

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ



Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών



Εκπόνηση διπλωματικής εργασίας με θέμα:

«Ανάπτυξη εφαρμογής εντοπισμού κρασιών σε Android»

«Development of a wine tracker application in Android»

Αποστολίδης Ευάγγελος

Επιβλέπον καθηγητής: Αλκιβιάδης Ακρίτας
Συν-επιβλέπον καθηγητής: Σταμούλης Γεώργιος

Βόλος, Σεπτέμβριος 2015

Ευχαριστίες,

Με την εκπόνηση αυτής της διπλωματικής εργασίας ολοκληρώνω και τις σπουδές μου στο τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του πανεπιστημίου Θεσσαλίας.

Πρώτο από όλους θα ήθελα να ευχαριστήσω τον επιβλέπον καθηγητή μου κύριο Αλκιβιάδη Ακρίτα για την δυνατότητα που μου έδωσε να ασχοληθώ με την ανάπτυξη μιας εφαρμογής Android, καθώς και για την πολύτιμη υποστήριξη και εμπιστοσύνη που μου έδειξε κατά τη διάρκεια της δημιουργίας της.

Επιπλέον θα ήθελα να ευχαριστήσω την οικογένεια μου που σ' όλη τη διάρκεια των σπουδών μου ήταν δίπλα μου και με στήριξε με κάθε μέσο ακόμη και στις πιο δύσκολες στιγμές. Η συμπαράσταση που μου έδειξε ήταν ανεκτίμητη και ξέρω πως θα είναι πάντα δίπλα μου.

Τέλος θέλω να ευχαριστήσω τους φίλους μου για την συμπαράσταση που μου έδειξαν όλο αυτό το καιρό καθώς και για τις πολύτιμες συμβουλές τους οι οποίες με βοήθησαν εξαιρετικά και για τις οποίες θα τους είμαι πάντα ευγνώμων .

Η εργασία είναι αφιερωμένη στους γονείς μου,Κωνσταντίνο και Αναστασία και στον αδερφό μου Γιώργο.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΕΧΟΜΕΝΑ.....	3
ΠΕΡΙΛΗΨΗ.....	4
ABSTRACT.....	5
1.ΕΙΣΑΓΩΓΗ.....	6
1.1Περιγραφή εφαρμογής.....	6
1.2Ροή εφαρμογής.....	7
2.ΕΡΓΑΛΕΙΑ.....	19
2.1 Android Studio 1.0.2.....	19
2.2 LunaPic.....	20
2.3 Συσκευές εφαρμογής.....	20
3.ΠΕΡΙΓΡΑΦΗ ΚΩΔΙΚΑ.....	21
3.1 Εισαγωγή στον κώδικα.....	21
3.2 Αρχεία .java.....	21
3.2.1 MainActivity.....	21
3.2.2 Pairings.....	23
3.2.2.1 FoodtoWine.....	24
3.2.2.2 WinetoFood.....	32
3.2.3 Search.....	40
3.2.4 Wineries.....	45
3.2.5 SplashScreen.....	54
3.3 Αρχεία .xml.....	56
3.3.1 Activity_main.xml.....	56
3.3.2 listview_main.xml.....	59
3.3.3 listview_item.xml.....	61
3.3.4 nearby_layout.xml.....	64
3.4 AndroidManifest.xml.....	66
3.5 build.gradle.....	69
4.ΣΥΝΔΙΑΣΜΟΙ ΚΡΑΣΙΟΥ ΜΕ ΦΑΓΗΤΟ.....	70
5.ΠΙΘΑΝΕΣ ΕΠΕΚΤΑΣΕΙΣ ΕΦΑΡΜΟΓΗΣ.....	71
6.ΒΙΒΛΙΟΓΡΑΦΙΑ.....	72

ΠΕΡΙΛΗΨΗ

Η παρούσα διπλωματική εργασία με τίτλο «Ανάπτυξη εφαρμογής εντοπισμού κρασιών σε Android» έχει ως στόχο τη δημιουργία μίας Android εφαρμογής η οποία θα μπορεί να χρησιμοποιηθεί από οποιαδήποτε συσκευή η οποία χρησιμοποιεί Android λειτουργικό.

Το αντικείμενο της εφαρμογής δεν επιλέχθηκε τυχαία. Τα ελληνικά προϊόντα ως γνωστών είναι υψηλής ποιότητας και προτιμούνται γιαυτό το λόγο, αντίθετα υστερούν σε προώθηση ειδικά μέσω των νέων τεχνολογιών.

Το ελληνικό κρασί αποτελεί ένα τέτοιο προϊόν και γιαυτό επιλέχθηκε. Μέσω του “WineTracker” ο χρήστης μπορεί να βρει το κρασί που ταιριάζει καλύτερα στις προτιμήσεις του ,επιπλέον του δίνεται η δυνατότητα να βρει το κατάλληλο γεύμα με το οποίο θα το συνοδεύσει καθώς και να ενημερωθεί πλήρως για τα χαρακτηριστικά και την παραγωγή του αγαπημένου του κρασιού και όλα αυτά μέσα από ένα ευχάριστο περιβάλλον .

Η δημιουργία της εφαρμογής έγινε σχεδόν εξολοκλήρου με τη χρήση του εργαλείου Android Studio το οποίο αποτελεί ένα ελεύθερο λογισμικό ειδικό για την δημιουργία Android εφαρμογών .

ABSTRACT

The current diploma thesis with title «Development of a wine tracker application in Android» has as goal to create an Android application which can be used by every device which runs Android operating system.

The subject of the application hasn't been chosen randomly. It is well known that greek products are of high quality and this is the reason why they are preferred, although they lack of promotion especially through the new technologies.

This is the reason why greek wine was chosen as subject for this application. Through the "WineTracker" user can find which wine best suits with his/her preferences, furthermore is given the possibility to find the proper meal to accompany the wine and be informed about the characteristics and the production of his/her favourite wine and all these through a pleasant interface .

Almost the entire application was made with the Android Studio which is a free software specialized in Android application development

1.ΕΙΣΑΓΩΓΗ

1.1 Περιγραφή εφαρμογής

Το “WineTracker” είναι μια εφαρμογή η οποία έχει ως στόχο να δώσει στον χρήστη όσο το δυνατόν περισσότερες πληροφορίες σχετικά με το ελληνικό κρασί.

Η εφαρμογή χωρίζεται γενικά σε τρία μέρη:

1. Συνδυασμούς (Pairings)
- 2.Ευρετήριο (Search)
- 3.Οινοποιεία (Wineries)

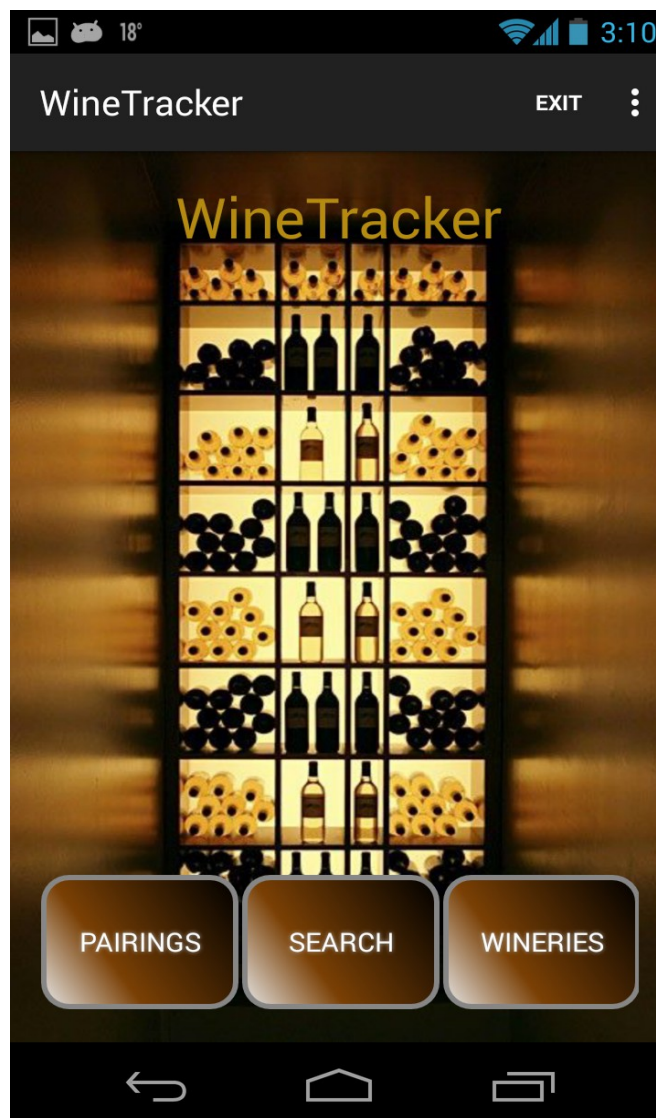
Στους συνδυασμούς δίνεται η δυνατότητα στο χρήστη να βρει το κατάλληλο τρόπο να ταιριάζει το φαγητό με το κρασί είτε δίνοντας σαν επιλογή το κρασί είτε το φαγητό

Στο ευρετήριο είναι καταχωρημένα όλα τα κρασιά με τις πληροφορίες που τα συνοδεύουν. Επιλέγοντας κάποιες ειδικά χαρακτηριστικά και την ονομασία μπορεί ο χρήστης να βρει άμεσα το κρασί που τον ενδιαφέρει.

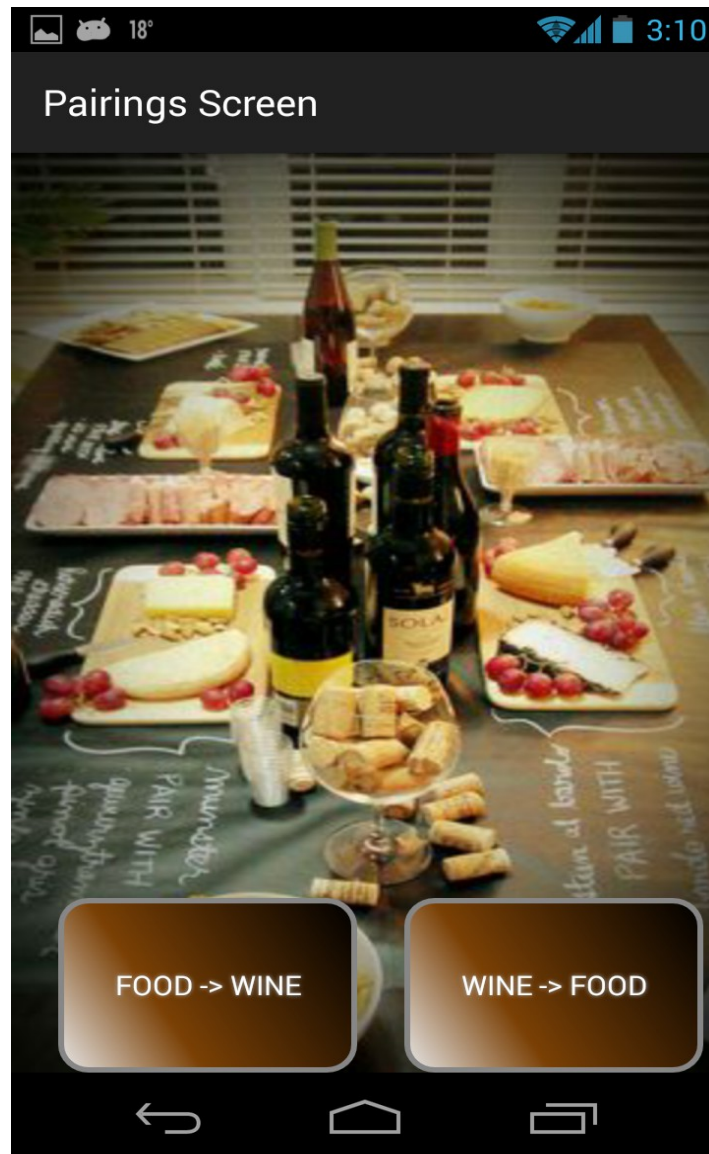
Τέλος στα οινοποιεία δίνεται η δυνατότητα στο χρήστη να δει πάνω στο χάρτη που βρίσκονται τα οινοποιεία όπου παράγονται τα κρασιά και να συλλέξει πληροφορίες σχετικά μ' αυτά.

1.2 Ροή εφαρμογής

Αρχικά στην εφαρμογή εμφανίζεται η αρχική σελίδα που είναι και η κεντρική. Από εκεί ο χρήστης μπορεί να επιλέξει ποιο από τα τρία μέρη της εφαρμογής θέλει να επισκεφθεί. Οι επιλογές δίνονται μέσω τριών κουμπιών με τα αντίστοιχα ονόματα Pairings , Search και Wineries .



Στη συνέχεια επιλέγοντας το κουμπί Pairings δίνεται η δυνατότητα να διαλέξει ανάμεσα σε δύο επιλογές Wine->Food και Food->Wine όπου στην πρώτη ανάλογα με το κρασί που επέλεξε δίνεται και το φαγητό που ταιριάζει ενώ αντίθετα στη δεύτερη επιλογή δίνοντας το φαγητό παίρνει τα κρασιά με τα οποία ταιριάζει.



Επιλέγοντας Food->Wine το περιβάλλον είναι το εξής:



Ο χρήστης μπορεί να επιλέξει ανάμεσα σε 10 φαγητά και να πάρει σαν αποτέλεσμα τα κρασιά με τα οποία ταιριάζουν καλύτερα.

Σε περίπτωση που επιλέξει Wine->Food τότε το περιβάλλον θα είναι το εξής:

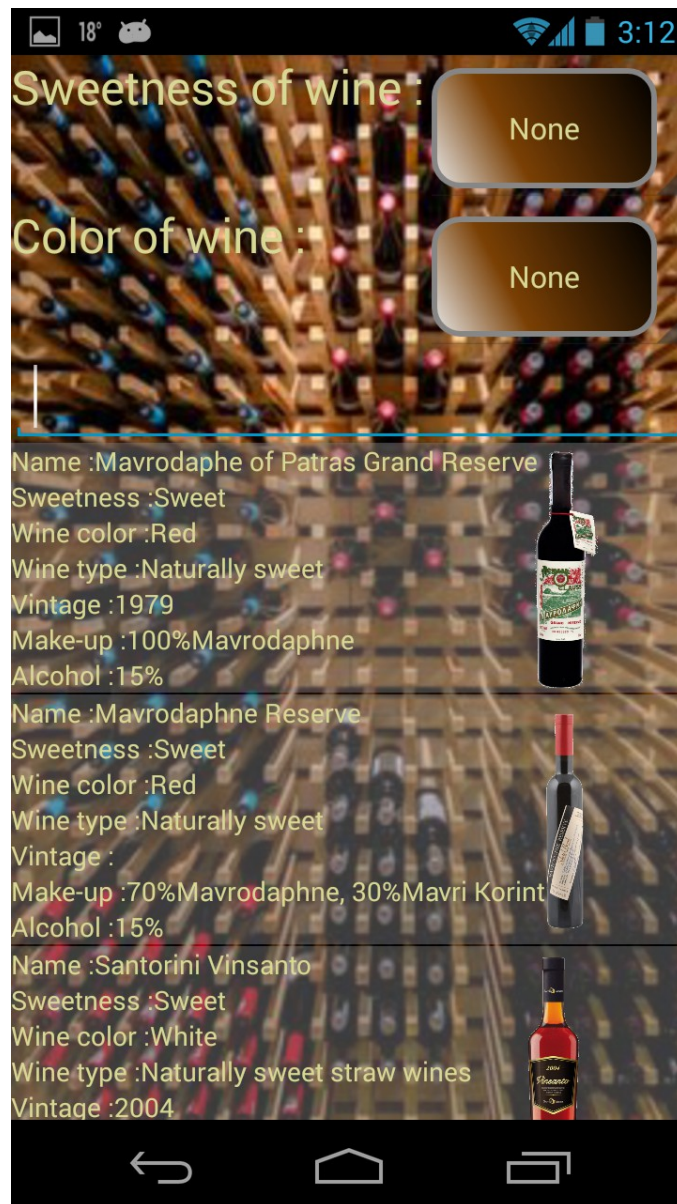


Τώρα επιλέγοντας κάποιο από τα κρασιά παίρνουμε και τα φαγητά με τα οποία ταιριάζει καλύτερα.

