２０１０年度　卒業研究

「各個人から天気情報を収集した共有型天気予報の検討」

指導教員：渡辺　恭人

千葉商科大学　政策情報学部

学籍番号０７４００２２

坂本　啓明

概要

テレビ、インターネット、携帯電話等により天気予報を知らせるサービスは多くなっている。例えば、携帯電話の ｉ チャネルにより登録先の天気情報が１週間分と今日の天気予報を３時間ごとに分けて表示してされている。これは、テレビ、インターネットも表示形式は違うが、ｉチャネルと同じことが言える。しかし、３時間ごとの天気が分かったとしても、必ずしもその時間に晴れたり、雨が降ったりとした保証はない。重要なことは、数時間の先の天気を１００％理解している人はいないということだ。

そこで本研究では、各地の個人利用者からリアルタイムな天気情報を収集し、それらの情報を元にし、天気情報を共有したり、個人利用者に対して、必要となる天気情報と適宜提供する仕組みを検討する。

目次

概要・・ 2
１章　背景と目的・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・ 5
１．１　背景・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・ 5
１．２　目的・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・ 5
１．３ 本論文の構成 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・ 5
２章　現状と問題点・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・ 6
２．１　天気予報の的中率とズレ・・・・・・・・・・・・・・・・・・・・・・・・・・・ 6
２．２　様々なサイトによる天気予報・・・・・・・・・・・・・・・・・・・・・・・・・ 8
２．２．１　Weather Eye の場合・・・・・・・・・・・・・・・・・・・・・・・・・ 8
２．２．２　Weather news の場合 ・・・・・・・・・・・・・・・・・・・・・・・・11
２．２．３　Yahoo!天気情報の場合・・・・・・・・・・・・・・・・・・・・・・・ 13
２．２．４　気象庁の場合・・・・・・・・・・・・・・・・・・・・・・・・・・・・ 16
２．２．５　i チャネルの場合 ・・・・・・・・・・・・・・・・・・・・・・・・・・ 19
２．２．６　ＥＺフラッシュニュース・・・・・・・・・・・・・・・・・・・・・・・ 20
２．３　問題点・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・21
３章　解決法の検討・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・23
３．１　解決へのアプローチ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・23
３．２　解決法の提案・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・24
３．３　モデル・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・24
４章　設計・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・26
４．１　設計目標・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・26
４．２　システムの構成図と情報の流れ・・・・・・・・・・・・・・・・・・・・・・・・26
４．３　データベース・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・27
４．４　予報データ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・27
５章　実装・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・28
５．１ 実装環境 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・28
５．１．１ クライアント環境・・・・・・・・・・・・・・・・・・・・・・・・・・・28
５．１．２ サーバ環境・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・28
５．１．３ プログラミング言語・・・・・・・・・・・・・・・・・・・・・・・・・・28

５．２　各機能の実装・・・・・・・・・・・・・・・・・・・・・・・・・・・・・28
５．２．１　天気サイト・・・・・・・・・・・・・・・・・・・・・・・・ 28
５．２．２　天気予報表示・・・・・・・・・・・・・・・・・・・・・・・ 31
５．２．３　実天気登録・・・・・・・・・・・・・・・・・・・・・・・・ 37
５．２．４　実天気全件表示・・・・・・・・・・・・・・・・・・・・・・ 40
５．２．５　天気検索・・・・・・・・・・・・・・・・・・・・・・・・・ 42
５．３　実装したシステムの流れ・・・・・・・・・・・・・・・・・・・・・・・45
６章　評価・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・46
６．１　動作確認・・・・・・・・・・・・・・・・・・・・・・・・・・・・・46
６．１．１　天気予報表示・・・・・・・・・・・・・・・・・・・・・・・ 46
６．１．２　実天気登録・・・・・・・・・・・・・・・・・・・・・・・・ 47
６．１．３　実天気全件表示・・・・・・・・・・・・・・・・・・・・・・ 48
６．１．４　天気検索・・・・・・・・・・・・・・・・・・・・・・・・・ 49
６．２　一般による利用者の評価・・・・・・・・・・・・・・・・・・・・・・・50
６．３　考察・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・52
７章　まとめと今後の課題・・・・・・・・・・・・・・・・・・・・・・・・・53
７．１　まとめ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・53
７．２　今後の課題・・・・・・・・・・・・・・・・・・・・・・・・・・・・53
参考文献・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・54
謝辞・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・55

１章　背景と目的

１．１　背景

現代の気象予報は、過去のものに比べて精度が高くなってきている。そして、その情報もテレビ、インターネット、携帯電話等によって配信されているが、予報どおりとまでは言えない。例えば、携帯電話とインターネットの気象予報を比較して見ると、３時間ごとに分けられた天気予報に違いがある。また、インターネット上で天気予報を配信しているサイトをいくつか見て見ると携帯電話とインターネットを比較した時に同じように、予報に違いがある。

このように、一般人がどれを利用するかにより天気予報の情報は違ったものになる事が言える。

１．２　目的

信頼性が高い天気情報と各地一般個人からの現在地の天気予報を組み合わせた天気予報情報サイトにより、利用者が出かける、出かけた際に必要となる判断材料の提供準備を支援する事を目的とする。

１．３ 本論文の構成

本論文では第 2 章で、各天気予報サイトの現状、問題点等について述べる。また、第３章では、２章で述べたことを踏まえて解決法のアプローチ、解決法の提案等の必要な機能について述べる。第４章では３章で提案したことを元にし、第５章で設計したシステムの実装について述べる。第６章では、出来上がったシステムを評価と考察する。そして最後に、第７章で今後の課題を述べる。

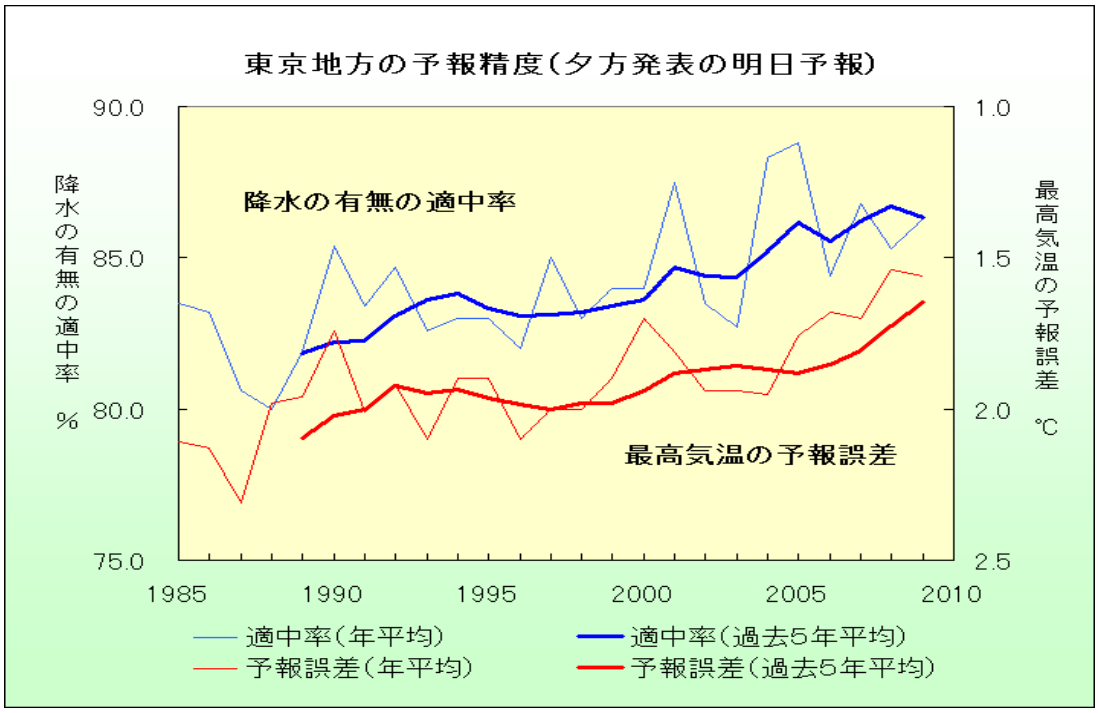
２章　現状と問題点

この章では、天気予報と実際の天気（以下実天気という）の現状について調べる。また、様々な天気予報を配信しているインターネットサイトと携帯サイトの現状を調査、分析し問題点を議論する。

２．１　天気予報の現状

人々が暮らしている中で、天気予報は出かける時や出かけた先で天気が予報と違っていた経験があると想定される。天気予報の的中率は、過去の天気予報よりも的中率が上がっているが、その予報はまだ１００％ではないのが現状である。

天気予報の的中率がどのくらいかというと、気象庁の調べでは下記図 2.1.1.1【5】のように的中率が発表されている。この的中率は、夕方１７時に発表した翌日の天気が「雨が降るか降らないか」を基準にされている。また、Yahoo!知恵袋【11】では「気象庁発表の天気予報は、的中率何%ぐらいですか？」の質問に対して、上記で説明したことと下記の解答がされていた。「例えば晴れと予報していながら実際は曇りでも「当たった」ことになるそうです。」【6】



※平成 21 年(2009 年)までを表示しています。次の更新は平成 23 年(2011 年)1 月 31 日頃の予定です。

図 2.1.1.1　天気予報の精度検証結果【5】

表 2－1　天気予報の的中率【9】

	1950 年	1975 年	2000 年	2006 年
的中率	約７２％	約７９％	約８４％	約８６％

過去５０年間の翌日の予報精度（いずれも過去５年平均値。気象庁調べ）

この図 2.1.1 に示したように、年々的中率が上がっていることがわかる。だけど、予報誤差（年平均）を見ると 2005 年から上がり、2008 年には過去一番高い結果を出している。予報誤差（過去 5 年平均）の方も同じことように、2005 年から誤差が上がり 2009 年では、過去一番高い結果が出ている。

この他に、文教大学の加藤雅典氏の論文【12】（予報精度の検証）を見ると次のように述べられている。（この論文では、予報と実際時間毎の天気が異なっている場合ズレとしている）2009 年 7 月 2 日から 11 月 30 日までに天気予報（東京）を収集した結果、「計測日数 135 日間に対し、予報数が 810 個、ズレは 292 個あり、的中率は 64.0%であった。」と述べられている。これだと 1 日に 2 回予報と異なることになる。気象庁が発表した的中率とは異なる。

気象庁と加藤雅典氏の論文を見ると、気象庁は天気予報の精度検証結果【5】2009 年的中率（年平均）が約 87%、誤差率（年平均）は約 84%に対し、加藤雅典氏の論文は的中率 64.0%、ズレ 36.0%という結果が出ている。このズレているところを無くすためには、予報とは別に実天気を収集する必要があると考えている。

実天気を収集しその情報を発信することで、予報ではズレているところを収集した実天気で補助すれば、天気のズレを減らせることになる。例えば、実天気で神奈川が 9 時に晴れていて、同時刻東京は雨だったとしよう、日本の天気は西から東に移り変わるので、この実天気を知っていれば東京の天気は晴れるということになる。これにより、予報で違っていたところも実天気を取り入れることにより的中率が上がることになる。

