

---

# FILLIT

지구를 위해 버리고, 지구를 위해 채우다

---

IT공학전공

1814843

김유진

# 목차

---

01 프로젝트 설명

02 프로젝트 관계도

03 프로젝트 기능

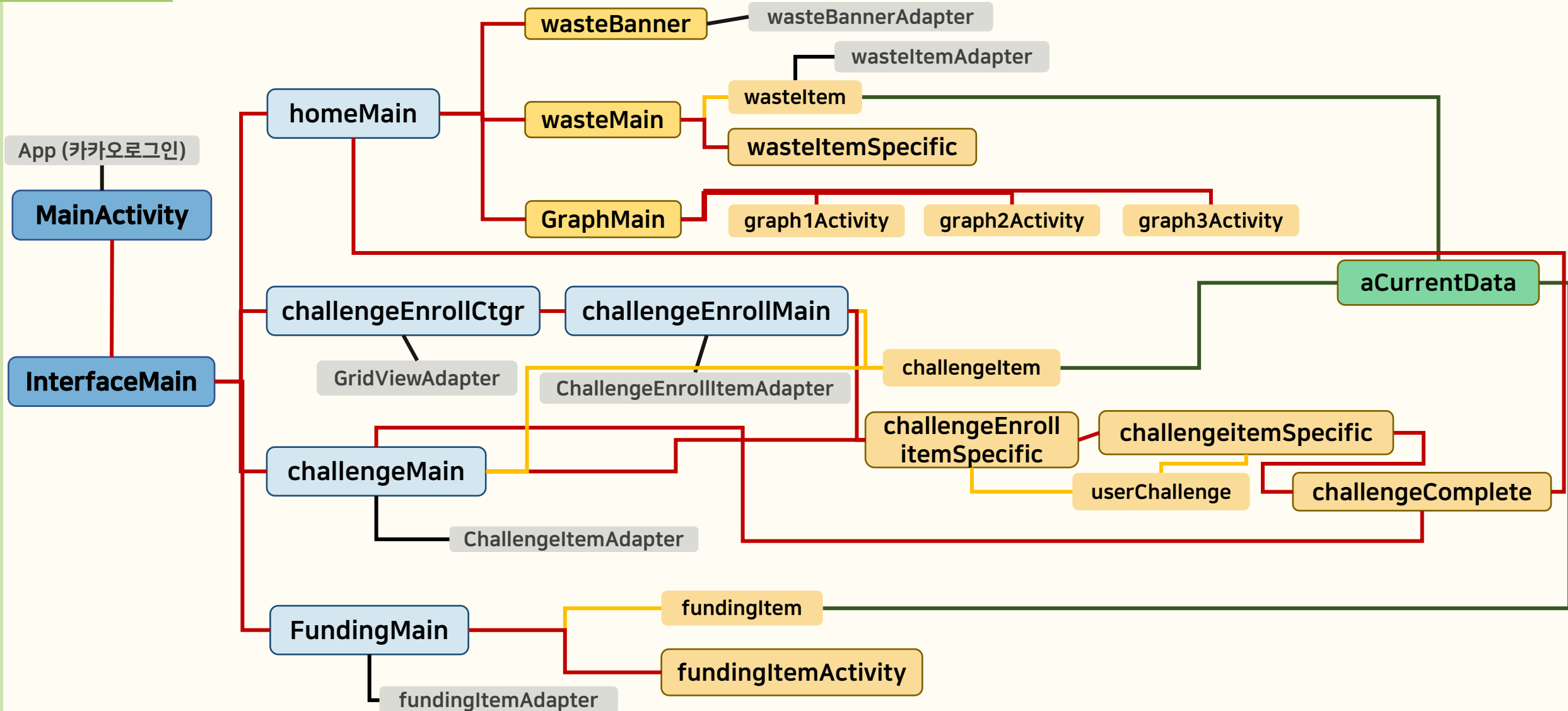
04 시연

05 Q&A

---

FILLIT : 지구를 위해 버리고, 지구를 위해 채우다

- 분리 배출 쓰레기에 대한 정보를 제공하고 이를 일상 속에서도 실천하도록 돕는 환경 어플리케이션
- 간단한 조작법을 통해 올바른 분리배출에 대한 정보를 제공
- 사용자가 환경을 위해 올바른 분리배출을 하고, 환경을 보호를 실천할 수 있도록 사용자에게 동기부여를 제공
- 환경을 위한 챌린지를 등록
- 챌린지 기간을 선택하여 챌린지 등록
- 하루에 1개 사진을 업로드하며 챌린지 수행
- 챌린지 완료시 포인트를 지급
- 그 포인트를 환경 관련 기업에 후원할 수 있도록 하여, 사용자가 환경을 보호하고 있다는 것을 시각적으로 느끼게 함



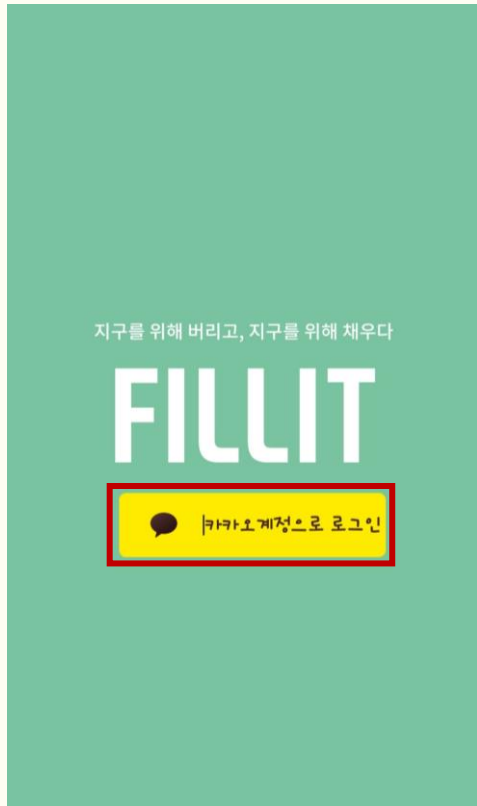
## 03

## 'FILLIT' - 프로젝트 기능

MainActivity.java

App.java

- 카카오 계정 로그인



카카오계정 로그인

---

heather0220@naver.com

---

.....

