

9. 실습 보고서
- COVID-19 정보 앱 -

보 고 자 성 명	20181151 김 지 원
보 고 자 학 과	산 업 경 영 공 학 과
담 당 교 수 성 명	정 진 만 교 수 님
과 목 명	모 바 일 프 로 그 래밍 01분 반
제 출 날 짜	2020.05.18

- 목차 -

1. 서론	3p
1) 개요	
2) 요청사항 분석	
3) 추가 사항	
4) 초기 디자인	
2. 본론	
1) res > layout	4p
① activity_main	
② activity_profile	
③ activity_diag	
④ activity_pass	
⑤ activity_info	
2) 소스 코드 원본	7p
① MainActivity.java	
② activity_profile.java	
③ activity_diag.java	
④ activity_pass.java	
⑤ activity_info.java	
3) 소스 코드 분석	12p
① MainActivity.java	
② activity_profile.java	
③ activity_diag.java	
④ activity_pass.java	
⑤ activity_info.java	
4) 결과 화면	19p
3. 결론	
1) COVID-19 정보 앱 장점	21p
2) COVID-19 정보 앱 단점	21p
3) 배운 점	22p

1. 서론

1) 개요

최근 코로나 바이러스로 인해 우리의 삶이 상당 부분이 변화하였다. 실시간으로 변하는 확진자 정보와 그에 대응하기 위한 개인의 예방책, 그리고 그 도중에 필요한 정보를 얻기 위해 검색과 다양한 어플리케이션을 이용한다. 이러한 정보를 얻을 수 있는 통합 어플리케이션을 제작하였다. 그리고 현재 한남대학교도 일부 수업을 대면 강의로 전환하면서 건물 출입 시 발열 체크와 출입자 관리를 철저하게 진행하고자 노력하고 있다. 며칠간 이를 진행해본 결과 건물을 출입할 때마다 발열 체크 확인증을 소지하고 다녀야 한다는 불편함이 있고, 단순히 체온만 확인할 뿐 다른 증상의 진단이 불가능하다는 단점이 있다. 따라서 이 불편함을 어플리케이션을 제작하여 해결할 수 있다. 이번에 제작한 어플리케이션은 개인 정보를 입력한 뒤 자가진단을 하여 자가진단 확인증을 화면에 출력할 수 있게 한다. 출입 시 해당 화면을 관리인에게 제시하거나 바코드를 출입 확인 기기에 인식시켜 출입하는 방식의 기능을 추가하였다.

2) 요점사항 분석

① Explicit Intent를 사용하여 자가진단 테스트를 만들어 결과를 Main Activity에 전달

: Explicit Intent를 사용하여 자가진단 테스트를 만들고 테스트를 완료한 결과를

“자가진단 확인증”의 Activity 혹은 “코로나 정보” Activity로 전달한다.

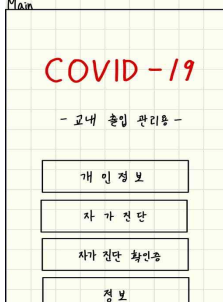
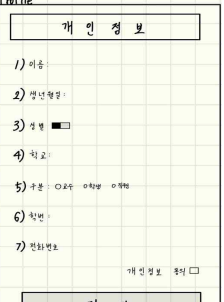
② 3개 이상의 Implicit Intent를 사용하여 다양한 정보 제공

: Implicit Intent인 call과 3가지 web 연결 기능으로 정보를 제공하였다.

3) 추가 사항

자가진단 후 그 결과를 어떤 방식으로 활용하면 유용한 어플리케이션을 만들 수 있을까를 고민한 결과 진단 결과를 교내 건물 출입 시 활용하면 좋을 것 같아 자가진단을 통과하면 자가진단 확인증을 발급해주고, 이를 활용하여 건물 출입을 가능하게 하는 방식의 기능을 추가하였다.

4) 초기 디자인

	<p>- 어플리케이션의 첫 실행 화면으로 어플리케이션의 이름과 사용할 수 있는 4가지 기능을 Button으로 추가하여 클릭 시 화면이동이 일어난다.</p>		<ul style="list-style-type: none">- Main 화면에서 첫 번째 Button인 ‘개인정보’를 클릭하면 개인정보를 입력할 수 있는 화면으로 이동하도록 Explicit Intent를 활용한다.- 개인정보를 기입하는 기능은 Edit Text와 Switch, Radio Button, Check Box으로 구현한다.- 가장 하단에 정보를 저장하는 Button을 배치한다.
-------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<p>Diagnosis</p> <p>자 가 진단</p> <p>1) 열 <input type="text"/> 26.5</p> <p>2) 의심증상 <input type="checkbox"/> 체온 <input type="checkbox"/> 기침</p> <p>3) 폐렴 위험 여부 <input type="checkbox"/></p> <p>4) 2주 내외에 해외 방문 <input type="checkbox"/></p> <p>5) 2주 내외에 해외 입국 <input type="checkbox"/></p> <p>저 장</p>	<ul style="list-style-type: none"> - Main 화면에서 두 번째 Button인 '자가진단'을 클릭하면 자가진단 테스트를 할 수 있는 화면으로 이동하도록 Explicit Intent를 활용한다. - 자가진단 테스트의 항목은 Seek Bar, Check Box를 이용하여 구현한다. - 가장 하단에 정보를 저장하는 Button을 배치한다. 	<p>Pass</p> <p>유효기간 : 2020-05-17</p> <p>자가 진단 확인증</p> <p>바 코 드</p>	<ul style="list-style-type: none"> - 자가 진단 테스트를 통과 할 시, 이동하여 출력되는 자가진단 확인증이다. - 상단에 유효기간을 출력하여 그 해당 날짜에 맞는 년, 월, 일이 출력된다. - Text View를 사용하여 확인증에 출력되어야 할 정보를 기입한다. - 마지막에 건물 출입 시 사용할 수 있는 유효한 바코드를 출력한다. (임시로 Image View를 사용하여 바코드 이미지 출력)
<p>Info</p> <p>1339 전화</p> <p>선별 진료소</p> <p>확진자 정보</p> <p>공적 마스크</p>	<ul style="list-style-type: none"> - 자가 진단 테스트의 통과 기준에 미달되었거나, Main 화면에서 네 번째 Button인 '정보'를 클릭하면 COVID-19에 관련된 정보를 얻을 수 있는 화면으로 이동하도록 Explicit Intent를 활용한다. - 4가지 정보를 얻을 수 있는 Button을 추가한다. - Implicit Intent를 이용하여 Button 클릭 시 전화가 연결되고, 해당 정보가 있는 Web에 접속된다. 		

