

## 참빛설계학기 주차별 활동보고서

주 차	10주차
-----	------

팀 명	편--안
팀 원	남윤창

프로젝트명	NoSQL-based PC Café Integrated Management System		
활동기간	2019.10.28. ~ 2019.11.3.	활동시간	6시간
금주목표	사용자용 PC Program : Javascript를 사용한 Firebase DB와의 연동 개발		
활동내용	<p><b>User PC Program Structure</b></p> <pre> graph TD     A[초기 화면 - 사용자가 App에서 사용을 시작하기 전 PC를 부팅한 상태 - App에서 사용자의 예약을 기다리는 상태] -- "App에서 사용자가 좌석 신청을 한 경우" --&gt; B[비밀번호 입력 대기 - 사용자가 App에서 좌석 사용을 시작 했을 경우 - 사용자가 비밀번호 입력하기를 기다리는 상태]     B -- "사용자가 올바른 비밀번호를 입력한 경우" --&gt; C[사용중 - 비밀번호 입력 후 PC사용 가능해진 상태 - 사용자의 이름 - 사용자의 사용한 Point, 잔여 Point - 사용 시작 시간 - PC 종료버튼]     C -- "사용자가 종료버튼을 눌렀을시 혹은 잔여 Point가 부족할시" --&gt; A                     </pre> <p>### 초기화면 ###</p> <p>1. Javascript - Firebase DB 연동</p> <pre> &lt;script src="https://www.gstatic.com/firebasejs/6.3.5/firebase-app.js"&gt;&lt;/script&gt; &lt;script src="https://www.gstatic.com/firebasejs/6.3.5/firebase-auth.js"&gt;&lt;/script&gt; &lt;script src="https://www.gstatic.com/firebasejs/6.3.5/firebase-database.js"&gt;&lt;/script&gt; - Javascript에서 Firebase API를 사용할 수 있도록 접근 권한 부여  var firebaseConfig = {   apiKey: "AIzaSyDgYmW0tVWMfv-IVURyvQNqlNPynsG0sTQ",   authDomain: "p-y-89803.firebaseio.com",   databaseURL: "https://p-y-89803.firebaseio.com",   projectId: "p-y-89803",   storageBucket: "",   messagingSenderId: "551920152722",   appId: "1:551920152722:web:9f21c5206ca8187f" }; // Initialize Firebase firebase.initializeApp(firebaseConfig);                     </pre>		

- 기존 App에서 사용한 Firebase App과의 연동 (DB 연동)

## 2. Firebase database 변수 생성

```
database = firebase.database();
var pcCafeRef = database.ref('PCcafe/3POP/광운대');
```

- firebase DB 연동하여 변수 생성 -> database변수로 p--y DB 접근 권한 부여
- 임시로 DB의 3POP 광운대를 경로설정 하여 Test

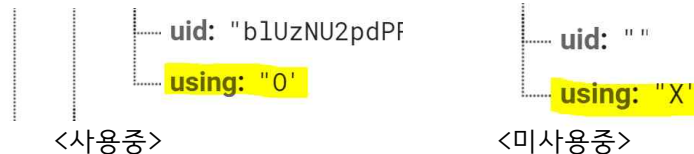
## 3. 해당 PC 정보 DB에서 열람

```
var seat_num = data.key;
document.getElementById('number_seat').innerText = seat_num;

var seatRef = firebase.database().ref('PCcafe/3POP/광운대/'+seat_num);

seatRef.child('using').on('value', function(seat_info){
    if(seat_info.val() == 'O'){
        location.replace('2.html'); }
});
```

- 해당 PC의 정보를 DB에서 읽어온 후 사용중 여부 판별
- > seatRef변수 : DB의 해당경로 접근
- > .on('value', function(seat\_info) { }) : DB의 '모든' Value의 changing event를 항상 대기
- > seat\_info.val() == 'O' : 사용중 여부 판별
- > 사용중이라면 초기화면에서 비밀번호 대기 화면으로 이동하도록 설정