---

로그인

[로그인 정보를 잊으셨나요?](#)

[새로운 카카오계정 만들기](#)

```
public class App extends Application {
    private static volatile App instance = null;

    private static class KakaoSDKAdapter extends KakaoAdapter {
        public ISessionConfig getSessionConfig() {

            return new ISessionConfig() {
                @Override
                public AuthType[] getAuthTypes() { return new AuthType[] {AuthType.KAKAO_TALK}; }

                @Override
                public boolean isUsingWebviewTimer() { return false; }

                @Override
                public boolean isSecureMode() { return false; }

                @Override
                public ApprovalType getApprovalType() { return ApprovalType.INDIVIDUAL; }

                @Override
                public boolean isSaveFormData() { return true; }
            };
        }
    }

    @Override
    public IApplicationConfig getApplicationConfig() {
        return new IApplicationConfig() {
            @Override
            public Context getApplicationContext() {
                return App.getGlobalApplicationContext();
            }
        };
    }
}
```

# 03

## 'FILLIT' - 프로젝트 기능

```
class homeMain extends Fragment {
```



```
class challengeEnrollMain extends Fragment implements OnItemClickListener{
```



```
class challengeMain extends Fragment implements OnItemClickListener{
```



```
class fundingMain extends Fragment implements OnItemClickListener{
```



### Interface.java

- 홈, 등록, 챌린지, 후원

-> Fragment 이용

```
public void callMenu(int id) {
    FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
    switch (id) {
        case R.id.menuHome:
            clearMenu();
            imageHome.setImageResource(R.drawable.home_on);
            textHome.setTextColor(Color.parseColor( colorString: "#6A6A6A"));
            fragmentTransaction.replace(R.id.frameMain, new homeMain());
            break;
        case R.id.menuEnroll:
            clearMenu();
            imageEnroll.setImageResource(R.drawable.enroll_on);
            textEnroll.setTextColor(Color.parseColor( colorString: "#6A6A6A"));
            fragmentTransaction.replace(R.id.frameMain, new challengeEnrollCtgr());
            break;
        case R.id.menuChallenge:
            clearMenu();
            imageChallenge.setImageResource(R.drawable.map_on);
            textChallenge.setTextColor(Color.parseColor( colorString: "#6A6A6A"));
            fragmentTransaction.replace(R.id.frameMain, new challengeMain());
            break;
        case R.id.menuFunding:
            clearMenu();
            imageFunding.setImageResource(R.drawable.funding_on);
            textFunding.setTextColor(Color.parseColor( colorString: "#6A6A6A"));
            fragmentTransaction.replace(R.id.frameMain, new fundingMain());
            break;
    }
    fragmentTransaction.commit();
}
```

## 03

## 'FILLIT' - 프로젝트 기능

```

public class challengeItem {
    String title;
    String des;
    String regdate;
    String deadline;
    Integer chalPoint = 100;
    //Set<Long> days1 = new TreeSet<>();
    Integer days = 0;
    static int progress = 0;

    public void clone(challengeItem original) {
        title = original.title;
        des = original.des;
        regdate = original.regdate;
        deadline = original.deadline;
        days = original.days;
        chalPoint = original.chalPoint;
    }
}

```

```

public class fundingItem {
    String fund_inst;
    String inst_des;
    int img;
    int seek_max;
    int tar_point; // 펀딩 목표 포인트
    int acu_point; // 펀딩 전체에서 누적 포인트
    int left_point; // 펀딩 전체에서 달성까지 남은 포인트
    int fund_point;
    int mypoint;
    int rest_point;
}

```

## aCurrentData.java

- wasteItem
- wasteItemBanner
- challengeItem
- fundingItem

```

public class aCurrentData {
    public static ArrayList<wasteItem> listWaste = new ArrayList<>();
    public static ArrayList<wasteItemBanner> listWasteBanner = new ArrayList<>();
    public static ArrayList<challengeItem> listChallengeEnroll = new ArrayList<>();
    public static ArrayList<challengeItem> listMyChallenge = new ArrayList<>();
    public static ArrayList<fundingItem> listFunding = new ArrayList<>();
}

```

# 03

## 'FILLIT' - 프로젝트 기능

### FILLIT

분리배출의 핵심 4가지



#### 1. 비운다.

용기 안에 담겨 있는 내용물은 깨끗이 비우고 배출합니다.



마스크



플라스틱



캔/유리



종이



비닐



스트로폼

환경 그래프

Nice job!

Test Ad  
This is a 468x60 test ad.



홈



등록



챌린지



후원

분리배출의 핵심 4가지



#### 2. 헹군다.

재활용품에 묻어있는 이물질, 음식물 등은 닦거나 한 번 헹궈서 배출한다.

분리배출의 핵심 4가지



#### 3. 분리한다.

라벨 등의 다른 재질 부분은 제거하여 배출합니다.

분리배출의 핵심 4가지



#### 4. 섞지 않는다.

종류별, 재질별로 구분하여 분리수거함으로 배출합니다.

### HomeMain.java

### wasteBannerAdapter

#### - 분리배출의 핵심 4가지

#### -> RecyclerView 이용

```
viewList2 = view.findViewById(R.id.recyclerView2);
viewList2.setLayoutManager(new LinearLayoutManager(this.getContext(), LinearLayoutManager.HORIZONTAL, false));
// 부드럽게 넘기기
snapHelper.attachToRecyclerView(viewList2);
adapter2 = new wasteItemBannerAdapter(aCurrentData.listWasteBanner);
viewList2.setAdapter(adapter2);
```

```
public class wasteItemBannerAdapter extends RecyclerView.Adapter<wasteItemBannerAdapter.ViewHolder>{
    // 아이템 뷰를 저장하는 뷰홀더 클래스.
    public class ViewHolder extends RecyclerView.ViewHolder {
        ImageView imageBanner;
        ViewHolder(View itemView) {
            super(itemView);
            // 뷰 객체에 대한 참조. (hold strong reference)
            imageBanner = itemView.findViewById(R.id.imageBanner);
            itemView.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    int position = getAdapterPosition();
                }
            });
        }
    }
    // 생성자에서 데이터 리스트 객체를 전달받음.
    wasteItemBannerAdapter(ArrayList<wasteItemBanner> item) {
        aCurrentData.listWasteBanner = item;
    }
    // onCreateViewHolder() - 아이템 뷰를 위한 뷰홀더 객체 생성하여 리턴.
    @Override
    public wasteItemBannerAdapter.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        Context context = parent.getContext();
        LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        View view = inflater.inflate(R.layout.waste_banner_item, parent, attachToRoot: false);
```



# 03

## 'FILLIT' - 프로젝트 기능 FILLIT

분리배출의 핵심 4가지



### 1. 비운다.

용기 안에 담겨 있는 내용물은  
깨끗이 비우고 배출합니다.



마스크



플라스틱



캔/유리



종이



비닐



스트로폼

환경 그래프

Nice job!

Test Ad  
This is a 468x60 test ad.