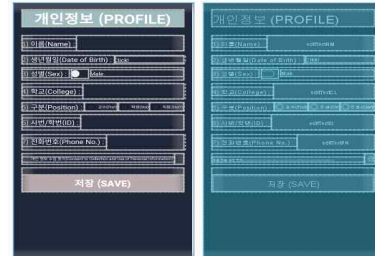
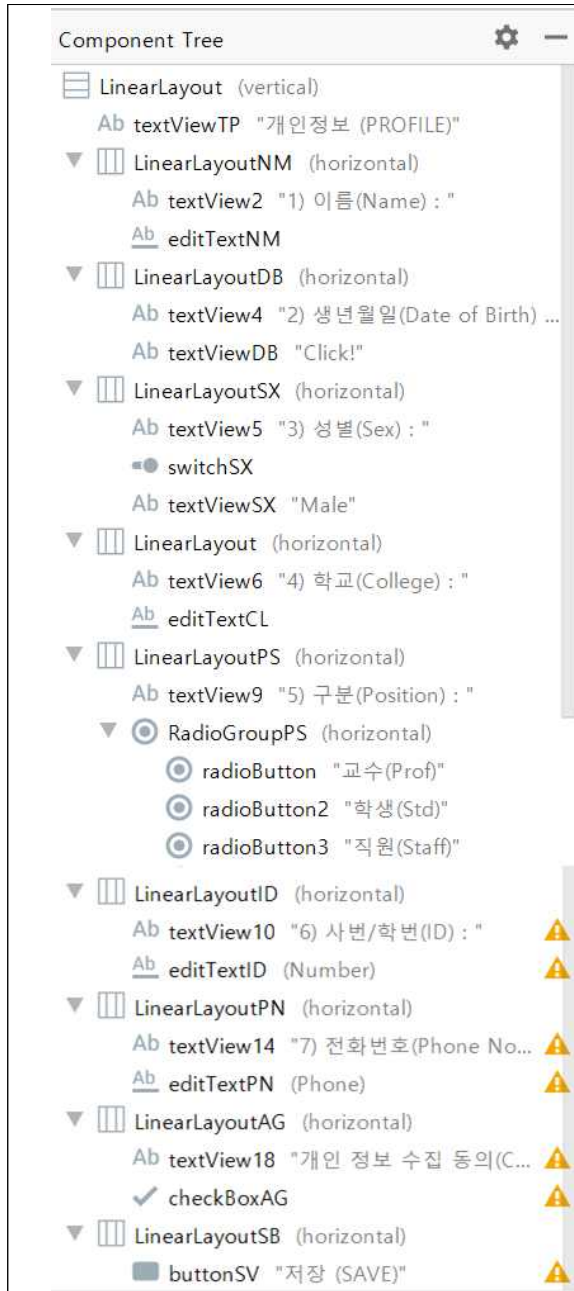
2. 본문

1) res > layout

① activity_main

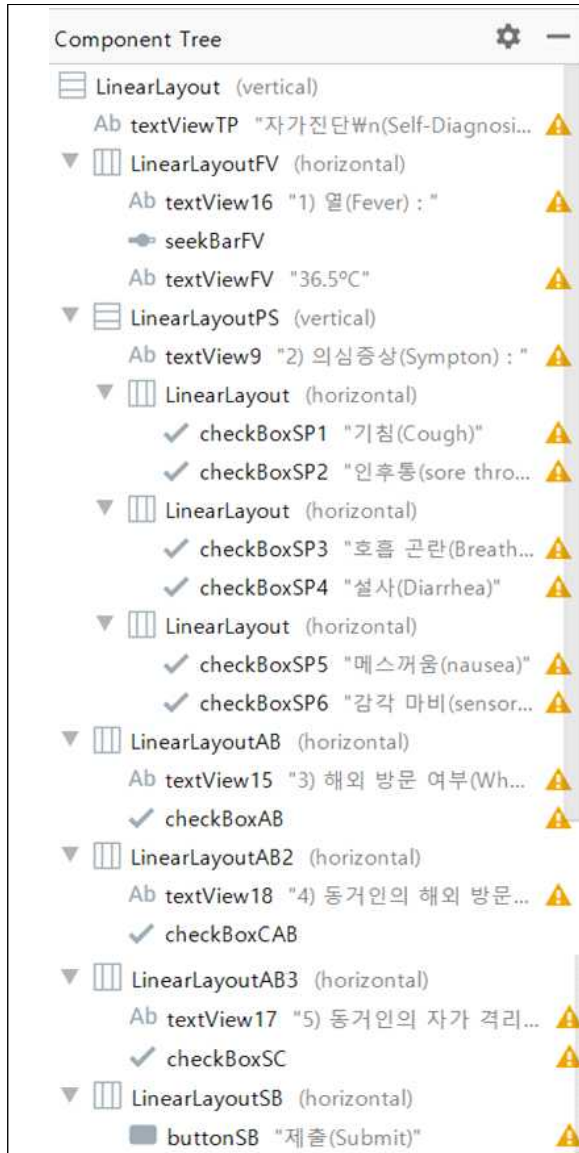
<p>Component Tree</p> <ul style="list-style-type: none"> LinearLayout (vertical) <ul style="list-style-type: none"> Ab textView3 "Mabe By 20181151 김지원" Ab textView "COVID-19" Ab textView2 "- 교내 출입 관리용 -" buttonPF "개인 정보 (Personal Data)" buttonSD "자가진단 (SELF DIAGNOSIS)" buttonCT "자가진단 확인증 (CERTIFIC..." buttonIF "정보 (INFO)" 	<div data-bbox="938 1323 1316 1585"> </div> <ul style="list-style-type: none"> - vertical Linear Layout에 학번, 어플리케이션 제목, 부제목을 출력하기 위한 Text View 3개와 Explicit Intent 기능을 넣을 Button 4개를 추가한다.
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

