

# GoodsReview のシステム内容について (2021 年 7 月 26 日版)

平岩 健

**概要：**本稿は、GoodsReview のシステム内容をまとめたものである。大きく分けると、機能の概要、プログラミング言語やデータベースなどの使用した技術、データベース設計、画面遷移などをまとめたシステム設計、Apache や PHP の環境設定という流れになっている。

## About Goods Review Application (version 2021/7/26)

HIRAIWA KEN

### 1. 機能概要

GoodsReview は、ユーザーが登録した商品に対して、他のユーザーが評価を投稿する Web アプリケーションである。この機能を利用するためには、新規登録画面でユーザー名とパスワードを登録する必要がある。登録が完了すると、ログインが可能となる。トップページは、日用品や電化製品などのジャンルを選ぶページである。ジャンルを選択するとユーザーが登録した商品の一覧が表示されている。商品をクリックすると、レビュー画面に遷移する。レビュー画面では、ユーザーのコメントや、総合評価が表示される。

以上が GoodsReview の機能概要である。

### 2. 利用技術

#### 2.1 PHP

PHP(Personal Home Page) は 1995 年に生まれたプログラミング言語である。PHP は主に Web アプリケーションの開発で使われており、ハイパーテキストを生成したり、動的な画面を作るのに適している。実行速度や実装の容易さからよく利用される言語となっている。

PHP が、Web アプリケーションの開発で多く利用される理由は大きく 3 つある。一つ目は、html ファイルの中に PHP 専用タグを埋め込むことにより、プログラムをその

まま記述することができることである。これにより、コード量が少ない PHP のプログラムに対して新しくファイルを作る必要がなくなり、ファイルの量を減らすことができる。

二つ目は、データベースとの連携が可能なことである。PHP の大きな特徴として、データベースとの相性の良さが挙げられる。データベース管理システムで使用できる関数が一通り標準で搭載されているため、データベースと連携させることが容易である。

三つ目は、フレームワークが多くあることである。フレームワークを活用することでプログラミングを簡略化することができ、作業の効率を向上させることが出来る。PHP の代表的なフレームワークとして「CakePHP」「ZendFramework」、「Laravel」などが挙げられる。

#### 2.2 MariaDB/MySQL

データベースとは、コンピュータ上で集積・整理された情報群のことである。データベースには種類があり、PHP と最も親和度が高いのは MySQL 互換の MariaDB である。MariaDB はフリーでオープンソースであることや、MySQL に対する高い互換性から、Linux 系の OS が標準でサポートするデータベースが MySQL から MariaDB に変更された。MariaDB は、世界で最も利用されているデータベースであり、高速で動作し、ストレージエンジンを変

更できることが特徴である。

データベースには、階層型、ネットワーク型、リレーショナル型がある。現在は、リレーショナルデータベースが広く普及している。リレーショナルデータベースの管理システムをRDBMS(Relational Database Management System)といい、データを表形式で保存し、各表ごとの関係も保存できる。データベースにデータを保存しておくことでデータの種類が増えてきても、決まった手順でデータの保存、検索、更新、削除処理をすることが出来る。

## 2.3 HTML

HTML(Hyper Text Markup Language)とは、マークアップ言語の一つである。HTMLは、Webページに文字や画像を配置することが出来る。記述方法は、タイトルや見出し、段落箇条書きなど文書が持つ内容を、タグと呼ばれる特殊な文字列で囲む形式である。

## 2.4 CSS

CSS(Cascading Style Sheets)は、HTMLの要素をどのように装飾するかを指定する言語である。Webページが表示される時の色、サイズ、レイアウトなどの表示スタイルや、プリンタなどの機器で印刷による出力スタイル、音声で読み上げられる際の再生スタイルなどのように、Webページの表示、出力、再生するかについても指定することが可能である。