HomeMain.java

WasteMain

wasteItemSpecific

- 카테고리별로 분리배출 정보 제공

- 광고

-> adView 이용

```
public class wasteMain extends Fragment implements OnItemClickListener {
    RecyclerView listView;
    wasteItemAdapter adapter;
    ArrayList<wasteItem> items;
    String ctgr;

    public void setItems(ArrayList<wasteItem> newOne) {
        public void setCtgr(String ctgr) { this.ctgr = ctgr; }

    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.waste_main, container, false);
        TextView textCtgr = view.findViewById(R.id.textCtgr);
        if (items.size() == 0) {
            textCtgr.setText("정보가 없습니다.");
        } else {
            textCtgr.setText(ctgr);

            listView = view.findViewById(R.id.recyclerView);
            listView.setLayoutManager(new LinearLayoutManager(this.getContext(), LinearLayoutManager.HORIZONTAL, false));

            // 부드럽게 넘기기
            PagerSnapHelper snapHelper = new PagerSnapHelper();
            snapHelper.attachToRecyclerView(listView);

            adapter = new wasteItemAdapter(items, this);
            listView.setAdapter(adapter);
        }
        return view;
    }

    @Override
    public void onClick(wasteItem newOne) {
        MobileAds.initialize(this.getContext(), new OnInitializationCompleteListener() {
            @Override
            public void onInitializationComplete(InitializationStatus initializationStatus) {
                adView = view.findViewById(R.id.adView);
                AdRequest adRequest = new AdRequest.Builder().build();
                adView.loadAd(adRequest);
            }
        });
    }
}
```

# 03

## 'FILLIT' - 프로젝트 기능

HomeMain.java

GraphMain

- 환경 그래프

-> 1) 원 그래프 2) 꺾은선 그래프 3) 막대 그래프

graph1Activity

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_graph1);

    pieChart = (PieChart) findViewById(R.id.piechart);

    entries = new ArrayList<>();

    PieEntryLabels = new ArrayList<String>();
    AddValuesToPIEENTRY();

    AddValuesToPieEntryLabels();

    pieDataSet = new PieDataSet(entries, label: "'재질별'");

    pieData = new PieData(PieEntryLabels, pieDataSet);

    pieDataSet.setColors(ColorTemplate.COLORFUL_COLORS);

    pieChart.setData(pieData);
    pieData.setValueTextSize(18f);
    pieChart.animateY( durationMillis: 3000);

}

public void AddValuesToPIEENTRY(){
```

graph2Activity

```
LineChart lineChart = (LineChart) findViewById(R.id.chart);

LineDataSet dataset = new LineDataSet(entries, label: "생활폐기물");

ArrayList<String> labels = new ArrayList<>();

LineData data = new LineData(labels, dataset);
dataset.setColors(ColorTemplate.COLORFUL_COLORS);
lineChart.setData(data);
lineChart.animateY( durationMillis: 3000);
```

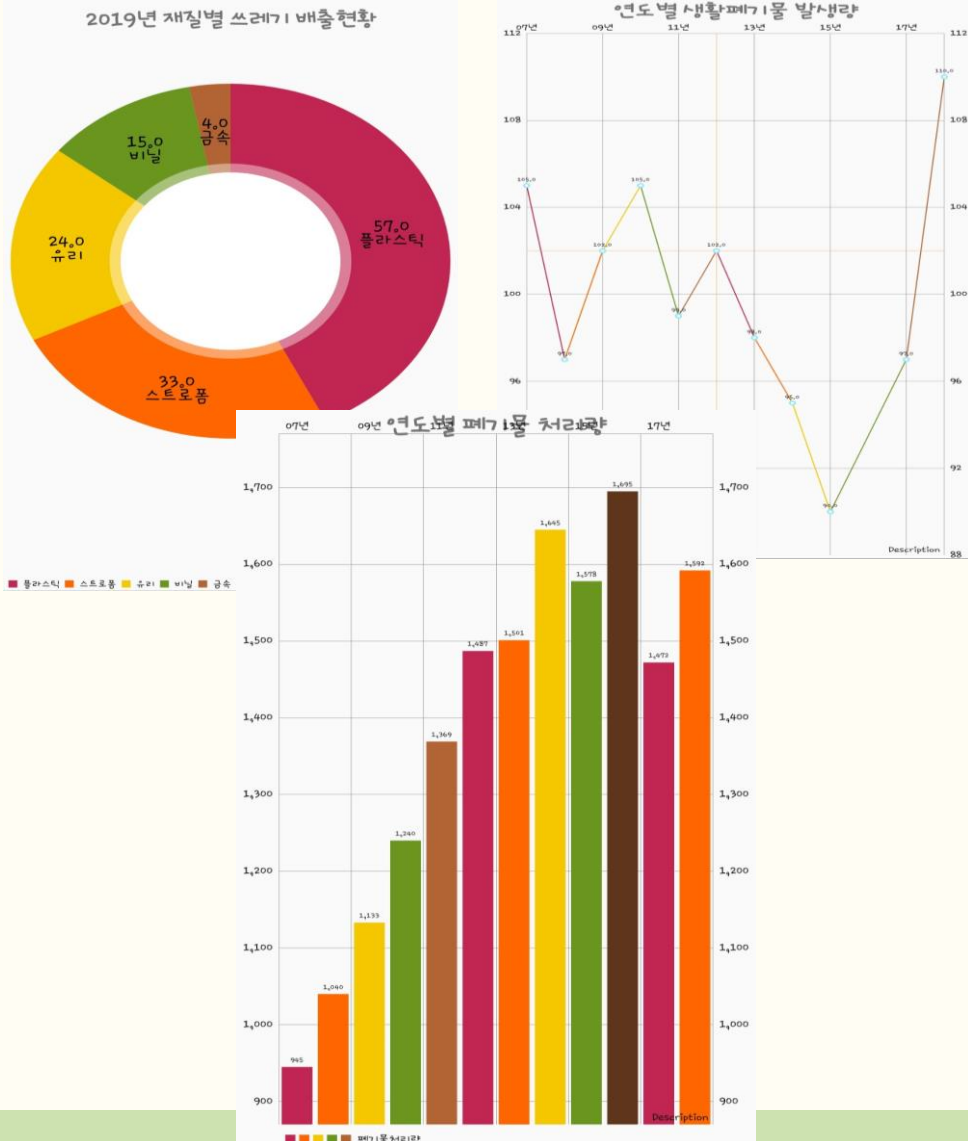
graph3Activity

```
BarChart chart = findViewById(R.id.barchart);

ArrayList entries = new ArrayList();

BarDataSet bardataset = new BarDataSet(entries, label: "폐기물처리량");

BarData data = new BarData(day, bardataset);
bardataset.setColors(ColorTemplate.COLORFUL_COLORS);
chart.setData(data);
chart.animateY( durationMillis: 3000);
```



## 03

## 'FILLIT' - 프로젝트 기능

FILLIT



마스크



플라스틱



캔 / 유리



종이



비닐



기타

Nice job!

Test Ad

This is a 468x60 test ad.



홈



등록



챌린지



후원

challengeErollCtgr.java

GridViewAdapter

- 카테고리별 챌린지 등록

-&gt; GridView 이용

```

public class challengeEnrollCtgr extends Fragment {
    private GridViewAdapter mAdapter;
    private ArrayList<String> listCategory;
    private ArrayList<Integer> listPicture;
    private GridView gridView;
    private AdView adView;
    static int num = 0;

    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.challenge_enroll_ctgr, container, attachToRoot: false);

