```java
package com.hakankursun.adapter;

import java.io.File;
import java.util.List;

import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.net.Uri;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.LayoutInflater;
import android.view.MotionEvent;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.View.OnTouchListener;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.LinearLayout;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.RadioGroup.OnCheckedChangeListener;
import android.widget.TextView;

import com.jeremyfeinstein.slidingmenu.lib.app.SlidingFragmentActivity;
import com.hakankursun.R;
import com.hakankursun.menu.MainActivity;
import com.hakankursun.pojo.Cevap;
```

```java
import com.hakankursun.util.Const;

import com.hakankursun.util.Util;


public class CheckListDuzenleListAdapter extends BaseAdapter implements
                OnClickListener {
        private static List<Cevap> cevap;

        private Context context;

        private static ViewImageBtn viewImageBtn;

        private ViewHolder holder;


        public CheckListDuzenleListAdapter(Context context, List<Cevap> cevap) {
                this.context = context;

                this.cevap = cevap;

        }



        @Override
        public View getView(final int position, View convertView, ViewGroup parent) {


                View v = convertView;


                if (v == null) {
                        LayoutInflater vi = (LayoutInflater) context
                                        .getSystemService(Context.LAYOUT_INFLATER_SERVICE);

                        v = vi.inflate(R.layout.row_checklist_ekle_item, null);

                        holder = new ViewHolder();


                        holder.txtSoru = (TextView) v.findViewById(R.id.txtSoru);

                        holder.radioGroup = (RadioGroup) v.findViewById(R.id.radioGroup);

                        holder.editAciklama = (EditText) v.findViewById(R.id.editAciklama);

                        holder.btnFotoCek = (ImageButton) v.findViewById(R.id.btnFotoCek);
```

```java
holder.btnResimGaleriSec = (ImageButton) v
        .findViewById(R.id.btnResimGaleriSec);
holder.btnResimOnizle = (ImageButton) v
        .findViewById(R.id.btnResimOnizle);
holder.btnResimSil = (ImageButton) v.findViewById(R.id.btnResimSil);
holder.btnAciklama = (ImageButton) v.findViewById(R.id.btnAciklama);
holder.lytDetay = (LinearLayout) v.findViewById(R.id.lytDetay);
holder.lytDetay.setVisibility(View.GONE);


holder.txtSoru.setTag(holder.lytDetay);
holder.btnAciklama.setTag(holder.editAciklama);


ViewImageBtn viewImageBtn = new ViewImageBtn();
viewImageBtn.btnResimOnizle = holder.btnResimOnizle;
viewImageBtn.btnResimSil = holder.btnResimSil;
viewImageBtn.position = position;


holder.btnFotoCek.setTag(viewImageBtn);
holder.btnResimGaleriSec.setTag(viewImageBtn);


holder.btnResimOnizle.setTag(viewImageBtn);
holder.btnResimSil.setTag(viewImageBtn);


holder.txtSoru.setOnClickListener(this);
holder.btnAciklama.setOnClickListener(this);
holder.btnFotoCek.setOnClickListener(this);
holder.btnResimGaleriSec.setOnClickListener(this);
holder.btnResimOnizle.setOnClickListener(this);
holder.btnResimSil.setOnClickListener(this);


holder.btnResimOnizle.setVisibility(View.GONE);
```

```java
                    holder.btnResimSil.setVisibility(View.GONE);


                    holder.editAciklama.setOnTouchListener(new OnTouchListener() {
                            // Setting on Touch Listener for handling the touch inside
                            // ScrollView
                            @Override
                            public boolean onTouch(View v, MotionEvent event) {
                                    // Disallow the touch request for parent scroll on touch of
                                    // child view
                                    v.getParent().requestDisallowInterceptTouchEvent(true);
                                    return false;
                            }
                    });


        holder.editAciklama.addTextChangedListener(new TextWatcher() {
                public void afterTextChanged(Editable s) {
                        cevap.get(position).setAciklama(s.toString());
                }


                public void onTextChanged(CharSequence s, int start, int before,
                                int count) {
                        // Do nothing.
                }


                public void beforeTextChanged(CharSequence s, int start, int count,
                                int after) {
                        // Do nothing.
