

2012 年度卒業研究

コーディネート支援のための
所有衣服管理とアイテム提案機能の検討

指導教員 渡辺 恭人

学籍番号 0940166

丸山 みゆき

提出日：2013 年 1 月 24 日

目次

1.背景・目的	4
1-1 背景	
1-2 目的	
2.現状と問題点	6
2-1 コーディネートの現状	
2-1-1 服の選び方	
2-1-2 決め方の手順	
2-1-3 同じ服を着ることを避ける期間	
2-2 既存のアプリケーション比較	
2-2-1 Real Closet	
2-2-2 CODENOTE	
2-2-3 岡野さんの卒業研究	
2-2-4 カレンダー表示の参考アプリケーション	
2-3 問題点	
3.解決法の検討	13
3-1 解決のアプローチ	
3-2 必要な機能の提案	
3-2-1 持っている服の管理機能	
3-2-2 コーディネートの管理機能	
3-2-3 カレンダー機能	
3-2-4 ソーシャル化機能	
3-2-5 アイテムの提案機能	
4.設計と実装	15
4-1 アプリケーション設計	
4-2 実装 データベース設計	
4-2-1 持っている服管理テーブル	

4-2-2	カテゴリテーブル	
4-2-3	カテゴリ内訳テーブル	
4-2-4	色テーブル	
4-2-5	コーディネート管理テーブル	
4-3	実装 アイテム登録	
4-4	実装 アイテム全件表示	
4-5	実装 コーディネート登録	
4-6	実装 除外条件	
5.評価と考察		53
5-1	動作検証	
5-1-1	トップページ	
5-1-2	アイテム登録	
5-1-3	アイテム全件表示	
5-1-4	コーディネート登録	
5-1-5	除外条件	
5-2	機能評価	
5-3	考察	
6.まとめと今後の課題		64
6-1	まとめ	
6-2	今後の課題	
参考文献		66
謝辞		67

1.背景・目的

1-1 背景

私たちは毎日着ていく服を選びコーディネートしている。多くの女性は、頻繁に同じコーディネートや同じ服を続けて着ることを避け、様々な服を着てコーディネートを楽しみたいと思う。コーディネートは、頭の中で考え鏡の前で合わせたり、多くの服を引っ張り出して何回も着替え直したりする。しかし、服が散らかってしまうしコーディネートが決まるまでに長時間掛かってしまうことがある。コーディネートが決まらず迷う時は、持っている服をすべて把握していないこと、いつどんなコーディネートをしたのか覚えていないことが原因でコーディネートを決めるのに時間が掛かってしまう。朝起きてから出かけるまで慌ただしく時間がない中、コーディネートに多くの時間をかけず、支度を済ませたい。コーディネートを考える時間が短縮できれば、出かける前に余裕ができ効率的に時間を使うことが可能になる。

最近の流行をファッション雑誌やショップのスタッフ、街のおしゃれな人のコーディネートをチェックし、買い物に出かけることがある。買い物に行った時に気に入った服があっても、持っている服を把握していないことが原因で、持っている服と合うか似た服を持っていないかと考えて買うのを躊躇してしまい結局買わないことがある。せっかく買い物に出かけたのだから、持っている服とコーディネートしやすく気に入った服を買い、満足度の高い買い物がしたい。また、いろいろな人がコーディネートを投稿し共有できれば、いつでも見ることができ、一人ではパターン化マンネリ化しがちなコーディネートに他の人のアイデアを取り入れることで、新しいコーディネートをすることが可能になる。

1-2 目的

所有している衣服の情報と過去のコーディネートの情報を管理し、毎日のコーディネートを決めるときの服選びに悩む時間の短縮につなげることを目的とする。

持っている服の管理ができると、服を見ることができない外出先でも確認ができ、買い物に行ったときに持っている服との相性や似た服がないかの確認ができ、コーディネートの幅が広げられるアイテムやほしい服が明確になり、満足度の高い買い物ができる。

過去のコーディネートの管理ができると、いつどんなコーディネートしたのかがわかり、頻繁に同じコーディネートや同じ服を着ることを避けることができる。季節の変わり目など服装での温度調節が難しいときにも1年前の過去のコーディネートを参考にしてあらか

じめ対策をとることができる。

いろいろな人とコーディネートが共有ができると、今までの自分のコーディネートになかったアイデアが見つかり、コーディネートの幅を広げることができる。また、コメントとしてアドバイスをもらえたりしたりもできるので、さらにアイデアを取り入れることができる。

服を選びコーディネートにかかる時間の短縮につながり、同じ服を頻繁に着ることを避け持っている服を活用すること、コーディネートについてのアドバイスやアイデアを取り入れることができ、いろいろな服を着て毎日のコーディネートを楽しむことが可能になる。

2.現状と問題点

2-1 コーディネートの現状

コーディネートは、休日など時間があるときに数日分まとめて考えておいたり、出かける当日の朝に決めることが多い。当日決める時は、その日の天気・気温・予定を考慮して決めている。まとめて考えるときは、予定を考慮することは可能だが、天気や気温を考慮することは難しい。

コーディネートを決める時、頭の中で考え鏡の前で合わせたり、多くの服を引っ張り出して何回も着替え直したりする。多くの服を出すと服が散らかってしまうしコーディネートが決まるまでに長時間掛かってしまうことがある。また、着たい服がすぐに決まる時と、なかなか決まらない時がある。着たい服が決まらないのは、周りから同じ服ばかり着ていると思われたくなく、同じ服や同じコーディネートを頻繁に着ることを避けたいと思っているからである。最近着ていない服やコーディネートを考慮し、持っている服を見てこの服だとどんなコーディネートができるか、この服は最近いつ着たかななどを考えて、その日着る服を選んでいる。

2-1-1 服の選び方

ここではどのようにアイテムを選びコーディネートしてくのかを述べる。

- 1 トップス・ボトムスどちらから選んでいくのか決める。
- 2 選んだトップス・ボトムスの中から着たい服を1つ決める。
- 3 決めたアイテムを中心に全体を決めていく。3日以内に着たアイテムと、決めたアイテムと他のアイテムが同じ組み合わせにならないように選んでいく。
- 4 トップス・ボトムスが決まったら、鏡で全体をチェックし、変だと思ったら変更する。
- 5 全体を見て、アウターや靴、小物を決める。アウターとボトムスは同系色になるべくしない。同系色になったときは、靴で色を持ってくるなど、小物等でバランスを調節する。

以上のことから、コーディネートはアウターや靴からコーディネートを決めることは少なく、トップス・ボトムスを中心に考えていることがわかる。アウターや靴などは全体のバランスをとる役割も兼ねており、これらはトップス・ボトムスが決まってから選んでいる。

2-1-2 決め方の手順

ここではアイテムやコーディネートを決める時に、注意している点について述べる。

- 1 1つ目のアイテムは柄・色など特に気にしないで着たい服を選ぶ。
- 2 1つ目のアイテムに応じて、色・柄に注意しながらアイテムを決める。1つ目のアイテムで柄があるものに決めたなら、他のアイテムは柄がないものにする。柄は全体で1種類だけ、色は全体で3色までにする。以降のアイテムも同様。

以上のことから、色は全体で3色、柄は全体で1つまでとしていることがわかる。たくさん柄や色を使うと、ごちゃごちゃした印象になってしまう。柄は1種類までとし、色は3色までにすると、まとまった印象になりコーディネートする上で、注意が必要なことである。

2-1-3 同じ服を着ることを避ける期間

周りから同じ服しか着ていないと思われたいために、アイテムやコーディネートを決めるときあらかじめ選択の候補に入れないものがある。

季節外のもの、3日前までに着たもの、1週間前の同じ曜日に着たもの、である。

3日前までに着たものの「3日」は、雑誌（mina と non-no）の限られたアイテムを着回してコーディネートする企画を参考に調べたところ、同じ服を一番短いサイクルで着回していたのが、ボトムスは2日、トップスは3日であった。トップスの中でも、カーディガンなど衣服の上から羽織るアイテムは2日、インに着るアイテムは3日であった。このことから、多くの人は同じ服を3日は着るのを避けたいと思っている。

1週間前の同じ曜日着たものを除外するのは、大学の授業など同じ曜日だけしか会わない人がいる場合があり、同じ曜日に着たものを2週続けて着ることを避けたいと思っている。

2-2 既存のアプリケーション比較

既存のコーディネートアプリケーションは、大きく分けて2種類の内容のアプリケーションがある。アイテム情報を管理しコーディネートを作成するアプリケーションと、ソーシャル化されていてコーディネートの投稿ができ、アプリの利用者とコメントや Like ボタンでコミュニケーションが取れるアプリケーションの2種類である。

ここでは、コーディネート作成アプリケーションの Real Closet と、コーディネート投

稿アプリケーションの CODENOTE を紹介と比較をする。

2-2-1 Real Closet (コーディネート作成アプリケーション)

表 2-2-1 : Real Closet の機能分析

機能	アイテムの登録・管理 コーディネート登録・管理 カレンダー コーディネート投稿 (Twitter) トレンドチェック 登録アイテムのグラフ (色・季節・カテゴリ別)
長所	アイテム情報の登録項目が多く、より細かく把握できる
短所	自分一人で活用するアプリ コーディネート投稿が Twitter のみ
価格	無料

持っている服の管理と、コーディネートの作成と管理がメイン機能である。



図 2-2-1-1 : Real Closet トップ画面

持っている服は、アイテム情報が細かく登録でき、詳細にアイテムの情報を把握することができる。図 2-2-1-2 がアイテムを登録する画面である。



図 2-2-1-2 : アイテム登録画面

コーディネート作成は、図 2-2-1-3 の左画面の左下にある追加ボタンを押すと、登録してあるアイテムから選択、カメラで撮影や保存してある画像を選択することができる。選択した画像の大きさを変更することができる。左画面の季節登録を押すと中央画面に移り、季節が複数選択・登録できる。左画面のカテゴリ登録を押すと右画面に移り、どんなシーンでのコーディネートなのか登録ができる。左画面の右上の保存ボタンを押すとコーディネートが保存される。



図 2-2-1-3 : コーディネート登録画面

2-2-2 CODENOTE (ソーシャル化されたコーディネート投稿アプリケーション)

表 2-2-2 : CODENOTE の機能分析

機能	<p>コーディネート投稿</p> <p>投稿されたコーディネートとアイテムの閲覧 (新着・人気・おすすめ)</p> <p>プロフィール</p> <p>ソーシャル化 (フォロー、LIKE、コメント)</p>
----	--

	検索
長 所	ソーシャル化されていて、他ユーザーとコーディネート共有できる コーディネートに使用したアイテムのブランド名を登録することができる
短 所	持っている服の管理はできない
価 格	無料