        MobileAds.initialize(this.getContext(), new OnInitializationCompleteListener() {
            @Override
            public void onInitializationComplete(InitializationStatus initializationStatus) {}
        });

        adView = view.findViewById(R.id.adView);
        AdRequest adRequest = new AdRequest.Builder().build();
        adView.loadAd(adRequest);

        prepareList();
        // prepared arraylist and passed it to the Adapter class
        mAdapter = new GridViewAdapter(getActivity(), listCategory, listPicture);
        // Set custom adapter to gridView
        gridView = (GridView) view.findViewById(R.id.gridView1);
        gridView.setAdapter(mAdapter);

        // Implement On Item click listener
        gridView.setOnItemClickListener((arg0, arg1, position, arg3) -> {
            Toast.makeText(getActivity(), mAdapter.getItem(position), Toast.LENGTH_SHORT).show();
        });

        if(convertView==null)
        {
            view = new ViewHolder();
            convertView = inflater.inflate(R.layout.challenge_cgtr_row, root: null);

            view.txtViewTitle = (TextView) convertView.findViewById(R.id.textView1);
            view.imgViewFlag = (ImageView) convertView.findViewById(R.id.imageView1);

            convertView.setTag(view);
        }
        else
        {
            view = (ViewHolder) convertView.getTag();

            view.txtViewTitle.setText(listCountry.get(position));
            view.imgViewFlag.setImageResource(listFlag.get(position));
        }
    }
}

```

# 03

## 'FILLIT' - 프로젝트 기능

원하는 챌린지를 등록해보세요!

①

마스크 분리배출

마스크 일반쓰레기통에 잘 버리기

③

유리/캔 분리배출

내용물을 비우고 물로 헹구어 이물질 제거 후 배출하기

유리/캔 분리배출

상표 제거 가능한 경우 상표를 제거한 후 배출

유리/캔 분리배출

깨지지 않은 유리를 버리기

추가

캔 / 유리

⑥

대중교통 이용하기

자가용 이용하지 않기

비누 사용 줄이기

천연 수세미로 걸거지 하기

②

플라스틱 분리배출

내용물을 비우고 물로 헹구어 이물질 제거 후 배출하기

일회용 컵 사용하지 않기

카페 이용시 텀블러 사용하기

일회용 빨대 사용하지 않기

재사용 가능한 빨대 사용하기

플라스틱 사용하지 않기

배달 음식 이용 시 일회용품 받지 않기 추가

④

종이 분리배출

비닐 코팅된 표지, 공책의 스프링 등을 제거한 후 배출하기

⑤

비닐 분리배출

내용물을 비우고 물로 헹구어 이물질 제거 후 배출하기

### challengeEnrollMain.java

### ChallengeEnrollItemAdapter

- 카테고리별 챌린지 목록

-> RecyclerView 이용

- 사용자가 입력하여 챌린지 추가

-> AlertDialog 이용

```
public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.challenge_enroll_main, container, attachToRoot: false);
    viewList = view.findViewById(R.id.recyclerView);
    viewList.setLayoutManager(new LinearLayoutManager(this.getContext()));
    InitializeChallengeData();

    // 부드럽게 넘기기
    PagerSnapHelper snapHelper = new PagerSnapHelper();
    snapHelper.attachToRecyclerView(viewList);

    adapter = new challengeEnrollItemAdapter(aCurrentData.ListChallengeEnroll, listener, this);
    viewList.setAdapter(adapter);
}
```

```
public class challengeEnrollItemAdapter extends RecyclerView.Adapter<challengeEnrollItemAdapter.ViewHolder> {

    private ArrayList<challengeItem> mData = null ;
    private OnItemClickForChallenge mCallback;

    // 아이템 뷰를 저장하는 뷰홀더 클래스.
    public class ViewHolder extends RecyclerView.ViewHolder {
        TextView textTitle;
        TextView textDescription;

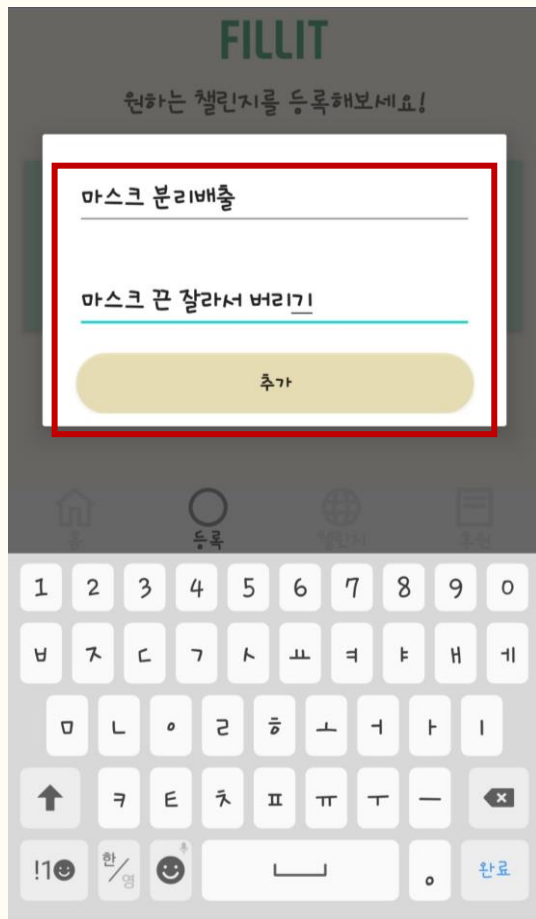
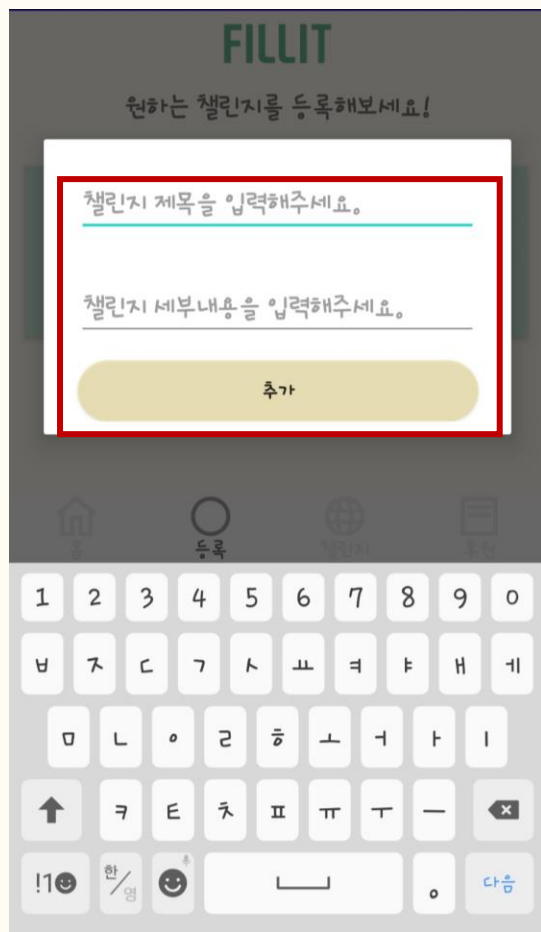
        ViewHolder(View itemView) {
            super(itemView) ;

