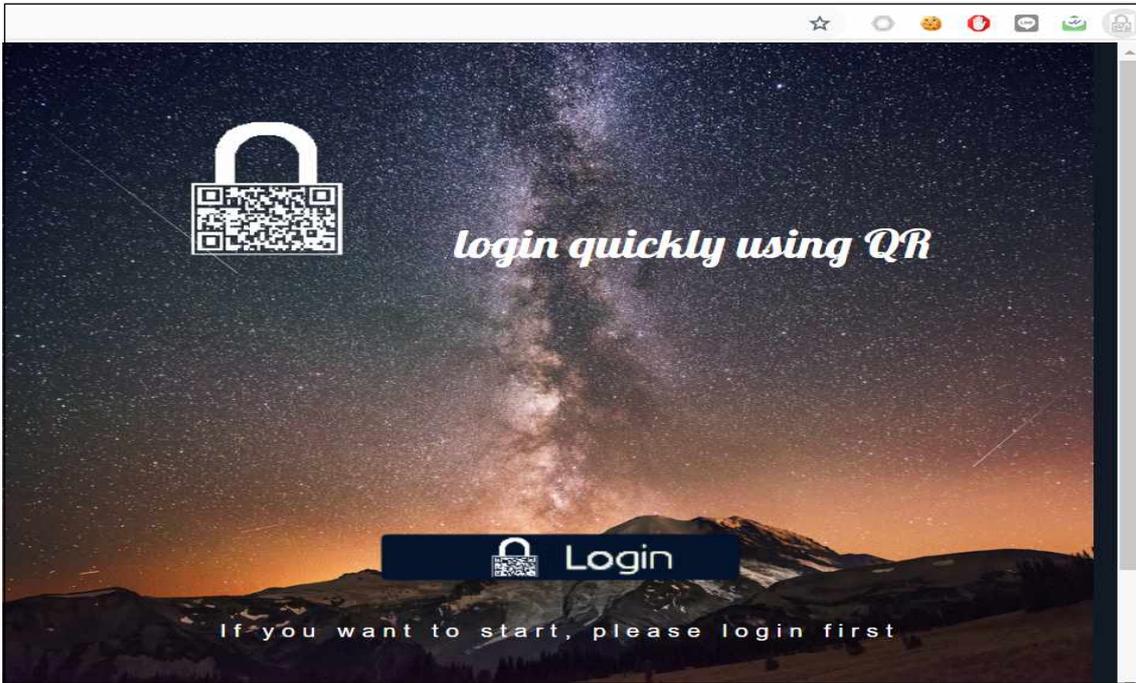
위 [그림 3.4.4]과 같이 로그인 후 QR code 발급페이지를 가면 30초를 주기로 QR code가 갱신되는 것을 확인할 수 있다. 이로 인해 유출시 사고를 미연에 방지할 수 있으며 보다 안전하게 사용이 가능하다.



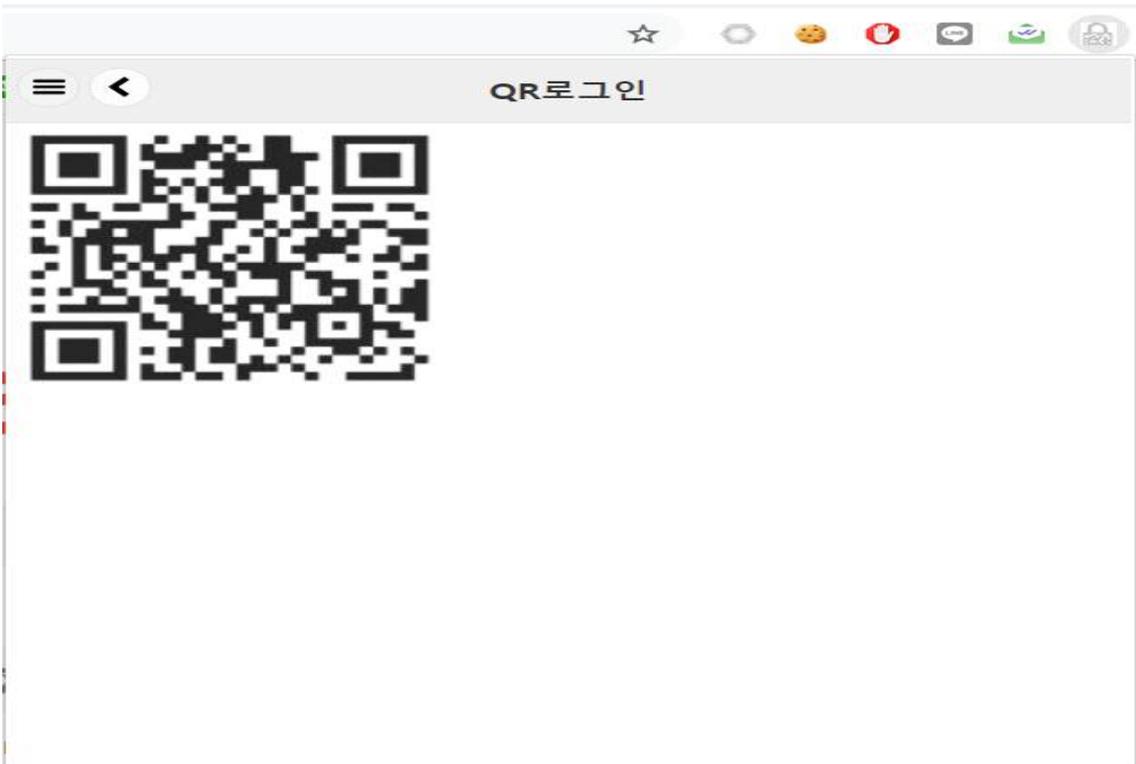
[그림 3.4.5] QR code 만료시 에러

QR code가 갱신이 되면 원래 사용하던 QR code는 폐기된다. 즉 데이터베이스 상에서는 완전 사라지고 위와 같은 에러가 출력된다.