② activity_profile



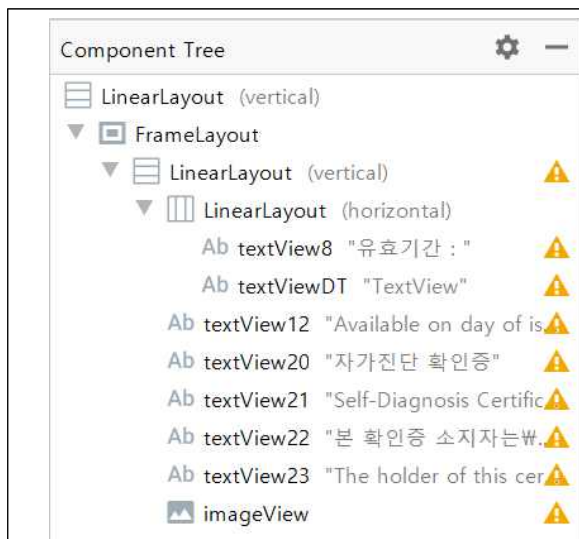
- vertical Linear Layout에 Text View로 상단에 해당 화면의 이름을 출력한다.
- horizontal Linear Layout 9개를 추가하여 개인정보 질문 항목을 제작한다.
- 이름(NM) : edit Text
- 생년월일(DB) : text View
- 성별(SX) : switch
- 학교(CL) : edit Text
- 구분(PS) : radio Button
- 사번/학번(ID) : edit Text
- 전화번호(PN) : edit Text
- 개인정보 수집 동의(AG) : check Box
- 위와 같은 각 항목의 맞게 View를 제작한다.
- 가장 하단에 저장하는 Button을 추가한다.

③ activity_diag




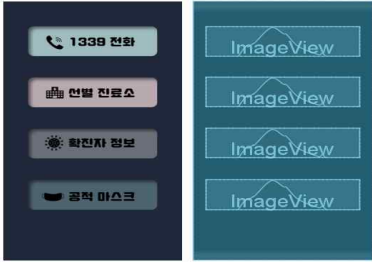
- vertical Linear Layout에 Text View로 상단에 해당 화면의 이름을 출력한다.
- horizontal Linear Layout 6개를 추가하여 개인정보 질문 항목을 제작한다.
- 열(FV) : seek Bar
- 의심증상(PS) : check Box (3줄로 나눠서 항목 출력)
- 해외 방문 여부(AB) : check Box
- 동거 가족의 해외 방문 여부(CAB) : check Box
- 동거 가족의 자가 격리 여부(SC) : check Box
- 위와 같은 각 항목의 맞게 View를 제작한다.
- 가장 하단에 저장하는 Button을 추가한다.

④ activity_pass



- vertical Linear Layout에 Frame Layout 하나를 추가 하여 디자인적 요소를 가미하였다.
- text View 7개를 넣어 자가 격리 확인증에 필요한 정보를 출력한다.
- 가장 하단에 바코드 Image View를 추가한다. (학교의 공공시설을 사용할 수 없기 때문에 이미지로 대체하였다.)

⑤ activity_info

	 <p>-vertical Linear Layout에 Button 4개를 추가하여 Implicit Intent 기능을 구현한다.</p>
-----------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------

2) 소스 코드 원본

① MainActivity.java

```
package com.example.week9_0515_assignment;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    Button btn_PF;
    Button btn_SD;
    Button btn_CT;
    Button btn_IF;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        btn_PF = findViewById(R.id.buttonPF);
        btn_PF.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(getApplicationContext(), activity_profile.class);
                // startActivityForResult(intent, 2);
                startActivity(intent);
            }
        });

        btn_SD = findViewById(R.id.buttonSD);
        btn_SD.setOnClickListener(new View.OnClickListener() {

            Intent intent = getIntent();
            int PF_check = intent.getIntExtra("profile_check", 2);

            @Override
            public void onClick(View v) {
                if (PF_check == 1)
                {
                    Intent intent = new Intent(getApplicationContext(), activity_diag.class);
                    // startActivityForResult(intent, 3);
                    startActivity(intent);
                }
                else if (PF_check != 1)
                {
                    Toast.makeText(getApplicationContext(), "개인정보를 먼저 입력하십시오.", Toast.LENGTH_LONG).show();
                    Intent intent = new Intent(getApplicationContext(), activity_profile.class);
                    startActivity(intent);
                }
            }
        });
    }
}
```

```

    }
}
});

btn_CT = findViewById(R.id.buttonCT);
btn_CT.setOnClickListener(new View.OnClickListener() {

    Intent intent = getIntent();
    int DG_check = intent.getIntExtra("diagnosis_check",2);

    @Override
    public void onClick(View v) {
        if(DG_check == 1)
        {
            Intent intent = new Intent(getApplicationContext(), activity_pass.class);
            startActivity(intent);
        }
        else if(DG_check != 1)
        {
            Toast.makeText(getApplicationContext(),"자가진단을 먼저 통과하십시오.",Toast.LENGTH_LONG).show();
            Intent intent = new Intent(getApplicationContext(), activity_diag.class);
            startActivity(intent);
        }

    }
});

btn_IF = findViewById(R.id.buttonIF);
btn_IF.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(getApplicationContext(), activity_info.class);
        startActivity(intent);
    }
});
}
}

```

② activity_profile.java

```

package com.example.week9_0515_assignment;

import androidx.appcompat.app.AppCompatActivity;

import android.app.DatePickerDialog;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.CompoundButton;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.RadioGroup;
import android.widget.Switch;
import android.widget.TextView;
import android.widget.Toast;

import java.util.Calendar;
import java.util.TimeZone;

public class activity_profile extends AppCompatActivity {

    EditText et_NM;
    TextView tv_DB;
    DatePickerDialog DP;
    Switch sw_SX;
    TextView tv_SX;
    EditText et_CL;
    RadioGroup rg_PS;
    EditText et_ID;
    EditText et_PN;
    CheckBox cb_AG;
    Button btn_SV;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_profile);

        et_NM = findViewById(R.id.editTextNM);
    }
}

