

2012年度 卒業研究

「共通の趣味、娯楽を持つ友人等との情報交換システムの構築」

学籍番号0840145

氏名 谷口 雄大

指導教員 渡辺 恭人

## 目次

### 1 背景・目的

#### 1-1 背景

#### 1-2 目的

#### 1-3 本論文の構成

### 2 現状と問題点

#### 2-1 ソーシャルネットワークワーキングサービスとは

#### 2-2 ソーシャルネットワークワーキングの現状

#### 2-3 ソーシャルネットワークワーキングの活用性・メリット

#### 2-4 ソーシャルネットワークワーキングサービスの共通する問題点・デメリット

##### 2-4-1 個人情報保護の問題

##### 2-4-2 現状あるソーシャルネットワークワーキングサービスの意見交換

### 3 解決法

#### 3-1 解決すべき問題

#### 3-2 解決方法の提案

### 4-1 設計

#### 4-1-1 設計目標

#### 4-1-2 機能の検討

#### 4-1-3 システム構成

### 4-2 実装

#### 4-2-1 実装環境

#### 4-2-2 クライアント環境

### 5 評価と考察

#### 5-1 評価

5-1-1 機能評価

5-1-2 比較評価

5-2 考察

6 まとめと今後の課題

6-1 まとめ

6-2 今後の課題

参考文献

謝辞

## 1 背景・目的

### 1-1 背景

現在 twitter、Facebook などソーシャルネットワーキングサービスが無数に存在するが、特定の趣味や好きな物の情報について個人的な意見や批判をしあう場がなかなかないのが現状である。もっと専門的であり新鮮な情報が得られ、友人・知人間のコミュニケーションを円滑にする、新しいタイプのソーシャルネットワーキングサービスが必要だと考えた。そこで本研究では、趣味や好きな音楽、ゲーム、お店などを大きな区分でジャンル分けを行う。そして評価が高い場所や逆に評価が低い場所または自分が実際に買ってよかった、もしくは聞いて良かった、人に勧めたいと思える商品やサービスなどを自分のページで公開することでほかの人にも情報を得てもらう。ある一部分の情報収集に特化させることでホームページまたはブログなどより確実に確かで新しい情報を得ることが出来る。商品や店舗の画像も乗せる事ができ、その分野に詳しい人が求めている情報を細かく得ることが出来る。共通ジャンルの多い人とはソーシャルネットワーキング上で「フレンド」登録ができ、商品やサービスについてより深く情報交換することが出来る。また実際に使って改良点や必要な機能を検討し、設計、実装を行っていくことでより実用性の高い web アプリケーションを設計していく。

### 1-2 目的

本研究の目的は上記のソーシャルネットワーキングの中でもさらに音楽に特化したサービスに焦点を当て、現状すでにある音楽共有サービス（Youtube、Myspace 等）とは違う新しい形のソーシャルネットワーキングサービスの目玉となるジャンルを細分化するシステム部分を作ることである。これによって多数派の意見交換、情報共有だけでなく少数派の意見の人々の距離を縮める。

### 1-3 本論文の構成

本論文では、2章で現状あるソーシャルネットワーキングサービスの特徴、そのサービスの問題点を述べる。3章では、2章で判明した問題点の解決法を提案し、検討する。4章では3章で提案、検討した解決法の中から解決法を選択し、システムの設計、実装について述べる。

## 2 現状と問題点

本章では現状あるソーシャルネットワーキングサービスとその問題点を調査し、分析する事で分かった問題点について述べていく。

### 2-1 ソーシャルネットワーキングシステムとは

まずソーシャルという言葉だが、これは社会的な社交的などという意味の英単語である。ソーシャルネットワーキングシステムまた twitter は、すでに実社会で友達関係にあるユーザー同士が情報を交換するのを便利にしたコミュニティサービスであり、会員になると、最初にプロフィールを入力し自分のページを立ち上げる。ただしネット上で知り合いになった友人同士でサービスを利用する事もある。そして個人のページから、メッセージ交換のほか、日記や友達リスト、同好の人々が集まるコミュニティなどをチェック出来るようになっている。国内では、mixi、GREE、個人の立ち上げたソーシャルネットワーキングサービスサイトが稼働しており、どれも無料のサービスであることが多い。最大規模の mixi は、5月9日時点で約62万のユーザーを抱えていた。ソーシャルネットワーキングシステムは、今までのほかのインターネット上のコミュニティサービスと異なり閉鎖的な印象を与える。その理由は、基本的にユーザー登録に招待制を採用しているからだ。前述したソーシャルネットワーキングサービスはどれも、誰でもすぐに会員になれるというサービスではなく基本的に、すでにサービスを利用している友達から招待を受け取らないと参加できない。つまり、ソーシャルネットワーキングサービス上で最初にコミュニケーションするユーザー同士は、すでに顔見知りの仲なのである。そもそもソーシャルネットワーキングサービス、コミュニティサイトは世界中の不特定多数のユーザーとコミュニケーションすることは想定していなく、自分が友達とコミュニケーションするために活用するのが主目的であり、インターネットという便利なものがあるのだから、それを友達同士のやりとりに役立てようという発想からでたアイデアだ。ただそれだけでなく、既存の友達の範疇を越えて、サービスの中で新しい友達関係を作り出せるようにした点もソーシャルネットワーキングサービスの特徴だ。ソーシャルネットワーキングでは、最初に顔の見える友達関係から信頼の輪を作り、その信頼の輪の中から新しい人間関係を築き

