

Aチーム ECサイト制作

総リーダー/成果物リーダー: 谷森 瑞季
技術リーダー : 大野 真由美
デイリースクラムリーダー : 鈴木 薫平
進捗管理リーダー : 藤井 勇気

目 次

1. システム概要
2. 追加機能紹介
3. システム実演
4. 個人の成果報告
5. まとめ
6. 質疑応答

1. システムの概要

コンセプト

ユーザが使いやすいUI

エラーメッセージ・注文後のメール文等、細かい部分でのユーザへの気配り

チーム内共通意識

SQL発行回数を少なくしパフォーマンスを向上

きれいなコードを書くように心掛ける

会社でする作業と家でする作業を意識的に分ける

基本機能一覧

- ユーザ登録をする
- ログイン/ログアウトをする
- 商品一覧表示/商品検索をする
- 商品詳細を表示する
- ショッピングカートに商品を追加/削除する
- ショッピングカートの中身を表示する
- 注文確認画面を表示する
- 注文をする
- 注文履歴を表示する

2. 追加機能紹介

追加機能一覧

ユースケース	追加機能概要
ユーザ登録	<ul style="list-style-type: none">・パスワードのハッシュ化・郵便番号空住所を自動で取得・郵便番号にハイフンがあればエラーメッセージ表示
ログイン/ログアウト	<ul style="list-style-type: none">・ログイン前にログイン後のページにアクセスしようとするログイン画面へ遷移 (SpringSecurity)
商品一覧表示	<ul style="list-style-type: none">・ページング機能・並び順を選択して変更・ロコミ機能*
商品検索	<ul style="list-style-type: none">・商品名のオートコンプリート機能
商品詳細を表示する	<ul style="list-style-type: none">・画面のラジオボタン/チェックボックス/ドロップダウンを変更したら合計金額を自動で変更
カートの中身を表示	<ul style="list-style-type: none">・ログイン前に注文した商品をログイン後にもカート上に反映
注文をする	<ul style="list-style-type: none">・注文時、配達時間に過去日時を選択するとエラーメッセージを表示・注文後、自動メールを送信する・注文履歴を表示する

* オリジナルの追加機能

3. システム実演

～ デモ画面へ移行します。そのままお待ちください。 ～

4. 個人の成果報告

個人成果報告内容

- 担当/追加した機能
- 工夫した点
- 苦労した点

【総リーダー/成果物リーダー: 谷森瑞季】

①担当/追加した機能

- 商品一覧表示
- 商品検索
- ページング機能(商品一覧表示時のみ)

②工夫した点

- ユーザが分かりやすい画面表示
 - 検索結果がない時のメッセージの表示位置
 - ページングの際、分かりやすいボタンを採用

③苦労した点

- 長くなってしまったコードを for 文などを使用して簡略化して記述すること
(チームメンバー協力のもと達成)

【デイリースクラムリーダー: 鈴木薫平】

①担当/追加した機能

- ユーザ登録(郵便番号から住所自動取得)
- 注文確認画面表示 等

②工夫した点

- ユーザにとって利便性を高める
 - 郵便番号から住所自動取得 : zipcoda(挙動が不安定) → **AjaxZip3**を使用
- Githubでcommitする時のコメントを丁寧に書くこと
- 変数名を分かりやすく書くこと

③苦労した点

- 配達時間のデータ型変換 (String, LocalDateTime, TimeStamp)
- ordersテーブルのデータの更新 : ショッピングカート担当者との連携

【技術リーダー:大野真由美】

①担当/追加した機能

- オリジナル: 口コミ機能(投稿 / 表示 / 評価順での商品一覧ソート)
- 注文履歴表示(日付での絞り込み可)
- 商品詳細表示(金額リアルタイム計算)
- クレジットカード決済 WebAPI導入

等

②工夫した点

- 直感的に操作しやすいサイトにする
 - 注文履歴絞り込み時、入力日付が歯抜けでも自動補完
 - 注文履歴 / 商品の詳細画面から各一覧画面に戻った際、元のページ番号や検索条件を反映
- 綺麗なコードを目指す(if文ネストしない・機能のメソッド / クラス化 等)

③苦労した点

- オリジナル機能追加では、他メンバーの作成機能に影響しないよう、慎重に開発した
 - 既存クラス/メソッドの繋がりを把握し、再利用するもの / 新規作成するものを判断
※チームメンバーの javadoc・コメントがとても役立った
- 注文情報取得時の SQL(5tablesから情報取得)→ResultSetExtractorの記述が複雑であった

【進捗管理リーダー: 藤井勇気】

①担当/追加した機能

- ショッピングカート機能(追加 / 表示 / 削除)
- リファラ情報を取得し、ログイン後任意のページに遷移する機能
- 注文確定後のメール送信機能 等

②工夫した点

- 最初から完璧を目指さない → とにかく動くものを作る
- 自分が作成したメソッドを他のメンバーが使用する時に理解しやすいように工夫
→ 何のための処理かという意図が伝わるようにコメントを残す
- 同じ記述を2回以上書かない(同様の処理はメソッド化し、都度呼び出す)

③苦労した点

- ショッピングカートの追加機能
→ カートの中身の有無によって処理を分けなければいけない
- spring security導入後、ログイン前のカートの内容をログイン後に反映する機能の修正
→ 非ログイン時のuserId = セッションIDを数値化したものを使用。
spring securityはログイン前後でセッション IDを自動で変更してしまう仕様になっている。
その設定をオフにすることで対応。

5. ま と め

ま と め

改善点	<ul style="list-style-type: none">● 結果的に動作を優先し、あまり綺麗なコードを書けなかった● 機能を実装するとき、実装時に掛かる見積もり時間を把握せず、見切り発車が多かった● 他のチームメンバーが記述したコードについて理解していない部分もある
良かった点	<ul style="list-style-type: none">● チーム内での発言や技術共有を活発に行うことができた● 他のチームと情報共有(技術共有)をすることが出来た● Gitでmergeする際、競合することが少なく、スムーズな進行ができた

6. 質疑応答

ご清聴ありがとうございました。