## 2.5 JavaScript

JavaScriptとは、動的なWebページを作るためのプログラミング言語である。HTMLやCSSなど、他のプログラミング言語と組み合わせて使うことで、ブラウザ上でアニメーションを動作させることが出来ます。近年では、Node.jsという、ブラウザ側だけでなくサーバー側でも動くJavaScriptも普及している。

## 2.6 HTTP

http(HyperText Transfer Protocol)とは、HTMLなどのハイパーメディア文章を転送するためのアプリケーション層プロトコルである。この通信規約を定めることにより、インターネットを利用する環境が異なっても、同じ手順でデータをやり取りすることが可能となる。

しかし、httpは情報を暗号化していない。そのため、クレジットカードの情報や個人情報などの情報のように、第三者に見られてはいけない情報を取り扱うときはhttps(HyperText Transfer Protocol Secure)を利用する。このプロトコルはhttpとは違い、情報内容を暗号化して通信をしている。

## 2.7 Apache

Apacheとは、Webサーバーソフトウェアの一つである。サーバーを構築するにはWebサーバーソフトウェアが必要である。Apacheの主な特徴は、オープンソースソフトウェア(OSS)で、無償で利用可能な点である。また、LinuxだけでなくWindowsなど複数のサーバーOS上で動作が出来る点である。

# 3. システム設計

## 3.1 システム概要

### 3.1.1 ログイン

GoodsReviewにて、商品を登録と商品のレビュー投稿にはログインが必要である。初めて、利用する方は新規登録が必要である。新規登録に必要な情報は「ユーザー名」と「パスワード」である。ユーザー名は、他の人と同一のものにならないような工夫がされている。新規登録画面が表示されると、登録ボタンが使用不可の状態になっている。このページではデータベースと常にやり取りがされており、ユーザー名を入力するとその値がすでに使用済みかどうかを判定するように設計されている。もし、まだ使用されていないユーザー名が入力されていれば、登録ボタンが使用可能となり、会員情報がデータベースに登録される。会員情報の登録が完了し、ログイン画面にユーザー名とパスワードを入力すると、セッションが発行され、ログインが完了する。

### 3.1.2 商品登録

商品登録は、他のユーザーに評価して欲しい商品を登録する画面である。登録に必要な情報は、「商品名」、「商品の画像」、「商品の概要」である。項目が全て入力され確認画面を押すと、確認画面に遷移する。その際に、投稿された画像はサイズを統一させるために画像処理をしている。登録ボタンを押すと「商品名」と「商品の概要」は入力された通りにデータベースに登録され、「商品の画像」は画像処理が完了した画像のパスが登録される。登録が完了すると、リストに登録された商品が表示される。

### 3.1.3 商品のレビュー投稿

商品のレビュー投稿は、商品に対しての評価する登録する画面である。登録に必要な情報は、「商品の評価」と「コメント」である。評価は1点から5点までの間で選択することが出来て、星のマークをクリックすることによって点数を変更する。コメントはtextareaに書き込み、改行やスペースが含まれていてもその情報をデータベースに書き込むことが出来る文字列操作がされている。レビューを投稿すると商品のレビュー画面で、自分と他のユーザーが投稿したレビューを閲覧することが出来る。

## 3.2 データベース設計

GoodsReviewでは、memberテーブル、goodsテーブル、

goodsReview テーブルの 3 つを使用している。

表 1 は、member テーブルの Field、Type、Key を表している。member テーブルは、ユーザーを管理するテーブルである。username が主キーとなり、ユーザーネームが他のユーザーと重複しない設計となっている。password には、sha256 でハッシュ化された文字列が格納される。それにより、テーブルの中を見られた場合でも、パスワードを推測するのが難しくなっている。

Field	Type	Key
username	varchar(25)	PRI
password	varchar(128)	