(2) Chrome Extension기반 client



[그림 3.4.6] 크롬 확장프로그램 초기 화면



[그림 3.4.7] QR코드 출력 화면

4. 결 론

4.1 기대효과

QR Code와 SSO(Single Sign On)인증 시스템을 통해 로그인 기능을 간소화 시키고 번거로움을 감소시키고, 또한 QR Code 스캔을 통한 결제시스템 등 다양한 방향으로 적용 가능하다.

모바일 어플리케이션 개발로 한층 더 간편하게 사용 가능하다. 또한 QR Code 스캔을 통한 결제 시스템 구현이 가능하고 사원증, 출입증 등 물리보안에서도 활용 가능하다.

4.2. 향후 개선 방향

QR Code를 통한 광고 및 결제 시스템 등을 추가하여 보다 다양한 서비스를 편리하게 제공함으로써 사용자들의 만족도를 상승시키고, 단일 ID와 PW만을 사용함으로써 SSO의 유용성을 입증한다.

인증 방식을 JWT 난수화 토큰 인증방식과 결합하여 더욱 보안에 만전을 기할 수 있고 QR Code 인증방식을 모듈화하여 다른 여러 프로젝트에 바로 결합하여 개발의 편리함과 사용의 편리함까지 충족시킬수 있다고 생각한다.

5. 첨부

5.1 소스 코드

(1) Laravel

웹페이지 소스(첫 화면)
<pre><!DOCTYPE html> <html lang="en"> <head> <meta charset="utf-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="csrf-token" content="{{ csrf_token() }}" /> <meta name="viewport" content="width=device-width, initial-scale=1"> <title>@yield('title')</title> <link href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css" rel="stylesheet"> <link href="https://cdnjs.cloudflare.com/ajax/libs/bootswatch/3.3.5/flatly/bootstrap.min .css" rel="stylesheet"> <link href="//cdn.datatables.net/1.10.15/css/jquery.dataTables.min.css" rel="stylesheet"> <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.1/css/font-awesome.mi n.css"> <link href="css/style.css" rel="stylesheet"> <link rel="stylesheet" href="node_modules/bxslider/dist/jquery.bxslider.css"> <!-- Fonts --> <link href="https://fonts.googleapis.com/css?family=Raleway:100,600" rel="stylesheet" type="text/css"> <!-- Styles --> <style> html, body { background-color: #fff; color: #636b6f; font-family: 'Raleway', sans-serif; font-weight: 100; height: 100vh; margin: 0;</pre>

```
}
.full-height {
  height: 100vh;
}
.flex-center {
  align-items: center;
  display: flex;
  justify-content: center;
}
.position-ref {
  position: relative;
}
.top-right {
  position: absolute;
  right: 80px;
  top: 80px;
}
.content {
  text-align: center;
}
.title {
  font-size: 84px;
}
.text {
  font-size: 50px;
}
.links > a {
  color: #636b6f;
  padding: 0 25px;
  font-size: 12px;
  font-weight: 600;
  letter-spacing: .1rem;
  text-decoration: none;
  text-transform: uppercase;
}
.m-b-md {
  margin-bottom: 30px;
}
body {
  padding-top: 100px;
```

```

    }
    </style>
    @yield('style')
</head>
<body>
    <nav class="navbar navbar-default navbar-fixed-top">
        <div class="container">
            <!-- Brand and toggle get grouped for better mobile display -->
            <div class="navbar-header">
                <button type="button" class="navbar-toggle"
data-toggle="collapse" data-target="#navbar-collapse-1">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand"
href="{{url('/')}}"><h2>Security-M</h2></a>
            </div>
            <div class="collapse navbar-collapse"
id="navbar-collapse-1">
                <ul class="nav navbar-nav navbar-right">
                    @if (!Sentinel::getUser())
                        <li><a href="{{ url('login')
}}"><h3>Login</h3></a></li>
                        <li><a href="{{
url('qrLogin-option1') }}"><h3>Profile</h3></a></li>
                        <li><a href="{{ url('qrLogin') }}"><h3>Qr Login</h3></a></li>
                        <li><a href="{{ url('register')
}}"><h3>Register</h3></a></li>
                    @else
                        <li><a href="{{ url('/')
}}">Home</a></li>
                        <li><a href="{{url('my-qr-code')}}">My Qr Code</a></li>
                        <li><a href="{{url('user/logout/now')}}">Logout</a></li>
                    @endif
                </ul>
            </div>
        </div><!-- /.container-fluid -->
    </nav>

```

```

<div class="container">
    @yield('content')
</div>
<hr/>
<div class="container">
    <p><h4>2018. Created by Security M</h4></p>
    <br/>
</div>
<!-- Scripts -->
<script
src="//cdnjs.cloudflare.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
<script
src="//cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/3.3.4/js/bootstrap.min.js">
</script>
<script
src="//cdn.datatables.net/1.10.15/js/jquery.dataTables.min.js"></script>
<script type="text/javascript">
$.ajaxSetup({
    headers: {
        'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')
    }
});
</script>

<script
src="node_modules/bxslider/dist/jquery.bxslider.min.js"></script>
<script type="text/javascript">
$(document).ready(function(){
    $('.bxslider').bxSlider();
});
</script>
@yield('scripts')
</body>
</html>

