

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.cookbook">
    <uses-feature android:name="android.hardware.camera"
        android:required="false"
    />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".RecipeListActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".RecipeActivity"
            android:parentActivityName=".RecipeListActivity">
        </activity>
        <activity android:name=".RecipeDisplayActivity"
            android:parentActivityName=".RecipeListActivity">
        </activity>
        <provider
            android:name="android.support.v4.content.FileProvider"
            android:authorities="com.example.cookbook.fileprovider"
            android:exported="false"
            android:grantUriPermissions="true">
            <meta-data
                android:name="android.support.FILE_PROVIDER_PATHS"
                android:resource="@xml/files"/>
        </provider>
    </application>

</manifest>
```

```
package com.example.cookbook;

import java.util.Date;

public class Comment {
    private String mTxt;
    private User mUser;
    private Date mDate;

    public Comment(String s, User u){
        mTxt=s;
        mUser=u;
        mDate=new Date();
    }

    public String getTxt() {
        return mTxt;
    }

    public User getUser() {
        return mUser;
    }

    public Date getDate() {
        return mDate;
    }

    public void setDate(Date date) {
        mDate = date;
    }

    @Override
    public String toString(){
        return mTxt+" (" +mUser.getNameComplete()+")";
    }
}
```

```
package com.example.cookbook;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.util.Log;

import java.io.File;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.List;
import java.util.Random;
import java.util.UUID;

public class CookBook {
    private static CookBook ourInstance ;
    // private List<Recipe> mRecipes;
    private Context mContext;
    private SQLiteDatabase mDatabase;
    private static final String TAG = "DebugCookbook";

    public static CookBook get(Context context) {
        if (ourInstance==null){
            ourInstance= new CookBook(context);
        }
        return ourInstance;
    }

    private CookBook(Context context) {
        mContext=context.getApplicationContext();
        mDatabase=new RecipeBaseHelper(mContext)
            .getWritableDatabase();
        //For initialisation of recipedB database : adding random recipes
        //for (int i = 0; i < 5; i++){addRecipe(randomRecipe());}
        //addRecipe(MyOldRecipe());
    }

    public Recipe getRecipe(UUID id) {
        RecipeCursorWrapper cursor=queryRecipes(
            RecipeDbSchema.RecipeTable.Cols.UUID+" =?",
            new String[] {id.toString()})
        );
        try{
            if (cursor.getCount()==0) {return null;}
            cursor.moveToFirst();
            return cursor.getRecipe();
        } finally {cursor.close();}
    }
}
```

```

public void updateRecipe(Recipe r) {
    String uuidString=r.getId().toString();
    r.setDate(new Date()); //last modification
    ContentValues values=getContentValues(r);
    mDatabase.update(RecipeDbSchema.RecipeTable.NAME, values,
        RecipeDbSchema.RecipeTable.Cols.UUID+"=?",
        new String[] {uuidString});
}

private RecipeCursorWrapper queryRecipes(String whereClause, String[] whereArgs){
    Cursor cursor=mDatabase.query(
        RecipeDbSchema.RecipeTable.NAME,
        null, // select all columns
        whereClause,
        whereArgs,
        null, null, null
    );
    return new RecipeCursorWrapper(cursor);
}

public void addRecipe(Recipe r){
//    mRecipes.add(r);
    ContentValues values=getContentValues(r);
    mDatabase.insert(RecipeDbSchema.RecipeTable.NAME, null, values);
}

public void removeRecipe(Recipe r){
    String uuidString=r.getId().toString();
    mDatabase.delete(RecipeDbSchema.RecipeTable.NAME,
        RecipeDbSchema.RecipeTable.Cols.UUID+"=?",
        new String[] {uuidString});
    return;
}

public List<Recipe> getRecipes(){
    List<Recipe> recipes=new ArrayList<>();
    RecipeCursorWrapper cursor=queryRecipes(null, null);
    try{
        cursor.moveToFirst();
        while (!cursor.isAfterLast()){
            recipes.add(cursor.getRecipe());
            cursor.moveToNext();
        }
    } finally { cursor.close();}
    return recipes;
}

private static ContentValues getContentValues(Recipe recipe) {
    ContentValues values=new ContentValues();
    values.put(RecipeDbSchema.RecipeTable.Cols.UUID, recipe.getId().toString());
    values.put(RecipeDbSchema.RecipeTable.Cols.OWNER, recipe.getSerializedOwner());
    values.put(RecipeDbSchema.RecipeTable.Cols.TITLE, recipe.getTitle());
    values.put(RecipeDbSchema.RecipeTable.Cols.SOURCE, recipe.getSource());
    values.put(RecipeDbSchema.RecipeTable.Cols.SOURCE_URL,

```



```

recipe.getSource_url().toString());
    values.put(RecipeDbSchema.RecipeTable.Cols.DATE, recipe.getDate().getTime());
    values.put(RecipeDbSchema.RecipeTable.Cols.NBPERS, recipe.getNbPers());
    for(int i=0;i<recipe.getNbStepMax();i++){
        values.put(RecipeDbSchema.RecipeTable.Cols.STEP[i], recipe.getStep(i+1));
    }
    for(int i=0;i<recipe.getNbIngMax();i++){
        values.put(RecipeDbSchema.RecipeTable.Cols.ING[i], recipe.getIngredient(i
+1));
    }
    values.put(RecipeDbSchema.RecipeTable.Cols.SEASON, recipe.getSeason().name());
    values.put(RecipeDbSchema.RecipeTable.Cols.DIFFICULTY,
recipe.getDifficulty().name());
    values.put(RecipeDbSchema.RecipeTable.Cols.COMMENTS,
recipe.getSerializedComments());
    values.put(RecipeDbSchema.RecipeTable.Cols.NOTES, recipe.getSerializedNotes());
    return values;
}

public File getPhotoFile(Recipe r){
    if (r==null) {return null;}
    File filesDir= mContext.getFilesDir();
    return new File(filesDir, r.getPhotoFilename());
}

public Boolean deleteImage(Recipe r){
    Boolean success=false;
    File filesDir= mContext.getFilesDir();
    File im=new File(filesDir, r.getPhotoFilename());
    if (im.exists()){
        im.delete();
        success=true;
    }
    return success;
}

// methode pour generer des recettes aléatoires pour la phase de mise au point
private Recipe randomRecipe(){
    Recipe r;
    Random rand=new Random();
    // initialisations users
    User fab=new User("Devaux_Lion de ML","Fabrice");
    fab.setId(UUID.fromString("c81d4e2e-bcf2-11e6-869b-7df92533d2db"));
    User lucile=new User("Devaux_Lion de ML","Lucile");
    lucile.setId(UUID.fromString("c81d4e2e-bcf2-11e7-869b-7df92533d2db"));
    User vero=new User("Devaux_Lion de ML","Véronique");
    vero.setId(UUID.fromString("c81d4e2e-bcf3-11e6-869b-7df92533d2db"));
    // Init pour alea
    User[] mUserName={fab, lucile, vero};
    String[] mIngPrinc={"Poulet","Boeuf","Canard","Thon", "Homard"};
    String[] mFacon={"a la moutarde","a la marocaine", "laque", "a la mexicaine",
"a l indienne", "braise", "saute"};
    String[] mSource={"Marmiton","Cyril Lignac", "Fabrice", "Veronique", "Cuisine
actuelle"};

```

```

String[] mS1={"Coupez","Tranchez", "Emincez","Coupez en lamelles","Coupez en
cubes"};
String[] mS2={"les tomates","les oignons", "les poivrons","les pommes de
terre","les aubergines"};
String[] mS3={"le poivre", "le curry", "la sauce soja", "le cumin", "le
piment"};
String[] mS4={"Mettre au four",
"Faites revenir","Faites bouillir","Saisir","Faites mijoter"};
String[] mURL={"http://www.marmiton.org","http://www.cuisinechef.com","http://
www.lacuisinedefabrice.fr"};
RecipeSeason[]
mSeason={RecipeSeason.WINTER,RecipeSeason.SUMMER,RecipeSeason.ALLYEAR};
RecipeDifficulty[] mDifficulty={RecipeDifficulty.EASY, RecipeDifficulty.QUICK,
RecipeDifficulty.ELABORATE, RecipeDifficulty.SOPHISTICATED };
String[] iNum={"5","20","3","100","2"};
String[] iUnit={"g","dl","cuillerées à café", "litres", "pincées"};
String[] iIngt={"poivre moulu","ail","lait","porc", "sucre","boeuf","pomme de
terre","pruneaux"};
User[] iNotesAuthor={fab, lucile, vero, lucile, vero};
String[] iComment={"Plus de sel", "Avec des courgettes, c'est délicieux", "A
servir avec du riz", "J'aime", "Un peu fade!",
"Bon pour l'hiver", "Plus de poivre", "Un peu juste en quantité"}; //8
//setting
r=new Recipe();
r.setOwner(mUserName[rand.nextInt(3)]);
String s1=mIngPrinc[rand.nextInt(5)];
r.setTitle(s1+" "+mFacon[rand.nextInt(7)]);
r.setSource(mSource[rand.nextInt(5)]);
String urls=mURL[rand.nextInt(3)];
try {
    URL url= new URL(urls);
    r.setSource_url(url);
} catch (MalformedURLException e){
    Log.d(TAG, "random recipe URL >" +urls+"< Failed");}
r.setSeason(mSeason[rand.nextInt(3)]);
r.setDifficulty(mDifficulty[rand.nextInt(4)]);
r.setNbPers(3+rand.nextInt(3));
r.setStep(1,mS1[rand.nextInt(5)]+" le "+s1+".");
r.setStep(2,"Apres quelques minutes, ajoutez au "+s1+
" "+mS2[rand.nextInt(5)]+
" et "+mS2[rand.nextInt(5)]);
r.setStep(3,"Salez, poivrez et ajoutez "+mS3[rand.nextInt(5)]);
r.setStep(4,mS4[rand.nextInt(5)]+ " pendant environ "+ (5+ rand.nextInt(10))+ "
minutes.");
r.setStep(5,"");
r.setStep(6,"");
r.setStep(7,"");
r.setStep(8,"");
r.setStep(9,"");
r.setDate(new Date());
int ix=rand.nextInt(7)+3;
for(int i=0; i<ix;i++){
    r.setIngredient(i+1,iNum[rand.nextInt(5)]+" "+
iUnit[rand.nextInt(5)]+" de "+iIngt[rand.nextInt(8)]);

```

```

    }
    Date d=new Date();
    Calendar c = Calendar.getInstance();
    Comment comment;
    Note note;
    for(int i=0; i<rand.nextInt(5);i++){
        c.set(2000+rand.nextInt(20),rand.nextInt(12),1+rand.nextInt(27), 0, 0);
        comment=new
Comment(iComment[rand.nextInt(8)],iNotesAuthor[rand.nextInt(5)]);
        note=new Note(1+rand.nextInt(5),iNotesAuthor[rand.nextInt(5)]);
        d=c.getTime();
        comment.setDate(d);
        note.setDate(d);
        r.addComment(comment);
        r.addNote(note);
    }
    c.set(2018+rand.nextInt(3),rand.nextInt(12),1+rand.nextInt(27), 0, 0);
    r.setDate(c.getTime());
    return r;
}