表 1 member テーブル

表 2 は、goods テーブルの Field、Type、Key を表している。goods テーブルは、ユーザーが登録した商品の情報を管理するテーブルである。id が主キーとなり、登録された商品を一意に識別することが出来る。category は登録された商品が属するカテゴリー、goodsName はユーザーが入力した商品名、overview はその商品の概要が登録される。ユーザーが概要を入力できる文字列は 400 文字が最大だが、スペースや改行が入力されると特殊文字に置き換えてテーブルに格納されるため、最大文字列を 4100 文字と設定されている。addUsername は、商品を登録した人のユーザー名が登録される。商品を登録するためにはログインが必要である。ログインが完了するとユーザー名をセッションで管理する。それにより、addUsername にユーザー名を登録することが可能となっている。evaluationTotal と commentNum はそれぞれ、ユーザーが付けた点数の平均と、ユーザーのレビューの件数が登録される。また、modified はレビューが更新された日付が登録される。evaluationTotal と commentNum は、レビューが投稿されるたびに書き変わる。そのため、Type が timestamp である modified が自動的に更新される。modified によって、商品一覧では更新された順番に並び替えることが可能となる。

Field	Type	Key
id	mediumint(8) unsigned	PRI
category	varchar(128)	
goodsName	varchar(400)	
imagePath	varchar(100)	
addUsername	varchar(25)	
evaluationTotal	double unsigned	
overview	varchar(4100)	
modified	timestamp	
commentNum	mediumint(8) unsigned	

表 2 goods テーブル

表 3 は、goodsReview テーブルの Field、Type、Key を表

している。goodsReview は、ユーザーが投稿したレビュー情報を管理するテーブルである。goodsReview テーブルには、主キーが設定されていない。goodsId は、goods テーブルに商品が登録される時に割り振られた id が登録される。これにより、そのコメントがどの商品に対してのレビューであるのかがわかるようになっている。username は、レビューをした人のユーザー名が登録される。goods テーブルに商品を登録する時と同様に、商品のレビューを登録するためにはログインが必要である。ログインが完了するとユーザー名をセッションで管理する。それにより、username にユーザー名を登録することが可能となっている。evaluation は、商品に対しての評価が登録される。evaluation は、1 から 5 の整数の値が登録されるので Type が tinyint となっている。comment は、商品に対してのコメントである。これも、goods テーブルに概要を登録するときと同様に、改行と空白が入力されたときに特殊文字に置換されるため、最大文字長を 4100 と設定されている。postedDate は、レビューが投稿された時間が登録される。Type が timestamp であるため、レビューが投稿されると自動的に登録される。

Field	Type	Key
goodsId	mediumint(8) unsigned	
username	varchar(25)	
evaluation	tinyint(3) unsigned	
comment	varchar(4100)	
postedDate	timestamp	

表 3 goodsReview テーブル

### 3.3 画面遷移

図 1 は、GoodsReview の画面遷移図である。初めに <http://localhost> にアクセスすると TOP ページが表示される。TOP ページからは、新規登録画面、ログイン画面、商品一覧画面に遷移することができる。新規登録画面では、ユーザー名とパスワードを入力して登録ボタンを押すと、登録完了画面に遷移する。すでに会員の方は、ログイン画面に移ることができる。ログイン画面では、ログインが完了すると TOP ページに移動する。商品一覧画面では、商品登録画面、グッズレビュー画面に遷移することができる。また、TOP ページに戻ることもできる。商品登録画面に遷移する際に、ログインが完了していない場合には、ログイン画面を経由する。商品登録画面で、商品名、画像、概要を入力して確認ボタンを押すと、商品情報確認画面に遷移する。入力した情報を変更したい場合には戻るボタンを押して、商品登録画面に戻ることができる。商品情報確認画面で登録ボタンを押すと、商品一覧画面に遷移する。グッズレビュー画面では、レビュー登録画面に遷移することができる。また、TOP ページと商品一覧画面に戻ることがで