投稿されたコーディネートを見たりコメントしたりすることがメインになっている。



図 2-2-2-1 : CODENOTE 投稿されたコーディネート新着順

マイページでは、図 2-2-2-2 の左画面が表示される。コーディネートの画像を選択すると図 2-2-2-2 の右画面のコーディネートの詳細な画面に移動する。コーディネートの詳細画面では、Like の総数と Like したユーザー、コメントの総数とコメント、コーディネートに使用したアイテムのブランド名が表示される。



図 2-2-2-2 : マイページのコーディネート画面

図 2-2-2-1 の画面下中央の投稿ボタンを押すと図 2-2-2-3 の左画面になり、カメラで撮影またはアルバムから画像を選び投稿する画像の選択をする。コーディネートに使用したブランドをカテゴリごとに 5 つまで入力することができる。入力し、図 2-2-2-3 の右画面の右上の投稿ボタンを押すとコーディネートの投稿が完了する。



図 2-2-2-3 : コーディネート投稿の流れ

2-2-3 岡野愛美さんの卒業研究

表 2-2-3 : 岡野さんの卒業研究の機能分析

機能	アイテムの登録 コーディネート登録 カレンダー 天気情報
----	---------------------------------------

持っている服の情報と収納してある場所を管理し、コーディネート時間を短く効率化を目指すアプリケーションである。持っている服の収納してある場所を表示することで、コーディネート時に服を探す手間を短縮し、天気予報などの気象情報を利用することによって、気温や天候を考慮したコーディネートが可能となっている。

2-2-4 カレンダー式表示の参考アプリ

My365 (1日1枚、思い出を写真で残す。写真ブログ)

表 2-2-4 : My365 の機能分析

機能	写真投稿 投稿された写真の閲覧 (人気・新着・イベント) マイページ ソーシャル化 (フォロー・イネ・コメント)
長所	1か月分の写真が全て一目で見ることができる
価格	無料

1日1枚の投稿なので毎日コーディネートを投稿した場合、いつどんなコーディネート

をしたか一目でわかり、確認が簡単にできる。画像を選択すると、大きい画像とタイトルがポラロイド風に表示される。画像の下には、いいねとコメントが表示される。



図：My365 マイページ

2-3 問題点

既存のアプリケーションは、服を管理しコーディネートを作成・保存することが目的の Real Closet と、コーディネートを通じて他ユーザーとコミュニケーションを取ることが目的の CODENOTE がある。2つのアプリケーションは目的や使い方が異なっており、目的に応じてそれぞれのアプリケーションを使い分ける必要がある。

コーディネートは、数日分まとめて考えておいたり、出かける当日の朝に決めることがある。当日の出かける前に服を選びコーディネートを考えてみると、着たい服が決まらなかったりしてコーディネートに長時間かかってしまう。着たい服が決まらなく迷うのは、周りから同じ服ばかり着ていると思われたくなく、同じ服やコーディネートを続けて頻繁に着ることを避け、同じ服を3日間は着ることを避けたいと思っているからである。

既存のコーディネート作成アプリケーションでは持っている服を管理し、コーディネートを作成・保存することが目的で、服の着用履歴を管理することと、着たい服が決まらず迷った時に服を選ぶ手助けをすることはできない。

同じ服を着ることを避けるには、コーディネートに使用したアイテムの情報を覚えておく必要がある。コーディネートを写真で撮ったりメモするなど覚えておく方法はあるが、撮り忘れたりメモをなくしてしまったりすることがあり、コーディネートに使用した服の情報を正確に記憶することは難しい。

毎日のコーディネートを考える時に、着たい服を簡単に選べてコーディネートに悩む時間を短縮につなげることを目指す。

3.解決法の検討

3-1 解決のアプローチ

アイテム情報を詳細に管理することで、外出先や買い物に行った時など実際に服を見ることができないときでも、持っている服を確認することができる。正確な情報を登録できるようにする。コーディネートに使用したアイテムの履歴を管理することで、いつどのアイテムを使用してコーディネートしたか、確認することができる。

アイテム提案機能を付けることで、着たい服が決まらない時に悩む時間を短縮することができるようになる。アイテムを提案する時には、あらかじめ除外する条件に合うものを候補の中から外し、条件に当てはまらないアイテムの中から提案する。

3-2 必要な機能の提案

3-2-1 持っている服の管理機能

登録内容は、カテゴリ、色、柄の有無、使用季節、画像のアイテムの基本情報に加え、着用回数、履歴の情報を登録する。アイテムごとの詳細なデータと使用状況を管理する。

実際に服を出さなくても持っている服のデザインや柄・色・形の確認ができる。買い物など外出先でも見ることが可能になり、持っている服を確認しコーディネートを考えながら買い物することができる。

3-2-2 コーディネートの管理機能

登録内容は、トップスとボトムスを必須項目とする。アウター、靴、バック、帽子、その他小物をオプション項目とし、これらのオプション項目は選択していなくてもコーディネートの登録を可能にする。トップスは重ね着をすることがあるため、2つ登録することができるようにする。また、ワンピースを着る際、ワンピースの上からトップスやボトムスを重ね着したりすることが考えられるため、ワンピースの登録はトップスに含めることとする。

3-2-3 カレンダー機能

カレンダー表示は、参考アプリの My365 のマイページのように表示をする。他のアプリケーションよりカレンダーと画像が大きく表示されるため、いつどんなコーディネートをしたのかわかりやすい。その日の予定を入力・管理すると、毎回予定の確認をしなくて

済み予定を考慮してコーディネートを考えることが可能になる。

3-2-4 ソーシャル化機能

コーディネートの投稿と、投稿されたコーディネートの共有、コメントができる。雑誌以外でも手軽におしゃれな人のコーディネートを見ることができる。一般の利用者が多いので、お気に入りのブランドの服を使ったコーディネート、アイテムの着回しや活用の仕方など参考にしやすい。実現に関しては今後の課題とする。

3-2-5 アイテムの提案機能

コーディネートを登録するときに、カテゴリや柄・色を選択して該当するアイテムを表示する。除外条件を、季節外のもの、3日前までに着たもの、1週間前の同じ曜日に着たものとし、除外条件に当てはまるものを該当するアイテムの中から外し、条件に当てはまらないアイテムを提案として表示する。表示方法は、着用回数の少ない順と最終着用日が遠い順を選択できるようにする。

この機能があることで、最近着た服を思い出して着たい服を選んで悩む時間の短縮につなげることができ、同じ服を頻繁に続けて着ることを避けることができる。

アイテム提案機能の流れを図 3-2-5 に示す。

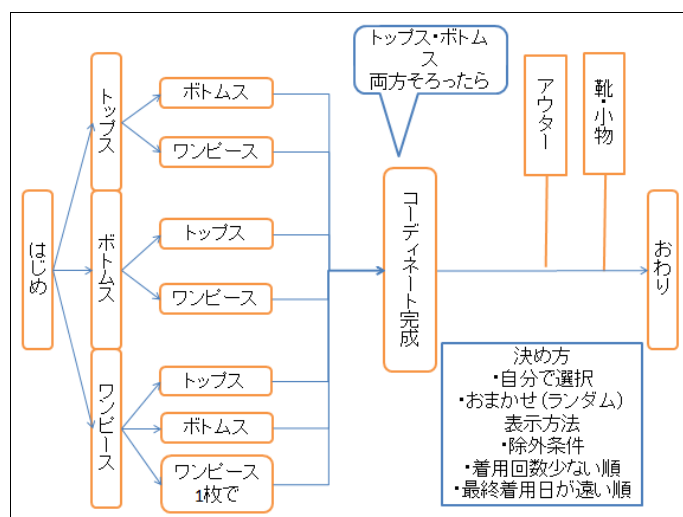


図 3-2-5 : アイテム提案機能の流れ

4.設計・実装

4-1 アプリケーション設計

4-1-1 アプリケーションの全体イメージ（トップページ）

トップページから、アイテムページ、コーディネートページ、カレンダーページ、コーディネート投稿ページの各ページに移動ができる。

トップページのイメージを図 4-1-1 に示す。

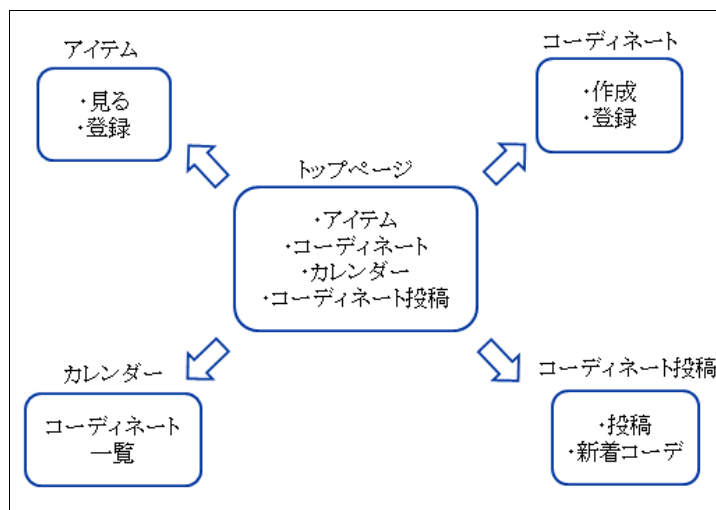


図 4-1-1：トップページのイメージ

4-1-2 アイテムページ

アイテムページでは、新しいアイテムの登録と今持っているアイテムの確認ができる。アイテムの確認は、カテゴリを選ぶとカテゴリ内のアイテムが一覧で表示され、その中から一つアイテムを選択するとそのアイテムの詳細情報が表示される。登録は、新しいアイテムの追加ができる。アイテムの画像と詳細情報を入力するとアイテムの追加登録が完了する。

アイテムページのイメージを図 4-1-2 に示す。

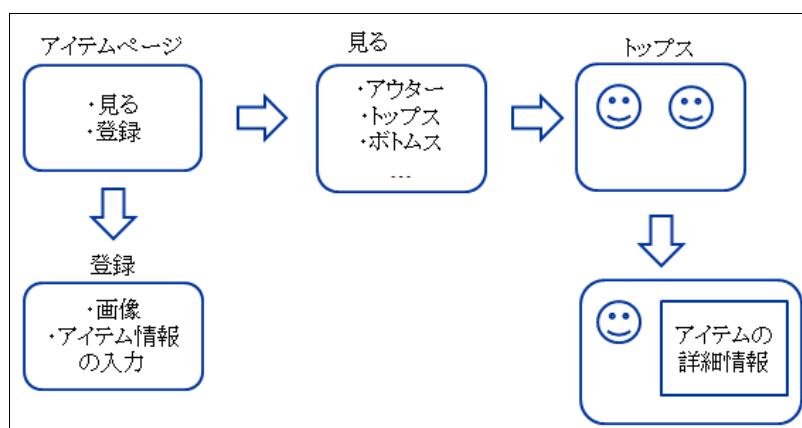


図 4-1-2 : アイテムページのイメージ

4-1-3 コーディネートページ

コーディネートページでは、登録されたコーディネートを一覧で見ることと、コーディネートの登録ができる。登録方法は2通りあり、完成したコーディネートを写真で撮りアイテムの情報を入力して登録する方法と、提案機能を使い、各アイテムを一覧から選んでいき登録する方法の2通りである。