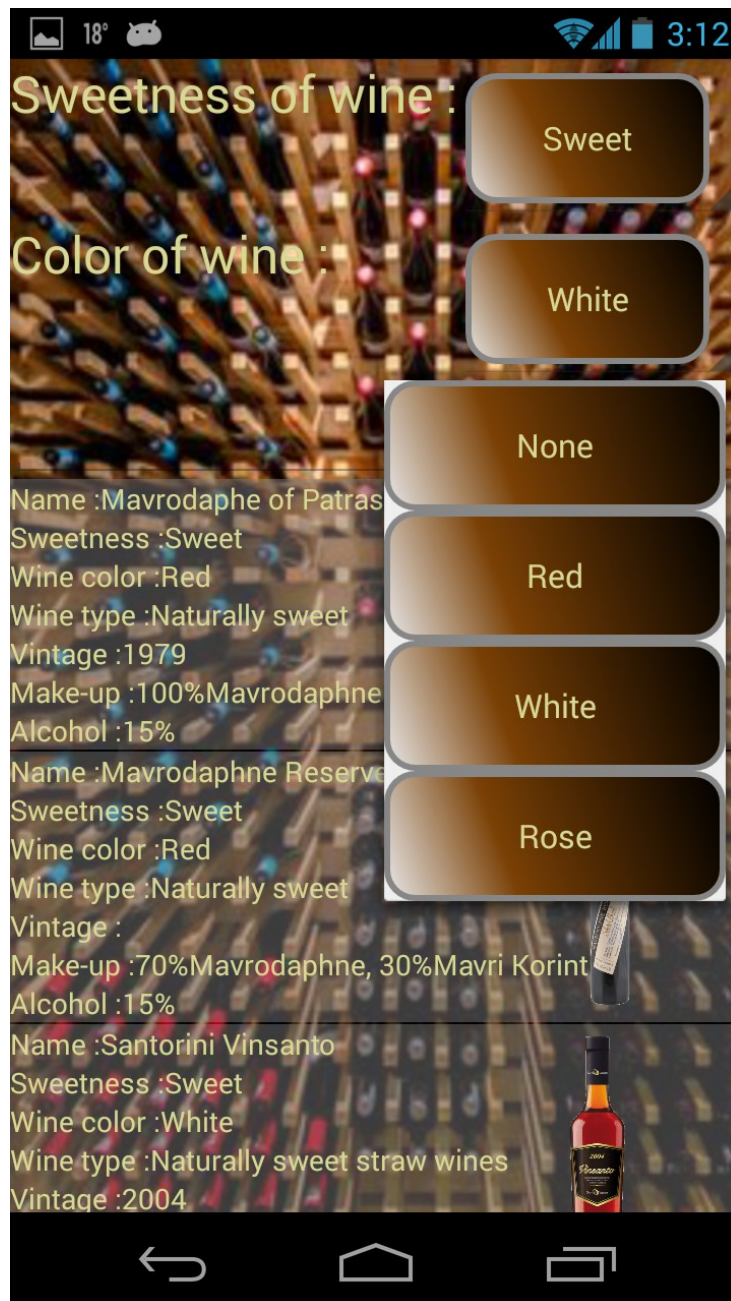
Για παράδειγμα επιλέγοντας το Santorini Visanto θα πάρουμε σαν αποτέλεσμα το εξής:








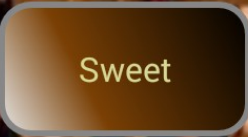
Στο πεδίο Search ο χρήστης έχει την δυνατότητα να περιηγηθεί στα κρασιά και μέσω κάποιων επιλογών να βρει γρήγορα και εύκολα το κρασί που του ταιριάζει .




Όπως φαίνεται και στην εικόνα ο χρήστης μπορεί να επιλέξει ανάμεσα στο χρώμα και στην γλυκύτητα του κρασιού και εν συνεχεία να το ψάξει αν επιθυμεί και μέσω του ονόματός του.




19°      9:49


Sweetness of wine : 

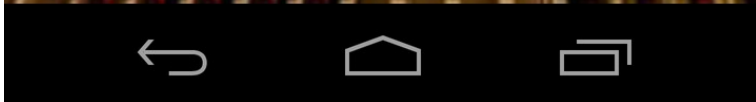
Color of wine : 

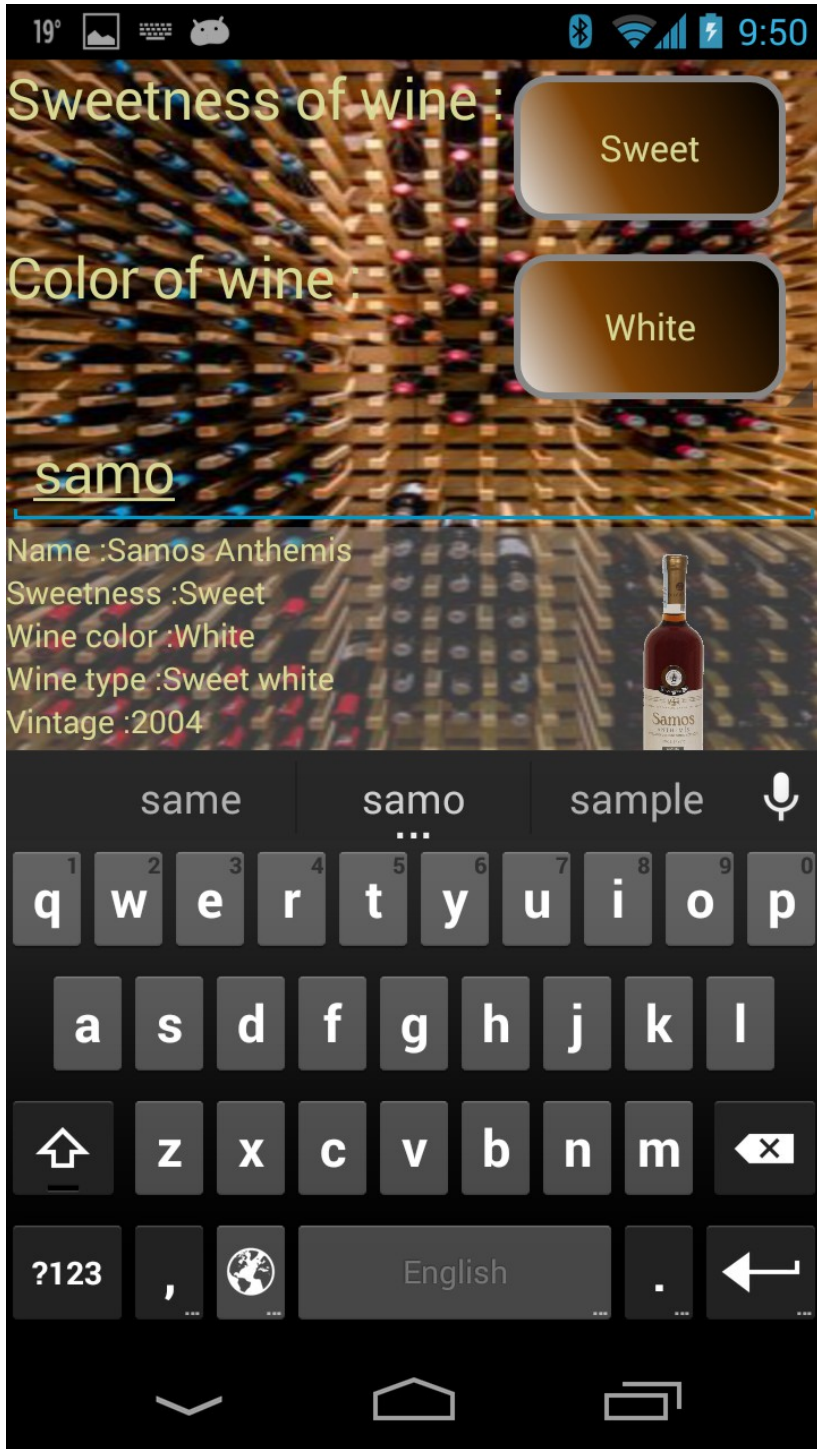
Name :Mavrodaphe of Patras Grand Reserve
Sweetness :Sweet
Wine color :Red
Wine type :Naturally sweet
Vintage :1979
Make-up :100%Mavrodaphne
Alcohol :15%



Name :Mavrodaphne Reserve
Sweetness :Sweet
Wine color :Red
Wine type :Naturally sweet
Vintage :
Make-up :70%Mavrodaphne, 30%Mavri Korint
Alcohol :15%



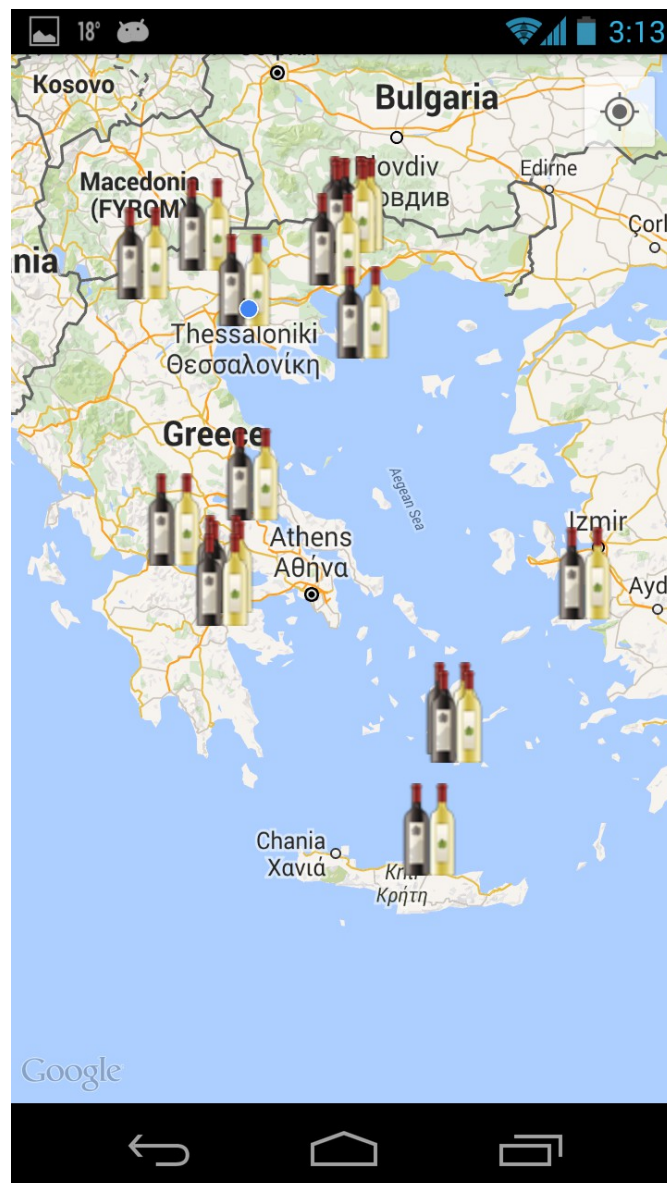




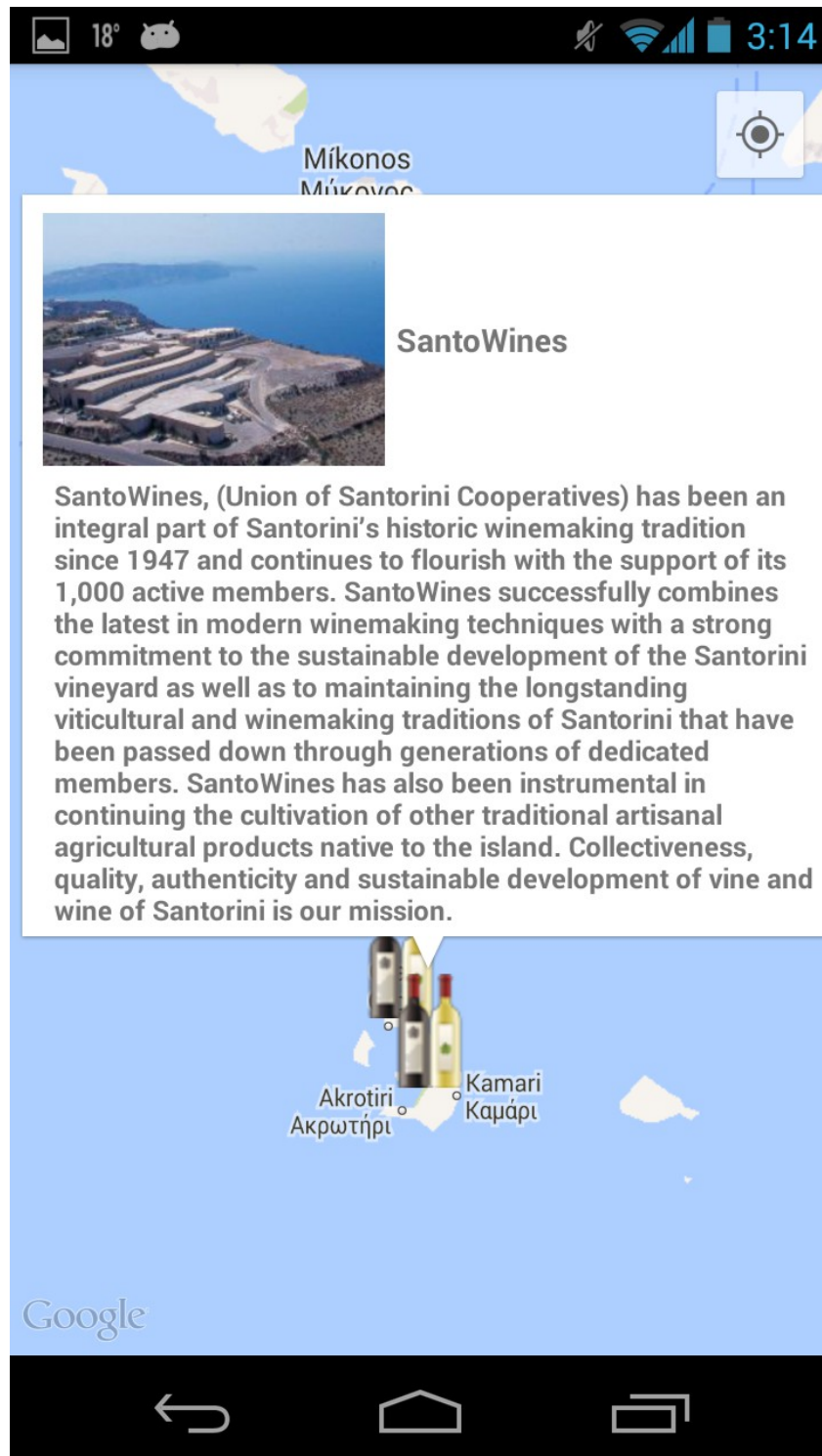
Επιλέγοντας ένα από τα κρασιά του ευρετηρίου έχει τη δυνατότητα να δει όλες τις σχετικές πληροφορίες. Πιο συγκεκριμένα μαζί με την εικόνα του μπουκαλιού από κάτω αναγράφεται το όνομα, η γλυκύτητα, το διοξείδιο του άνθρακος, το χρώμα, ο τύπος του κρασιού, η παλαιότητα, η παραγωγή, η περιεκτικότητα σε αλκοόλ, η μέθοδος παραγωγής και η γεωγραφική ένδειξη.



Τέλος στο πεδίο Wineries υπάρχει ο χάρτης με τα οινοποιεία στα οποία παρασκευάστηκαν τα κρασιά της εφαρμογής. Για κάθε οινοποιείο δίνεται και μια περιγραφή σχετικά μ'αυτό.




Μια περιγραφή του SantoWines στη Σαντορίνη μαζί με τη φωτογραφία του οينوποιείου.



18°

Míkonos
Μίκονος



SantoWines

SantoWines, (Union of Santorini Cooperatives) has been an integral part of Santorini's historic winemaking tradition since 1947 and continues to flourish with the support of its 1,000 active members. SantoWines successfully combines the latest in modern winemaking techniques with a strong commitment to the sustainable development of the Santorini vineyard as well as to maintaining the longstanding viticultural and winemaking traditions of Santorini that have been passed down through generations of dedicated members. SantoWines has also been instrumental in continuing the cultivation of other traditional artisanal agricultural products native to the island. Collectiveness, quality, authenticity and sustainable development of vine and wine of Santorini is our mission.