きる。レビュー登録画面に遷移する際に、ログインが完了していない場合には、ログイン画面を経由する。レビュー登録画面で。評価とコメントを入力して確認ボタンを押すと、レビュー情報確認画面に遷移する。入力した情報を変更したい場合には戻るボタンを押して、レビュー登録画面に戻ることができる。。レビュー情報確認画面で登録ボタンを押すと、グッズレビュー画面に遷移する。

以上が、GoodsReview の画面遷移である。

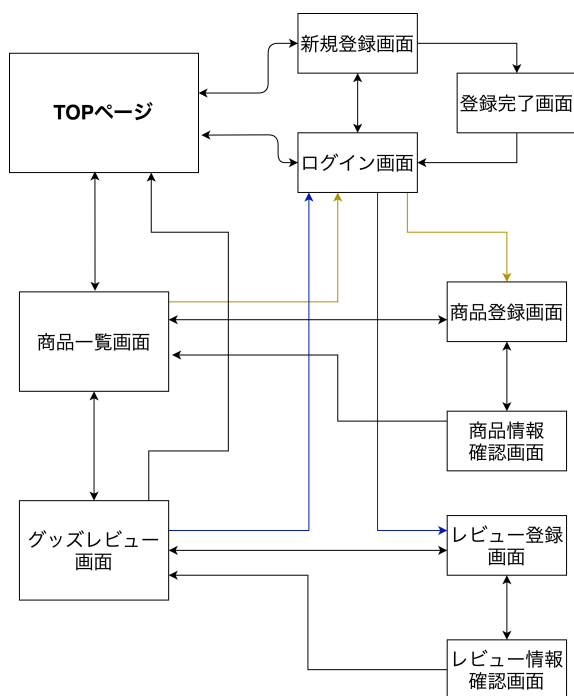


図 1 画面遷移図

### 3.4 システム詳細

GoodsReview のサーバールートである goodsReview フォルダには、「index.php」、「display」、「processing」が置かれている。**index.php** では、TOP ページを表している。日用品、ファッションなどのようにカテゴリーの画像が配置されており、この画像を a タグを囲むことで画像をクリックすると指定された URL に遷移するようになっている。

**display フォルダ**では、主にフロントエンドの処理をするファイルが置かれている。display フォルダの中は、ファイルが「stylesheet.css」、フォルダが「originalImages」、「changeSizeImages」、「defaultImg」、「goodsList」、「login」、「registerGoods」、「registerReview」、「review」となっている。**stylesheet.css** では、index.php を含めて、display フォルダに存在する全てのファイルに適用している。主な処理は、header の大きさの調整、バックグラウンドの画像を適用することである。

**originalImages フォルダ**と **changeSizeImages フォルダ**では、商品を登録する時に入力された画像を保存する

フォルダである。originalImage フォルダでは、入力された画像をそのまま保存され、changeSizeImage フォルダでは、imageResize.php で適切なサイズに変更された画像が保存される。

**defaultImg フォルダ**では、index.php に使用されているカテゴリーの画像、ロゴの画像、バックグラウンドの画像が保存されている。

**goodsList フォルダ**では、商品の一覧が表示する処理をする「goodsList.php」が置かれている。**goodsList.php** では、php で goods テーブルからカテゴリーが一致する商品の id、goodsName、imagePath、modified を取得する。そのデータを data 変数に格納する。もし、goodsName にスペースを表す特殊文字が入っていた場合、スペースに置き換える処理をする。そして、data 変数を json 形式にエンコードをする。JavaScript では、エンコードされたデータを用いて、htmlStr 変数に html のプログラムを表す文字列を追加する。そして、表品一覧を表示する id が goodsListArea の部分に innerHTML 関数を用いて htmlStr 変数を代入する処理を行う。