２．２　様々なサイトによる天気予報

２．２．１　Weather Eye の場合（インターネット）

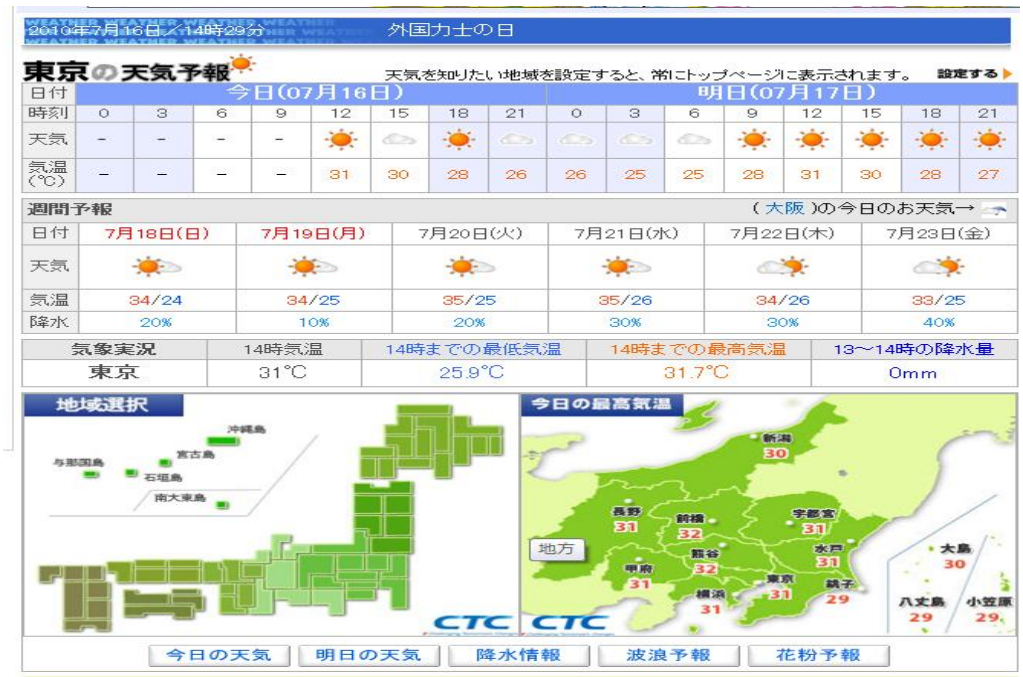


図 2.2.1.1　Weather Eye トップページ【1】

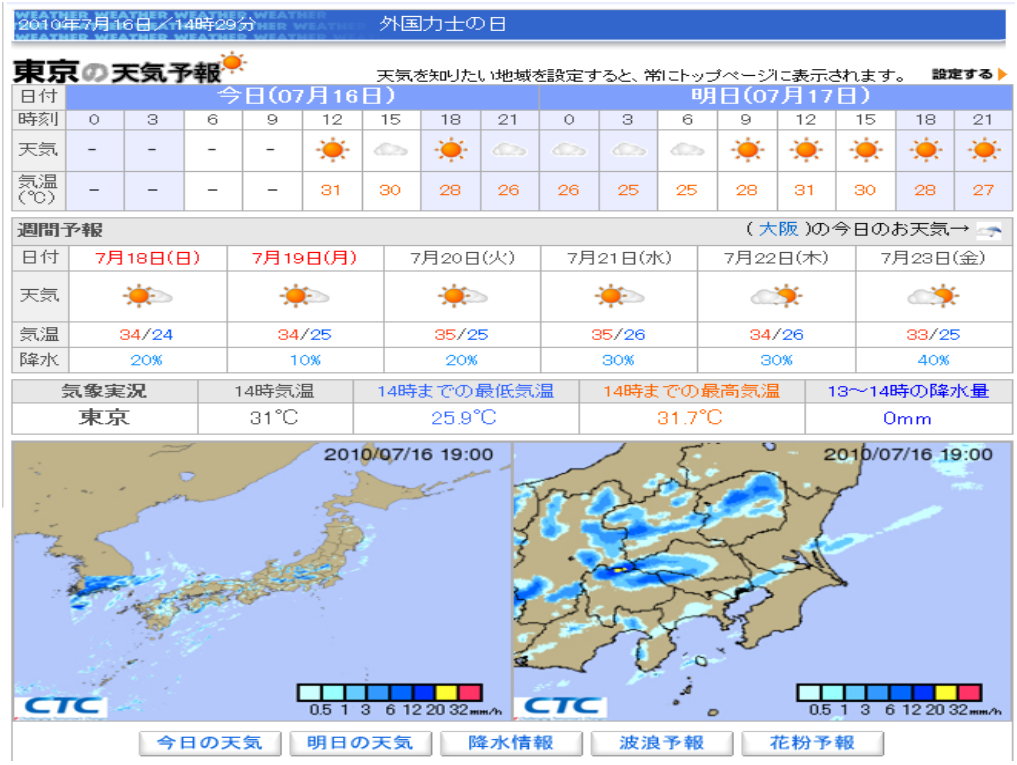


図 2.2.1.2　降水情報【1】

Weather Eye は伊藤忠テクノソリューションズ株式会社が運営する気象提供サイトである。

Weather Eye は右上部分に地域設定（設定していないと東京になる）を設定した場合３時間ごとの天気と６日分の週間予報が表示されている。また、アクセスした時間に応じて、最高気温、最低気温、降水量表示されている。

中段部分には地域選択が出来る日本地図画面と設定してある地域一帯の今日の天気画面に天気、気温、降水量が入れ替わり表示される。その他にボタンで「今日の天気」「明日の天気」「降水情報」に加え、「波浪予報」と「花粉予報」がある。「今日の天気」「明日の天気」「花粉予報」のボタンをクリックすると上記の２つの画面が変わり表示されるが、「降水情報」「波浪予報」は日本地図全体と設定地域一帯の画面が表示され、雲と波の流れが予測される。

図 2.2.1.3　リアルタイムお天気【1】

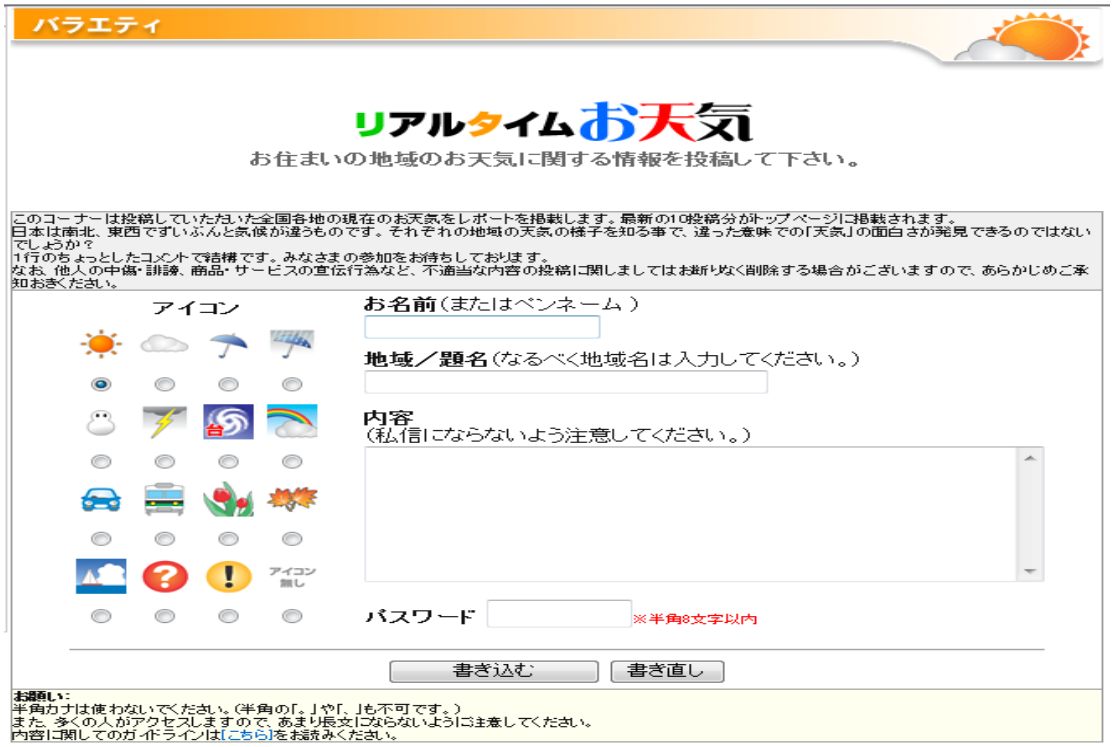


図 2.2.1.4　リアルタイムお天気登録ページ【1】

下段部分には、各地域の人から天気の情報が投稿されている。しかし投稿される情報は最新の１０件分がトップページに掲載されるだけなので、判断材料にするには十分ではない。また、最新の１０件は、地域別に分けられていないため投稿を見る時にごちゃごちゃに表示されている。

投稿の仕方は下段右上に「投稿はコチラ」というロゴがありそこから投稿画面に進む事が出来る。投稿内容はアイコン、ペンネーム、地域／題名、内容と４つに分けられパスワードを入力し、書き込みボタンをクリックすれば投稿される仕組みとなっている。

２．２．２　Weather news の場合（インターネット）

Weather news は株式会社 Weather news が運営する気象提供サイトである。

図 2.2.2.1　Weather news 全国天気【2】

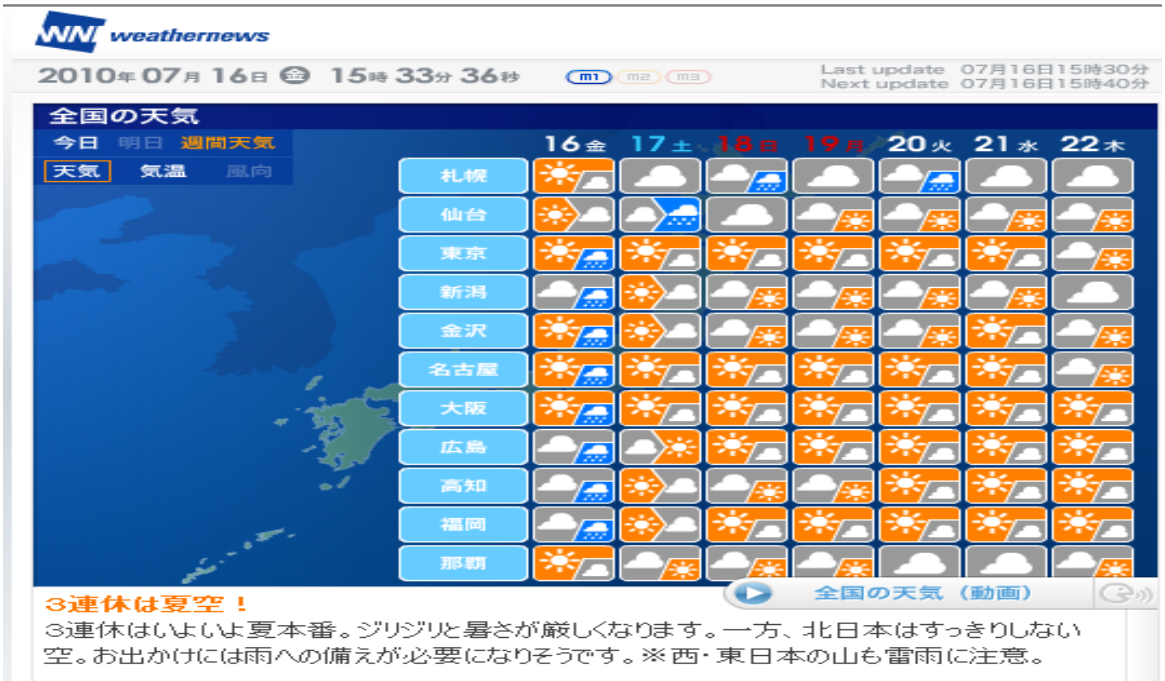


図 2.2.2.2　週間天気【2】

日本地図には、今日の天気が表示されているが左上にある「気温」「風向」を選択すれば切り替わるようになっている。明日以降の天気を知れたい場合は、「明日」「週間天気」に変更すればよい。また、右下の「全国の天気（動画）」はその日に配信された天気予報を見ることが出来る。インターネット配信のほか、BS デジタル放送（独立データ放送）910ch「ウェザーニュース」でも視聴できる。右側に「キーワード入力」場所があり（例）東京都葛飾区西水元までと詳しく表示される。検索ジャンルも数多くあり知らない場所であっても心配らない。また、天気予報は１日に約６回配信されている

図 2.2.2.3　ウェザーリポート【2】

ウェザーニュース会員から各天気の情報を投稿できる場所が設けられ、写真とコメントにより天気が解りやすくなっている。投稿数も３００件以上あるが、都道府県ごとなどに分けられていないため、今自分がいる場所の天気情報を探すのには時間がかかってしまう。また、投稿されているものは天気のことに加えて日記感覚で書く人が多い。（図 2.2.2 参照）