// ma recette de lasagne
private Recipe MyOldRecipe(){
    Recipe r;
    User fab=new User("Devaux_Lion de ML","Fabrice");
    fab.setId(UUID.fromString("c81d4e2e-bcf2-11e6-869b-7df92533d2db"));
    r=new Recipe();
    r.setOwner(fab);
    r.setId(UUID.fromString("f819bcc2-ab09-4ed8-8a7c-98fade172da2")); // to recover
the image
    r.setTitle("Lasagnes pour Lucile");
    r.setSource("Moi");
    String urls="https://www.familycookbook.com";
    try {
        URL url= new URL(urls);
        r.setSource_url(url);
    } catch (MalformedURLException e){
        Log.d(TAG, "lasagne recipe URL >" +urls+"< Failed");}
    r.setSeason(RecipeSeason.ALLYEAR);
    r.setDifficulty(RecipeDifficulty.EASY);
    r.setNbPers(4);
    r.setStep(1,"Émincer oignons et carottes. Faire fondre beurre dans une grande
poêle sur 7. Mettre les oignons. Réduire a 6. Couvert 5 min.");
    r.setStep(2,"Rassembler l'oignon fondu dans un coin. Faire revenir la viande
sur 8 qq min.");
    r.setStep(3,"Ajouter carottes, concentré, tomates en morceaux, poivrons, sel,
poivre");
    r.setStep(4,"Couvrir et réduire sur 6. Laisser mijoter 15 minutes.");
    r.setStep(5,"Dans un grand plat, disposer une couche de lasagnes, une couche de
sauce mijotée, de fromage et bouts de mozza. Recommencer deux fois.");
    r.setStep(6,"Finir avec un peu de crème, une couche de lasagnes, puis napper de
béchamel.");
    r.setStep(7,"Enfourner a 180 C pendant 30 minutes.");
    r.setStep(8,"");

```

```
r.setStep(9, "");
r.setDate(new Date());
r.setIngredient(1, "1 oignon");
r.setIngredient(2, "400 g de viande hachée");
r.setIngredient(3, "1 petite boîte de concentré de tomate");
r.setIngredient(4, "3 tomates");
r.setIngredient(5, "1 boule de mozza");
r.setIngredient(6, "2 carottes");
r.setIngredient(7, "des poivrons");
r.setIngredient(8, "15 feuilles de lasagnes");
r.setIngredient(9, "200 g de fromage râpé");
r.setIngredient(10, "de la bechamel");
r.setIngredient(11, "de la crème fraîche (option)");
r.setIngredient(12, "");
return r;
```

```
}
```

```
}
```



```
package com.example.cookbook;

import java.util.Date;

public class Note {
    private Integer mNote;
    private User mUser;
    private Date mDate;

    public Note(Integer note, User u){
        mNote=note;
        mUser=u;
        mDate=new Date();
    }

    public Integer getNote() {
        return mNote;
    }

    public User getUser() {
        return mUser;
    }

    public Date getDate() {
        return mDate;
    }

    public void setNote(Integer note) {
        mNote = note;
    }

    public void setUser(User user) {
        mUser = user;
    }

    public void setDate(Date date) {
        mDate = date;
    }
}
```

```
package com.example.cookbook;

import android.app.Activity;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Point;

import java.io.File;

public class PictureUtils {
    public static Bitmap getScaledBitmap(String path, int destWidth, int destHeight) {
        BitmapFactory.Options options=new BitmapFactory.Options();
        options.inJustDecodeBounds=true;
        BitmapFactory.decodeFile(path, options);
        float srcWidth=options.outWidth;
        float srcHeight=options.outHeight;
        int inSampleSize=1;
        if (srcHeight>destHeight || srcWidth>destWidth) {
            float heightScale=srcHeight / destHeight;
            float widthScale=srcWidth/destWidth;
            inSampleSize=Math.round(heightScale > widthScale ? heightScale :
widthScale);
        }
        options=new BitmapFactory.Options();
        options.inSampleSize=inSampleSize;
        return BitmapFactory.decodeFile(path, options);
    }
    @SuppressWarnings("deprecation")
    public static Bitmap getScaledBitmap(String path, Activity activity){
        Point size=new Point();
        activity.getWindowManager().getDefaultDisplay().getSize(size);
        return getScaledBitmap(path, size.x, size.y);
    }
}
```

```
package com.example.cookbook;

import android.app.Activity;
import android.app.AlertDialog;
import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.DialogFragment;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.Button;
import android.widget.RatingBar;

import java.util.List;
import java.util.UUID;

public class RatePickerFragment extends DialogFragment {
    public static final String EXTRA_RATE = "com.example.cookbook.rate";
    private static final String ARG_ID= "recipeId";
    private static final String TAG = "RatePickerFragment";
    private int mRate;
    private UUID mUUID;
    public static RatePickerFragment newInstance(UUID uuid) {
        Bundle args = new Bundle();
        args.putSerializable(ARG_ID, uuid);
        RatePickerFragment fragment = new RatePickerFragment();
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public Dialog onCreateDialog(Bundle savedInstanceState) {
        mUUID = (UUID) getArguments().getSerializable(ARG_ID);
        View v= LayoutInflater.from(getActivity())
            .inflate(R.layout.rank_dialog, null);
        RatingBar ratingBar = (RatingBar)v.findViewById(R.id.dialog_ratingbar);
        ratingBar.setRating(4);
        AlertDialog.Builder dialog= new AlertDialog.Builder(getActivity())
            .setView(v)
            .setTitle(R.string.dialog_star_title)
            .setPositiveButton(android.R.string.ok,
                new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        mRate=(int) ratingBar.getRating();
                        addNotetoRecipe();
                        sendResult(Activity.RESULT_OK, mRate);
                    }
                });
        AlertDialog dia=dialog.show();
        Button b=dia.getButton(DialogInterface.BUTTON_POSITIVE);
        if (b != null) {
```

```
        b.setTextColor(getResources().getColor(R.color.fg_icon));
    }
    return dia;
}

private void sendResult(int resultCode, int rate) {
    if (getTargetFragment() == null) {
        return;
    }
    // renvoyer le new rating (cela ne sert à rien, juste pour essayer)
    Intent intent = new Intent();
    intent.putExtra(EXTRA_RATE, rate);
    getTargetFragment()
        .onActivityResult(getTargetRequestCode(), resultCode, intent);
}

private void addNotetoRecipe(){
    SessionInfo loSession= SessionInfo.get(getActivity());
    CookBook locookbook=CookBook.get(getActivity());
    Recipe r=locookbook.getRecipe(mUUID);
    User lu=loSession.getUser();
    r.addNote(new Note(mRate, lu));
    locookbook.updateRecipe(r);
}
}
```

```
package com.example.cookbook;

import android.util.Log;

import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
import com.google.gson.reflect.TypeToken;

import java.lang.reflect.Type;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Date;
import java.util.UUID;

public class Recipe {
    private UUID mId;
    private User mOwner;
    private Date mDate; //date de dernière modification en local
    private String mTitle;
    private String mSource;
    private URL mSource_url;
    private int mNbPers;
    private String[] mSteps;
    private double mNoteAvg;
    private ArrayList<Note> mNotes;
    private ArrayList<Comment> mComments;
    private RecipeSeason mSeason;
    private RecipeDifficulty mDifficulty;
    private String[] mIngredients;

    private static final int NBSTEP_MAX=9;
    private static final int NBING_MAX=15;
    private static final int NBCOM_MAX=20;
    private String TAG="DebugRecipe";
    private String DEFAULT_URL="https://www.familycookbook.com";

    public Recipe() {
        this(UUID.randomUUID());
    }

    public Recipe( UUID id){
        mId=id;
        mDate=new Date();
        mSteps = new String[NBSTEP_MAX];
        mIngredients= new String[NBING_MAX];
        for(int i=0;i<NBSTEP_MAX;i++){mSteps[i]="";}
        for(int i=0;i<NBING_MAX;i++){mIngredients[i]="";}
        mNbPers=4;
        mSeason=RecipeSeason.ALLYEAR;
        mDifficulty=RecipeDifficulty.UNDEFINED;
    }
}
```

```
mComments=new ArrayList<Comment>();
mNotes=new ArrayList<Note>();
try {mSource_url=new URL(DEFAULT_URL);
} catch (MalformedURLException e) {}
}

public UUID getId() {
    return mId;
}

public void setId(UUID id) {mId = id; }

public Date getDate() {
    return mDate;
}

public String getTitle() {
    return mTitle;
}

public void setDate(Date date) {
    mDate = date;
}

public void setTitle(String title) {
    mTitle = title;
}

public int getNbPers() {
    return mNbPers;
}

public void setNbPers(int nbPers) {
    mNbPers = nbPers;
}

// -----STEP-----
public void setStep(Integer i, String step) {
    if ((i > 0) && (i <= NBSTEP_MAX)) {
        mSteps[i - 1] = step;
    }
}

public String getStep(Integer i){
    if ((i>0)&&(i<=NBSTEP_MAX)) {return mSteps[i-1];}
    else{ return "";}
}

public int getNbStep(){
    int j=0;
    for(int i=NBSTEP_MAX; i>0; i--){
        if (!mSteps[i-1].isEmpty()) {j=i; break;}}
    return j;
}
```

```
}

public int getNbStepMax(){return NBSTEP_MAX;}

//-----Ingredients-----
public void setIngredient(Integer i, String ing) {
    if ((i > 0) && (i <= NBING_MAX)) {
        mIngredients[i-1] = ing;
    }
}

public String getIngredient(Integer i){
    if ((i>0)&&(i<=NBING_MAX)) {return mIngredients[i-1];}
    else{ return "";}
}

public int getNbIng(){
    int j=0;
    for(int i=NBING_MAX; i>0; i--){
        if (!mIngredients[i-1].isEmpty()) {j=i; break;}}
    return j;
}

public int getNbIngMax(){return NBING_MAX;}

//----- Owner -----
public User getOwner() {
    return mOwner;
}

public void setOwner(User owner) {
    mOwner = owner;
}

//----- Source -----
public String getSource() {
    return mSource;
}

public void setSource(String source) {
    mSource = source;
}

public URL getSource_url() {
    return mSource_url;
}

public String getSource_url_name() {
    String ret=mSource_url.toString();
    if (ret.equals("https:") || ret.equals("http:")){ret="";}
    return ret;
}

public void setSource_url(URL source_url) {
    mSource_url = source_url;
}
```

```
// -----Enum -----m

public RecipeSeason getSeason() {
    return mSeason;
}

public void setSeason(RecipeSeason season) {
    mSeason = season;
}

public boolean IsSummer(){
    switch (mSeason) {
        case SUMMER:
            return true;
        case ALLYEAR:
            return true;
        default:
            return false;
    }
}

public boolean IsWinter(){
    switch (mSeason) {
        case WINTER:
            return true;
        case ALLYEAR:
            return true;
        default:
            return false;
    }
}

public RecipeDifficulty getDifficulty() {
    return mDifficulty;
}

public void setDifficulty(RecipeDifficulty difficulty) {
    mDifficulty = difficulty;
}