登録方法を2通りにしたのは、コーディネートに悩まずすんなりと決まった時に、決まっているアイテムを一覧の中から選んでいくのは手間であり、簡単にコーディネートを登録する方法があったほうが継続的なコーディネートの管理ができるからである。

コーディネートページのイメージを図 4-1-3 に示す。

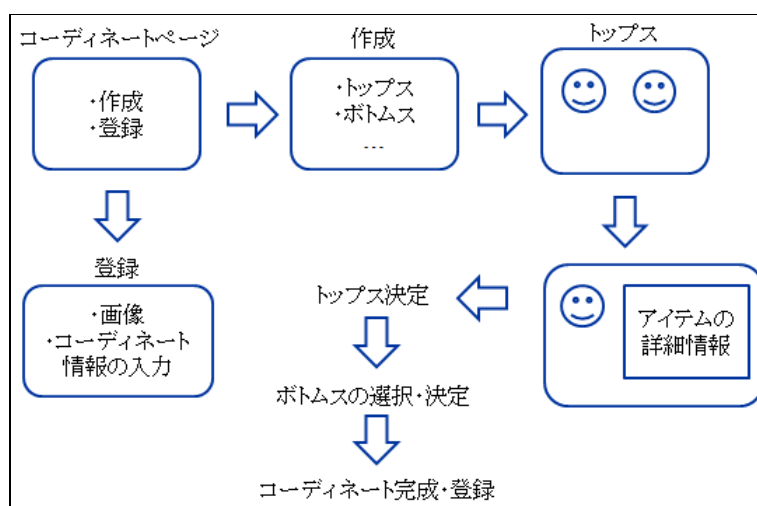


図 4-1-3 : コーディネートページのイメージ

4-1-4 カレンダーページ

カレンダーページでは、予定とコーディネートを見ること、コーディネートの登録ができる。

日にちを選ぶと、その日の予定が表示される。その日のコーディネートが登録されていない時はコーディネートの登録ができる。この場合は、コーディネートページの写真とアイテム情報の入力で登録をする。登録されると、カレンダーの日にちが表示されている部分に登録した画像が表示され、日にちは左上に小さく表示される。日にちを選ぶとその日の予定とコーディネートの画像が表示される。

カレンダーページのイメージを図 4-1-4 に示す。

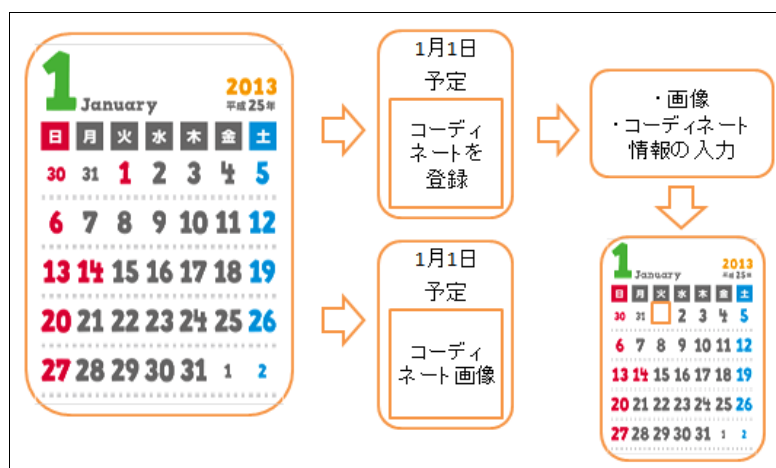


図 4-1-4 : カレンダーページのイメージ

4-1-5 コーディネート投稿ページ

コーディネート投稿ページでは、自分のコーディネートの投稿と、他ユーザーが投稿したコーディネートを見ることができる。

自分が投稿する場合は、登録したコーディネートの一覧の中から投稿したいコーディネートを選択して投稿する。新着コーデを選択すると、他ユーザーが投稿したコーディネートが新着順に一覧で表示される。気になったコーディネートを選択すると、コーディネートの大きな画像と使用したアイテムの情報が表示される。

コーディネート投稿ページのイメージを図 4-1-5 に示す。

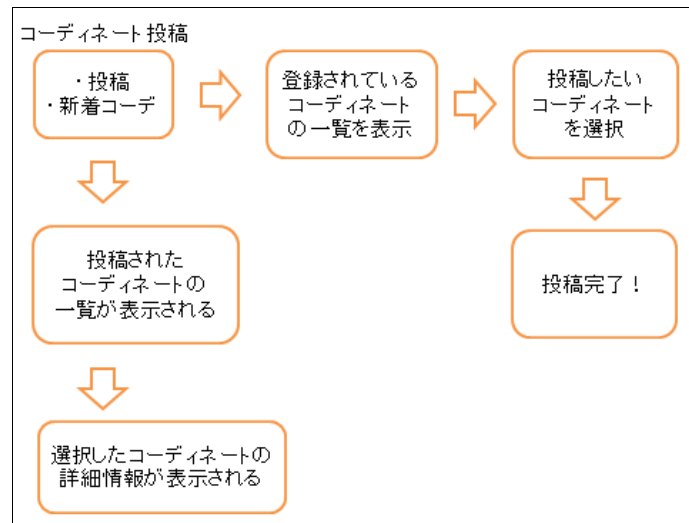


図 4-1-5：コーディネート投稿ページのイメージ

4-2 実装 データベース設計

4-2-1 持っている服管理テーブル

持っている服を管理するためのテーブルを作成する。データベースのテーブル設計を表 4-2-1-1 に示す。

表 4-2-1-1：テーブル名 closet

	フィールド名	説明	型
番号	id	ID	Int, auto-increment, index
カテゴリ	cate	別表	Int
カテゴリ内訳	cate_sub	別表	Int
色	color	別表	Int
柄の有無	gara	あり(1)なし(0)	Int
季節(春)	Spring	あり(1)なし(0)	Int
季節(夏)	Summer	あり(1)なし(0)	Int
季節(秋)	Autumn	あり(1)なし(0)	Int
季節(冬)	Winter	あり(1)なし(0)	Int

着用回数	Kaisuu	整数	Int
登録日	Touroku	日付	Date
最終着用日	Last	日付	Date
着用日履歴		別テーブル※	
画像	img	ファイル名	char(20)
備考	biko	メモ	Text

表 4-2-1-2 でテーブルを作成する。後から追加することがあるため、id を連番の設定をする。

表 4-2-1-2 : closet テーブル作成

```
mysql> create table closet ( id int, cate int, cate_sub int , color int, gara int, spring
int, summer int, autumn int, winter int, kaisuu int, touroku date, last date, img
char(20), biko text);
Query OK, 0 rows affected (0.00 sec)

mysql> alter table closet add index ind1(id);
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table closet modify id int auto_increment;
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> show fields from closet;
+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
```

id	int(11)	YES		NULL		
cate	int(11)	YES		NULL		
cate_sub	int(11)	YES		NULL		
color	int(11)	YES		NULL		
gara	int(11)	YES		NULL		
spring	int(11)	YES		NULL		
summer	int(11)	YES		NULL		
autumn	int(11)	YES		NULL		
winter	int(11)	YES		NULL		
kaisuu	int(11)	YES		NULL		
touroku	date	YES		NULL		
last	date	YES		NULL		
img	char(20)	YES		NULL		
biko	text	YES		NULL		
+-----+-----+-----+-----+-----+-----+						
14 rows in set (0.00 sec)						

4-2-2 カテゴリテーブル

Closet テーブルで必要となるカテゴリのテーブルを作成する。表 4-2-2-1 の内容を登録する。データベースのテーブル設計を表 4-2-2-2 に示す。

表 4-2-2-1 : カテゴリテーブルの登録内容

	カテゴリ
1	トップス
2	ボトムス
3	ワンピース
4	アウター
5	靴
6	小物

表 4-2-2-2 : テーブル名 cate

	フィールド名	説明	型
カテゴリ番号 (通し)	cate	1~6	Int, auto-increment, index
名称	name	トップス、ボトムスなど	varchar(10)

表 4-2-2-3 でテーブルを作成する。

表 4-2-2-3 : cate テーブル作成

```
mysql> create table cate ( cate int , name varchar(10));
Query OK, 0 rows affected (0.00 sec)

mysql> alter table cate add index ind1(cate);
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table cate modify cate int auto_increment;
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> show fields from cate;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| cate  | int(11)       | NO   | MUL | NULL    | auto_increment |
| name  | varchar(10)  | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

4-2-3 カテゴリ内訳テーブル

Closet テーブルで必要となるカテゴリの詳しい分類（カテゴリ内訳）のテーブルを作成する。表 4-2-3-1 の内容を登録する。データベースのテーブル設計を表 4-2-3-2 に示す。

表 4-2-3-1 : カテゴリ内訳テーブルの登録内容

カテゴリ	カテゴリ内訳
トップス	シャツ
	カットソー
	ニット
	パーカー
	カーディガン
ボトムス	パンツ
	スカート
ワンピース	ワンピース
アウター	コート
	ジャケット
靴	ブーツ
	サンダル
	パンプス
	スニーカー
小物	帽子
	バック
	その他

表 4-2-3-2 : テーブル名 cate_sub

	フィールド名	説明	型
通し番号	id	ID	Int, auto-increment, index
名称	name	シャツ、カットソーなど	varchar(15)
カテゴリ番号	cate	別表	int

表 4-2-3-3 でテーブルを作成する。

表 4-2-3-3 : cate_sub テーブル作成

```
mysql> create table cate_sub (id int, name varchar(15), cate int);
Query OK, 0 rows affected (0.01 sec)

mysql> alter table cate_sub add index ind1(id);
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table cate_sub modify id int auto_increment;
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> show fields from cate_sub;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | NO   | MUL | NULL    | auto_increment |
| name  | varchar(15)   | YES  |     | NULL    |                |
| cate  | int(11)       | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

4-2-4 色テーブル

Closet テーブルで必要となる色のテーブルを作成する。表 4-2-4-1 の内容でどんな色味なのかを登録する。データベースのテーブル設計を表 4-2-4-2 に示す。

表 4-2-4-1 : 色テーブルの登録内容

	色
1	赤系

2	ピンク系
3	紫系
4	青系
5	緑系
6	黄色系
7	オレンジ系
8	ベージュ系
9	ブラウン系
10	黒系
11	グレー系
12	白系
13	その他

表 4-2-4-2 : テーブル名 color

	フィールド名	説明	型
色番号 (通し)	color	ID	Int, auto-increment, index
名称	name	赤系、ピンク系など	varchar(10)