Akrotiri
Ακρωτήρι

Kamari
Καμάρι

Google

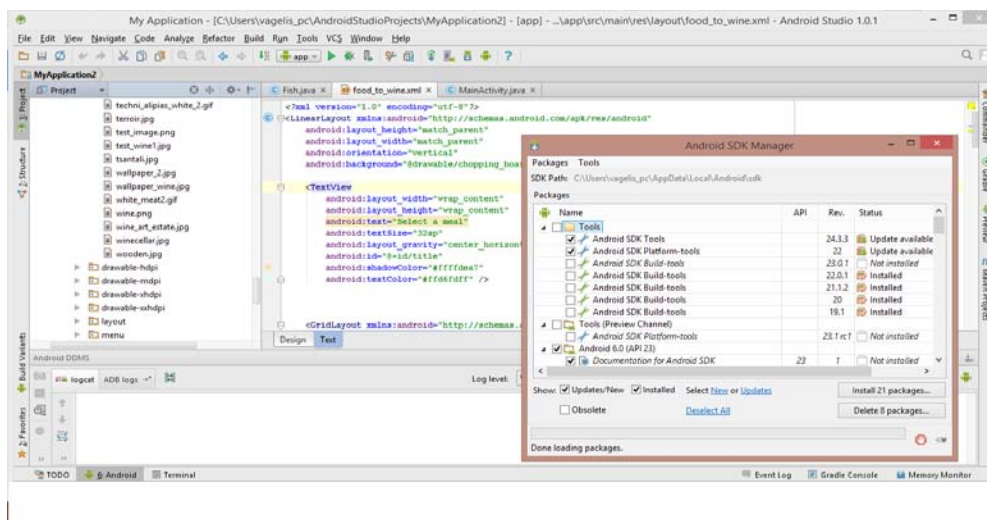
2. ΕΡΓΑΛΕΙΑ

2.1 Android Studio 1.0.2

Το εργαλείο που χρησιμοποιήθηκε περισσότερο για την δημιουργία της εφαρμογής είναι το Android Studio. Πρόκειται για ένα ολοκληρωμένο προγραμματιστικό περιβάλλον (IDE) για ανάπτυξη εφαρμογών στην Android πλατφόρμα. Το Android Studio είναι διαθέσιμο για Windows , Mac OS X και Linux, στη συγκεκριμένη εφαρμογή χρησιμοποιήθηκε η Windows έκδοση.

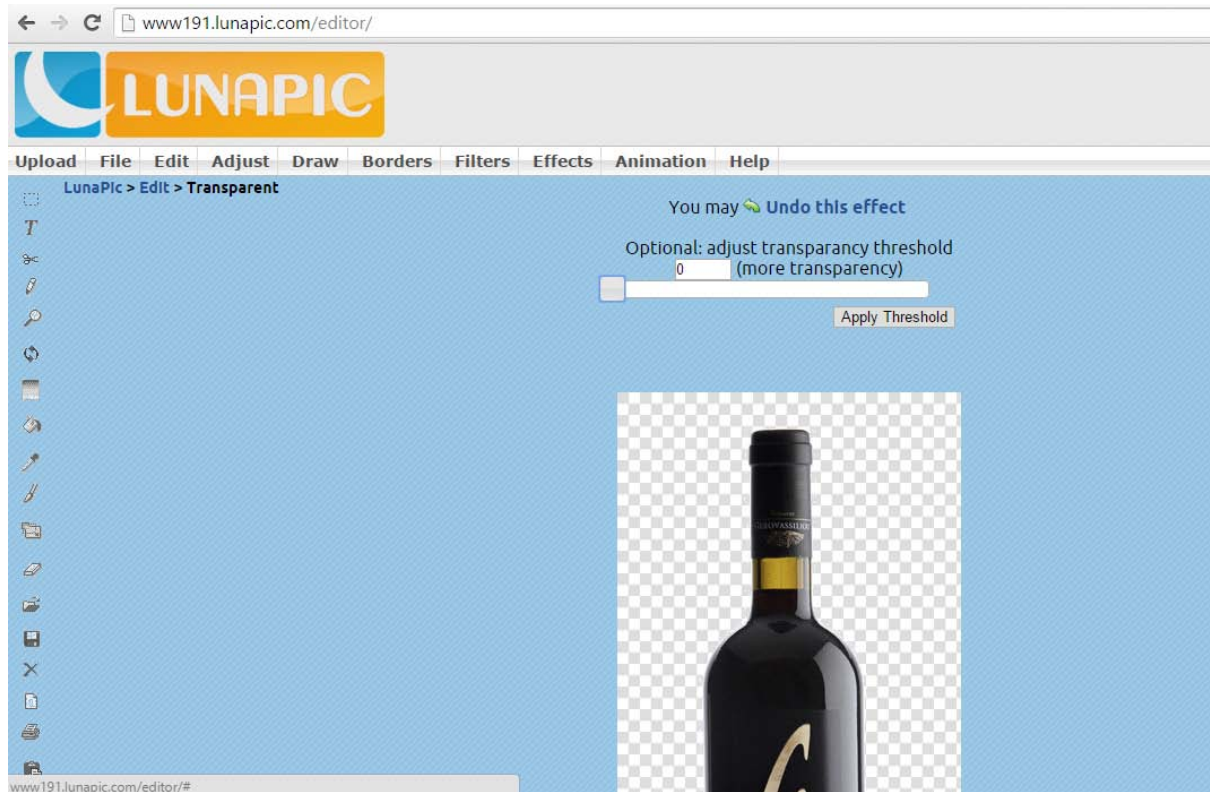
Μέσω του Android Studio γράφτηκε εξολοκλήρου ο κώδικας της εφαρμογής. Το εύκολο και ευχάριστο περιβάλλον του βοηθά στην γρήγορη και σωστή ανάπτυξη κώδικα. Η εφαρμογή έγινε με τη χρήση αρχείων .java , .xml , .jpg , .gif , .png. Τα αρχεία .java χρησιμοποιήθηκαν για τη δημιουργία της λειτουργικότητας της εφαρμογής καθώς μέσα σε αυτά καλούνται όλες οι συναρτήσεις και οι μέθοδοι που είναι απαραίτητες για την υλοποίηση της εφαρμογής. Τα αρχεία .xml χρησιμοποιήθηκαν για την εικόνα της εφαρμογής . Όλα τα κουμπιά οι λίστες και οι εικόνες είναι υλοποιημένα μέσα σ' αυτά τα αρχεία . Τέλος τα τρία τελευταία αρχεία , .jpg, .gif, .png είναι τα αρχεία εικόνων τις οποίες και βλέπουμε στην εφαρμογή.

Επιπλέον το εργαλείο αυτό μέσω του SDK Manager δίνει τη δυνατότητα να χειριστεί κανείς εύκολα τα διάφορα πακέτα και εργαλεία που χρειάζεται για την δημιουργία μιας ολοκληρωμένης εφαρμογής. SDK Tools , SDK Platform-tools , SDK Platform, System Image, Android Support Library , Android Support Repository , Google Play services , Google Repository είναι μερικά από τα πακέτα που χρησιμοποιήθηκαν για την υλοποίηση του "WineTracker" .



2.2 LunaPic

Το LunaPic είναι ένας διαδικτυακός συντάκτης φωτογραφιών (online photo editor) μέσω του οποίου έγινε η επεξεργασία των περισσότερων φωτογραφιών που χρησιμοποιούνται στην εφαρμογή.



2.3 Συσκευές Εφαρμογής

Τέλος το εργαλείο που χρησιμοποιήθηκε για την υλοποίηση της εφαρμογής ήταν και Samsung Galaxy Nexus το οποίο έπαιξε το ρόλο του “δοκιμαστή” emulator και πάνω στο οποίο τρέχει και η εφαρμογή .

3. ΠΕΡΙΓΡΑΦΗ ΚΩΔΙΚΑ

3.1 Εισαγωγή στον κώδικα

Ο κώδικας της εφαρμογής αποτελείται από τριών ειδών αρχεία. Από τα αρχεία java τα οποία είναι γραμμένα και σε γλώσσα Java και αποτελούν το λειτουργικό μέρος της εφαρμογής, τα αρχεία xml τα οποία είναι γραμμένα σε γλώσσα XML (Extensible Markup Language) και αποτελούν το σχεδιαστικό κομμάτι και τέλος τα αρχεία png, jpg, gif που αποτελούν τις εικόνες που χρησιμοποιήθηκαν. Σε αυτό το σημείο θα παρουσιαστούν τα σημαντικότερα κομμάτια του κώδικα καθώς τα αρχεία είναι πολλά και μεγάλα.

3.2 Αρχεία .java

Συνολικά δημιουργήθηκαν 26 java αρχεία. Η εφαρμογή μπορεί να χωριστεί σε 4 μέρη, την αρχική σελίδα η οποία αποτελείται κυρίως από το αρχείο MainActivity και εν συνεχεία από το Pairings, το Search και το Wineries τα οποία αποτελούνται ξεχωριστά από περισσότερα αρχεία.

3.2.1 MainActivity

```
package com.example.vagelis_pc.myapplication;

import android.app.ActionBar;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;

public class MainActivity extends AppCompatActivity {

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

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is
        present.
    }
}
```

```

        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
        if (id == R.id.action_settings) {
            return true;
        } else if (id == R.id.exit_the_app) {
            finish();
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    public void gotoPairings(View view) {

        Intent gotoPairingsScreen = new Intent(this, PairingsScreen.class);

        final int result = 1;

        gotoPairingsScreen.putExtra("callingActivity", "MainActivity");

        startActivity(gotoPairingsScreen);
    }

    public void gotoSearch(View view) {

        Intent gotoSearchScreen = new Intent(this, SearchScreen.class);

        final int result = 1;

        gotoSearchScreen.putExtra("callingActivity", "MainActivity");

        startActivity(gotoSearchScreen);
    }

    public void gotoNearby(View view) {

        Intent gotoNearbyScreen = new Intent(this, NearbyScreen.class);

        final int result = 1;

        gotoNearbyScreen.putExtra("callingActivity", "MainActivity");

        startActivity(gotoNearbyScreen);
    }
}

```

```
}
```

Όπως παρατηρεί κανείς στην MainActivity υπάρχουν κάποιες βασικές μέθοδοι όπως η `onCreateOptionsMenu` και η `onOptionsItemSelected` η οποίες χρησιμοποιούνται για τη δημιουργία του menu στο επάνω μέρος της οθόνης , εκεί που βρίσκεται και το κουμπί Exit το οποίο τερματίζει την εφαρμογή. Πέρα από αυτές τις μεθόδους υπάρχουν και άλλες τρεις οι οποίες χρησιμοποιούνται για τη μετάβαση μας από την MainActivity στα άλλα βασικά μέρη της εφαρμογής μέσω των κουμπιών. Αυτές είναι `goToPairings` η `goToSearch` και η `goToNearby` .

3.2.2 Pairings

```
package com.example.vagelis_pc.myapplication;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.ActionBarActivity;
import android.view.View;

public class PairingsScreen extends ActionBarActivity{

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pairings_layout);
    }

    public void goToFoodtoWine(View view) {

        Intent gotoFoodtoWine = new Intent(this, FoodtoWine.class);

        final int result = 1;

        gotoFoodtoWine.putExtra("callingActivity", "PairingsScreen");

        startActivity(gotoFoodtoWine);
    }

    public void goToWinetoFood(View view) {

        Intent gotoWinetoFood = new Intent(this, WinetoFood.class);

        final int result = 1;
```

```

        gotoWinetoFood.putExtra("callingActivity", "PairingsScreen");

        startActivity(gotoWinetoFood);
    }
}

```

Η κλάση `PairingsScreen` μοιάζει μ' αυτήν της `MainActivity`. Αποτελείται από 2 βασικές μεθόδους ,την `goToFoodtoWine` και την `goToWinetoFood` η οποίες και αυτές οδηγούν με τη σειρά τους σε μετάβαση μέσω κουμπιών στις αντίστοιχες κλάσεις `FoodtoWine` και `WinetoFood`.

3.2.2.1 FoodtoWine

```

package com.example.vagelis_pc.myapplication;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;

/**
 * Created by vagelis_pc on 8/27/2015.
 */
public class FoodtoWine extends Activity {

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.food_to_wine);
    }

    public void goToDessert(View view) {

        Intent gotoDessert = new Intent(this, Dessert.class);

        final int result = 1;

        gotoDessert.putExtra("callingActivity", "FoodtoWine");

        startActivity(gotoDessert);
    }

    public void goToFish(View view) {

        Intent gotoFish = new Intent(this, Fish.class);

        final int result = 1;

        gotoFish.putExtra("callingActivity", "FoodtoWine");

        startActivity(gotoFish);
    }
}

```



```

}

public void goToWhitemeat(View view) {

    Intent gotoWhitemeat = new Intent(this,WhiteMeat.class);

    final int result = 1;

    gotoWhitemeat.putExtra("callingActivity","FoodtoWine");

    startActivity(gotoWhitemeat);
}

public void goToSoftcheese(View view) {

    Intent gotoSoftcheese = new Intent(this,SoftCheese.class);

    final int result = 1;

    gotoSoftcheese.putExtra("callingActivity","FoodtoWine");

    startActivity(gotoSoftcheese);
}

public void goToRedmeat(View view) {

    Intent gotoRedmeat = new Intent(this,RedMeat.class);

    final int result = 1;

    gotoRedmeat.putExtra("callingActivity","FoodtoWine");

    startActivity(gotoRedmeat);
}

public void goToHardcheese(View view) {

    Intent gotoHardcheese = new Intent(this,HardCheese.class);

    final int result = 1;

    gotoHardcheese.putExtra("callingActivity","FoodtoWine");

    startActivity(gotoHardcheese);
}

public void goToGreenvegies(View view) {

    Intent gotoGreenvegies = new Intent(this,GreenVegies.class);

    final int result = 1;

    gotoGreenvegies.putExtra("callingActivity","FoodtoWine");
}

```

```

        startActivity(gotoGreenvegies);
    }

    public void goToRoastedvegies(View view) {

        Intent gotoRoastedvegies = new Intent(this, RoastedVegies.class);

        final int result = 1;

        gotoRoastedvegies.putExtra("callingActivity", "FoodtoWine");

        startActivity(gotoRoastedvegies);
    }

    public void goToCarbs(View view) {

        Intent gotoCarbs = new Intent(this, Carbs.class);

        final int result = 1;

        gotoCarbs.putExtra("callingActivity", "FoodtoWine");

        startActivity(gotoCarbs);
    }

    public void goToCuredsmokedmeat(View view) {

        Intent gotoCuredsmokedmeat = new
Intent(this, CuredSmokedMeat.class);

        final int result = 1;

        gotoCuredsmokedmeat.putExtra("callingActivity", "FoodtoWine");

        startActivity(gotoCuredsmokedmeat);
    }

}

```

Η κλάση FoodtoWine αποτελείται και αυτή από δέκα μεθόδους οι οποίες και αυτές μέσω κουμπιών οδηγούν σε νέες κλάσεις και αντιπροσωπεύουν τα 10 φαγητά μέσα από τα οποία μπορεί ο χρήστης να επιλέξει για να δει με ποιο κρασί μπορεί να τα ταιριάξει

```

package com.example.vagelis_pc.myapplication;

import android.app.Activity;
import android.os.Bundle;
import android.text.Editable;

```

```

import android.text.TextWatcher;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Spinner;

import java.util.ArrayList;
import java.util.Locale;

/**
 * Created by vagelis_pc on 8/30/2015.
 */
public class CuredSmokedMeat extends Activity {
    // Declare Variables

    ListView list;
    ListViewAdapterThree adapter;
    String[] name;
    String[] sweetness;
    String[] carbon_dioxide;
    String[] wine_color;
    String[] wine_type;
    String[] vintage;
    String[] varietal_makeup;
    String[] alcohol;
    String[] viticulture;
    String[] geographical_indication;
    int[] flag;
    ArrayList<Details> arraylist = new ArrayList<Details>();

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.food_listview_main);
        // Generate sample data
        name = new String[]{"Mavrodaphe of Patras Grand Reserve",
"Mavrodaphe Reserve", "Santorini Vinsanto", "Ktima Biblia Chora Merlot",
"Areti Red",
        "Domaine Gerovassiliou Viognier", "Domaine Gerovassiliou
Evangelo", "Domaine Gerovassiliou Syrah",
        "Samos Anthemis", "Samos Nectar 2007", "Alpha Xinomavro
Reserve Old Vines 2007", "Alpha Syrah",
        "Alpha Estate Rose", "Alpha Syrah Single Vineyard Turtles",
"Alpha Estate Red Blend 2006",
        "Alpha Estate White Sauvignon Blanc", "Alpha One", "Alpha
Xinomavro Single Vineyard Hedgehog",
        "Alpha Estate Utopia Tannat", "Domaine Katsaros Red",
"Oenotria Land Cabernet Sauvignon-Agiorgitiko",
        "Chateau Julia Merlot", "Domaine Mercouri Cava 2003",
"Emphasis Syrah", "Stalactite", "Kapnias Red 2006 Vineyard Zygos",
        "Terroir", "Sigalas Vinsanto Santorini", "Megas Oenos",
"Techni Alipias White", "Gaia Estate"};

        sweetness = new String[]{"Sweet", "Sweet", "Sweet", "Dry", "Dry",
"Dry", "Dry", "Dry", "Sweet", "Sweet",