２．２．３　Yahoo!天気情報の場合（インターネット）

Yahoo!天気情報は Yahoo! JAPAN が運営する気象提供サイトである。

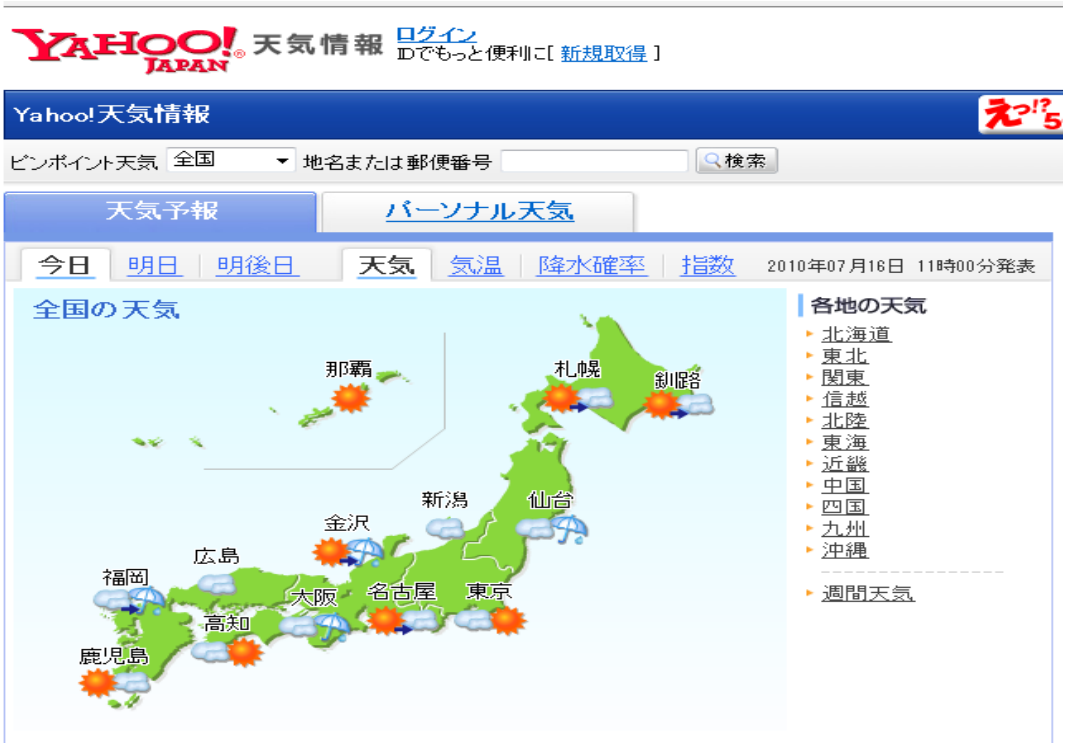


図 2.2.3.1　Yahoo!天気情報トップページ【3】

Yahoo! 天気情報の場合はピンポイント検索と地名か郵便番号検索の２つが上段に配置されている。ピンポイント検索は都道府県を選んで検索する。例えば葛飾区西水元等が出てくる。地名と郵便番号を入力するとさらに絞り込め、自分が知りたい天気がすぐに解る。郵便番号のみの場合は、１つに絞られる。そして、下の図のように今日明日の天気が３時間ごとに表示される。その下に週間天気が６日分表示される。

ピンポイント天気 東京都 地名または郵便番号 検索

Yahoo!天気情報トップ > 関東・信越 > 東京都 > 東京（東京） > 葛飾区 > 〒125-0031

[PR] まさかの価格破壊！45歳(男性)月額1,935円の死亡保険

ピンポイント天気 2010年7月16日 16時30分発表 パーソナル天気に追加

今日の天気 - 7月16日(金)

時間	0時	3時	6時	9時	12時	15時	18時	21時
天気	晴れ	晴れ	曇り	曇り	晴れ	晴れ	晴れ	晴れ
気温(℃)	27	27	26	29	31	31	28	27
湿度(%)	---	---	---	---	---	---	72	82
降水量(mm/h)	0	0	0	0	0	0	0	0
風向 風速(m/s)	南南西 7	南南西 4	南西 1	南西 2	南南西 4	南南西 6	南 5	南 3

※モノクロ表示の部分は、予報ではなく、その時間の実際の天気です。

みんなで実況！今の天気 葛飾区の今の天気を教えてください。※17時50分～18時50分の実況数 使い方

晴れ 5人 曇り 0人 雨 1人

現在、集計結果が正しく表示されません。しばらくお待ちください。

明日の天気 - 7月17日(土)

時間	0時	3時	6時	9時	12時	15時	18時	21時
天気	曇り	曇り	曇り	曇り	曇り	晴れ	晴れ	晴れ
気温(℃)	27	26	26	29	31	30	28	27
湿度(%)	88	94	94	74	62	64	74	82
降水量(mm/h)	0	0	0	0	0	0	0	0
風向 風速(m/s)	南 4	南南西 3	南西 2	南南西 3	南 4	南 6	南 5	南 4

天気情報トップ

図 2.2.3.2　ピンポイント天気（郵便番号）【3】

Yahoo!天気情報

ピンポイント天気 東京都 地名または郵便番号 検索

Yahoo!天気情報トップ > 関東・信越 > 東京都 > 東京（東京）

[PR] 持病があっても今から申し込める終身医療保険ありますか

今日の天気 2010年7月16日 11時00分発表

My Y! RSS

7月16日(金)

曇時々晴

最高気温(℃)[前日差] 31 [0]　最低気温(℃)[前日差] 26 [-1]

降水確率(%)	---	---	30	30
時間帯(時)	0-6	6-12	12-18	18-24

風: 南の風23区西部では南の風やや強く　波: 1メートル

みんなで実況！今の天気 東京(東京)の今の天気を教えてください。※13時30分～14時30分の実況数 使い方

晴れ 766人 曇り 20人 雨 29人

明日の天気 2010年7月16日 11時00分発表

7月17日(土)

曇後晴

最高気温(℃)[前日差] 32 [+1]　最低気温(℃)[前日差] 26 [0]

降水確率(%)	20	10	10	20
時間帯(時)	0-6	6-12	12-18	18-24

風: 南の風23区西部では南の風やや強く　波: 0.5メートル後1メートル

図 2.2.3　地名検索【3】

中段には、日本地図画面、各地方リンク、週間天気が表示されている。日本地図は今日の天気が表示され画面の上には、明日、明々後日までの天気が表示できるようになっている。天気表示以外に降水量と気温、指数（洗濯、傘など）というものがある。また、上段にある検索方法とは違い地図上の地名をクリックすると地方の画面に切り替わり地方の天気がわかる。最大、（例）関東→東京都→東京までいくと、今日、明日の天気と週間天気が表示される。しかし、今日の天気欄を見ると時間分けられているのは、降水確率だけであり、天気に関しては現状の天気だけになってしまう。また、そこからピンポイント天気で検索すると３時間ごとの天気が表示されている。さらに、各個人が自分の地域天気を実況する場所が存在する。これは、晴れ、雨、曇りの３つに別れ時間内に自分の地域に当てはまるものを選び、選んだものが次の時間に反映されると言う仕組みになっている。（但し雪に関しては、受付中止と表示されている。）

日付	7月17日（土）	7月18日（日）	7月19日（月）	7月20日（火）	7月21日（水）	7月22日（木）	7月23日（金）
北海道（札幌）	曇り 27/20 10%	曇後雨 28/20 60%	曇時々雨 25/21 60%	曇り 26/21 40%	曇時々晴 26/21 30%	曇時々晴 27/21 30%	曇時々晴 26/20 30%
東北（仙台）	曇り 29/22 10%	曇時々晴 29/21 30%	晴時々曇 32/23 20%	晴時々曇 32/23 20%	曇時々晴 31/23 30%	曇り 30/23 30%	曇り 28/22 40%
関東（東京）	曇後晴 32/26 10%	晴時々曇 34/24 20%	晴時々曇 34/25 10%	晴時々曇 35/25 20%	晴時々曇 35/26 30%	曇時々晴 34/26 30%	曇時々晴 33/25 40%
信越（新潟）	曇時々晴 30/24 10%	晴時々曇 29/23 30%	晴時々曇 29/24 20%	晴時々曇 29/23 20%	晴時々曇 29/23 20%	曇時々晴 29/23 40%	曇り 28/23 40%
北陸（金沢）	曇時々雨 31/23 10%	晴時々曇 33/23 20%	晴時々曇 32/23 20%	晴時々曇 32/24 20%	晴時々曇 31/24 20%	曇時々晴 31/24 30%	曇り 31/23 40%