```



```

tv_DB = findViewById(R.id.textViewDB);
final Calendar c = Calendar.getInstance();
c.setTimeZone(TimeZone.getTimeZone("Asia/Seoul"));
final int d = c.get(Calendar.DAY_OF_MONTH);
final int m = c.get(Calendar.MONTH);
final int y = c.get(Calendar.YEAR);
tv_DB.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Calendar c = Calendar.getInstance();
        c.setTimeZone(TimeZone.getTimeZone("Asia/Seoul"));
        int d = c.get(Calendar.DAY_OF_MONTH);
        int m = c.get(Calendar.MONTH);
        int y = c.get(Calendar.YEAR);
        DP = new DatePickerDialog(activity profile.this, new DatePickerDialog.OnDateSetListener() {
            @Override
            public void onDateSet(DatePicker view, int year, int month, int dayOfMonth) {
                tv_DB.setText(""+year+"/"+(month+1)+"/"+dayOfMonth);
            }
        }, y, m, d);
        DP.show();
    }
});

sw_SX = findViewById(R.id.switchSX);
tv_SX = findViewById(R.id.textViewSX);
sw_SX.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
        boolean SX = sw_SX.isChecked();
        if(SX == true)
        {
            tv_SX.setText(""+"Female");
        }
        else
        {
            tv_SX.setText(""+"Male");
        }
    }
});

et_CL = findViewById(R.id.editTextCL);
rg_PS = findViewById(R.id.RadioGroupPS);
et_ID = findViewById(R.id.editTextID);
et_PN = findViewById(R.id.editTextPN);
cb_AG = findViewById(R.id.checkBoxAG);

btn_SV = findViewById(R.id.buttonSV);
btn_SV.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(et_NM.getText().toString().matches("") || tv_DB.getText().toString().matches("Click!") ||
        et_CL.getText().toString().matches("") || et_ID.getText().toString().matches("") ||
        et_PN.getText().toString().matches("") || cb_AG.isChecked() == false)
        {
            Toast.makeText(getApplicationContext(), "정보를 모두 입력하십시오.", Toast.LENGTH_LONG).show();
        }
        else
        {
            Toast.makeText(getApplicationContext(), "정보가 저장되었습니다.", Toast.LENGTH_LONG).show();
            Intent intent = new Intent(getApplicationContext(), MainActivity.class);
            intent.putExtra("profile_check", 1);
            startActivityForResult(intent, 1);
        }
    }
});
}
}
}

```

③ activity_diag.java

```

package com.example.week9_0515_assignment;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

```

```

import android.widget.CheckBox;
import android.widget.SeekBar;
import android.widget.TextView;
import android.widget.Toast;

public class activity_diag extends AppCompatActivity {

    SeekBar sb_FV;
    TextView tv_FV;
    CheckBox cb_SP1;
    CheckBox cb_SP2;
    CheckBox cb_SP3;
    CheckBox cb_SP4;
    CheckBox cb_SP5;
    CheckBox cb_SP6;
    CheckBox cb_AB;
    CheckBox cb_CAB;
    CheckBox cb_SC;
    Button btn_SB;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_diag);

        sb_FV = findViewById(R.id.seekBarFV);
        tv_FV = findViewById(R.id.textViewFV);
        sb_FV.setMax(50);
        sb_FV.setProgress(15);
        sb_FV.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
                tv_FV.setText(">"+((progress)*0.1+35)+"%C");
            }
        });

        @Override
        public void onStartTrackingTouch(SeekBar seekBar) {

        }

        @Override
        public void onStopTrackingTouch(SeekBar seekBar) {

        }
    });

    cb_SP1 = findViewById(R.id.checkBoxSP1);
    cb_SP2 = findViewById(R.id.checkBoxSP2);
    cb_SP3 = findViewById(R.id.checkBoxSP3);
    cb_SP4 = findViewById(R.id.checkBoxSP4);
    cb_SP5 = findViewById(R.id.checkBoxSP5);
    cb_SP6 = findViewById(R.id.checkBoxSP6);
    cb_AB = findViewById(R.id.checkBoxAB);
    cb_CAB = findViewById(R.id.checkBoxCAB);
    cb_SC = findViewById(R.id.checkBoxSC);

    btn_SB = findViewById(R.id.buttonSB);
    btn_SB.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if(cb_SP1.isChecked() || cb_SP2.isChecked() || cb_SP3.isChecked() ||
            cb_SP4.isChecked() || cb_SP5.isChecked() || cb_SP6.isChecked() ||
            cb_AB.isChecked() || cb_CAB.isChecked() || cb_SC.isChecked() || sb_FV.getProgress() >= 25)
            {
                Toast.makeText(getApplicationContext(),"출입이 불가합니다.",Toast.LENGTH_LONG).show();
                Intent intent = new Intent(getApplicationContext(), activity_info.class);
                startActivityForResult(intent,5);
            }
            else
            {
                Toast.makeText(getApplicationContext(),"확인되었습니다.",Toast.LENGTH_LONG).show();

                Intent intent = new Intent(getApplicationContext(), activity_pass.class);
                intent.putExtra("diagnosis_check", 1);
                startActivityForResult(intent, 4);
            }
        }
    });
}

```

④ activity_pass.java

```
package com.example.week9_0515_assignment;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TextView;

import java.text.SimpleDateFormat;
import java.util.Calendar;

public class activity_pass extends AppCompatActivity {

    TextView tv_DT;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pass);

        tv_DT = findViewById(R.id.textViewDT);
        Calendar cal = Calendar.getInstance();
        SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");
        tv_DT.setText(format.format(Calendar.getInstance().getTime()));
    }
}
```

⑤ activity_info.java

```
package com.example.week9_0515_assignment;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageView;

public class activity_info extends AppCompatActivity {

    ImageView btn_CA;
    ImageView btn_CC;
    ImageView btn_CF;
    ImageView btn_MK;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_info);

        btn_CA = findViewById(R.id.imageViewCA);
        btn_CA.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                callPhone();
            }
        });

        btn_CC = findViewById(R.id.imageViewCC);
        btn_CC.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(Intent.ACTION_VIEW,
                    Uri.parse("https://www.mohw.go.kr/react/popup_200128_3.html"));
                startActivity(intent);
            }
        });

        btn_CF = findViewById(R.id.imageViewCF);
        btn_CF.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://coronamap.site/"));
                startActivity(intent);
            }
        });
    }
}
```

```

btn_MK = findViewById(R.id.imageViewMK);
btn_MK.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://00mask.com/#/map"));
        startActivity(intent);
    }
});

void callPhone() {
    Intent intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:1339"));

    if (ActivityCompat.checkSelfPermission(activity_info.this, Manifest.permission.CALL_PHONE) !=
        PackageManager.PERMISSION_GRANTED) {

        final String [] MYCALL = { Manifest.permission.CALL_PHONE};
        ActivityCompat.requestPermissions(this, MYCALL, 1004);
        // TODO: Consider calling
        // ActivityCompat#requestPermissions
        // here to request the missing permissions, and then overriding
        // public void onRequestPermissionsResult(int requestCode, String[] permissions,
        //                                     int[] grantResults)
        // to handle the case where the user grants the permission. See the documentation
        // for ActivityCompat#requestPermissions for more details.
        return;
    }
    startActivity(intent);
}
}

```

3) 소스 코드 분석

① MainActivity.java

```

package com.example.week9_0515_assignment;

import ...

public class MainActivity extends AppCompatActivity {

    Button btn_PF;
    Button btn_SD;
    Button btn_CT;
    Button btn_IF;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}