**login フォルダ**では、ログインや新規会員登録に関する処理をする「registerMember.php」、「completedRegistration.php」、「failRegistration.php」、「login.php」が置かれている。**registerMember.php** では、新規会員登録画面で行われる処理をする。goodsReview ではユーザー名が重複することが禁止されている。そのため、入力されたユーザー名が使用可能かどうかを判定する必要がある。そのため、JavaScript の setInterval 関数を用いて judgeExitMember 関数をコールするようにしている。judgeExitMember 関数では、Ajax を用いて入力されたユーザー名を judgeExitMember.php に受け渡し、すでに使われているユーザー名かどうかを判別する。そして、入力されたパスワードが確認用のパスワードと一致していれば、登録ボタンを使用可能にする。**completedRegistration.php** と **failRegistration.php** では、それぞれ、会員登録が成功した場合と、失敗した場合に表示される画面である。**login.php** では、ログイン画面で行われる処理をする。ログイン画面には、ユーザー名とパスワードの入力フォームがある。そのデータを POST 方式で authentication.php に受け渡す。

**registerGoods フォルダ**では、商品を登録する処理をする「registerInfo.php」、「checkInfo.php」が置かれている。**registerInfo.php** では、商品登録画面で行われる処理をする。商品登録画面では、商品名、画像、概要の入力フォームがある。そのデータを POST 方式で、checkInfo.php に受け渡す。**checkInfo.php** では、商品情報確認画面で行われる処理をする。初めに registerInfo.php から受け取った画像を適切なサイズに変更するために、imageResize.php に受け渡す。画像処理が完了したら、画像のファイル名を日付とランダムな数字とアルファベットの 10 文字列を用

いて変更し、重複を起こさないようにする。そして、受け取った商品名、概要とサイズが変更された画像を出力する。そして、登録ボタンを押すと、出力されたデータを POST 方式で、addGoods.php に受け渡す。

**registerReview フォルダ**では、商品のレビューを登録する処理をする「registerInfo.php」、「checkInfo.php」が置かれている。**registerInfo.php**では、レビュー登録画面で行われる処理をする。レビュー登録画面では、評価とコメントを入力する。評価は、星の画像をクリックすることで決定することができる。星の画像にはそれぞれidが割り振られており、JavaScriptによって画像がクリックされた時にchangeStar関数が呼び出されるようになっている。この関数は、クリックされた星の画像を元に、imgタグのsrcを書き換える処理をしている。また、evaluationのvalueを書き換える操作も行う。そして、そのデータを POST 方式で、checkInfo.php に受け渡す。**checkInfo.php**では、レビュー情報確認画面で行われている処理をする。registerInfo.phpから受け取ったevaluationを用いて、JavaScriptでcreateElementとsetAttribute関数を用いてタグを生成して、画面に評価を星の画像で表示している。また、データベースにコメントを登録できるように、スペースや改行は特殊文字に置き換えてからpostComment変数に文字列を代入している。そして、登録ボタンを押すと、出力されたデータを POST 方式で、addReview.php に受け渡す。

**review フォルダ**では、商品のレビュー画面を構成している「showReview.php」が置かれている。**showReview.php**では、goodsテーブルとgoodsReviewテーブルから商品の情報とそれに対するレビューの情報を取得する。showReview.phpに画面遷移する時には、createSession.phpを経由して商品を識別するidをもつセッションを発行する。それにより、goodsテーブルから適切な商品のgoodsName、imagePath、evaluationTotal、overviewを取得することができる。overviewに関しては、スペースや改行の特殊文字を元に戻す処理をする。また、evaluationTotalは、複数のユーザーの平均評価点なので、値が小数になることがある。画面に評価を星の画像を用いて出力する時には、整数部分と小数第一位の数字を用いる。それによって、使用する星の画像を変更している。これが、商品の情報を出力する処理である。goodsReviewテーブルから情報を取得するときは、goodsテーブルから取得した時と同様に商品を識別するidをもつセッションを用いて検索をする。それにより、username、evaluation、comment、postedDateを取得する。evaluationは、registerReviewフォルダの中のcheckInfo.phpと同様の処理をしており、JavaScriptでcreateElementとsetAttribute関数を用いてタグを生成して出力している。commentもスペースや改行の特殊文字を元に戻し、出力を行なっている。また、SQL文でORDER BY postedDate DESCを付け加えることで、レビューが新