図 2.2.3.3　週間天気【3】

週間天気は各地方の週間天気１週間分が表示されている。

２．２．４　気象庁の場合（インターネット）

気象庁は気象庁が運営する気象提供サイトである。

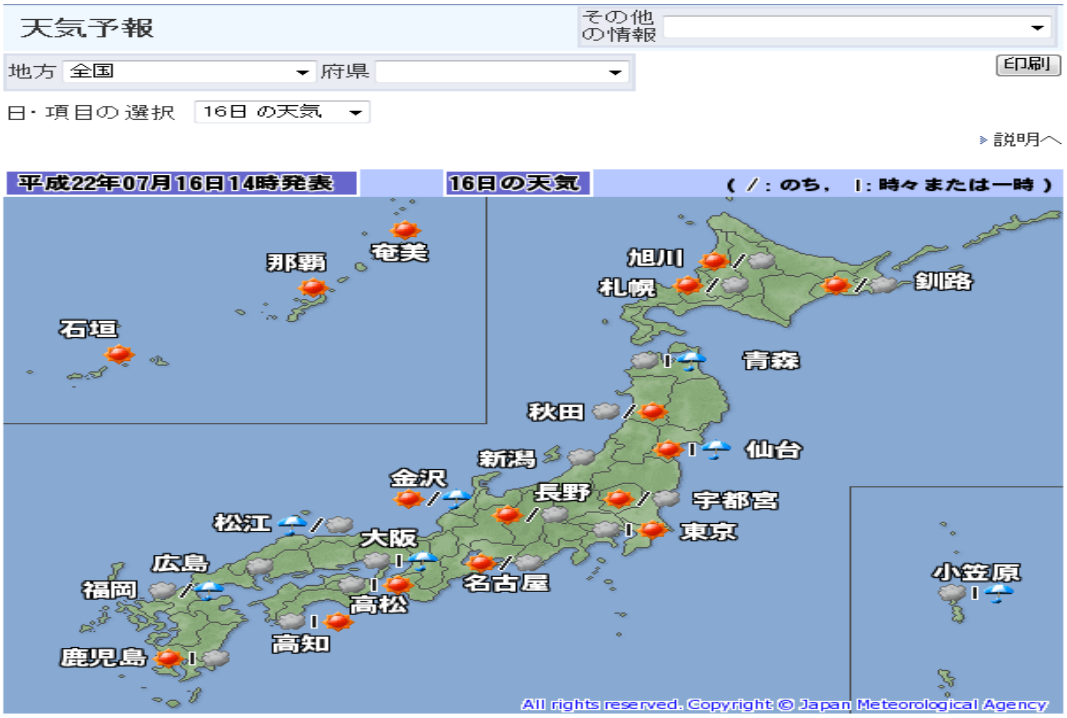


図 2.2.4.1　気象庁全国天気【4】

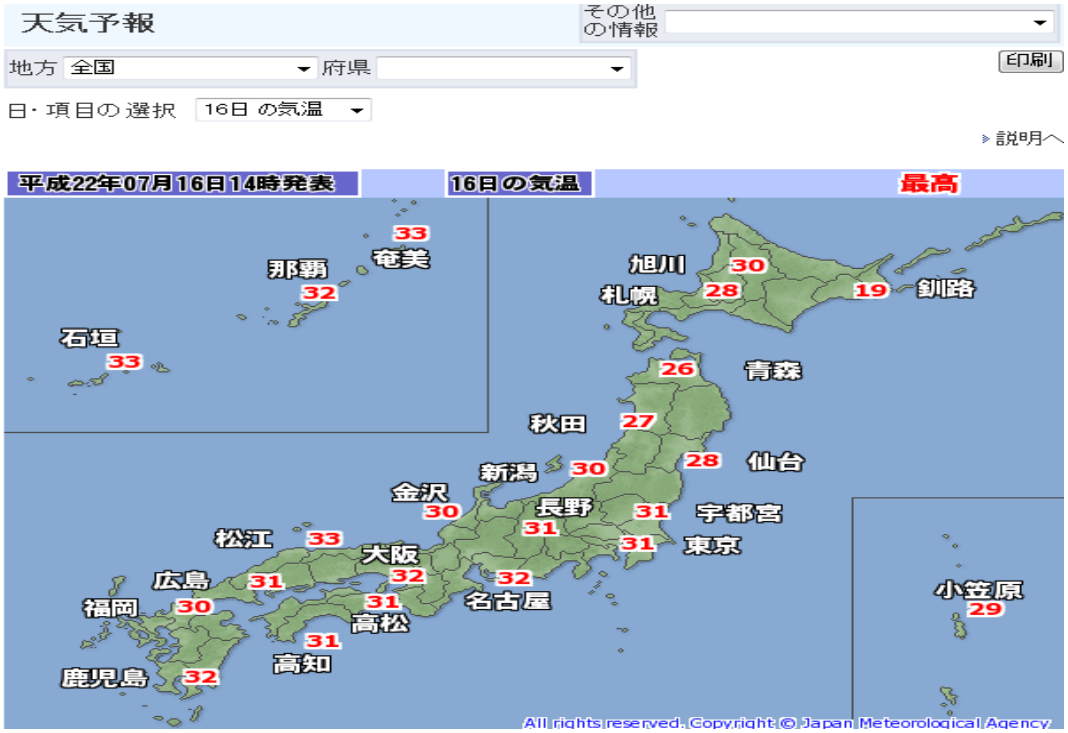


図 2.2.4.2　全国の気温【4】

上段にある地方、府県に検索事項選び知りたい場所の天気を検索できる。また、そ

の下にある（例）「１６日の天気」と書かれた欄には、３日分の天気予報と２日分の気温が選択できる。天気予報は毎日５時、１１時、１７時の１日３回発表されている。

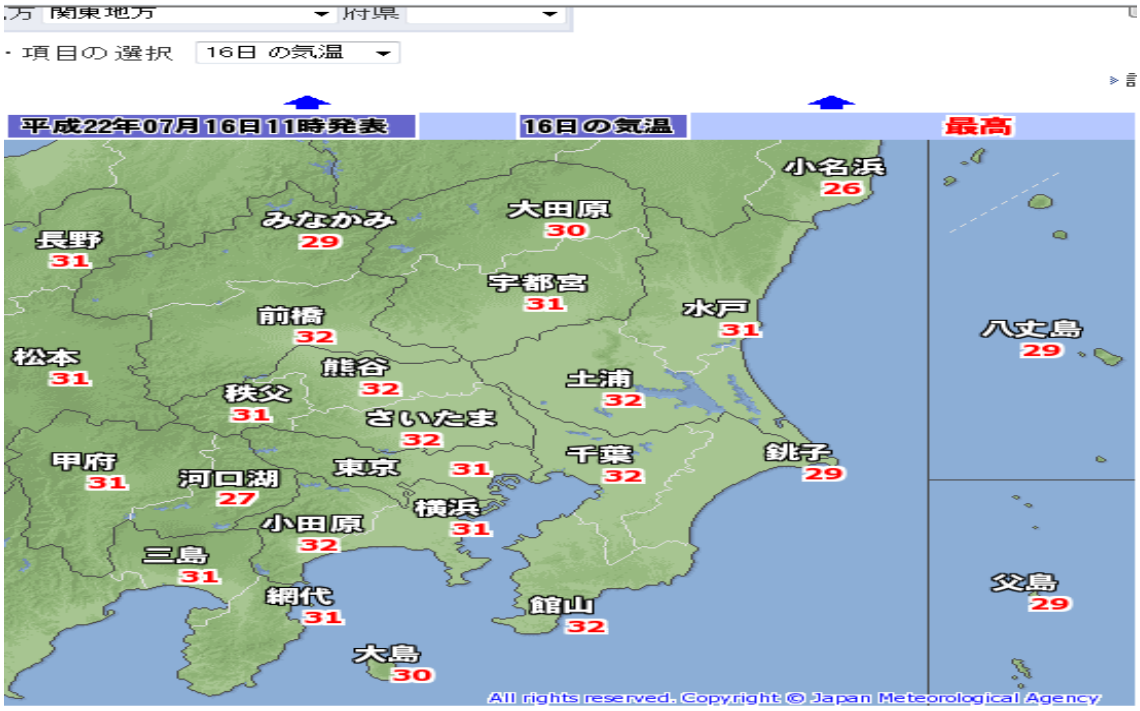


図 2.2.4.3　各地方の天気（関東）【4】

トップページの中段にある日本地図は、知りたい地方をクリックすると地方全域が表示される。さらに場所を絞り込む場合は地名をクリックすると地方の天気が表示される。

気象庁
Japan Meteorological Agency

サイトマップ

ホーム　防災気象情報　気象統計情報　気象等の知識　気象庁につい

ホーム > 防災気象情報 > 天気予報

天気予報：東京都　その他の情報

地方　府県 東京都　印刷

説明へ

16日11時 気象庁予報部 発表
天気予報(今日16日から明後日18日まで)

(/:のち, |:時々または一時)

東京地方 (地域時系列予報へ)		降水確率	気温予報
今日16日	南の風 23区西部 では 南の風 やや強く くもり 時々 晴れ 多摩西部 では 夕方から 雨 で 雷を伴う 波 1メートル	00-06 —% 06-12 —% 12-18 30% 18-24 30%	東京　日中の最高 31度
明日17日	南の風 23区西部 では 南の風 やや強く くもり 昼前から 晴れ 波 0.5メートル 後 1メートル	00-06 20% 06-12 10% 12-18 10% 18-24 20%	東京　朝の最低 26度　日中の最高 31度
明後日18日	南の風 晴れ 時々 くもり 波 0.5メートル 後 1メートル	(週間天気予報へ)	
伊豆諸島北部 (地域時系列予報へ)		**降水確率**	**気温予報**
今日16日	南西の風 くもり 夜 晴れ 波 1.5メートル	00-06 —% 06-12 —% 12-18 20% 18-24 10%	大島　日中の最高 30度
明日17日	南西の風 晴れ 波 1.5メートル	00-06 10% 06-12 10% 12-18 10% 18-24 10%	大島　朝の最低 24度　日中の最高 29度

図 2.2.4.4　各地の天気（東京）【4】

ここで表示されるのは、東京の場合、東京地方、伊豆諸島北部、南部、小笠原諸島の今日、明日、明後日の天気である。気温降水確率は、今日と明日のみとなっている。だが、３日間に共通することは、コメントが書かれていることである。雨の予報や風の方向等である。

２．２．５　ｉチャネルの場合（携帯サイト）

ｉチャネルは株式会社 Weather news が運営する気象携帯サイトである。

図 2.2.5　ｉチャネルトップ（左前半、右後半）【7】

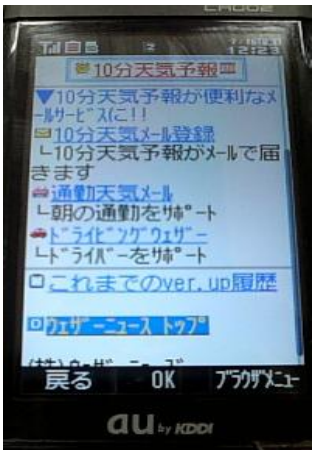
ｉチャネルとは､NTT ドコモの FOMA 端末向け情報配信サービスでありその中に天気予報がある。３時間ごとに分けられた今日の天気、気温、降水量と１週間分の週間天気が今日の天気と同じように表示されている。また、衛星画像、天気図、雨雲レーダー、台風情報などの情報も表示できる。中でも１０分天気は、自分の現在地をＧＰＳで検索し自分の場所の天気を入力することで１０分間隔の天気情報が手に入る。

２．２．６　EZ フラッシュニュース（携帯サイト）

EZ フラッシュニュースは株式会社 Weather news が運営する気象携帯サイトである。

トップ画面には、今日明日の天気確認できるボタンが備わっている。だが、ここで考えられることは、天気がどのように変わるかということがわからないのが使いにくい所である。しかし、より詳しく情報を知るには、画面真ん中にある「続きを見る」「全国天気」と２つのボタンがありこちらに行くと詳しい情報が表示されている。

図 2.2.6.1　EZ フラッシュニュートップ【8】

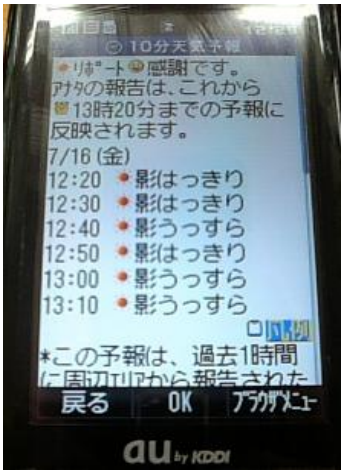


また、この携帯サイトでは、１０分天気予報というものがある。この機能は、一般個人からの携帯電話から寄せられた天気をリアルタイムに解析し、その情報を元に、１時間先までの天気を１０分後と尾に見ることが出来るサービスとなっている。

図 2.2.6.2　10 分天気【8】

この発信するやり方はいたって簡単である。「現時点の天気」と「ここ数時間後の天気」（五感で予想）を選択肢「次へ」のボタンを押すだけである。そうすると下の図のように表示される。

図 2.2.6.3　10 分天気登録ページ【8】



個人から発信された情報も表示しされており判断材料になる。しかし、このサイトではよりよい情報を得るためには、有料サイトになってしまう。

図 2.2.6.4　10 分天気ごとの予報【8】

調べたものを表にまとめたものが下記の表 1 になる。○はできている、△ふつう、×できていないを表している。

表 2－2　「既存サイトの比較」

	個人からの実天気発信	実天気の表示の仕方	コメント判断	アイコン、写真判断	天気の検索機能
Weather Eye	△	△	○	△	×
Weather news	○	△	△	△	×（○）
Yahoo!天気情報	○	○	×	○	×
気象庁	×	×	×	×	×
i チャネル	○	×	×	○	○
ＥＺフラッシュニュース	○	×	×	○	○

２．３　問題点

インターネット、携帯サイトを合計 6 つ調べた結果、どのサイトでも地域上での天気の表示の仕方が違うことがわかる。簡単に説明すると関東を選択した後に出てくる都道府県は、茨城、栃木、群馬、千葉、埼玉、東京、神奈川と 7 つに分けられる。さらに、この 7 つのうち東京を選んだとし、サイトによっては東京、伊豆諸島北部、南部、父島に分けられ場合と東京、お台場、新宿、練馬、世田谷、奥多摩等に分けられている。島に住んでいる人は島を選択すればいいが、東京に住んでいる人はサイトで

東京としか選べない場合、東京全体の天気は、晴れとサイトで表示されているが必ずしも晴れではないところが出てくる。東京と表示するにしても範囲を狭めて表示しなければ正確な天気予報とはいえない。また、配信されている天気予報の他に個人が投稿した天気情報も存在した。これは、正確な情報と嘘の情報があるにしろ、この機能は自分自身の目で見た、実感した天気を教えるのであって今までの天気予報とは、違うものと言える。しかし、各自から情報を表示できる機能を付け加えたとしても発信する人が、いないと付け加えてもまったく意味は無い。発信しやすいものを考える必要がある。発信させるにはサイトを利用するにはサイトにアクセスする必要があるため、先に実天気情報を投稿してからサイト内入れるようにすれば、実天気情報を多く収集できる。

３章　解決法の検討

３．１　解決へのアプローチ

２．３節で述べたように、天気予報は１つの地方地域だけとは限らない。そのため今までサイトの利用者は東京と見ると自分の地域だと勘違いしている場合がある。そこで都道府県を選択し、その後表示される表示を変更し正確な天気予報をサイトに上げる必要がある。

また、従来の天気予報とは違い各個人からリアルタイムの天気情報をサイトに投稿するという形もある。この機能を使う事で機械による予報とリアルによる予報の２つの予報が存在し、それらを重ね合わせることで的中率の高いものが出来るのではないだろうか。実際に、ウェザーニュースで配信された天気と現地の人から見た天気をまとめたものが下記の表 3－1 である。

表 3－1　天気予報と実天気の比較（10 月 24 日　東京）

時間	東京(WN)	東京(実)	時間のズレ	時間	東京(WN)	東京（実）	時間のズレ
0	曇り	曇り	○	１２	雨	曇り	×
1	曇り	曇り	○	１３	曇り	曇り	○
2	晴れ	晴れ	○	１４	曇り	曇り	○
3	曇り	曇り	○	１５	曇り	曇り	○
4	曇り	曇り	○	１６	曇り	曇り	○
5	曇り	晴れ	×	１７	曇り	雨	×
6	曇り	曇り	○	１８	曇り	雨	×
7	曇り	曇り	○	１９	曇り	雨	×
8	曇り	曇り	○	２０	曇り	雨	×
9	曇り	曇り	○	２１	曇り	雨	×
１０	曇り	曇り	○	２２	曇り	雨	×
１１	曇り	曇り	○	２３	雨	雨	○

３．２　解決法の提案

３．１節の事を踏まえて考えるとパソコン版のシステムを制作するために以下のことが必要である。

①広い地域に選択が１つだけならば全地域が同じ天気になってしまう。これを無くすために、狭い範囲にわけて選択場所を増やすのが理想である。

②本研究では東京と千葉の一部の地域に絞りシステム設計を行う。

③利用者が自分以外の情報を見る場合が多いと予想されるので、このサイトを利用する際には、現在地の天気を発信してからサイトに入れるようにすればよい。だが本研究では、今回は発信しなくても入れるようにする。

④収集した情報を地域ごとに分けるとこにより、自分が今探している場所の天気情報を、時間を掛けずに知ることができる。

３．３　共有型天気予報モデル

３．２節に述べたことと調べたことなどを考慮して、必要な機能を述べる

① 天気予報機能

現在提供されている、天気予報サイトから自作した Web サイト上に天気予報を配信させるとこが必要である。そして、例えば東京都選択した後に市町村区分けすることが必要である。

② 実天気登録機能

個人から実天気情報を収集するにあたり、登録する上で必要となる情報は、現地の天気、日時、発信者の現在地、発信者からのコメントが必要となる。

③ データベース

実天気を収集した後に、その情報を溜めておく場所が必要となる。そこで、データ

ベースを使い収集した実天気情報をここに記憶する。

④ 閲覧・検索機能

閲覧する場合は、天気予報と実天気情報が見られるようにする。そして、検索する場合は、自分が行く場所を入力し、実天気で登録された情報と照らし合わせて必要なものを表示させる。

