

事前公表

第60回 技能五輪全国大会

ウェブデザイン

- Web Technologies -

競技課題 M3

ウェブプログラミング

(アプリ、サーバサイドおよびバックエンド開発)

作業時間： 3時間

目次

目次	2
はじめに	3
プロジェクトと課題の説明	3
選手への指示	9

はじめに

このモジュールでは、PHP（フレームワーク）とデータベースを使用してバックエンド部分（管理画面とRESTful API）を構築します。

セキュリティへの配慮を行いつつ、各機能が問題なく動作するように構築してください。

プロジェクトと課題の説明

巣ごもり需要の高まりを受け、新規でデリバリー事業を始める企業があります。

下記の要件に合うよう、飲食店が利用する管理画面と、RESTful APIを構築してください。

また、各テーブル間の関係性がわかるよう、簡易的なER図を作成してください。

◎課題全般について

- データベース（userXX_m3）を使用して、用意されているスキーマを利用できる。
 - ※ XX はゼッケン番号（例：ゼッケン番号「1」の場合、user01_m3）
- 飲食店マスタには、下記に指定されている通りにデータを直接挿入すること。
 - 飲食店マスタ用に users テーブルを用意してあるが、利用しなくても良い。
- 必要なテーブル、及び必要項目は次の通り。「必要項目」とは最低限必要な項目を指し示すため、これ以外の項目が必要と判断する場合は適宜追加すること。
 - 飲食店の情報を格納する「飲食店マスタ」
 - 必要項目：店舗名、住所、電子メールアドレス、アカウント、パスワード
 - 各店舗の商品情報を格納する「商品マスタ」
 - 必要項目：店舗ID、商品名、価格
 - 複数の商品を1つの「セットメニュー」として格納する「セットメニューデータ」
 - 必要項目：店舗ID、セットメニュー名称
 - セットメニューに複数の商品に関連付ける「セットメニュー・商品リレーション」
 - 必要項目：セットメニューID、商品ID
 - お客様の注文を格納する「注文データ」
 - 必要項目：商品ID、セットメニューID、配送先住所

◎管理画面について

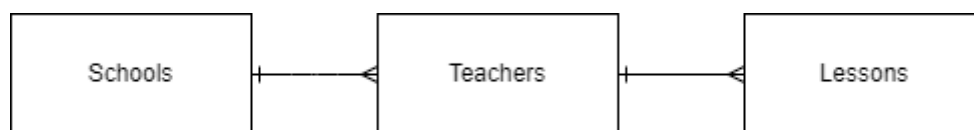
- 管理インターフェースを作成してください。
- 商品情報を新規登録できるインターフェースを構築してください。
- 商品情報リストにて、一覧表示ができるようにしてください。
- 商品情報は編集・更新・削除ができるようにしてください。
- 上限3点までの商品を1つのセットメニューとして新規登録ができるインターフェースを構築してください。
- 管理画面のURIは「/admin」とします。
- 仕様に関する追加情報は、下記に別途記載されています。

◎RESTful APIについて

- 商品情報をウェブサイトで表示するためのAPIを構築してください。
- 商品またはセットメニューをウェブサイトから注文するためのAPIを構築してください。
- APIのURIは「/api」とします。
- 仕様に関する追加情報は、下記に別途記載されています。

◎ER図について

- 各テーブルの関係性がわかるように関係性を記述してください。
アトリビューションを記載する必要はありません。
- ファイル形式はPDFで、名称を「userXX-ER.pdf」として提出してください。
※ XX はゼッケン番号（例：ゼッケン番号「1」の場合、user01-ER.pdf）
- 提出は「ER」ディレクトリを作成してアップロードしてください。
- カーディナリティは最低限「1対1」「1対多」を記載してください。
 - 参考例：
 - 1つの学校は複数の教員を持つ
 - 1人の教員は複数の授業を持つ



【管理画面の仕様に関する追加情報】

- **特徴:A1**
 - 飲食店ユーザとしてログインとログアウト
 - 飲食店ユーザとして機能にアクセスするため、管理画面にログイン／ログアウトできること。
 - 飲食店ユーザは以下の通りとする。
 - ID: admin
 - PASS: gorin
- シナリオ:**A1a**
 - 飲食店ユーザが正常にログイン
 - 「ログイン」ページにて、指定されたユーザ名とパスワードを入力後、「ログイン」ボタンを押すことで管理画面に遷移できること。
 - 管理画面に遷移後、「商品情報リスト」が表示されること。
 - 管理画面に遷移後、「ログアウト」ボタンが表示されること。
- シナリオ:**A1b**
 - 飲食店ユーザログイン失敗
 - 「ログイン」ページにて、間違ったアカウントとパスワードを入力後、「ログイン」ボタンを押した際、「アカウントまたはパスワードが正しくありません」というエラーメッセージが表示されること。
- シナリオ:**A1c**
 - 飲食店ユーザログアウト
 - ログイン状態で「ログアウト」ボタンを押すと、「ログイン」ページに転送されること。
 - ブラウザバックで履歴を戻っても、管理画面は表示されないこと。