しく投稿された内容が上に表示されるようになっている。

**processing フォルダ**では、バックエンドの処理をするファイルが置かれている。processingフォルダの中は、「addGoods.php」、「addReview.php」、「addUser.php」、「authentication.php」、「createSession.php」、「dbConnect.php」、「imageResize.php」、「judgeExitMember.php」、「sessionDestroy.php」が置かれている。**addGoods.php**は、registerGoodsフォルダの中に入っているcheckInfo.phpから受け取った情報をgoodsテーブルに格納する処理をする。**addReview.php**は、registerReviewフォルダの中に入っているcheckInfo.phpから受け取った情報をgoodsReviewテーブルに格納する処理を行う。また、goodsテーブルからevaluationTotalとcommentNumを取得する。そして、この情報とcheckInfo.phpから受け取った情報を元に、goodsテーブルのevaluationTotalとcommentNumを更新する。**addUser.php**は、registerMember.phpから受け取った情報をmemberテーブルに格納する処理をする。その際に、すでに使われているユーザー名や入力されたパスワードと確認用のパスワードが一致していない場合は、failRegistration.phpに遷移させて登録が失敗したことをユーザーに伝える。**authentication.php**は、ログイン画面で入力されたユーザー名とパスワードを用いて、ログイン認証を行う処理をする。初めに、ユーザー名でmemberテーブルを検索する。そして、見つかった情報からpasswordを取得する。login.phpから受けとったpasswordをsha256でハッシュ化した文字列と等しければ、セッションでusernameをセットする。そして、条件に応じた画面に遷移する。**createSession.php**では、セッションをセットする処理を行う。初めに、画面遷移元のファイル名を変数transitionFileに代入する。transitionFileがindex.phpの場合には、categoryとnameをセッションで管理するようにする。そして、goodsList.phpに画面遷移を行う。transitionFileがgoodsList.phpの場合、商品登録画面に遷移する場合と、グッズレビュー画面に遷移する場合がある。その判別方法として、GET方式でidが設定されているかをif文で判別する。設定してある場合は、セッションでidを管理が出来るようにしてshowReview.phpに画面遷移する。ログインを行っていない場合にはGET方式で変数backを設定し、login.phpに画面遷移させる。ログインが完了してある場合には、セッションでregisterGoods変数をtrueとしてregisterGoodsフォルダのregisterInfo.phpに画面遷移させる。transitionFileがshowReview.phpの場合、ログインが完了していない時はGET方式で変数backを設定し、login.phpに画面遷移する。ログインが完了している時はセッションで変数registerReviewをtrueに設定させ、registerフォルダのregisterInfo.phpに画面遷移させる。**dbConnect.php**では、データベースにア

クセスする処理をする。データベースにアクセスする時には PDO(PHP Data Object) クラスを使う。接続時にエラーを検知できるように、PDO::ATTR\_ERRMODE に PDO::ERRMODE\_EXCEPTION を設定し、SQL インジェクションの対策として PDO::ATTR\_EMULATE\_PREPARES を false で設定している。**imageResize.php** では、画像を適切なサイズに変更する処理をしている。画像のファイルパスを registerGoods フォルダの checkInfo.php から受け取る。png と jpeg 画像はそれぞれ imagecreatefrompng と imagecreatefromjpeg 関数で画像を表すイメージ ID を gding\_in 変数に格納する。その値を元に適切なサイズを求め、imagecreatetruecolor 関数で、指定の縦横のイメージ ID を作成し、gding\_out 変数に格納する。そして imagecopyresized 関数で、複製とサイズ変更を行い、imagepng 関数と imagejpeg 関数でリサイズしてファイルに出力する。最後に、imagedestroy 関数で利用したメモリを解放して、処理が完了する。**judgeExistMember.php** では、ユーザー名がすでに使用されているか判定する処理を行う。registerMember.php によって入力されたユーザー名を受け取り、SQL 文で検索をする。それにより、ヒットした件数を取得する。件数が 0 の場合は結果として 1 を、件数が 1 以上の場合は 0 を返す。**sessionDestroy.php** では、ログアウトをする処理を行う。ログアウトをすることが出来るのは TOP ページからのみである。そのため、遷移元のファイル名を調べ、index.php ならばそのユーザーのセッションを削除する。そして、再び TOP ページへ遷移させる。