⑤ Web サイト

Web サイトは、上記の①～④をリンクさせる。図にまとめると下記図 3.3.1 に示す。

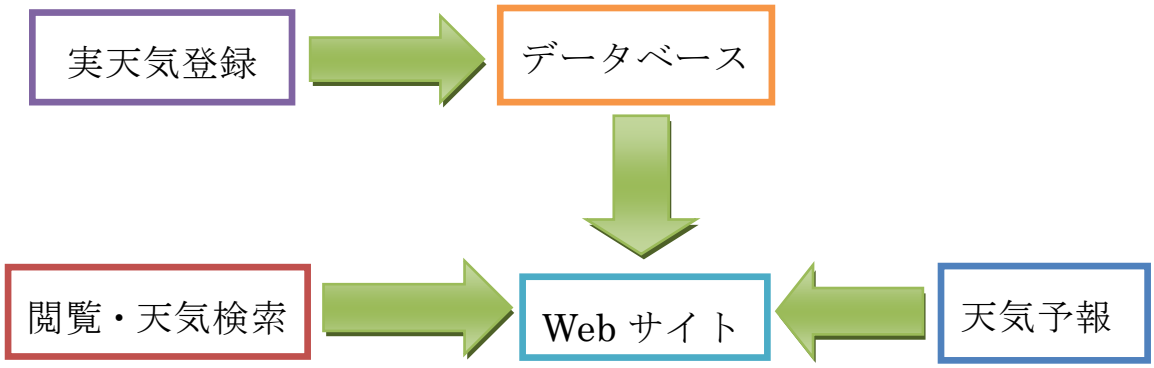


図 3.3.1　共有型天気予報モデル

登録された実天気情報、提供されている天気予報、天気検索を使用し第３者に利用してもらえる Web サイトを目指す。

４章　設計

本章では、3 章までに検討してきたことを踏まえて、必要なシステムの設計について述べる。

４．１　設計目標

本システムは、以下のような項目を実現することを目標として設計を行う。

・配信される天気予報を表示できるようにすること

・各地域の人から天気の情報を収集し、収集したものを利用者が見られるようにすること

・利用者がある目的地に向かう際に、検索機能をつけて表示すること

４．２　システムの構成図と情報の流れ

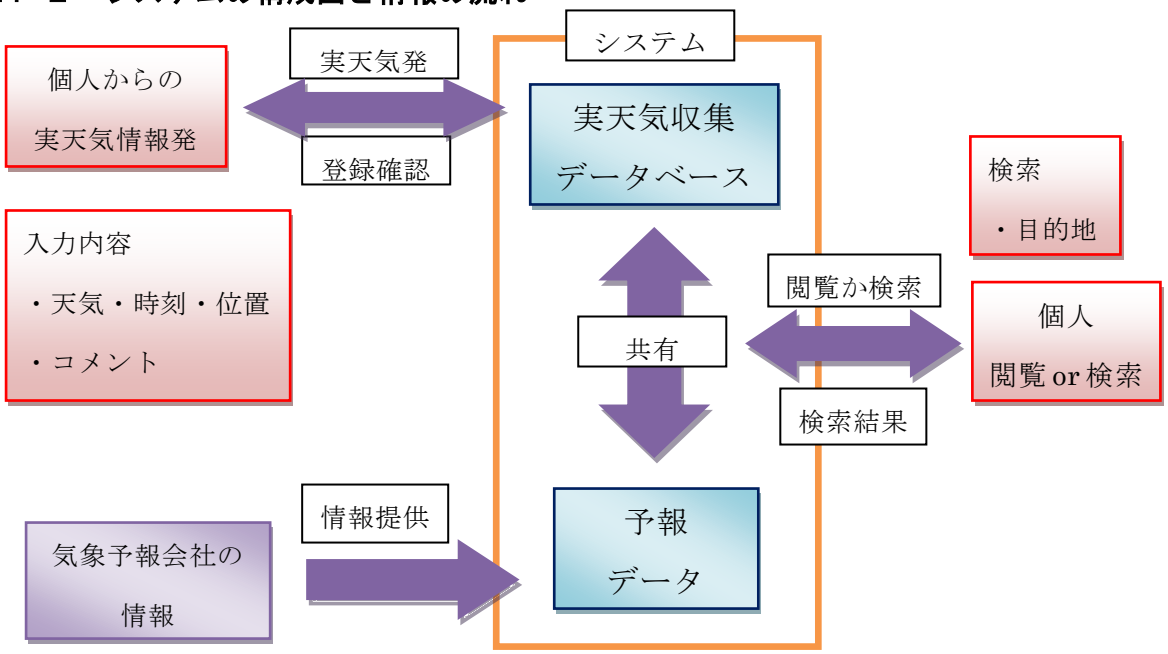


図 4－2　システムの構成図と情報の流れ

システムの構成図と情報の流れを図 4－2 に示す。設計するシステムは、PC から実天気登録、天気閲覧、天気検索を可能にする。

利用者は、PC から Web サーバ上の Web サイトにアクセスし、実天気の登録もしくは閲覧か目的地検索を行う。実天気を登録すると登録完了画面へ飛び、登録された実天気を目的地に合わせて表示する検索機能がある。閲覧する場合は、この検索機能を

使わなくても閲覧することが可能である。

４．３　データベース

本システムに必要な情報である、実天気データベース（WeatherR）を作成し、Web サイトにアクセスし個人から実天気情報を収集する。各レコードは唯一であり、同じものは存在しないようにするため、重複回避のためにインデックスを設定する。 テーブルの設計を表 4-1 に示す。

表 4－1　WeatherR

フィールド名	データ型	内容
renban	int(11)	重複回避のためのインデックス
weat	varchar(10)	天気
tim	timestamp	時間
pla	char(10)	位置
com	text	コメント

４．４　予報データ

予報データは、現在存在するサイト Weather Hacks の天気予報から RSS を利用し、Web サイトに渡すようにした。渡している情報は、今日の天気予報と最高、最低気温、そして天気概況のコメントが Web 上で見られるようになっている。

５章　実装

この章では、4章での設計に基づき、プロトタイプシステムの実装を行う。

５．１ 実装環境

５．１．１ 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 410 @1.46GHz

メモリ：1GB

５．１．２ 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

５．１．３ プログラミング言語

PHP、html を使用し実装する。

５．２　各機能の実装

５．２．１　天気サイト

tenki_s1.html と menu_tenki_s1.html で構成されている。

tenki_s1.html と menu_tenki_s1.html を使いメインとなる Web ページを作成しました。menu_tenki_s1.html を利用することで左側に天気予報情報、実天気登録、実天気全件表示、天気検索のハイパーリンクがそれぞれあり、それをクリックすると右側にそのページが表示されるようになっています。プログラムのソースは図 5.2.1.①、②に示す。

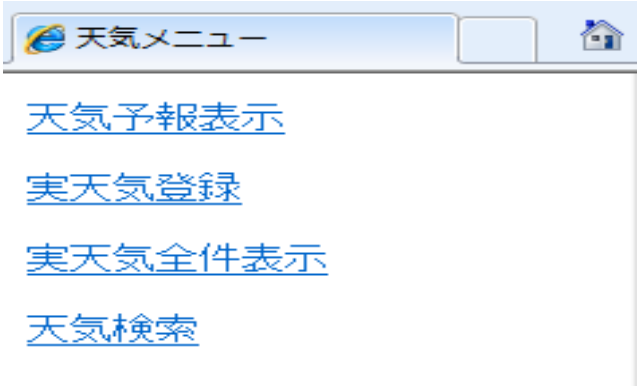


図 5.2.1.1　menu_tenki_s1.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html lang="ja">
    <head>
        <title>天気メニュー</title>
        <meta http-equiv="Content-Type" content="text/html;
charset=Shift_JIS">
    </head>
    <body>
<p><a href="wh2_s1.php" target = "main">天気予報表示</a></p>
<p><a href="tenki_touroku_s1.html" target = "main">実天気登録</a></p>
<p><a href="tenki_Z1_s1.php" target = "main">実天気全件表示</a></p>
<p><a href="ken_tenki5_s1.php" target = "main">天気検索</a></p>
    </body>
</html>
```

図 5.2.1.①　menu_tenki_s1.html

図 5.2.1.2　tenki_s1.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN">
<html lang="ja">
    <head>
        <meta http-equiv="Content-Type" content="text/html;
charset=Shift_JIS">
        <title>天気サイト</title>
</head>
<frameset cols =" 200,*">
    <frame name ="menu" src = "menu_tenki_s1.html">
    <frame name ="main" src = "wh2_s1.php">
    <noframes>
        <body>
            <p>フレーム対応のブラウザを使ってください</p>
        </body>
    </noframes>
```

```
</frameset>
</html>
```

図 5.2.1.②　tenki_s1.html

５．２．２　天気予報表示

wh2_s1.php と livedoor から構成されている。

ここでは、配信されている livedoor の天気予報を RSS を使用し wh2_s1.php のプログラムによって表示させている。また、天気概況のコメント後に市町村区のリンクを付け加え細かな天気が見られるようになっています。このリンクは、地名欄に入力されたものを$region で確認し、確認後入力した天気の情報とリンクが表示される。（本研究では、東京 23 区と千葉県の一部のみリンクは用意）プログラムのソースは図 5.2.2.③に示す。

図 5.2.2.1　wh2_s1.php

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

```
<title>天気予報情報</title>
<link rel="stylesheet" type="text/css" href="css/main.css">
<body>
<h1>天気予報情報</h1>
<?php

error_reporting(E_ALL);

set_time_limit(0);

ini_set('default_socket_timeout', 60*5);

$region = '東京';
if (isset($_REQUEST['region'])) {
	$region = trim($_REQUEST['region']);
}

$region_for_form = htmlspecialchars($region, ENT_QUOTES);

echo <<<HTML_FORM_END
<form action="" method="get">
<label>地名 ：</label>
<input type="text" name="region" size="20"
 value="$region_for_form"><br />
<input type="submit">
</form>
HTML_FORM_END;

if (is_null($region)) {
```

```
        exit ();
}

require_once 'Services/Livedoor/Weather.php';

$service = new Services_Livedoor_Weather();

$gottenResults = $service->get($region, 'today');

if (PEAR::isError($gottenResults)) {
            echo $gottenResults->message;exit();
}



$title = htmlspecialchars(
            $gottenResults['title'], ENT_QUOTES);
$link = htmlspecialchars(
            $gottenResults['link'], ENT_QUOTES);
$description = htmlspecialchars(
            $gottenResults['description'], ENT_QUOTES);
$telop = htmlspecialchars(
            $gottenResults['telop'], ENT_QUOTES);
$image = $gottenResults['image'];
$image_url = htmlspecialchars(
$image['url'], ENT_QUOTES);
$image_height = (integer)($image['height']);
$image_width = (integer)($image['width']);
$temperature = $gottenResults['temperature'];
$temperature_max =
```

```
            (integer)($temperature['max']['celsius']);
$temperature_min =
            (integer)($temperature['min']['celsius']);

$results_info_in_html = <<<HTML_RESULTS_INFO_END
<h2>$title</h2>
<h3><img src="$image_url"
  width="$image_width" height="$image_height"> $telop</h3>
<p>
<dt>最高気温</dt><dd>${temperature_max}度</dd>
<dt>最低気温</dt><dd>${temperature_min}度</dd>
</p>
<p>
$description <a href="$link">もっと読む</a>
</p>
<p>powerd by <a href="http://weather.livedoor.com/">
Livedoor 天気情報</a></p>
HTML_RESULTS_INFO_END;

echo $results_info_in_html;

$toukyo='東京';
$chiba='千葉';

if($region == $toukyo){
      echo"<a href=http://weather.livedoor.com/p/13/63/13101.html>・千代田区</a>";
      echo "       ";
      echo"<a href=http://weather.livedoor.com/p/13/63/13102.html>・中央区</a>";
      echo "       ";
```

```
echo"<a href=http://weather.livedoor.com/p/13/63/13103.html>・港区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13104.html>・新宿区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13105.html>・文京区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13106.html>・台東区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13107.html>・墨田区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13108.html>・江東区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13109.html>・品川区</a>";
echo "<br / >";
echo"<a href=http://weather.livedoor.com/p/13/63/13110.html>・目黒区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13111.html>・大田区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13112.html>・世田谷区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13113.html>・渋谷区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13114.html>・中野区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13115.html>・杉並区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13116.html>・豊島区</a>";
echo "　　";
echo"<a href=http://weather.livedoor.com/p/13/63/13117.html>・北区</a>";
```

```
    echo "      ";
    echo"<a href=http://weather.livedoor.com/p/13/63/13118.html>・荒川区</a>";
    echo "<br / >";
    echo"<a href=http://weather.livedoor.com/p/13/63/13119.html>・板橋区</a>";
    echo "      ";
    echo"<a href=http://weather.livedoor.com/p/13/63/13120.html>・練馬区</a>";
    echo "      ";
    echo"<a href=http://weather.livedoor.com/p/13/63/13121.html>・足立区</a>";
    echo "      ";
    echo"<a href=http://weather.livedoor.com/p/13/63/13122.html>・葛飾区</a>";
    echo "      ";
    echo"<a href=http://weather.livedoor.com/p/13/63/13123.html>・江戸川区</a>";
}
elseif($region == $chiba){
        echo"<a   href=http://weather.livedoor.com/p/12/67/12100.html>・千葉市
</a>";
        echo "      ";
        echo"<a   href=http://weather.livedoor.com/p/12/67/12203.html>・市川市
</a>";
        echo "      ";
        echo"<a   href=http://weather.livedoor.com/p/12/67/12204.html>・船橋市
</a>";
        echo "      ";
        echo"<a   href=http://weather.livedoor.com/p/12/67/12207.html>・松戸市
</a>";
}

?>
</body></html>
```