表 4-2-4-3 でテーブルを作成する。

表 4-2-4-3 : color テーブル作成

```
mysql> create table color (color int, name varchar(10));
Query OK, 0 rows affected (0.00 sec)

mysql> alter table color add index ind1(color);
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table color modify color int auto_increment;
```



```
Query OK, 0 rows affected (0.01 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> show fields from color;
```

```
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| color | int(11)       | NO   | MUL | NULL    | auto_increment |
| name  | varchar(10)  | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
```

```
2 rows in set (0.00 sec)
```

4-2-5 コーディネート管理テーブル

コーディネート管理するためのテーブルを作成する。データベースのテーブル設計を表 4-2-1-2 に示す。表 4-2-5-1 の内容を登録できるようにする。

コーディネートの登録は、服（アイテム）を `closet` テーブルに登録した時につけられる番号（`id`）を使って登録する。必須項目をトップス 1 とボトムスとし（ワンピースの場合はトップス 1 のみ）、オプション項目をトップス 2、アウター、靴、帽子、バック、小物とし、オプション項目はアイテムが選択されていなくても登録ができるようにする。

トップスが 2 つ登録できるのは、重ね着をすることがあるためである。ワンピースはトップス 1 に含まれ、優先的にトップス 1 に登録される。今回、見た目からでは見えないためインナーは登録項目から外している。

帽子・バック・小物のカテゴリ小物に分類されるものが 3 つ登録できるのは、帽子やバック、アクセサリなど小物に種類があり、より詳細なコーディネートを登録するためである。

表 4-2-5-1 : コーディネートテーブルの登録内容

必須項目	トップス 1（ワンピース含む） ボトムス
------	-------------------------

オプション項目 (なくても登録可能)	トップス 2 アウター 靴 帽子 バック 小物
-----------------------	--

表 4-2-5-2 : テーブル名 coordinate

	フィールド名	説明	型
番号	Id	id	Int, auto-increment, index
トップス 1	tops1	トップス	int
トップス 2	tops2	トップス	int
ボトムス	bottoms	ボトムス	int
アウター	outer1	アウター	int
靴	shoes	靴	int
帽子	hat	帽子	int
バック	bag	バック	int
小物	item	その他の小物	int
備考	biko	メモ	text
登録日	touroku	登録日	date

表 4-2-5-3 でテーブルを作成する。後から追加することがあるため、id を連番の設定をする。

表 4-2-5-3 : テーブル作成

```
mysql> create table coordinate (id int, tops1 int, tops2 int, bottoms int, outer1 int,
shoes int, hat int, bag int, item int, biko text, touroku date);
Query OK, 0 rows affected (0.00 sec)

mysql> alter table coordinate add index ind1(id);
Query OK, 0 rows affected (0.00 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> alter table coordinate modify id int auto_increment;
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> show fields from coordinate;
+-----+-----+-----+-----+-----+-----+
| Field  | Type   | Null  | Key  | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id     | int(11)| NO    | MUL  | NULL    | auto_increment |
| tops1  | int(11)| YES   |      | NULL    |                |
| tops2  | int(11)| YES   |      | NULL    |                |
| bottoms| int(11)| YES   |      | NULL    |                |
| outer1 | int(11)| YES   |      | NULL    |                |
| shoes  | int(11)| YES   |      | NULL    |                |
| hat    | int(11)| YES   |      | NULL    |                |
| bag    | int(11)| YES   |      | NULL    |                |
| item   | int(11)| YES   |      | NULL    |                |
| biko   | text   | YES   |      | NULL    |                |
| touroku| date   | YES   |      | NULL    |                |
+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)
```

4-3 実装 アイテム登録

持っている服の情報を登録するためのプログラムである。(表 4-3 を参照)

カテゴリと色をプルダウンメニューから選択し、柄の有無をチェックし、季節を選択、メモすることがあれば記入する。カテゴリ内訳はカテゴリを選択した後に選択する。すべての項目が入力されたら(メモすることがなければ入力されていなくても可能)、画像を選択し、データベースの closet に登録が完了する。

登録される情報は、カテゴリ (cate)、カテゴリ内訳 (cate_sub)、色 (color)、柄 (gara)、季節・春 (spring)、季節・夏 (summer)、季節・秋 (autumn)、季節・冬 (winter)、着用回数 (kaisuu)、登録日 (touroku)、最終着用日 (last)、画像 (img)、メモ (biko) である。着用回数は 0、最終着用日は登録日と同じ日とする。

表 4-3 : item_touroku.php

```
<html>
<head>
    <meta charset="UTF-8"/>
    <title>アイテム登録</title>
</head>
<body>
<?php
mb_language("uni");
mb_internal_encoding("utf-8");
mb_http_input("auto");
mb_http_output("utf-8");
$db = mysql_connect('localhost','a940***','*****');
mysql_query("SET NAMES utf8",$db);
mysql_select_db('db_a940166');

extract($_POST);
echo "<br>持っている服のデータを入力<br>";
$form      =      "<form      action='item_touroku.php'      method='POST'
enctype='multipart/form-data'>";
```

```

if ($cate == ""){
    $form .= "カテゴリー <br><select name='cate'>";
    $sql = "select * from cate";
    $result = mysql_query($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)){
            $form .= "<option value=\"" . $row['cate'] . "\">
$row['name'] </option>";
        }
    }
    $form .= "</select>";
} else {
    $form .= "<input type = 'hidden' name = 'cate' value =\"" . $cate . "\">";
}

if ($send <> " && $cate <> " && $cate_sub ==){
    $sql = "select * from cate where cate=\"" . $cate;
    $result = mysql_query($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 "<br>カテゴリー: " . $row['cate'] . "-" . $row['name'] .

```

```

"<br>";
    }

    $form .= "カテゴリ内訳<br><select name='cate_sub'>";

    $sql = "select * from cate_sub where cate=" . $cate;
    $result = mysql_query($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)){
            $form .= "<option value=\"" . $row['id'] . "\" .
$row['name'] .</option>";
        }

    }

    $form .="</select>";

    $form .="<input type = 'hidden' name = 'cate' value =\"" . $cate . "\">";
}

if ($cate_sub <> ""){
    $form .="<input type = 'hidden' name = 'cate_sub' value =\"" . $cate_sub . "\">";
}

if ($color == ""){
    $form .= "<br>カラー <br><select name='color'>";

    $sql = "select * from color";
    $result = mysql_query($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)){

            $form .= "<option value=\"" . $row['color'] . "\"> .
$row['name'] . "</option>";

            }

        }

        $form .="</select>";
    } else {

        $form .="<input type = 'hidden' name = 'color' value =\"" . $color . "\">";

    }

if ($gara == ""){

    $form .="<br>柄<br>";

    $form .="<input type = 'radio' name = 'gara' value = '1'>あり";

    $form .="<input type = 'radio' name = 'gara' value = '0'>なし";

} else {

    $form .="<input type = 'hidden' name = 'gara' value =\"" . $gara . "\">";

}

if ($spring == "" && $summer == "" && $autumn == "" && $winter == ""){

    $form .="<br>季節<br>";

    $form .="<input type = 'checkbox' name = 'spring' value = 1>春";

    $form .="<input type = 'checkbox' name = 'summer' value = 1>夏";

    $form .="<input type = 'checkbox' name = 'autumn' value = 1>秋";

    $form .="<input type = 'checkbox' name = 'winter' value = 1>冬";

} else {

    $season = 1;

    if($spring==""){ $spring = 0;}

    if($summer==""){ $summer = 0;}

```

```

if($autumn==""){ $autumn = 0;}
if($winter==""){ $winter = 0;}

$form .= "<input type = 'hidden' name = 'spring' value = " . $spring . ">";
$form .= "<input type = 'hidden' name = 'summer' value = " . $summer . ">";
$form .= "<input type = 'hidden' name = 'autumn' value = " . $autumn . ">";
$form .= "<input type = 'hidden' name = 'winter' value = " . $winter . ">";
}

if ($send <> '送信'){
    $form .= "<br>メモ<br>";

    $form .= "<textarea name='biko' rows='4' cols='40'></textarea><br>";
} else {
    $form .= "<input type = 'hidden' name = 'biko' value = " . $biko . ">";
}

if ($send <> " && $cate <> " && $cate_sub <> " && $color <> " && $season == 1 &&
$img == "){
    $form .= "<br>写真<br>";

    $form .= "ファイル名 : <input type='file' name='img'><br>";
}

$form .= "<br><br><input type='submit' name='send' value='送信'>
    <input type='reset' value='リセット'><br><br>
</form>";

echo $form;

$file = $_FILES[ 'img' ][ 'tmp_name' ];
$name = $_FILES[ 'img' ][ 'name' ];

if($send <> " && $name <> "){

```



```

// $file は データの一時ファイル名, $name は アップロードしたファイルの名称
// 正常にアップロードされていれば, img ディレクトリにデータを保存
    if ( $_FILES[ 'picture' ][ 'error' ] == UPLOAD_ERR_OK &&
        is_uploaded_file( $file ) ){
        move_uploaded_file( $file, 'img/' . $name );
        echo "アップロード成功！！";
    } else {
        echo "アップロード失敗！！";
    }

    $sql = "insert into closet values(0, $cate, $cate_sub, $color, $gara, $spring,
$summer, $autumn, $winter, 0, '$date', '$date', '$name', '$biko)";
    $result = mysql_query($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>";
    }
}
?>
</body>
</html>

```

4-4 実装 アイテム全件表示

登録された服の情報を一覧で表示するためのプログラムである（表 4-4 を参照）。

Id、画像、カテゴリ、カテゴリ内訳、色、柄のありなし、選択された季節、着用回数、登録日、最終着用日、メモが表示される。データベースは closet と連動している。

表 4-4 : item_zenken.php

```

<!DOCTYPE HTML>
<html lang="ja">
<head>
    <title>アイテム表示</title>
<meta charset="UTF-8" />
</head>
<body>
<?php
mb_language("uni");
mb_internal_encoding("utf-8");
mb_http_input("auto");
mb_http_output("utf-8");
$db = mysql_connect('localhost','a940***','*****');
mysql_query("SET NAMES utf8",$db);
mysql_select_db('db_a940166');

$sql= "select * from closet ";
$result = mysql_query($sql);
$rows = mysql_num_rows($result);
extract($_POST);
echo "<br>アイテム全件表示<br>";
if($rows =="){
    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["id"] . "<br>";
        echo "<img width=240 src =¥¥img/" . $row["img"] . "¥¥"><br>";
    }
}
}