- > location.replace('2.html') : 비밀번호 입력 html 파일로 전환

## ### 비밀번호 입력 대기 ###

### 1. 사용자가 비밀번호를 입력하지 않고 종료한 경우

```
if(seat_info.val() == 'X'){
    location.replace('1.html'); }
});
```

- 사용중 여부를 판별하여 사용중이 아니라면 다시 초기화면으로 돌아가도록 설정

### 2. 비밀번호를 입력한 경우

```
memberRef.orderByChild('pwd').equalTo(user_pw).on('child_added', function(user_info){
    if(uid == user_info.key){
        alert("로그인 성공!");
        location.replace("3.html");
    }
});
```

- 사용자가 본인의 비밀번호를 제대로 입력한 경우 "로그인 성공!"을 띄운 후 사용중 화면으로 넘어가도록 구현 : location.replace("3.html")

### ### 사용증 ###

#### 1. 사용자 정보 표시

- Table을 사용하여 보기 편하게 출력

```
<table cellpadding='0'>
  <tr>
    <th>Title</th>
    <th>Contents</th>
  </tr>

  <tr>
    <td>name</td>
    <td id="user_name"></td>
  </tr>
  <tr class='even'>
    <td>자리</td>
    <td id="number_seat"></td>
  </tr>

  <tr>
    <td>시작시간</td>
    <td id="start_time"></td>
  </tr>
  <tr class='even'>
    <td>사용한시간</td>
    <td id="used_time"></td>
  </tr>

  <tr>
    <td>현재포인트</td>
    <td id="current_point"></td>
  </tr>

  <tr class='even'>
    <td>사용포인트</td>
    <td id="used_point"></td>
  </tr>
</table>
```

```
document.getElementById('number_seat').innerText = seat_num;
document.getElementById('start_time').innerText = data.child('time_start').val();
document.getElementById('user_name').innerText = snap.child('name').val();
```

- 사용자의 이름, PC의 자리 번호, 시작 시간은 변하지 않기 때문에 초기에 한 번만 파이어 베이스에서 Load.

```
// user reference event
userRef.on('value', function(snap){

    cur_point = snap.child('point').val();

    document.getElementById('current_point').innerText = cur_point;
});
```

```
// pc seat reference event
seatRef.on('value', function(seat_info) {

    used_point = seat_info.child('point_used').val();
    document.getElementById('used_point').innerText = used_point;

    update_time = seat_info.child('time_used').val();
    document.getElementById('used_time').innerText = update_time;
```

- 사용한 시간, 현재 포인트, 사용 포인트는 실시간으로 파이어 베이스와 연동되어 변해야 하기 때문에 유저 정보 Reference와 좌석 정보 Reference의 변화가 생기는 이벤트가 발생할 경우 출력되는 정보를 Update.

## 2. 휴대폰, 혹은 사장님 프로그램에서 사용을 종료

```
if(seat_info.child('using').val() == 'X'){ // exit pc

    exit();

    location.replace('1.html');

}
```

-> 좌석의 사용 상태가 X로 변경되는 경우 exit 함수 실행

```
function exit() { // exit pc func
    seatRef.update({
        uid:"",
        point_used:0,
        time_used:0,
        using: "X"
    });
    userRef.update({
        seat_using:"",
        time_start:""
    })
}
```

-> 좌석과 유저 파이어 베이스의 사용과 관련된 정보들을 초기화.

## 3. 사용자가 PC에서 사용을 종료

```
<input type="button" id="exit_PC" onclick="exit()" value="EXIT"/>
```

버튼을 클릭하면 exit() 함수가 실행된다.

<p>활동사진</p>	<p>주간 활동 사진</p> 
<p>문헌자료/ 참고자료</p>	<p>[1] Firebase Database Reference(Javascript SDK) 공식문서 :  <a href="https://firebase.google.com/docs/reference/js/firebase.database.Reference.html">https://firebase.google.com/docs/reference/js/firebase.database.Reference.html</a>                  [2] Firebase Database Datasnapshot(Javascript SDK) 공식문서 :  <a href="https://firebase.google.com/docs/reference/js/firebase.database.DataSnapshot">https://firebase.google.com/docs/reference/js/firebase.database.DataSnapshot</a></p>
<p>차후 계획</p>	<p>- 9~10주차에서 개발한 사용자 Page에 대한 UI 개발(CSS)                  - 사장님이 한눈에 PC Cafe를 관리 할 수 있는 Admin Program 구현 계획 회의</p>
<p>지도교수 총평</p>	

2019. 10 . 30 .  
지도교수 (인)