

2008 年度卒業研究
「CD データベースを利用した検索サービスの検討」

担当教員：渡辺 恭人

学籍番号 0540068

照井 貴規

概要

音楽のダウンロード販売は普及しているが、現在多くの個人は多数の CD を所有している。しかし、日常生活で CD を購入する場合に、どのアーティストの CD を所有していて、どの CD を所有していないかがわからない場合がある。また、音楽データの入っている PC を持ち歩いているわけではないので、移動先などでは音楽データを見ることができない場合がある。そのような背景のもとに、本研究が進められた。携帯電話から所有している音楽データを検索し、見られるようにすることにメリットがあると考えた。iPod や、その他の携帯型デジタルオーディオプレーヤーを持ち歩いていなくても音楽データを見ることができ、CD の重複購入をすることがなくなる。

音楽データを登録、検索するサービスの設計を経て、実装に取り掛かり、携帯電話から所有している音楽データを検索するシステムを制作することに成功した。

目次

1. 背景と目的	4
1.1 背景	4
1.2 目的	4
1.3 本論文の構成	4
2. 現状と問題点	5
2.1 現状	5
① コンパクトディスクの歴史	5
② アナログレコードの歴史	6
③ アナログディスクと CD の生産数量	6
④ デジタルオーディオプレーヤー	8
⑤ 携帯型オーディオプレーヤーの歴史	9
⑥ 携帯音楽プレーヤーの種類	9
⑦ 2005 年の HDD／メモリ音楽プレーヤーの世界生産	10
⑧ 2005 年のデジタル携帯オーディオ・プレーヤーのブランド別生産シェア	12
⑨ 携帯デジタルオーディオプレーヤーの市場動向	12
⑩ デジタルオーディオデータのメリット・デメリット	13
2.2 問題点	13
3. 決法の検討	14
3.1 解決へのアプローチ	14
3.2 解決法	14
(1) 第一段階	15
(2) 第二段階	15
3.3 モデル	15
(1) 必要な情報の種類の検討	15
(2) 必要な機能とシステム構成	19
(3) 必要なデータベース	20
4. 設計と実装	21
4.1 実装環境	21
4.2 データベースの設計と構築	21

4.3 機能の実装	23
5. 評価	34
5.1 機能評価	34
5.2 考察	38
6. まとめと今後の課題	39
6.1 まとめ	39
6.2 今後の課題	39
参考文献	40
謝辞	41

1. 背景・目的

1.1 背景

音楽のダウンロード販売は普及しているが、現在多くの人は多数の CD を所有している。しかし、日常生活で CD を購入する場合に、どのアーティストの CD を所有していて、どの CD を所有していないかがわからない場合がある。また、音楽データの入っている PC を持ち歩いているわけではないので、移動先などでは音楽データを見ることができない場合がある。

したがって、店内にいるときや、移動先で個人の持っている CD を知りたくても、わからない場合が多く、間違って重複した CD を購入したり、新しい CD をどのように検索するかもわからないことが多く、さらに、好きなアーティストの CD がいつ発売するのかがわからない場合がある。音楽情報が含まれている iPod などの携帯型音楽プレーヤーをいつも携帯しているとは限らないため、携帯電話から Web を利用し、所有している CD などの音楽情報を検索する機能が必要になってくる。

1.2 目的

この研究では、個人の持っている CD のデータベースを作成することによって、使用する人がどの場所においても、持っている CD の音楽情報を把握でき、その音楽情報を表示できるようなシステムを検討する。さらに、持っている CD のアーティストの新しい CD 発売日などの情報を随時インターネットから検索して追加できる機能や、おすすめの CD 情報などを他の人と共有できる機能についても検討したい。そして、所有している CD の情報をデータベース化することによって、重複購入を防ぐだけでなく、その情報とネットの情報を関連付けることによって、利便性の高い CD 購入モデルを実現する。

1.3 本論文の構成

本論文では第 2 章で、現状、問題点、CD などの歴史、生産量などについて述べる。第 3 章で解決法の検討、必要な機能について述べる。第 4 章でデータベース、システムの設計と実装について述べる。第 5 章で、機能評価や動作確認について述べる。第 6 章でまとめと今後の課題について述べる。

2. 現状と問題点

2.1 現状

現在、筆者が所有している CD は、iTunes の中に CD をリッピングしているので、その時に mp3 形式に音楽ファイルを変換している。その時に、大抵の CD は音楽情報(ID3 データ)を含んでいるので iTunes に音楽 CD を取り込んだ場合にアーティスト名、トラック名、トラック番号、アルバム名、発売年などの情報が表示される。しかし、所有している CD の中には、ID3 データの音楽情報を含んでいない CD もあるので、その場合はインターネットを使用し音楽情報を含んでいない CD のタイトルを検索し、個人でアーティスト名、トラック名、トラック番号、アルバム名、発売年などの情報を iTunes の中の音楽データへ入力している。

① コンパクトディスク (CD) の歴史

1981 年のザルツブルク復活音楽祭で、ヘルベルト・フォン・カラヤン財団が、ソニー、フィリップス、ポリグラム・グループと協力し、急遽、CD の生産に踏み切ると発表し、1982 年 10 月、日本ではソニーから最初の CD プレーヤーと CD ソフトが発売された。同時に、レコード店で取扱いが始まったが、当初は「レコードよりも音質がよく、ノイズがないニューメディア」として扱われた。レコードと同じ商品の CD 版として売られ、価格もレコードより 2 割ほど高かった。翌 1983 年の早い時期に、米国およびその他の市場で販売開始となった。

レコード会社が親会社であるオーディオメーカーに配慮してレコード生産を縮小したこともあり、1986 年、販売枚数ベースで CD が LP を追い抜いた。その後、1990 年代にかけて、LP の生産量が減少していった。しかし 90 年代末期以降、ごくわずかな需要や、最近注目されつつあるアナログ音響ブームもあり LP が再生産されるケースが増えてきている。

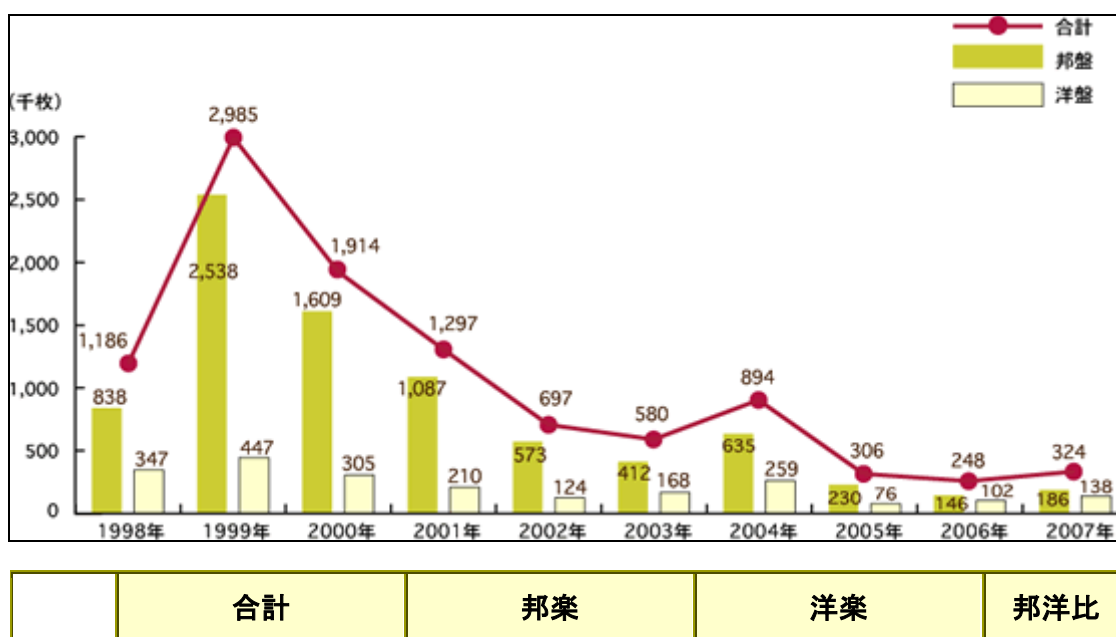
2006 年にはガラス基板の CD が発売された。通常のディスクと較べると、理論的に音質の劣化が起きにくいとされている。一方でコスト面やプレーヤーとの互換性に関して欠点もあげられている。