図 5.2.2.③　wh2_s1.php

５．２．３　実天気登録

tenki_touroku_s1.html と tenki2_s1.php で構成されている。

tenki_touroku_s1.html ではフォームにより、個人から実天気を収集し WeatherR データベースに記憶される。登録内容は、実天気、場所、コメントを入力し登録ボタンを押すだけである。実天気では、ラジオボタンを使い晴れ、雨、曇り、雪の４つから選択する。場所は、利用者の現在地を入力する。そして天気の情報をより詳しくするためにコメントを入力する。最後に登録ボタンを押すと、POST により送信される。登録されれば、tenki2_s1.php に行き「正常に登録されました」と表示される。プログラムのソースは図 5.2.3.④、⑤に示す。

図 5.2.3.1　tenki_touroku_s1.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
    <head>
        <title>実天気登録</title>
        <meta http-equiv="Content-Type"
         content="text/html; charset=Shift_JIS">
```

```
    </head>
    <body>
<p>現在地の天気を入力してください</p>
<form action="tenki2_s1.php" method="post" >

<p>天気 ：
<form action='radio.php' method='GET'>
   <img src="./../tenkiG/hare403.jpg" width=50></a>
   <input type="radio" name="gender" value="晴れ">晴れ
   <img src="./../tenkiG/ame.jpg" width=50></a>
   <input type="radio" name="gender" value="雨">雨
   <img src="./../tenkiG/kumori.jpg" width=50></a>
   <input type="radio" name="gender" value="曇り">曇り
   <img src="./../tenkiG/yuki.jpg" width=50></a>
   <input type="radio" name="gender" value="雪">雪</p>


<p>場所 ：<input type="text" name="pla" size="20"></p>
<p>コメント ：<textarea name="com" rows="10" size="50">
</textarea></p>
<p><input type="submit" value="登録">
<input type="reset" value="リセット"></p>
</form>
        </body>
</html>
```

図 5.2.3.④　tenki_touroku_s1.html

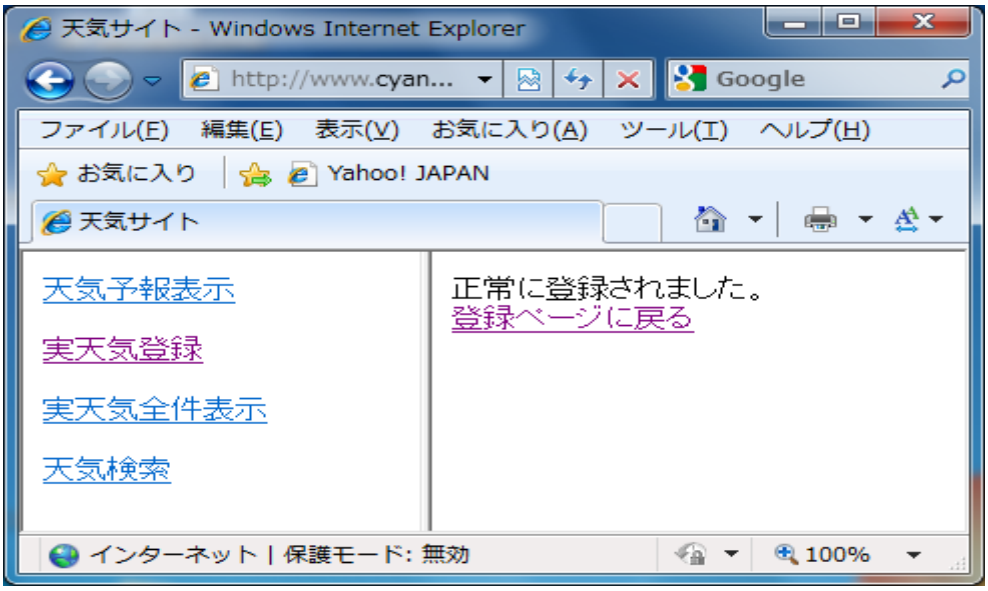


図 5.2.3.2　tenki2_s1.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
    <head>
            <title>実天気登録完了</title>
        <meta http-equiv="Content=Type"
    content="text/html; charset=Shift_JIS">
    </head>
    <body>
<?php
extract($_POST);
mysql_connect('localhost','a740022','hoge740022');
mysql_select_db('lesson_a740022');
$kyou = date("y-m-d");
$sql = "insert into WeatherR values(0,'$gender',
          NULL, '$pla', '$com')";
$result = mysql_query($sql);
if (!$result) {
```

```
        $message = '無効なクエリです。：' . mysql_error() . "<br>";
        $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>";
        echo "<a href=tenki_touroku_s1.html>登録ページに戻る</a>";
}
?>
    </body>
</html>
```

図 5.2.3.⑤　tenki2_s1.php

５．２．４　実天気全件表示

tenki_Z1_s1.php で構成されている。

tenki_Z1_s1.php は、実天気登録で各個人から登録されている実天気情報を$sql= "select * from WeatherR ";を利用することで、登録されている全てを閲覧することができる。プログラムのソースは図 5.2.4.⑥に示す。

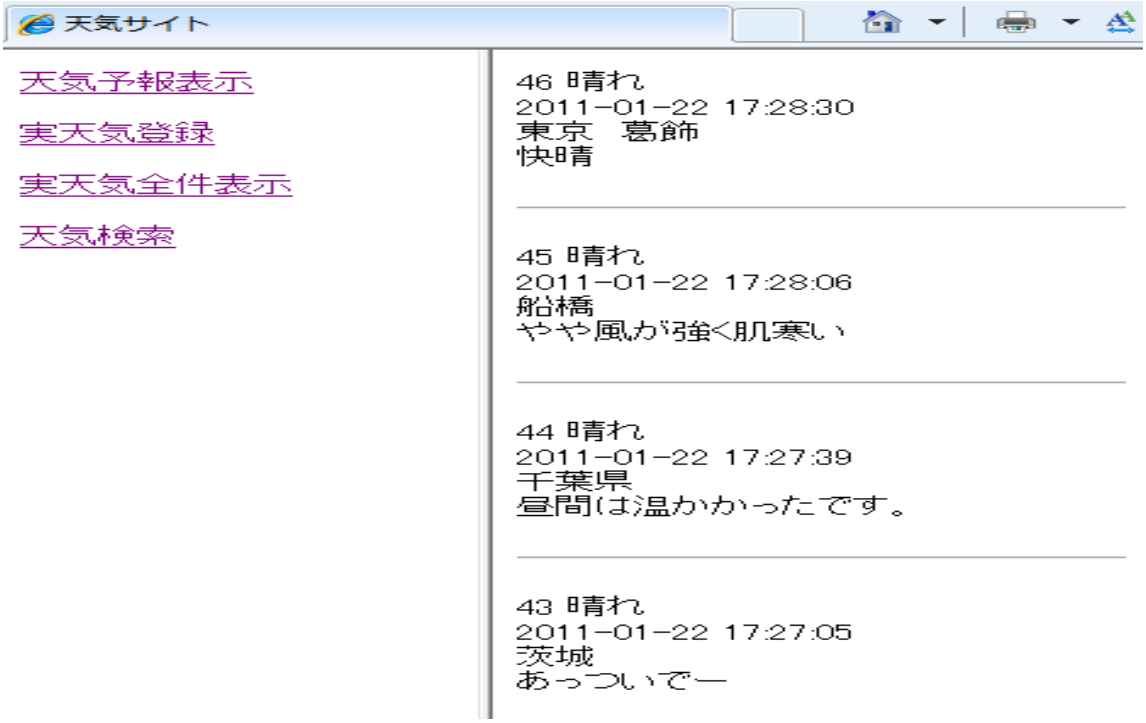


図 5.2.4.1　tenki_Z1_s1.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html  lang="ja">
    <head>
            <title>実天気全件表示</title>
        <meta http-equiv="Content=Type"
    content="text/html; charset=Shift_JIS">
    </head>
    <body>
<?php
mysql_connect('localhost','a740022','hoge740022');
mysql_select_db('lesson_a740022');
$sql= "select * from WeatherR ";
$result = mysql_query($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["weat"];
            echo "    ";
            echo $row["tim"];
            echo "<br />";
            echo $row["pla"];
            echo "<br />";
            echo $row["com"];
            echo "</p><hr />";
            }
        }
?>

</body>
</html>
```

図 5.2.4.⑥　tenki_Z1_s1.php

５．２．５　天気検索

ken_tenki05_s1.php で構成されている。

ken_tenki05_s1.php は、目的地を入力し、それを POST よって渡し検索する。そして入力した条件合う情報をデータベースから抽出し表示される。プログラムのソースは⑦に示す。

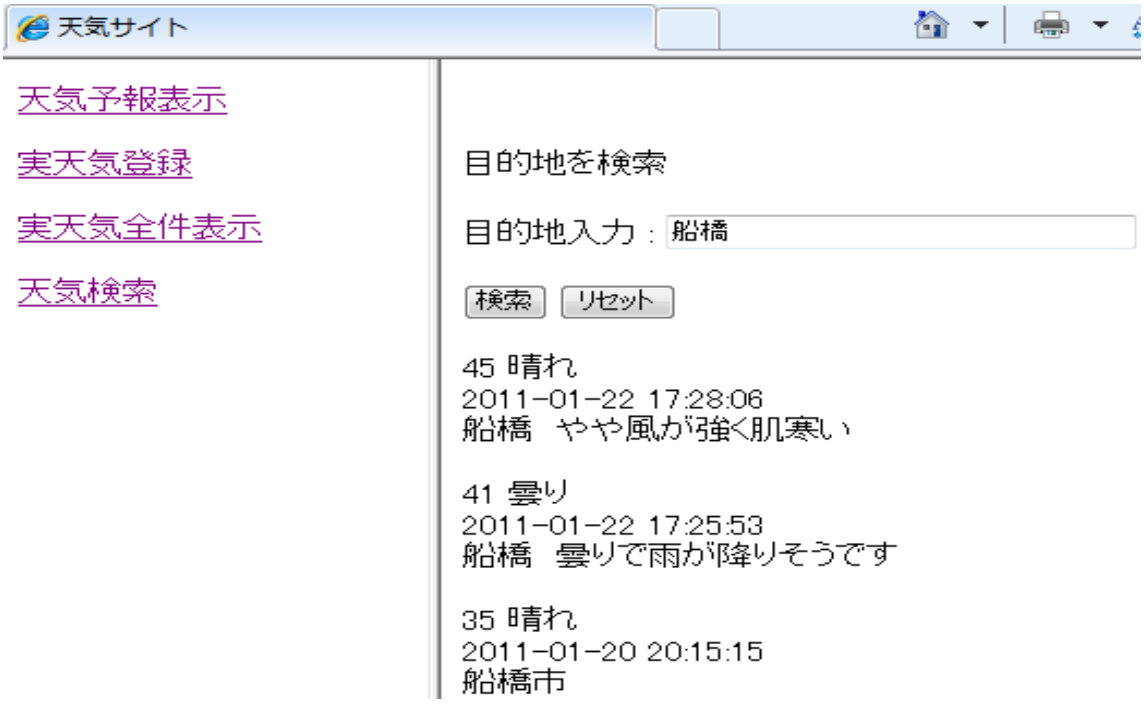


図 5.2.5.1　ken_tenki05_s1.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
    <head>
        <title>天気検索</title>
                    <meta    http-equiv="Content-Type"    content="text/html;
charset=Shift_JIS">
    </head>
    <body>
<?php>
extract($_POST);


echo "
<p>目的地を検索</p>
<form action=¥"ken_tenki5_s1.php¥" method=¥"post¥" >
<p> 目 的 地 入 力 ： <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>
";

if($nam<>''){
mysql_connect('localhost','a740022','hoge740022');
mysql_select_db('lesson_a740022');
$sql="select * from WeatherR where pla like '%$nam%' order by renban desc";
$result = mysql_query($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["weat"];
            echo "<br / >";
            echo $row["tim"];
            echo "<br / >";
            echo $row["pla"];
            echo "　";
            echo $row["com"];
            echo "</p>";

    }
```

```
    }
    }
    ?>

</body>
</html>
```

