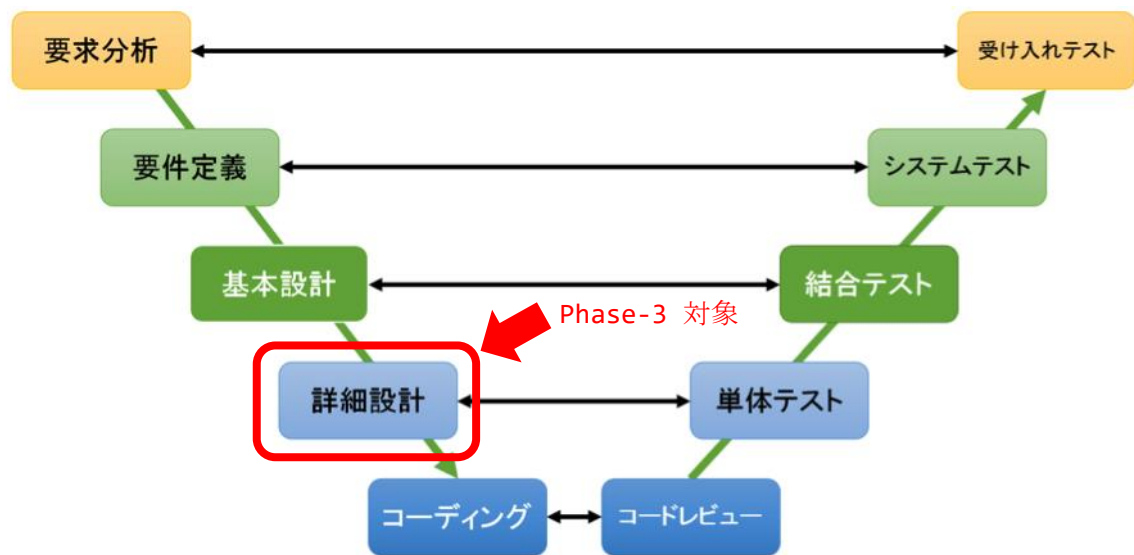


システム開発演習

Phase-3 補足説明資料

Phase-3 の概要	1
概要	1
【Phase-3 作成資料】	2
クラス/インタフェース設計	3
JSP/HTML 設計書	7

Phase-3 の概要



概要

この Phase では、いよいよ実際にコーディングをするためのクラス/インタフェース設計書を作成します。

ここまでで、Phase-2 で作成したアプリケーション構造でそれぞれのクラスやインタフェースの役割が明確になっていますので、フィールド、コンストラクタ、メソッドなどの設計をおこないます。

当然、Phase-2 で作成したコーディング/命名規約に則ったルールでクラス名、インタフェース名、フィールド名、メソッド名などを記述します。

クラス/インタフェース設計書が出来上がると、画面設計書に詳細な項目なども記述できるようになります。画面設計書を作成する段階で、未記入の項目をすべて埋めて完成させます。

特に設計書として作成するとは決まっていますが、作成しておくとな便利なものとして、リクエストオブジェクトやセッションオブジェクトに情報を登録するときのキーワードの命名ルール **name** 属性の値、画面を認識するために何らかの **ID** などを使う場合は、その **ID** の命名ルールなどを決めておくことでコーディングをするときに関係ミスが減ると思います。

コーディング/命名規約に追記したり、頻度が高いので別資料として作成したりすることも可能です。

【Phase-3 作成資料】

- ・ クラス/インタフェース設計
- ・ HTML 設計書

クラス/インタフェース設計

クラス定義書とは、一つ一つのクラスの仕様を日本語でまとめたドキュメントです。クラス定義書を作成することにより、クラスに必要な属性や操作および可視性などを、わかりやすくドキュメントとして残せます。

1 クラスの構造をドキュメント化し、各クラスの属性や操作などを明確にします。作成することによりクラスの属性や操作の詳細が、より明確になります。

クラス/インタフェース設計の内容は、クラス/インタフェースの詳細な情報の他に、クラス/インタフェース内にあるフィールド、コンストラクタ、メソッドの情報も記述します。各メソッドの内部仕様を決定し、実装担当者がクラス/インタフェース設計書内のメソッド仕様書をもとに実装できる程度に詳細化します。各メソッドで扱う属性や処理の内容が明確になります。