② アナログレコードの歴史

1900 年代初頭から数十年間アナログレコードが使われてきましたが、突如現れた CD に数年で生産量、販売数が追い抜かれてしまった。8cmCD が発売された 1988 年にはアナログレコードの生産量が激減し、CD 対アナログレコード の対比 が 9 対 1 になった。レコードが CD にその地位を奪われた理由として考えられるのは、通常のレコードの直径が約 30cm に対して、通常の CD は直径 12cm とサイズに大きな差がある。また、音質面では CD の方が上であり、再生回数はレコードの場合針で読み込むために徐々にレコードが針で擦り切れて傷ついたり、反ったりすると聴けなくなってしまう。更にレコード針は、音が悪くなったら交換しなければならない。それに比べ CD はレーザーで読み込むため半永久的に音質が変わらず再生することができる。他にも再生可能時間の問題もある。レコードには収録可能時間の問題から A 面、B 面というようにディスクの両面を使っていたが、CD はレコードよりも収録可能時間が長いため A 面、B 面という概念はない。また、レコードは保存の際、CD と違い室温や湿度もある程度は気にしなくてはならない。つまりレコードが衰退した理由は、サイズ、音質、収録可能時間、再生回数、保存性の 5 点にあると言える。

③ アナログディスクと CD の生産数量

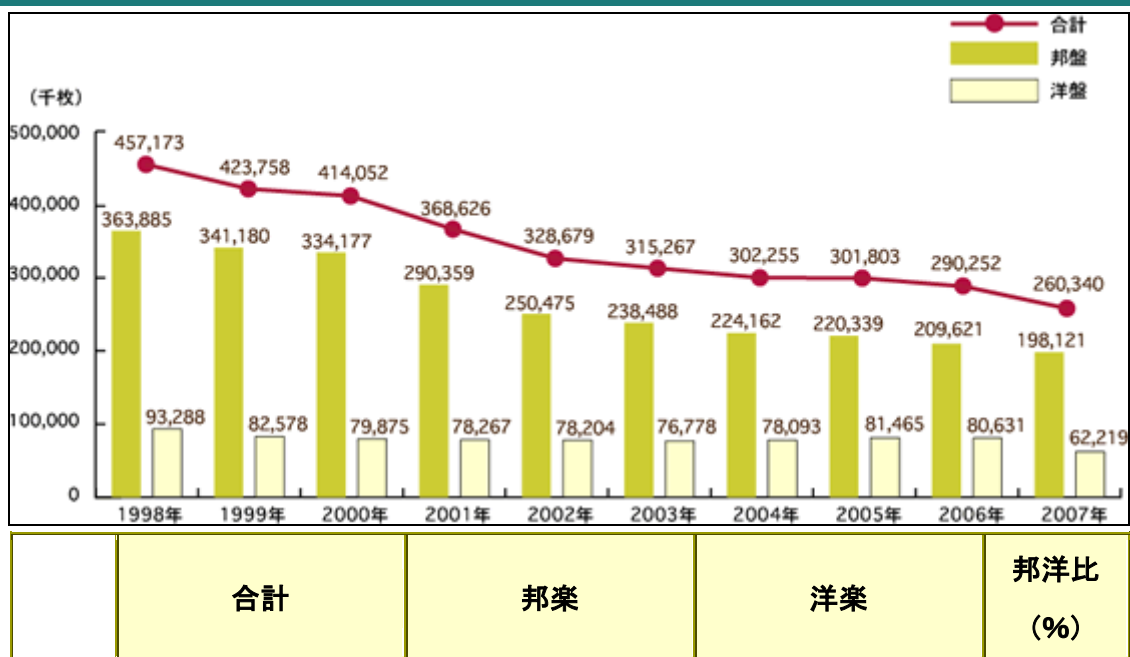
アナログディスク生産数量



							(%)	
	数量(千枚)	前年比(%)	数量(千枚)	前年比(%)	数量(千枚)	前年比(%)	邦楽	洋楽
1998 年	1,186	115	838	104	347	152	71	29
1999 年	2,985	252	2,538	303	447	129	85	15
2000 年	1,914	64	1,609	63	305	68	84	16
2001 年	1,297	68	1,087	68	210	69	84	16
2002 年	697	54	573	53	124	59	82	18
2003 年	580	83	412	72	168	135	71	29
2004 年	894	154	635	154	259	154	71	29
2005 年	306	34	230	36	76	29	75	25
2006 年	248	81	146	64	102	135	59	41
2007 年	324	130	186	127	138	135	57	43

図 2.1.③.1：過去 10 年間のアナログディスク生産実績

CD 生産数量



	数量(千 枚)	前年比 (%)	数量(千 枚)	前年比 (%)	数量(千 枚)	前年比 (%)	邦楽	洋楽
1998 年	457,173	100	363,885	102	93,288	94	80	20
1999 年	423,758	93	341,180	94	82,578	89	81	19
2000 年	414,052	98	334,177	98	79,875	97	81	19
2001 年	368,626	89	290,359	87	78,267	98	79	21
2002 年	328,679	89	250,475	86	78,204	100	76	24
2003 年	315,267	96	238,488	95	76,778	98	76	24
2004 年	302,255	96	224,162	94	78,093	102	74	26
2005 年	301,803	100	220,339	98	81,465	104	73	27
2006 年	290,252	96	209,621	95	80,631	99	72	28
2007 年	260,340	90	198,121	95	62,219	77	76	24

図 2.1.③.2：過去 10 年間の CD 生産実績

④ デジタルオーディオプレーヤー

デジタルオーディオプレーヤーとは、デジタル音楽ファイルを再生可能なオーディオプレーヤーで、特に携帯が可能なものを指す。このうち、MP3 プレーヤーは MP3 のみ再生可能、あるいは MP3 の再生に対応したプレーヤーを意味する。

デジタルオーディオプレーヤーはデジタル方式の音楽ファイルを再生する音響機器で、携帯電話会社の音楽配信サービスやインターネットの音楽販売サイトから購入したり、CD などから直接取り込んだ音楽ファイルを再生するのに用いられる。

デジタルオーディオプレーヤーという言葉は、文字どおりには 1980 年代に登場した CD プレーヤーや 1990 年代に登場した MD (ミニディスク)プレーヤーなども含まれるが、この言葉は通常、2000 年代に本格的に普及しはじめた記録媒体にフラッシュメモリや小型ハードディスクを使用した音楽プレーヤーを指す。2000 年代前半までは音楽ファイルの MPEG による音声圧縮形式方式として MP3 ファイルが使用されることが多く、MP3 プレーヤーという呼称が広く用いられたジャンルであるが、著作権保護などの観点から暗号化技術が進み、MP3 以外の形式の採用が進み、MP3 以外の圧縮形式として、WMA、

AAC、などのコーデックを使用する機種が出現し、2000 年代半ば頃からデジタルオーディオプレーヤーという呼称が用いられるようになった。

⑤ 携帯型オーディオプレーヤーの歴史

初期の携帯型オーディオプレーヤーは 1979 年に登場したウォークマンに代表されるアナログカセットテープベースのものであり、CD 発売後も価格や媒体の大きさなどからテープの置き換えには至らなかった。このため、よりコンパクトかつ携帯機器にふさわしい簡易なメカニズムのシステムの登場が待たれていた。1980 年代後半に高度な高能率符号化による圧縮技術の開発が進み、1992 年には MD（ミニディスク）とデジタルコンパクトカセットが登場している。

1999 年になると、USB への対応を強化した Windows98 Second Edition が発売された。これに合わせるかのように、Diamond は Rio をバージョンアップさせた "Rio500" を発売。当時流行していたスケルトン調のデザインや、Windows/Mac 問わず利用できる点、実際のオーディオ機器に近づいたシンプルなインターフェース、ジョグダイヤルによる快適な操作性などから高く評価され、ヒット商品となった。Rio500 では MP3 プレーヤーの歴史上初めて "DIGITAL AUDIO PLAYER" の文字が液晶下に刻まれている。また、Rio500 は Audible という語学コンテンツ向けフォーマットにも対応しており、音楽だけではない MP3 プレーヤーの可能性を開拓する。楽曲配信サービス Rioport.com にも対応しており、現在の iTunes Music Store のようなサービスが受けられたものの、普及率などの問題があり、展開が早すぎたため失敗した。