```

```

echo "カテゴリ : ";
if ($row["cate"]==1){echo "トップス";}
    elseif ($row["cate"]==2){echo "ボトムス";}
    elseif ($row["cate"]==3){echo "ワンピース";}
    elseif ($row["cate"]==4){echo "アウター";}
    elseif ($row["cate"]==5){echo "靴";}
    else {echo "小物";}
echo "<br>カテゴリ内訳 : ";
if ($row["cate_sub"]==1){echo "シャツ・ブラウス";}
    elseif ($row["cate_sub"]==2){echo "カットソー";}
    elseif ($row["cate_sub"]==3){echo "ニット";}
    elseif ($row["cate_sub"]==4){echo "パーカー";}
    elseif ($row["cate_sub"]==5){echo "カーディガン";}
    elseif ($row["cate_sub"]==6){echo "ズボン";}
    elseif ($row["cate_sub"]==7){echo "スカート";}
    elseif ($row["cate_sub"]==8){echo "ワンピース";}
    elseif ($row["cate_sub"]==9){echo "コート";}
    elseif ($row["cate_sub"]==10){echo "ジャケット";}
    elseif ($row["cate_sub"]==11){echo "ブーツ";}
    elseif ($row["cate_sub"]==12){echo "サンダル";}
    elseif ($row["cate_sub"]==13){echo "パンプス";}
    elseif ($row["cate_sub"]==14){echo "スニーカー";}
    elseif ($row["cate_sub"]==15){echo "帽子";}
    elseif ($row["cate_sub"]==16){echo "バック";}
    else {echo "その他";}
echo "<br>色 : ";
if ($row["color"]==1){echo "赤系";}
    elseif ($row["color"]==2){echo "ピンク系";}
    elseif ($row["color"]==3){echo "紫系";}
    elseif ($row["color"]==4){echo "青系";}

```

```
elseif ($row["color"]==5){echo "緑系";}
elseif ($row["color"]==6){echo "黄色系";}
elseif ($row["color"]==7){echo "オレンジ系";}
elseif ($row["color"]==8){echo "ベージュ系";}
elseif ($row["color"]==9){echo "ブラウン系";}
elseif ($row["color"]==10){echo "黒系";}
elseif ($row["color"]==11){echo "グレー系";}
elseif ($row["color"]==12){echo "白系";}
else {echo "その他";}

echo "<br>柄 : ";
if ($row["gara"]==0){echo "なし";}
    elseif ($row["gara"]==1){echo "あり";}
echo "<br>季節 : ";
if ($row["spring"]==1){echo "春";}
if ($row["summer"]==1){echo "夏";}
if ($row["autumn"]==1){echo "秋";}
if ($row["winter"]==1){echo "冬";}
echo "<br>着用回数 : ".$row["kaisuu"]."<br>";
echo "最終着用日 : ".$row["last"]."<br>";
echo "登録日 : ".$row["touroku"]."<br>";
echo "メモ : ".$row["biko"]."<br>";
echo "</p><hr />";
    }
}
?>
<A href="top.htm">トップページに戻る</A><br>
<A href="item.htm">アイテムページに戻る</A>
</body>
</html>
```

4-5 実装 コーディネート登録

コーディネートを登録するためのプログラムである（表 4-5-1 を参照）。

カテゴリとカテゴリ内訳を選択すると、該当するアイテムが一覧で表示され、画像をクリックすると、アイテムが決定される。これを登録したいアイテム分繰り返し行う。コーディネートが完成すると登録するためのプログラム（表 4-5-2 参照）に移動し、データベースの coordinate に登録が完了する。

表 4-5-1 : coordinate_sakusei2.php

```
<html>
<head>
    <meta charset="UTF-8"/>
    <title>コーディネート作成</title>
</head>
<body>
<?php
mb_language("uni");
mb_internal_encoding("utf-8");
mb_http_input("auto");
mb_http_output("utf-8");
$db = mysql_connect('localhost','a940***','*****');
mysql_query("SET NAMES utf8",$db);
mysql_select_db('db_a940166');

extract($_POST);
extract($_GET);
echo "<br>コーディネートを入力<br><br>";
if($stops1 == " && $cate_sub_ >=1 && $cate_sub_ <=5 || $cate_==3){
    $stops1 = $id;
}elseif ($stops2 == " && $cate_sub_ >=1 && $cate_sub_ <=5 ){
    $stops2 = $id;}
if(($cate_==1 || $cate_==3) && !isset($stops1)){
```

```

    $tops2 = $id;}
if($cate_==2){$bottoms = $id;}
if($cate_==4){$outer1 = $id;}
if($cate_==5){$shoes = $id;}
if($cate_sub_==15){$hat = $id;}
if($cate_sub_==16){$bag = $id;}
if($cate_sub_==17){$item = $id;}

if ($tops1>0){
    $sql2[0]= "select * from closet where id = " . $tops1;
    $result = mysql_query($sql2[0]);
    $rows = mysql_num_rows($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)){
        $fn[0]= $row["img"];
    }
}
if ($tops2>0){
    $sql2[1]= "select * from closet where id = " . $tops2;
    $result = mysql_query($sql2[1]);
    $rows = mysql_num_rows($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)){
        $fn[1]= $row["img"];
    }
}
if ($bottoms>0){
    $sql2[2]= "select * from closet where id = " . $bottoms;
    $result = mysql_query($sql2[2]);
    $rows = mysql_num_rows($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)){
    $fn[2]= $row["img"];
}
}
if ($outer1>0){
    $sql2[3]= "select * from closet where id = " . $outer1;
    $result = mysql_query($sql2[3]);
    $rows = mysql_num_rows($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)){
        $fn[3]= $row["img"];
    }
}
if ($shoes>0){
    $sql2[4]= "select * from closet where id = " . $shoes;
    $result = mysql_query($sql2[4]);
    $rows = mysql_num_rows($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)){
        $fn[4]= $row["img"];
    }
}
if ($hat>0){
    $sql2[5]= "select * from closet where id = " . $hat;
    $result = mysql_query($sql2[5]);
    $rows = mysql_num_rows($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)){
        $fn[5]= $row["img"];
    }
}

```

```

    }
}
if ($bag>0){
    $sql2[6]= "select * from closet where id = " . $bag;
    $result = mysql_query($sql2[6]);
    $rows = mysql_num_rows($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)){
        $fn[6]= $row["img"];
    }
}
if ($item>0){
    $sql2[7]= "select * from closet where id = " . $item;
    $result = mysql_query($sql2[7]);
    $rows = mysql_num_rows($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)){
        $fn[7]= $row["img"];
    }
}
if ($tops1>0){
    echo "トップス : " . $tops1 . "<br>";
    echo "<img width=240 src =¥¥img/" . $fn[0]. "¥¥"><br>";
} else {echo "トップス : 未選択<br>";}
if ($tops2>0){
    echo "トップス : " . $tops2 . "<br>";
    echo "<img width=240 src =¥¥img/" . $fn[1]. "¥¥"><br>";
} else {echo "トップス : 未選択<br>";}
if ($bottoms>0){

```



```

        echo "ボトムス : " . $bottoms . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[2]. "¥"><br>";
    } else {echo "ボトムス : 未選択<br>";}
    if ($outer1>0){
        echo "アウター : " . $outer1 . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[3]. "¥"><br>";
    } else {echo "アウター : 未選択<br>";}
    if ($shoes>0){
        echo "靴 : " . $shoes . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[4]. "¥"><br>";
    } else {echo "靴 : 未選択<br>";}
    if ($hat>0){
        echo "帽子 : " . $hat . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[5]. "¥"><br>";
    } else {echo "帽子 : 未選択<br>";}
    if ($bag>0){
        echo "バック : " . $bag . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[6]. "¥"><br>";
    } else {echo "バック : 未選択<br>";}
    if ($item>0){
        echo "小物 : " . $item . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[7]. "¥"><br>";
    } else {echo "小物 : 未選択<br>";}
    if ($biko <>){echo "メモ : " . $biko . "<br>";}else{echo "メモ : <br><hr />";}

    $form = "<form action='coordinate_sakusei2.php' method='POST' >";

    if ($cate== " "){
        $form .= "カテゴリー <br><select name='cate'>";
        $sql = "select * from cate";
    }

```

```

$result = mysql_query($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)){
    $form .= "<option value=\"" . $row['cate'] . "\">
$row['name'] .</option>";}
    }$form .="</select>";
} else {$form .="<input type = 'hidden' name = 'cate' value = " . $cate . ">";}

if ($cate <> " ){
    $sql = "select * from cate where cate=" . $cate;
    $result = mysql_query($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 "<br>カテゴリー: " . $row['cate'] . "-" . $row['name'] . "<br>";}
    }
    $form .= "カテゴリー内訳<br><select name='cate_sub'>";
    $sql = "select * from cate_sub where cate=" . $cate;
    $result = mysql_query($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)){
        $form .= "<option value=\"" . $row['id'] . "\"> " . $row['name'] . "</option>";}
    }
    $form .="</select>";
    $form .="<input type = 'hidden' name = 'cate' value = " . $cate . ">";
    //$form .="<input type = 'hidden' name = 'cate_sub' value = " . $cate_sub . ">";
}

if ($cate_sub <> "){

```

```

$form .= "<input type = 'hidden' name = 'cate_sub' value = " . $cate_sub . "'>";

if ($cate <>" && $cate_sub <>" ) {
//今日から 3 日前・7 日前の日にちを求める

    $kikan = date('y-m-d', strtotime("-3 days"));
    $lastweek = date('y-m-d', strtotime("-7 days"));

    $sql = "select * from closet where cate=" . $cate . " and cate_sub=" . $cate_sub . "
and (last <=" . $kikan . " or last =" . $lastweek . ")";
//ラストが今日-3 より小さい、またはラストが今日-7 の場合

    $result = mysql_query($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>" . $row["id"] . "<br>";

        echo  "<a href=¥"coordinate_sakusei2.php?id=" . $row["id"] .
"&cate_" . $row["cate"] . "&cate_sub_" . $row["cate_sub"]. "&tops1=" . $tops1 .
"&tops2=" . $tops2 .
"&bottoms=" . $bottoms . "&outer1=" . $outer1 . "&shoes=" . $shoes . "&hat=" . $hat . "&bag
=" . $bag . "&item=" . $item . "&biko=" . $biko . "¥">";

        echo "<img width=240 src =¥"img/" . $row["img"]. "¥"></a><br>";

        echo "カテゴリ : ";

        if ($row["cate"]==1){echo "トップス";}

            elseif ($row["cate"]==2){echo "ボトムス";}

            elseif ($row["cate"]==3){echo "ワンピース";}

            elseif ($row["cate"]==4){echo "アウター";}

            elseif ($row["cate"]==5){echo "靴";}

            else {echo "小物";}

        echo "<br>カテゴリ内訳 : ";

        if ($row["cate_sub"]==1){echo "シャツ・ブラウス";}

            elseif ($row["cate_sub"]==2){echo "カットソー";}