また、そのメソッドで実行する SQL 文や条件判断、例外処理を記述することで、どのような処理を行うメソッドなのかが、より明確になります。

しかし、コンストラクタや、メソッドの処理内容についての粒度に決まりはありません。現場によって、コードをそのまま書けるような細かい内容で記述するところもありますし、別の現場では必要な情報のみを記述して、作り手側に依存する場合があります。ですので、**処理内容の粒度は事前にチームで検討して統一**します。

とは言ってもいきなり統一するのは難しいかもしれませんので、チームの中で誰か最初に作成したものを雛形として参考にするという方法も一案だと思います。

JavaWeb システムの講義の修了課題で配布した設計書は、かなり詳細な書き方をしています。

クラス/インタフェース定義書				
●処理概要				
m_product テーブルへのCRUDの処理を行うDAO		作成者	杉原文恵	作成日
		承認者	久保雅彦	承認日
●クラス仕様				
クラス名	ProductDAO	パッケージ名	jp.co.kenkenshop.dao	
アクセス修飾子	public	修飾子	-	
継承元	-	実装元	-	
例外	-			
アノテーション				
●フィールド				
1	フィールド名	アクセス修飾子	修飾子	データ型
2	con	private	-	java.sql.Connection
3				初期化式/アノテーション
4				null
5				
6				
7				
8				
9				
10				
11				
12				
13				

項目名	説明
処理概要	
処理概要	クラス/インタフェースの概要説明
作成者	作成した人の名前(フルネーム)
作成日	作成した日
承認者	チーム内対面レビュー実施者
承認日	チーム内対面レビューで承認された日
クラス仕様	
クラス名	実際にコーディングするクラス名
パッケージ名	実際にコーディングするパッケージ
アクセス修飾子	アクセス修飾子(デフォルトの場合は -(ハイフン))
修飾子	アクセス修飾子以外の修飾子 (未設定の場合は-(ハイフン))
継承元	継承元のクラスを完全クラス名で記述 (未設定の場合は-(ハイフン))
実装元	実装元のインタフェースを完全インタフェース名で記述 (未設定の場合は-(ハイフン))
例外	-(ハイフン)を記述
アノテーション	アノテーションを記述
フィールド	
フィールド名	フィールド変数名
アクセス修飾子	アクセス修飾子(デフォルトの場合は-(ハイフン))
修飾子	アクセス修飾子以外の修飾子 (未設定の場合は-(ハイフン))
データ型	基本データ型以外のデータ型を記述する場合は、完全クラス名で記述
初期化式/アノテーション	フィールドの初期値がある場合は、その値を記述 アノテーションがある場合は、アノテーションを記述

クラス/インタフェース定義書				
●コンストラクタ				
コンストラクタ名	アクセス修飾子	修飾子		引数
ProductDAO	public	-		java.sql.Connection
アノテーション			例外	
-			-	
処理内容				
① 引数で受け取った Connection オブジェクトをフィールド変数に代入する				

項目名	説明
コンストラクタ名	実際にコーディングするコンストラクタ名
アクセス修飾子	アクセス修飾子(デフォルトの場合は -(ハイフン))
修飾子	アクセス修飾子以外の修飾子 (未設定の場合は-(ハイフン))
引数	引数に設定するデータ型を基本データ以外は完全クラス名で記述 (引数なしの場合は-(ハイフン)) 引数の変数名は記述しない 基本データ型などの場合は、引数がどんな意味を持つデータなのかを()を付けて日本語で説明する
アノテーション	アノテーションを記述
例外	呼び出し元に例外を報告する場合は、該当の例外クラスを完全クラス名で記述
処理内容	コンストラクタが呼び出されたときの処理を記述する