⑥ 携帯音楽プレーヤーの種類

アップル 1

iPod（第 1 世代）、iPod（第 3 世代）、iPod（第 4 世代）、iPod mini、iPod photo

アップル 2

U2 iPod、iPod shuffle、iPod mini（第 2 世代）、iPod photo（第 2 世代）、
iPod nano、iPod（第 5 世代）

アップル 3 2006/11/3

iPod（第 5.5 世代）、iPod nano、iPod shuffle

ソニー 1

NW-HD1、NW-HD2、NW-HD3、NW-E99、NW-HD5

ソニー 2 2005/9/12

NW-A1000、NW-A3000

東 芝 2005/8/17

gigabeat G20/G10、gigabeat F10/F20/F60/F40/F11/F21/F41

クリエイティブメディア 2005/5/12

Zen Micro、Creative Zen 20GB、Creative Zen Neeon

その他 2006/9/16

H320/H340、アイリバーH10、m:robe MR-500、Apple iPod + hp、

Apple iPod mini + hp、MS Zune

音楽携帯 2008/11/9

SD-Audio 対応端末

Magic Gate 対応端末

LISMO（リスモ）対応端末、au 端末

900i シリーズ、901i シリーズ、700i シリーズ、901iS シリーズ、N902i、902iS

シリーズ、903i シリーズ

海外音楽携帯 Nokia、Motorola、Sony Ericsson、Samsung

⑦ 2005 年の HDD/メモリ音楽プレーヤの世界生産

米 Apple Computer, Inc.が「iPod」で火を点けた携帯型デジタル音楽プレーヤの世界市場が急拡大し、生産台数は爆発的に伸びている。日経マーケット・アクセスの調査によると、記録媒体にフラッシュ EEPROM やハードディスク装置（HDD）を使ったデジタルオーディオの世界生産台数は、2004 年に 2852 万台、対前年比 114.4%増と 2 倍以上に伸びた。iPod によって急速に注目度が高まってきたデジタルオーディオ市場だが、数年以上前までは記録メディアにフラッシュ EEPROM（半導体メモリ）を使うことから「シリコンオーディオ」などと呼ばれた。そうした中で、Apple 社が大容量の HDD を使った iPod を投入。さらに、同社が 2003 年 4 月から低価格の音楽配信サービス

「iTunes Music Store」を開始すると、世界的なブロードバンド化の波に乗って市場が急拡大した。パソコン周辺機器メーカーも HDD 搭載のデジタルオーディオを相次いで

製品化し、HDD 音楽プレーヤーだけで 2004 年の世界生産台数は 1142 万台にのぼった。
2003 年に比べて 6 倍強の飛躍的な増加である。

一方、HDD 搭載機によってデジタルオーディオへの注目度が高まったことから、小型・軽量・低消費電力が売り物のフラッシュ EEPROM 搭載のデジタルオーディオも需要が急増した。2004 年のメモリ音楽プレーヤーの生産台数は HDD 音楽プレーヤーを上回る 1710 万台となった。これは 2003 年実績の約 15 倍である。