//----- photo filename-----

public String getPhotoFilename(){
    return "IMG"+getId().toString()+".jpg";
}

//----- ArrayList Comments et Notes-----

public void addComment(Comment c){ mComments.add(c);}

public ArrayList<Comment> getComments() {return mComments;}

public Comment getComment(int i){
    if (i<mComments.size()){
```



```
        return mComments.get(i);
    } else {return null;}

}

public int getNbComMax(){return NBCOM_MAX;} // nb max affiché

public void addNote(Note note){ mNotes.add(note);}
public ArrayList<Note> getNotes() {return mNotes;}
public Note getNote(int i){
    if (i<mNotes.size()){ return mNotes.get(i);}
    else {return null;}
}

public double getNoteAvg() {
    mNoteAvg=0;
    if (mNotes.size()!=0) {
        for(Note n:mNotes){mNoteAvg+=n.getNote();}
        mNoteAvg=mNoteAvg/mNotes.size();
    } else {mNoteAvg=0;}
    return mNoteAvg;
}

//----- Serialisation -----
public String getSerializedComments(){
    Gson gson = new Gson();
    return gson.toJson(mComments);
}

public void getCommentsDeserialised(String raw){
    Gson gson=new Gson();
    Type listOfNotesObject = new TypeToken<ArrayList<Comment>>() {}.getType();
    mComments=gson.fromJson(raw, listOfNotesObject);
}

public String getSerializedOwner(){
    Gson gson = new Gson();
    return gson.toJson(mOwner);
}

public void getOwnerDeserialized(String raw){
    Gson gson=new Gson();
    mOwner=gson.fromJson(raw, User.class);
}

public String getSerializedNotes(){
    Gson gson = new Gson();
    return gson.toJson(mNotes);
}

public void getNotesDeserialised(String raw){
    Gson gson=new Gson();
    Type listOfNotesObject = new TypeToken<ArrayList<Note>>() {}.getType();
    mNotes=gson.fromJson(raw, listOfNotesObject);
}
```

}

```
package com.example.cookbook;

import android.content.Context;
import android.content.Intent;
import android.support.v4.app.Fragment;

import java.util.UUID;

public class RecipeActivity extends SingleFragmentActivity {
    private static final String EXTRA_RECIPE_ID="com.example.cookbook.recipe_id";

    public static Intent newIntent(Context packageContexte, UUID recipeId) {
        Intent intent=new Intent(packageContexte, RecipeActivity.class);
        intent.putExtra(EXTRA_RECIPE_ID, recipeId);
        return intent;
    }

    @Override
    protected Fragment createFragment(){
        UUID recipeId=(UUID) getIntent().getSerializableExtra(EXTRA_RECIPE_ID);
        return RecipeEditFragment.newInstance(recipeId);
    }
}
```

```
package com.example.cookbook;

import android.database.Cursor;
import android.database.CursorWrapper;
import android.util.Log;

import java.net.MalformedURLException;
import java.net.URL;
import java.util.Date;
import java.util.UUID;

import static com.example.cookbook.RecipeDbSchema.*;

public class RecipeCursorWrapper extends CursorWrapper {
    public RecipeCursorWrapper(Cursor cursor){
        super(cursor);
    }
    private static final String TAG = "RecipeCursorWrapper";

    public Recipe getRecipe(){
        String uuidString = getString(getColumnIndex(RecipeTable.Cols.UUID));
        Recipe r=new Recipe(UUID.fromString(uuidString));
        String ownerString = getString(getColumnIndex(RecipeTable.Cols.OWNER));
        r.getOwnerDeserialized(ownerString);
        String title = getString(getColumnIndex(RecipeTable.Cols.TITLE));
        r.setTitle(title);
        String source = getString(getColumnIndex(RecipeTable.Cols.SOURCE));
        r.setSource(source);
        String sourceURL=getString(getColumnIndex(RecipeTable.Cols.SOURCE_URL));
        try {
            URL url = new URL(sourceURL);
            r.setSource_url(url);}
        catch (MalformedURLException e) {
            Log.d(TAG, "getURL from cursor failed");
        }
        long date = getLong(getColumnIndex(RecipeTable.Cols.DATE));
        r.setDate(new Date(date));
        int nbpers=getInt(getColumnIndex(RecipeTable.Cols.NBPERS));
        r.setNbPers(nbpers);
        String[] step= new String[r.getNbStepMax()];
        for(int i=0;i<r.getNbStepMax();i++){
            step[i]=getString(getColumnIndex(RecipeTable.Cols.STEP[i]));
            r.setStep(i+1, step[i]);
        }
        String[] ing=new String[r.getNbIngMax()];
        for(int i=0;i<r.getNbIngMax(); i++){
            ing[i]=getString(getColumnIndex(RecipeTable.Cols.ING[i]));
            r.setIngredient(i+1, ing[i]);
        }
        String season = getString(getColumnIndex(RecipeTable.Cols.SEASON));
        r.setSeason(RecipeSeason.valueOf(season));
        String difficulty = getString(getColumnIndex(RecipeTable.Cols.DIFFICULTY));
        r.setDifficulty(RecipeDifficulty.valueOf(difficulty));
        String comments = getString(getColumnIndex(RecipeTable.Cols.COMMENTS));
```

```
    r.getCommentsDeserialised(comments);  
    String notes = getString(getColumnIndex(RecipeTable.Cols.NOTES));  
    r.getNotesDeserialised(notes);  
    return r;  
  }  
}
```

```
package com.example.cookbook;

public class RecipeDbSchema {
    public static final class RecipeTable{
        public static final String NAME="recipes";
        public static final class Cols {
            public static final String UUID="uuid";
            public static final String OWNER="owner";
            public static final String TITLE="title";
            public static final String SOURCE="source";
            public static final String SOURCE_URL="source_url";
            public static final String DATE="date";
            public static final String NBPERS="nbpers";
            public static final String[] STEP={"etape1","etape2","etape3","etape4",
                "etape5","etape6","etape7","etape8","etape9"};
            public static final String[] ING={"ing01","ing02","ing03","ing04",
                "ing05","ing06","ing0e7","ing08","ing09","ing10","ing11",
                "ing12","ing13","ing14","ing15"};
            public static final String SEASON="season";
            public static final String DIFFICULTY="difficulty";
            public static final String COMMENTS="comments";
            public static final String NOTES="notes";
        }
    }
}
```

```
package com.example.cookbook;

public enum RecipeDifficulty {
    QUICK, EASY, ELABORATE, SOPHISTICATED, UNDEFINED;
    private static RecipeDifficulty[] list=RecipeDifficulty.values();

    public static RecipeDifficulty getDifficulty(int i){
        return list[i];
    }

    public static int getIndex(RecipeDifficulty r){
        for(int i=0;i<list.length;i++) {
            if (list[i].equals(r)){ return i;}
        }
        return 0;
    }
}
```

```
package com.example.cookbook;

import android.content.Context;
import android.content.Intent;
import android.support.v4.app.Fragment;

import java.util.UUID;

public class RecipeDisplayActivity extends SingleFragmentActivity {
    private static final String EXTRA_RECIPE_ID="com.example.cookbook.recipe_id";

    public static Intent newIntent(Context packageContexte, UUID recipeId) {
        Intent intent=new Intent(packageContexte, RecipeDisplayActivity.class);
        intent.putExtra(EXTRA_RECIPE_ID, recipeId);
        return intent;
    }

    @Override
    protected Fragment createFragment() {
        UUID recipeId=(UUID) getIntent().getSerializableExtra(EXTRA_RECIPE_ID);
        return RecipeDisplayFragment.newInstance(recipeId);
    }
}
```



```
package com.example.cookbook;

import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.provider.MediaStore;
import android.support.v4.app.Fragment;
import android.text.Editable;
import android.text.TextWatcher;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.RatingBar;
import android.widget.ScrollView;
import android.widget.Spinner;
import android.widget.TextView;

import java.io.File;
import java.text.DecimalFormat;
import java.util.UUID;

public class RecipeDisplayFragment extends Fragment {
    private static final String ARG_RECIPE_ID="recipe_id";
    private static final String TAG = "DebugRecipeDisplayFrag";
    private static final int REQUEST_PHOTO= 2;
    private Recipe mRecipe;
    private File mPhotoFile;
    private int mStepNb;
    private int mIngNb;
    private String newcomment;
    private String DEFAULT_URL="https://www.familycookbook.com";

    private SessionInfo mSession;
    private ImageView mDPhotoView;
    private TextView mDTitleText;
    private RatingBar mDRatingBar;
    private TextView mDRatingBarText;
    private TextView mDAuthorText;
    private TextView mDDifficulty;
    private ImageView mSunIcon;
    private ImageView mIceIcon;
    private TextView mDSourceText;
    private TextView mDSourceUrl;
    private TextView mDIngTitle;
    private TextView mDComTitle;
    private TextView[] mDStepText;
    private TextView[] mDIngText;
```

```

private TextView[] mDComText;
private EditText mDNexComment;
private ImageView mDEnterComment;
private ScrollView mScroll;

public static RecipeDisplayFragment newInstance(UUID recipeId) {
    Bundle args=new Bundle();
    args.putSerializable(ARG_RECIPE_ID, recipeId);
    RecipeDisplayFragment fragment=new RecipeDisplayFragment();
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    mRecipe=new Recipe();
    UUID recipeId=(UUID) getArguments().getSerializable(ARG_RECIPE_ID);
    mRecipe=CookBook.get(getActivity()).getRecipe(recipeId);
    mPhotoFile=CookBook.get(getActivity()).getPhotoFile(mRecipe);
    mSession= SessionInfo.get(getActivity());
    setHasOptionsMenu(true);
    mStepNb=mRecipe.getNbStep();
    mIngNb=mRecipe.getNbIng();
}

@Override
public void onPause(){
    super.onPause();
    CookBook.get(getActivity()).updateRecipe(mRecipe);
}