```

1. Import & 변수 선언

```

    btn_PF = findViewById(R.id.buttonPF);
    btn_PF.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent = new Intent(getApplicationContext(), activity_profile.class);
            startActivity(intent);
        }
    });
}

```

2. PF (개인정보 입력 화면)

- btn_PF인 Button을 클릭하면 activity_profile인 개인정보 입력창으로 전환되도록 Explicit Intent를 사용한다.
- 변수를 활성화 시킨 후, Onclick을 이용하여 Intent 객체를 생성하고, activity가 실행되도록 제작한다.

```

        btn_SD = findViewById(R.id.buttonSD);
        btn_SD.setOnClickListener(new View.OnClickListener() {

            Intent intent = getIntent();
            int PF_check = intent.getIntExtra( name: "profile_check", defaultValue: 2);

            @Override
            public void onClick(View v) {
                if(PF_check == 1)
                {
                    Intent intent = new Intent(getApplicationContext(), activity_diag.class);
                    startActivity(intent);
                }
                else if(PF_check != 1)
                {
                    Toast.makeText(getApplicationContext(), text: "개인정보를 먼저 입력하십시오.", Toast.LENGTH_LONG).show();
                    Intent intent = new Intent(getApplicationContext(), activity_profile.class);
                    startActivity(intent);
                }
            }
        });
    }
}

```

3. SD (자가 진단 실행 화면)

- 변수를 활성화 시킨 후, Onclick을 이용하여 Intent 객체를 생성하고, activity가 실행되도록 제작한다.
- activity_profile를 통해 전달 받은 "profile_check"를 getIntExtra로 가져와 PF_check 변수에 값을 저장한다.
- btn_SD인 Button을 클릭 시 PF_check가 1이면 activity_diag인 자가 진단 실행 창으로 전환되도록 Explicit Intent를 사용한다.
- btn_SD인 Button을 클릭 시 PF_check가 1이 아니면 Toast Message로 "개인정보를 먼저 입력하십시오."를 출력하고, activity_profile인 개인정보 입력창으로 전환되도록 Explicit Intent를 사용한다.

```

        btn_CT = findViewById(R.id.buttonCT);
        btn_CT.setOnClickListener(new View.OnClickListener() {

            Intent intent = getIntent();
            int DG_check = intent.getIntExtra( name: "diagnosis_check", defaultValue: 2);

            @Override
            public void onClick(View v) {
                if(DG_check == 1)
                {
                    Intent intent = new Intent(getApplicationContext(), activity_pass.class);
                    startActivity(intent);
                }
                else if(DG_check != 1)
                {
                    Toast.makeText(getApplicationContext(), text: "자가진단을 먼저 통과하십시오.", Toast.LENGTH_LONG).show();
                    Intent intent = new Intent(getApplicationContext(), activity_diag.class);
                    startActivity(intent);
                }
            }
        });
    }
}

```

4. CT (자가 진단 확인증 화면)

- 변수를 활성화 시킨 후, Onclick을 이용하여 Intent 객체를 생성하고, activity가 실행되도록 제작한다.
- activity_diag를 통해 전달 받은 "diagnosis_check"를 getIntExtra로 가져와 DG_check 변수에 값을 저장한다.
- btn_CT인 Button을 클릭 시 DG_check가 1이면 activity_pass인 자가 진단 확인증 출력 창으로 전환되도록 Explicit Intent를 사용한다.
- btn_CT인 Button을 클릭 시 DG_check가 1이 아니면 Toast Message로 "자가진단을 먼저 통과하십시오."를 출력하고, activity_diag인 자가 진단 실행 창으로 전환되도록 Explicit Intent를 사용한다.

<pre> btn_IF = findViewById(R.id.buttonIF); btn_IF.setOnClickListener(new View.OnClickListener() { @Override public void onClick(View v) { Intent intent = new Intent(getApplicationContext(), activity_info.class); startActivity(intent); } }); } } </pre>	
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

5. IF (코로나 관련 정보 제공 화면)

- btn_IF인 Button을 클릭하면 activity_info인 코로나 관련 정보 제공 창으로 전환되도록 Explicit Intent를 사용한다.
- 변수를 활성화 시킨 후, Onclick을 이용하여 Intent 객체를 생성하고, activity가 실행되도록 제작한다.

② activity_profile.java

<pre> package com.example.week9_0515_assignment; import ... public class activity_profile extends AppCompatActivity { EditText et_NM; TextView tv_DB; DatePickerDialog DP; Switch sw_SX; TextView tv_SX; EditText et_CL; RadioGroup rg_PS; EditText et_ID; EditText et_PN; CheckBox cb_AG; Button btn_SV; @Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity_profile); } } </pre>	
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

1. Import & 변수 선언

<pre> et_NM = findViewById(R.id.editTextNM); tv_DB = findViewById(R.id.textViewDB); final Calendar c = Calendar.getInstance(); c.setTimeZone(TimeZone.getTimeZone("Asia/Seoul")); final int d = c.get(Calendar.DAY_OF_MONTH); final int m = c.get(Calendar.MONTH); final int y = c.get(Calendar.YEAR); tv_DB.setOnClickListener(new View.OnClickListener() { @Override public void onClick(View v) { Calendar c = Calendar.getInstance(); c.setTimeZone(TimeZone.getTimeZone("Asia/Seoul")); int d = c.get(Calendar.DAY_OF_MONTH); int m = c.get(Calendar.MONTH); int y = c.get(Calendar.YEAR); DP = new DatePickerDialog(context: activity_profile.this, (view, year, month, dayOfMonth) -> { tv_DB.setText(""+year+"/"+(month+1)+"/"+dayOfMonth); }, y, m, d); DP.show(); } }); } } </pre>	
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

2. NM (이름 입력) & DB (생년월일 입력)

- edit Text인 et_NM 변수를 활성화 시켜 Text를 편집할 수 있게 한다.
- text View인 tv_DB 변수를 활성화 시킨 후, 해당 View를 클릭하면 날짜를 선택할 수 있게 Onclick와 DatePicker를 이용하여 제작한다.

<pre> sw_SX = findViewById(R.id.switchSX); tv_SX = findViewById(R.id.textViewSX); sw_SX.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() { @Override public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) { boolean SX = sw_SX.isChecked(); if(SX == true) { tv_SX.setText(""+Female"); } else { tv_SX.setText(""+Male"); } } }); </pre>	
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

3. SX (성별 입력)

- switch인 sw_SX와 textView인 tv_SX를 활성화 시킨다.
- sw_SX의 상태에 따라 tv_SX에 Female 혹은 Male이 출력된다.