```

```

elseif ($row["cate_sub"]==3){echo "ニット";}
elseif ($row["cate_sub"]==4){echo "パーカー";}
elseif ($row["cate_sub"]==5){echo "カーディガン";}
elseif ($row["cate_sub"]==6){echo "ズボン";}
elseif ($row["cate_sub"]==7){echo "スカート";}
elseif ($row["cate_sub"]==8){echo "ワンピース";}
elseif ($row["cate_sub"]==9){echo "コート";}
elseif ($row["cate_sub"]==10){echo "ジャケット";}
elseif ($row["cate_sub"]==11){echo "ブーツ";}
elseif ($row["cate_sub"]==12){echo "サンダル";}
elseif ($row["cate_sub"]==13){echo "パンプス";}
elseif ($row["cate_sub"]==14){echo "スニーカー";}
elseif ($row["cate_sub"]==15){echo "帽子";}
elseif ($row["cate_sub"]==16){echo "バック";}
else {echo "その他";}

echo "<br>色 : ";

if ($row["color"]==1){echo "赤系";}
elseif ($row["color"]==2){echo "ピンク系";}
elseif ($row["color"]==3){echo "紫系";}
elseif ($row["color"]==4){echo "青系";}
elseif ($row["color"]==5){echo "緑系";}
elseif ($row["color"]==6){echo "黄色系";}
elseif ($row["color"]==7){echo "オレンジ系";}
elseif ($row["color"]==8){echo "ベージュ系";}
elseif ($row["color"]==9){echo "ブラウン系";}
elseif ($row["color"]==10){echo "黒系";}
elseif ($row["color"]==11){echo "グレー系";}
elseif ($row["color"]==12){echo "白系";}
else {echo "その他";}

echo "<br>柄 : ";

```

```

        if ($row["gara"]==0){echo "なし";}
            elseif ($row["gara"]==1){echo "あり";}
        echo "<br>季節 : ";
        if ($row["spring"]==1){echo "春";}
        if ($row["summer"]==1){echo "夏";}
        if ($row["autumn"]==1){echo "秋";}
        if ($row["winter"]==1){echo "冬";}
        echo "<br>着用回数 : ".$row["kaisuu"]."<br>";
        echo "登録日 : ".$row["touroku"]."<br>";
        echo "最終着用日 : ".$row["last"]."<br>";
        echo "メモ : ".$row["biko"]."<br>";
        echo "</p><hr />";
    }
}
if ($send <> '決定'){
    $form .="<br><br>メモ<br>";
    $form .="<textarea name='biko' rows='4' cols='40'></textarea><br>";
} else {
    $form .="<input type = 'hidden' name = 'biko' value ="" . $biko . ""><br>";
}

$form .="<input type = 'hidden' name = 'tops1' value ="" . $tops1 . "">";
$form .="<input type = 'hidden' name = 'tops2' value ="" . $tops2 . "">";
$form .="<input type = 'hidden' name = 'bottoms' value ="" . $bottoms . "">";
$form .="<input type = 'hidden' name = 'outer1' value ="" . $outer1 . "">";
$form .="<input type = 'hidden' name = 'shoes' value ="" . $shoes . "">";
$form .="<input type = 'hidden' name = 'hat' value ="" . $hat . "">";
$form .="<input type = 'hidden' name = 'bag' value ="" . $bag . "">";
$form .="<input type = 'hidden' name = 'item' value ="" . $item . "">";
$form .="<input type = 'hidden' name = 'biko' value ="" . $biko . "">";

```

```

$form .= "<br><br><input type='submit' name='send' value='送信'>
        <input type='reset' value='リセット'><br><br>
        </form>";
echo $form;

if(($tops1 <>" && bottoms <>") || $cate_ ==3){
    echo "<a href=¥"coordinate_sakusei3.php?id=" . $row["id"] . "&cate_=" .
$row["cate"] . "&cate_sub_=" . $row["cate_sub"] . "&tops1=" . $tops1 . "&tops2=" . $tops2 .
"&bottoms=" . $bottoms . "&outer1=" . $outer1 . "&shoes=" . $shoes . "&hat=" . $hat . "&bag
=" . $bag . "&item=" . $item . "&biko=" . $biko . "¥">決定</a><br><hr />";}
?>
<br><A href="top.htm">トップページに戻る</A><br>
<A href="coordinate.htm">コーディネートページに戻る</A>
</body>
</html>

```

表 4-5-2 : coordinate_sakusei3.php

```

<html>
<head>
    <meta charset="UTF-8"/>
    <title>コーディネート登録</title>
</head>
<body>
<?php
mb_language("uni");
mb_internal_encoding("utf-8");
mb_http_input("auto");
mb_http_output("utf-8");
$db = mysql_connect('localhost','a940***','*****');

```

```

mysql_query("SET NAMES utf8",$db);
mysql_select_db('db_a940166');

extract($_POST);
extract($_GET);

$kyou = date('y-m-d');

$sql ="insert into coordinate values(0, '$stops1', '$stops2', '$bottoms', '$outer1', '$shoes',
'$hat', '$bag', '$item', '$biko', '$kyou)";
$result = mysql_query($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><br>";
    echo "<A href=top.htm>トップページに戻る</A><br>";
    echo "<A href=coordinate.htm>コーディネートページに戻る</A><br><br>";
}

if($stops1 =='' && $cate_sub_ >=1 && $cate_sub_ <=5 || $cate_==3){
    $stops1 = $id;
}elseif ($stops2 =='' && $cate_sub_ >=1 && $cate_sub_ <=5 ){
    $stops2 =$id;}
if(($cate_==1 || $cate_==3) && !isset($stops1)){
    $stops2 =$id;}
if($cate_==2){$bottoms =$id;}
if($cate_==4){$outer1 =$id;}
if($cate_==5){$shoes =$id;}

```

```

if($cate_sub_==15){$hat =$id;}
if($cate_sub_==16){$bag =$id;}
if($cate_sub_==17){$item =$id;}

if ($tops1>0){
    $sql2[0]= "select * from closet where id = " . $tops1;
    $result = mysql_query($sql2[0]);
    $rows = mysql_num_rows($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)){
        $fn[0]= $row["img"];
    }
}
if ($tops2>0){
    $sql2[1]= "select * from closet where id = " . $tops2;
    $result = mysql_query($sql2[1]);
    $rows = mysql_num_rows($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)){
        $fn[1]= $row["img"];
    }
}
if ($bottoms>0){
    $sql2[2]= "select * from closet where id = " . $bottoms;
    $result = mysql_query($sql2[2]);
    $rows = mysql_num_rows($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)){
        $fn[2]= $row["img"];
    }
}

```



```

}
if ($outer1>0){
    $sql2[3]= "select * from closet where id = " . $outer1;
    $result = mysql_query($sql2[3]);
    $rows = mysql_num_rows($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)){
        $fn[3]= $row["img"];
    }
}
}
if ($shoes>0){
    $sql2[4]= "select * from closet where id = " . $shoes;
    $result = mysql_query($sql2[4]);
    $rows = mysql_num_rows($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)){
        $fn[4]= $row["img"];
    }
}
}
if ($hat>0){
    $sql2[5]= "select * from closet where id = " . $hat;
    $result = mysql_query($sql2[5]);
    $rows = mysql_num_rows($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)){
        $fn[5]= $row["img"];
    }
}
}
if ($bag>0){
    $sql2[6]= "select * from closet where id = " . $bag;

```

```

$result = mysql_query($sql2[6]);
$rows = mysql_num_rows($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)){
    $fn[6]= $row["img"];
}
}
if ($item>0){
    $sql2[7]= "select * from closet where id = " . $item;
    $result = mysql_query($sql2[7]);
    $rows = mysql_num_rows($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)){
        $fn[7]= $row["img"];
    }
}
if ($tops1>0){
    echo "トップス : " . $tops1 . "<br>";
    echo "<img width=240 src =¥"img/" . $fn[0]. "¥"><br>";
} else {echo "トップス : 未選択<br>";}
if ($tops2>0){
    echo "トップス : " . $tops2 . "<br>";
    echo "<img width=240 src =¥"img/" . $fn[1]. "¥"><br>";
} else {echo "トップス : 未選択<br>";}
if ($bottoms>0){
    echo "ボトムス : " . $bottoms . "<br>";
    echo "<img width=240 src =¥"img/" . $fn[2]. "¥"><br>";
} else {echo "ボトムス : 未選択<br>";}
if ($outer1>0){

```

```

        echo "アウター : " . $outer1 . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[3]. "¥"><br>";
    } else {echo "アウター : 未選択<br>";}
    if ($shoes>0){
        echo "靴 : " . $shoes . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[4]. "¥"><br>";
    } else {echo "靴 : 未選択<br>";}
    if ($hat>0){
        echo "帽子 : " . $hat . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[5]. "¥"><br>";
    } else {echo "帽子 : 未選択<br>";}
    if ($bag>0){
        echo "バック : " . $bag . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[6]. "¥"><br>";
    } else {echo "バック : 未選択<br>";}
    if ($item>0){
        echo "小物 : " . $item . "<br>";
        echo "<img width=240 src =¥"img/" . $fn[7]. "¥"><br>";
    } else {echo "小物 : 未選択<br>";}
    if ($biko <>){echo "メモ : " . $biko . "<br>";}else{echo "メモ : <br>";}
?>
</body>
</html>

```

4-6 実装 除外条件 (3日前と1週間前に登録されたアイテムを表示しない)

登録日から3日前までに使用したアイテムと、7日前に使用したアイテムをコーディネートのアイテム選択時に表示させないための機能である。アイテムが最後にいつ着たかというのは、アイテムの最終着用日 (closet の last) のデータを使用する。

表示方法としては、今日から3日前の日をちと7日前の日をちを調べその日にちとアイテムの最終着用日を比較して表示させるかどうかを決める。例えば、今日が2013年1月

20日だったら、3日前は2013年1月17日で、7日前は2013年1月13日である。アイテムの最終着用日が2013年1月18日の場合は一覧に表示されず、2013年1月16日の場合は表示される。

PHP

```
$kikan = date('Y-m-d', strtotime("-3 days"));  
$lastweek = date('Y-m-d', strtotime("-7 days"));  
$sql = "select * from closet where cate=" . $cate . " and cate_sub=" . $cate_sub . " and (last  
<= ". $kikan . " or last = ". $lastweek . "));"
```

5.評価と考察

5-1 動作検証

本システムが正常に動作するかの確認を行う。

5-1-1 トップページ

トップページ (top.htm) を開くと図 5-1-1 の画面が表示される。アイテムをクリックすると、アイテムページ (item.htm) に、コーディネートページ (coordinate.htm) に移動する。

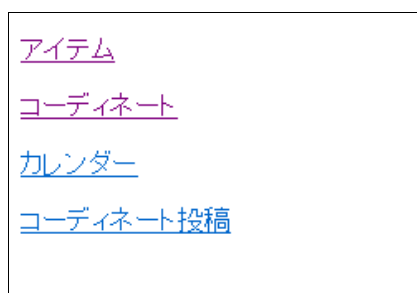


図 5-1-1-1 : トップページ画面 (top.htm)

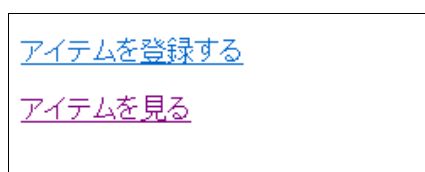


図 5-1-1-2 : アイテムページ画面 (item.htm)