@Override
public void onCreateOptionsMenu(Menu menu, MenuInflater inflater) {
    super.onCreateOptionsMenu(menu, inflater);
    inflater.inflate(R.menu.fragment_recipe_display, menu);
    MenuItem menuItem = menu.findItem(R.id.recipe_menu_edit);
    if(!mRecipe.getOwner().getId().equals(mSession.getUser().getId())){
        //getActivity().invalidateOptionsMenu();
        menuItem.setVisible(false);
    }
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.recipe_menu_report:
            Intent i=new Intent(Intent.ACTION_SEND);
            i.setType("text/plain");
            i.putExtra(Intent.EXTRA_TEXT, getRecipeReport());

i.putExtra(Intent.EXTRA_SUBJECT,getString(R.string.recipe_report_subject));
            startActivity(i);
            return true;
        case R.id.recipe_menu_edit:

```

```

        Intent intent=
RecipeActivity.newIntent(getActivity(),mRecipe.getId());
        startActivity(intent);
        return true;
    case R.id.recipe_menu_delete:
        if(!CookBook.get(getActivity()).deleteImage(mRecipe)) {
            Log.d(TAG, "DeleteRecipeFromMenu : no delete image");
        }
        CookBook.get(getActivity()).removeRecipe(mRecipe);
        getActivity().onBackPressed();
    default:
        return super.onOptionsItemSelected(item);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState){
    final View v=inflater.inflate(R.layout.fragment_display_recipe, container,
false);
    mScroll=(ScrollView) v.findViewById(R.id.fragment_recipe_scroll);
    mDTitleText =(TextView) v.findViewById(R.id.recipe_display_title);
    mDPhotoView=(ImageView) v.findViewById(R.id.recipe_display_photo);
    updatePhotoView();
    mDRatingBar=(RatingBar) v.findViewById(R.id.recipe_display_ratingBar);
    mDRatingBarText=(TextView) v.findViewById(R.id.recipe_display_ratingBar_txt);
    mDDifficulty=(TextView) v.findViewById(R.id.recipe_display_difficulty);
    mSunIcon=(ImageView) v.findViewById(R.id.recipe_img_sun) ;
    mIceIcon=(ImageView) v.findViewById(R.id.recipe_img_ice) ;
    mDAuthorText=(TextView) v.findViewById(R.id.recipe_display_author);
    mDSourceText=(TextView) v.findViewById(R.id.recipe_display_source);
    mDSourceUrl=(TextView) v.findViewById(R.id.recipe_display_source_url);
    mDIngTitle=(TextView) v.findViewById(R.id.recipe_display_title_ing);
    mDComTitle=(TextView)v.findViewById(R.id.recipe_display_comment_title);
    final int[] rIngID=
{R.id.recipe_display_I01,R.id.recipe_display_I02,R.id.recipe_display_I03,
R.id.recipe_display_I04,R.id.recipe_display_I05,R.id.recipe_display_I06,
R.id.recipe_display_I07,R.id.recipe_display_I08,R.id.recipe_display_I09,
R.id.recipe_display_I10,R.id.recipe_display_I11,R.id.recipe_display_I12,
R.id.recipe_display_I13,R.id.recipe_display_I14,R.id.recipe_display_I15};
    mDIngText= new TextView[mRecipe.getNbIngMax()];
    for(int i=0;i<mRecipe.getNbIngMax();i++) {
        mDIngText[i] = (TextView) v.findViewById(rIngID[i]);
    }
    final int[] rID=
{R.id.recipe_display_S1,R.id.recipe_display_S2,R.id.recipe_display_S3,
R.id.recipe_display_S4,R.id.recipe_display_S5,R.id.recipe_display_S6,
R.id.recipe_display_S7,R.id.recipe_display_S8,R.id.recipe_display_S9};
    mDStepText=new TextView[mRecipe.getNbStepMax()];
    for(int i=0;i<mRecipe.getNbStepMax();i++) {

```

```

        mDStepText[i] = (TextView) v.findViewById(rID[i]);
    }
    final int[] rComID=
    {R.id.recipe_display_C01,R.id.recipe_display_C02,R.id.recipe_display_C03,
    R.id.recipe_display_C04,R.id.recipe_display_C05,R.id.recipe_display_C06,
    R.id.recipe_display_C07,R.id.recipe_display_C08,R.id.recipe_display_C09,
    R.id.recipe_display_C10,R.id.recipe_display_C11,R.id.recipe_display_C12,
    R.id.recipe_display_C13,R.id.recipe_display_C14,R.id.recipe_display_C15,
    R.id.recipe_display_C16,R.id.recipe_display_C17,R.id.recipe_display_C18,
    R.id.recipe_display_C19,R.id.recipe_display_C20};
    mDComText=new TextView[mRecipe.getNbComMax()];
    for(int i=0;i<mRecipe.getNbComMax();i++) {
        mDComText[i] = (TextView) v.findViewById(rComID[i]);
    }
    updateUI();
    mDNexComment=(EditText) v.findViewById(R.id.recipe_display_enter_comment);
    mDNexComment.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int
after) {

        }

        @Override
        public void onTextChanged(CharSequence s, int start, int before, int count) {
            newcomment=s.toString();
        }

        @Override
        public void afterTextChanged(Editable s) {

        }
    });
    mDEnterComment=(ImageView) v.findViewById(R.id.recipe_img_enter);
    mDEnterComment.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mRecipe.addComment(new Comment(newcomment, mSession.getUser()));
            mDNexComment.setText("...");
            updateUI();
        }
    });
    return v;
}
private void updatePhotoView(){
    if (mPhotoFile==null || !mPhotoFile.exists()){
        mDPhotoView.setImageDrawable(getResources().getDrawable(R.drawable.ic_recipe_camera));
    }
}

```

```

    } else {
        Bitmap bitmap=PictureUtils.getScaledBitmap(mPhotoFile.getPath(),
getActivity());
        mDPhotoView.setImageBitmap(bitmap);
    }
}
private void updateUI(){
    String s;
    User u=mRecipe.getOwner();
    int ingMax=mRecipe.getNbIngMax();
    int stepMax=mRecipe.getNbStepMax();
    int comMax=mRecipe.getNbComMax();
    int NbCom=mRecipe.getComments().size();
    int gone=View.GONE;
    int visible=View.VISIBLE;
    DecimalFormat df = new DecimalFormat("#.##");
    mDTitleText.setText(mRecipe.getTitle());
    mDRatingBar.setRating((float) mRecipe.getNoteAvg());
    mDRatingBarText.setText(df.format(mRecipe.getNoteAvg())+" (" +
mRecipe.getNotes().size()+")");
    s=getString(R.string.recipe_display_author,u.getName(),u.getFamily());
    int idiff= RecipeDifficulty.getIndex(mRecipe.getDifficulty());
    String[] stringArrayDiff =
getResources().getStringArray(R.array.recipe_difficulty_array);
    mDDifficulty.setText(stringArrayDiff[idiff]);
    mSunIcon.setImageResource((mRecipe.IsSummer()) ? R.drawable.ic_recipe_sun :
R.drawable.ic_recipe_sun_disabled);
    mIceIcon.setImageResource((mRecipe.IsWinter()) ? R.drawable.ic_recipe_ice :
R.drawable.ic_recipe_ice_disabled);
    mDAuthorText.setText(s);
    mDSourceText.setText(mRecipe.getSource());
    if (mRecipe.getSource_url_name().equals(DEFAULT_URL)) {
        mDSourceUrl.setText("");
    }
    else {mDSourceUrl.setText(mRecipe.getSource_url_name());}
    s=getString(R.string.recipe_display_title_ing, ""+mRecipe.getNbPers());
    mDIngTitle.setText(s);
    for(int i=0;i<ingMax;i++){
        if (mIngNb>0){mDIngText[i].setText("- "+mRecipe.getIngredient(i+1));}
        if (i>=0){mDIngText[i].setVisibility((mIngNb>i)? visible:gone);}
    }
    for(int i=0;i<stepMax;i++){
        if (mStepNb>0){mDStepText[i].setText((i+1)+". "+mRecipe.getStep(i+1));}
        if (i>=0){mDStepText[i].setVisibility((mStepNb>i)? visible:gone);}
    }
    s=getString(R.string.recipe_display_title_comment,
""+mRecipe.getComments().size());
    mDComTitle.setText(s);
    for(int i=0;i<comMax;i++){
        if (NbCom>0){
            if (i<NbCom) {mDComText[i].setText("- "+mRecipe.getComment(NbCom-
i-1).toString());}
            else {mDComText[i].setText("");}
        }
        if (i>=0){mDComText[i].setVisibility((NbCom>i)? visible:gone);}
    }
}

```

```
    }  
}  
private String getRecipeReport(){  
    String report;  
    Integer iplus;  
    report =getString(R.string.recipe_report_title, mRecipe.getTitle()+"\n";  
    report  
+=getString(R.string.recipe_report_owner,mRecipe.getOwner().getNameComplete()+"\n";  
    //String dateFormat = "dd MMM yyyy";  
    //String dateString=DateFormat.format(dateFormat,mRecipe.getDate()).toString();  
    if(!mRecipe.getSource_url_name().equals("")){  
        report += getString(R.string.recipe_report_url,  
mRecipe.getSource_url_name()+"\n";  
    }  
    for(int i=0;i<mRecipe.getNbIng();i++){  
        report += getString(R.string.recipe_report_ing, mRecipe.getIngredient(i  
+1))+"\n";  
    }  
    for(int i=0;i<mRecipe.getNbStep();i++){  
        iplus=i+1;  
        report += getString(R.string.recipe_report_step, iplus+"",  
            mRecipe.getStep(i+1))+"\n";  
    }  
    report +=getString(R.string.recipe_report_final);  
    return report;  
}  
}
```

```
package com.example.cookbook;

import android.app.Activity;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.content.pm.ResolveInfo;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.support.v4.app.Fragment;
import android.support.v4.content.FileProvider;
import android.text.Editable;
import android.text.TextWatcher;
import android.text.format.DateFormat;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.ScrollView;
import android.widget.Spinner;
import android.widget.TextView;

import java.io.File;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.Date;
import java.util.List;
import java.util.UUID;

public class RecipeEditFragment extends Fragment {
    private static final String ARG_RECIPE_ID="recipe_id";
    private static final String DIALOG_DATE="DialogDate";
    private static final int REQUEST_DATE=0;
    private static final int REQUEST_PHOTO= 2;
    private static final String TAG = "DebugRecipeEditFragment";
    //Log.d(TAG, "onTextChanged de mSource_url input>" + s.toString()+"<");
    private Recipe mRecipe;
    private File mPhotoFile;
    private EditText mTitleField;
    private EditText mSourceField;
    private EditText mSourceUrl;
    private EditText mNbPersField;
    // private Button mDateButton;
    private TextView[] mSTextView;
    private EditText mNewStepField;
```