上げられるようになっている。見ず知らずのユーザー同士がコミュニケーションする場合でも友達の友達はみな友達だという考え方が基盤にあるからと考えられる。

## 2-2 ソーシャルネットワークサービスの実状

日本では、インターネット初期から数多く存在していた Web で日記をかくサービスやインターネットコミュニティなどを上手く取り込む、または各新聞社やマスコミの記事を取り扱うなどしてポータルサイトとのような役割を持っているものが見られる。企業・教育機関でも内部向けコミュニケーションや、内定者、学校の卒業生の囲い込みなどで色々な用途に使われている。これは主に Facebook で多くみられる。

YouTube

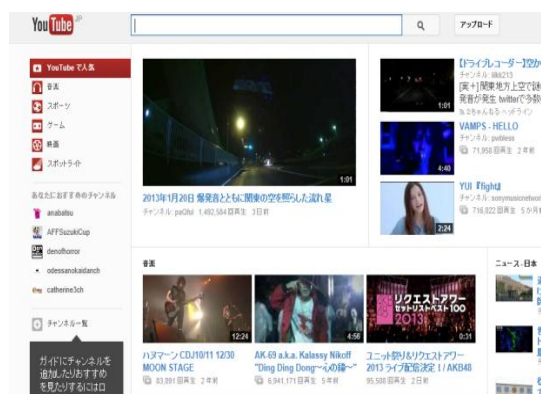


図1 : YouTube

Tumblr.



図2 : Tumblr.

近年では GREE、モバゲーなど様々な形でソーシャルネットワークサービスは普及している。自治体や非営利団体・企業等もそれらを活用しており地域型のサービスも各地で立ち上がっている。最近では YouTube (図1) や Tumblr. (図2) といった画像共有・動画共有サイトが人気になった。それによって日本でもまだ普及していないが AmebaVision のような共有機能が類似しているものが新しくサービス開始されている。総務省の発表では、2006年3月末現在の日本でのソーシャルネットワークサービス利用者数は約716万人に達しており、これは前年の2005年の約6.5倍の数字である。このことからスマートフォンの普及と同時に爆発的に急速に認知度が高まっ

ていることがよくわかる。データによればソーシャルネットワークワーキングサービスの利用者数は2007年3月に1042万人に拡大しており、2009年1月のソーシャルネットワークワーキングサービス会員数は、約7134万人にもなると言われた。例を挙げるとすれば、米国の比較的大きなソーシャルネットワークワーキングシステムの Myspace 公式の発表だと米国の会員数のみでも6000万人を記録していて全世界総ユーザー数は2006年末の時点で約1億2000万人を超える。この時期の Myspace は一ヶ月に約600万人という脅威のペースでユーザーを増やしており、若者に人気のある著名なアーティストたちがこぞって参加したこともこれに拍車をかけていた。ちなみに日本でのサービス開始はかなり遅く2006年の11月にベータ版が開設された。そして近年では日本でも多く利用されている Facebook だが2011年の段階で約8億人もの人が利用する最大級のソーシャルネットワークワーキングサービスになった。前述した自治体や非営利団体・企業等も多く利用している。そしてこれは世界の全体の流れとして今までの閉鎖型のソーシャルネットワークワーキングサービス利用者は減っており、開放型のサービスに流行は流れているようである。日本では Twitter が圧倒的な人気を誇っておりミニブログタイプのサービスも今後はやっていくと考えられている。

### 2-3 ソーシャルネットワークワーキングサービスの活用性メリット

現状様々なソーシャルネットがあり、そのどれに対しても多くの人が登録している。ソーシャルネットワークワーキングサービスが持つ機能の一つにコミュニティ機能がある。これは共通の話題を持った人達はそのテーマについて議論する場である。1つのテーマのコミュニティに何万人もの人が登録しているコミュニティもあり、それについて個々が様々な意見を述べられる。そして考えられるのは何故多くの人がこのサービスを利用するかということだ。ひとつには、短時間で多くの人と繋がりを持つことだ。たった一つのコミュニティでも何万人の人と繋がれる可能性があり、普段の日常生活では年齢層や職業などのせいで限られてしまうこともソーシャルネットワークワーキングサービスの世界では関係なく広い年齢層、職業、様々な境遇の人たちと話すことも出来る。人種、年齢など様々な人が集まるソーシャルネットワークワーキングサービスだからこ