- **特徴:A2**
 - 商品管理
 - 飲食店ユーザが、商品を管理できること。
- シナリオ:A2a
 - 商品情報の新規登録
 - ログイン状態で「商品新規登録」ボタンを押下することで、入力画面に遷移すること。
 - 必要な項目を入力後、「登録」ボタンを押下することで、商品情報がデータベースに保存されること。
 - 正常に登録された場合、「商品情報が登録されました」というメッセージが表示されること。
- シナリオ:A2b
 - 既存の商品情報リスト表示
 - 商品情報リストに、登録した商品が表示されていること。
- シナリオ:A2c
 - 既存の商品情報を編集する
 - 商品情報リストから「編集」ボタンを押下することで、編集画面に遷移すること。
 - 編集画面には、リストで選択した商品情報が予め入力されていること。
 - 情報を編集して「保存」ボタンを押下することで、データベース上の商品情報が更新されること。
 - 正常に更新された場合、「商品情報が更新されました」というメッセージが表示されること。
- シナリオ:A2d
 - 既存の商品情報を削除する
 - 商品情報リストから「削除」ボタンを押下することで、「削除してよろしいですか？」というダイアログが表示されること。
 - 「OK」を押下した場合、データベース上の商品情報が削除されること。
 - 「キャンセル」を押下した場合、商品情報リストに留まること。

- **特徴:A3**
 - セットメニュー管理
 - 飲食店ユーザが、セットメニューを管理できること。
- シナリオ:A3a
 - セットメニューの新規登録
 - ログイン状態で「セットメニュー新規登録」ボタンを押下することで、入力画面に遷移すること。
 - 入力画面では、商品を増減できるインターフェースがあること。
 - 必要な項目を入力、及び商品を選択後、「登録」ボタンを押下することで、セットメニューがデータベースに保存されること。
 - 正常に登録された場合、「セットメニューが登録されました」というメッセージが表示されること。
- シナリオ:A3b
 - 既存のセットメニューを編集する
 - セットメニューリストから「編集」ボタンを押下することで、編集画面に遷移すること。
 - 編集画面には、既に登録されたセットメニューが表示されており、商品を増減できるインターフェースがあること。
 - 情報を編集して「保存」ボタンを押下することで、データベース上のセットメニューが更新されること。
 - 正常に更新された場合、「セットメニューが更新されました」というメッセージが表示されること。
- シナリオ:A3c
 - 既存のセットメニューリスト表示
 - セットメニューリストに、登録したセットメニューが表示されていること。
- シナリオ:A3d
 - 既存のセットメニュー情報の削除
 - セットメニューリストから「削除」ボタンを押下することで、「削除してよろしいですか?」というダイアログが表示されること。
 - 「OK」を押下した場合、データベース上のセットメニューが削除されること。
 - 「キャンセル」を押下した場合、セットメニューリストに留まること。

【RESTful APIの仕様に関する追加情報】

- **特徴:B1**
 - 商品情報の読み取り（エンドポイント：`/api/items`）
 - アプリケーションから商品情報のJSONデータ読み取りができること。
- **特徴:B2**
 - 商品情報の絞り込み（エンドポイント：`/api/items?x=y`）
 - APIの引数にshop_id（数値）を加えることで、店舗での絞り込みができること。
 - APIの引数にprice（数値）を加えることで価格での絞り込みができること。
 - APIの引数にtitle（文字列）を加えることで、商品名での絞り込み（LIKE部分検索）ができること。
 - 絞り込みについて、複数の引数が増えられている場合はAND検索での絞り込みとすること。
- **特徴:B3**
 - セットメニューの読み取り（エンドポイント：`/api/sets`）
 - アプリケーションからセットメニューのJSONデータ読み取りができること。
- **特徴:B4**
 - セットメニューの絞り込み（エンドポイント：`/api/sets?x=y`）
 - APIの引数にshop_id（数値）を加えることで、店舗での絞り込みができること。
- **特徴:B5**
 - 商品、またはセットメニューの注文（エンドポイント：`/api/order`）
 - アプリケーションからの注文データを、POSTで受信できること。
 - 受け取るパラメータは以下。
 - 商品ID（item_id）、またはセットメニューID（sets_id）。
 - どちらかは必須とする。
 - 空の場合はnullを入れることとする。
 - 配送先住所（address）の文字列。
- **特徴:B6**
 - HTTPステータスコードの返却
 - データ取得成功時には、HTTPステータスコード200を返すこと。
 - データ更新成功時には、HTTPステータスコード201を返すこと。
 - エラー発生時には、HTTPステータスコード404を返すこと。

選手への指示

1. 提供されている material フォルダ内のファイルを使うことができます。
また必要に応じて、提供されているフレームワークを使用できます。
2. 「 m3 」という名前のサーバ上のディレクトリに作業中のアプリを保存してください。
メインファイルがindex.html または index.php という名前であることを確認してください。
3. フレームワークは Laravel、CodeIgniter、Yiiが提供されています。
SFTPでアップする際はパーミッションの設定を行う必要があります。
 - a. Laravelの場合
storage配下をすべて777に設定 (例 : `sudo chmod -R 777 storage/`)
 - b. CodeIgniterの場合
writable配下をすべて777に設定 (例 : `sudo chmod -R 777 writable/`)
 - c. Yiiの場合
web/assets配下をすべて777に設定 (例 : `sudo chmod -R 777 web/assets/`)
4. 時間管理は自身で行ってください。
5. 注意事項 : 競技サーバにアップロードされたデータのみが採点対象となります。