            // 뷰 객체에 대한 참조. (hold strong reference)
            textTitle = itemView.findViewById(R.id.textTitle);
            textDescription = itemView.findViewById(R.id.textDescription);

            itemView.setOnClickListener((view) -> {
                int position = getAdapterPosition();
                mCallback.onClick(mData.get(position));
            });
        }
    }
}
```

## 03

## 'FILLIT' - 프로젝트 기능



## challengeEnrollMain.java

## ChallengeEnrollItemAdapter

- 카테고리별 챌린지 목록

-&gt; RecyclerView 이용

- 사용자가 입력하여 챌린지 추가

-&gt; AlertDialog 이용

```

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.challenge_enroll_main, container, attachToRoot: false);
    viewList = view.findViewById(R.id.recyclerView);
    viewList.setLayoutManager(new LinearLayoutManager(this.getContext()));
    InitializeChallengeData();

    // 부드럽게 넘기기
    PagerSnapHelper snapHelper = new PagerSnapHelper();
    snapHelper.attachToRecyclerView(viewList);

    adapter = new challengeEnrollItemAdapter(aCurrentData.ListChallengeEnroll, listener, this);
    viewList.setAdapter(adapter);

    plusbutton = view.findViewById(R.id.plusbutton);
    plusbutton.setOnClickListener((view) -> {
        AlertDialog.Builder builder = new AlertDialog.Builder(challengeEnrollMain.this.getContext());

        View view2 = LayoutInflater.from(challengeEnrollMain.this.getContext())
            .inflate(R.layout.edit_box, root: null, attachToRoot: false);
        builder.setView(view2);

        final Button buttonsubmit = (Button) view2.findViewById(R.id.button_dialog_submit);
        final EditText editTextTitle = (EditText) view2.findViewById(R.id.edittext_dialog_title);
        final EditText editTextSubtitle = (EditText) view2.findViewById(R.id.edittext_dialog_subtitle);

        final AlertDialog dialog = builder.create();
        // 3. 다이얼로그에 있는 삽입 버튼을 클릭하면
        buttonsubmit.setOnClickListener((v) -> {
            // 4. 사용자가 입력한 내용을 가져와서
            strTitle = editTextTitle.getText().toString();
            strSubtitle = editTextSubtitle.getText().toString();

            challengeItem newOne = new challengeItem();
            newOne.title = strTitle;
            newOne.des = strSubtitle;
        });
    });
}

```

## 03

## 'FILLIT' - 프로젝트 기능

FILLIT

마스크 분리배출

마스크 끈 잘라서 버리기

<	12월 2020						>
월	화	수	목	금	토	일	
30	1	2	3	4	5	6	
7	8	9	10	11	12	13	
14	15	16	17	18	19	20	
21	22	23	24	25	26	27	
28	29	30	31	1	2	3	

일 기준 포인트    100    전체 포인트    0

등록 기간    0

마스크 끈 잘라서 버리기

<	12월 2020						>
월	화	수	목	금	토	일	
30	1	2	3	4	5	6	
7	8	9	10	11	12	13	
14	15	16	17	18	19	20	
21	22	23	24	25	26	27	
28	29	30	31	1	2	3	

일 기준 포인트    100    전체 포인트    300

등록 기간    2020년 12월 10일 목요일 ~ 2020년 12월 12일 토요일

등록

## challengeErollItemSpecific.java

challengeItem

## - 챌린지 날짜 기간 등록

-&gt; CalendarView 이용

```

calendarView.setCalendarOrientation(OrientationHelper.HORIZONTAL);
calendarView.setSelectionType(SelectionType.RANGE);
calendarView.getConnectedDaysManager().setConnectedDaysList(null);
calendarView.setWeekendDayTextColor(Color.RED);
calendarView.setSelectionManager(new RangeSelectionManager(() -> {
    textPointTotal.setText(String.valueOf(item.chalPoint * calendarView.getSelectedDates().size()));
    final List<Calendar> days = calendarView.getSelectedDates();
    for( int i=0; i<days.size(); i++)
    {
        Calendar calendar = days.get(i);
        final int day = calendar.get(Calendar.DAY_OF_MONTH);
        final int month = calendar.get(Calendar.MONTH);
        final int year = calendar.get(Calendar.YEAR);
        String week = new SimpleDateFormat( pattern: "EE").format(calendar.getTime());
        if(i == 0)
        {
            String day_full = year + "년 " + (month+1) + "월 " + day + "일 " + week + "요일";
            item.regdate = day_full;
        }
        if(i == days.size()-1)
        {
            String day_full = year + "년 " + (month+1) + "월 " + day + "일 " + week + "요일";
            item.deadline = day_full;
        }
    }
    enrolldate.setText(item.regdate + " ~ " + item.deadline);
}));

```



## 03

## 'FILLIT' - 프로젝트 기능

FILLIT

달성시 획득 포인트: 800

등록기간:

2020년 12월 10일 목요일 ~ 2020년 12월 17일 목요일

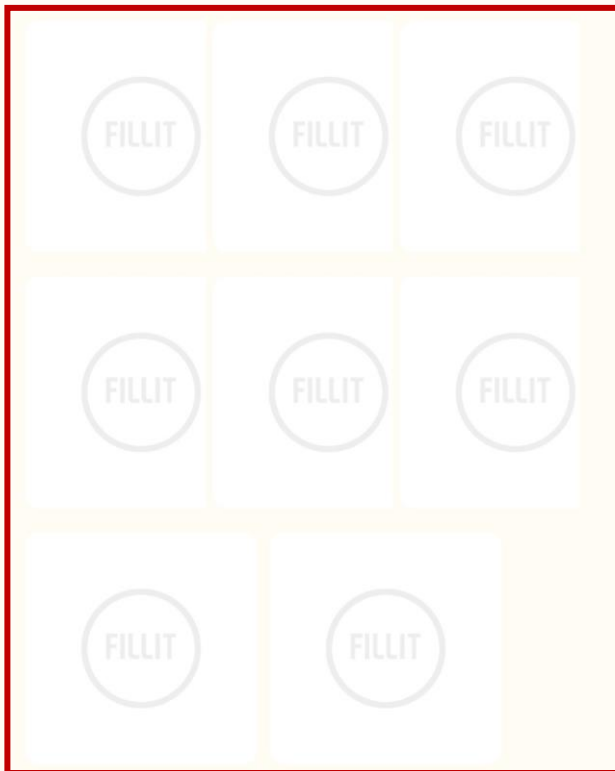
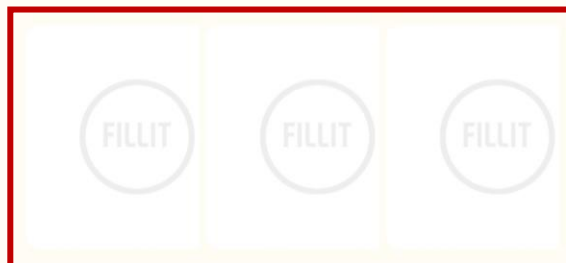
마스크 분리배출

마스크 끈 잘라서 버리기

달성시 획득 포인트: 300

등록기간:

2020년 12월 10일 목요일 ~ 2020년 12월 12일 토요일



challengeItemSpecific.java

challengeItem

- 챌린지 등록 기간 동안 1일당 1개 사진 업로드
- > 이미지 동적 생성