そ多くの考えが聞け、さらに視野が広がり新たな考えが生まれる。自分の趣味や好きな音楽、店舗の情報など、共通のコミュニティを見つけることで多くの人と繋がりを持つことができ、自分のキャパシティ、情報を得られる部分を広げることが出来ると考えられる。また、プロフィールやブログなど個人を特定出来る情報には公開の制限を設けているところもメリットの一つではないだろうか。いくら不特定多数の人と話がしたいと考えていても個人情報をすぐに相手へ伝えるのは危険である。そういった場面ではネット上でなく現実世界の友人にみに公開するなど、情報の公開に制限をつけることでより気軽に安全にソーシャルネットワーキングを利用する事が出来る。また、ほとんどのソーシャルネットワーキングには個人間のメッセージ機能も搭載されているので、コミュニティなどを通して知り合った人と個人的に話をする事も可能である。携帯やPCのメールアドレスを教える事に抵抗があっても、ソーシャルネットワーキング内であれば個人的にメッセージを交わしても良いという場面は多くあるだろう。広く情報を交換でき、なおかつプライバシーを守る事が出来る機能が備わっている事が、ソーシャルネットワーキングの人気でありメリットだ。

## 2-4 ソーシャルネットワーキングサービスの共通する問題点・デメリット

### 2-4-1 個人情報保護の問題

ソーシャルネットワーキングの個人情報保護の問題だが、ネット上の個人情報から実生活へ影響をあたえる問題が多くを占める。たとえばインターネット上の情報からの詐欺行為やストーカーなどである。しかしある程度の問題はサービスを提供する側のシステム作りや現状の法律で解決出来る。たとえば今運営されているソーシャルネットワーキングサービスサイトでは何処にでもあるプライバシー設定などである。これは信頼出来る友人などにしか情報を公開しないように設定でき、自分が好ましく思っていない人物や見られたくない人物に対してブロック（自分の情報を一切見られないようにする。）することができる。この様に利用出来るものはすべて利用し自分の情報は自分で守るというスタイルをとることでほとんどの問題は解決出来るだろう。またおもしろいデータもある。あるインターネットセキュリティーの会社が行ったアンケートでは、アンケートを行った全体の約27%が、ソーシャルネットワークやメールや経由で、不審なリンクやファイルが添付されたメッセージを受け取ったことがあると回答しており、一般的に、ほとんどのユーザーはソーシャルネットワークに脅威



が存在することを認識してなおかつ回答者の約55%がこの問題を認識していると回答している。そしてほぼ同数のユーザーが、電話番号や自宅の住所のような重要な個人情報をソーシャルネットワークのページには乗せないと回答しており回63%は、個人的に知らない人を友達リスト等に追加することはなく68%が知らない人から受けとったリンクをクリックしないようにしているという。これをみて分かるように意外とソーシャルネットワークサービスを利用しているユーザーは対策等を個人的に行っているようである。ほかにも近年盛り上がっている Twitter 等では位置情報を通知する機能が付いているがこれもよほどの事がなければ利用する人は少ないようだ。また複数のユーザー名、パスワードを利用する。自分の個人情報がどのようにどこまで一般に公開されているかしっかり認識する、インターネットブラウザのクッキー情報をブロックする、ソーシャルネットワーキングを利用する際にプロフィールに誕生日、年齢などを公開しない、使わないサービスは退会、削除を行い、情報を残さないなど様々な方法で個人的に情報を守る事が重要である。

#### 2-4-2 現状あるソーシャルネットワーキングサービスの意見交換

今主流のソーシャルネットワーキングサービスは2章で述べた通り意見交換の場としてはとてもメリットが多く使いやすいが、使えば使うほど周りとの関係性から協調性を求めてしまう。それによってありきたりな意見しか言えないソーシャルネットワーキング上の「空気」が出来てしまう。これはソーシャルネットワーキングのジレンマといえる。インターネットを使っても結局は人間と人間の関係性という部分は変えることができない。特に音楽主体のソーシャルネットワーキング等ではその傾向が強く、批判意見の交換場所や大衆的ではない物について語れるような場は無いといってもいいだろう。

### 3 解決法

#### 3-1 解決すべき問題

ここでは2章で述べた現状あるソーシャルネットワーキングサービスの意見交換のシステムの問題を解決していきさらにメリットである部分をどう生かしていくかという事について述べる。

意見交換の問題点としては

- ・匿名性を維持しつつ批判意見の言い合える場がない
- ・ソーシャルネットワーキングサービス上の全体の空気、批判を恐れて意見を言えない
- ・よりマニアックな意見、情報を共有出来るグループを作りにくい

この3点が大きな問題であり、これを解決しつつソーシャルネットワーキングサービスのメリットを活かせるシステムづくりをする必要がある。

#### 3-2 解決方法の提案

ここでは上記した問題の解決策を提案する。今回考えたのが今までのソーシャルネットと違う形のサービスだ。全体のシステムは今までのソーシャルネットワーキングサービスが、自分が探しているコミュニティ、情報へのアクセスの仕方がアナログで本当にその情報を求めている人だけが情報を得られるようなシステムである。

全体の説明をするとまずサービスを利用する人は大まかなジャンルを選択していき自分の必要な情報のあるコミュニティなどを検索していく。ここまでは従来有るソーシャルネットワーキングサービスと変わらない。ほかのサービスと違う部分はジャンル分けの末端部分の情報を自分自身で加えることが出来る部分で、その書き加えた部分をタグ化することで元のジャンル分けよりも詳細に検索することが出来る。そこで本研究ではそのジャンル分け、ジャンル追加、追加したジャンルを検索出来るシステムを構築する。