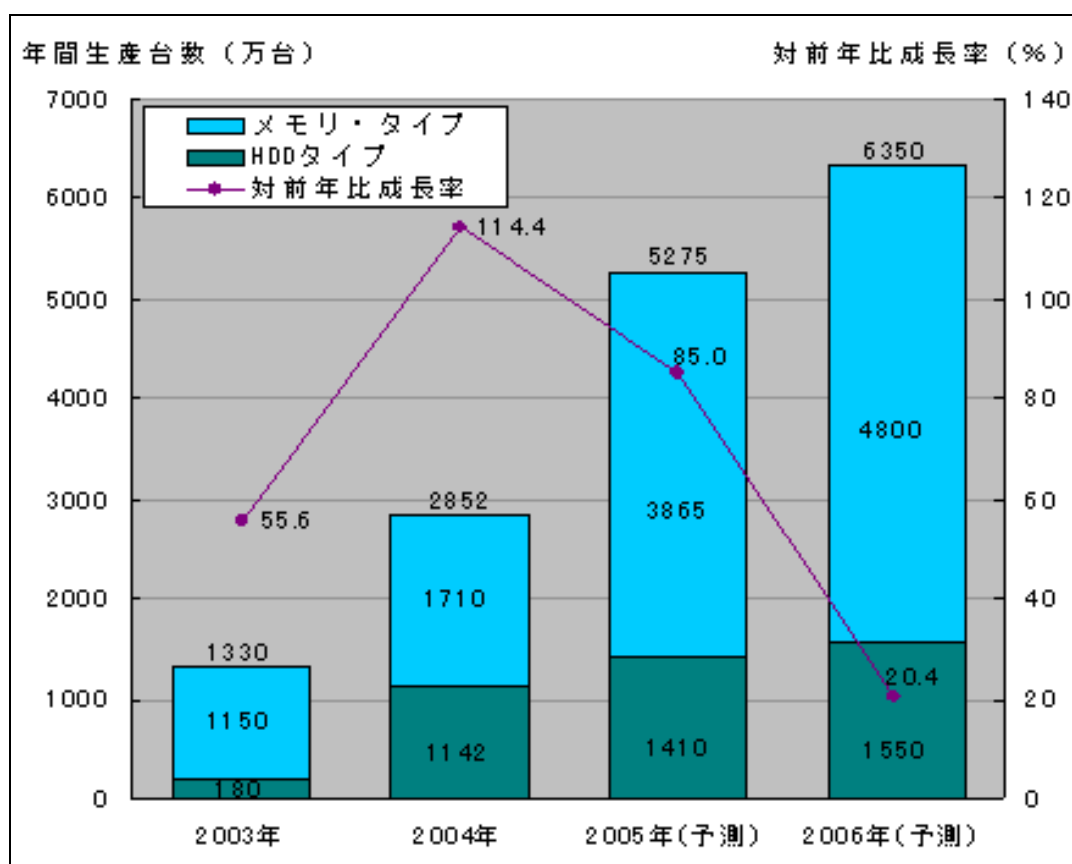


図 2.1.⑦.1 : 世界の携帯型デジタル音楽プレーヤー生産台数（日経 MA 調べ）

⑧ 2005 年のデジタル携帯オーディオプレーヤーのブランド別生産シェア

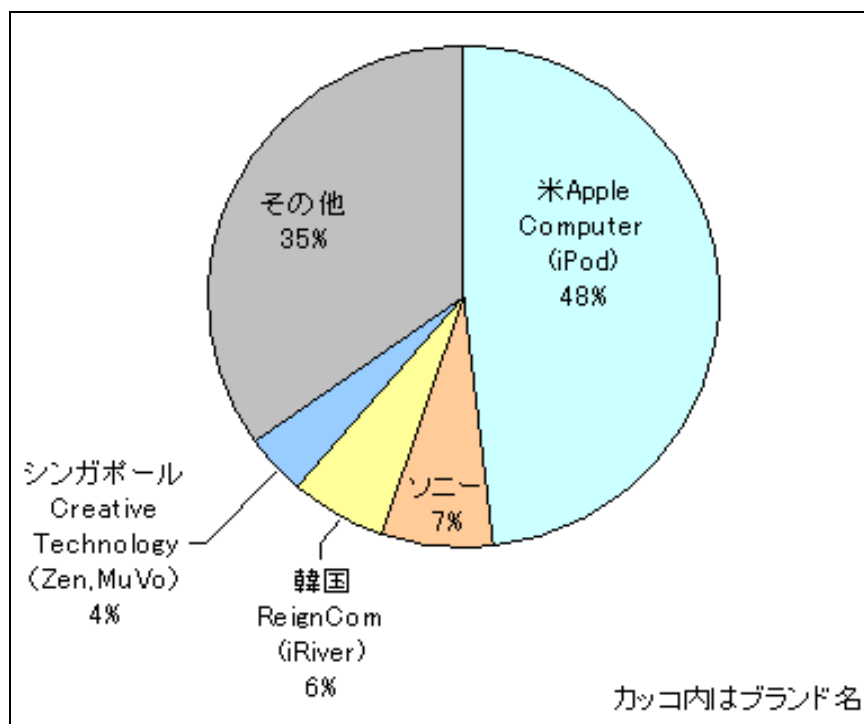


図 2.1.⑧.1 : 2005 年のデジタル携帯オーディオ・プレーヤーのブランド別生産シェア

⑨ 携帯デジタルオーディオプレーヤーの市場動向

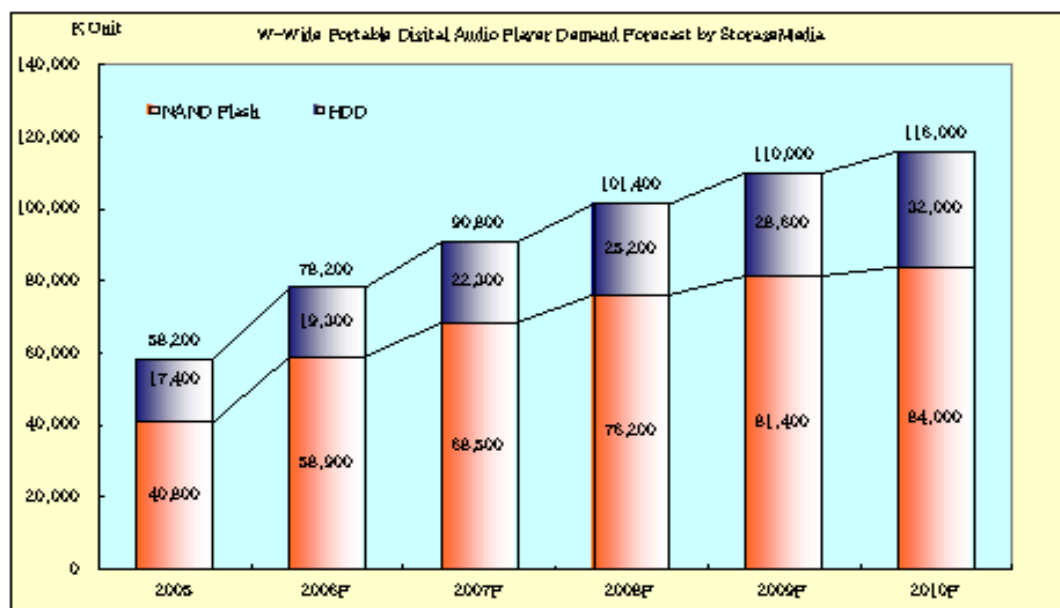


図 2.1.⑨.1 : 2005 年携帯デジタルオーディオプレーヤーのメディア方式別出荷シェア

⑩ デジタルオーディオデータのメリット・デメリット

デジタルオーディオプレーヤーで音楽を聴く方法には iTunes Store などから購入してダウンロードするやり方と、CD を購入したりレンタルしたりしてそこからデジタルデータを取り込むやり方がある。iTunes Store などから曲を購入しダウンロードする方法でのメリットは、1 曲単位で曲を購入しダウンロードすることができるので自分のほしい曲だけをダウンロードすることができ、CD を購入するよりもダウンロードする場合のほうが低コストである。デメリットは、パソコンや、ハードディスクなどにしかデータがないので保存先のデータを消去してしまったり、パソコンの故障などによりデータが消えてしまった場合は曲を失うことになるため、新たに曲をダウンロードしなくてはならない。その場合は CD を購入したり、レンタルする方法よりもコストが掛かってしまうという問題もある。

2.2 問題点

このように CD をたくさん購入したりレンタルしたりするうちに、自分がどの CD の曲を持っているかがわからなくなることで、それによって、間違っただけにもっている曲の CD を購入したり、レンタルしてしまうことがある。このようなことで、無駄な出費をしてしまう。

デジタルデータとして所有している音楽から、必要な情報を抜き出して、いつでもどこでも自分の所有している音楽リストを見られるようになることで、問題が解決される事が期待される。

3. 解決法の検討

この章では、携帯電話での実装に向けて、機能やシステムの構造を検討する。

3.1 解決へのアプローチ

携帯電話の **Web** を利用し、個人の持っている音楽データを検索できるようにすることによって、すでにもっている曲の入った **CD** を購入したり、レンタルしてしまうことがなくなり、無駄な出費をなくすことが出来る。

現状から、iTunes に **CD** を取り込んだときに iTunes のライブラリにアーティスト名、トラック名、トラック番号、アルバム名、発売年などの情報が表示されているので、その情報を使えばデータベースを製作することが可能である。

データベースと **Web** サイトのマッシュアップが出来ていない。また、他人と共有できるサービスも必要である。

3.2 解決法

はじめに、個人の **PC** に入っている曲のリストを自動作成できるようなシステムを制作し、そのデータをデータベースへ送信できる機能を制作する。さらに、曲などを検索できる検索インタフェースも制作する。iTunes に **CD** をリッピングした場合に個人の指定しているフォルダに **MP3** のファイルが入るので、そのときに表示される情報 (**MP3ID3** タグ)を使ってデータベースを制作する。**ID3** タグ (アイディースリータグ、**ID3 tag**) は、**MP3** ファイルの中に「アーティスト、作成年、曲名」などの情報を書き込むための規格である。

データベースは **MySQL** などで作成する。また、データベースと **Web** サイトをマッシュアップする場合はアマゾンなどのサイトを連携して使用できるようにする。そして、必要な情報をメールで知らせるような機能を作成する。出来れば、他人との共有できる機能を作成する。

このような機能が必要な理由は、iTunes で **CD** の曲を取り込むときに使うコンピュータは持ち歩かなかったり、いつも電源が入ってなかつたり、ネットワークにつながってなかつたりすることがある。そこで、曲のリストをリッピングしたファイルから作成し、それをネットワーク上のデータベースに登録することによって、いつでもどこでも個人の曲リストにアクセスすることができるから必要である。また、**Web** サイトとマッシュ

アップすることで自動的にほしい情報が入手することが出来る。そして、そのほかの人と情報を共有することが出来れば、個人以外の人からのおすすめの曲などの情報を得ることが出来る。

(1) 第一段階

- 個人の PC に入っている曲のリスト自動作成
- データのデータベースへの送信と登録
- 曲の検索インタフェース

(2) 第二段階

- データベースと他のサイトとの連携
- CD を持っているアーティストの新曲情報をメール配信

3.3 モデル

(1) 必要な情報の種類の検討

ここでは、いつでもどこでも個人が持っている曲リストにアクセスするために、それぞれの曲が持つ情報とその種類について検討する。

CD から曲をリッピングする際に最もよく利用されていると思われる iTunes を例に、曲情報について調査する。

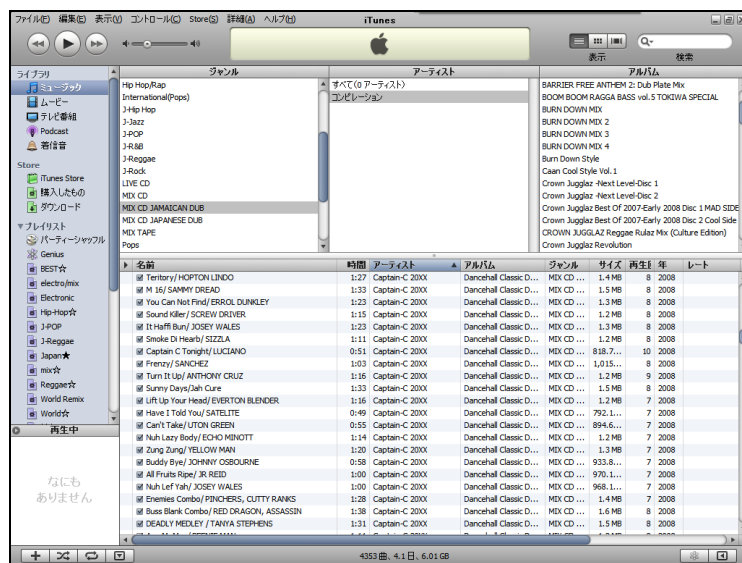


図 3.3.1 : iTunes の利用画面

① iTunes で CD をリッピングする(図 3.3.2)

iTunes を起動して、CD を CD ドライブに挿入すると、iTunes は自動的に CD を読み込み、CD 情報から CDDDB というインターネット上の曲情報データベースにアクセスし、情報を取得する。