```

private ImageButton mNewStepEnter;
private ImageButton mNewStepBack;
private int mStepNb;
private TextView[] mSTextIngView;
private EditText mNewIngField;
private ImageButton mNewIngEnter;
private ImageButton mNewIngBack;
private int mIngNb;
private Spinner mSeasonSpinner;
private Spinner mDifficultySpinner;
private ScrollView mScroll;
private ImageView mPhotoView;

public static RecipeEditFragment newInstance(UUID recipeId){
    Bundle args=new Bundle();
    args.putSerializable(ARG_RECIPE_ID, recipeId);
    RecipeEditFragment fragment=new RecipeEditFragment();
    fragment.setArguments(args);
    return fragment;
}

@Override
public void onCreate(Bundle savedInstanceState){
    super.onCreate(savedInstanceState);
    mRecipe=new Recipe();
    UUID recipeId=(UUID) getArguments().getSerializable(ARG_RECIPE_ID);
    mRecipe=CookBook.get(getActivity()).getRecipe(recipeId);
    setHasOptionsMenu(true);
    mPhotoFile=CookBook.get(getActivity()).getPhotoFile(mRecipe);
    mStepNb=mRecipe.getNbStep();
    mIngNb=mRecipe.getNbIng();
}

@Override
public void onPause(){
    super.onPause();
    CookBook.get(getActivity()).updateRecipe(mRecipe);
}

@Override
public void onCreateOptionsMenu(Menu menu, MenuInflater inflater) {
    super.onCreateOptionsMenu(menu, inflater);
    inflater.inflate(R.menu.fragment_recipe, menu);
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.recipe_menu_report:
            Intent i=new Intent(Intent.ACTION_SEND);
            i.setType("text/plain");
            i.putExtra(Intent.EXTRA_TEXT, getRecipeReport());
            i.putExtra(Intent.EXTRA_SUBJECT,getString(R.string.recipe_report_subject));

```



```

        startActivity(i);
        return true;
    case R.id.recipe_menu_delete:
        if(!CookBook.get(getActivity()).deleteImage(mRecipe)) {
            Log.d(TAG, "DeleteRecipeFromMenu : cannot delete image");
        }
        CookBook.get(getActivity()).removeRecipe(mRecipe);
        getActivity().onBackPressed();
    default:
        return super.onOptionsItemSelected(item);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState){
    final View v=inflater.inflate(R.layout.fragment_recipe_edit, container, false);
    mScroll=(ScrollView) v.findViewById(R.id.fragment_recipe_scroll);
    PackageManager packageManager=getActivity().getPackageManager();
    final Intent captureImage=new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    boolean canTakePhoto=(mPhotoFile != null) &&
        (captureImage.resolveActivity(packageManager)!=null);
    mPhotoView=(ImageView) v.findViewById(R.id.recipe_photo);
    mPhotoView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Uri uri= FileProvider.getUriForFile(getActivity(),
                "com.example.cookbook.fileprovider", mPhotoFile);
            captureImage.putExtra(MediaStore.EXTRA_OUTPUT, uri);
            List<ResolveInfo> cameraActivities=getActivity()
                .getPackageManager().queryIntentActivities(captureImage,
                    PackageManager.MATCH_DEFAULT_ONLY);
            for(ResolveInfo activity : cameraActivities){
                getActivity().grantUriPermission(activity.activityInfo.packageName,
                    uri, Intent.FLAG_GRANT_WRITE_URI_PERMISSION);
            }
            startActivityForResult(captureImage, REQUEST_PHOTO);
        }
    });
    updatePhotoView();
    mTitleField= (EditText) v.findViewById(R.id.recipe_title);
    mTitleField.setText(mRecipe.getTitle());
    mTitleField.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int
after) {

            // blank
        }

        @Override
        public void onTextChanged(CharSequence s, int start, int before, int count) {
            mRecipe.setTitle(s.toString());
        }
    }

```

```

        @Override
        public void afterTextChanged(Editable s) {
            //blank
        }
    });

    mSourceField= (EditText) v.findViewById(R.id.recipe_source);
    mSourceField.setText(mRecipe.getSource());
    mSourceField.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int
after) {

            // blank
        }

        @Override
        public void onTextChanged(CharSequence s, int start, int before, int count)
{
            mRecipe.setSource(s.toString());
        }

        @Override
        public void afterTextChanged(Editable s) {

        }
    });

    mSourceUrl=(EditText) v.findViewById(R.id.recipe_source_url);
    mSourceUrl.setText(mRecipe.getSource_url().toString());
    mSourceUrl.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int
after) {

        }

        @Override
        public void onTextChanged(CharSequence s, int start, int before, int count)
{
            if (s.toString().equals("")) {} else {
                try {
                    URL url=new URL(s.toString());
                    mRecipe.setSource_url(url);
                } catch (MalformedURLException e) {
                    Log.d(TAG, "onTextChanged de mSource_url >" +s.toString()+"<
Failed");
                }
            }
        }

        @Override
        public void afterTextChanged(Editable s) {

```

```

    });

    mNbPersField= (EditText) v.findViewById(R.id.recipe_nbpers);
    mNbPersField.setText(mRecipe.getNbPers()+"");
    mNbPersField.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int
after) {

        }

        @Override
        public void onTextChanged(CharSequence s, int start, int before, int count)
{
            if (s.toString().equals("")) {} else {
                int nb_entered = Integer.parseInt(s.toString());
                if ((nb_entered > 0) && (nb_entered < 13)) {
                    mRecipe.setNbPers(nb_entered);
                }
            }
        }

        @Override
        public void afterTextChanged(Editable s) {

        }
    });

    mSeasonSpinner= (Spinner) v.findViewById(R.id.recipe_season);
    ArrayAdapter<CharSequence> adapterSeason =
ArrayAdapter.createFromResource(getContext(),
        R.array.recipe_season_array, android.R.layout.simple_spinner_item);
    mSeasonSpinner.setAdapter(adapterSeason);
    mSeasonSpinner.setSelection(RecipeSeason.getIndex(mRecipe.getSeason()));
    mSeasonSpinner.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {
            mRecipe.setSeason(RecipeSeason.getSeason(position));
        }

        @Override
        public void onNothingSelected(AdapterView<?> parent) {

        }
    });

    mDifficultySpinner= (Spinner) v.findViewById(R.id.recipe_difficulty);
    ArrayAdapter<CharSequence> adapterDifficulty =
ArrayAdapter.createFromResource(getContext(),

```

```

        R.array.recipe_difficulty_array, android.R.layout.simple_spinner_item);
mDifficultySpinner.setAdapter(adapterDifficulty);

mDifficultySpinner.setSelection(RecipeDifficulty.getIndex(mRecipe.getDifficulty()));
mDifficultySpinner.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {
        mRecipe.setDifficulty(RecipeDifficulty.getDifficulty(position));
    }

    @Override
    public void onNothingSelected(AdapterView<?> parent) {

    }
});

final int[] rID= {R.id.recipe_S1,R.id.recipe_S2,R.id.recipe_S3,R.id.recipe_S4,
R.id.recipe_S5,R.id.recipe_S6,R.id.recipe_S7,R.id.recipe_S8,R.id.recipe_S9};
mTextView= new TextView[mRecipe.getNbStepMax()];
for(int i=0;i<mRecipe.getNbStepMax();i++) {
    mTextView[i] = (TextView) v.findViewById(rID[i]);
}
mNewStepEnter=(ImageButton) v.findViewById(R.id.recipe_step_enter);
mNewStepBack=(ImageButton) v.findViewById(R.id.recipe_step_back);
updateListStep(v);

mNewStepField= (EditText) v.findViewById(R.id.recipe_new_step);
mNewStepField.addTextChangedListener(new TextWatcher() {
    @Override
    public void beforeTextChanged(CharSequence s, int start, int count, int
after) {

        // blank
    }

    @Override
    public void onTextChanged(CharSequence s, int start, int before, int count)
{
        mRecipe.setStep(mStepNb+1, s.toString());
    }
    @Override
    public void afterTextChanged(Editable s) {

    }
});

mNewStepEnter.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // Toast.makeText(getActivity(), mRecipe.getStep(mStepNb+1)+"/"+mStepNb,
        // Toast.LENGTH_SHORT).show();
        mNewStepField.setVisibility(updateListStep(v));
    }
});

```

```

        mNewStepField.getText().clear();
    }
});

mNewStepBack.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mRecipe.setStep(mStepNb, "");
        mRecipe.setStep(mStepNb+1, "");
        mNewStepField.setVisibility(updateListStep(v));
        mNewStepField.getText().clear();
    }
});

final int[] rIngID=
{R.id.recipe_I01,R.id.recipe_I02,R.id.recipe_I03,R.id.recipe_I04,
  R.id.recipe_I05,R.id.recipe_I06,R.id.recipe_I07,R.id.recipe_I08,
  R.id.recipe_I09,R.id.recipe_I10,R.id.recipe_I11,R.id.recipe_I12,
  R.id.recipe_I13,R.id.recipe_I14,R.id.recipe_I15};
mTextIngView= new TextView[mRecipe.getNbIngMax()];
for(int i=0;i<mRecipe.getNbIngMax();i++) {
    mTextIngView[i] = (TextView) v.findViewById(rIngID[i]);
}
mNewIngEnter=(ImageButton) v.findViewById(R.id.recipe_ing_enter);
mNewIngBack=(ImageButton) v.findViewById(R.id.recipe_ing_back);
updateListIng(v);

mNewIngField= (EditText) v.findViewById(R.id.recipe_new_ing);
mNewIngField.addTextChangedListener(new TextWatcher() {
    @Override
    public void beforeTextChanged(CharSequence s, int start, int count, int
after) {

    }

    @Override
    public void onTextChanged(CharSequence s, int start, int before, int count) {
        mRecipe.setIngredient(mIngNb+1, s.toString());
    }

    @Override
    public void afterTextChanged(Editable s) {

    }
});
mNewIngEnter.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mNewIngField.setVisibility(updateListIng(v));
        mNewIngField.getText().clear();
    }
});
mNewIngBack.setOnClickListener(new View.OnClickListener() {

```

```

        @Override
        public void onClick(View v) {
            mRecipe.setIngredient(mIngNb, "");
            mRecipe.setIngredient(mIngNb+1, "");
            mNewIngField.setVisibility(updateListIng(v));
            mNewIngField.getText().clear();
        }
    });

    return v;
}

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (resultCode != Activity.RESULT_OK) {
        return;
    }

    if (requestCode==REQUEST_PHOTO){
        Uri uri=FileProvider.getUriForFile(getActivity(),
            "com.example.cookbook.fileprovider", mPhotoFile);
        getActivity().revokeUriPermission(uri,
Intent.FLAG_GRANT_WRITE_URI_PERMISSION);
        updatePhotoView();
    }
}

private String getRecipeReport(){
    String report="";
    Integer iplus=0;
    report +=getString(R.string.recipe_report_title, mRecipe.getTitle())+"\n";
    report
+=getString(R.string.recipe_report_owner,mRecipe.getOwner().getNameComplete())+"\n";

    if(!mRecipe.getSource_url_name().equals("")){
        report += getString(R.string.recipe_report_url,
mRecipe.getSource_url_name())+"\n";
    }
    for(int i=0;i<mRecipe.getNbIng();i++){
        report += getString(R.string.recipe_report_ing, mRecipe.getIngredient(i
+1))+"\n";
    }
    for(int i=0;i<mRecipe.getNbStep();i++){
        iplus=i+1;
        report += getString(R.string.recipe_report_step, iplus+"",
            mRecipe.getStep(i+1))+"\n";
    }
    report +=getString(R.string.recipe_report_final);
    return report;
}

private void updatePhotoView(){
    if (mPhotoFile==null || !mPhotoFile.exists()){

```