```

로그인 페이지

```
@extends('frontLayout.app')
@section('title')
Login
@stop
@section('content')
<div class = "container">
  <div class="wrapper">
    @if (Session::has('message'))
      <div class="alert
alert-{{(Session::get('status')== 'error')?'danger':Session::get('status')}} "
alert-dismissible fade in id="sessions-hide">
        <a href="#" class="close" data-dismiss="alert"
aria-label="close">&times;</a>
        <strong>{{Session::get('status')}}</strong> {!! Session::get('message') !!}
      </div>
    @endif
    {{ Form::open(array('url' => route('login'), 'class' => 'form-horizontal
form-signin','files' => true)) }}
    <h3 class="form-signin-heading">Please Login</h3>
    <hr class="colorgraph"><br>
    {!! csrf_field() !!}
    <div class="form-group {{ $errors->has('email') ? 'has-error' : ''}}">
      <div class="col-sm-12">
        {!! Form::text('email', null, ['class' => 'form-control','placeholder
'=>'E-mail']) !!}
        {!! $errors->first('email', '<p class="help-block">:message</p>') !!}
      </div>
    </div>
    <div class="form-group {{ $errors->has('password') ? 'has-error' : ''}}">
      <div class="col-sm-12">
        {!! Form::password('password', ['class' =>
'form-control','placeholder '=>'Password']) !!}
        {!! $errors->first('password', '<p
class="help-block">:message</p>') !!}
      </div>
    </div>
    <button class="btn btn-lg btn-primary btn-block" name="Submit"
```

```
value="Login" type="Submit">Login</button>
  <div class="login-register">
    <a href="{{url('register')}}">Register</a>
    <a href="{{url('password/reset')}}">Forget Password</a>
    @if ($errors->has('global'))
      <span class="help-block danger">
        <strong style="color:red" >{{ $errors->first('global')
}}</strong>
      </span>
    @endif
  </div>
</form>
</div>
</div>
@endsection
@section('scripts')
@endsection
```

QR 로그인 페이지

```
@extends('frontLayout.app')
@section('title')
Starter Qr login
@stop
@section('styles')
@stop
@section('content')
    <!-- Header -->
    <div class="container-fluid header_se">
        <div class="col-md-2">
        </div>
        <div class="col-md-8">
            @if(!Sentinel::getUser())
                <div class="row">
                    <div id="reader" class="center-block" style="width:300px;height:250px">
                    </div>
                </div>
                <div class="row">
                    <div id="message" class="text-center">
                    </div>
                </div>
            @else
                <h1>Hallo! {{Sentinel::getUser()->first_name}}</h1>
            @endif
        </div>
        <div class="col-md-2">
        </div>
    </div>
    <!-- /.Header -->
@endsection
@if( !Sentinel::getUser())
@section('scripts')
<script type="text/javascript" src="{{
URL::asset('/qr_login/jsqrcode-combined.min.js') }}"></script>
<script type="text/javascript" src="{{ URL::asset('/qr_login/html5-qrcode.min.js')
}}"></script>
<script type="text/javascript">
    $('#reader').html5_qrcode(function(data){
        $('#message').html('<span class="text-success send-true">Scanning
```

```

now....</span>');
    if (data!='') {
        $.ajax({
            type: "POST",
            cache: false,
            url : "{{action('QrLoginController@checkUser')}}",
            data: {data:data},
            success: function(data) {
                console.log(data);
                if (data==1) {
                    //location.reload()
                    $(location).attr('href', '{{url('/')}}');
                }else{
                    return confirm('등록된 QR code가 아닙니다');
                }
                //
            }
        })
    }else{return confirm('There is no data');}
},
function(error){
    $('#message').html('Scaning now ....' );
}, function(videoError){
    $('#message').html('<span class="text-danger camera_problem"> there was
a problem with your camera </span>');
}
);
</script>
@endsection
@endif

```

등록 페이지

```
@extends('frontLayout.app')
@section('title')
Register
@stop
@section('content')
<div class = "container">
  <div class="wrapper">
    <div class="panel-heading">
      <div class="panel-title text-center">
        <h1 class="title">Register</h1>
        <hr />
      </div>
    </div>
    @if (Session::has('message'))
      <div class="alert
alert-{{(Session::get('status')== 'error')?'danger':Session::get('status')}} "
alert-dismissible fade in id="sessions-hide">
        <a href="#" class="close" data-dismiss="alert"
aria-label="close">&times;</a>
        <strong>{{Session::get('status')}}!</strong> {!! Session::get('message') !!}
      </div>
    @endif
    {{ Form::open(array('url' => route('register'), 'class' => 'form-horizontal
form-signin','files' => true)) }}
    {!! csrf_field() !!}

    <div class="form-group {{ $errors->has('first_name') ? 'has-error' :
}}">
      <label for="first_name" class="cols-sm-2 control-label">First
Name</label>
      <div class="cols-sm-10">
        <div class="input-group">
          <span class="input-group-addon"><i class="fa fa-user fa"
aria-hidden="true"></i></span>
          {!! Form::text('first_name', null, ['class' =>
'form-control','placeholder '=>'Enter your first name']) !!}
        </div>
        {!! $errors->first('first_name', '<p
class="help-block">:message</p>') !!}
      </div>
    </div>
    <div class="form-group {{ $errors->has('last_name') ? 'has-error' :
```

```

    "}}">
        <label for="last_name" class="cols-sm-2 control-label">Last
Name</label>
        <div class="cols-sm-10">
            <div class="input-group">
                <span class="input-group-addon"><i class="fa fa-user fa"
aria-hidden="true"></i></span>
                {!! Form::text('last_name', null, ['class' =>
'form-control', 'placeholder' => 'Enter your last name']) !!}
            </div>
            {!! $errors->first('last_name', '<p
class="help-block">:message</p>') !!}
        </div>
    </div>
    <div class="form-group {{ $errors->has('email') ? 'has-error' : ''}}">
        <label for="email" class="cols-sm-2 control-label">Your
Email</label>
        <div class="cols-sm-10">
            <div class="input-group">
                <span class="input-group-addon"><i class="fa fa-envelope fa"
aria-hidden="true"></i></span>
                {!! Form::email('email', null, ['class' =>
'form-control', 'placeholder' => 'E-mail']) !!}
            </div>
            {!! $errors->first('email', '<p class="help-block">:message</p>')
!!}
        </div>
    </div>
    <div class="form-group {{ $errors->has('password') ? 'has-error' :
''}}">
        <label for="password" class="cols-sm-2
control-label">Password</label>
        <div class="cols-sm-10">
            <div class="input-group">
                <span class="input-group-addon"><i class="fa fa-lock fa-lg"
aria-hidden="true"></i></span>
                {!! Form::password('password', ['class' =>
'form-control', 'rel' => 'gp', 'data-size' => '10', 'data-character-set' => 'a-z,A-Z,0-9,#',
'placeholder' => 'Enter your Password']) !!}
            </div>
            {!! $errors->first('password', '<p
class="help-block">:message</p>') !!}