図 5.2.5.⑦　ken_tenki05_s1.php

５．３　実装したシステムの流れ

５．２節で実装したものの全体の流れを図 5.3.1 に示す。

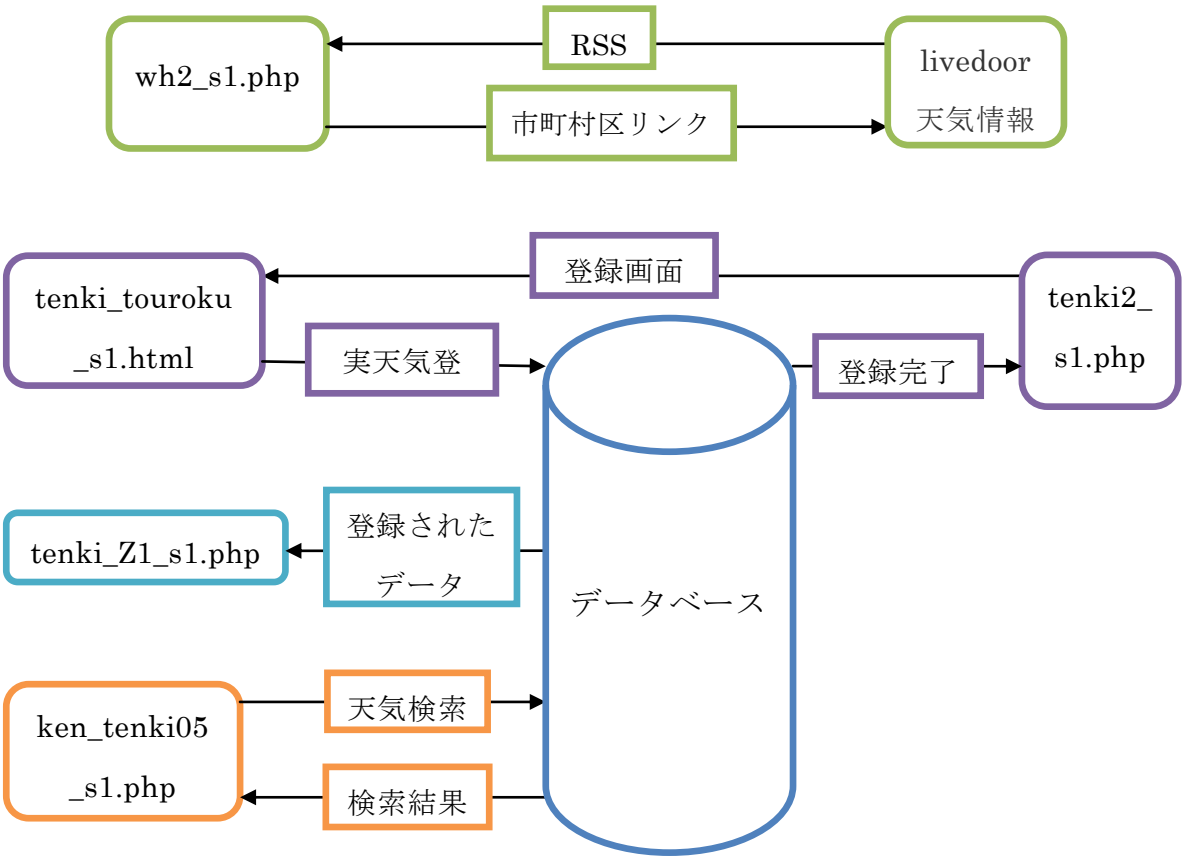


図 5.3.1　システム全体の流れ

６章　評価

この章では、５章に実装したシステム動作の確認を検証し、動作確認後学生にアンケートを行い、システムを使用した感想を聞き評価に繋げる。

６．１　各機能の動作確認

６．１．１　天気予報表示

初めに、tenki_s1.html へアクセスし、メニューバーと天気予報表示画面が表示される。地名欄に入力し天気予報が表示できるか試してみる。すると、都道府県と県庁所在地が一緒のところは表示されるが、都道府県と県庁所在地が違うところは県庁所在地を入力する必要がある。

図 6.1.1.1　天気予報表示（東京検索）

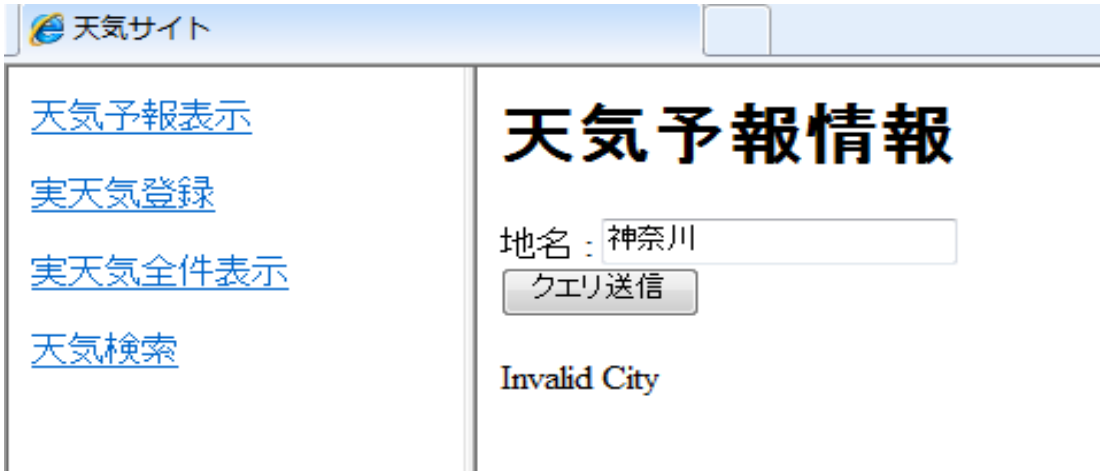


図 6.1.1.2　天気予報表示（神奈川検索）

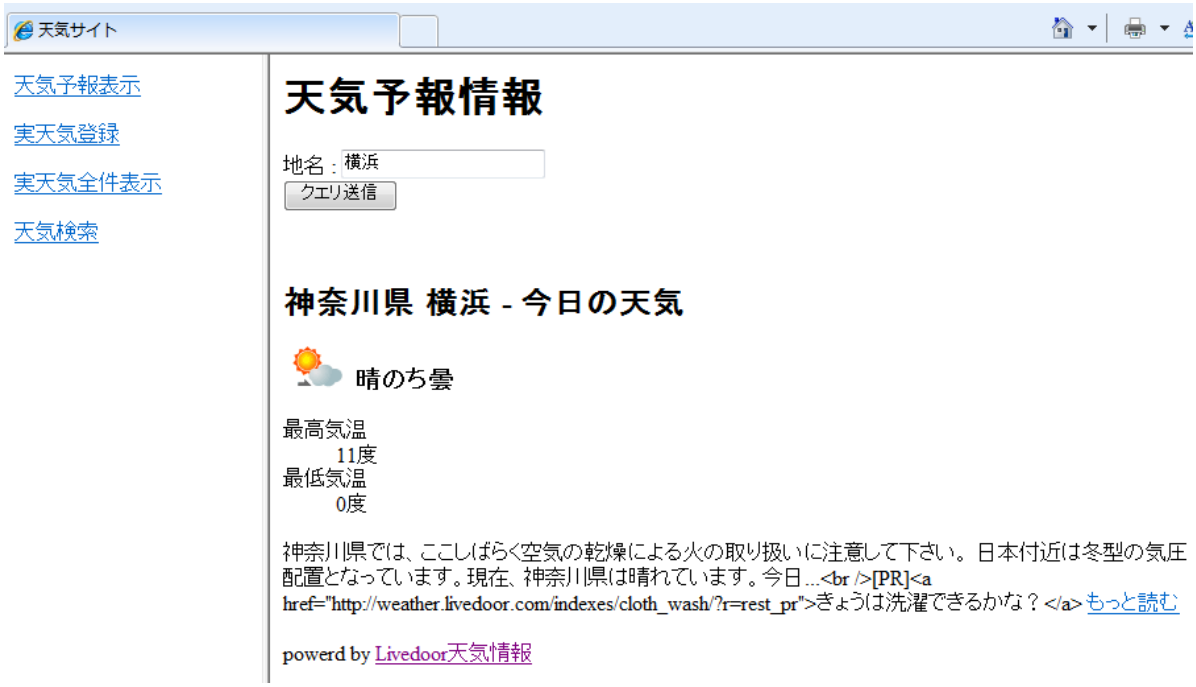


図 6.1.1.3　天気予報表示（横浜検索）

６．１．２　実天気登録

続いて実天気登録は、メニューバーから実天気登録を押すと右側のページが切り替わる。天気を登録するには、天気マークのラジオボタンを一つ選択し、現在地の場所を入力、最後にコメントを入力し、登録ボタンを押す。すると、ページが切り替わる。