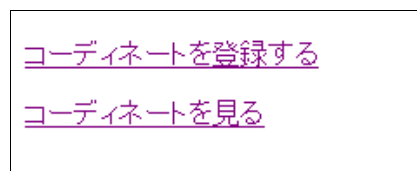


図 5-1-1-3 : コーディネートページ画面 (coordinate.htm)

5-1-2 アイテム登録ページ

アイテムの登録画面である。トップページ画面 (図 5-1-1-1 参照) から「アイテム」をクリックすると、アイテムページ画面 (図 5-1-1-2 参照) に移動する。「アイテムを登録する」をクリックすると、アイテム登録ページの画面 (図 5-1-2-1 参照) に移動する。カテゴリと色をプルダウンメニューから選択 (図 5-1-2-2 参照) し、柄の有無と季節を選択しメモすることがあれば記入 (図 5-1-2-3 参照) をする。送信をクリックすると、カテゴリ内訳の選択 (図 5-1-2-4 参照) をする。送信をクリックすると、アイテム画像をフォルダ一から参照 (図 5-1-2-5 参照) をする。画像を参照し、送信をクリックすると、登録が完

了（図 5-1-2-6 参照）する。

持っている服のデータを入力

カテゴリ
トップス

カラー
赤系

柄
あり なし

季節
春 夏 秋 冬

メモ

[トップページに戻る](#)
[アイテムページに戻る](#)

図 5-1-2-1 : アイテム登録ページ画面

持っている服のデータを入力

持っている服のデータを入力

カテゴリ
靴

カラー
赤系
ピンク系
紫系
青系
緑系
黄色系
オレンジ系
ベージュ系
ブラウン系
黒系
グレー系
白系
その他

季節
秋 冬

図 5-1-2-2 : カテゴリ・色の選択画面

柄
あり なし

季節
春 夏 秋 冬

メモ

図 5-1-2-3 : 柄・季節の選択、メモの記入画面

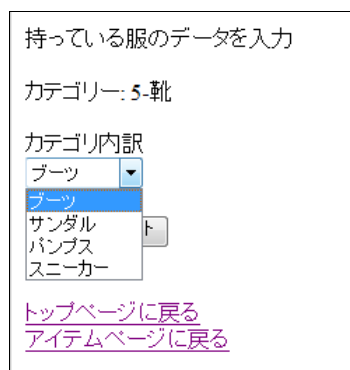


図 5-1-2-4 : カテゴリ内訳の選択画面

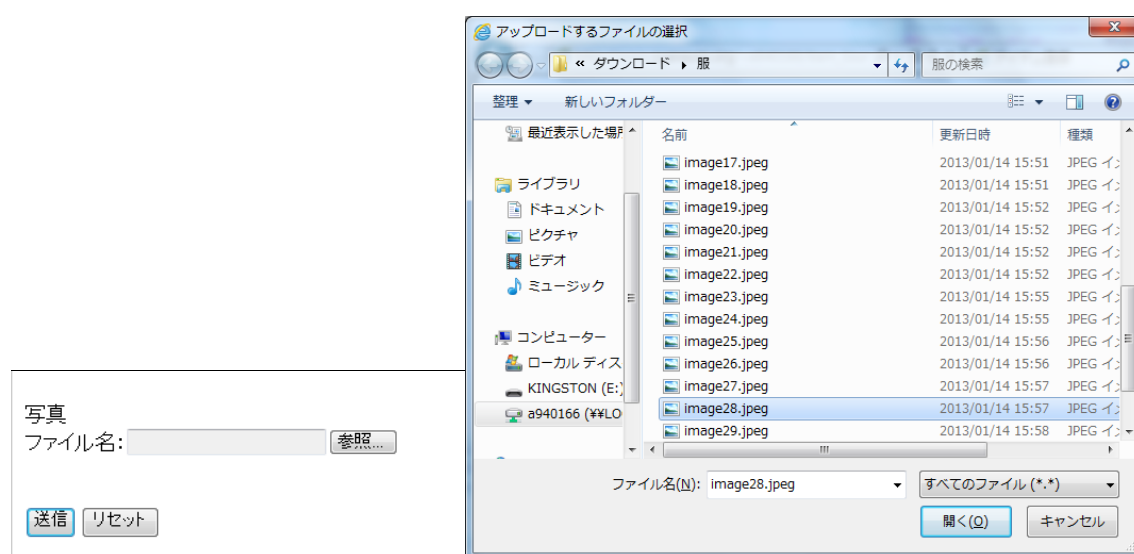


図 5-1-2-5 : アイテム画像の選択画面



図 5-1-2-6 : アイテム登録完了画面

5-1-3 アイテム全件表示

アイテムページ画面（図 5-1-1-2 参照）から「アイテムを見る」をクリックすると、アイテムの全件が表示される。（図 5-1-3 参照）アイテムの登録順に見ることができる。

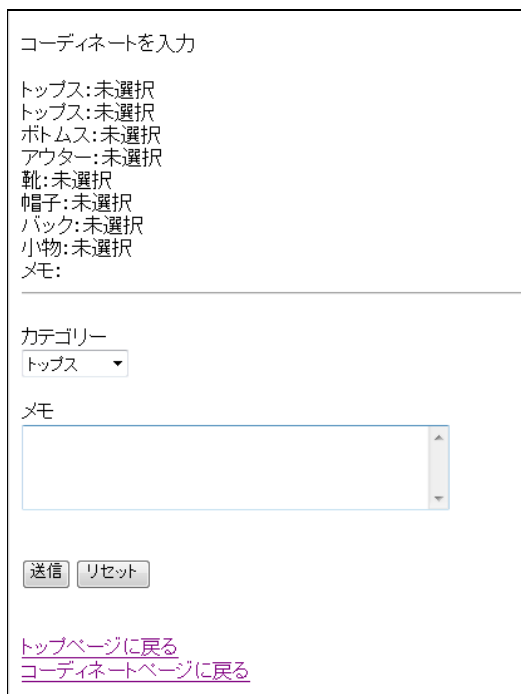


図 5-1-3 : アイテム全件表示画面

5-1-4 コーディネート作成

コーディネートページ画面（図 5-1-1-3 参照）から「コーディネートに登録する」をクリックすると、コーディネート入力画面（図 5-1-4-1 参照）に移動する。カテゴリを選択し送信をクリックしカテゴリ内訳を選択する（図 5-1-4-2 参照）と、選択した内容で登録されているアイテムが一覧で表示（図 5-1-4-3 参照）される。アイテム画像をクリックすると、アイテムが決定され入力画面上部の未選択にアイテム id と画像が表示（図 5-1-4-4

参照) される。カテゴリの選択からアイテムの決定まで、登録したいアイテム分繰り返し行う。コーディネートが決まったら、送信ボタンの下にある決定ボタンをクリックすると、コーディネートの登録が完了 (図 5-1-4-5 参照) する。



コーディネートを入力

トップス:未選択
トップス:未選択
ボトムス:未選択
アウター:未選択
靴:未選択
帽子:未選択
バック:未選択
小物:未選択
メモ:

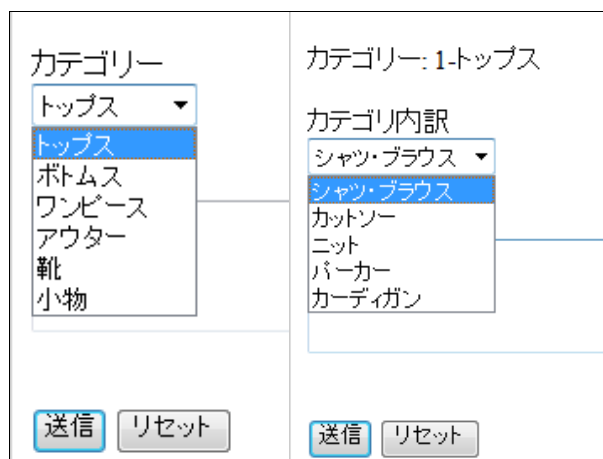
カテゴリ
トップス

メモ

送信 リセット

[トップページに戻る](#)
[コーディネートページに戻る](#)

図 5-1-4-1 : コーディネート入力画面



カテゴリ
トップス

トップス
ボトムス
ワンピース
アウター
靴
小物

送信 リセット

カテゴリ: 1-トップス

カテゴリ内訳
シャツ・ブラウス
シャツ・ブラウス
カットソー
ニット
パーカー
カーディガン

送信 リセット

図 5-1-4-2 : カテゴリ・カテゴリ内訳選択画面

8



カテゴリ: トップス
 カテゴリ内訳: シャツ・ブラウス
 色: 青系
 柄: なし
 季節: 春秋冬
 着用回数: 0
 登録日: 2013-01-14
 最終着用日: 2013-01-14
 メモ:

9



図 5-1-4-3 : 選択された内容のアイテム一覧表示画面

コーディネートを入力

トップス: 8



トップス: 未選択
 ボトムス: 未選択
 アウター: 未選択
 靴: 未選択
 帽子: 未選択
 バッグ: 未選択
 小物: 未選択
 メモ:



アウター: 未選択
 靴: 34



帽子: 未選択
 バッグ: 未選択
 小物: 未選択
 メモ:

図 5-1-4-4 : アイテム決定時の画面

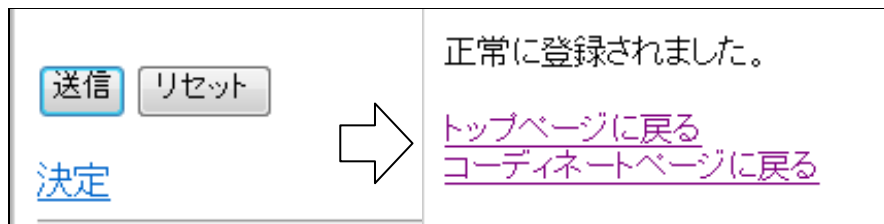


図 5-1-4-5 : コーディネート登録完了画面

5-1-5 除外条件

除外する機能が正しい仕組みかどうか確認を行う。アイテムの最終着用日は 2013 年 1 月 14 日である。カテゴリが 1、カテゴリ内訳が 1 のアイテムで 3 日前の日にちが 2013 年 1 月 16 日の場合を表 5-1-6-1 に示す。実行結果はカテゴリ (cate) が 1、カテゴリ内訳 (cate_sub) が 1、最終着用日 (last) が 2013-01-14 となっており、2013 年 1 月 16 日より古い日付のアイテムが表示されていて、正しく機能が動作している。

表 5-1-5 : 3 日前の日にちが 2013 年 1 月 16 日の場合の実行結果

```
mysql> select * from closet where cate=1 and cate_sub=1 and last <= "2013-01-16";
```

id	cate	cate_sub	color	gara	spring	summer	autumn	winter	kaisuu
8	1	1	4	0	1	0	1	1	1
0	2013-01-14	2013-01-14	image.jpeg						
9	1	1	12	0	1	0	1	1	1
0	2013-01-14	2013-01-14	image1.jpeg						
16	1	1	1	1	0	1	0	0	0
0	2013-01-14	2013-01-14	image8.jpeg						