```

```

        </div>
    </div>
    <div class="form-group {{ $errors->has('password_confirmation') ?
'has-error' : ''}}">
        <label for="confirm" class="cols-sm-2 control-label">Confirm
Password</label>
        <div class="cols-sm-10">
            <div class="input-group">
                <span class="input-group-addon"><i class="fa fa-lock fa-lg"
aria-hidden="true"></i></span>
                {!! Form::password('password_confirmation', ['class' =>
'form-control', 'rel'=>'gp' , 'data-size'=>'10' , 'data-character-set'=>'a-z,A-Z,0-9,#'
, 'placeholder '=>'Confirm your Password']) !!}

                </div>
                {!! $errors->first('password_confirmation', '<p
class="help-block">:message</p>') !!}
            </div>
        </div>
    <div class="form-group {{ $errors->has('password') ? 'has-error' :
''}} ">
        <button class="btn btn-primary btn-lg btn-block register-button"
type="submit" >Register</button>

    </div>
    <div class="login-register">
        <a href="{{url('login')}}">Login</a>
        @if ($errors->has('global'))
        <span class="help-block danger">
            <strong style="color:red" >{!! $errors->first('global')
}}</strong>
        </span>
        @endif
    </div>
    {{ Form::close() }}
</div>
</div>
@endsection
@section('scripts')
@endsection

```

로그인 컨트롤러

```
<?php
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;
use App\MyBroker;
use Illuminate\Foundation\Auth\AuthenticatesUsers;
use Illuminate\Http\Request;

class LoginController extends Controller
{
    use AuthenticatesUsers;

    protected $redirectTo = '/home';

    public function __construct()
    {
        $this->middleware('guest')->except('logout');
    }
    public function login(Request $request, MyBroker $myBroker)
    {
        $this->validateLogin($request);

        if($myBroker->loginUser($request->get('email'),$request->get('password'))){

            if ($this->hasTooManyLoginAttempts($request)) {
                $this->fireLockoutEvent($request);
                return $this->sendLockoutResponse($request);
            }
            if ($this->attemptLogin($request)) {
                return $this->sendLoginResponse($request);
            }
        }

        $this->incrementLoginAttempts($request);
        return $this->sendFailedLoginResponse($request);
    }
    public function logout(Request $request)
    {
        $broker = new MyBroker();
        $broker->logout();
        $this->guard()->logout();
        $request->session()->flush();
        $request->session()->regenerate();
        return redirect('/');
    }
}
```

QR 로그인 컨트롤러

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use Sentinel;
use App\User;
class QrLoginController extends Controller
{
    public function index(Request $request) {

        return view('auth.QrLogin');
    }
    public function indexoption2(Request $request) {

        return view('auth.QrLogin2');
    }
    public function ViewUserQrCode($value='')
    {
        return view('backEnd.users.viewqrcode');
    }
    public function checkUser(Request $request) {
        $result =0;
        if ($request->data) {
            $user =
User::where('QRpassword',$request->data)->first();
            if ($user) {
                Sentinel::authenticate($user);
                $result =1;
            }else{
                $result =0;
            }
        }

        return $result;
    }
    public function QrAutoGenerate(Request $request)
    {
        $result=0;
        if ($request->action = 'updateqr') {
            $user = Sentinel::getUser();
            if ($user) {
```

```
$qrLogin=bcrypt($user->personal_number.$user->email.str_random(40));
        $user->QRpassword= $qrLogin;
        $user->update();
        $result=1;
    }

}

return $result;
}
}
```

```

SSO server
<?php

namespace App;
use SSO/Server

class MySSOserver extends Server
{
    /**
     * 등록된 SSO broker
     * 원하면 소스코드에 추가 가능
     */
    private static $brokers = [
        'Yong' => ['secret'=>'patricksaint'],
    ];
    public function getBrokerInfo($brokerId)
    {
        return isset(self::$brokers[$brokerId]) ? self::$brokers[$brokerId] : null;
    }

    public function authenticate($username, $password)
    {
        if (!isset($username)) {
            return ValidationResult::error("username isn't set");
        }
        if (!isset($password)) {
            return ValidationResult::error("password isn't set");
        }
        if(Auth::attempt(['email' => $username, 'password' => $password])){
            return ValidationResult::success();
        }
        return ValidationResult::error("can't find user");
    }
    public function getUserInfo($username)
    {
        $user = User::where('email',$username)->first();
        return $user ? $user : null;
    }
    public function getUserById($id){
        return User::findOrFail($id);
    }
}
}

```

```

sso client(broker)
<?php

namespace App;
use Illuminate\Support\Facades\Auth;
use SSO\Broker;

class MyBroker extends Broker
{
    public function __construct()
    {

Parent::__construct(env('SSO_SERVER_URL'),env('SSO_CLIENT_ID'),env("SSO_CLIENT_SECRET"));
        $this->attach(true);
    }
    protected function request($method, $command, $data = null)
    {
        if (!$this->isAttached()) {
            throw new NotAttachedException('No token');
        }
        $url = $this->getRequestUrl($command, !$data || $method === 'POST' ? [] : $data);
        $ch = curl_init($url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
        curl_setopt($ch, CURLOPT_CUSTOMREQUEST, $method);
        curl_setopt($ch, CURLOPT_HTTPHEADER, ['Accept: application/json', 'Authorization: Bearer ' . $this->getSessionID()]);
        if ($method === 'POST' && !empty($data)) {
            $post = is_string($data) ? $data : http_build_query($data);
            curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
        }
        $response = curl_exec($ch);
        if (curl_errno($ch) != 0) {
            $message = 'Server request failed: ' . curl_error($ch);
            throw new Exception($message);
        }
        $statusCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);
        list($contentType) = explode(':', curl_getinfo($ch, CURLINFO_CONTENT_TYPE));

```

