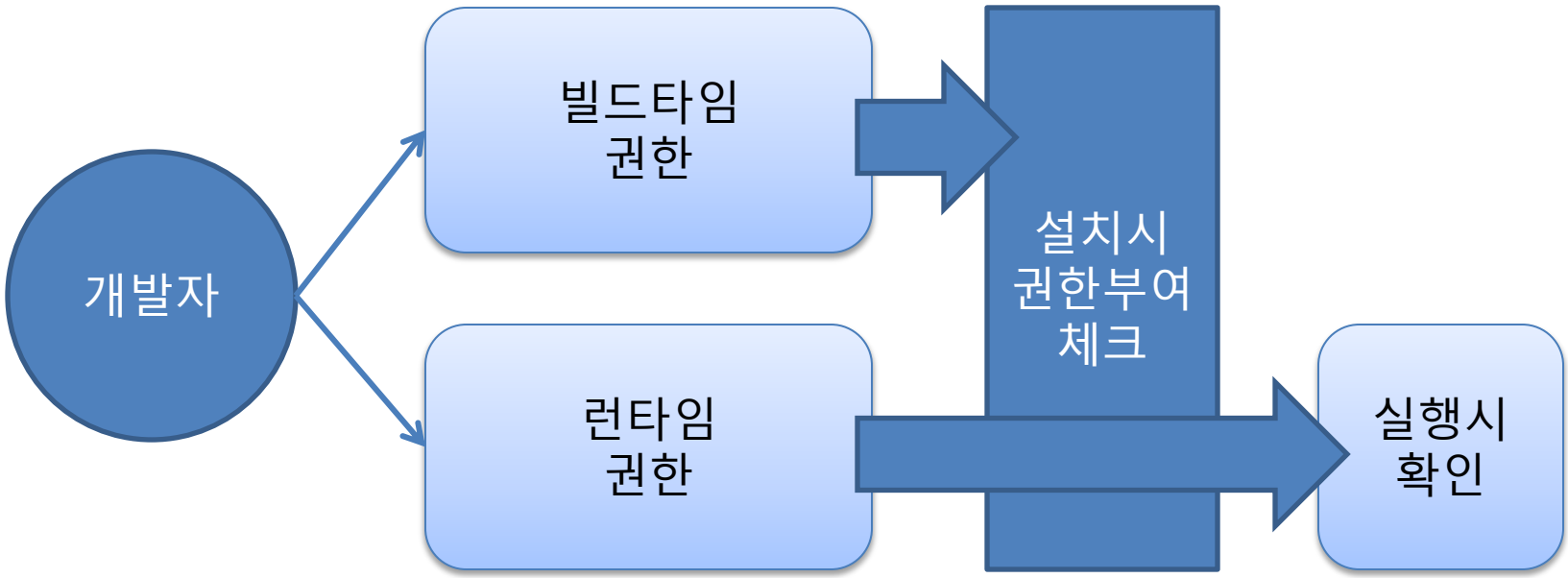
Android
016 Runtime Permission

1. Runtime Permission - Api 23 이상

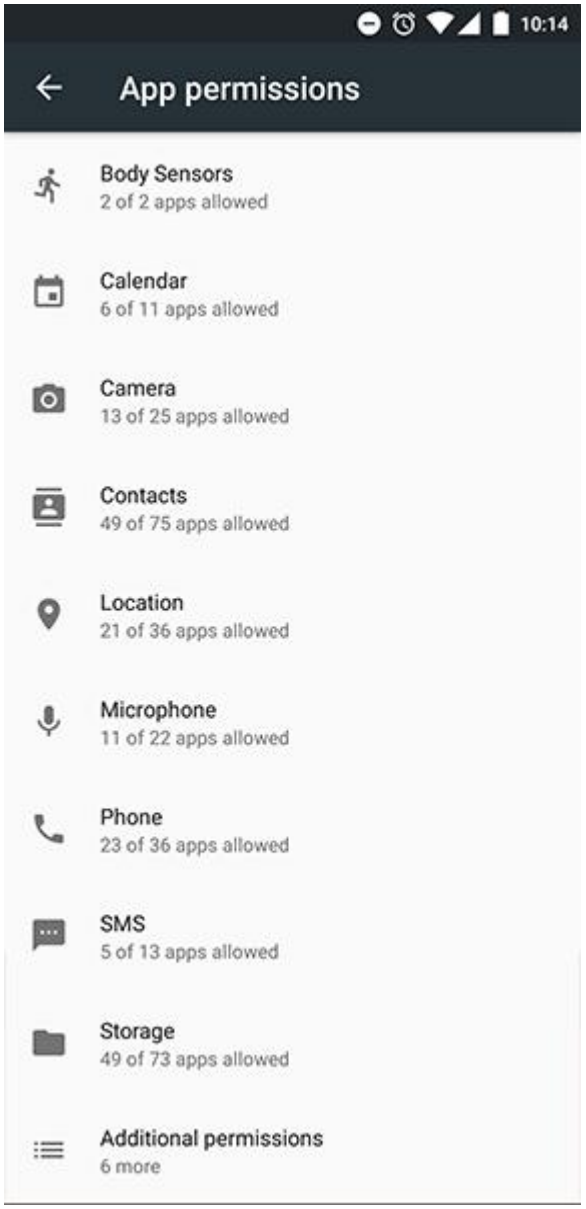
1. 안드로이드는 마시멜로우(6.0) Api Level 23 이상 부터는 사용자가 접근 권한이 필요한 기능을 수행할때, 사용자로 하여금 해당 권한을 앱에 허락 할 것인지 묻고, 개발자가 아닌 사용자가 자신의 디바이스의 접근 권한을 결정하는 형태로 권한설정 구조를 변경했다