## 4. 設計と実装

3章に基づき、その一部としてソーシャルネットワーキングサービス内のジャンル分けの機構の設計をし、プロトタイプの実装を行う。

### 4-1 設計

#### 4-1-1 設計目標

設計の目標としては3章で書いた細かなジャンル分けシステムを作る。細分化されたジャンルをユーザーが個々で登録、利用出来るシステムである。

#### 4-1-2 機能の検討

##### 実装機能

今回は主に Java のカードレイアウトを使って実装していく。カードレイアウトとは様々なコンポーネントから構成されているアプレット上で次々と画面を切り替えたいときに使うもので、パネルクラスを併用しそれぞれをカードとして表示するものである。

#### 1. ジャンルの登録システム

これは今回作るシステムでも特徴的な部分である。ユーザーが個人で登録でき、ジャンルの細分化に役立つ。まず大まかなジャンルの選択を行うが今回は大きなカテゴリ5個[J-ROCK J-POP Classic 邦楽 洋楽]と小さなカテゴリ20個[ビジュアル メタル グループビート パンク 女性歌手 男性歌手 女性グループ 男性グループ グループ 古典派 ロマン派 印象派 現代 歌謡曲 演歌 アニメソング 唱歌 ロック ジャズ ブルース]の計25個から選択していき末端の入力部分に自分の登録したいカテゴリを入力する。

#### 2. 登録したジャンルの検索

上記のジャンルの登録システムで登録したジャンルと大きな分類から検索をかけることが出来る。検索部分も登録部分と同じくジャンル計25個から選択し検索する。

### 4-1-3 システム構成

システムの流れは（図1）の通りである。

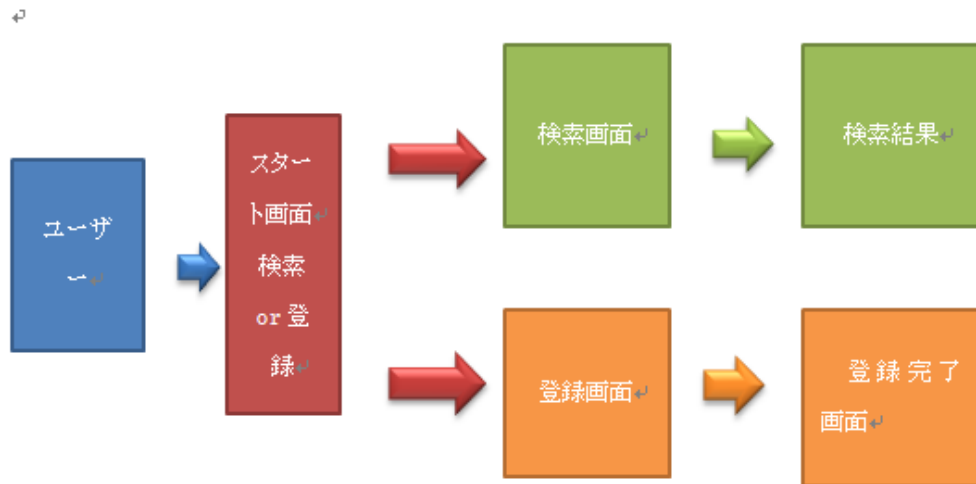


図 1

### 4-2 実装

#### 4-2-1 実装環境

BLUE J: オーストラリアのDeakin大学、英国のKent大学の合同プロジェクトで開発された教育用のJavaの開発環境である。JDK (Javaの言語を利用して開発をするためのコマンドラインの開発ツール一式) がインストールされている環境であればどこでも使うことができる。

#### 4-2-2 クライアント環境

OS : Microsoft WindowsXP

CPU : Intel(R) Core(TM)2 Quad CPU Q9550 @ 2.83GHz

メモリ : 2GB

・カードレイアウト:カードレイアウトの前提を述べる部分

```
/**
 * クラス CardLayoutTester の注釈をここに書きます.
 *
 * @author (谷口雄大)
```

```

* @version (2012/Dec/12)
* @update(2013/Jan/12)
*/

import java.awt.*;
import java.awt.event.*;

public class CardLayoutTester extends Frame implements ActionListener,
ItemListener {

    public static void main(String [ ] args) {
        new CardLayoutTester( );
    }

    CardLayout card;
    Panel toppanel, querypanel, registerpanel;
    Button queryb, registerb, qbackb, rbackb;
    Button querygo, registergo;
    List qtoplist, qsublist, qsubsublist;
    List rtoplist, rsublist, rsubsublist;
    TextField registername;
    TextArea queryresult;
    Label registerstatus;

```

図4-2-2-1 カードレイアウト

- ・カテゴリとサブカテゴリの一覧:ここで登録する25個のジャンルを定義する

```

String [ ] category = { "J-ROCK", "J-POP", "Classic", "邦楽", "洋楽
" };
String [ ] [ ] subcategory = {
    { "ビジュアル", "メタル", "グループ", "ビート", "パンク" },
    { "女性歌手", "男性歌手", "女性グループ", "男性グループ", "グループ
" },

```

```
{ "古典派", "ロマン派", "印象派", "現代" },  
{ "歌謡曲", "演歌", "アニメソング", "唱歌" },  
{ "ロック", "ジャズ", "ブルース" }  
};
```