```

```

        "Dry", "Dry", "Dry", "Dry", "Dry", "Dry", "Dry", "Dry",
"Dry", "Dry", "Dry", "Dry", "Dry", "Dry", "Sweet", "Dry",
        "Dry", "Sweet", "Dry", "Dry", "Dry"};

    carbon_dioxide = new String[]{"Still", "Still", "Still",
        "Still", "Still", "Still", "Still", "Still", "Still",
"Still",
        "Still", "Still", "Still", "Still", "Still", "Still",
"Still",
        "Still", "Still", "Still", "Still", "Still", "Still",
"Still",
        "Still", "Still"};

    wine_color = new String[]{"Red", "Red",
        "White", "Red", "Red", "White",
        "Red", "Red", "White", "White",
        "Red", "Red", "Rose", "Red", "Red", "White",
        "Red", "Red", "Red", "Red", "Red", "Red", "Red",
        "Red", "White", "Red", "Red", "White", "Red",
        "White", "Red"};

    wine_type = new String[]{"Naturally sweet", "Naturally sweet",
"Naturally sweet straw wines", "Dry red",
        "Dry red", "Dry white", "Dry red", "Dry red", "Sweet
white", "Naturally sweet", "Dry red",
        "Dry red", "Dry rosé", "Dry red", "Dry red", "Dry white",
"Dry red", "Dry red", "Dry red", "Dry red", "Dry red",
        "Dry red", "Dry red", "Dry red", "Naturally sweet", "Dry
red", "Dry red",
        "Naturally sweet straw wines", "Dry red", "Dry white", "Dry
red"};

    vintage = new String[]{"1979", "", "2004", "2006", "2008", "2012",
"2008",
        "2009", "2004", "2007", "2007", "2006", "2009", "2008",
"2006", "2010",
        "2007", "2008", "2007", "2005", "2007", "2008", "2003",
"2007", "2005", "2006",
        "2004", "2004", "2008", "2010", "2007"};

    varietal_makeup = new String[]{"100%Mavrodaphne", "70%Mavrodaphne,
30%Mavri Korinth.", "85%Assyrtiko 15%Aidani",
        "100% Merlot", "100% Agiorgitiko", "100% Viognier",
        "92%Syrah,8%Viognier", "100% Syrah", "100% Muscat", "100%
Muscat", "100% Xinomavro", "100% Syrah",
        "50% Syrah, 50% Xinomavro", "100% Syrah",
"60%Syrah,20%Merlot,20%Xinomavro",
        "100% Sauvignon Blanc", "60% Montepulciano, 40%
Negroamaro", "100% Xinomavro",
        "95% Tannat, 5% Xinomavro", "Cabernet Sauvignon",
"90%Cabernet Sauvignon,10%Agiorgitiko", "100% Merlot",
        "80% Refosco,20% Mavrodaphne", "100% Syrah", "100% Muscat
of Cephalonia", "100% Cabernet Sauvignon",
        "100% Agiorgitiko", "75% Assyrtiko,25% Aidani",
"80%Agiorgitiko ,20%Cabernet Sauvignon",
        "90% Sauvignon Blanc,10% Assyrtiko", "100% Agiorgitiko"};

```

```

        alcohol = new String[]{"15%", "15%", "11%", "14.5%", "14%",
"13.5%", "13.5%",
        "13.5%", "15%", "14%", "13.5%", "14.5%", "13%", "14%",
"14.5%", "13%", "14%",
        "13.5%", "14.5%", "14%", "14.5%", "15%", "13.5%", "14%",
"15%", "14%", "14%", "9%", "13.7%", "13%", "14.08%"};

        viticulture = new String[]{"conventional", "conventional",
"conventional", "organic", "organic", "sustainable", "sustainable",
"sustainable", "conventional", "conventional", "sustainable",
        "sustainable", "conventional", "sustainable",
"sustainable", "sustainable",
        "sustainable", "sustainable", "sustainable",
"conventional", "organic", "conventional",
        "conventional", "conventional", "organic", "conventional",
"organic",
        "conventional", "conventional", "conventional",
"conventional"};

        geographical_indication = new String[]{"PDO Patras", " PDO Patras",
"PDO Santorini", "PGI Pangeon", "PGI Pangeon", "PGI Epanomi", "PGI
Epanomi",
        "PGI Epanomi", "PDO Samos", "PDO Samos", " PDO Amynteo",
        "PGI Florina", "PGI Florina", "PGI Florina", "PGI Florina",
        "PGI Florina", "PGI Florina", "PDO Amynteo", "PGI Florina",
"PGI Krania", "PGI Attiki", "PGI Drama", "PGI Letrina"
        , "PGI Drama", "PDO Muscat of Cephalonia", "PGI Valley of
Atalanti", "PDO Nemea", "PDO Santorini", "PGI Peloponnese", "PGI Drama",
"PDO Nemea"};

        flag = new int[]{R.drawable.mavrodafni_2,
R.drawable.mavrodaphne_reserve_2,
        R.drawable.santorini_vinsanto_2,
R.drawable.ktima_biblia_chora_merlot_2,
        R.drawable.areti_red_2,
R.drawable.domaine_gerovassiliou_vioignier_2,
R.drawable.domaine_gerovassiliou_evangelo_2,
        R.drawable.domaine_gerovassiliou_syrah_2,
R.drawable.samos_anthemis_2, R.drawable.samos_nectar_2007_2
        , R.drawable.alpha_xinomavro_merlot,
R.drawable.alpha_syrah_2,
        R.drawable.alpha_estate_rose_2,
R.drawable.alpha_syrah_single_vineyard_turtles_2,
R.drawable.alpha_estate_blend_2006_2,
R.drawable.alpha_estate_white_sauvignon_blanc_2
        , R.drawable.alpha_one_2,
R.drawable.alpha_xinomavro_single_vineyard_hedgehog_2,
        R.drawable.alpha_estate_utopia_tannat_3,
R.drawable.domaine_katsaros_red_2,
R.drawable.oenotria_land_cabernet_sauvignon_agioritiko_2,
        R.drawable.chateau_julia_merlot_2,
        R.drawable.domaine_mercouri_cava_2003,
R.drawable.emphasis_syrah_2, R.drawable.stalactite_2,
R.drawable.kapnias_red_2,

```

```

        R.drawable.terroir, R.drawable.test_image,
R.drawable.megas_oenos_2, R.drawable.techni_alipias_white_2,
R.drawable.gaia_estate_2};

    // Locate the ListView in listview_main.xml

    list = (ListView) findViewById(R.id.listview);

    for (int i = 0; i < sweetness.length; i++) {
        Details wp = new Details(name[i], sweetness[i],
carbon_dioxide[i],
        wine_color[i], wine_type[i], vintage[i],
varietal_makeup[i], alcohol[i], viticulture[i], geographical_indication[i],
flag[i]);
        // Binds all strings into an array

        if ((wp.getSweetness().equals("Sweet") &&
wp.getWine_color().equals("Red")) || (wp.getSweetness().equals("Sweet") &&
wp.getWine_color().equals("White"))
        || (wp.getSweetness().equals("Dry") &&
wp.getWine_color().equals("Rose")) || (wp.getSweetness().equals("Dry") &&
wp.getWine_color().equals("Red"))){

            arraylist.add(wp);
        }
    }
    // Pass results to ListViewAdapter Class
    adapter = new ListViewAdapterThree(this, arraylist);

    // Binds the Adapter to the ListView
    list.setAdapter(adapter);
}
}

```

Πέρνουμε για παράδειγμα τη κλάση `CuredSmokedMeat` η οποία αναφέρεται στα αλαντικά. Στη κλάση αυτή όπως ακριβώς και στις άλλες 9 ομοίες τις υπάρχουν αρχικά τα δεδομένα των κρασιών. Αυτά έχουν καταχωρηθεί στατικά σε πίνακες. Στη συνέχεια αυτά τα δεδομένα περνάνε μέσα από μια καινούρια κλάση, την `Details` η οποία είναι υπεύθυνη για την εμφάνιση των χαρακτηριστικών των κρασιών.

```

package com.example.vagelis_pc.myapplication;

/**
 * Created by vagelis_pc on 7/10/2015.
 */
public class Details {
    private String name;
    private String sweetness;

```

```

private String carbon_dioxide;
private String wine_color;
private String wine_type;
private String vintage;
private String varietal_makeup;
private String alcohol;
private String viticulture;
private String geographical_indication;
private int flag;

public Details(String name,String sweetness, String carbon_dioxide,
String wine_color,String wine_type,String vintage,String
varietal_makeup,String alcohol,String viticulture,String
geographical_indication,
            int flag) {
    this.name = name;
    this.sweetness = sweetness;
    this.carbon_dioxide = carbon_dioxide;
    this.wine_color = wine_color;
    this.wine_type = wine_type;
    this.vintage = vintage;
    this.varietal_makeup = varietal_makeup;
    this.alcohol = alcohol;
    this.viticulture = viticulture;
    this.geographical_indication = geographical_indication;
    this.flag = flag;
}

public String getName() {return this.name;}

public String getSweetness() {
    return this.sweetness;
}

public String getCarbon_dioxide() {
    return this.carbon_dioxide;
}

public String getWine_color() {
    return this.wine_color;
}

public String getWine_type() {return this.wine_type;}

public String getVintage() {return this.vintage;}

public String getVarietal_makeup() {return this.varietal_makeup;}

public String getAlcohol() {return this.alcohol;}

public String getViticulture() {return this.viticulture;}

public String getGeographical_indication() {return
this.geographical_indication;}

```

```

    public int getFlag() {
        return this.flag;
    }
}

```

3.2.2.2 WinetoFood

```

package com.example.vagelis_pc.myapplication;

import android.app.Activity;
import android.os.Bundle;
import android.text.Editable;
import android.text.TextWatcher;
import android.widget.AdapterView;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Spinner;

import java.util.ArrayList;
import java.util.Locale;

/**
 * Created by vagelis_pc on 8/27/2015.
 */
public class WinetoFood extends Activity {

    ListView list;
    ListViewAdapterTwo adapter;
    String[] name;
    String[] sweetness;
    String[] carbon_dioxide;
    String[] wine_color;
    String[] wine_type;
    String[] vintage;
    String[] varietal_makeup;
    String[] alcohol;
    String[] viticulture;
    String[] geographical_indication;
    int[] flag;
    ArrayList<Details> arraylist = new ArrayList<Details>();

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.wine_to_food_layout);
        // Generate sample data

```



```

        name = new String[]{"Mavrodaphe of Patras Grand Reserve",
"Mavrodaphne Reserve", "Santorini Vinsanto", "Ktima Biblia Chora Merlot",
"Areti Red",
        "Domaine Gerovassiliou Viognier", "Domaine Gerovassiliou
Evangelo", "Domaine Gerovassiliou Syrah",
        "Samos Anthemis", "Samos Nectar 2007", "Alpha Xinomavro
Reserve Old Vines 2007", "Alpha Syrah",
        "Alpha Estate Rose", "Alpha Syrah Single Vineyard Turtles",
"Alpha Estate Red Blend 2006",
        "Alpha Estate White Sauvignon Blanc", "Alpha One", "Alpha
Xinomavro Single Vineyard Hedgehog",
        "Alpha Estate Utopia Tannat", "Domaine Katsaros Red",
"Oenotria Land Cabernet Sauvignon-Agiorgitiko",
        "Chateau Julia Merlot", "Domaine Mercouri Cava 2003",
"Emphasis Syrah", "Stalactite", "Kapnias Red 2006 Vineyard Zygos",
        "Terroir", "Sigalas Vinsanto Santorini", "Megas Oenos",
"Techni Alipias White", "Gaia Estate"};

        sweetness = new String[]{"Sweet", "Sweet", "Sweet", "Dry", "Dry",
"Dry", "Dry", "Dry", "Sweet", "Sweet",
        "Dry", "Dry", "Dry", "Dry", "Dry", "Dry", "Dry", "Dry", "Dry",
"Dry", "Dry", "Dry", "Dry", "Dry", "Dry", "Sweet", "Dry",
        "Dry", "Sweet", "Dry", "Dry", "Dry"};

        carbon_dioxide = new String[]{"Still", "Still", "Still",
"Still", "Still", "Still", "Still", "Still",
"Still", "Still", "Still", "Still", "Still", "Still", "Still",
"Still", "Still", "Still", "Still", "Still", "Still", "Still",
"Still", "Still", "Still", "Still", "Still", "Still", "Still",
"Still", "Still"};

        wine_color = new String[]{"Red", "Red",
"White", "Red", "Red", "White",
"Red", "Red", "White", "White",
"Red", "Red", "Rose", "Red", "Red", "White",
"Red", "Red", "Red", "Red", "Red", "Red", "Red",
"Red", "White", "Red", "Red", "White", "Red",
"White", "Red"};

        wine_type = new String[]{"Naturally sweet", "Naturally sweet",
"Naturally sweet straw wines", "Dry red",
        "Dry red", "Dry white", "Dry red", "Dry red", "Sweet
white", "Naturally sweet", "Dry red",
        "Dry red", "Dry rosé", "Dry red", "Dry red", "Dry white",
"Dry red", "Dry red", "Dry red", "Dry red", "Dry red",
        "Dry red", "Dry red", "Dry red", "Naturally sweet", "Dry
red", "Dry red",
        "Naturally sweet straw wines", "Dry red", "Dry white", "Dry
red"};

        vintage = new String[]{"1979", "", "2004", "2006", "2008", "2012",
"2008",
        "2009", "2004", "2007", "2007", "2006", "2009", "2008",
"2006", "2010",

```

```

"2007", "2008", "2007", "2005", "2007", "2008", "2003",
"2007", "2005", "2006",
"2004", "2004", "2008", "2010", "2007"};

varietal_makeup = new String[]{"100%Mavrodaphne", "70%Mavrodaphne,
30%Mavri Korinth.", "85%Assyrtiko 15%Aidani",
"100% Merlot", "100% Agiorgitiko", "100% Viognier",
"92%Syrah,8%Viognier", "100% Syrah", "100% Muscat", "100%
Muscat", "100% Xinomavro", "100% Syrah",
"50% Syrah, 50% Xinomavro", "100% Syrah",
"60%Syrah,20%Merlot,20%Xinomavro",
"100% Sauvignon Blanc", "60% Montepulciano, 40%
Negroamaro", "100% Xinomavro",
"95% Tannat, 5% Xinomavro", "Cabernet Sauvignon",
"90%Cabernet Sauvignon,10%Agiorgitiko", "100% Merlot",
"80% Refosco,20% Mavrodaphne", "100% Syrah", "100% Muscat
of Cephalonia", "100% Cabernet Sauvignon",
"100% Agiorgitiko", "75% Assyrtiko,25% Aidani",
"80%Agiorgitiko ,20%Cabernet Sauvignon",
"90% Sauvignon Blanc,10% Assyrtiko", "100% Agiorgitiko"};

alcohol = new String[]{"15%", "15%", "11%", "14.5%", "14%",
"13.5%", "13.5%",
"13.5%", "15%", "14%", "13.5%", "14.5%", "13%", "14%",
"14.5%", "13%", "14%",
"13.5%", "14.5%", "14%", "14.5%", "15%", "13.5%", "14%",
"15%", "14%", "14%", "9%", "13.7%", "13%", "14.08%"};

viticulture = new String[]{"conventional", "conventional",
"conventional", "organic", "organic", "sustainable", "sustainable",
"sustainable", "conventional", "conventional", "sustainable",
"sustainable", "conventional", "sustainable",
"sustainable", "sustainable",
"sustainable", "sustainable", "sustainable", "sustainable",
"conventional", "organic", "conventional",
"conventional", "conventional", "organic", "conventional",
"organic",
"conventional", "conventional", "conventional",
"conventional"};

geographical_indication = new String[]{"PDO Patras", " PDO Patras",
"PDO Santorini", "PGI Pangeon", "PGI Pangeon", "PGI Epanomi", "PGI
Epanomi",
"PGI Epanomi", "PDO Samos", "PDO Samos", " PDO Amynteo",
"PGI Florina", "PGI Florina", "PGI Florina", "PGI Florina",
"PGI Florina", "PGI Florina", "PDO Amynteo", "PGI Florina",
"PGI Krania", "PGI Attiki", "PGI Drama", "PGI Letrina"
, "PGI Drama", "PDO Muscat of Cephalonia", "PGI Valley of
Atalanti", "PDO Nemea", "PDO Santorini", "PGI Peloponnese", "PGI Drama",
"PDO Nemea"};

flag = new int[]{R.drawable.mavrodafni_2,
R.drawable.mavrodaphne_reserve_2,
R.drawable.santorini_vinsanto_2,
R.drawable.ktima_biblia_chora_merlot_2,