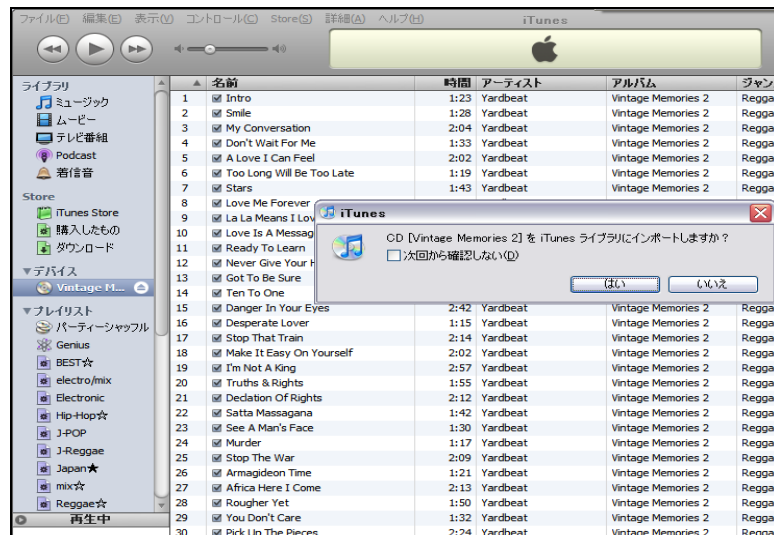


図 3.3.2 : iTunes での CD の情報表示

② iTunes にリッピングしてある CD の詳細情報(図 3.3.3)



図 3.3.3 : iTunes での曲情報表示

- ③ CD によっては ID3 データの情報がないものがあるので、その場合は手打ちで情報を打ち込むことが出来る。ジャンルや音量、その他の設定も行なうことが出来る。

(図 3.3.4)

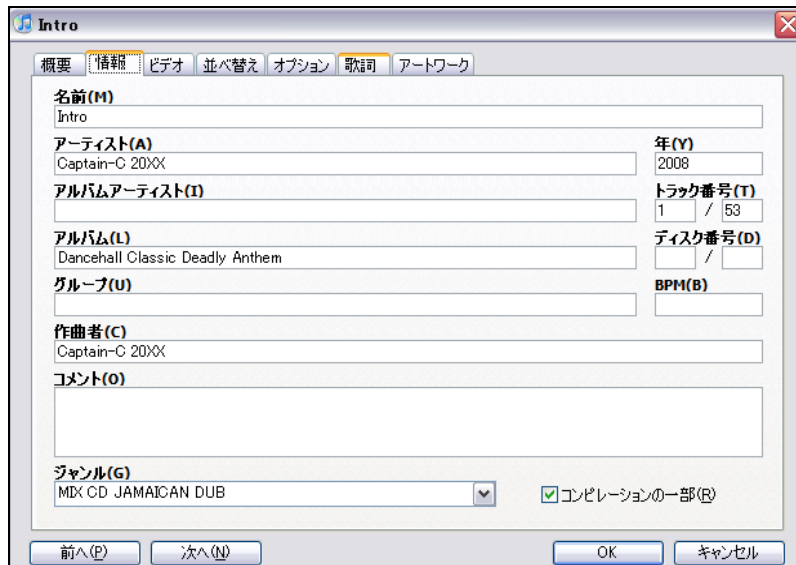


図 3.3.4 : iTunes での曲の詳細情報表示

- ④ リッピングした CD データの保存先

リッピングした曲は、1 曲 1 ファイルでハードディスク等に保存される。通常 iTunes Music というフォルダに保存される。(図 3.3.5)

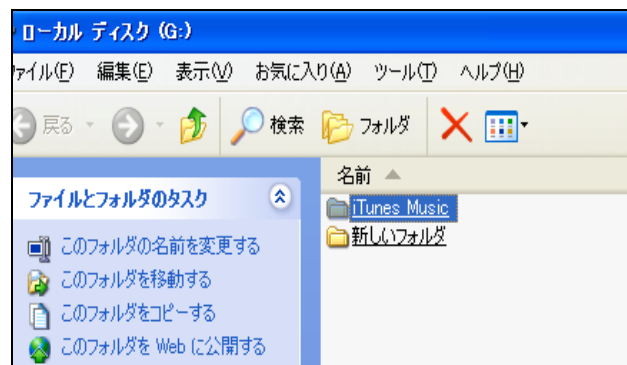


図 3.3.5 : iTunes での曲保存フォルダ

- ⑤ また、リッピングした CD データはアーティストごとに保存される。(図 3.3.6)



図 3.3.6 : iTunes でのアーティストごとの保存フォルダ

- ⑥ アーティストのフォルダには、アルバムごとにフォルダが作成される。(図 3.3.7)

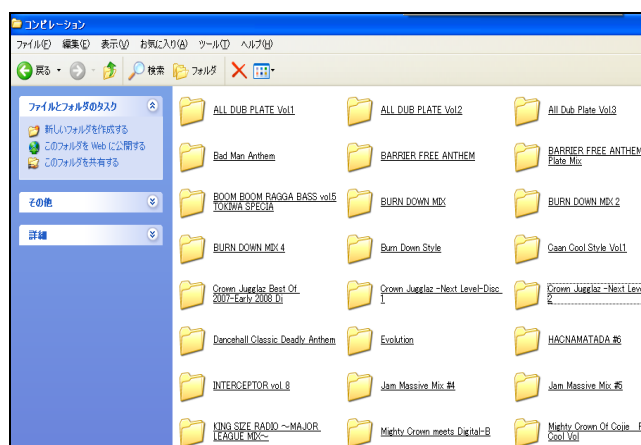


図 3.3.7 : iTunes でのアーティストごとの保存フォルダ詳細

- ⑦ 曲ファイルは、アルバム名のフォルダに保存される。(図 3.3.8)



図 3.3.8 : アルバムごとの保存フォルダ内の曲ファイル

⑧ 曲に含まれている情報(図 3.3.9)



図 3.3.9 : 曲ファイルに含まれる情報

曲ファイルにはそれぞれ曲のデータが記録されており、アーティスト、アルバム名、発行年、トラック番号、ジャンル等がある。MP3 でエンコードされたファイルの場合は、ID3 タグと呼ばれる領域に格納されている。これらの情報から必要な情報を読み取って、データベースに送信する。

(2) 必要な機能とシステム構成

利用者所有のパソコン

- iTunes のフォルダから曲情報を集める。
- 集めた情報をデータベースに送信する。
- 曲情報を送信する場合は、自動または 1 アクションでデータを送信できるようにする。