図4-2-2-2 カテゴリ・サブカテゴリ

- 登録用のデータベース:登録時のデータベースを定義する

```
String [ ][ ][ ] database;  
  
public CardLayoutTester( ) {  
    // ウィンドウとカードの初期化  
    super( "Card Layout Tester" );  
    card = new CardLayout( );  
    setLayout( card );  
    setSize( 500, 500 );  
}
```

図4-2-2-3 登録データベース

- トップ画面:トップ画面の登録、検索ボタンの設定部分

```
toppanel = new Panel( );  
Label toplabel = new Label( "スタート画面" );  
toppanel.add( toplabel );  
Panel buttonpanel = new Panel( );  
buttonpanel.setLayout( new GridLayout( 1, 2, 30, 30 ) );  
queryb = new Button( "検索" );  
queryb.addActionListener( this );  
buttonpanel.add( queryb );  
registerb = new Button( "登録" );  
registerb.addActionListener( this );  
buttonpanel.add( registerb );  
toppanel.add( buttonpanel );
```

```
add( "top", toppanel );
```

図4-2-2-4 トップ画面

- ・ 検索画面のシステム:登録したデータを検索する部分

```
querypanel = new Panel( );
Label querylabel = new Label( "検索指定" );
querypanel.add( querylabel );
// 検索用のパネル
Panel categorypanel = new Panel( );
categorypanel.setLayout( new GridLayout( 1, 3, 30, 30 ) );
// トップカテゴリリスト
qtoplist = new List( 10 );
for ( int i=0; i <category.length; i++ ) {
    qtoplist.add( category[ i ] );
}
qtoplist.addItemListener( this );
categorypanel.add( qtoplist );
// サブリストとサブサブリスト
qsublist = new List( 10 );
categorypanel.add( qsublist );
qsubsublist = new List( 10 );
categorypanel.add( qsubsublist );
querypanel.add( categorypanel );
// 検索をするボタン
querygo = new Button( "検索実行" );
querygo.addActionListener( this );
querypanel.add( querygo );
// トップに戻るボタン
qbackb = new Button( "戻る" );
qbackb.addActionListener( this );
querypanel.add( qbackb );
```

```
add( "query", querypanel );
// 検索結果を表示するラベルとテキストフィールド
querypanel.add( new Label( "検索結果" ) );
queryresult = new TextArea( 5, 40 );
querypanel.add( queryresult );
```

図4-2-2-5 検索画面

- 登録画面のシステム:25個のジャンルから選択していきデータを登録する部分

```
registerpanel = new Panel( );
Label registerlabel = new Label( "登録情報" );
registerpanel.add( registerlabel );
registername = new TextField( 20 );
registerpanel.add( registername );
// 登録するカテゴリ用のパネル
Panel registersubpanel = new Panel( );
registersubpanel.setLayout( new GridLayout( 1, 3, 30, 30 ) );

// トップカテゴリリスト
rtoplist = new List( 10 );
for ( int i=0; i <category.length; i++ ) {
    rtoplist.add( category[ i ] );
}
rtoplist.addItemListener( this );
// サブリストとサブサブリスト
registersubpanel.add( rtoplist );
rsublist = new List( 10 );
registersubpanel.add( rsublist );
rsubsublist = new List( 10 );
registersubpanel.add( rsubsublist );
registerpanel.add( registersubpanel );
// 登録をするボタン
```



```

registergo = new Button( "登録実行" );
registergo.addActionListener( this );
registerpanel.add( registergo );
// トップに戻るボタン
rbackb = new Button( "戻る" );
rbackb.addActionListener( this );
registerpanel.add( rbackb );
add( "register", registerpanel );

registerstatus = new Label( "登録情報入力後¥n カテゴリとサブカテゴ
リを選択して¥n 登録実行を押します。" );
registerpanel.add( registerstatus );

// ウィンドウとカードの表示
addWindowListener( new MyWindowAdapter( ) );
setVisible( true );
card.show( this, "top" );

// データベースのイニシャライズ
database = new String[ category.length ][][];
for ( int i=0; i < category.length; i++ ) {
    database[ i ] = new String[ subcategory[ i ].length ][];
    for ( int j=0; j < subcategory[ i ].length; j++ ) {
        database[ i ][ j ] = new String[ 10 ]; // 各サブカテゴリ
で10個まで登録可能
    }
}

// ボタンが押されたときの処理
public void actionPerformed( ActionEvent ae ) {
    if ( ae.getSource( ) == queryb ) {
        card.show( this, "query" );
    }
}

```

```

    } else if ( ae.getSource( ) == registerb ) {
        card.show( this, "register" );
        registerstatus.setText( "登録情報入力後¥n カテゴリとサブカテゴ
リを選択して¥n 登録実行を押します。" );
    } else if ( ae.getSource( ) == qbackb || ae.getSource( ) ==
rbackb ) {
        card.show( this, "top" );
    } else if ( ae.getSource( ) == querygo ) {
        // 検索処理
        queryresult.setText( "検索されたものの結果¥n" );
        int sel = qtoplist.getSelectedIndex( );
        int subsel = qsublist.getSelectedIndex( );
        if ( sel < category.length && subsel <
subcategory[ sel ].length ) {
            for ( int i=0; i < database[ sel ][ subsel ].length;
i++ ) {
                if ( database[ sel ][ subsel ][ i ] != null ) {
queryresult.append( database[ sel ][ subsel ][ i ] + "¥n" );
                }
            }
        }
    } else if ( ae.getSource( ) == registergo ) {