```

```

        R.drawable.aret_i_red_2,
R.drawable.domaine_gerovassiliou_vio_gnier_2,
R.drawable.domaine_gerovassiliou_evangel_o_2,
        R.drawable.domaine_gerovassiliou_syrah_2,
R.drawable.samos_anthemis_2, R.drawable.samos_nectar_2007_2
        , R.drawable.alpha_xinomavro_merlot,
R.drawable.alpha_syrah_2,
        R.drawable.alpha_estate_rose_2,
R.drawable.alpha_syrah_single_vineyard_turtles_2,
R.drawable.alpha_estate_blend_2006_2,
R.drawable.alpha_estate_white_sauvignon_blanc_2
        , R.drawable.alpha_one_2,
R.drawable.alpha_xinomavro_single_vineyard_hedgehog_2,
        R.drawable.alpha_estate_utopia_tannat_3,
R.drawable.domaine_katsaros_red_2,
R.drawable.oenotria_land_cabernet_sauvignon_agioritiko_2,
        R.drawable.chateau_julia_merlot_2,
        R.drawable.domaine_mercouri_cava_2003,
R.drawable.emphasis_syrah_2, R.drawable.stalactite_2,
R.drawable.kapnias_red_2,
        R.drawable.terroir, R.drawable.test_image,
R.drawable.megas_oenos_2, R.drawable.techni_alipias_white_2,
R.drawable.gaia_estate_2};

    // Locate the ListView in listview_main.xml
    list = (ListView) findViewById(R.id.listview);

    for (int i = 0; i < sweetness.length; i++) {
        Details wp = new Details(name[i], sweetness[i],
carbon_dioxide[i],
        wine_color[i], wine_type[i], vintage[i],
varietal_makeup[i], alcohol[i], viticulture[i], geographical_indication[i],
flag[i]);
        // Binds all strings into an array
        arraylist.add(wp);
    }

    // Pass results to ListViewAdapter Class
    adapter = new ListViewAdapterTwo(this, arraylist);

    // Binds the Adapter to the ListView
    list.setAdapter(adapter);

}

}

```

Σε αυτή τη κλάση υπάρχουν πάλι τα δεδομένα σε στατική μορφή. Εδώ χρησιμοποιείται η κλάση `ListViewAdapterTwo` η οποία δείχνει όλα τα δεδομένα οργανωμένα σε λίστα .

```
package com.example.vagelis_pc.myapplication;
```

```

/**
 * Created by vagelis_pc on 8/28/2015.
 */
import android.content.Context;
import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.ImageView;
import android.widget.TextView;

import java.util.ArrayList;
import java.util.List;
import java.util.Locale;

public class ListViewAdapterTwo extends BaseAdapter {

    // Declare Variables
    Context mContext;
    LayoutInflater inflater;
    private List<Details> detailsList = null;
    private ArrayList<Details> arraylist;

    public ListViewAdapterTwo(Context context,
                               List<Details> detailsList) {
        mContext = context;
        this.detailsList = detailsList;
        inflater = LayoutInflater.from(mContext);
        this.arraylist = new ArrayList<Details>();
        this.arraylist.addAll(detailsList);
    }

    public class ViewHolder {
        TextView name;
        TextView sweetness;
        TextView wine_color;
        TextView wine_type;
        TextView alcohol;
        ImageView flag;
    }

    @Override
    public int getCount() {
        return detailsList.size();
    }

    @Override
    public Details getItem(int position) {
        return detailsList.get(position);
    }

    @Override
    public long getItemId(int position) {

```

```

        return position;
    }

    public View getView(final int position, View view, ViewGroup parent) {
        final ViewHolder holder;
        if (view == null) {
            holder = new ViewHolder();
            view = inflater.inflate(R.layout.listview_item_two, null);

            holder.name = (TextView) view.findViewById(R.id.name);
            holder.sweetness = (TextView)
view.findViewById(R.id.sweetness);
            holder.wine_color = (TextView)
view.findViewById(R.id.wine_color);
            holder.wine_type = (TextView)
view.findViewById(R.id.wine_type);
            holder.alcohol = (TextView) view.findViewById(R.id.alcohol);
            holder.flag = (ImageView) view.findViewById(R.id.flag);

            view.setTag(holder);
        } else {
            holder = (ViewHolder) view.getTag();
        }
        // Set the results into TextViews
        holder.name.setText(detailsList.get(position).getName());
        holder.sweetness.setText(detailsList.get(position).getSweetness());
        holder.wine_color.setText(detailsList.get(position)
            .getWine_color());
        holder.wine_type.setText(detailsList.get(position).getWine_type());
        holder.alcohol.setText(detailsList.get(position).getAlcohol());
        holder.flag.setImageResource(detailsList.get(position)
            .getFlag());
        // Listen for ListView Item Click
        view.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View arg0) {
                // Send single item click data to SingleItemView Class
                Intent intent = new Intent(mContext,
NewSingleItemView.class);

                intent.putExtra("name",
                    (detailsList.get(position).getName()));
                // Pass all data rank
                intent.putExtra("sweetness",
                    (detailsList.get(position).getSweetness()));
                // Pass all data country
                intent.putExtra("carbon dioxide",
                    (detailsList.get(position).getCarbon_dioxide()));
                // Pass all data population
                intent.putExtra("wine color",
                    (detailsList.get(position).getWine_color()));

                intent.putExtra("wine type",
                    (detailsList.get(position).getWine_type()));

                intent.putExtra("vintage",

```

```

        (detailsList.get(position).getVintage());

        intent.putExtra("varietal makeup",
            (detailsList.get(position).getVarietal_makeup()));

        intent.putExtra("alcohol",
            (detailsList.get(position).getAlcohol()));

        intent.putExtra("viticulture",
            (detailsList.get(position).getViticulture()));

        intent.putExtra("geographical indication",
(detailsList.get(position).getGeographical_indication()));

        // Pass all data flag
        intent.putExtra("flag",
            (detailsList.get(position).getFlag()));
        // Start SingleItemView Class
        mContext.startActivity(intent);
    }
});

    return view;
}

// Filter Class

}

```

Μέσω της `ListViewAdapterTwo` εμφανίζονται τα κρασιά σε μια λίστα από την οποία μπορεί ο χρήστης να επιλέξει το κρασί της αρεσκείας του και στη συνέχεια να δει με ποια φαγητά ταιριάζει. Αυτό επιτυγχάνεται μέσω της `NewSingleItemView` .

```

package com.example.vagelis_pc.myapplication;

/**
 * Created by vagelis_pc on 8/28/2015.
 */
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.ImageView;
import android.widget.TextView;

public class NewSingleItemView extends Activity {
    // Declare Variables
    ImageView imgflag1;
    ImageView imgflag2;
    ImageView imgflag3;
    ImageView imgflag4;

```

```

ImageView imgflag5;

String sweetness;
String wine_color;
int[] food;

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.newsingleitemview);
    // Get the intent from ListViewAdapter

    food = new int[]
{R.drawable.dessert2,R.drawable.fish2,R.drawable.white_meat2,R.drawable.sof
t_cheese2,

R.drawable.cured_smoked_meat2,R.drawable.red_meat2,R.drawable.hard_cheese2,
R.drawable.green_veggies2
    ,R.drawable.roasted_veggies2,R.drawable.carbs2};

    Intent i = getIntent();

    sweetness = i.getStringExtra("sweetness");

    wine_color = i.getStringExtra("wine_color");

    // Locate the ImageView in singleitemview.xml
    imgflag1 = (ImageView) findViewById(R.id.flag1);
    imgflag2 = (ImageView) findViewById(R.id.flag2);
    imgflag3 = (ImageView) findViewById(R.id.flag3);
    imgflag4 = (ImageView) findViewById(R.id.flag4);
    imgflag5 = (ImageView) findViewById(R.id.flag5);

    if(sweetness.equals("Sweet") && wine_color.equals("Red")){
        imgflag1.setImageResource(food[0]);
        imgflag2.setImageResource(food[4]);
        imgflag3.setImageResource(food[3]);
    }

    if(sweetness.equals("Sweet") && wine_color.equals("White")){
        imgflag1.setImageResource(food[3]);
        imgflag2.setImageResource(food[6]);
        imgflag3.setImageResource(food[4]);
    }

    if(sweetness.equals("Dry") && wine_color.equals("Rose")){
        imgflag1.setImageResource(food[4]);
        imgflag2.setImageResource(food[5]);
        imgflag3.setImageResource(food[6]);
    }

```

```

        imgflag4.setImageResource(food[9]);
    }

    if(sweetness.equals("Dry") && wine_color.equals("Red")){
        imgflag1.setImageResource(food[2]);
        imgflag2.setImageResource(food[5]);
        imgflag3.setImageResource(food[4]);
        imgflag4.setImageResource(food[8]);
        imgflag5.setImageResource(food[6]);
    }

    if(sweetness.equals("Dry") && wine_color.equals("White")){
        imgflag1.setImageResource(food[1]);
        imgflag2.setImageResource(food[7]);
        imgflag3.setImageResource(food[8]);
    }

}
}
}

```

Εδώ γίνεται ο έλεγχος των κρασιών και ανάλογα την γεύση τους και το χρώμα τους γίνεται και επιλογή του φαγητού που ταιριάζει καλύτερα.

3.2.3 Search

```

package com.example.vagelis_pc.myapplication;

import java.util.ArrayList;
import java.util.Locale;

import android.content.Intent;
import android.os.Bundle;
import android.app.Activity;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Spinner;
import android.view.View.OnClickListener;

public class SearchScreen extends Activity {

    // Declare Variables
    Button ok_button;
    Spinner spinner;

```



```

Spinner spinner_sweetness;
ListView list;
ListAdapter adapter;
EditText editsearch;
String [] name;
String[] sweetness;
String[] carbon_dioxide;
String[] wine_color;
String[] wine_type;
String[] vintage;
String[] varietal_makeup;
String[] alcohol;
String[] viticulture;
String[] geographical_indication;
int[] flag;
ArrayList<Details> arraylist = new ArrayList<Details>();

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.listview_main);
    // Generate sample data
    name = new String[] {"Mavrodaphe of Patras Grand Reserve",
"Mavrodaphne Reserve" , "Santorini Vinsanto" , "Ktima Biblia Chora
Merlot" , "Areti Red" ,
        "Domaine Gerovassiliou Viognier" , "Domaine Gerovassiliou
Evangelos" , "Domaine Gerovassiliou Syrah" ,
        "Samos Anthemis" ,"Samos Nectar 2007" ,"Alpha Xinomavro
Reserve Old Vines 2007" ,"Alpha Syrah" ,
        "Alpha Estate Rose" ,"Alpha Syrah Single Vineyard
Turtles" ,"Alpha Estate Red Blend 2006" ,
        "Alpha Estate White Sauvignon Blanc" ,"Alpha One" ,"Alpha
Xinomavro Single Vineyard Hedgehog" ,
        "Alpha Estate Utopia Tannat" ,"Domaine Katsaros
Red" ,"Oenotria Land Cabernet Sauvignon-Agiorgitiko" ,
        "Chateau Julia Merlot" ,"Domaine Mercouri Cava
2003" ,"Emphasis Syrah" ,"Stalactite" ,"Kapnias Red 2006 Vineyard Zygus" ,
        "Terroir" ,"Sigalas Vinsanto Santorini" ,"Megas
Oenos" ,"Techni Alipias White" ,"Gaia Estate"};

    sweetness = new String[] { "Sweet", "Sweet", "Sweet", "Dry", "Dry",
"Dry", "Dry", "Dry", "Dry", "Sweet", "Sweet" ,
"Dry","Dry","Dry","Dry","Dry","Dry","Dry","Dry","Dry","Dry","Dry","Dry","Dr
y","Dry","Sweet","Dry",
        "Dry","Sweet","Dry","Dry","Dry"};

    carbon_dioxide = new String[] { "Still", "Still", "Still",
        "Still", "Still", "Still", "Still", "Still",
        "Still", "Still", "Still", "Still", "Still", "Still", "Still",
        "Still", "Still", "Still", "Still", "Still", "Still", "Still",
        "Still", "Still"};

    wine_color = new String[] { "Red", "Red",
        "White", "Red", "Red", "White",
        "Red", "Red", "White", "White" ,

```

```

"Red", "Red", "Rose", "Red", "Red", "White",
"Red", "Red", "Red", "Red", "Red", "Red", "Red",
"Red", "White", "Red", "Red", "White", "Red",
"White", "Red"};

```

```

wine_type = new String[] { "Naturally sweet", "Naturally sweet",
"Naturally sweet straw wines", "Dry red",
"Dry red", "Dry white", "Dry red", "Dry red", "Sweet
white", "Naturally sweet", "Dry red",
"Dry red", "Dry rosé", "Dry red", "Dry red", "Dry white", "Dry
red", "Dry red", "Dry red", "Dry red", "Dry red",
"Dry red", "Dry red", "Dry red", "Naturally sweet", "Dry
red", "Dry red",
"Naturally sweet straw wines", "Dry red", "Dry white", "Dry
red"};

```

```

vintage = new String[] { "1979", "", "2004", "2006", "2008",
"2012", "2008",
"2009", "2004",
"2007", "2007", "2006", "2009", "2008", "2006", "2010",
"2007", "2008", "2007", "2005", "2007", "2008", "2003", "2007", "2005", "2006",
"2004", "2004", "2008", "2010", "2007"};

```

```

varietal_makeup = new String[] { "100%Mavrodaphne", "70%Mavrodaphne,
30%Mavri Korinth.", "85%Assyrtiko 15%Aidani",
"100% Merlot", "100% Agiorgitiko", "100% Viognier",
"92%Syrah, 8%Viognier", "100% Syrah", "100% Muscat", "100%
Muscat", "100% Xinomavro", "100% Syrah",
"50% Syrah, 50% Xinomavro", "100%
Syrah", "60%Syrah, 20%Merlot, 20%Xinomavro",
"100% Sauvignon Blanc", "60% Montepulciano, 40%
Negroamaro", "100% Xinomavro",
"95% Tannat, 5% Xinomavro", "Cabernet
Sauvignon", "90%Cabernet Sauvignon, 10%Agiorgitiko", "100% Merlot",
"80% Refosco, 20% Mavrodaphne", "100% Syrah", "100% Muscat of
Cephalonia", "100% Cabernet Sauvignon",
"100% Agiorgitiko", "75% Assyrtiko, 25%
Aidani", "80%Agiorgitiko, 20%Cabernet Sauvignon",
"90% Sauvignon Blanc, 10% Assyrtiko", "100% Agiorgitiko"};

```

```

alcohol = new String[] { "15%", "15%", "11%", "14.5%", "14%",
"13.5%", "13.5%",
"13.5%", "15%",
"14%", "13.5%", "14.5%", "13%", "14%", "14.5%", "13%", "14%",
"13.5%", "14.5%", "14%", "14.5%", "15%", "13.5%", "14%", "15%", "14%", "14%", "9%", "1
3.7%", "13%", "14.08%"};

```

```

viticulture = new String[] { "conventional", "conventional",
"conventional", "organic", "organic", "sustainable", "sustainable",
"sustainable", "conventional", "conventional", "sustainable",
"sustainable", "conventional", "
sustainable", "sustainable", "sustainable",
"sustainable", "sustainable", "sustainable", "conventional", "organic", "convent
ional",

```

```

"conventional","conventional","organic","conventional","organic",
"conventional","conventional","conventional","conventional"};

    geographical_indication = new String[] { "PDO Patras", " PDO
Patras", "PDO Santorini", "PGI Pangeon", "PGI Pangeon", "PGI Epanomi", "PGI
Epanomi",
        "PGI Epanomi", "PDO Samos", "PDO Samos"," PDO Amynteo",
        "PGI Florina","PGI Florina","PGI Florina","PGI Florina",
        "PGI Florina","PGI Florina","PDO Amynteo","PGI
Florina","PGI Krania","PGI Attiki","PGI Drama","PGI Letrina"
        ,"PGI Drama","PDO Muscat of Cephalonia","PGI Valley of
Atalanti","PDO Nemea","PDO Santorini","PGI Peloponnese","PGI Drama","PDO
Nemea"};

    flag = new int[] { R.drawable.mavrodafni_2,
R.drawable.mavrodaphne_reserve_2,
        R.drawable.santorini_vinsanto_2,
R.drawable.ktima_biblia_chora_merlot_2,
        R.drawable.areti_red_2,
R.drawable.domaine_gerovassiliou_vioignier_2,
R.drawable.domaine_gerovassiliou_evangelo_2,
        R.drawable.domaine_gerovassiliou_syrah_2,
R.drawable.samos_anthemis_2, R.drawable.samos_nectar_2007_2
        ,
R.drawable.alpha_xinomavro_merlot,R.drawable.alpha_syrah_2,

R.drawable.alpha_estate_rose_2,R.drawable.alpha_syrah_single_vineyard_turtl
es_2,R.drawable.alpha_estate_blend_2006_2,R.drawable.alpha_estate_white_sau
vignon_blanc_2
        ,
R.drawable.alpha_one_2,R.drawable.alpha_xinomavro_single_vineyard_hedgehog_
2,

R.drawable.alpha_estate_utopia_tannat_3,R.drawable.domaine_katsaros_red_2,R
.drawable.oenotria_land_cabernet_sauvignon_agioritiko_2,
        R.drawable.chateau_julia_merlot_2,

R.drawable.domaine_mercuri_cava_2003,R.drawable.emphasis_syrah_2,R.drawabl
e.stalactite_2,R.drawable.kapnias_red_2,

R.drawable.terroir,R.drawable.test_image,R.drawable.megas_oenos_2,R.drawabl
e.techni_alipias_white_2,R.drawable.gaia_estate_2};

    // Locate the ListView in listview_main.xml
    list = (ListView) findViewById(R.id.listview);

    for (int i = 0; i < sweetness.length; i++)
    {
        Details wp = new
Details(name[i],sweetness[i],carbon_dioxide[i],
wine_color[i],wine_type[i],vintage[i],varietal_makeup[i],alcohol[i],viticul
ture[i],geographical_indication[i],flag[i]);
        // Binds all strings into an array
        arraylist.add(wp);
    }