## 4. 実装

### 4.1 実装環境

GoodsReview の実装に XAMPP を使用した。XAMPP とは、Web アプリケーションの実行環境を構築するソフトウェアであり、Apache、MariaDB、PHP、Perl を使用することが可能である。XAMPP のバージョンは 8.0.3 である。また、使用した OS(Operating System) は MacOS である。

### 4.2 環境設定

GoodsReview は、Apache という Web サーバーソフトウェアによって動作している。以下では、Apache の設定ファイルである httpd.conf の内容について述べる。

#### 4.2.1 Listen

Listen には、Apache が利用するポート番号を指定する。ポート番号とは、通信をするソフトウェア指定する番号である。ここではポート番号は 80 番に指定している。

#### 4.2.2 LoadModule

LoadModule とは、オブジェクトファイルやライブラ

リをリンクし、使用モジュールのリストに追加するものである。ここでは、「LoadModule authn\_file\_module modules/mod\_authn\_file.so」と指定している。

#### 4.2.3 ServerAdmin

ServerAdmin では、サーバー管理者のメールアドレスを指定することができる。このメールアドレスはエラーが発生した時に画面内に表示される。今回の GoodsReview はローカルで実装されるため「ServerAdmin you@example.com」というようにデフォルトの設定になっている。

#### 4.2.4 ServerName

ServerName では、サーバーの名前を設定する。これには複数の設定方法があり「サーバー名:ポート番号」、「サーバー名」、「IP アドレス」で指定する事が出来る。ここでは、「ServerName localhost」と設定している。

#### 4.2.5 DocumentRoot

DocumentRoot では、公開コンテンツを設置するディレクトリ名を設定する。ここに置いた index.html ファイルや index.php ファイルはブラウザで確認できる。画像やスタイルシート関連もこのディレクトリ以下に配置する。ここでは「DocumentRoot "/Applications/XAMPP/xamppfiles/htdocs"」と設定されている。

#### 4.2.6 AllowOverride

AllowOverride では、.htaccess ファイルの有効、無効を設定する。設定値が「All」の場合は有効であり、設定値が「None」の場合は無効である。ここでは「All」と設定されている。

#### 4.2.7 DirectoryIndex

DirectoryIndex では、ブラウザから Apache へディレクトリへのリクエストがあった場合にどのファイルを返すのか設定します。ここでは、「DirectoryIndex index.php index.html」と設定されている。

## 5. まとめ

第一章では、機能概要について述べた。新規会員登録を行いログインをするまでの手順や、商品のレビューを閲覧できるまでの流れについてまとめた。

第二章では、利用技術について述べた。GoodsReview で用いられているプログラミング言語や使用されているデータベースについて説明した。また、Web アプリケーションを動作させている Apache についても述べた。

第三章では、システム設計について述べた。テーブルで使われているカラムやデータの型や画面遷移を図表を用いて説明した。また、ファイルやディレクトリの使用用途や動作などについても述べた。

第四章では、実装について述べた。OS や XAMPP のバージョンなどの実装環境や、Web サーバーソフトウェアである Apache の設定ファイルである httpd.conf について説明した。