```

図4-2-2-6 登録画面のシステム

・登録の処理:データの登録処理の部分

```

        int sel = rtoplist.getSelectedIndex( );
        int subsel = rsublist.getSelectedIndex( );
        if ( sel < category.length && subsel <
subcategory[ sel ].length ) {
            for ( int i=0; i < database[ sel ][ subsel ].length;
i++ ) {
                if ( database[ sel ][ subsel ][ i ] == null ) {
                    database[ sel ][ subsel ][ i ] =

```

```

registername.getText( );
                registerstatus.setText( registername.getText( )
+ "は¥n" +
                category[ sel ] + "カテゴリの" +
subcategory[ sel ][ subsel ] + "の¥n" +
                (i+1) + "番目に登録されました" );
                break;
            }
        }
    }
}

// カテゴリリストが選ばれたときの処理
public void itemStateChanged( ItemEvent ie ) {
    if ( ie.getSource( ) == rtoplist ) {
        int sel = rtoplist.getSelectedIndex( );
        rsublist.removeAll( );
        if ( sel < category.length ) {
            for ( int i=0; i < subcategory[ sel ].length; i++ ) {
                rsublist.add( subcategory[ sel ][ i ] );
            }
        }
    } else if ( ie.getSource( ) == qtoplist ) {
        int sel = qtoplist.getSelectedIndex( );
        qsublist.removeAll( );
        if ( sel < category.length ) {
            for ( int i=0; i < subcategory[ sel ].length; i++ ) {
                qsublist.add( subcategory[ sel ][ i ] );
            }
        }
    }
}
}

```

```
// ×ボタンでウィンドウを消すための処理
class MyWindowAdapter extends WindowAdapter {
    public void windowClosing( WindowEvent we ) {
        setVisible( false );
        System.exit( 0 );
    }
}
}
```

図4-2-2-7 登録処理

## 5. 評価と考察

### 5-1 評価

今回の本研究における評価だが、初期に想定していたソーシャルネットワークにジャンル分けのシステムを組み込んだものは出来ず、完全な実装までは至らなかった。しかしメインのジャンル分けの機能部分に関してはジャンルの登録、検索が一通り動かせる部分まで実装することが出来た。

#### 5-1-2 比較評価

既存のサービスとの比較だが現在主流のソーシャルネットワーキングサービスの Myspace、mixi には見ないシステムを作ることができた。Facebook プロフィール編集画面の趣味を入力出来る部分が自分の作ろうとしていたシステムに近いもので、さらにシンプルで分かり易い。

### 5-2 動作確認

#### 5-2-1 登録・検索スタート画面

手順としてまずはスタート画面でジャンルの検索を行うか登録を行うかを選択する。

(図5-2-1-1)



図5-2-1-1 メニュー画面

#### 5-2-2 ジャンルの登録画面

図5-2-1-1のメニュー画面から登録ボタンを選択することでジャンルの登録画面に移動する。(図5-2-2-1)

ここでは左のリストから大まかなジャンルを選択していく。左端で選択したものからさらに4つのジャンルに絞られ、その中から選択した項目に自分の登録したいジャンルを入力し、登録実行ボタンを押すことで登録が完了する。登録する数、登録するジャンルに制限はなくそれぞれの好きなようにユーザーが登録することが出来る。(図5-2-2-2)