2. Runtime Permission 종류

1. 안드로이드 Runtime Permission은 아래와 같다

Permission Group	Permissions
CALENDAR	•READ_CALENDAR •WRITE_CALENDAR
CAMERA	•CAMERA
CONTACTS	•READ_CONTACTS •WRITE_CONTACTS •GET_ACCOUNTS
LOCATION	•ACCESS_FINE_LOCATION •ACCESS_COARSE_LOCATION
MICROPHONE	•RECORD_AUDIO
PHONE	•READ_PHONE_STATE •CALL_PHONE •READ_CALL_LOG •WRITE_CALL_LOG •ADD_VOICEMAIL •USE_SIP •PROCESS_OUTGOING_CALLS
SENSORS	•BODY_SENSORS
SMS	•SEND_SMS •RECEIVE_SMS •READ_SMS •RECEIVE_WAP_PUSH •RECEIVE_MMS
STORAGE	•READ_EXTERNAL_STORAGE •WRITE_EXTERNAL_STORAGE



3. 권한 획득처리

1. 권한 획득하기 전 권한 유효성 체크
checkSelfPermission(String) != PackageManager.PERMISSION_GRANTED
현재 앱이 특정 권한을 갖고 있는지 확인 가능

2. 설명이 필요할 경우 처리
shouldShowRequestPermissionRationale(String)
권한 획득이 필요한 이유를 설명해야 한다면 다음 옵션을 추가하여 별도 처리가 가능. 사용자가 이전에 권한 요청을 거부한 경우에 true반환. 이 경우, 권한 요청을 위한 대화창에는 '다시 묻지 않기' 체크박스와 함께 표시된다. 사용자가 이를 선택하면 이후에 앱이 requestPermissions 메서드를 호출해도 권한 요청 대화창이 표시되지 않고, 바로 사용자가 해당 권한을 거부할 때와 동일 하게 콜백 함수가 호출된다.

3. 권한 획득을 위한 API
Activity.requestPermissions(String[], int)
위의 권한 중 Group과 permission 2가지를 선택적으로 던질 수 있다. 한 번에 1개가 아닌 String[] 배열로 넘겨 한 번에 필요한 permission을 한 번에 획득할 수 있다.

4. 결과처리 - onRequestPermissionResult(int, String[], int[])
권한 획득에 대한 성공/실패에 대한 정보를 담은 callback. 다음 함수 내에서 배열로 전달되므로 필요한 퍼미션이 잘 받아졌는지 확인하여 이후 처리가 가능.

4. 권한 체크 코드

1. 아래와 같은 구조로 권한을 체크할 수 있다

```
public final static int REQ_CODE = 100;

private void checkPermission() {
    // 권한이 없을 경우
    if (checkSelfPermission(this, Manifest.permission.READ_EXTERNAL_STORAGE)
                != PackageManager.PERMISSION_GRANTED) {
        // 사용자가 임의로 권한을 취소시킨 경우
        if (shouldShowRequestPermissionRationale(this, Manifest.permission.READ_EXTERNAL_STORAGE)) {
            // 권한 재요청
            requestPermissions(this, new String[]{Manifest.permission.READ_EXTERNAL_STORAGE},  REQ_CODE);
        } else {
            // 권한 요청 (최초 요청)
            requestPermissions(this, new String[]{Manifest.permission.READ_EXTERNAL_STORAGE},  REQ_CODE);
        }
    }
}
```


5. 결과처리 코드

1. 아래와 같은 구조로 권한을 체크할 수 있다

```
@Override
public void onRequestPermissionsResult(int requestCode, String permissions[], int[] grantResults) {
    switch (requestCode) {
        case PERMISSIONS_EXT:
            // If request is cancelled, the result arrays are empty.
            if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                // 동의 및 로직 처리
                Log.e(TAG, ">>> 동의함.");
            } else {
                // 동의 안함
                Log.e(TAG, ">>> 동의를 해주셔야 합니다.");
            }
            return;
        default :
        // 예외 케이스
    }
}
```