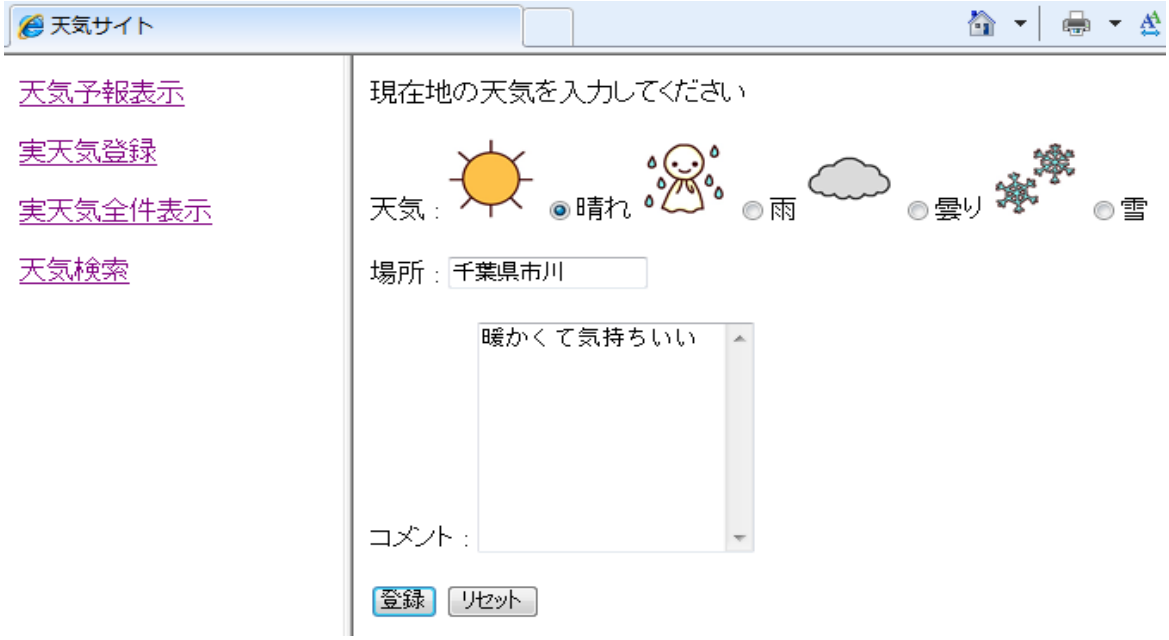


図 6.1.2.1　実行画面（実天気登録）

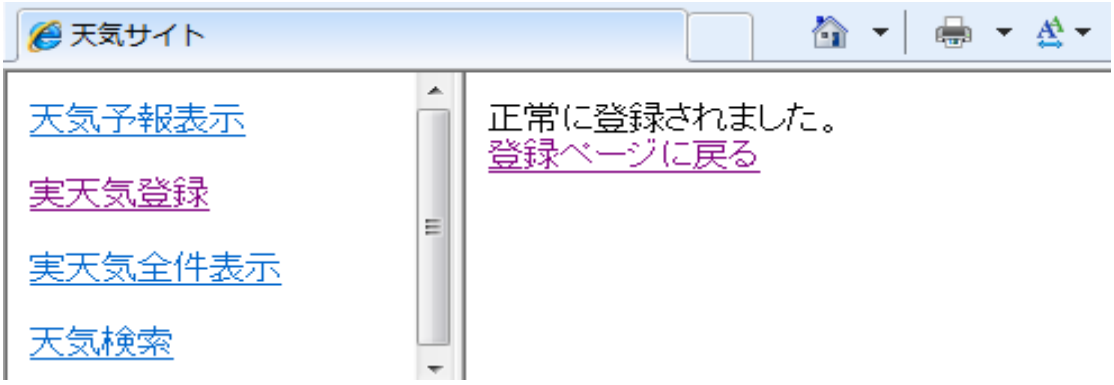


図 6.1.2.2　実行画面（実天気登録）

６．１．３　実天気全件表示

実天気全件表示では、６．１．２節で登録された情報が全て表示される。

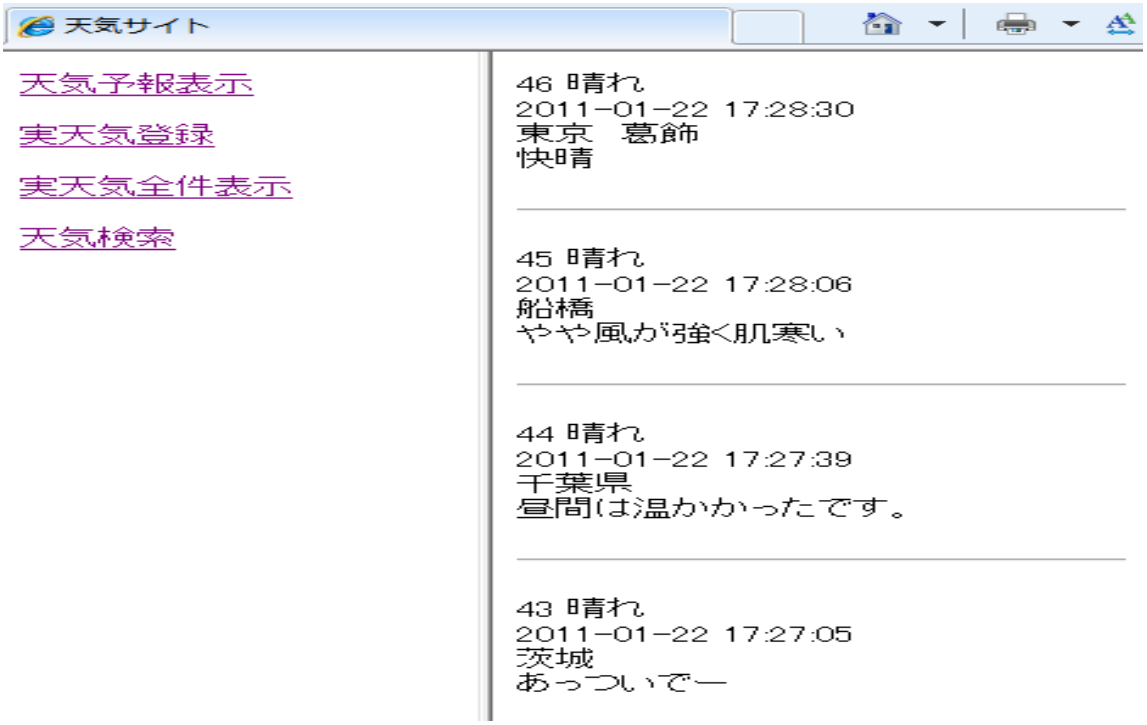


図 6.1.3.1　実行画面（実天気全件表示）

６．１．４　天気検索

天気検索は、自分が行きたい場所、行く場所の実天気を検索する。検索方法は、目的地入力欄に行きたい場所を入力し、検索ボタンを押す。そして、入力した条件あったものを抽出して表示される。

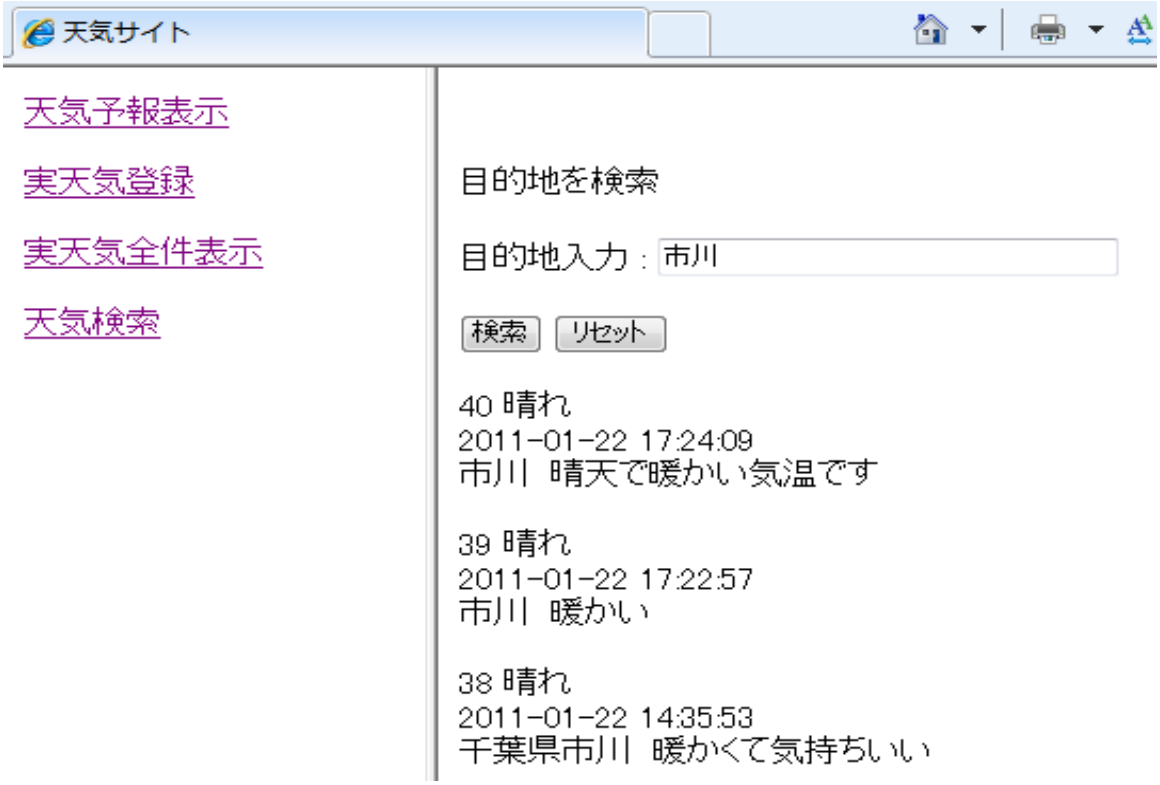


図 6.1.4.1　実行画面（天気検索）

６．２　一般による利用者の評価（学生）

本節では、学生を対象に本システムを利用してもらい、その評価について感想を述べてもらった。協力してくれた学生は１０名で、本学の学生である。質問項目は①天気予報表示を使用した感想、②実天気登録を使用した感想、③実天気全件表示を使用した感想、④天気検索を使用した感想、⑤システム全体の感想になる。その結果、集計したのが以下である。

システムの良い点

・気温や温度情報などある所が便利だと思う。 ・天気の詳細の文章も全て載せているのではなく、続きを読むことができるようになっているのが良い。 ・現地と予報の比較を出来るのは面白いと思う。

・アイコンと一緒にチェックマークがあったので選びやすくて良かった。
・現地の人それぞれの体感した天気を送信できるから、感じがつかみやすいと思う。

・本人が現地にいなくても、これから行こうと思う場所の天気が他人の体感によって表示されているので洋服の選択も対策ができると思う。
・ユーザーが地域の今のデータを入力できればどこで自分が向かっている場所の天候が登録されていれば、天候の心配が無いため助かるかと。

システムの悪い点

・現在の天気を表示するところをわかりやすく区別するために線や、枠で分けて表示するともっと見やすくなると思う。
・説明を聞かないと使い方が分からない。また、東京都は２３区あるが千葉県は４つの市しかないので、地名と入れるよりは都道府県名と記入したほうが分かりやすい。

・間違った情報を登録される危険性もあるので信頼性が問われる。
・「現在地の天気を入力してください」ではなく「現在地の今の天気を入力してください」と書いたほうが分かりやすい。

・実天気全権表示機能はどのような人を利用対象とするものか。
・都道府県だけでなく市町村など細かいところもあると良いかもしれない。（建物の名前など）
「現在地の天気を入力してください」ではなく「現在地の今の天気を入力してください」と書いたほうが分かりやすい。

・検索の際に目的地検索になってると地名を入れてしまう可能性があると思います。

・利用者を増やし機能させるかである。
・レイアウトが少し見づらいので、何か分かりやすいようにすると良いと思う。
・個人的には livedoor に飛ぶ際、別タブで開いたらよかったと思います。

利用者は実天気の情報を利用する場合、自分が出かける際に必要な情報を確認し出かけるための対策をとっている。だが、悪い点も多く存在した。作成した Web サイトのレイアウトが見やすくない点や各ページの言い回しや登録する上での追加項目などが学生に評価してわかった。今回の調査で得られた情報からシステムの改善が必要とわかった。

６．３　考察

本研究の目的は天気予報と個人からの実天気を収集し、利用者が出かける、出かけた際に必要となる判断材料の提供準備を支援することである。目的完了までとはいかなかったが、天気予報表示、実天気登録、実天気全件表示、天気検索の実装はできた。天気予報表示では、livedoor から天気予報を取得し表示させ、実天気を登録する場合は、わかりやすくラジオボタンを使用し天気を入力することを省いた。天気検索では、自分がこれから行く目的地を入力することで、登録されている実天気情報からその入力にあったものを抽出している。このシステムを使い出かける、出かけた際に必要となる判断材料になってほしい。

７章　まとめと今後の課題

７．１　まとめ

本研究では、実天気と配信の天気予報を共有させ、利用者が出かける、出かけた際に必要となる判断材料の提供準備を支援するシステムを目指した。天気予報の現状調査から始まり、既存している天気予報サイトを調査していき、天気予報の的中率を１００％に近いものにすることで、人々が出かける、出かけた際に提供されるサービス方法を深く考えさせられた。

システム制作では、実天気登録の登録内容、天気予報表示の細かな天気予報情報表示、そして他のサイトではない実天気情報を使用した天気検索などを実装した。そして、このシステムを学生に使ってもらい評価した。

７．２　今後の課題

本論文の実装は不十分であり、より改善が必要である。まず６．２節で学生に評価してもらったことを改善するとこである。このサイトをどう使えばいいのかわからない人のために、新しくトップページを設け、そこに各ページの使用方法を書き誰が利用してもわかるようになる。各ページの入力欄の言い方を変更する。実天気登録内容に新しく市町村区の欄を設け、今まで場所入力欄を都道府県にする。これにより細かな実天気情報が提供されるので、天気検索の検索方法も変える必要がある。そして、Web 上での見栄えを良くするためにフレームを使用し見やすくすることが必要である。

その他に、既存する天気予報サイトでは、写真などを表示できていたので、実天気登録下面に付け加える。そして、実天気全件表示、天気検索では天気のところは、晴れ、雨、曇り、雪と表示されていたが、実天気登録で使用した画像を使い、画像を表示する。こうした取り組みを今後の課題として検討する。

参考文献

【1】　図 2.2.1　Weather Eye　　http://www.weather-eye.com/

【2】　図 2.2.2　Weather news　http://weathernews.jp/

【3】　図 2.2.3　Yahoo!天気情報　http://weather.yahoo.co.jp/weather/

【4】　図 2.2.4　気象庁　http://www.jma.go.jp/jp/yoho/

【5】　図 2.1.1　気象庁｜天気予報の精度検証結果
http://www.data.jma.go.jp/fcd/yoho/kensho/yohohyoka_top.html

【6】　気象庁｜検証方法の説明
http://www.data.jma.go.jp/fcd/yoho/kensho/explanation.html

【7】　図 2.2.5　i チャネル

【8】　図 2.2.6　EZ フラッシュニュース

【9】　表 2-1　　天気予報の的中率／アウトドア趣味に関する総合情報サイト
http://outdoor.ymnext.com/weather-03.html

【10】 Weather Hacks　http://weather.livedoor.com/weather_hacks/

【11】 Yahoo!知恵袋
http://detail.chiebukuro.yahoo.co.jp/qa/question_detail/q1110873326

【12】 平成 21 年度「天気予報のズレの測定方法とその検証」文教大学　情報学部
経営情報学科　加藤雅典

【13】「ゼロからできる PHP＋MySQL　Web システム構築」　著者　星野　努

【14】「MySQL&PHP Web データベース入門」　著者　田中　ナルミ

【15】「SQL クックブック　―データベースエキスパートのための実践レシピ集―」
著者　Anthony Molinaro　訳者　木下　哲也

【16】PHP で Livedoor のお天気 Web サービスを使う
http://d.hatena.ne.jp/p4life/20060210/1139512547

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・参考文献及び既存サービス・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、私の実力以上のすばらしい卒業論文となりました。渡辺ゼミでは、プログラミングやデータベース等を教えていただきました。情報システムの授業から始まり、システムの基礎やネットワークの基礎を教えていただきました。私自身プログラミングやデータベース等の知識がなく、未経験からのスタートになりましたが、渡辺　恭人准教授の的確なご指導のお陰で私自身の大きな自信となり、プログラミングやデータベースに興味を持つことができました。この教えを活かしより自分のスキルアップを目指していきます。

また、卒業研究で評価をしてくださった学生の皆様には、お礼申し上げます。ここまでの過程に様々な疑問や問題がありましたが、その都度、渡辺　恭人准教授を初めゼミの皆様に助言していただき、ここまで研究を進める事が出来ました。

最後に、私の論文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。