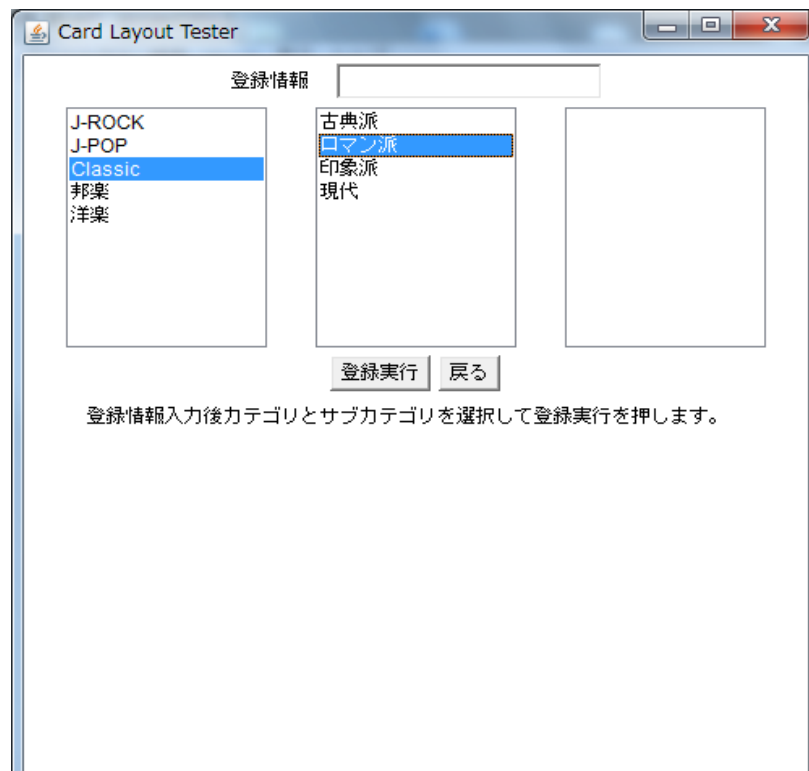


図5-2-2-1 ジャンル登録画面

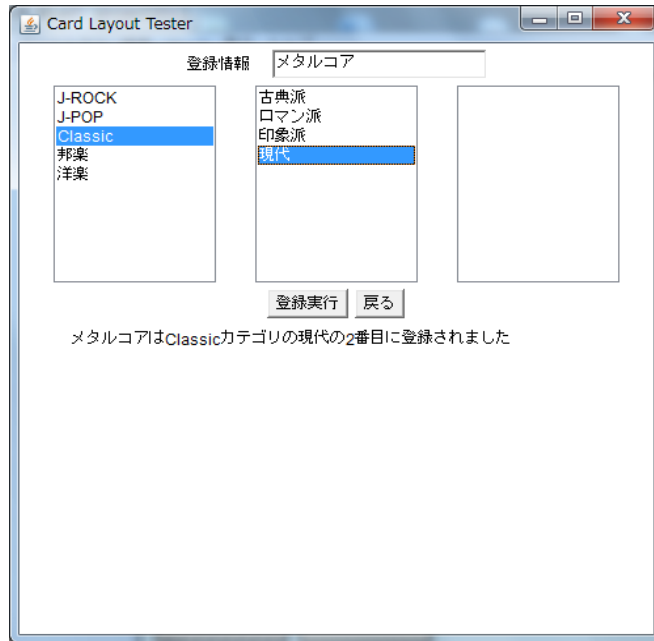
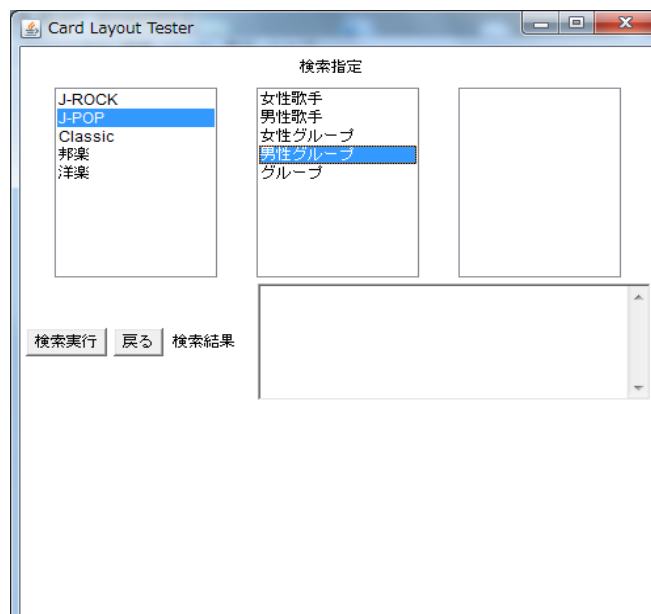


図5-2-2-2 ジャンル登録完了画面

### 5-2-3 ジャンルの検索画面

図5-2-1-1のスタート画面から検索ボタンを押すことで（図5-2-3-1）のジャンル検索画面に移動する。ここでも5-2-2節のジャンル登録画面と同じ操作方法で、左端でジャンルを、中央ではアーティストの種類を選択し検索を行う。検索を行うと（図5-2-3-2）にあるように（図5-2-2-2）で登録した結果が画面下部のテキスト欄に表示される。