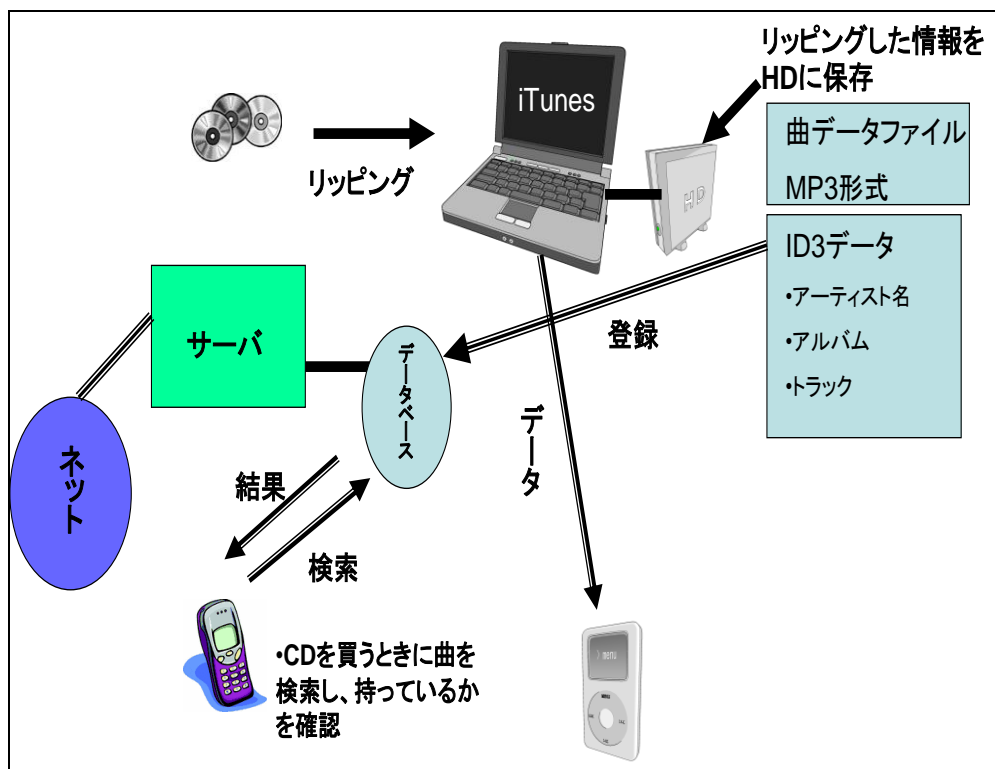


図 3.3.10

この図(図 3.3.10)のように、はじめに iTunes に CD をリッピングし、ID3 データをデータベースに登録する。携帯電話の WEB を利用してサーバへアクセスする。携帯電話からアーティスト名、アルバム名、トラック名などで CD 情報を検索する。そして、検索した結果が送信される。このことから、データベース(サーバ)を設計する必要がある。

(3) 必要なデータベース

- 利用者所有のパソコンから送信されたデータを登録するフォームが必要。また、CD によっては、ID3 データを含んでいない CD があるため、手で入力するものも必要になる。
- 検索のための情報を入力するフォームが必要。アーティスト名、アルバム名、ジャンルなど、どの情報で検索するか。
- 入力した情報をデータベースに検索を要求。
- 検索結果を表示する。その CD はすでに持っていますなどの結果を表示。

4.設計と実装

4.1 実装環境

- クライアント環境

オペレーティングシステム：Microsoft Windows XP Home Edition Version 2002
Service Pack 2

CPU：Intel® Celeron® M 1.5GHz

メモリ：1.24GB

サーバーソフトウェア

VertrigoServ2.21

Apache/PHP/MySQL/SQLite/SQLiteManage/PhpMyAdmin/Zend/Optimizer を
windowsにまとめてインストールすることができるようになっているのがオープン
ソースで開発されている VertrigoServ である。

- サーバ環境（学外：www.cyaneum.org）

オペレーティングシステム：Ubuntu 8.04 Server

CPU:：Pentium M 1.6GHz

メモリ：1GB

Web サーバ：Apache 2.2.8

データベース：MySQL 5.0.51

プログラミング言語

PHP 5.2.4

4.2 データベースの設計と構築

テーブルの設計

項目内容	フィールド	属性
連番	renban	Int
アーティスト名	artistname	varchar(30)
アルバム名	albumname	varchar(30)
トラック名	trackname	varchar(30),

Localhost へログインし、試験的なテーブルを製作する。(図 4.2.1)



図 4.2.1

データベース作成

```
mysql> create database sotu_a540068;
```

Query OK, 1 row affected (0.00 sec)

フィールドの設定

```
mysql> use sotu_a540068;
```

Database changed

```
mysql> create table mp3_data_kensaku ( renban int, artistname varchar(30) ,  
albumname varchar(30), trackname varchar(30));
```

Query OK, 0 rows affected (0.00 sec)

連番設定

```
mysql> alter table mp3_data_kensaku add index ind1(renban);
```

Query OK, 0 rows affected (0.00 sec)

Records: 0 Duplicates: 0 Warnings: 0

```
mysql> alter table mp3_data_kensaku modify renban int auto_increment;
```

Query OK, 0 rows affected (0.00 sec)

Records: 0 Duplicates: 0 Warnings: 0

4.3 機能の実装

① データの登録

mp3 ファイルを指定先のファイルから検索し、検索した mp3 ファイルの結果をデータベースへ送信、登録するためのプログラム

mp3id3get.php

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
<?php
$dir = "G:\¥¥iTunes Music";

//引数 $path にはディレクトリ、またはファイルの絶対パスを指定。
function getfileinfo($path) {
    $total_size = 0;

    //指定したのがファイルだった場合はサイズを返して終了。
    if (is_file($path)) {
        $filename = basename($path);
        if (strstr($path, ".mp3")){
            $filename
            =
            mb_convert_encoding($filename, "UTF-8", "SJIS");
            echo $filename . "<br />";
        }
        # $u_path = mb_convert_encoding($path, "UTF-8", "SJIS");
        # echo $u_path . "<br />";

        $tag = id3_get_tag( $path );
        # $id3 = mb_convert_encoding($tag, "UTF-8", "SJIS");
```



```

#          var_dump($tag);
#          echo $tag["title"] . "<br />";
          $artistname          =
mb_convert_encoding($tag["artist"],"UTF-8","SJIS");
          echo $artistname . "<br />";
          $albumname          =
mb_convert_encoding($tag["album"],"UTF-8","SJIS");
          echo $albumname . "<br />";
          $trackname          =
mb_convert_encoding($tag["title"],"UTF-8","SJIS");
          echo $trackname . "<br />";
          //データベースへ送信
          mysql_connect('localhost','root','ja5ma13ica');
          mysql_select_db('test');
          $sql = "insert into terui_test values(0, '$artistname',
'$albumname', '$trackname')";

          $result = mysql_query($sql);
          if (!$result) {
              $message = '無効なクエリ:' . mysql_error() . "¥n";
              $message .= 'クエリの内容:' . $sql;
              echo "<p>";
              echo $message;
              echo "</p>";
          } else {
              echo "正常に登録されました。<br>";
          }
      }

      return filesize($path);
  } elseif (is_dir($path)) {
      $basename = basename($path);

```

```

//カレントディレクトリと上位ディレクトリを指している場合はここで終了。
if ($basename == '.' || $basename == '..') {
    return 0;
}

//ディレクトリ内のファイル一覧を入手。
$file_list = scandir($path);

foreach ($file_list as $file) {
    //ディレクトリ内の各ファイルを引数にして、自分自身を呼び出す。
    $total_size += getfileinfo($path.'/'.$file);
}
return $total_size;

} else {
    return 0;
}
}

//関数を実行。
getfileinfo($dir);

?>

</body>
</html>

```

登録結果の表示(図 4.3.1)

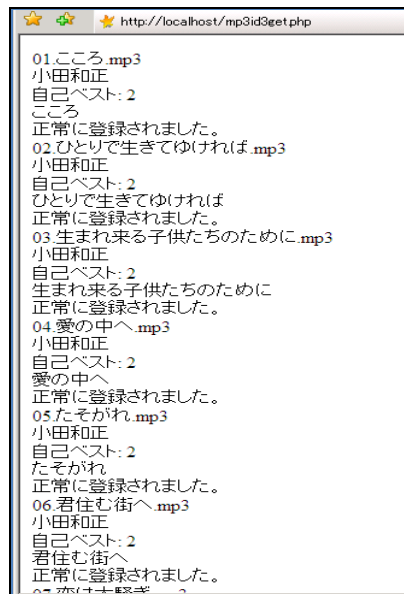


図 4.3.1 : 登録結果の表示

② 全件表示

- mp3id3get.php のプログラムでデータベースへ送信、登録した mp3 データの登録結果の全件を表示するためのプログラム。

mp3_zenken.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
  <head>
    <title>ID 全件表示</title>
    <meta http-equiv="Content-Type"
    content="text/html; charset=UTF-8">
  </head>
  <body>
    <?php
    mysql_connect('localhost','root','ja5ma13ica');
    mysql_select_db('test');

    $sql= "select * from itunes_data_kensaku";
    $result = mysql_query($sql);
```

```

$rows = mysql_num_rows($result);

if($rows == 0){
    echo "<p>該当データがありません。 </p>";
}

else {
    while($row = mysql_fetch_array($result)){
        echo "<p>";
        echo $row["renban"];
        echo "  ";
        echo $row["artistname"];
        echo "  ";
        echo $row["albumname"];
        echo "  ";
        echo $row["trackname"];
        echo "<br />";
        echo "</p><hr />";
    }
}

?>

</body>

</html>