```
}  
public function loginUser($username, $password){  
    try{  
        $this->login($username, $password);  
    }  
    catch(NotAttachedException $e){  
        return false;  
    }  
    catch(Exception $e){  
        return false;  
    }  
    return true;  
}  
public function loginCurrentUser($returnUrl = '/home')  
{  
    if ($user = $this->getUserInfo()) {  
        Auth::loginUsingId($user['id']);  
        return redirect($returnUrl);  
    }  
}  
}
```

```
ENV
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:+/nzpw0edE8Qoe71uQs3x+8iT99gimLjMUEzePyioFs=
APP_DEBUG=true
APP_LOG_LEVEL=debug
APP_URL=http://106.10.46.131:8080

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=patrick
DB_USERNAME=root
DB_PASSWORD=root

SSO_SERVER_URL=http://localhost/server/public/
SSO_CLIENT_ID=yong
SSO_CLIENT_SECRET=patrickssaint

BROADCAST_DRIVER=log
CACHE_DRIVER=file
SESSION_DRIVER=file
QUEUE_DRIVER=sync
```

(1) Chrome Extension

```
웹페이지 소스(첫 화면)
<!DOCTYPE html>
<html lang="en">
<head>
  <title>QRCode -login quickly using QR</title>
  <link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
  <link rel="stylesheet" href="./css/uikit.docs.min.css" />
  <link rel="stylesheet" href="./css/uikit.almost-flat.min.css" />
  <link rel="stylesheet" href="./css/jquery.autocomplete.css" />
  <link rel="stylesheet" href="./css/index.css" />
  <script type="text/javascript" src="js/properties.js"></script>
  <style type="text/css" >
    body{ text-align:center; }
    #wrapper {
      width:500px;
      margin : 0 auto ;
      text-align:left;
    }
    p {
      font-family: 'Lobster', cursive;
    }
    p.a {
      position: absolute;
      top:60px;
      left:189px;
      font-size: 36px;
      color:#FFF;
      margin-right:-17px;
    }
  </style>
</head>
<body>

  <div id="mainframe">
    <div>
      <span class="logo_r" style="margin-top:100px;" id="wrapper">
        <div style="margin-top:-30px;">
          
          <p class="a">login quickly using QR</p>
        </div>
      </span>
    </div>
  </div>
</body>
</html>
```

```
        </div>
        <br style="clear:both;">

        </span>
    <br/>

    <div class="logo_footer">
        <div><a href="/html/login.html"></a></div>
        <div class="logo_footer_name" >If you want to start, please login
first</div>
    </div>

</div>

</div>
<div class="tm-footer">
    <div class="uk-container uk-container-center uk-text-center">

        <ul class="uk-subnav uk-subnav-line">
            <li><a href=".">Home</a></li>
            <li><a target="_top" href="">Enterprise Solution</a></li>
            <li><a href="/term">Term of Service</a></li>
            <li><a href="/privacy">Privacy Policy</a></li>
        </ul>
        <div class="uk-panel">
            <a href="."><span
style="margin-left:20px;">Copyright&copy;2018</span></a>
        </div>

    </div>
</div>
<div id="fb-root"></div>
<script type="text/javascript" src="js/popup.js"></script>
</body>
</html>
```

Login.js (로그인)

```
Properties.set(Properties.KEY_ACTION,Properties.ACTION_LOGIN);
Properties.get(Properties.KEY_REMEMBER,function(result){
    //console.log(result);
    if (result[Properties.KEY_REMEMBER]){

Properties.get([Properties.KEY_LOGIN,Properties.KEY_PASSWORD],function(result){
    //console.log(result);
    if (result[Properties.KEY_LOGIN]){
        $('#txt_email').val(result[Properties.KEY_LOGIN]);

$('#txt_password').val(result[Properties.KEY_PASSWORD]);
    }
    });
}
});
function initHTML() {
    h1.text(getMessageLocale('title_login'));
    isConnected();
    $('#txt_email').attr('placeholder',getMessageLocale('hint_login'));
    $('#txt_password').attr('placeholder',getMessageLocale('hint_password'));
    $('#remember').text(getMessageLocale('remember'));
    $('#btn_submit').text(getMessageLocale('btn_login'));
}
$( document ).ready(function() {
    $('#btn_submit').on('click', function(){
        //console.log('click');
        $.mobile.loading( 'show', { theme: "a", textVisible: true });
        chrome.extension.sendMessage({type: "login", username:
$('#txt_email').val(),password:
$('#txt_password').val(),remember:$('#chck_rememberme').is(":checked")});
    });
});
```

```
main.js (로그인 후 첫 화면)
Properties.set(Properties.KEY_ACTION,Properties.ACTION_MAIN);
chrome.extension.onMessage.addListener(
    function(request, sender, sendResponse) {
        //console.log(request);
        switch (request.type) {
            case "active":
Properties.set(Properties.KEY_ACTION,Properties.ACTION_MAIN);
                break;
            }
        });
$( document ).ready(function() {
    chrome.extension.sendMessage({type: "main"});
    $('.tab').on('click', function(){
        $('.tab').removeClass('ui-btn-active');
        var self = $(this);
        self.addClass('ui-btn-active');
        $('.tab-content').hide();
        var content = $(self.attr('href'));
        content.show();
        var heightList =
$(window).height()-$('.header').outerHeight()-$('#navbar').outerHeight();
        var title = content.find('.title');
        var getPro = content.find('#get-pro');
        var baner = content.find('.banner-bottom');

content.find('.ui-listview').height(heightList-title.outerHeight()-getPro.outerHeight(true)-baner.outerHeight(true));