```

```

    }

    // Pass results to ListViewAdapter Class
    adapter = new ListViewAdapter(this, arraylist);

    // Binds the Adapter to the ListView
    list.setAdapter(adapter);

    spinner = (Spinner) findViewById(R.id.search_spinner);
    ArrayAdapter adapter1 = ArrayAdapter.createFromResource(
        this,
        R.array.color_of_wines_array,R.layout.spinner_layout);

    adapter1.setDropDownViewResource(R.layout.spinner_layout);

    spinner.setAdapter(adapter1);

    spinner_sweetness = (Spinner) findViewById(R.id.sweetness_spinner);

    ArrayAdapter adapter2 = ArrayAdapter.createFromResource(
        this,
        R.array.sweetness_of_wine,R.layout.sweetness_spinner_layout);

    adapter2.setDropDownViewResource(R.layout.sweetness_spinner_layout);

    spinner_sweetness.setAdapter(adapter2);

    // Locate the EditText in listview_main.xml
    editsearch = (EditText) findViewById(R.id.search);

    // Capture Text in EditText
    editsearch.addTextChangedListener(new TextWatcher() {

        @Override
        public void afterTextChanged(Editable arg0) {
            // TODO Auto-generated method stub

            String text =
editsearch.getText().toString().toLowerCase(Locale.getDefault());

            String spinner_text = spinner.getSelectedItem().toString();

            String spinner_text_sweet =
spinner_sweetness.getSelectedItem().toString();

```

```

        adapter.filter(text, spinner_text, spinner_text_sweet);
    }

    @Override
    public void beforeTextChanged(CharSequence arg0, int arg1,
        int arg2, int arg3) {
        // TODO Auto-generated method stub
    }

    @Override
    public void onTextChanged(CharSequence arg0, int arg1, int
arg2,
        int arg3) {
        // TODO Auto-generated method stub
    }
});
    }
}
}

```

Η SearchScreen είναι η κλάση στην οποία ο χρήστης μπορεί να ψάξει ανάμεσα στα κρασιά χρησιμοποιώντας και κάποια εργαλεία όπως τα δύο spinners τα οποία αφορούν το χρώμα και τη γεύση του κρασιού καθώς και editText όπου μπορεί να ψάξει το κρασί γράφοντας το όνομά του . Τα στοιχεία των κρασιών είναι και πάλι αποθηκευμένα στατικά σε πίνακες και εμφανίζονται σε λίστα μέσω τις ListViewAdapter η οποία είναι ίδια με την ListViewAdapterTwo εκτός μερικών μικρών διαφορών.

3.2.4 Wineries

```

package com.example.vagelis_pc.myapplication;
import android.app.Activity;
import android.location.Location;
import android.support.v7.app.ActionBarActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;

import com.google.android.gms.maps.CameraUpdate;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.LocationSource;
import com.google.android.gms.maps.MapFragment;

import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;

```

```

import android.widget.Toast;

import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.CameraPosition;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.maps.model.Marker;

import java.util.ArrayList;
import java.util.HashMap;

public class NearbyScreen extends Activity {

    static final LatLng location = new LatLng(37.59 , 23.44);
    private GoogleMap mMap;
    private HashMap<Marker, MyMarker> mMarkersHashMap;
    private ArrayList<MyMarker> mMyMarkersArray = new
ArrayList<MyMarker>();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.nearby_layout);
        mMarkersHashMap = new HashMap<Marker, MyMarker>();

        mMyMarkersArray.add(new MyMarker("Boutari (Goumenissa) - Domaine
Filiria", "wine", "The Filiria Domaine, Boutari's privately owned vineyard,
adheres to organic viticulture techniques and was planted in 1991. Today it
is a model vineyard, both by local and Greek standards. The majority of the
land is planted with the leading varieties of the area - Xinomavro and
Negoska - as well as with international varieties, such as Chardonnay,
Syrah, Merlot and Cabernet Sauvignon. ", Double.parseDouble("40.951625"),
Double.parseDouble("22.449231")));
        mMyMarkersArray.add(new MyMarker("Domaine Gerovassiliou",
"ktima_gerovasileiou", "Domaine Gerovassiliou is located in Epanomi, a few
kilometres southeast of Thessaloniki, and was established by Vangelis
Gerovassiliou. After finishing his studies at the Agricultural School of
Aristotle University of Thessaloniki, Vangelis Gerovassiliou specialised in
oenology and viticulture at the University of Bordeaux. All Domaine
Gerovassiliou wines are produced from grapes grown in its own vineyards and
are Epanomi Regional Wines. It is no wonder that they have received
international awards and distinctions as a result of their good quality and
timeless consistency", Double.parseDouble("40.450943"),
Double.parseDouble("22.924872")));
        mMyMarkersArray.add(new MyMarker("Gaia Wines (Nemea)",
"gaia_wines", "Gaia Wines (Nemea) began production in 1994 with 9,800
bottles of Thalassitis white wine, which quickly became established in
Greece and abroad as an example of unique Santorini wines. Today, Gaia
operates two wineries: one in Koutsi, Nemea, which is the heart of the
company, and one in Exo Gonia, Santorini. About 40% of its production is
exported, especially to Europe, North America, Australia and Brazil. Gaia's
goal has been and still remains today to expand and highlight the quality
characteristics of the native Greek grape varieties and to achieve their
recognition in international markets. By expanding the potential of the
Nemea Agiorgitiko and the Santorini Assyrtiko over the years, Gaia has

```

created a line of wines that have earned numerous international distinctions and have won over both Greek and foreign consumers.",
Double.parseDouble("37.855094"), Double.parseDouble("22.667544")));
mMyMarkersArray.add(new MyMarker("Alexakis", "alexakis", "The Alexakis family is one of the most important representatives of Cretan wine. It owns the largest private winery in Crete, with ultra modern equipment, while, over its long course, it has worked with almost all the growers on the island. The Alexakis wines draw on a deep knowledge of the Cretan vineyards (varieties, soil, local weather conditions and viticulture); long years of collaboration have allowed the company to choose the most suitable vineyards and the best growers, to produce wines of the highest quality under its scientific guidance. Much of their wine is exported (to Russia, USA) - its major oenological ambassadors are the "Alexakis-Vidiano", "Alexakis-Kotsifali-Syrah" and "Alexakis-Dafni".",
Double.parseDouble("35.298699"), Double.parseDouble("25.115608")));

mMyMarkersArray.add(new MyMarker("Tsantali (Metoxi Chromitsa Mount Athos)", "tsantali", "The Tsantali family has been creating wines and distilling tsipouro and ouzo since 1890. The family's philosophy focuses on the creation of quality wines and distillates which respect the age-old tradition of the wine growing regions. With the rebirth of historical vineyards and varieties, with the adoption of biological and integrated viticulture, with a strong viticultural and oenological team and a complete sales force with exports in over 45 countries, the Tsantali Winery is the international leader in the effort of portraying the best of wines and distillates from Greece.",
Double.parseDouble("40.153712"),
Double.parseDouble("24.331504")));

mMyMarkersArray.add(new MyMarker("Wine Art Estate",
"wine_art_estate", "The \"Wine Art\" Estate is owned by two engineers, Giannis Papadopoulos and Giannis Kalaitzidis. In 1993, Giannis Papadopoulos, out of personal interest and care, undertook to revive an old family vineyard in Mikrochori, Drama. Two years later, the involvement of both men with wine brought love and the need for creativity. The \"Wine Art\" Estate was established and the release of their first wines on the market followed, with the Techni Alipias line of white, rosé and red. In 1999, Idisma Drios Chardonnay launched the monovarietal Idisma Drios line of estate wines to which Assyrtiko, Merlot and Syrah wines were gradually added. In exceptionally good years, Nebbiolo wine was released in a limited number of bottles.",
Double.parseDouble("41.102876"),
Double.parseDouble("24.177424")));

mMyMarkersArray.add(new MyMarker("Domaine Skouras", "skouras", "The Domaine Skouras was established in 1986. Over the years, it has created a disciplined, innovative and serious wine-producing presence. The company owns two wineries in the Peloponnese, in the well-known region of Nemea. One is located in the village of Gymno, where the estate has privately owned vineyards growing the Agiorgitiko variety, and the other, the principal winery, is on the outskirts of Malandreni, where there is great grape-growing potential. This winery is a modern, vibrant facility that is open to visitors and has a production capacity of 10,000 hl. Grapes in this region are mostly planted on slopes and hills, at altitudes ranging from 350 m to 100 m in rather poor but varied soils - clay, rocky and granular - and in unique terroirs that emerge in the regional landscape. The Domaine Skouras and its associates mostly cultivate the local varieties of Agiorgitiko, Moschofilero, Rodhitis and Kydonitsa along with the foreign

varieties of Viognier, Chardonnay, Merlot, Cabernet Franc and Syrah.",
Double.parseDouble("37.689877"), Double.parseDouble("22.654758")));

mMyMarkersArray.add(new MyMarker("Domaine Sigalas",
"domaine_sigalas","Sigalas Estate was established in 1991. The winery was
initially housed in the traditional "canava" (underground wine cellar) of
the family of Paris Sigalas. In 1998 a new vinification, bottling and aging
unit was built in a privately owned area and put into operation. Continuous
modernization and expansion investments shaped the current production
infrastructure, able to process and bottle about 300,000 bottles per
year.Sigalas Estate exports wine to Germany, France, Belgium, Austria, the
Netherlands, the United Kingdom, Cyprus, Switzerland, Poland, Japan, Hong
Kong, Sweden, Russia, the U.S., Canada and Australia.",
Double.parseDouble("36.471397"), Double.parseDouble("25.394157")));

mMyMarkersArray.add(new MyMarker("Estate Papaioannou",
"estate_papaioanou","Thanasis Papaioannou, the founder of Estate
Papaioannou, is one of the first Greek producers who contributed to and
promoted the idea that wine should keep its special features associated
with the variety, the position of the vineyard, and the procedures followed
by the winegrower - producer in favour of the biological balance. Estate
Papaioannou has privately owned vineyards in various areas of Nemea. In
general, it employs organic methods and grows different grape varieties,
based on the composition of the soil in each position",
Double.parseDouble("37.807016"), Double.parseDouble("22.706712")));

mMyMarkersArray.add(new MyMarker("Domaine Hatzimichalis
(Atalanti)", "domaine_hatximixalis","Domaine Hatzimichalis, since its
establishment, in 1973, starring with passion and knowledge at the rebirth
of Greek wine and the revival of viticulture in the region of Lokrida. Its
history is closely tied to the region of Atalanti, which has been chosen by
Dimitris Hadjimichalis for the planting of the first 9 hectares of Domaine's
vineyard.This choice was based on an ancient wine tradition, but mostly at
its unique mesoklima, characterized by the physical cooling air currents,
generated in winter and summer from Parnassos mountain to the sea and vice
versa. These currents form ideal conditions for high quality
viticulture.Today, the proprietary vineyard, planted with international and
Greek varieties, reaches 220 hectares. Most of them lie on the slopes of
the valley. This diversity enables the creation of a broad collection of
wines with distinctive style, which fully express the unique terroir of the
region.", Double.parseDouble("38.673641"),
Double.parseDouble("23.023365")));

mMyMarkersArray.add(new MyMarker("Ktima Pavlidis",
"ktima_pavlidis","Ktima Pavlidis is situated in the region of Drama, in
northern Greece, in a narrow valley surrounded amphitheatrically by three
big mountains (Falakro, Menikio, and Pangeo); in the middle of them, the
Aegean Sea. This geographical modification provides our vineyards with a
special mesoclimate, with cool winds during the night that prolong the
ripening period and give the opportunity to berries to reach the optimum
maturity level slowly. ", Double.parseDouble("41.150186"),
Double.parseDouble("24.146432")));

mMyMarkersArray.add(new MyMarker("Domaine Costa Lazaridi",
"domaine_costa_lazaridi","Domaine Costa Lazaridis was founded in Adriani,
Drama, in 1992. It is the extension of a family-owned winery that began to
make small quantities of wine in the early 1990s.Starting with a 10-hectare

vineyard, it now cultivates more than 230 hectares of privately owned vineyards with native Greek and other international varieties, maintaining low production to ensure the best possible grape quality. The winery and its 15,000 m² covered facilities feature exceptionally modern vinification equipment, areas for receiving and hosting visitors as well as all the required equipment for staging conferences and other events.",
Double.parseDouble("41.131126"), Double.parseDouble("24.259754")));

mMyMarkersArray.add(new MyMarker("Oenotria Land Costa Lazaridi", "costa_lazaridi", "The Oenotria Land Costa Lazaridi winery began operations in 2006 in Kapandriti, Attica. It consists of a complex of facilities about 10,500 m² in size, just half an hour from the centre of Athens, with many operating capabilities that include a modern, fully equipped winery, aging cellars and a number of other spaces. Among these are museums with collections of objects, tools and machines used in winemaking and distilling from the early 17th century to today and a number of indoor and outdoor areas for hosting events. The Oenotria Land vineyards, whose development began in 2000, consist of 20 privately owned hectares of land on the shores of Lake Marathonas and where Greek and foreign red grape varieties are organically grown.", Double.parseDouble("41.131134"), Double.parseDouble("24.259797")));

mMyMarkersArray.add(new MyMarker("Union of Vinicultural Cooperatives of Samos (EOSS)", "samos", "The Union of Vinicultural Cooperatives of Samos (EOSS) was founded in 1934 with a membership of 26 local cooperatives representing all the grape growers on the island. From the outset, it offered substantial help to farmers and protected their rights with regard to commercial traders. Since its founding, the EOSS has collected the grapes and produced wine at its two wineries in Malagari and Karlovasi. It then distributes the wine, either in bottled or bulk form, throughout Greece and the rest of the world. The EOSS's exports are very important and amount to 80% of its annual production (about 7,000 tonnes of wine). France alone imports about 60% of the island's production, while other countries that import significant quantities of Samos wines include Australia, Austria, Belgium, Canada, China, Cyprus, the Czech Republic, Denmark, England, Germany, Italy, Japan, the Netherlands, Serbia, Sweden, Switzerland, the USA, and others.", Double.parseDouble("37.750306"), Double.parseDouble("26.961621")));

mMyMarkersArray.add(new MyMarker("Alpha Estate", "alpha_estate", "The Alpha Estate is in the Amynteo grape-growing zone in northwestern Macedonia in an important geomorphologic position with a lovely view of Petres and Vegoritida lakes. It is located between the picturesque village of Nymfaio, the magnificent Pisoderi (Vigla) ski resort and the cosmopolitan ski area of Agios Athanasios (Voras-Kaimaktsalan mountains). The model, privately owned, unified vineyard is at an altitude of 620-680 m near the lakes, with frequent, medium northwesterly winds, relatively low rainfall and sandy soils of low fertility. These elements make this a unique terroir ideal for producing complex wines. The Alpha Estate is the result of a collaboration of two visionaries, second generation vintner Makis Mavridis and oenologist and winemaker Angelos Iatridis. Their goal is to highlight and promote Greek vineyards and Greek wine on an international level.", Double.parseDouble("40.694870"), Double.parseDouble("21.707085")));

mMyMarkersArray.add(new MyMarker("Biblia Chora Estate", "biblia_chora_estate", "The Biblia Chora Estate was established in 2001 by