```

mPhotoView.setImageDrawable(getResources().getDrawable(R.drawable.ic_recipe_camera));
    } else {
        Bitmap bitmap=PictureUtils.getScaledBitmap(mPhotoFile.getPath(),
getActivity());
        mPhotoView.setImageBitmap(bitmap);
    }
}

private int updateListStep(View v) {
    int imax=mRecipe.getNbStepMax(), iplus;
    String[] display=new String[imax];
    mStepNb = mRecipe.getNbStep();
    int gone=View.GONE;
    int visible=View.VISIBLE;
    mSTextView[0].setText("1...");
    mSTextView[0].setVisibility(visible);
    for(int i=0;i<imax;i++){
        iplus=i+1; display[i]=iplus+". "+mRecipe.getStep(i+1);
        if (mStepNb>0){mSTextView[i].setText(display[i]);}
        if (i>=0){mSTextView[i].setVisibility((mStepNb>i)? visible:gone);}
    }
    mNewStepBack.setVisibility((mStepNb==0)? gone:visible);
    mNewStepEnter.setVisibility((mStepNb==imax)? gone:visible);
    return ((mStepNb==imax)?gone:visible);
}

private int updateListIng(View v) {
    int imax=mRecipe.getNbIngMax(), iplus;
    String[] display=new String[imax];
    mIngNb = mRecipe.getNbIng();
    int gone=View.GONE;
    int visible=View.VISIBLE;
    mSTextIngView[0].setText("-...");
    mSTextIngView[0].setVisibility(visible);
    for(int i=0;i<imax;i++){
        iplus=i+1; display[i]="-"+mRecipe.getIngredient(i+1);
        if (mIngNb>0){mSTextIngView[i].setText(display[i]);}
        if (i>=0){mSTextIngView[i].setVisibility((mIngNb>i)? visible:gone);}
    }
    mNewIngBack.setVisibility((mIngNb==0)? gone:visible);
    mNewIngEnter.setVisibility((mIngNb==imax)? gone:visible);
    return ((mIngNb==imax)?gone:visible);
}
}

```

```
package com.example.cookbook;

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import com.example.cookbook.RecipeDbSchema.RecipeTable;

public class RecipeBaseHelper extends SQLiteOpenHelper {
    private static final int VERSION=1;
    private static final String DATA_BASE_NAME="recipeBase.db"; // REINIT BASE
    public RecipeBaseHelper(Context context) {
        super(context, DATA_BASE_NAME, null, VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String st="",si="";
        Recipe r=new Recipe();
        for(int i=0;i<r.getNbStepMax();i++){
            st=st+RecipeTable.Cols.STEP[i];
            st=st+", ";
        }
        for(int i=0;i<r.getNbIngMax();i++){
            si=si+RecipeTable.Cols.ING[i];
            si=si+", ";
        }
        db.execSQL("create table "+ RecipeTable.NAME+"("
            + "_id integer primary key autoincrement, "+
            RecipeTable.Cols.UUID+", "+
            RecipeTable.Cols.OWNER+", "+
            RecipeTable.Cols.TITLE+", "+
            RecipeTable.Cols.SOURCE+", "+
            RecipeTable.Cols.SOURCE_URL+", "+
            RecipeTable.Cols.DATE+", "+
            RecipeTable.Cols.NBPERS+", "+
            st +
            si +
            RecipeTable.Cols.SEASON+", "+
            RecipeTable.Cols.DIFFICULTY+", "+
            RecipeTable.Cols.COMMENTS+", "+
            RecipeTable.Cols.NOTES+
            ")"
        );
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    }
}
```



```
package com.example.cookbook;

import android.support.v4.app.Fragment;

public class RecipeListActivity extends SingleFragmentActivity {
    @Override
    protected Fragment createFragment(){
        return new RecipeListFragment();
    }
}
```

```
package com.example.cookbook;

import android.app.Activity;
import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.RatingBar;
import android.widget.TextView;
import android.widget.Toast;
import android.widget.Toolbar;

import java.io.File;
import java.text.DecimalFormat;
import java.util.Collections;
import java.util.List;
//-----
// print 24/5/2021
// Saved GitHub V01
//-----

public class RecipeListFragment extends Fragment {
    private RecyclerView mRecipeRecyclerView;
    private RecipeAdapter mAdapter;
    private SessionInfo mSession;
    private static final String SAVED_SORT_STATUS="sort";
    private int mSortOption;
    private int finalRate;
    private static final int maskSortTitle=1 <<3;
    private static final int maskSortSource=1;
    private static final int maskSortNote=1 <<2;
    private static final int maskSortSeason=1 <<4;
    private static final int maskSortDifficulty=1 <<5;
    private static final String TAG = "DebugRecipeListFragment";
    //Log.d(TAG, "OnCreateView coming back mSortOption=" + mSortOption);
```

```

private static final String DIALOG_RATE = "DialogRate";
private static final int REQUEST_RATE = 0;

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setHasOptionsMenu(true);
    mSession= SessionInfo.get(getActivity());
    mSortOption=0;
    AppCompatActivity activity = (AppCompatActivity) getActivity();
    User u=mSession.getUser();

    activity.getSupportActionBar().setSubtitle(getString(R.string.recipe_display_author,u.g
etName(),u.getFamily()));
}

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (resultCode != Activity.RESULT_OK) {
        return;
    }
    if (requestCode == REQUEST_RATE) {
        Integer rate = (Integer) data
            .getSerializableExtra(RatePickerFragment.EXTRA_RATE);
        updateUI();
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fragment_recipe_list, container, false);
    mRecipeRecyclerView = (RecyclerView)
view.findViewById(R.id.recipe_recycler_view);
    mRecipeRecyclerView.setLayoutManager(new LinearLayoutManager(getActivity()));
    if (savedInstanceState!=null){
        mSortOption=savedInstanceState.getInt(SAVED_SORT_STATUS);
    }
    updateUI();
    return view;
}

@Override
public void onResume(){
    super.onResume();
    updateUI();
}

@Override
public void onSaveInstanceState(Bundle outState) {
    super.onSaveInstanceState(outState);
    outState.putInt(SAVED_SORT_STATUS, mSortOption);
}

```

```

@Override
public void onCreateOptionsMenu(Menu menu, MenuInflater inflater){
    super.onCreateOptionsMenu(menu, inflater);
    inflater.inflate(R.menu.fragment_recipe_list, menu);
}

@Override
public boolean onOptionsItemSelected(MenuItem item){
    switch (item.getItemId()){
        case R.id.new_recipe:
            Recipe recipe=new Recipe();
            SessionInfo session=SessionInfo.get(getActivity());
            recipe.setOwner(session.getUser());
            Cookbook.get(getActivity()).addRecipe(recipe);
            Intent intent= RecipeActivity.newIntent(getActivity(),recipe.getId());
            startActivity(intent);
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}

private void updateUI() {
    Cookbook cookbook=CookBook.get(getActivity());
    List<Recipe> recipes=cookbook.getRecipes();
    if ((mSortOption & maskSortTitle) == maskSortTitle) {
        Collections.sort(recipes,
            (r1, r2)->{return(r1.getTitle().compareTo(r2.getTitle()));});
    }
    if ((mSortOption & maskSortNote) == maskSortNote) {
        Collections.sort(recipes,
            (r1, r2)->{return((int)(r2.getNoteAvg()-r1.getNoteAvg()));});
    }
    if ((mSortOption & maskSortSource) == maskSortSource) {
        Collections.sort(recipes,
            (r1, r2)-
>{return(r1.getOwner().getName().compareTo(r2.getOwner().getName()));});
    }
    if ((mSortOption & maskSortSeason) == maskSortSeason) {
        Collections.sort(recipes,
            (r1, r2)->{return(RecipeSeason.getIndex(r1.getSeason())
                -RecipeSeason.getIndex(r2.getSeason()));});
    }
    if ((mSortOption & maskSortDifficulty) == maskSortDifficulty) {
        Collections.sort(recipes,
            (r1, r2)->{return(RecipeDifficulty.getIndex(r1.getDifficulty())
                -RecipeDifficulty.getIndex(r2.getDifficulty()));});
    }
    if (mAdapter==null){
        mAdapter=new RecipeAdapter(recipes);
        mRecipeRecyclerView.setAdapter(mAdapter);
    } else {
        mAdapter.setRecipes(recipes);
        mAdapter.notifyDataSetChanged();
    }
}

private class RecipeHolder extends RecyclerView.ViewHolder

```

```

        implements View.OnClickListener {
private TextView mTitleTextView;
private TextView mSourceTextView;
private TextView mNoteTextView;
private TextView mDifficulty;
private ImageView mEditIcon;
private ImageView mPhotoView;
private ImageView mSunIcon;
private ImageView mIceIcon;
private File mPhotoFile;
private Recipe mRecipe;
private RatingBar mRating;
public RecipeHolder(LayoutInflater inflater, ViewGroup parent){
    super(inflater.inflate(R.layout.list_item_recipe, parent, false));
    itemView.setOnClickListener(this);
    mTitleTextView= (TextView) itemView.findViewById(R.id.recipe_title);
    mSourceTextView= (TextView) itemView.findViewById(R.id.recipe_source);
    mNoteTextView= (TextView) itemView.findViewById(R.id.recipe_note);
    mEditIcon=(ImageView) itemView.findViewById(R.id.recipe_edit);
    mSunIcon=(ImageView) itemView.findViewById(R.id.recipe_img_sun);
    mIceIcon=(ImageView) itemView.findViewById(R.id.recipe_img_ice);
    mDifficulty=(TextView) itemView.findViewById(R.id.recipe_difficulty);
    mPhotoView=(ImageView) itemView.findViewById(R.id.recipe_photo);
    mPhotoFile=CookBook.get(getActivity()).getPhotoFile(mRecipe);
    mRating=(RatingBar) itemView.findViewById(R.id.recipe_list_ratingBar);
    mRating.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            //Toast.makeText(getActivity(), "Call star",
Toast.LENGTH_SHORT).show();
        }
    });
    mTitleTextView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mSortOption=mSortOption ^ maskSortTitle;
            updateUI();
        }
    });
    mNoteTextView.setOnClickListener(new View.OnClickListener(){
        @Override
        public void onClick(View v){
            mSortOption=mSortOption ^ maskSortNote;
            updateUI();
        }
    });
    mSourceTextView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mSortOption=mSortOption ^ maskSortSource;
            updateUI();
        }
    });
    //-

```