Properties.set(Properties.KEY_CURRENT_TAB_POSITION,self.attr('data-tab'));
        return false;
    });
    Properties.remove(Properties.KEY_TEXT_NOTIFY_JSON);
    Properties.remove(Properties.KEY_COUNT_NOTIFY);
});
function loading() {
    $.mobile.loading( 'show', { theme: "b", text: getMessageLocale('loading'),
textVisible: true });
}
function initHTML() {
    loading();
}
```

```

    h1.text(getMessageLocale('chats'));
    isConnected();
    $('#link-sites').text(getMessageLocale('qrcode'));
    $('#link-active').text(getMessageLocale('active'));
    $('#link-archive').text(getMessageLocale('archive'));
}

```

```

content.js (로그인 후 첫 화면)
var accessToken;
var language = window.navigator.userAgent || window.navigator.language;
var localized;
chrome.extension.sendMessage({
    type: "locale"
});

var h1 = $('.header h1');

chrome.extension.sendMessage({
    type: "access_token"
});

chrome.runtime.onMessage.addListener(
    function(request, sender, sendResponse) {
        //console.log(request);
        switch (request.type) {
            case "data.locale":
                localized = request.data;
                $('#qrcode').text(getMessageLocale("qrcode"));

                $('#user_profile').text(getMessageLocale("user_profile"));
                $('#about').text(getMessageLocale("about"));
                $('#logout').text(getMessageLocale("logout"));
                $('#popup_ok').text(getMessageLocale("ok"));

                $('#popup_cancel').text(getMessageLocale("cancel"));
                initHTML();
                //localizeHtmlPage();
                break;
            case "login.success":
                document.location.href =
chrome.extension.getURL('html/main.html');

```

```

                break;
            case "login.error":
                $.mobile.loading("hide");
                $("#popupDialog
h1").text(getMessageLocale("error"));
                if (request.data) {
                    $("#popupDialog
h3").text(request.data.error_msg);
                } else {
                    $("#popupDialog
h3").text(getMessageLocale("not_connection"));
                }
                $("#popupDialog").popup("open");
                break;
        }
    });
    $('#main').on('click', function() {
        document.location.href = chrome.extension.getURL('html/main.html');
    });
    $('#user_profile').on('click', function() {
        document.location.href = chrome.extension.getURL('html/profile.html');
    });
    $('#qrcode').on('click', function() {
        document.location.href = chrome.extension.getURL('html/qrcode.html');
    });
    $('#logout').on('click', function() {
        Properties.set(Properties.KEY_CHECK_LOGIN, false);
        document.location.href = chrome.extension.getURL('html/login.html');
    });
    $('#back').on('click', function() {
        window.history.back();
    });

```

5.2 참고 자료

- (1) Google API Chart (QR code)
- (2) StackOver Flow (stackoverflow.com)
- (3) Laravel-korea (<http://www.laravel-korea.org>)
- (4) 라라벨 한국어 매뉴얼 (<https://laravel.kr/docs>)
- (5) Google Chrome Extension Tutorial

5.3 발표 PPT 자료

QR코드를 통한 간편 로그인

2018.11.7

지도 교수 : 이병천 교수님

4조 Security-M
(지승우 이승용 박종범 백진이)

조원 편성



지승우

SSO, QR Code 구현
/ 서버 구축



이승용

SSO, QR Code 구현
/ 웹 페이지 구축



박종범

DB 설계 / 구축



백진이

웹 페이지 구축

3

주제 선정

◆ ID와 패스워드에 의한 로그인 방식의 획기적인 변화 필요

- ✓ 문자형 ID와 패스워드 ⇨ QR Code 등 활용
- ✓ 간편한 타겟 인식 및 암기식 보안체계의 불편 극복



◆ QR Code를 활용한 간편 로그인 기술 및 SSO(Single Sign On) 연구개발

- ✓ Laravel Framework 기반 웹 어플리케이션 개발
- ✓ QR Code 사용자 인증 방식 개발
- ✓ SSO 구현

4

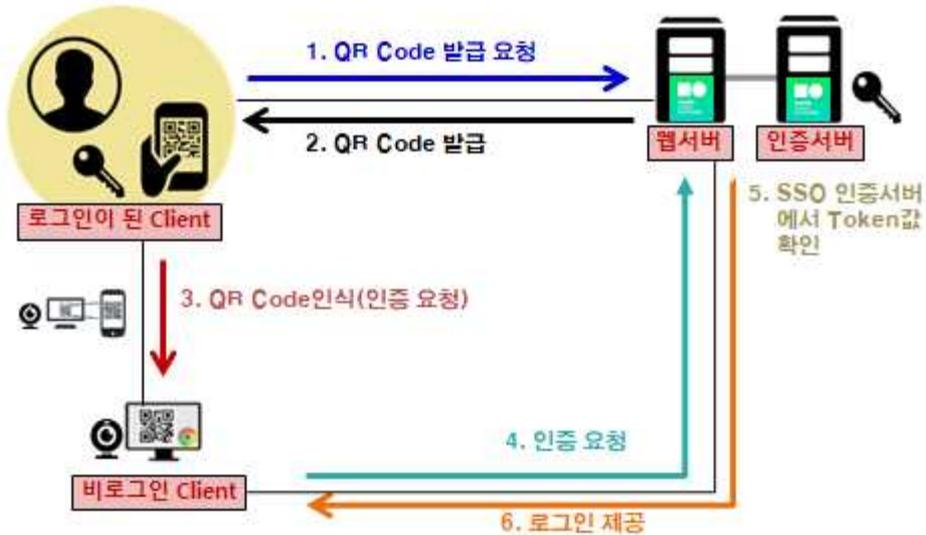
추진 일정

(단위 : 월)

구분 \ 기간(2018년)	3	4	5	6	7	8	9	10	11
주제 선정 및 계획 수립									
QR 생성 및 인식 구현									
서버 및 웹페이지 구축									
QR 인증 로그인, SSO 구현									
시스템 통합 및 결과 종합									

5

구상도



6

개발 환경 및 시스템 구축(1/12)

개발 환경

- ◆ 서버 및 DB ⇨ APM(Apache, PHP, MySQL)



- ◆ 웹 서비스 ⇨ Laravel 5.4 Framework, Chrome Extension
- ◆ 웹 호스팅 ⇨ Naver Cloud Platform Ubuntu Server

7

개발 환경 및 시스템 구축(4/12)

DB 구축 : MySQL 설치

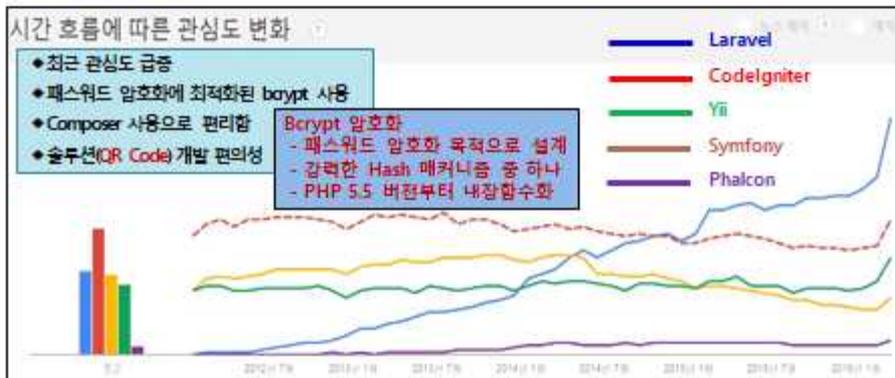
The screenshot shows the MySQL Installer interface. A window titled 'MySQL Router Configuration' is open, with the option 'Configure MySQL Router for InnoDB cluster' checked. Below it, a terminal window shows the command `mysql -u root -p` being executed. The terminal output includes the MySQL monitor welcome message and the server version: `Server version: 5.7.24-0ubuntu0.16.04.1 (Ubuntu)`. A yellow box highlights the command `mysql -u root -p` with the text '패스워드 입력' (password input).

10

개발 환경 및 시스템 구축(5/12)

Laravel 프레임워크 설치(1/3)

◆ 선정 이유



11

개발 환경 및 시스템 구축(6/12)

Laravel 프레임워크 설치(2/3)

◆ Composer 설치

- ✓ 의존성 관리를 위해 Composer 사용

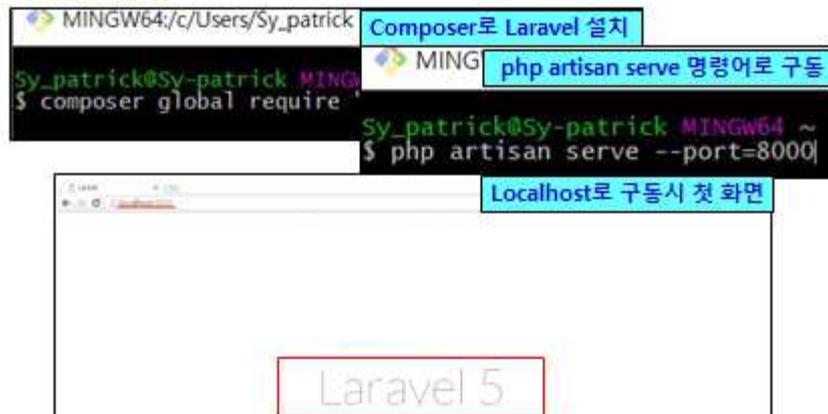


12

개발 환경 및 시스템 구축(7/12)

Laravel 프레임워크 설치(3/3)

◆ Laravel 설치



13

개발 환경 및 시스템 구축(8/12)

QR Code