<pre> et_CL = findViewById(R.id.editTextCL); rg_PS = findViewById(R.id.RadioGroupPS); et_ID = findViewById(R.id.editTextID); et_PN = findViewById(R.id.editTextPN); cb_AG = findViewById(R.id.checkBoxAG); </pre>	
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

4. CL (학교명 입력) & PS (구분 선택) & ID (학번/사번 입력) & PN (전화번호 입력) & AG (개인정보 수집 동의 확인)

- edit Text인 et_NM 변수를 활성화 시켜 Text를 편집할 수 있게 한다.
- text View인 tv_DB 변수를 활성화 시킨 후, 해당 View를 클릭하면 날짜를 선택할 수 있게 Onclick와 DatePicker를 이용하여 제작한다.

<pre> btn_sv = findViewById(R.id.buttonSV); btn_sv.setOnClickListener(new View.OnClickListener() { @Override public void onClick(View v) { if(et_NM.getText().toString().matches(regex: "") tv_DB.getText().toString().matches(regex: "Click!") et_CL.getText().toString().matches(regex: "") et_ID.getText().toString().matches(regex: "") et_PN.getText().toString().matches(regex: "") cb_AG.isChecked() == false) { Toast.makeText(getApplicationContext(), text: "정보를 모두 입력하십시오.", Toast.LENGTH_LONG).show(); } else { Toast.makeText(getApplicationContext(), text: "정보가 저장되었습니다.", Toast.LENGTH_LONG).show(); Intent intent = new Intent(getApplicationContext(), MainActivity.class); intent.putExtra(name: "profile_check", value: 1); startActivityForResult(intent, requestCode: 1); } } }); </pre>	
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

5. SV (저장)

- btn_sv인 Button을 활성화시킨다.
- 해당 Button을 클릭 했을 때 기입하지 않은 정보가 있다면 Toast Message로 “정보를 모두 입력하십시오.”라고 출력한다.
- 해당 Button을 클릭 했을 때 기입하지 않은 정보가 없다면 Toast Message로 “정보가 저장되었습니다.”라고 출력하고, Explicit Intent를 이용하여 Main Activity로 이동한다.
- Main Activity로 이동할 때 putExtra로 “profile_check”라는 변수에 1을 저장하여 전달한다.

③ activity_diag.java

```
package com.example.week9_0515_assignment;

import ...

public class activity_diag extends AppCompatActivity {

    SeekBar sb_FV;
    TextView tv_FV;
    CheckBox cb_SP1;
    CheckBox cb_SP2;
    CheckBox cb_SP3;
    CheckBox cb_SP4;
    CheckBox cb_SP5;
    CheckBox cb_SP6;
    CheckBox cb_AB;
    CheckBox cb_CAB;
    CheckBox cb_SC;
    Button btn_SB;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_diag);
    }
}
```

1. Import & 변수 선언

```
sb_FV = findViewById(R.id.seekBarFV);
tv_FV = findViewById(R.id.textViewFV);
sb_FV.setMax(50);
sb_FV.setProgress(15);
sb_FV.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
        tv_FV.setText(""+((progress)*0.1+35)+"°C");
    }

    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {
    }

    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {
    }
});
```

2. FV (체온 입력)

- seek Bar인 sb_FV 변수와 text View인 tv_FV를 활성화 시킨다.
- seek Bar의 최댓값 Max와 현재 값 Progress를 설정한다.
- 체온을 최저 35℃, 최대 40℃라고 가정하고 seek Bar를 50으로 나눈 후 한 칸 증가할 때 마다 0.1℃ 높아지게 설정한다.
- Listener를 사용하여 seek Bar가 변할 때마다 tv_FV의 값이 실시간으로 변화하게 한다.

```
cb_SP1 = findViewById(R.id.checkBoxSP1);
cb_SP2 = findViewById(R.id.checkBoxSP2);
cb_SP3 = findViewById(R.id.checkBoxSP3);
cb_SP4 = findViewById(R.id.checkBoxSP4);
cb_SP5 = findViewById(R.id.checkBoxSP5);
cb_SP6 = findViewById(R.id.checkBoxSP6);
cb_AB = findViewById(R.id.checkBoxAB);
cb_CAB = findViewById(R.id.checkBoxCAB);
cb_SC = findViewById(R.id.checkBoxSC);
```

2. FV (체온 입력)

- seek Bar인 sb_FV 변수와 text View인 tv_FV를 활성화 시킨다.
- seek Bar의 최댓값 Max와 현재 값 Progress를 설정한다.
- 체온을 최저 35℃, 최대 40℃라고 가정하고 seek Bar를 50으로 나눈 후 한 칸 증가할 때 마다 0.1℃ 높아지게 설정한다.
- Listener를 사용하여 seek Bar가 변할 때마다 tv_FV의 값이 실시간으로 변화하게 한다.


```

btn_SB = findViewById(R.id.buttonSB);
btn_SB.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(cb_SP1.isChecked() || cb_SP2.isChecked() || cb_SP3.isChecked() ||
           cb_SP4.isChecked() || cb_SP5.isChecked() || cb_SP6.isChecked() ||
           cb_AB.isChecked() || cb_CAB.isChecked() || cb_SC.isChecked() || sb_FV.getProgress() >= 25)
        {
            Toast.makeText(getApplicationContext(), text: "출입이 불가합니다.", Toast.LENGTH_LONG).show();
            Intent intent = new Intent(getApplicationContext(), activity_info.class);
            startActivityForResult(intent, requestCode: 5);
        }
        else
        {
            Toast.makeText(getApplicationContext(), text: "확인되었습니다.", Toast.LENGTH_LONG).show();

            Intent intent = new Intent(getApplicationContext(), activity_pass.class);
            intent.putExtra( name: "diagnosis_check", value: 1);
            startActivityForResult(intent, requestCode: 4);
        }
    }
});
}
}

```

5. SB (제출)

- btn_SB Button을 활성화시킨다.
- 해당 Button을 클릭 했을 때 체크된 항목이 있거나 체온이 37.5℃를 초과할 경우 Toast Message로 “출입이 불가합니다.”라고 출력한 후 Explicit Intent를 이용하여 activity_info로 이동한다.
- 해당 Button을 클릭 했을 때 체크된 항목이 없거나 체온이 37.5℃ 이하일 경우 Toast Message로 “확인되었습니다.”라고 출력하고, Explicit Intent를 이용하여 activity_pass로 이동한다.
- activity_pass로 이동할 때 putExtra로 “diagnosis_check”라는 변수에 1을 저장하여 전달한다.