two well known winemakers, Vassilis Tsaktsarlis and Vangelis Gerovassiliou. The two combined their love for wine and their experience in creating good quality wines to develop a model organic vineyard of 35 hectares at the foot of Mount Pageo in Kokkinochori, Kavala. The area has a long history of grape-growing and winemaking. In antiquity, a grape variety known as "biblia ambelos" was cultivated here for wine production and the region has been named "Biblia Chora" for this reason. The estate vineyards are at an altitude of 400 m, on infertile and rocky soil, and cultivation follows the principles of organic farming, with particular respect for the environment. Other vine-growing tasks (e.g. leaf thinning, harvest) are done using traditional methods and without the use of any particularly high-tech means.", Double.parseDouble("40.813799"), Double.parseDouble("24.000887")));

```
mMyMarkersArray.add(new MyMarker("SantoWines",  
"santo_wines","SantoWines, (Union of Santorini Cooperatives) has been an  
integral part of Santorini's historic winemaking tradition since 1947 and  
continues to flourish with the support of its 1,000 active members.  
SantoWines successfully combines the latest in modern winemaking techniques  
with a strong commitment to the sustainable development of the Santorini  
vineyard as well as to maintaining the longstanding viticultural and  
winemaking traditions of Santorini that have been passed down through  
generations of dedicated members. SantoWines has also been instrumental in  
continuing the cultivation of other traditional artisanal agricultural  
products native to the island. Collectiveness, quality, authenticity and  
sustainable development of vine and wine of Santorini is our mission.",  
Double.parseDouble("36.386659"), Double.parseDouble("25.437232")));
```

```
mMyMarkersArray.add(new MyMarker("Cavino", "cavino","Cavino is  
active in the field of wine and spirits production since 1958. The modern  
winery is located in the northern part of Egialia, Prefecture of Achaia,  
next to private vineyards, where the tradition of wine producing dates from  
antiquity.Highly qualified agriculturalists, specialized in viniculture,  
strictly supervise all of Cavino's vineyards, both private and those owned  
by partners. Continuous care and scientific monitoring guarantee the high  
quality of our wine grapes.The stainless tanks of Cavino's winery with  
cooling plate for keeping a stable temperature during the fermentation, the  
underground air-conditioned aging cellars with small oak barrels from  
France, the high-end automated bottling lines, and the air-conditioned  
rooms where we store the bottled wines complete our company's passion and  
lust to produce high quality wine.", Double.parseDouble("38.250752"),  
Double.parseDouble("22.081105")));
```

```
setUpMap();  
plotMarkers(mMyMarkersArray);
```

```
CameraPosition position = new  
CameraPosition.Builder().target(location).zoom(6).build();
```

```

mMap.animateCamera(CameraUpdateFactory.newCameraPosition(position));
mMap.setMyLocationEnabled(true);

}

private void plotMarkers(ArrayList<MyMarker> markers)
{
    if(markers.size() > 0)
    {
        for (MyMarker myMarker : markers)
        {
            // Create user marker with custom icon and other options
            MarkerOptions markerOption = new
MarkerOptions().position(new LatLng(myMarker.getmLatitude(),
myMarker.getmLongitude()));

markerOption.icon(BitmapDescriptorFactory.fromResource(R.drawable.wine));

            Marker currentMarker = mMap.addMarker(markerOption);
            mMarkersHashMap.put(currentMarker, myMarker);

            mMap.setInfoWindowAdapter(new MarkerInfoWindowAdapter());
        }
    }

private void setUpMap() {
    // Do a null check to confirm that we have not already instantiated
the map.
    if (mMap == null) {
        // Try to obtain the map from the SupportMapFragment.
        mMap = ((MapFragment)
getFragmentManager().findFragmentById(R.id.map)).getMap();

        // Check if we were successful in obtaining the map.

        if (mMap != null) {
            mMap.setOnMarkerClickListener(new
GoogleMap.OnMarkerClickListener() {
                @Override
                public boolean
onMarkerClick(com.google.android.gms.maps.model.Marker marker) {
                    marker.showInfoWindow();
                    return true;
                }
            });
        } else
            Toast.makeText(getApplicationContext(), "Unable to create
Maps", Toast.LENGTH_SHORT).show();
    }
}
}

```

```

private int manageMarkerIcon(String markerIcon)
{
    if (markerIcon.equals("mpoutari_goumenissa"))
        return R.drawable.mpoutari_goumenissa;
    else if (markerIcon.equals("ktima_gerovasileiou"))
        return R.drawable.ktima_gerovasileiou;
    else if (markerIcon.equals("gaia_wines"))
        return R.drawable.gaia_wines;
    else if (markerIcon.equals("alexakis"))
        return R.drawable.alexakis;
    else if (markerIcon.equals("tsantali"))
        return R.drawable.tsantali;
    else if (markerIcon.equals("wine_art_estate"))
        return R.drawable.wine_art_estate;
    else if (markerIcon.equals("skouras"))
        return R.drawable.skouras;
    else if (markerIcon.equals("mpoutari_goumenissa"))
        return R.drawable.mpoutari_goumenissa;
    else if (markerIcon.equals("domaine_sigalas"))
        return R.drawable.domaine_sigalas;
    else if (markerIcon.equals("estate_papaioanou"))
        return R.drawable.estate_papaioanou;
    else if (markerIcon.equals("domaine_hatximixalis"))
        return R.drawable.domaine_hatximixalis;
    else if (markerIcon.equals("ktima_pavlidis"))
        return R.drawable.ktima_pavlidis;
    else if (markerIcon.equals("domaine_costa_lazaridi"))
        return R.drawable.domaine_costa_lazaridi;
    else if (markerIcon.equals("costa_lazaridi"))
        return R.drawable.costa_lazaridi;
    else if (markerIcon.equals("alpha_estate"))
        return R.drawable.alpha_estate;
    else if (markerIcon.equals("samos"))
        return R.drawable.samos;
    else if (markerIcon.equals("biblia_chora_estate"))
        return R.drawable.biblia_chora_estate;
    else if (markerIcon.equals("santo_wines"))
        return R.drawable.santo_wines;
    else if (markerIcon.equals("cavino"))
        return R.drawable.cavino;
    else
        return R.drawable.mpoutari_goumenissa;
}

```

```

public class MarkerInfoWindowAdapter implements
GoogleMap.InfoWindowAdapter
{
    public MarkerInfoWindowAdapter ()
    {

    }

    @Override
    public View getInfoWindow(Marker marker)

```

```

    {
        return null;
    }

    @Override
    public View getInfoContents(Marker marker)
    {
        View v =
getLayoutInflater().inflate(R.layout.infowindow_layout, null);

        MyMarker myMarker = mMarkersHashMap.get(marker);

        ImageView markerIcon = (ImageView)
v.findViewById(R.id.marker_icon);

        TextView markerLabel =
(TextView)v.findViewById(R.id.marker_label);

        TextView anotherLabel =
(TextView)v.findViewById(R.id.another_label);

markerIcon.setImageResource(manageMarkerIcon(myMarker.getmIcon()));

        markerLabel.setText(myMarker.getmLabel());

        anotherLabel.setText(myMarker.getmContext());

        return v;
    }

}
}

```

Σε αυτό το σημείο στο κώδικα γίνεται η χρήση του Google Maps . Κάνοντας `import com.google.android.gms.maps.GoogleMap;` δίνεται η δυνατότητα μέσω της σύνδεσης στο διαδίκτυο να φορτώσει την τοποθεσία και τον χάρτη πάνω στον οποίο εμφανίζονται τα οιοποιεία .Ο χάρτης έχει ρυθμιστεί έτσι ώστε να εστιάζει κατευθείαν στην Ελλάδα. Με τη χρήση της κλάσης `MyMarker` περνιούνται οι πληροφορίες των οιοποιείων και αυτά στη συνέχεια μέσω μια λίστας εμφανίζονται στον χάρτη.

```

package com.example.vagelis_pc.myapplication;

/**
 * Created by vagelis_pc on 5/20/2015.
 */
public class MyMarker {
    private String mLabel;

```

```

private String mIcon;
private Double mLatitude;
private Double mLongitude;
private String mContext;

public MyMarker(String label, String icon, String context, Double
latitude, Double longitude)
{
    this.mLabel = label;
    this.mLatitude = latitude;
    this.mLongitude = longitude;
    this.mIcon = icon;
    this.mContext = context;
}

public String getmContext()
{
    return mContext;
}

public void setmContext(String mContext)
{
    this.mContext = mContext;
}

public String getmLabel()
{
    return mLabel;
}

public void setmLabel(String mLabel)
{
    this.mLabel = mLabel;
}

public String getmIcon()
{
    return mIcon;
}

public void setmIcon(String icon)
{
    this.mIcon = icon;
}

public Double getmLatitude()
{
    return mLatitude;
}

public void setmLatitude(Double mLatitude)
{
    this.mLatitude = mLatitude;
}

public Double getmLongitude()

```



```

    {
        return mLongitude;
    }

    public void setmLongitude(Double mLongitude)
    {
        this.mLongitude = mLongitude;
    }
}

```

3.2.5 SplashScreen

```

package com.example.vagelis_pc.myapplication;

/**
 * Created by vagelis_pc on 5/19/2015.
 */
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;

public class SplashScreen extends Activity{

    private static int SPLASH_SCREEN_DELAY = 3000;

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

        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                // Executed after timer is finished (Opens MainActivity)
                Intent intent = new Intent(SplashScreen.this,
MainActivity.class);
                startActivity(intent);

                // Kills this Activity
                finish();
            }
        }, SPLASH_SCREEN_DELAY);
    }
}

```

Τέλος υπάρχει και η κλάση `SplashScreen` η οποία μας εμφανίζει στη οθόνη για μερικά δευτερόλεπτα (`SPLASH_SCREEN_DELAY`) πρώτου εισέλθουμε στην `MainActivity` και την κεντρική σελίδα την εξής εικόνα :



3.3 Αρχεία .xml

Στο κατάλογο `res/layout` υπάρχουν όλα τα αρχεία `.xml` που είναι υπεύθυνα για την εικόνα της εφαρμογής. Τα αρχεία αυτά είναι γραμμένα σε γλώσσα XML (Extensible Markup Language) . Η XML είναι μία γλώσσα σήμανσης, που περιέχει ένα σύνολο κανόνων για την ηλεκτρονική κωδικοποίηση κειμένων.

3.3.1 Activity_main.xml

```
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:gravity="center|bottom"
    tools:context=".MainActivity"
    android:weightSum="1"
```

```
android:orientation="vertical"  
android:background="@drawable/background2">
```

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="0dp"  
    android:text="@string/Title_WineTracker"  
    android:textSize="32sp"  
    android:layout_gravity="center_horizontal"  
    android:layout_weight="0.50"  
    android:id="@+id/title"  
    android:shadowColor="#ffffdea7"  
    android:textColor="#ffb68b12" />
```

```
<LinearLayout  
    android:layout_width="wrap_content"  
    android:layout_height="0dp"  
    android:orientation="horizontal"  
    android:layout_gravity="bottom"  
    android:layout_weight="0.50"  
    android:weightSum="1">
```

```
<Button  
    android:id="@+id/angry_btn"  
    android:text="Pairings"  
    android:textColor="#FFFFFF"  
    android:textSize="15sp"  
    android:padding="15dp"  
    android:layout_margin="1dp"  
    android:onClick="goToPairings"  
    android:layout_width="108dp"  
    android:layout_height="77dp"  
    android:background="@drawable/button_bg"  
    android:shadowColor="#A8A8A8"  
    android:shadowDx="0"  
    android:shadowDy="0"  
    android:shadowRadius="5"  
    android:layout_gravity="bottom" />
```

```
<Button  
    android:id="@+id/angry_btn2"  
    android:text="Search"  
    android:textColor="#FFFFFF"  
    android:textSize="15sp"  
    android:padding="15dp"  
    android:layout_margin="1dp"  
    android:onClick="goToSearch"  
    android:layout_width="108dp"  
    android:layout_height="77dp"  
    android:background="@drawable/button_bg"  
    android:shadowColor="#A8A8A8"  
    android:shadowDx="0"  
    android:shadowDy="0"  
    android:shadowRadius="5"  
    android:layout_gravity="bottom" />
```

```

<Button
    android:id="@+id/angry_btn3"
    android:text="Wineries"
    android:textColor="#FFFFFF"
    android:textSize="15sp"
    android:padding="15dp"
    android:layout_margin="1dp"
    android:onClick="goToNearby"
    android:layout_width="108dp"
    android:layout_height="77dp"
    android:background="@drawable/button_bg"
    android:shadowColor="#A8A8A8"
    android:shadowDx="0"
    android:shadowDy="0"
    android:shadowRadius="5"
    android:layout_gravity="bottom" />

```

```
</LinearLayout>
```

```
</LinearLayout>
```

Στο activity_main.xml είναι γραμμένος ο κώδικας που αφορά την αρχική σελίδα της εφαρμογής. Αποτελείται από ένα TextView το οποίο είναι και ο τίτλος της εφαρμογής και στη συνέχεια από τρία κουμπιά (Button). Τα κουμπιά είναι κατασκευασμένα στο αρχείο buttonshape.xml .

```

<?xml version="1.0" encoding="utf-8"?>

<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle" >
    <corners
        android:radius="14dp"
    />
    <gradient
        android:angle="45"
        android:centerX="35%"
        android:centerColor="#7A4005"
        android:startColor="#E8E8E8"
        android:endColor="#000000"
        android:type="linear"
    />
    <padding
        android:left="0dp"
        android:top="0dp"
        android:right="0dp"
        android:bottom="0dp"
    />
    <size
        android:width="152dp"
        android:height="80dp"
    />
    <stroke
        android:width="3dp"
        android:color="#878787"

```

```
    />  
</shape>
```

Εδώ ορίζεται το σχήμα και το χρώμα των κουμπιών που χρησιμοποιήθηκαν στην εφαρμογή. Επιπλέον υπάρχουν άλλα δύο αρχεία το `button_bg.xml` και το `buttonselected.xml` μέσω των οποίων καθίσταται δυνατό να ξεχωρίσει ο χρήστης πότε ένα κουμπί είναι επιλεγμένο αλλάζοντας του το χρώμα .

```
<?xml version="1.0" encoding="utf-8"?>  
<selector xmlns:android="http://schemas.android.com/apk/res/android" >  
    <item android:state_pressed="true"  
android:drawable="@drawable/buttonselected"/>  
    <item android:state_focused="true"  
android:drawable="@drawable/buttonselected"/>  
    <item android:drawable="@drawable/buttonshape"/>  
</selector>
```

```
<gradient  
    android:angle="45"  
    android:centerX="35%"  
    android:centerColor="#FFA914"  
    android:startColor="#E8E8E8"  
    android:endColor="#000000"  
    android:type="linear"  
/>
```

3.3.2 listview_main.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"  
    android:background="@drawable/cellar2">  
  
    <EditText  
        android:id="@+id/search"  
        android:layout_width="fill_parent"  
        android:layout_height="wrap_content"  
        android:textColor="#D9D996"  
        android:textSize="25dp"  
        android:layout_below="@+id/search_spinner"  
        android:layout_alignParentLeft="true"  
        android:layout_alignParentStart="true">  
  
        <requestFocus />
```

```

</EditText>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="25dp"
    android:textColor="#D9D996"
    android:gravity="left"
    android:text="@string/wine_color_text"
    android:id="@+id/textView"
    android:layout_below="@+id/sweetness_spinner"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />

<Spinner
    android:id="@+id/search_spinner"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_alignTop="@+id/textView"
    android:layout_toEndOf="@+id/textView"
    android:layout_alignLeft="@+id/sweetness_spinner"
    android:layout_alignStart="@+id/sweetness_spinner" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="25dp"
    android:textColor="#D9D996"
    android:gravity="left"
    android:text="@string/sweetness_of_wine_title"
    android:id="@+id/textView2"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />

<Spinner
    android:id="@+id/sweetness_spinner"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:layout_toEndOf="@+id/textView2"
    android:layout_toRightOf="@+id/textView2" />

<ListView
    android:id="@+id/listview"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:layout_alignRight="@+id/search_spinner"
    android:layout_alignEnd="@+id/search_spinner"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_below="@+id/search" />

</RelativeLayout>

```

Σ'αυτό το αρχείο υλοποιείται η σελίδα του "Search" δηλαδή του ερευνηρίου της εφαρμογής. Αποτελείται από δύο TextView δίπλα στα οποία υπάρχει από ένα spinner στα οποία αναγράφονται οι επιλογές που έχει ο χρήστης ανάμεσα στο χρώμα και στη γεύση του κρασιού που θέλει να ψάξει. Επιπλέον αποτελείται από ένα EditText στο οποίο μπορεί να γράψει την ονομασία του κρασιού και να δοκιμάσει να το ψάξει μέσω ονόματος και τέλος υπάρχει και η ListView η λίστα με τα κρασιά.

```
<?xml version="1.0" encoding="utf-8"?>

<CheckedTextView xmlns:android="http://schemas.android.com/apk/res/android"

    style="?android:attr/spinnerDropDownItemStyle"
    android:gravity="center"
    android:singleLine="true"
    android:layout_width="match_parent"
    android:layout_height="?android:attr/listPreferredItemHeight"
    android:ellipsize="marquee"
    android:textColor="#D9D996"

    android:background="@drawable/button_bg"

    android:checked="false" />
```

Εδώ βρίσκεται ο κώδικας υλοποίησης του spinner .