```
public function QrAutoGenerate(Request $request) Bcrypt로 1회성 패스워드 생성
{
    $result=0;
    $baseqr=0;
    if ($request->action = 'updateqr') {
        $user = Sentinel::getUser();
        if ($user) {
            $qrLogin=bcrypt($user->personal_number.$user->email.str_random(40));
            $user->QRpassword= $qrLogin;
            $user->update();
            $result=1;}
    }
```

```
<div class=" text-center">
@if(Sentinel::getUser()->QRpassword)

<p> 당신의 QR code 위와 같습니다.</p>
@endif
```

DB에 암호화되어 저장된 패스워드값을 Base64이미지 인코딩

14

개발 환경 및 시스템 구축(9/12)

QR Code

```
@section('scripts') 30초 마다 QR Code 갱신
<script>
//Delete Items
var sum = 0;
$(document).ready(function(){
setTimeout("qr_timer()", 30000); // 30000ms(30초)가 경과하면 qr_timer()함수를 실행
});
function qr_timer(){
$.ajax({
type: "POST",
cache: false,
url : "{{action('QrLoginController@QrAutoGenerate')}}",
data: {action: 'updateqr'},
success: function(data) {
if (data==1) {
location.reload();
}else{
alert( 'Ups error :P ');
}}}})
```

유출 시 발생하는 위험을 미연에 방지

15

개발 환경 및 시스템 구축(10/12)

SSO(Single Sign On)



- ◆ 한 번의 인증 과정으로 여러 사이트나 서비스를 이용할 수 있는 체계
- ◆ 단일 계정 로그인 또는 단일 인증이라고도 부른다.

16

개발 환경 및 시스템 구축(11/12)

SSO 작업 : 토큰 생성 및 검증

```
protected function generateSessionId($brokerId, $token)
{
    $broker = $this->getBrokerInfo($brokerId);
    if (!isset($broker))
        return null;
    return "SSO-{$brokerId}-{$token}-" . hash('sha256', 'session' . $token .
        $broker['secret']);
}
```

SSO 서버에서 토큰 생성