```

全件表示画面(図 4.3.2)

1	Mr.Children	シフクノオト	空風の帰り道
2	小田和正	自己ベスト:2	こころ
3	小田和正	自己ベスト:2	ひとりで生きてゆけれ
4	小田和正	自己ベスト:2	生まれ来る子供たちの
5	小田和正	自己ベスト:2	愛の中へ
6	小田和正	自己ベスト:2	たそがれ

図 4.3.2 : 全件表示

データの検索

- アーティスト名だけでの検索を試験的に制作してみる

cd_kensaku.php

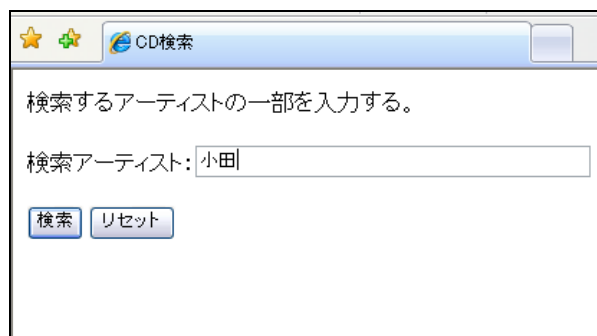
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
<head>
<title> CD 検索</title>
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
</head>
<body>
<?php
extract($_POST);
echo "
<p>検索するアーティストの一部を入力する。</p>
<form action="php echo cd_kensaku.php ?" method="php echo post ?" >
<p>検索アーティスト : <input type="text" name="nam" value="php echo $nam ?"
size="40"></p>
<p><input type="submit" value="検索">
```

```

<input type="reset" value="リセット"></p>
</form>
";
if($nam<>"){
mysql_connect('localhost','root','20cuc08!');
mysql_select_db('sotu_a540068');
$sql= "select * from mp3_data_kensaku where artistname like '%$nam%'";
$result = mysql_query($sql);
$rows = mysql_num_rows($result);
if($rows == 0){
echo "<p>あなたはこの曲を持ってないみたいです。 </p>";
}
else {
while($row = mysql_fetch_array($result)){
echo "<p>";
echo $row["renban"];
echo "  ";
echo $row["artistname"];
echo "<br />";
echo $row["albumname"];
echo "  ";
echo $row["trackname"];
echo "<br />";
echo "<hr>";
}
}
}
?>
</body>
</html>

```

アーティスト名検索画面試作(図 4.3.3)



CD検索

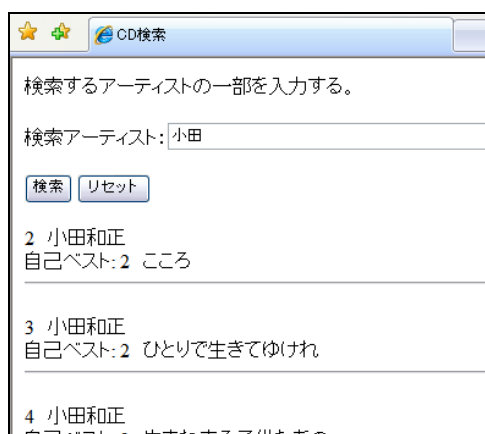
検索するアーティストの一部を入力する。

検索アーティスト: 小田

検索 リセット

図 4.3.3 : アーティスト名検索

アーティスト名検索結果画面試作(図 4.3.4)



CD検索

検索するアーティストの一部を入力する。

検索アーティスト: 小田

検索 リセット

2	小田和正	自己ベスト: 2	こころ
3	小田和正	自己ベスト: 2	ひとりで生きてゆけれ
4	小田和正	自己ベスト: 2	生まれ来る子供たちの

図 4.3.4 : 検索結果

③ アーティスト、アルバム、曲名検索

- アーティスト名、アルバム名、曲名を検索するプログラム

cd_kensaku2.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
<head>
<title> CD 検索</title>
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
</head>
```

```

<body>
<?php
extract($_POST);
echo "
<p>検索するアーティストの一部を入力する。</p>
<form action=¥"cd_kensaku2.php¥" method=¥"post¥" >
<p>アーティスト検索 : <input type=¥"text¥" name=¥"nam¥" value=¥"$nam¥"
size=¥"40¥"></p>
<p><input type=¥"submit¥" value=¥"検索¥">
<input type=¥"reset¥" value=¥"リセット¥"></p>
</form>
<p>
<p>検索するアルバム名の一部を入力する。</p>
<form action=¥"cd_kensaku2.php¥" method=¥"post¥" >
<p>アルバム検索 : <input type=¥"text¥" name=¥"alu¥" value=¥"$alu¥"
size=¥"40¥"></p>
<p><input type=¥"submit¥" value=¥"検索¥">
<input type=¥"reset¥" value=¥"リセット¥"></p>
</form>
<p>
<p>検索する曲名の一部を入力する。</p>
<form action=¥"cd_kensaku2.php¥" method=¥"post¥" >
<p>曲名検索 : <input type=¥"text¥" name=¥"kyoku¥" value=¥"$kyoku¥"
size=¥"40¥"></p>
<p><input type=¥"submit¥" value=¥"検索¥">
<input type=¥"reset¥" value=¥"リセット¥"></p>
</form>
";
if($nam<>""){
    mysql_connect('localhost','root','20cuc08!');

```



```

mysql_select_db('sotu_a540068');

$sql= "select * from mp3_data_kensaku where artistname like '%$nam%';

} else
if ($alu<>"){
mysql_connect('localhost','root','20cuc08!');
mysql_select_db('sotu_a540068');
$sql= "select * from mp3_data_kensaku where albumname like '%$alu%';
} else
if ($kyoku<>"){
mysql_connect('localhost','root','20cuc08!');
mysql_select_db('sotu_a540068');
$sql= "select * from mp3_data_kensaku where trackname like
'%'$kyoku%";
}

if ($nam<>" || $kyoku<>" || $alu<>"){
$result = mysql_query($sql);
$rows = mysql_num_rows($result);
if($rows == 0){
echo "<p>あなたはこの曲を持ってないみたいです。 </p>";
}
else {
while($row = mysql_fetch_array($result)){
echo "<p>";
echo $row["renban"];
echo " ";
echo $row["artistname"];
echo "<br />";
echo $row["albumname"];
echo " ";

```

```
        echo $row["trackname"];
        echo "<br />";
        echo "<hr>";
    }
}
?>
</body>
</html>
```

5. 評価

5.1 機能評価

機能の動作確認

- ① 初めに、パソコンでデータ検索ができるか試してみる。
- kensaku2.php へアクセスをすると以下のような画面が表示される(図 5.1.①.1)

図 5.1.①.1 : 検索画面

- アーティスト検索にキーワードを入力し検索のボタンをクリックすると以下のような画面が表示される(図 5.1.①.2)

図 5.1.①.2 : アーティストの検索結果

- アルバム名にキーワードを入力し検索した場合は以下のような画面が表示さする (図 5.1.①.3)

検索するアーティストの一部を入力する。
 アーティスト検索:

検索するアルバム名の一部を入力する。
 アルバム検索: 自己

検索する曲名の一部を入力する。
 曲名検索:

2	小田和正	自己ベスト: 2	こころ
3	小田和正	自己ベスト: 2	ひとりで生きてゆけれ
4	小田和正	自己ベスト: 2	生まれ来る子供たちの

図 5.1.①.3 : アルバムの検索結果

- 曲名にキーワードを入力し検索すると以下のような画面が表示される (図 5.1.①.4)

検索するアーティストの一部を入力する。
 アーティスト検索:

検索するアルバム名の一部を入力する。
 アルバム検索:

検索する曲名の一部を入力する。
 曲名検索: こころ

2	小田和正	自己ベスト: 2	こころ
---	------	----------	-----

図 5.1.①.4 : 曲の検索結果

- キーワードを含むデータを登録していなかった場合 (図 5.1.①.5)

検索するアーティストの一部を入力する。
 アーティスト検索:

検索するアルバム名の一部を入力する。
 アルバム検索:

検索する曲名の一部を入力する。
 曲名検索: あ

あなたはこの曲を持ってないみたいです。

図 5.1.①.5 : 検索した結果該当するデータがなかった場合の表示

② 携帯電話からでもプログラムが表示できるか試す。

- アーティスト検索、アルバム検索、曲名検索の表示画面は以下のような画面になる(図 5.1.②.1/図 5.1.②.2/図 5.1.②.3)

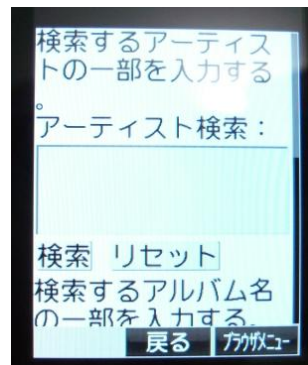


図 5.1.②.1 : アーティスト検索画面

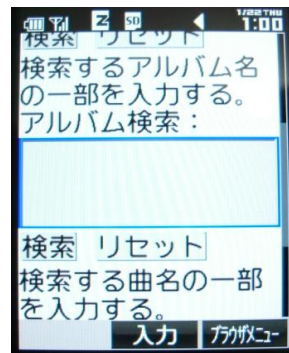


図 5.1.②.2 : アルバム検索画面

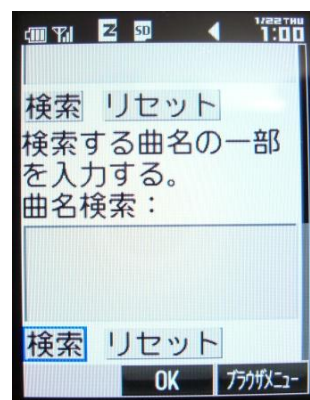


図 5.1.②.3 : 曲名検索画面

- 携帯電話から検索できるかを試す。
アーティスト検索にキーワードを入力し検索のボタンをクリックする(図 5.1.②.4)

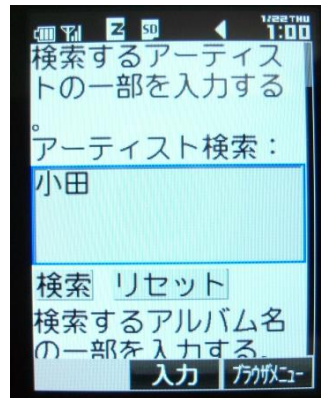


図 5.1.②.4 : アーティスト検索にキーワードを入れる

- アーティスト検索の結果を表わす場合以下のような画面が表示される(図 5.1.②.5)

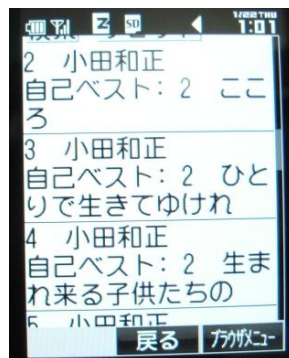


図 5.1.②.5 : 検索結果表示

- 曲名検索にキーワードを入力し検索する場合には以下の画面になる(図 5.1.②.6)

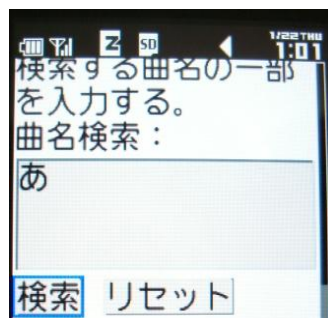


図 5.1.②.6 : 曲名にキーワードを入れる

- キーワードが含まなかった場合の表示
キーワードに当てはまるデータが登録されていなかった場合に「あなたはこの曲を持っていないみたいです。」と以下の図のように表示される(図 5.1.②.7)

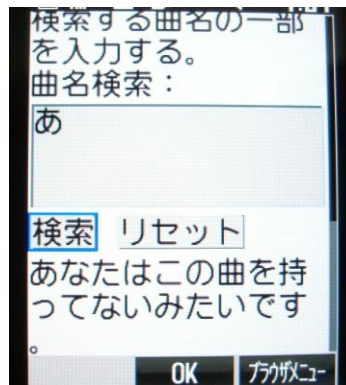


図 5.1.②.7 : データがなかった場合の表示

5.2 考察

当初の目的どおり、移動先や、パソコンが近くにない場合、携帯電話の Web を利用し、パソコンに入っている音楽データを検索するために製作したシステムであった。その点については、一度データを登録しておくことにより、携帯電話の Web を利用して、アーティスト名、アルバム名、曲名の検索で、キーワード入力を行い入力された情報に当てはまったデータを表示させるという動作を行うことができた。また、検索方法をアーティスト名、アルバム名、曲名に分けたことで検索方法を絞ることができるようになった。今後は、検索方法をアーティスト名、アルバム名、曲名だけではなく細かく検索できるようなシステムが必要になる。また、Web とのマッシュアップ機能も必要である。Web サイトをマッシュアップする場合はアマゾンなどのサイトを連携して使用できるようにする。Web とのマッシュアップ機能が出来るようになることで、新曲情報などや、お気に入りのアーティストの情報などがメール配信できる機能が出来る。

6. まとめと今後の課題

6.1 まとめ

本研究では、移動先や、パソコンが近くにならない場合に携帯電話を利用して、個人の所有している音楽データの情報を検索することができるシステムを目指した。

システム製作では、iTunes の mp3 データの保存先から mp3 データを検索し、さらに登録するためのシステムを製作した。登録した mp3 データからアーティスト名、アルバム名、曲名を検索するためのシステムを制作し、その結果や登録した全件を表示するシステムも制作することができた。

6.2 今後の課題

現在のシステムで検索できるアーティスト名、アルバム名、曲名のほかにジャンル検索などそれ以外の検索項目を増やす。そして、今回の研究で出来なかった Web とのマッシュアップ機能などが出来る機能を製作する。Web サイトをマッシュアップする場合はアマゾンなどのサイトを連携して使用できるようにし、持っているアーティストの新曲情報を提供できる機能も検討したい。また、お気に入りのアーティストの新曲情報などをメールに配信できる機能や、個人だけの使用ではなく、他人と共有できる機能も今後検討したい。

参考文献

- 【1】 財団法人音楽産業・文化振興財団／The Foundation for Promotion of Music Industry and Culture (英文略称 PROMIC)
<<http://www.promic.net/profile/>>
- 【2】 IT 用語辞典 e-Words
<<http://e-words.jp/w/Flash20Lite.html>>
- 【3】 著者 石田豊 タイトル MySQL 入門以前
出版社毎日 コミュニケーションズ ,発行年 2005/03
- 【4】 著者 石田 豊 共著 タイトル Web2.0 BOOK
出版社 毎日コミュニケーションズ 発行年 2005/03
- 【5】 社団法人日本レコード協会
<http://www.riaj.or.jp/data/analog/analog_q.html>
- 【6】 Tech-on!
<<http://techon.nikkeibp.co.jp/article/NEWS/20050422/104071/>>
- 【7】 IT+PLUS
<<http://it.nikkei.co.jp/business/column/data.aspx?ichiran=True&n=MMITab099018102005&Page=4>>
- 【8】 「主要デジタル機器及び関連デバイスの 2010 年への展望～需要・生産・販売・価格・コスト～」
<<http://www.emdata.co.jp/Newbook1.htm>>
- 【9】 iTunes Music Store
<<http://www.apple.com/jp/itunes/whatson/>>

謝辞

本研究を行うに当たり、ご指導を頂いた渡辺 恭人准教授に感謝いたします。論文の構成・プログラミング・執筆に至るまで、親身にご指導いただいたお陰で、私の実力以上の卒業論文となりました。感謝の念でいっぱいです。私自身の大きな自身になったと思います。

渡辺ゼミでは、和やかに、且つ的確な指導をしていただきました。モバイルコンピューティングとは何かの授業から始まり、最新のモバイル事情、今後のモバイル環境の変化について、教えていただきました。また、データベースや PHP などに関してもわかりやすく教えていただきました。その授業の中で、興味をもつテーマを聞く事ができ、その議題から本研究が始まりました。本研究が卒業論文として提出出来たことは大変嬉しく思えます。ここまでの過程には多くの問題がありましたが、その都度、渡辺准教授にアドバイスをいただき、また手助けしていただいたおかげで、ここまで研究を進めることができました。

最後に、私の卒論に関わって下さった方全員にもう一度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。