                }
        });


        holder.radioGroup
```

```java
                        .setOnCheckedChangeListener(new OnCheckedChangeListener() {


                @Override

                public void onCheckedChanged(RadioGroup group, int
checkedId) {


                                RadioButton radioButton = (RadioButton) group
                                        .findViewById(checkedId);
                                Byte idx = (byte) group.indexOfChild(radioButton);
                                cevap.get(position).setCevap(idx);
                        }
                });



        v.setTag(holder);
} else {
        holder = (ViewHolder) v.getTag();
}






Cevap cevapItem = getItem(position);


holder.txtSoru.setText((position + 1)
                + context.getString(R.string.nsoru) + cevapItem.getSoru());


holder.editAciklama.setText(cevapItem.getAciklama());


RadioButton radioButton = (RadioButton) holder.radioGroup
```

```java
                        .getChildAt(cevapItem.getCevap());
                radioButton.toggle();



                if (cevapItem.getResim() != null) {
                        File file = new File(cevapItem.getResim());
                        if (file.exists()) {
                                holder.btnResimOnizle.setVisibility(View.VISIBLE);
                                holder.btnResimSil.setVisibility(View.VISIBLE);
                        } else {
                                holder.btnResimOnizle.setVisibility(View.GONE);
                                holder.btnResimSil.setVisibility(View.GONE);
                                cevap.get(position).setResim(null);
                        }
                }



                return v;

        }

        private class ViewHolder {

                public TextView txtSoru;
                public RadioGroup radioGroup;
                public ImageButton btnFotoCek, btnResimGaleriSec, btnResimOnizle,
                                btnResimSil, btnAciklama;
                public EditText editAciklama;
                public LinearLayout lytDetay;

        }
```

```java
@Override
public int getCount() {
        return cevap.size();
}


@Override
public Cevap getItem(int position) {
        return cevap.get(position);
}


@Override
public long getItemId(int position) {
        return position;
}


@Override
public void onClick(View v) {
        switch (v.getId()) {
        case R.id.txtSoru:
                LinearLayout lytDetay = (LinearLayout) v.getTag();
                if (lytDetay.getVisibility() == View.VISIBLE) {
                        lytDetay.setVisibility(View.GONE);
                } else {


                        lytDetay.setVisibility(View.VISIBLE);
                }
                break;
        case R.id.btnAciklama:
                Const.TextViewToSpeach = (EditText) v.getTag();
                MainActivity.startSpeachToText();
                break;
```

```java
case R.id.btnFotoCek:

        viewImageBtn = (ViewImageBtn) v.getTag();

        MainActivity.startImageCapture();

        break;
case R.id.btnResimGaleriSec:

        viewImageBtn = (ViewImageBtn) v.getTag();

        MainActivity.startGalleryPick();

        break;


case R.id.btnResimOnizle:

        viewImageBtn = (ViewImageBtn) v.getTag();

        File file = new File(cevap.get(viewImageBtn.position).getResim());

        if (file.exists()) {

                Intent intent = new Intent();

                intent.setAction(Intent.ACTION_VIEW);

                intent.setDataAndType(

                                Uri.parse("file://" +
cevap.get(viewImageBtn.position).getResim()),

                                "image/*");

                ((SlidingFragmentActivity) context).startActivity(intent);


        } else {

                viewImageBtn.btnResimOnizle.setVisibility(View.GONE);

                viewImageBtn.btnResimSil.setVisibility(View.GONE);

                cevap.get(viewImageBtn.position).setResim(null);

                Util.TamamDialog(context, context.getString(R.string.uyari),

                                context.getString(R.string.resim_bulunamadi));

        }


        break;
case R.id.btnResimSil:
```

```java
                    viewImageBtn = (ViewImageBtn) v.getTag();


                    AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(
                            context);
                    alertDialogBuilder.setTitle(context.getString(R.string.uyari));
                    alertDialogBuilder.setMessage(context
                            .getString(R.string.checklist_resim_sil));


                    alertDialogBuilder.setPositiveButton(context.getResources()
                            .getString(R.string.evet),
                            new DialogInterface.OnClickListener() {
                                public void onClick(final DialogInterface dialog,
                                        final int which) {

cevap.get(viewImageBtn.position).setResim(null);
                                        viewImageBtn.btnResimOnizle
                                                .setVisibility(View.GONE);

viewImageBtn.btnResimSil.setVisibility(View.GONE);
                                }
                            });


                    alertDialogBuilder.setNegativeButton(context.getResources()
                            .getString(R.string.hayir),
                            new DialogInterface.OnClickListener() {
                                public void onClick(final DialogInterface dialog,
                                        final int which) {


                                }
                            });


                    AlertDialog dialog = alertDialogBuilder.create();
```

```java
                dialog.show();

                break;

        default:

                break;

        }


    }


    public class ViewImageBtn {

        public ImageButton btnResimOnizle, btnResimSil;

        public int position;

    }


    public static void setResim(String imagePath) {

        viewImageBtn.btnResimOnizle.setVisibility(View.VISIBLE);

        viewImageBtn.btnResimSil.setVisibility(View.VISIBLE);

        cevap.get(viewImageBtn.position).setResim(imagePath);

    }


    public List<Cevap> getCevapList() {

        return cevap;

    }

}
```