```

mEditIcon.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
    }
});
mSunIcon.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mSortOption=mSortOption ^ maskSortSeason;
        updateUI();
    }
});
mIceIcon.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mSortOption=mSortOption ^ maskSortSeason;
        updateUI();
    }
});
mDifficulty.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mSortOption=mSortOption ^ maskSortDifficulty;
        updateUI();
    }
});
mEditIcon.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent=
RecipeActivity.newIntent(getActivity(),mRecipe.getId());
        startActivity(intent);
    }
});
mPhotoView.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent
intent=RecipeDisplayActivity.newIntent(getActivity(),mRecipe.getId());
        startActivity(intent);
    }
});
}
public void bind(Recipe recipe){
    mRecipe=recipe;
    mTitleTextView.setText(mRecipe.getTitle());
    mSourceTextView.setText("(" +mRecipe.getOwner().getName()+")");
    mRating.setRating((float) mRecipe.getNoteAvg());
    DecimalFormat df = new DecimalFormat("#.##");
    mNoteTextView.setText(df.format(mRecipe.getNoteAvg())+"/5");
    mPhotoFile=CookBook.get(getActivity()).getPhotoFile(mRecipe);
    int idff= RecipeDifficulty.getIndex(mRecipe.getDifficulty());
    String[] stringArray =
getResources().getStringArray(R.array.recipe_difficulty_array);

```

```

        mDifficulty.setText(stringArray[idff]);
        if (mPhotoFile==null || !mPhotoFile.exists()){
mPhotoView.setImageDrawable(getResources().getDrawable(R.drawable.ic_recipe_see));
        } else {
            Bitmap bitmap=PictureUtils.getScaledBitmap(mPhotoFile.getPath(),
getActivity());
            mPhotoView.setImageBitmap(bitmap);
        }
        mIceIcon.setImageResource((mRecipe.IsWinter()) ? R.drawable.ic_recipe_ice :
R.drawable.ic_recipe_ice_disabled);
        mSunIcon.setImageResource((mRecipe.IsSummer()) ? R.drawable.ic_recipe_sun :
R.drawable.ic_recipe_sun_disabled);

mEditIcon.setVisibility((mRecipe.getOwner().getId().equals(mSession.getUser().getId()))
? View.VISIBLE : View.GONE);
    }

    @Override
    public void onClick(View v){
        FragmentManager manager = getFragmentManager();
        RatePickerFragment dialog =
RatePickerFragment.newInstance(mRecipe.getId());
        dialog.setTargetFragment(RecipeListFragment.this, REQUEST_RATE);
        dialog.show(manager, DIALOG_RATE);
    }
}

private class RecipeAdapter extends RecyclerView.Adapter<RecipeHolder> {
    private List<Recipe> mRecipes;
    public RecipeAdapter(List<Recipe> recipes){
        mRecipes=recipes;
    }

    @NonNull
    @Override
    public RecipeHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        LayoutInflater inflater=LayoutInflater.from(getActivity());
        return new RecipeHolder(inflater, parent);
    }

    @Override
    public void onBindViewHolder(@NonNull RecipeHolder recipeHolder, int i) {
        Recipe recipe=mRecipes.get(i);
        recipeHolder.bind(recipe);
    }

    @Override
    public int getItemCount() {
        return mRecipes.size();
    }

    public void setRecipes(List<Recipe> recipes){

```

```
        mRecipes=recipes;
    }
}
```



```
package com.example.cookbook;

public enum RecipeSeason {
    WINTER,SUMMER,ALLYEAR;

    private static RecipeSeason[] list=RecipeSeason.values();

    public static RecipeSeason getSeason(int i){
        return list[i];
    }

    public static int getIndex(RecipeSeason r){
        for(int i=0;i<list.length;i++) {
            if (list[i].equals(r)){ return i;}
        }
        return 0;
    }
}
```

```
package com.example.cookbook;

import android.content.Context;

import java.util.UUID;

public class SessionInfo {
    private static SessionInfo ourInstance;
    private User mUser;
    private Context mContext;

    public static SessionInfo get(Context context) {
        if (ourInstance==null){
            ourInstance= new SessionInfo(context);
        }
        return ourInstance;
    }

    private SessionInfo(Context context) {
        mContext=context.getApplicationContext();
        mUser=new User("Devaux_Lion de ML","Fabrice");
        // UID constant pour phase debug
        mUser.setId(UUID.fromString("c81d4e2e-bcf2-11e6-869b-7df92533d2db"));
    }

    public String getUsernameComplete() {
        return mUser.getNameComplete();
    }

    public User getUser() {
        return mUser;
    }
}
```

```
package com.example.cookbook;

import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v7.app.AppCompatActivity;

public abstract class SingleFragmentActivity extends AppCompatActivity {
    protected abstract Fragment createFragment();
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_fragment);
        FragmentManager fm = getSupportFragmentManager();
        Fragment fragment = fm.findFragmentById(R.id.fragment_container);
        if (fragment == null) {
            fragment = createFragment();
            fm.beginTransaction()
                .add(R.id.fragment_container, fragment)
                .commit();
        }
    }
}
```

```
package com.example.cookbook;

import android.util.Log;

import java.util.UUID;

public class User {
    private UUID mId;
    private String mFamily;
    private String mName;
    private static final String TAG = "DebugUser";

    public User(String family, String name){
        mFamily=family;
        mName=name;
        mId =UUID.randomUUID();
    }

    public User(UUID uuid){
        mId =uuid;
        // waiting for user database
        switch(mId.toString()){
            case "c81d4e2e-bcf2-11e6-869b-7df92533d2db":
                mFamily="Devaux_Lion de ML";
                mName="Fabrice";
                return;
            case "c81d4e2e-bcf2-11e7-869b-7df92533d2db":
                mFamily="Devaux_Lion de ML";
                mName="Lucile";
                return;
            case "c81d4e2e-bcf3-11e6-869b-7df92533d2db":
                mFamily="Devaux_Lion de ML";
                mName="Véronique";
                return;
            default:
                mFamily="not found";
                mName="not found";
                return;
        }
    }

    public UUID getId() {
        return mId;
    }

    public void setId(UUID id) {
        this.mId = id;
    }

    public String getFamily() {
        return mFamily;
    }

    public void setFamily(String family) {
```

```
        mFamily = family;
    }

    public String getName() {
        return mName;
    }

    public void setName(String name) {
        mName = name;
    }

    public boolean IsEqualUser(User u){
        return (this==u);
    }

    public String getNameComplete() {
        return mName+" de "+ mFamily;
    }

    public boolean IsEqual(User r){return (mId==r.getId());}

}
```

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/fragment_container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"/>
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/fragment_container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="16dp"
    android:orientation="vertical">

    <ScrollView
        android:id="@+id/fragment_recipe_display_scroll"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:scrollbarStyle="insideInset">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginLeft="8dp"
                android:layout_marginTop="8dp"
                android:orientation="horizontal">

                <LinearLayout
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:orientation="vertical">

                    <ImageView
                        android:id="@+id/recipe_display_photo"
                        android:layout_width="100dp"
                        android:layout_height="100dp"
                        android:layout_marginBottom="8dp"
                        android:background="@android:color/darker_gray"
                        android:cropToPadding="true"
                        android:scaleType="centerCrop" />

                </LinearLayout>

                <LinearLayout
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:layout_weight="1"
                    android:orientation="vertical">

                    <TextView
                        android:id="@+id/recipe_display_title"
                        style="@style/CookBookText.Title"
```

```
        android:layout_marginStart="8dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/recipe_title_hint" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:layout_marginTop="8dp"
    android:orientation="horizontal">

    <RatingBar
        android:id="@+id/recipe_display_ratingBar"
        style="?android:attr/ratingBarStyleSmall"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:layout_marginBottom="8dp"
        android:numStars="5"
        android:progressTint="@color/fg_enter_label"
        android:stepSize="0.1"
        />

    <TextView
        android:id="@+id/recipe_display_ratingBar_txt"
        style="@style/CookBookText.Small"
        android:layout_marginStart="8dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/list_item_recipe_note_label" />
</LinearLayout>

<TextView
    android:id="@+id/recipe_display_author"
    style="@style/CookBookText.Small"
    android:layout_marginStart="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_display_author" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:layout_marginTop="8dp"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/recipe_display_difficulty"
        style="@style/CookBookText"
        android:layout_width="120dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
```



```
        android:text="difficulty" />
    <ImageView
        android:id="@+id/recipe_img_sun"
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:layout_marginStart="16dp"
        android:layout_marginEnd="8dp"
        android:tint="@color/fg_icon"
        android:src="@drawable/ic_recipe_sun" />
    <ImageView
        android:id="@+id/recipe_img_ice"
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:layout_marginStart="16dp"
        android:layout_marginEnd="8dp"
        android:tint="@color/fg_icon"
        android:src="@drawable/ic_recipe_ice" />
</LinearLayout>

</LinearLayout>

</LinearLayout>

<TextView
    android:id="@+id/recipe_display_source"
    style="@style/CookBookText"
    android:layout_marginStart="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_source_hint" />

<TextView
    android:id="@+id/recipe_display_source_url"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:autoLink="web|none"
    android:textColorLink="@color/fg_icon"
    android:text="@string/recipe_source_url_hint" />

<TextView
    android:id="@+id/recipe_display_title_ing"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    style="@style/CookBookText.SubTitle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_display_title_ing"/>

<TextView
    android:id="@+id/recipe_display_I01"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I02"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I03"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I04"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I05"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I06"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I07"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I08"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I09"
```

```
        android:layout_marginStart="8dp"
        style="@style/CookBookText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I10"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I11"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I12"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I13"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I14"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:id="@+id/recipe_display_I15"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label"/>
<TextView
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    style="@style/CookBookText.SubTitle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
```

```
        android:text="@string/recipe_display_title_step"/>

<TextView
    android:id="@+id/recipe_display_S1"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S1_label"/>
<TextView
    android:id="@+id/recipe_display_S2"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S2_label"/>
<TextView
    android:id="@+id/recipe_display_S3"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S3_label"/>
<TextView
    android:id="@+id/recipe_display_S4"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S4_label"/>
<TextView
    android:id="@+id/recipe_display_S5"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S5_label"/>
<TextView
    android:id="@+id/recipe_display_S6"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S6_label"/>
<TextView
    android:id="@+id/recipe_display_S7"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S1_label"/>
<TextView
    android:id="@+id/recipe_display_S8"
    android:layout_marginStart="8dp"
```

```
        style="@style/CookBookText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/recipe_S8_label"/>
<TextView
    android:id="@+id/recipe_display_S9"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_S9_label"/>

<TextView
    android:id="@+id/recipe_display_comment_title"
    style="@style/CookBookText.SubTitle"
    android:layout_marginTop="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:text="@string/recipe_display_title_comment" />

<TextView
    android:id="@+id/recipe_display_C01"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C02"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C03"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C04"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C05"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C06"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C07"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C08"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C09"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C10"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C11"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C12"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C13"
    android:layout_marginStart="8dp"
```

```
        style="@style/CookBookText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C14"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C15"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C16"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C17"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C18"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C19"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<TextView
    android:id="@+id/recipe_display_C20"
    android:layout_marginStart="8dp"
    style="@style/CookBookText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_C_label"/>
<LinearLayout
```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:orientation="horizontal">
    <EditText
        android:id="@+id/recipe_display_enter_comment"
        style="@style/EnterValueField"
        android:layout_width="310dp"
        android:layout_height="wrap_content"
        android:text="@string/recipe_comment_hint"/>
    <ImageView
        android:id="@+id/recipe_img_enter"
        android:layout_width="20dp"
        android:layout_height="20dp"
        android:layout_marginStart="16dp"
        android:layout_marginEnd="8dp"
        android:tint="@color/fg_icon"
        android:src="@drawable/ic_recipe_enter" />
    </LinearLayout>
</LinearLayout>
</ScrollView>
```