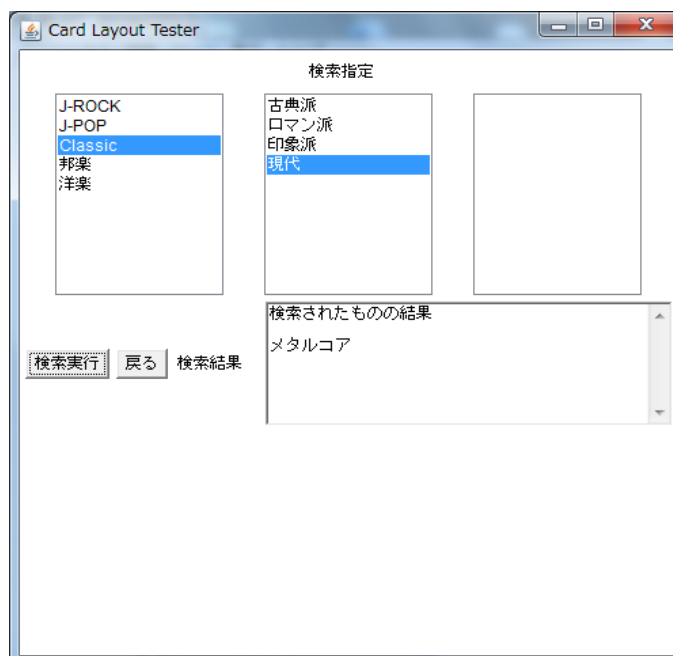


図5-2-3-2 ジャンル検索結果画面

### 5-3 考察

本研究では趣味、音楽等のジャンル分けシステムを使ったソーシャルネットワークキングシステムを目標に進めてきた。元々の予定ではジャンル分けシステムにソーシャルネットワークキングのシステムをあわせて新しい物を完成させる予定であった。そしてジャンル分けされたデータの可視化、共有等出来るシステムを導入する予定であった。しかし最終的にはメインのジャンル分けの部分のシステムしか完成には至らなかった。完成した部分に関しても入力画面のデザインや視覚的に分かりにくく、初めて使う人には少し使いづらいようなデザインになってしまった。だが今回実装出来たシステムは現状のソーシャルネットワークキングサービスにはない様なシステムになっており、もっとも重要な機能部分はできたといえる。

今回手を付けることのできなかったソーシャルネットワークキングのシステム部分は各ジャンルのコミュニティにチャットを実装する予定であったが、もしもそれが実装することが出来れば、ジャンル分けで細分化されたユーザーが集まり、よりその部分の情報に特化した人たちが意見交換、情報交換出来るような自分が理想とする新しい形のサービスができたと考える。



## 6. まとめと今後の課題

### 6-1 まとめ

本研究では今あるソーシャルネットワーキングサービスにはない新しいシステムを構築し、それを利用することで現状あるソーシャルネットワーキングサービスとは違ったサービスを目指した。まず今数多くの人に利用されているソーシャルネットワーキングサービスを分析していきそこから現状の問題点、そしてそれに対する改善点を考えていった。設計に関しては前述した問題点である、匿名性を維持しつつ批判意見の言い合える場がない点、ソーシャルネットワーキングサービス上の全体の空気、批判を恐れて意見を言えない点、よりマニアックな意見、情報を共有出来るグループを作りにくい点の3つに分けた。そしてそれを元にどのようなシステムを作れば問題点が改善でき、なおかつ新しいソーシャルネットワーキングサービスの仕組みを作るかを考察し解決法を考えた。システムの実装ではジャンルの登録、ジャンルの検索の主な機能は実装することが出来たが、それをソーシャルネットワーキングサービスと連動させる、タグ付けによる検索といった部分は実装まで至らなかった。しかしシステム自体は様々な場面に活用出来るようなものになったと考える。

### 6-2 今後の課題

- ・ソーシャルネットワーキングサービスと今回実装したシステムを連動させる  
新たにソーシャルネットワーキングのシステムを作り、それに今回実装したジャンル分けの機能を搭載して一つのサービスにする。
- ・ソーシャルネットワーキングサービスのコミュニティ上のチャット機能の追加  
上記した新しく作ったソーシャルネットワーキングサービスにチャット機能を追加し更なる情報の共有、意見交換等を出来るようにする。
- ・タグ付けによるさらなるジャンルの細分化  
今回実装したプログラムでジャンル分けされたデータをユーザーが個々にタグ付けることができ、更に細かいジャンル分けを行う。
- ・実装したジャンル登録、検索画面のデザインの変更  
今回実装した検索、登録画面のデザインを変更しより使いやすいユーザーインターフェイスに変更する。
- ・タグ付け、ジャンル分けされたデータの視覚化

タグ付け、ジャンル分けされたデータをツリー状に視覚化することで、サービスを利用するユーザーが感覚的に検索、登録サービスを使えるようにする。

## 参考文献

[1]Facebook

<https://ja-jp.facebook.com/> 2012年12月22日

[2]mixi

<http://mixi.jp/> 2012年11月12日

[3]twitter

<http://twitter.com/> 2012年10月10日

[4]Myspace

<https://new.myspace.com/> 2012年10月11日

[5]Java初心者入門講座

<http://sunjava.seesaa.net/category/3643077-1.html> 2013年1月4日

[6]箕原辰夫著 MacでJava! ラトルズ出版 2005年

[7]Javaの道

<http://javaroad.jp/java2sdk.htm> 2012年12月20日

[8]BlueJ Teaching-Java Learning-Java 2012年11月9日

<http://bluej.org/>

[10]CardLayout (カードレイアウト) の基礎

<http://java.keicode.com/lib/swing-layout-cardlayout.php> 2012年10月28日

[11]デビッド・カークパトリック著 日経 BP 社出版 フェイスブック 若き天才の野望 (5億人をつなぐソーシャルネットワークはこう生まれた) 2011年

[12] 山崎秀夫著 ミクシィで何ができるのか？ 青春新書 2007年

## 謝辞

本研究において助言、指導をしてくださった渡辺恭人准教授には大変感謝しております。渡辺准教授には忙しい中、論文の構成や執筆に至るまで親身にご指導頂いたおかげで、遅れていた本研究を完成させる事ができました。完成までの過程には多くの問題点や疑問がありましたが渡辺准教授にアドバイスを細かく頂いたお陰でここまで進める事ができました。また本研究ではプログラミング面では箕原辰夫教授にもご協力いただいて完成まで持つことが出来ました。本当に感謝してもしきれません。

最後に、本研究に関わって下さったすべての方に感謝を述べさせて頂き、謝辞とさせていただきます。ありがとうございました。