3 rows in set (0.00 sec)

5-2 機能評価

設計目標だった機能を評価したものを表 5-2 に示す。

表 5-2：実装結果一覧

番号	設計目標の機能	実装状況
1	アイテムの登録	○
2	アイテムの表示	○
3	コーディネート登録（アイテムからの選択）	○
4	コーディネート登録（画像から）	×
5	コーディネートの表示	△
6	カレンダー表示	△
7	アイテムの提案機能	△

1 アイテムの登録

設計目標は、アイテムの詳細な情報を登録することである。アイテムのカテゴリや色・柄・着用季節の基本情報に加え着用回数や履歴を登録することができるようになった。また同時に画像も登録することができ、よりアイテムの情報を把握し分かりやすくなった。基本的な情報に加えアイテムの画像も登録できたことによって、アイテムの詳細な情報を登録することができ、正確な情報を把握することが可能になった。

2 アイテムの表示

設計目標は、カテゴリを選択し表示させることと、アイテムを着用回数や最終着用日などで並び替えて表示することである。登録したアイテムを登録順に表示することができた。しかし、カテゴリを選択し選んだカテゴリ内のアイテムを表示することと、アイテムを並び替えて表示することができなかった。カテゴリを選択して表示させると、カテゴリ内のアイテムが見やすくなり、並び替えができると、アイテムの検索が簡単にできるようになる。すばやく様々な表示方法で検索できた方が、買い物に行ったときに似た服を持っていないか確認が簡単にできるので使いやすくなる。

3 コーディネートの登録（アイテムからの選択）

設計目標では、オリジナルで考えていた除外条件を組み込んで条件に当てはまらないア

アイテムを表示しコーディネートを作成することであった。カテゴリを選択しアイテムを表示させ選んでいき、コーディネートを登録することができた。除外条件の評価は7で行う。作成時に選んだアイテムの画像が表示されるので、選んだアイテムを考慮してコーディネートを考えることが可能になった。夏にアウターは着ないので、未選択のカテゴリがあるときでも登録することができた。しかし、登録時に使用したアイテムの着用回数と最終着用日の更新ができなかった。更新ができると、アイテムの使用履歴がわかり、アイテムからどんなコーディネートをしたかの検索もできるようになる。

4 コーディネートの登録（画像から）

コーディネートの登録方法をアイテムから選択し作成する方法（3のアイテム選択からの登録）と、完成したコーディネート画像を登録し使用したアイテムを入力する方法の2通りを考えていた。設計目標で、すでにコーディネートが決まっているときに、1つ1つアイテムを選択していくことが手間になると思い、コーディネート画像で簡単に登録することを可能にすることだった。しかし、コーディネートの画像を使って登録をする方法の実装ができなかった。この登録方法ができるようになると、作成し登録する時間がなかった時でもコーディネートを残すことができるので、継続的にコーディネートの履歴を残すことが可能になる。

5 コーディネートの表示

登録したコーディネートの画像を表示することを目標としていたが、アイテムの登録番号(id)は表示できたが、画像を表示することができなかった。登録してあるカテゴリのアイテムの登録番号を表示し、登録していないカテゴリを非表示にすることができた。アイテムの画像が表示できれば登録されたコーディネートが見やすくなる。登録日を指定することができれば、先の予定で着たいコーディネートを予約することができ、その日に合わせて着たい服を避けることができる。

6 カレンダー表示

カレンダーを表示することができた。しかし設計目標であった予定の管理とコーディネートを表示登録することはできなかった。カレンダーでコーディネートを表示することができると、5のコーディネートの表示を見なくても、いつどんなコーディネートをしたか

分かりやすくなる。同時に予定を入力することができると、予定の確認が簡単にでき予定に合わせてコーディネートを考えることが可能になる。カレンダーから先の予定で着たいコーディネートを登録できると、その日に合わせて着たい服を避けることができる。

7 アイテム提案機能

避けたい期間内に使用したアイテムをコーディネート登録のアイテム選択の際に一覧の中に表示しないようにすることを目標とした。登録する日から3日前までと7日前に使用したアイテムを表示されないようにする仕組みが可能になった。季節外のアイテムを表示されないようにすることはできなかった。表示方法も着用回数が少ない順と最終着用日が遠い順を選択できるようにしたかったが、登録順に表示することしかできなかった。よって評価は△とする。

5-3 考察

本研究の目的は、所有している衣服の情報管理と、過去のコーディネートを管理することで、毎日のコーディネートを決める時の服選びに悩む時間の短縮につなげることである。目標としていた機能すべてを実装することはできなかったが、アイテムの登録、アイテム選択からのコーディネートの登録、コーディネート登録日から3日前までに使用したアイテムの非表示にする仕組みを実装することができた。

アイテムの登録機能は、アイテムの詳細な情報と画像を登録することができ、登録順にアイテムを表示することができた。コーディネートの登録機能は、登録アイテムを選択してコーディネートを作成することができ、登録したコーディネートを登録順に表示することができた。コーディネートを表示するとき、選択していないカテゴリを表示しないことができた。コーディネート作成時に、登録日から3日前までに使用したアイテムを一覧で表示させない仕組みができた。しかし、コーディネートを登録するときに、アイテムの着用履歴と着用回数の更新ができていない。

本システムの実装により、コーディネートを考えるときに同じ服やコーディネートを頻繁に着たくないが覚えていられないという問題で、覚えていなくてもアイテム選択時に表示されないため選択することがなくなり、結果的に同じ服を頻繁に着ることが避けることができる仕組みを構築できた。

今回実装できなかったカレンダーができていれば、カレンダー式でコーディネートの確

認が一目でわかり予定を考慮してコーディネートを考えることが可能になる。さらにソーシャル化ができれば、他ユーザーとコーディネートを共有しアドバイスやコミュニケーションをとることができ、さらにコーディネートを楽しむことができるようになる。これらの機能を実装し、より良いアプリケーションを目指し取り組むべきである。

6.まとめと今後の課題

6-1 まとめ

本研究の目的は、所有している衣服の情報管理と、過去のコーディネート进行管理することで、毎日のコーディネートを決める時の服選びに悩む時間の短縮につなげることを目指した。現状のアイテムの決定方法からコーディネートが完成するまでの手順、同じアイテムをどのくらいの間隔で着用するのかを雑誌の着回しの企画で調べ、既存のアプリケーションの機能を分析し問題点が明らかになった。そして問題点を解決するために必要な機能を検討し、設計から実装を行い評価した。

問題点として、服選びで悩むのは、人は同じ服を3日は着ることを避けたいと思っているため、既存のアプリケーションでは服の着た履歴情報を記憶できる機能はないということである。この問題を解決するために、服の着た履歴を管理し、コーディネートを登録するときに3日前までに使用したアイテムは一覧に表示させない仕組みを構築した。この仕組みにより、アイテム選択時に表示されず、結果的に同じ服を3日は使用することを避けることが可能になった。

また、所有衣服の詳細な情報を管理することで、外出先や買い物時に確認ができ、 unnecessaryな買い物を避けることができ満足度の高い買い物につなげることも可能になった。コーディネートを管理することで、コーディネートに使用したアイテムが明確になり日付とともに管理することが可能になった。

目標であったコーディネート時の服選びに悩む時間を短縮につなげる仕組みを作ることができた。

6-2 今後の課題

今回実装でできなかった機能の完成を目指す。画像からコーディネートを登録する機能、カレンダーで予定の管理とコーディネートを表示する機能の2つである。また、今後取り入れたいのは、ソーシャル化である。

(1) 画像からのコーディネート登録

時間がないときやすでにコーディネートが決まっているときに利用でき、アイテムから選択しコーディネートを登録する手間を減らすことができる。よってアイテムを選ぶのが面倒だから登録しないということがなくなり、継続的にコーディネートの管理が可能になる。

(2) カレンダー

予定とともにコーディネートが管理できることを目指す。予定の入力ができる、予定を確認しなくて済み、予定を考慮してコーディネートすることができる。カレンダーでコーディネートを表示することができる、一覧でコーディネートを表示しなくても見やすくなる。

(3) ソーシャル化

ソーシャル化し、他ユーザーとコミュニケーションをとることを目指す。他ユーザーとコーディネート共有することができる、今までの自分のコーディネートになかったアイデアが見つかり、コーディネートの幅を広げることができる。また、コメントとしてアドバイスをもらえたりしたりもできるので、さらにアイデアを取り入れることができる。コーディネートの幅が広がり新しいアイデアを取り入れることができ、コーディネートをさらに楽しむことができる。

以上を今後の課題とし、理想のアプリケーションを実現するためにも、改善に向けて継続して検討していく。結果的にさらにコーディネート楽しむことができるアプリケーションになるように目指していく。

参考文献

【1】 mina

主婦の友社 2012/6月号 2013/1月号

【2】 non-no

集英社 2011/12月号 2012/6月号

【3】 Real Closet

<https://itunes.apple.com/jp/app/riarukurozetto-yang-fu-guan/id404137059?mt=8>

2012年9月8日

【4】 CODENOTE

<https://itunes.apple.com/jp/app/codenote-fasshonkodineto-gong/id446321633?mt=8>

2012年9月13日

【5】 My365

<https://itunes.apple.com/jp/app/my365-1ri1mei-xie-zhenwokarendani/id472931728?mt=8>

2012年9月11日

【6】 日時を取得する(time, date)

<http://www.phpbook.jp/func/date/index1.html>

2013年1月12日

【7】 PHPによる日付・時刻・時間の計算・演算

<http://nob-log.info/2011/11/02/php%E3%81%AB%E3%82%88%E3%82%8B%E6%97%A5%E4%BB%98%E3%83%BB%E6%99%82%E5%88%BB%E3%83%BB%E6%99%82%E9%96%93%E3%81%AE%E8%A8%88%E7%AE%97%E3%83%BB%E6%BC%94%E7%A9%E97/>

2013年1月12日

【8】 岡野愛美 「コーディネート支援収納ボックスの構築」 2011年度卒業研究

謝辞

本研究を行うにあたり、ご指導いただいた渡辺 恭人教授に大変感謝しております。論文の構成をはじめ、プログラミングや執筆に至るまでとても親身にご指導して頂きました。よって、より本来の設計に近い実装が可能となり、実力以上の結果を出すことができました。本研究のテーマ決定時には不安もありましたが、最後まで検討できたのは、渡辺教授のお陰です。本当にありがとうございました。

渡辺ゼミでは、プログラミングを基礎から教えていただき、思った通りに実行できた時のプログラミングの楽しさと、1字でも間違えると正しく動作しないプログラミングの難しさを知ることができました。

最後に、私の卒業研究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。