```

newActivity = new ImageView[item.chalPoint / 100];
// 이미지 동적 생성
for (int i = 0; i < item.chalPoint / 100; i++) {
    num = i;
    if (i % 3 == 0) {
        rowImages = new LinearLayout(this.getContext());
        rowImages.setOrientation(LinearLayout.HORIZONTAL);
        rowImages.setLayoutParams(settingRow);
    }

    newActivity[num] = new ImageView(this.getContext());
    newActivity[num].setTag(num);
    LinearLayout slotActivity = new LinearLayout(this.getContext());
    slotActivity.setOrientation(LinearLayout.HORIZONTAL);

    newActivity[num].setImageResource(R.drawable.image_default);

    newActivity[num].setOnClickListener(myListener);
    //number = num;

    LinearLayout.LayoutParams settingImage = new LinearLayout.LayoutParams( width: 400, height: 400);
    settingImage.setMargins( left: 10, top: 0, right: 10, bottom: 0);
    newActivity[num].setLayoutParams(settingImage);
    newActivity[num].setScaleType(ImageView.ScaleType.CENTER_CROP);
    LinearLayout.LayoutParams settingSlot = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.WRAP_CONTENT, ViewGroup.LayoutParams.WRAP_CONTENT);
    settingSlot.weight = 1;
    settingSlot.gravity = Gravity.CENTER;
    slotActivity.setLayoutParams(settingSlot);
    slotActivity.addView(newActivity[num]);
    rowImages.addView(slotActivity);
    if (i % 3 == 2 || i + 1 == item.chalPoint / 100) {
        layoutImages.addView(rowImages);
    }
}

```



# 03

## 'FILLIT' - 프로젝트 기능

### FILLIT

#### 마스크 분리배출

마스크 끈 잘라서 버리기

달성시 획득 포인트: 300

등록기간:

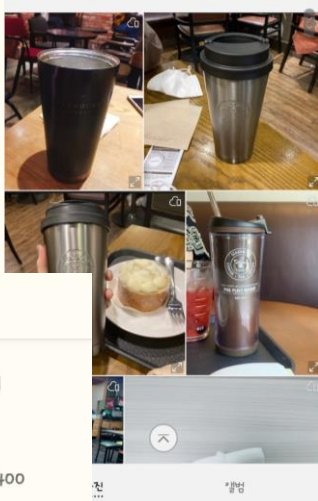
2020년 12월 10일 목요일 ~ 2020년 12월 12일 토요일

#### 업로드할 이미지 선택

일련번호

사진촬영

항목 선택



### FILLIT

#### 일회용 컵 사용하지 않기

카페 이용시 텀블러 사용하기

달성시 획득 포인트:

400

등록기간:

2020년 12월 10일 목요일 ~ 2020년 12월 13일 일요일



### FILLIT

## challengeItemSpecific.java

challengeItem

### - 챌린지 등록 기간 동안 1일당 1개 사진 업로드

-> Camera, Gallery 이용

```

if (requestCode == PICK_FROM_ALBUM) {
    Uri photoUri = data.getData();
    Log.d(TAG, "msg: "PICK_FROM_ALBUM photoUri : " + photoUri);

    Cursor cursor = null;

    try {
        String[] proj = {MediaStore.Images.Media.DATA};

        assert photoUri != null;
        cursor = getActivity().getContentResolver().query(photoUri, proj, selection: null, null, null);

        assert cursor != null;
        int column_index = cursor.getColumnIndexOrThrow(MediaStore.Images.Media.DATA);

        cursor.moveToFirst();

        tempFile = new File(cursor.getString(column_index));

        Log.d(TAG, "msg: "tempFile Uri : " + Uri.fromFile(tempFile));

    } finally {
        if (cursor != null) {
            cursor.close();
        }
    }

    setImage();

    requestCode, int resultCode, Intent data) {
        requestCode, resultCode, data);

        = RESULT_OK {
            new File(mCurrentPhotoPath);
            p = MediaStore.Images.Media
            Bitmap(getActivity().getContentResolver(), Uri.fromFile(file));
            = null) {
                if (ei = new ExifInterface(mCurrentPhotoPath);
                    intation = ei.getAttributeInt(ExifInterface.TAG_ORIENTATION,
                        ExifInterface.ORIENTATION_UNDEFINED);
                    rotatedBitmap = null;
                    orientation) {
                        ExifInterface.ORIENTATION_ROTATE_90:
                            rotatedBitmap = rotateImage(bitmap, angle: 90);
                            break;
                        ExifInterface.ORIENTATION_ROTATE_180:

    private void setImage() {
        BitmapFactory.Options options = new BitmapFactory.Options();
        Bitmap originalBm = BitmapFactory.decodeFile(tempFile.getAbsolutePath(), options);
        Log.d(TAG, "msg: "setImage : " + tempFile.getAbsolutePath());

        newActivity[number].setImageBitmap(originalBm);
        //num_setImage();
        tempFile = null;
        number+=1;
        challengeItem newOne = new challengeItem();
        newOne.progress+=1;
    }

```



## 03

## 'FILLIT' - 프로젝트 기능

## challengeErollItemSpecific.java

challengeItem

- Firebase 서버 연동하여 challengeItem의 데이터 저장

```
mDatabase = FirebaseDatabase.getInstance().getReference();
```

```
readUser();
HashMap resultdata = new HashMap<>();
resultdata.put("title", title);
resultdata.put("des", des);
resultdata.put("days", days);
resultdata.put("chalPoint", chalPoint);
resultdata.put("regdate", regdate);
resultdata.put("deadline", deadline);
```