```
</LinearLayout>
```



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/fragment_container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="16dp"
    android:orientation="vertical">

    <ScrollView
        android:id="@+id/fragment_recipe_scroll"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:scrollbarStyle="insideInset">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginLeft="8dp"
                android:layout_marginTop="8dp"
                android:orientation="horizontal">

                <LinearLayout
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:orientation="vertical">

                    <ImageView
                        android:id="@+id/recipe_photo"
                        android:layout_width="100dp"
                        android:layout_height="100dp"
                        android:background="@android:color/darker_gray"
                        android:cropToPadding="true"
                        android:scaleType="centerCrop" />

                </LinearLayout>

                <LinearLayout
                    android:layout_width="0dp"
                    android:layout_height="wrap_content"
                    android:layout_weight="1"
                    android:orientation="vertical">

                    <TextView
                        style="?android:listSeparatorTextViewStyle"
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content"
                        android:text="@string/recipe_title_label" />

                </LinearLayout>

            </LinearLayout>

        </LinearLayout>

    </ScrollView>

</LinearLayout>
```

```
        <EditText
            android:id="@+id/recipe_title"
            style="@style/EnterValueField"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/recipe_title_hint" />

    </LinearLayout>

</LinearLayout>

<TextView
    style="?android:listSeparatorTextViewStyle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_source_label" />

<EditText
    android:id="@+id/recipe_source"
    style="@style/EnterValueField"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="text"
    android:text="@string/recipe_source_hint"
    />

<TextView
    style="?android:listSeparatorTextViewStyle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_source_url_label" />

<EditText
    android:id="@+id/recipe_source_url"
    style="@style/EnterValueField"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textUri"
    android:text="@string/recipe_source_url_hint" />

<android.support.constraint.ConstraintLayout
    android:id="@+id/linearlayout2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <TextView
        android:id="@+id/textView"
        style="?android:listSeparatorTextViewStyle"
        android:layout_width="0dp"
        android:layout_height="29dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
```

```
        android:text="@string/recipe_nbpers_label"
        app:layout_constraintEnd_toStartOf="@+id/recipe_nbpers"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

<EditText
    android:id="@+id/recipe_nbpers"
    style="@style/EnterValueField"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:gravity="right"
    android:inputType="number"
    android:minWidth="30dp"
    android:text="@string/recipe_nbpers_hint"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toEndOf="@+id/textView"
    app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:orientation="horizontal">

    <Spinner
        android:id="@+id/recipe_season"
        style="@style/EnterValueField"
        android:layout_width="wrap_content"
        android:layout_height="32dp"

        android:layout_alignParentStart="true"
        android:layout_marginEnd="8dp"
        android:layout_toStartOf="@+id/recipe_difficulty"
        android:minWidth="150dp"
        android:spinnerMode="dropdown"
        tools:background="@color/bg_enter_val" />

    <Spinner
        style="@style/EnterValueField"
        android:id="@+id/recipe_difficulty"
        android:layout_width="wrap_content"
        android:layout_height="32dp"
        android:minWidth="150dp"
        android:spinnerMode="dropdown"
        tools:background="@color/bg_enter_val"
        android:layout_alignParentEnd="true"/>

</RelativeLayout>
```

```
<TextView
    style="?android:listSeparatorTextViewStyle"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_ingredient" />

<TextView
    android:id="@+id/recipe_I01"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I02"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I03"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I04"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I05"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I06"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I07"
    android:layout_marginLeft="8dp"
```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I08"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I09"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I10"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I11"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I12"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I13"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
    android:id="@+id/recipe_I14"
    android:layout_marginLeft="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/recipe_I_label" />

<TextView
```

```
        android:id="@+id/recipe_I15"
        android:layout_marginLeft="8dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/recipe_I_label" />

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:layout_marginBottom="8dp"
    android:text="@string/recipe_new_ing_label" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:orientation="horizontal">

    <ImageButton
        android:id="@+id/recipe_ing_back"
        android:layout_width="30dp"
        android:layout_height="30dp"
        android:layout_gravity="left"
        android:src="@drawable/ic_recipe_back" />

    <EditText
        android:id="@+id/recipe_new_ing"
        style="@style/EnterValueField"
        android:layout_width="290dp"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:text="@string/recipe_new_ing_hint" />

    <ImageButton
        android:id="@+id/recipe_ing_enter"
        android:layout_width="30dp"
        android:layout_height="30dp"
        android:layout_gravity="right"
        android:src="@drawable/ic_recipe_enter" />
</LinearLayout>

<TextView
    style="?android:listSeparatorTextViewStyle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_step" />

<TextView
    android:id="@+id/recipe_S1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
```

```
        android:layout_marginLeft="8dp"
        android:text="@string/recipe_S1_label" />

<TextView
    android:id="@+id/recipe_S2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_S2_label" />

<TextView
    android:id="@+id/recipe_S3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_S3_label" />

<TextView
    android:id="@+id/recipe_S4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_S4_label" />

<TextView
    android:id="@+id/recipe_S5"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_S5_label" />

<TextView
    android:id="@+id/recipe_S6"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_S6_label" />

<TextView
    android:id="@+id/recipe_S7"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_S7_label" />

<TextView
    android:id="@+id/recipe_S8"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:text="@string/recipe_S8_label" />

<TextView
    android:id="@+id/recipe_S9"
```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="8dp"
        android:text="@string/recipe_S9_label" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="8dp"
            android:layout_marginBottom="8dp"
            android:text="@string/recipe_new_recipe_label" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="8dp"
            android:orientation="horizontal">

            <ImageButton
                android:id="@+id/recipe_step_back"
                android:layout_width="30dp"
                android:layout_height="30dp"
                android:layout_gravity="left"
                android:src="@drawable/ic_recipe_back" />

            <EditText
                android:id="@+id/recipe_new_step"
                style="@style/EnterValueField"
                android:layout_width="290dp"
                android:layout_height="wrap_content"
                android:inputType="text"
                android:text="@string/recipe_new_recipe_hint" />

            <ImageButton
                android:id="@+id/recipe_step_enter"
                android:layout_width="30dp"
                android:layout_height="30dp"
                android:layout_gravity="right"
                android:src="@drawable/ic_recipe_enter" />

        </LinearLayout>
    </LinearLayout>
</ScrollView>
</LinearLayout>
```



```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.RecyclerView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/recipe_recycler_view"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

</android.support.v7.widget.RecyclerView>
```

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/linearLayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="8dp">

    <ImageView
        android:id="@+id/recipe_photo"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:layout_marginStart="4dp"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        android:background="@android:color/darker_gray"
        android:cropToPadding="true"
        android:scaleType="centerCrop"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/recipe_title"
        style="@style/CookBookText.Title"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:text="@string/list_item_recipe_title_label"
        android:textColor="@android:color/holo_orange_light"
        android:textStyle="bold"
        app:layout_constraintStart_toEndOf="@+id/recipe_photo"
        app:layout_constraintTop_toTopOf="parent"
        tools:textSize="14sp" />

    <TextView
        android:id="@+id/recipe_source"
        android:layout_width="0dp"
        android:layout_height="21dp"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:gravity="right"
        android:text="@string/list_item_recipe_source_label"
        app:layout_constraintStart_toEndOf="@+id/recipe_title"
        app:layout_constraintTop_toTopOf="parent" />

    <RatingBar
        android:id="@+id/recipe_list_ratingBar"
        style="?android:attr/ratingBarStyleSmall"
        android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
android:progressTint="@color/fg_enter_label"
android:layout_marginStart="8dp"
android:layout_marginBottom="8dp"
android:stepSize="0.1"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintStart_toEndOf="@+id/recipe_photo"
tools:numStars="5" />
```

<TextView

```
android:id="@+id/recipe_note"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:layout_marginStart="8dp"
android:layout_marginBottom="8dp"
android:text="@string/list_item_recipe_note_label"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintStart_toEndOf="@+id/recipe_list_ratingBar" />
```

<TextView

```
android:id="@+id/recipe_difficulty"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginEnd="16dp"
android:layout_marginBottom="8dp"
android:textStyle="italic"
android:text="@string/list_item_difficulty"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toStartOf="@+id/recipe_img_sun" />
```

<ImageView

```
android:id="@+id/recipe_edit"
android:layout_width="20dp"
android:layout_height="20dp"
android:layout_marginTop="8dp"
android:layout_marginEnd="4dp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:srcCompat="@android:drawable/ic_menu_edit" />
```

<ImageView

```
android:id="@+id/recipe_img_sun"
android:layout_width="20dp"
android:layout_height="20dp"
android:layout_marginEnd="48dp"
android:layout_marginBottom="8dp"
android:src="@drawable/ic_recipe_sun"
android:tint="@color/fg_icon"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent" />
```

<ImageView

```
android:id="@+id/recipe_img_ice"
android:layout_width="20dp"
```

```
android:layout_height="20dp"
android:layout_marginStart="16dp"
android:layout_marginEnd="8dp"
android:layout_marginBottom="8dp"
android:tint="@color/fg_icon"
android:src="@drawable/ic_recipe_ice"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toEndOf="@+id/recipe_img_sun" />
```

```
</android.support.constraint.ConstraintLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="2dp">

    <RatingBar
        android:layout_marginTop="5dp"
        android:layout_marginBottom="5dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:clickable="true"
        android:id="@+id/dialog_ratingbar"
        android:scaleX="0.5"
        android:scaleY="0.5"
        android:progressTint="@color/fg_enter_label"
        android:layout_gravity="center"
        android:numStars="5"
        android:stepSize="1.0"
        android:rating="2"
        android:isIndicator="false"/>

</LinearLayout>
```