```
Private static $brokers = [
    'joongbu' => ['secret'=>'789789'],
    'lee' => ['secret'=>'123123'],
    'yong' => ['secret'=>'345345'],
];
```

서버에 SSO id, passwd 등록

Name	Value	Domain	Path
sso_token_yong	ak5xrk2donc4kscgw54swws4w	localhost	/

Client로그인 후 개발자 도구(F12)에서 토큰 값 확인 가능

17

개발 환경 및 시스템 구축(12/12)

SSO 작업 : 설정값 서버 전송

```
public function __construct() {  
    parent::__construct(  
        env('SSO_SERVER_URL'),  
        env('SSO_CLIENT_ID'),  
        env("SSO_CLIENT_SECRET"));  
    $this->attach(true);  
}
```

설정 값 전송

ID, SECRET 정보를 서버로 전송

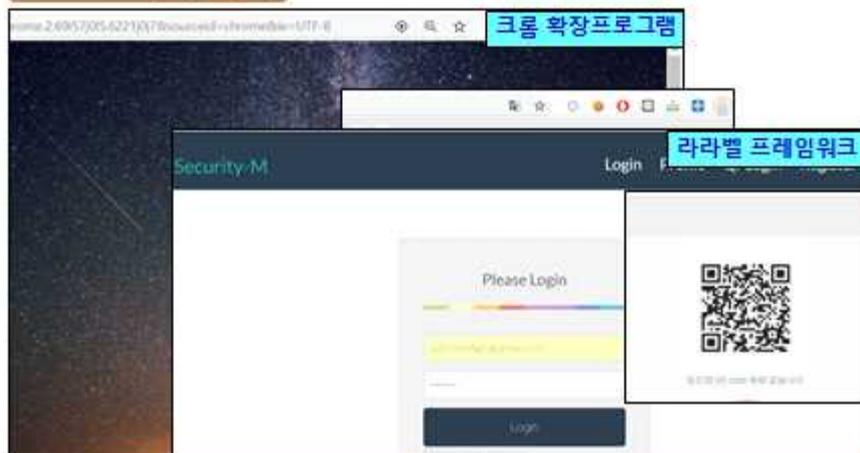
```
public function __construct($url, $broker, $secret, $cookie_lifetime = 3600) {  
    if (!$url) throw new \InvalidArgumentException("SSO server URL Error");  
    if (!$broker) throw new \InvalidArgumentException("SSO broker id Error");  
    if (!$secret) throw new \InvalidArgumentException("SSO broker secret Error");  
    $this->url = $url;  
    $this->broker = $broker;  
    $this->secret = $secret;  
    $this->cookie_lifetime = $cookie_lifetime;  
}
```

서버 URL, ID, SECRET 중 하나라도 오류시 에러코드 발생

18

개발 결과 및 시스템 운영(1/7)

SSO : 서버 접속



같은 계정으로 여러개의 서비스에 로그인 가능

19

개발 결과 및 시스템 운영(2/7)

메인 화면

- ◆ ID/PASSWD 로그인, QR code 로그인, 등록



20

개발 결과 및 시스템 운영(3/7)

등록 및 로그인

- ◆ 등록 -> 로그인 -> QR Code 확인



21

개발 결과 및 시스템 운영(4/7)

QR 코드 로그인



22

개발 결과 및 시스템 운영(5/7)

DB 검증

email	password	QRpassword	permissions	last_login	first_name	last_name
admin@admin.com	\$2y\$10\$wceikAMG8aw9C0eukLh0py7jgs...	\$2y\$10\$uomf2Wg9kas4CC0zdyuKZ84H4b...	ROLE_	2016.10.10 12:00:00	qwer	qwer
joongbu@joongbu.com	\$2y\$10\$Aubd887Tg538f7a*Qm+L8Pawf5C7w...	4R66koddECftXimtAiiFkFLDoe1XjLwY5VH5CHid	ROLE_	NULL	Seung yong	Lee

email

admin@admin.com

qwer@qwer.com

joongbu@joongbu.com

QRpassword

4r66koddECftXimtAiiFkFLDoe1XjLwY5VH5CHid

Seung yong Lee

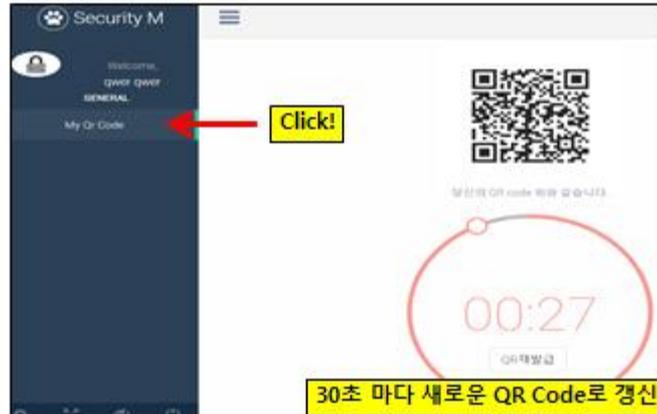
이름, 이메일, 패스워드, 암호화된 QR코드 확인 가능

23

개발 결과 및 시스템 운영(6/7)

QR Code

- ◆ 유출 및 도용 방지를 위한 QR Code 갱신



24

개발 결과 및 시스템 운영(7/7)

QR Code

- ◆ QR Code 변경 시 DB에서 QR 정보는 사라지고 예전 QR Code로는 사용이 절대 불가



25

결론 및 기대효과

◆ 결론

- ✓ QR Code를 통해 로그인 간소화 및 번거로움 감소
- ✓ 중앙 집중적인 사용자 관리가 가능하고,
단일 ID와 PW만 사용함으로써 SSO의 유용성 입증

◆ 기대효과

- ✓ 추후 QR Code 스캔을 통한 결제시스템 구현 가능
- ✓ 출입 통제 등의 다양한 방면에서 활용 가능

26



Q&A

감사합니다