```
writeNewUser(chID, title, des, days, chalPoint, regdate, deadline);
```

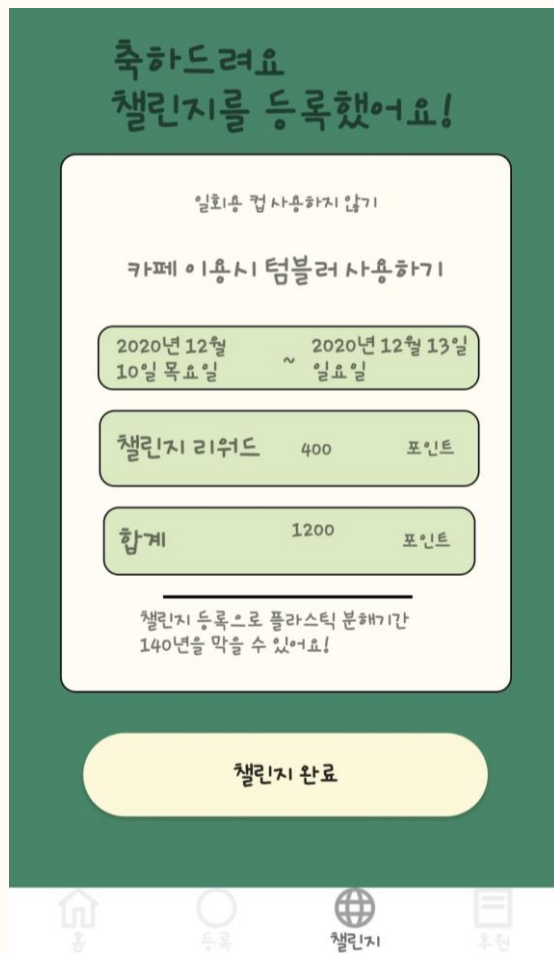
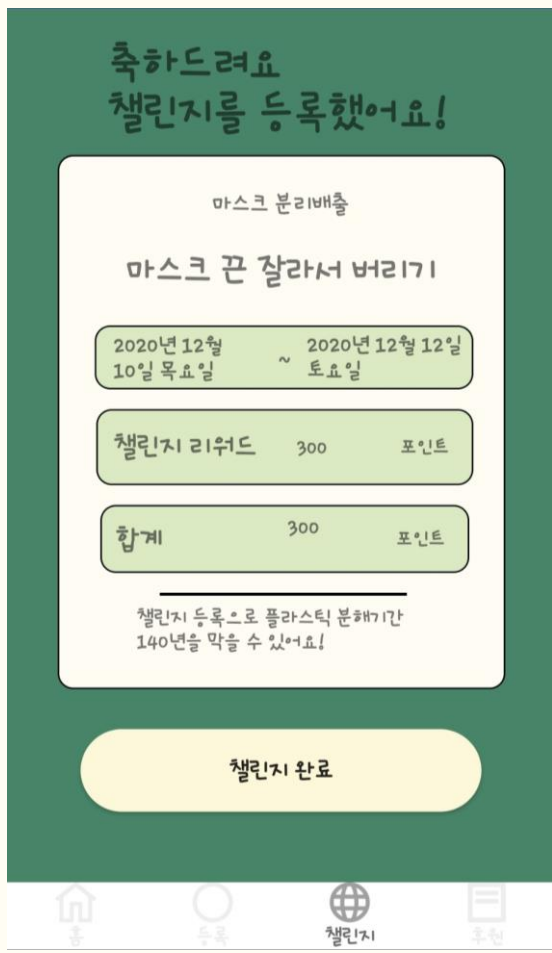
```
private void writeNewUser(int chID, String title, String des, int days, int chalPoint, String regdate, String deadline) {
    userChallenge userChallenge = new userChallenge(title, des, days, chalPoint, regdate, deadline);

    mDatabase.child(String.valueOf(nickname)).child(String.valueOf(chID)).setValue(userChallenge)
        .addOnSuccessListener((OnSuccessListener) (aVoid) -> {
            // Write was successful!
            Toast.makeText(getActivity(), text: "저장을 완료했습니다.", Toast.LENGTH_SHORT).show();
        })
        .addOnFailureListener((e) -> {
            // Write failed
            Toast.makeText(getActivity(), text: "저장을 실패했습니다.", Toast.LENGTH_SHORT).show();
        });
}

private void readUser(){
    mDatabase.child(String.valueOf(nickname)).child(String.valueOf(chID)).addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
            // Get Post object and use the values to update the UI
            if(dataSnapshot.getValue(userChallenge.class) != null){
                userChallenge post = dataSnapshot.getValue(userChallenge.class);
                Log.w( tag: "FireBaseData", msg: "getData" + post.toString());
            } else {
                Toast.makeText(getActivity(), text: "데이터 없음...", Toast.LENGTH_SHORT).show();
            }
        }
    })
}
```

# 03

## 'FILLIT' - 프로젝트 기능



### ChallengeComplete.java

- Firebase 서버와 연동하여 데이터 저장
- 등록 완료된 챌린지 목록
- 최종 포인트 업데이트

finalproject-dc5fa-default-rtdb

김유진

0

chalPoint: 300

days: 3

deadline: "2020년 12월 12일 토요일"

des: "마스크 끈 잘라서 버리기"

regdate: "2020년 12월 10일 목요일"

title: "마스크 분리배출"

1

chalPoint: 400

days: 4

deadline: "2020년 12월 13일 일요일"

des: "카페 이용시 텀블러 사용하기"

regdate: "2020년 12월 10일 목요일"

title: "일회용 컵 사용하지 않기"

# 03

## 'FILLIT' - 프로젝트 기능



### challengeMain.java

### ChallengeItemAdapter

- 등록된 챌린지 목록
  - > RecyclerView 이용
- Firebase 서버 연동하여 challengeItem 데이터 읽기

```
mDatabase = FirebaseDatabase.getInstance().getReference();

// Read from the database
mDatabase.child(String.valueOf(nickname)).addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        challengeItem newOne = new challengeItem();
        newOne.title = dataSnapshot.child("title").getValue(String.class);
        newOne.des = dataSnapshot.child("des").getValue(String.class);
        newOne.days = dataSnapshot.child("days").getValue(Integer.class);
        newOne.chalPoint = dataSnapshot.child("chalPoint").getValue(Integer.class);
        newOne.regdate = dataSnapshot.child("regdate").getValue(String.class);
        newOne.deadline = dataSnapshot.child("deadline").getValue(String.class);
        aCurrentData.listMyChallenge.add(newOne);
        adapter.notifyDataSetChanged();
        chID2++;
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {
        Log.w( tag: "TAG: ", msg: "Failed to read value", databaseError.toException());
    }
});
```

## 03

## 'FILLIT' - 프로젝트 기능

## fundingMain.java

## fundingItemAdapter

## - 환경 관련 기업 목록

## -&gt; RecyclerView 이용

FILLIT

그린피스

GREENPEACE

그린피스는 1971년 태어난 독립적인 국제환경단체로 지구 환경보호와 평화를 위해 비폭력 직접행동의 평화적인 방식으로 캠페인을 진행하고 있습니다.

목표 포인트	50000
후원 포인트	11000
잔여 포인트	39000

후원 하기

FILLIT

WWF



WWF는 지구상의 다양한 생명체와 이들이 서식하는 아름다운 자연환경을 보전하는 일을 하고 있습니다. 이와 함께 인류가 동식물과 자연환경에 미치는 영향을 줄이는데 힘쓰고 있습니다

목표 포인트	10000
후원 포인트	9500
잔여 포인트	500

후원 하기

FILLIT

생명의 숲



생명의 숲은 시민의 힘으로 나무를 심고 숲을 가꾸고 보전하며, 숲의 공공성을 높여 누구나 숲의 가치를 누리는 건강한 사회를 만들어 가고자 합니다.

목표 포인트	200000
후원 포인트	64000
잔여 포인트	136000

후원 하기