④ activity_pass.java

```

package com.example.week9_0515_assignment;

import ...

public class activity_pass extends AppCompatActivity {

    TextView tv_DT;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pass);

        tv_DT = findViewById(R.id.textViewDT);
        Calendar cal = Calendar.getInstance();
        SimpleDateFormat format = new SimpleDateFormat( pattern: "yyyy-MM-dd" );
        tv_DT.setText(format.format(Calendar.getInstance().getTime()));
    }
}

```

1. Import & 변수 선언 & DT (유효기간 출력)

- text View인 tv_DT 변수를 활성화 시킨다.
- tv_DT에 어플리케이션을 실행하는 날짜가 출력되도록 한다.

⑤ activity_info.java

```
package com.example.week9_0515_assignment;

import ...

public class activity_info extends AppCompatActivity {

    ImageView btn_CA;
    ImageView btn_CC;
    ImageView btn_CF;
    ImageView btn_MK;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_info);
    }
}
```

1. Import & 변수 선언

```
btn_CA = findViewById(R.id.imageViewCA);
btn_CA.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        callPhone();
    }
});
```

2. CA (1339 전화)

- btn_CA인 Button을 클릭하면 전화하는 창으로 전환되도록 Implicit Intent를 사용한다.
- onCreate()외부에 있는 callPhone()을 불러온다.

```
btn_CC = findViewById(R.id.imageViewCC);
btn_CC.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.mohw.go.kr/react/popup_200128_3.html"));
        startActivity(intent);
    }
});
```

3. CC (선별 진료소 웹페이지)

- btn_CC인 Button을 클릭하면 선별 진료소 웹페이지로 전환되도록 Implicit Intent를 사용한다.

```
btn_CF = findViewById(R.id.imageViewCF);
btn_CF.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://coronamap.site/"));
        startActivity(intent);
    }
});
```

3. CF (확진자 정보 웹페이지)

- btn_CF인 Button을 클릭하면 확진자 정보 웹페이지로 전환되도록 Implicit Intent를 사용한다.

```
btn_MK = findViewById(R.id.imageViewMK);
btn_MK.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://00mask.com/#/map"));
        startActivity(intent);
    }
});
```

4. MK (공적 마스크 정보 웹페이지)

- btn_MK인 Button을 클릭하면 공적 마스크 정보 웹페이지로 전환되도록 Implicit Intent를 사용한다.

```

void callPhone() {
    Intent intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:1339"));

    if (ActivityCompat.checkSelfPermission( context: activity_info.this, Manifest.permission.CALL_PHONE) != PackageManager.PERMISSION_GRANTED) {


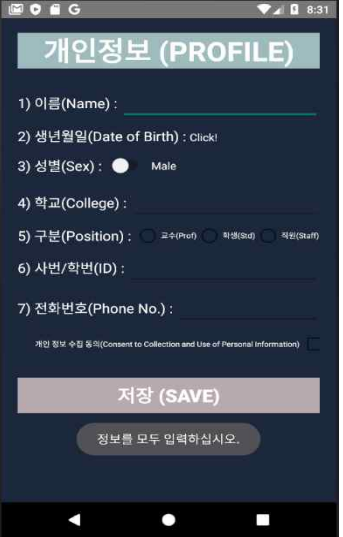
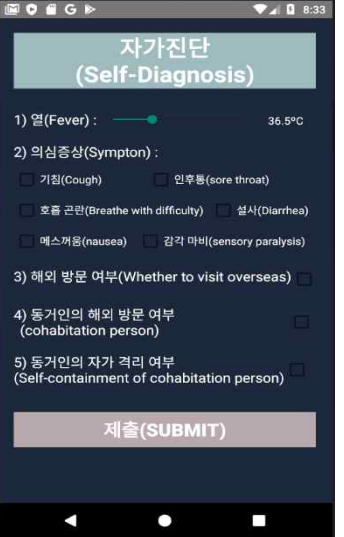


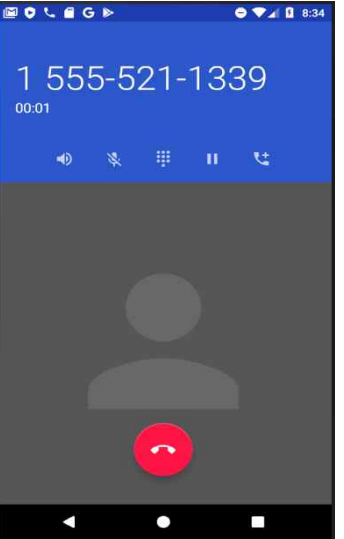
        final String [] MYCALL = { Manifest.permission.CALL_PHONE};
        ActivityCompat.requestPermissions( activity: this, MYCALL, requestCode: 1004);
        // TODO: Consider calling
        // ActivityCompat#requestPermissions
        // here to request the missing permissions, and then overriding
        // public void onRequestPermissionsResult(int requestCode, String[] permissions,
        // int[] grantResults)
        // to handle the case where the user grants the permission. See the documentation
        // for ActivityCompat#requestPermissions for more details.
        return;
    }
    startActivity(intent);
}
}

```

5. callPhone()

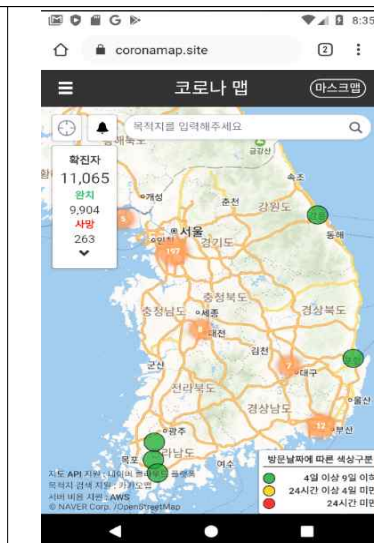
- 전화하는 창으로 전환되도록 Implicit Intent를 사용한다.
- 이 어플리케이션에서 전화 연결이 처음일 경우 전화 기능을 사용할지 여부를 물어보는 메시지를 띄운다.