3.3.3 listview_item.xml

Το listview_item.xml είναι το αρχείο xml στο οποίο γίνεται η υλοποίηση της λίστας μέσω της οποίας ο χρήστης μπορεί να πάρει μερικές πληροφορίες για το κάθε κρασί καθώς και να δει και σε φωτογραφία την εικόνα του.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#70878787"
    >

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



```

        android:text="@string/namelabel"
        android:textColor="#D9D996"

    />

<TextView
    android:id="@+id/name"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_toRightOf="@+id/namelabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/sweetnesslabel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/namelabel"
    android:text="@string/sweetnesslabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/sweetness"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/name"
    android:layout_toRightOf="@+id/sweetnesslabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/wine_colorlabel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/sweetnesslabel"
    android:text="@string/wine_colorlabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/wine_color"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/sweetness"
    android:layout_toRightOf="@+id/wine_colorlabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/wine_typelabel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/wine_colorlabel"
    android:text="@string/wine_typelabel"
    android:textColor="#D9D996"
/>

<TextView

```

```

        android:id="@+id/wine_type"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/wine_color"
        android:layout_toRightOf="@+id/wine_typelabel"
        android:textColor="#D9D996"
    />

<TextView
    android:id="@+id/vintagelabel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/wine_typelabel"
    android:text="@string/vintagelabel"
    android:textColor="#D9D996"
/>

<TextView
    android:id="@+id/vintage"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/wine_type"
    android:layout_toRightOf="@+id/vintagelabel"
    android:textColor="#D9D996"
/>

<TextView
    android:id="@+id/variatal_makeuplabel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/vintagelabel"
    android:text="@string/variatal_makeuplabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/variatal_makeup"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/vintage"
    android:layout_toRightOf="@+id/variatal_makeuplabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/alcohollabel"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/variatal_makeuplabel"
    android:text="@string/alcohollabel"
    android:textColor="#D9D996"/>

<TextView
    android:id="@+id/alcohol"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#D9D996"
    android:layout_below="@+id/variatal_makeup"
    android:layout_toRightOf="@+id/alcohollabel"

```

```

        />
<ImageView
    android:id="@+id/flag"
    android:layout_width="130sp"
    android:layout_height="130sp"
    android:layout_alignParentRight="true"
    android:padding="1dp"
    android:layout_alignBottom="@+id/alcohol" />
</RelativeLayout>

```

Υπάρχουν Textview αντικείμενα τα οποία παίζουν το ρόλο της ετικέτας καθώς και των πληροφοριών για το κάθε κρασί. Τέλος υπάρχει και το ImageView στο οποίο υπάρχει η εικόνα του κρασιού.

3.3.4 nearby_layout.xml

Στο nearby_layout.xml ενσωματώνεται ο χάρτης πάνω στον οποίο έχουν τοποθετηθεί τα οινοποιεία .

```

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.vagelis_pc.myapplication.NearbyScreen"
    tools:ignore="MergeRootFrame" >

    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.MapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</FrameLayout>

```

Εδώ χρησιμοποιείται ένα fragment μέσω του οποίου καθίσταται δυνατή η φόρτωση του χάρτη στην εφαρμογή.

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/marker_layout"
    android:orientation="vertical"

```

```

android:layout_width="match_parent"
android:layout_height="wrap_content">

<LinearLayout
    android:id="@+id/station_info_layout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:weightSum="1">

    <ImageView
        android:id="@+id/marker_icon"
        android:layout_width="150dp"
        android:layout_height="120dp"/>

    <TextView
        android:id="@+id/marker_label"
        android:layout_marginLeft="5dp"
        android:layout_width="38dp"
        android:layout_height="wrap_content"
        android:textStyle="bold"
        android:layout_gravity="center_vertical"
        android:layout_weight="0.63" />

</LinearLayout>

<TextView
    android:id="@+id/another_label"
    android:layout_marginLeft="5dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textStyle="bold"
    android:textSize="12dp"/>

</LinearLayout>

```

Μέσω του infowindow.xml εμφανίζονται στο χρήστη η πληροφορίες σχετικά με τα οιονοποιεία . Αρχικά υπάρχει ένα ImageView και ένα TextView στα οποία εμφανίζεται η εικόνα του ειονοποιείου και δίπλα η ονομασία του. Επιπλέον υπάρχει και άλλο ένα Textview στο οποίο αναγράφεται η περιγραφή του.

Τα υπόλοιπα xml αρχεία τα οποία χρησιμοποιήθηκαν για την δημιουργία της εφαρμογής χρησιμοποιούν τις ίδιες τεχνικές και για χάριν συντομίας δεν συμπεριελήφθησαν στην εργασία.

3.4 AndroidManifest.xml

Το πιο ξεχωριστό αρχείο xml αποτελεί το αρχείο AndroidManifest.xml . Σε αυτό γίνεται μια γενική περιγραφή του τι υπάρχει στην εφαρμογή μας και τι επιτρέπεται να γίνει μέσω αυτής.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.vagelis_pc.myapplication" >

    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission
android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>
    <!-- The following two permissions are not required to use Google Maps
Android API v2, but are recommended. -->
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>

    <uses-feature android:glEsVersion="0x00020000"
android:required="true"/>

    <application
        android:allowBackup="true"
        android:icon="@drawable/icon_new"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".RedMeat"
            android:label="Red Meat"
            android:theme="@style/AppTheme"/>

        <activity android:name=".WhiteMeat"
            android:label="White Meat"
            android:theme="@style/AppTheme"/>

        <activity android:name=".SoftCheese"
            android:label="Soft Cheese"
            android:theme="@style/AppTheme"/>
    </application>
</manifest>
```

```

<activity android:name=".RoastedVegies"
    android:label="Roasted Vegies"
    android:theme="@style/AppTheme"/>

<activity android:name=".HardMeat"
    android:label="Hard Meat"
    android:theme="@style/AppTheme"/>

<activity android:name=".HardCheese"
    android:label="Hard Cheese"
    android:theme="@style/AppTheme"/>

<activity android:name=".GreenVegies"
    android:label="Green Vegies"
    android:theme="@style/AppTheme"/>

<activity android:name=".Fish"
    android:label="Fish"
    android:theme="@style/AppTheme"/>

<activity android:name=".Dessert"
    android:label="Dessert"
    android:theme="@style/AppTheme"/>

<activity android:name=".CuredSmokedMeat"
    android:label="Cured Smoked Meat"
    android:theme="@style/AppTheme"/>

<activity android:name=".Carbs"
    android:label="Carbs"
    android:theme="@style/AppTheme"/>

<activity android:name=".FoodtoWine"
    android:label="Food to Wine"
    android:theme="@style/AppTheme"/>

<activity android:name=".WinetoFood"
    android:label="Wine to Food"
    android:theme="@style/AppTheme"/>

<activity android:name=".PairingsScreen"
    android:label="Pairings Screen"
    android:theme="@style/AppTheme"/>

<activity android:name=".NearbyScreen"
    android:label="Nearby Screen"
    android:theme="@style/AppTheme"/>

<activity android:name=".SearchScreen"
    android:label="Search Screen"
    android:theme="@style/AppTheme"/>

<activity
    android:name=".SplashScreen"
    android:label="@string/app_name"
    android:screenOrientation="portrait"

```

```

        android:theme="@android:style/Theme.Black.NoTitleBar">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name=".NewSingleItemView" >
        </activity>

        <activity android:name=".SingleItemView" >
        </activity>

        <meta-data android:name="com.google.android.maps.v2.API_KEY"
android:value="*****"/>
        </application>

</manifest>

```

Αρχικά δίνεται η δυνατότητα να υπάρξει σύνδεση με το διαδίκτυο έτσι ώστε να γίνει δυνατή η πρόσβαση στα GoogleMaps .Επιπλέον ορίζεται το αρχικό εικονίδιο της εφαρμογής και η ονομασία της. Επίσης αρχεία .java τα οποία κάνουν χρήση extends Activity είναι υποχρεωτικό να οριστούν στο AndroidManifest.xml καθώς επίσης και το αρχείο SplashScreen στο οποίο εμφανίζεται η αρχική εικόνα της εφαρμογής. Τέλος στο αρχείο γράφεται και ο κωδικός που χρησιμοποιείται για την χρήση του GoogleMaps API όπου για χάρη προστασίας έχει αποκρυφθεί.

3.5 build.gradle

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 22
    buildToolsVersion "22.0.1"

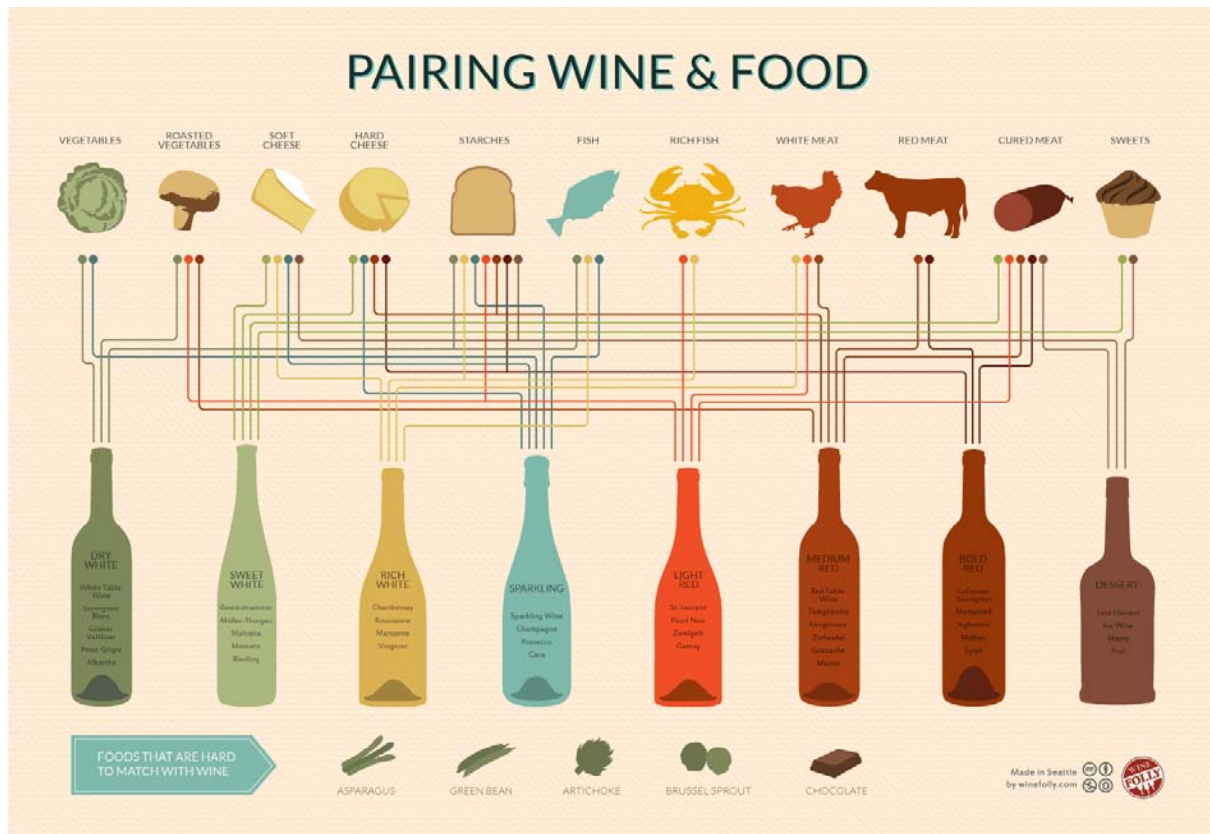
    defaultConfig {
        applicationId "com.example.vagelis_pc.myapplication"
        minSdkVersion 15
        targetSdkVersion 22
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'),
'proguard-rules.pro'
        }
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    compile 'com.android.support:appcompat-v7:22.1.1'
    compile 'com.google.android.gms:play-services:7.3.0'
}
```

Πρόκειται για το αρχείο στο οποίο ορίζεται η ελάχιστη έκδοση στην οποία μπορεί να τρέξει η εφαρμογή καθώς και αυτή στην οποία στοχεύουμε. Επιπλέον δίνεται η δυνατότητα μέσω των dependencies να αναγνωρίσει η εφαρμογή κάποιες βιβλιοθήκες καθώς και τα εργαλεία του play-services μέσα στο οποίο υπάρχει και Google Maps Api.

4.ΣΥΝΔΙΑΣΜΟΙ ΚΡΑΣΙΟΥ ΜΕ ΦΑΓΗΤΟ

Οι συνδυασμοί των κρασιών της εφαρμογής με τα διάφορα είδη φαγητού έγινε μέσω του παρακάτω πίνακα



Στον πίνακα αυτό το ταίριασμα γίνεται με βάση το χρώμα και την γεύση του κάθε κρασιού. Η ίδια μέθοδος ακολουθήθηκε και στο WineTracker.

5. ΠΙΘΑΝΕΣ ΕΠΕΚΤΑΣΕΙΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Το WineTracker από μόνο του αποτελεί μια ολοκληρωμένη εφαρμογή που δίνει αρκετές δυνατότητες στον χρήστη. Παρόλα αυτά θα μπορούσαν να εφαρμοστούν πολλές επεκτάσεις οι οποίες μπορούν να βελτιώσουν τη λειτουργικότητα ακόμη και τις δυνατότητες της εφαρμογής.

Μια πρώτη σκέψη είναι η αντικατάσταση των δεδομένων από στατικά σε δυναμικά. Στα οποία θα έχει τη δυνατότητα ο χρήστης να επεμβαίνει κάνοντας ένα προσωπικό λογαριασμό τον οποίο θα έχει τη δυνατότητα να διαμορφώσει ανάλογα τις προσωπικές του προτιμήσεις. Επιπλέον χρησιμοποιώντας μια συνολική, δυναμική και απομακρυσμένη βάση δεδομένων θα μπορεί να δοθεί η δυνατότητα σε εγγεγραμμένους χρήστες να προσθέσουν τα δικά τους κρασιά αυξάνοντας έτσι και την ποικιλία η οποία παρέχεται μέσω της εφαρμογής.

Επιπλέον μια μελλοντική επέκταση θα μπορούσε να είναι και η δυνατότητα αναρτήσεων σχολίων κάτω από κάθε προϊόν. Με αυτό τον τρόπο θα μπορέσει να δοθεί μια πιο ολοκληρωμένη εικόνα στο χρήστη και η εφαρμογή θα αποκτήσει και μια πιο άμεση σχέση μαζί του.

Σχεδιαστικά δώθηκε πολύ μεγάλη έμφαση στην εφαρμογή, όμως υπάρχουν περιθώρια βελτίωσης ειδικότερα στην οριζόντια προβολή. Καθώς επίσης και στη μετάφραση της και σε άλλες γλώσσες (Ελληνικά, Γερμανικά, κλπ) αφού η μόνη η οποία είναι διαθέσιμη αυτή τη στιγμή είναι η Αγγλική.

Εφόσον υλοποιηθούν κάποιες από τις προαναφερθείς επεκτάσεις άμεσως στόχος είναι να ανέβει η εφαρμογή στο Google Store έτσι ώστε οι να μπορούν να την κατεβάσουν και να την χρησιμοποιήσουν όλοι οι λάτρεις του κρασιού.

6. ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1]Πληροφορίες σχετικά με τα κρασιά
http://www.newwinesofgreece.com/el/map_of_wines/index.html,1711
- [2]Πληροφορίες σχετικά με την χρήση Google Maps
<http://www.rogcg.com/blog/2014/04/20/android-working-with-google-maps-v2-and-custom-markers>
- [3]Πληροφορίες σχετικά με τη χρήση και εγκατάσταση Google Maps
<https://www.youtube.com/user/derekbanas>
- [4] Γενικές πληροφορίες για τη δημιουργία Android εφαρμογής
<http://stackoverflow.com/>
- [5] Γενικές πληροφορίες για το Android
<http://developer.android.com/index.html>
- [6]Πληροφορίες σχετικά με την ανάπτυξη κώδικα
<http://download.androidhive.info/>
- [7]Πληροφορίες για την υλοποίηση λιστών σε Android εφαρμογή
<http://www.androidbegin.com/tutorial/android-search-filter-listview-images-and-texts-tutorial/>
- [8]Σχετικές πληροφορίες για την δημιουργία των κουμπιών
<http://angrytools.com/android/button/>
- [9]Χρήση εικόνων που ενσωματώθηκαν στην εφαρμογή
<https://gr.pinterest.com/>
- [10]Επιπλέον βοήθεια για την δημιουργία των λιστών της εφαρμογής
<http://wptrafficanalyzer.in/blog/listview-with-images-and-text-using-simple-adapter-in-android/>
- [11]Χρήσιμα παραδείγματα και κώδικας για την ανάπτυξη Android εφαρμογής
<https://github.com/>