メソッド名	アクセス修飾子	修飾子	戻り値	引数
getAllProduct	public	-	ArrayList<Product>	String (カテゴリーID)
アノテーション			例外	
-			SQLException	
処理内容				
<ul style="list-style-type: none">・ 引数に受け取ったカテゴリーIDで下記の SQL 文を発行し取得結果を ArrayList<Product> オブジェクトに設定し、戻り値として返す・ SQL : SELECT p.product_id, p.product_name, p.product_kana, c.category_id, c.category_name, p.price, p.size, p.image_name, s.quantity, p.available_flag FROM m_category c LEFT OUTER JOIN m_products p ON p.category_id = c.category_id LEFT OUTER JOIN m_stock s ON p.product_id = s.product_id WHERE c.available_flag = '0'・ 引数に受け取ったカテゴリーIDが "" (空文字) ではない場合、下記を追記する AND p.category_id=?・ 最後に下記を追記する ORDER BY c.category_id, p.product_id・ 引数に受け取ったカテゴリーIDが "" (空文字) ではない場合 IN パラメータの 1 番目 : 引数で受け取ったカテゴリーID・ Product オブジェクト, Category オブジェクト, Stock オブジェクトを生成し、対応するプロパティに値をセットする・ 例外処理は、SQLException を受け取り、printStackTrace() メソッドを使用しトレース情報をコンソールに出力し、throw キーワードを使用し例外を呼出しメソッドに報告する・ finally 処理 ResultSet オブジェクト、PreparedStatement オブジェクトを close() する				

項目名	説明
メソッド名	実際にコーディングするメソッド名
アクセス修飾子	アクセス修飾子(デフォルトの場合は -(ハイフン))
修飾子	アクセス修飾子以外の修飾子 (未設定の場合は-(ハイフン))
戻り値	戻り値のデータ型を記述。クラス/インタフェースの場合は完全クラス名で記述
引数	引数に設定するデータ型を基本データ以外は完全クラス名で記述 (引数なしの場合は-(ハイフン)) 引数の変数名は記述しない 基本データ型などの場合は、引数がどんな意味を持つデータなのかを()を付けて日本語で説明する
アノテーション	アノテーションを記述
例外	呼び出し元に例外を報告する場合は、該当の例外クラスを完全クラス名で記述
処理内容	メソッドが呼び出されたときの処理を記述する 例外処理がある場合は、例外処理の記述、finally 句の記述が漏れないようにする

JSP/HTML 設計書

Phase-2 で作成した、画面設計書にコーディングで必要な情報を追記していきます。

Phase-3 で画面設計書を完成させます。

UI	画面設計書	システム名	ショッピングシステム	版数	作成者	作成日	承認者	承認日
		画面ID	U_002	初版	鈴木	2017/3/10	久保	2017/3/14
		画面名	ログイン	2.0版				
		JSP名	login.jsp	版				

■トップ画面

■処理概要

- ログインを行う
- ログイン後は、トップ画面に戻る

■項目

- ① ユーザーID（メールアドレス）、パスワード入力 欄
- ② エラーメッセージ欄（文字色は赤とする）

ユーザーID	必須チェック：ユーザーIDを省略することはできません。
パスワード	必須チェック：パスワードを省略することはできません。
ユーザーID、パスワード	入力チェック：メールアドレス、もしくはパスワードが正しくありません。

項目名	説明
HTML 名	HTML 作成時のファイル名(Phase-3 で記入)

No.	項目名(和名)	name属性	I/O	データ型	桁	必	位置	タグ	備考
1	ユーザーID	mail	I	String		●		<input type="text">	
2	パスワード	password	I	String		●		<input type="password">	
3	[ログイン]ボタン							<input type="submit">	
4	ページID	pId		String				<input type="hidden">	value="login"
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									

■機能補足

項目名	説明
項目名(和名)	入出力項目名
name 属性	タグを使用する際の name 属性値 (name 属性値を設定する必要がない場合は未記入)
I/O	I : 入力項目 0 : 出力項目
データ型	入出力時のデータ型。データ型を記述(基本データ型、クラス名のみ)
桁	入力時に桁数制限がある場合は桁数を記述 (表示サイズではないので注意)
必	入力項目の場合、必須入力の場合に●を記述
位置	出力項目の場合、左：左詰め、中：中央揃え、右：右詰めに記述
タグ	HTML のタグを記述
備考	設定値などが必要な場合は記述
機能補足	コーディングする際に注意事項などがある場合は、箇条書きで記述

- ※ 固定文字は画面設計書の画面で判断できるので記述しない
- ※ HTML 設計時の作成者や日付等に関しては 1.0 版として記述