4) 결과 화면

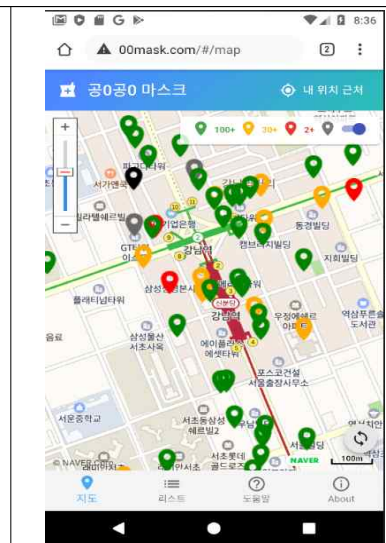
		
앱 실행 시 초기 화면	“개인 정보” Button 클릭 시 화면	“자가진단” Button 클릭 시 화면
		
자가진단 통과 시 출력 화면	자가진단을 통과하지 못하거나 초기 화면에서 “Info” Button 클릭 시 화면	“1339 전화” Button 클릭 시 화면



“선별 진료소” Button 클릭 시 화면



“확진자 정보” Button 클릭 시 화면



“공적 마스크” Button 클릭 시 화면

3. 결론

1) COVID-19 정보 앱 장점

① 필수 사항인 자가진단 기능과 코로나 정보 제공기능 외에 출입관리 기능을 추가하였다.

: 자가 진단을 시행 한 결과를 활용할 수 있는 아이디어를 생각하다가 최근 교내 출입 관리를 하면서 발열검사 확인증을 항상 지참해야하는 불편함을 느끼고 있었다. 이를 어플리케이션을 이용하여 종이를 들고 다니는 불편함을 해소하고 출입 관리 시 체온 검사만 시행했었지만 자가 진단 문진표를 작성하면서 더 정확하고 세세하게 출입 관리가 가능해질 것으로 기대된다.

② 개인정보가 입력되지 않았을 때 자가진단 실행을 시도하거나 자가진단을 하지 않았을 때

자가진단 확인증을 시도하지 못하도록 설정하였다.

: 자가진단을 시행하고 제출 했을 때 제출한 사람의 정보가 없으면 그 조사 자료는 무의미한 정보가 된다. 따라서 개인정보를 저장하지 않은 사람이 자가 진단 Button을 클릭할 때 개인정보를 먼저 저장하도록 화면을 개인정보 입력창으로 전환한다. 개인정보를 저장하였는지 확인하기 위해 activity_profile에서 저장될 경우 “profile_check” 변수에 1 값을 저장하여 보낸다. MainActivity에서 이 변수를 가져와 확인하여 개인정보가 저장되었는지 아니지를 판단한다.

또한, 자가진단을 시행하지 않고 자가진단 확인증을 발급받을 수 없다. 따라서 자가진단을 제출하지 않고서 자가진단 확인증 Button을 클릭하면 자가진단을 먼저 진행하도록 화면을 자가진단 테스트 창으로 전환한다. 자가진단을 했는지 확인하기 위해 activity_dig 에서 테스트가 통과하면 “diagnosis_check” 변수에 1 값을 저장하여 보낸다. MainActivity에서 이 변수를 가져와 확인하여 자가진단 문진표를 제출하였는지 아닌지를 판단한다.

2) COVID-19 정보 앱 단점

① 개인정보 화면에서 입력한 정보가 다시 접속할 경우 출력되지 않는다.

: 개인정보 입력 화면에서 이름부터 개인정보 수집 동의까지 입력하고 저장한 후 다시 Main 화면으로 돌아갔다가 다시 개인정보 입력 화면에 접속하면 기입했던 정보들이 비어서 출력된다. 개인정보가 저장되어 있다는 의미인 “profile_check”의 값이 1임은 변함이 없지만 유저 입장에서 자신의 정보가 저장되지 않을 것으로 착각하여 불편함을 줄 수 있다.

② 자가진단 확인증의 바코드가 유효하지 않다.

: 현재 교내 건물에 출입할 때 학생증에 부착된 바코드를 인식시켜 출입문의 연 후 입장이 가능하다. 해당 아이디어를 이용하면 좋을 것 같아 어플리케이션에 바코드를 넣어 제작하였지만 이 바코드는 이미지일 뿐 유효한 바코드가 아니다. 학교의 보안 시스템을 함부로 사용할 수 없기 때문에 임시로 바코드를 넣었지만 만약 실제로 해당 어플이 사용한다면 유효한 바코드를 출력시켜 사용하는 것이 매우 편리할 것으로 예상된다.

③ Implicit Intent의 call 기능이 완전하지 못하다.

: implicit Intent의 call을 이용하기 위해 전화번호를 입력해야한다. “tel:”의 기본형은 “010-0000-0000”이다. 하지만 “1339”전화는 국번 없이 4자리의 전화번호이기 때문에 해당 기본형과는 차이가 있다. 우선 기본형을 무시한 채로 “1339”를 기입하여 실행시켰더니 마지막 4자리만 정확히 나오고 앞에 자리 숫자들은 원하지 않은 숫자들이 출력되어 해당 전화번호로 전화가 걸렸다. 조사해 본 결과 119나 112와 같이 긴급전화를 이용하는 것은 통신사의 권한이기 때문에 사용할 수 없다고 안내되어 있었다.

3) 배운점 & 느낀점

지금까지 어플리케이션을 제작하면서 이번 주에 학습한 intent기능을 수행하고 싶었다. 막상 학습하고 새로운 어플을 제작하려고 시도해 보니 다양한 아이디어가 떠올라서 여러 개의 activity를 제작하게 되었다. 각각의 activity를 제작하는 것은 지금까지 제작한 어플리케이션과 동일한 방식으로 진행했기 때문에 제작 시간을 오래 걸렸지만 어려움 없이 만들 수 있었다. 하지만 여러 activity에서 사용되는 변수를 이용할 때는 어려움을 느꼈다. 물론 실습시간에 이에 해당하는 내용을 학습하였지만 완전히 이해하지 못한 상태로 이용하려고 하다보니 계속 에러가 발생하였다. 그래서 강의 영상을 반복해서 듣고, 차근차근 응용하여 접목시켰다. 결과적으로 소스코드도 완성할 수 있었고 과제를 하면서 실습 수업 내용도 이해할 수 있었다.

어플리케이션이 점점 복잡해지면서 어려운 점이 변수의 종류가 많아진다는 점이 에러를 발생하는 요인 중 하나였다. 여러 개의 변수가 사용되고 비슷하게 사용되어 에러가 발생하고 이를 쳐다보고 있어도 잘못된 점을 못찾아 시간을 많이 허비하였다. 하지만 이렇게 실수를 많이 하고 고민을 많이 한 결과 다음에 변수를 사용할 때 더 조심스러워 지고 에러 내용을 확인하고 변수 명도 꼼꼼히 확인할 수 있게 되었다.