```
public void onClick(fundingItem newOne) {
    ConstraintLayout funding_list = viewList.findViewById(R.id.funding_list);

    TextView funding = viewList.findViewById(R.id.funding);
    TextView inst_desc = viewList.findViewById(R.id.inst_desc);

    Animation anim = AnimationUtils.loadAnimation(getActivity(), R.anim.funding_translate_top);
    funding_list.startAnimation(anim);

    inst_desc.setVisibility(View.GONE);
    funding.setVisibility(View.GONE);

    funding_list.setVisibility(View.VISIBLE);
}

@Override
public void onScroll(fundingItem newOne) {
    TextView textNowPoint = viewList.findViewById(R.id.textNowPoint);
    TextView textPoint = viewList.findViewById(R.id.textPoint);
    TextView textRestPoint = viewList.findViewById(R.id.textRestPoint);

    textNowPoint.setVisibility(View.VISIBLE);
    textPoint.setVisibility(View.VISIBLE);
    textRestPoint.setVisibility(View.VISIBLE);

    Animation anim = AnimationUtils.loadAnimation(getActivity(), R.anim.funding_translate_top);
    viewList.startAnimation(anim);
}

@Override
public void onFundingClick(fundingItem newOne) {
    ConstraintLayout funding_list = viewList.findViewById(R.id.funding_list);
    ConstraintLayout funding_click = viewList.findViewById(R.id.funding_click);
    pointSpent = viewList.findViewById(R.id.fund_point);
}
```

# 03

## 'FILLIT' - 프로젝트 기능

FILLIT

FILLIT

그린피스

GREENPEACE

WWF



후원하기

나의 현재 포인트 700 포인트



나의 현재 포인트

후원 포인트 210 포인트

잔여 포인트 490 포인트

후원 완료

후원하기

나의 현재 포인트 700 포인트



잔여 포인트

후원 포인트 0 포인트

잔여 포인트 700 포인트

후원 완료

fundingMain.java

fundingItemAdapter

### - 후원하기

-> animation, seekbar 이용

-> 나의 현재 포인트 또는 잔여 포인트 중 더 큰 값을 최대로 후원 가능

-> 후원시 후원 포인트, 전체 포인트 증가

```
seekBar.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {

        int position = getAdapterPosition();
        tar_pointt = mData.get(position).tar_point;
        final int acu_point = mData.get(position).acu_point;
        final int left_pointt = tar_pointt - acu_point;
        final int seek_max = min(mytotalPoint, left_pointt);
        fund_point.setText(String.valueOf(seekBar.getProgress() * (seek_max/10)));
        fund_pointt = seekBar.getProgress() * (seek_max/10);

        int rest = mytotalPoint - progress * (seek_max/10);
        if(rest >= 0)
            rest_point.setText(String.valueOf(rest));
        else
            rest_point.setText(String.valueOf(0));

        override
        public void onStartTrackingTouch(SeekBar seekBar)

        override
        public void onStopTrackingTouch(SeekBar seekBar) {
            //point.setText("세네로 돌아감");
        }
    }
});
```

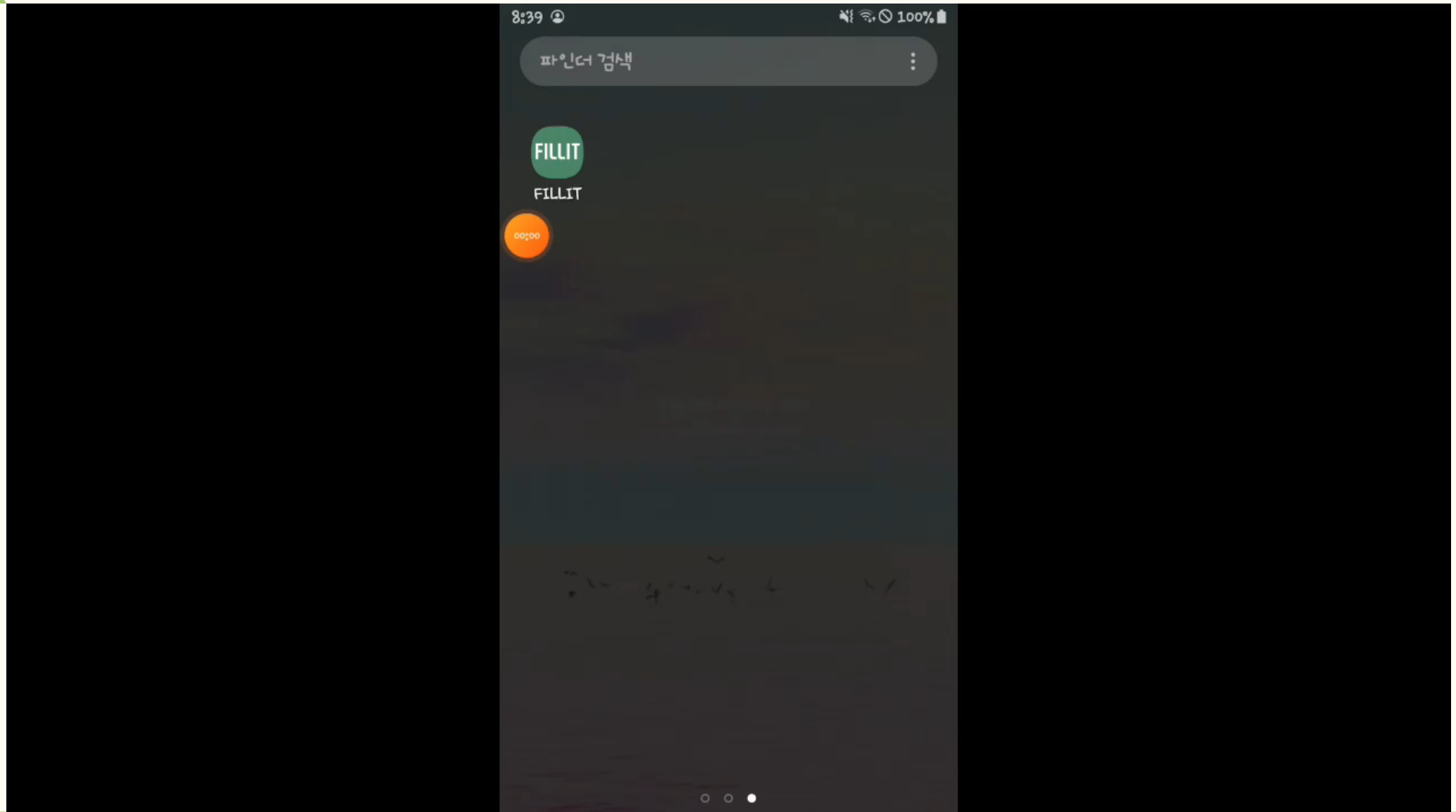
감사합니다

김유진 님

후원이 완료되었어요!

FILLIT





Q & A