

第 21 回 若年者ものづくり競技大会
「ソフトウェアアプリケーション開発」職種

競技課題

作成者：ソフトウェアアプリケーション開発競技委員会

作成日：2026 年 7 月 24 日

■ 競技課題のタイトル

富山湾直送！回転寿司チェーン仕入業務支援デスクトップアプリケーション

■ 競技課題の概要

富山県内の地場回転寿司チェーンを舞台に、本部と通信しながら「現場のリアルタイム業務」を支えるデスクトップアプリケーションを開発する。

富山県内の人気回転寿司チェーン「富山湾直送！とやま白えび丸」では、毎朝市場から届く食材の鮮度管理が課題となっている。大手チェーンのような全自動システムではなく、現場のアルバイトも含めた従業員が、「その日の水揚げ状況に合わせて仕入れ業務を柔軟に操作できるツール」が必要とされている。仕入れた食材を管理するのは※在庫管理サーバーである。

また、オンラインによりテイクアウト注文を受け付ける※本部サーバーと API 通信を行い、テイクアウト注文の受信と連携した表示・管理を行う機能がある。

競技者に求められるのは、厨房の隅にある端末で動作する、高機能かつ堅牢な仕入業務支援のためのアプリケーションの構築と基本的な API 通信の機能実装である。

※在庫管理サーバーは会場の LAN 上にある SQL サーバー、本部サーバーは競技端末内で Mockoon により仮想的に構築されている。

■ 課題対象とするアプリケーション

「富山湾直送！とやま白えび丸」仕入業務支援デスクトップアプリケーション

■ 必要とされる機能

以下は必要とされると思われる機能のうち一部（代表的なもの）である。このうちの一部の実装あるいはこれ以外の実装を行っても構わない。

・ログイン機能：

従業員 ID とパスワードでログインする簡易認証機能

・在庫表示機能：

在庫管理サーバーに接続し、在庫食材のリストを表示する画面。商品名、価格、在庫数などが一覧で確認できる機能。

・在庫管理機能：

新たに市場から届いたネタ（ブリ、シロエビ、バイ貝等）をリストから選び、商品名、価格、数量を入力・保存する機能。

・**廃棄登録機能：**

現場で発生した廃棄（食材の落下、鮮度切れ）を理由とともに登録する機能。（不正な値をチェックする機能も必要）

・**Mockoon（テイクアウト注文サーバー）との API 連携機能**

※ テイクアウト注文サーバー（Mockoon）のベース URL は `http://localhost:3001/api/v1` とする（ポート番号や実装のヒントは競技資料も参照すること）

※ 以下は Mockoon との連携機能の例である。

※ テイクアウト注文サーバー（Mockoon）が提供するエンドポイントは以下の 4 種類である。環境定義ファイルを編集して、エンドポイントやレスポンスの内容を変更してはならない。

※「**Mockoon = 本部サーバー役（起動済み・環境ファイルは編集禁止）**」／**Postman = 動作確認用の API クライアント。アプリの実装には必須ではない。**

（１）予約注文の自動受信 (Polling/Fetch)：

定期的に、または「更新ボタン」押下時に、本部のテイクアウト注文サーバー（Mockoon）から新規のテイクアウト注文一覧（顧客名、メニュー、受取時間等）を取得し、未対応リストに表示する。

API: `GET /api/v1/orders/pending`

（適切な受注メッセージ、エラーメッセージ、注文ステータスの表示を含む）

（２）注文ステータス更新 (Update)：

受注確定・調理完了・キャンセルなど、注文のステータスをテイクアウト注文サーバーに送信して更新する。

API: `PUT /api/v1/orders/:id`

（リクエストボディに status フィールドを含むこと。受付可能な値：confirmed / completed / cancelled）

※ テイクアウト注文サーバーは仮想的に実現しているので、ステータス更新を保持しない。更新の送信自体は成功するが、再度注文一覧を取得すると元の状態で返る。そのため、ステータスの保持・管理はアプリケーション側で行うこと。

（詳細は競技資料も参照）

（３）本日のおすすめ情報取得 (Fetch)：

テイクアウト注文サーバー（Mockoon）から「今日の全国相場・イチオシ魚種」情報を取得し、システム内に表示する。

API: GET /api/v1/specials

（４）オンライン販売サイトへの在庫状況の反映（Update）：

店舗で特定のネタが品切れになった（または在庫数が変化した）際、その商品の販売可否と在庫数をテイクアウト注文サーバー経由で本部のオンライン販売サイトに送信し、反映させる。これにより、店舗で売り切れたネタが本部サイト上で注文されてしまう事態を防ぐ。

API: PUT /api/v1/items/:id

- ・パスの :id には商品コード（例: ITEM-001）を指定。
- ・リクエストボディに available（販売可否 : boolean）と stock_count（在庫数 : integer）を含める。
- ・反映を自動（在庫が 0 になった時点で自動送信）で行うか、手動（従業員の操作で送信）で行うかは競技者の設計判断とする。

（テイクアウト注文サーバー、オンライン販売サイトの仕様や構成を設計・実装することは本課題の範囲外（スコープ外）とする）

■ 競技資料について

以下の参考資料がデスクトップの「競技資料」フォルダに格納されている。必要に応じて参照すること。

・**ヒアリング資料.docx** 店舗責任者へのヒアリング記録。要件定義・設計ドキュメントの作成に活用すること。

・**API 仕様書_サンプル.docx** API 仕様書の記入様式。「API 仕様書」を作成する際に使用すること。**API 実装上の留意点も記載されているので必ず目を通すこと**

・**データサンプル_SQL.docx** 在庫管理サーバー（SQL Server）への初期データ投入用 SQL スクリプトのサンプル。DB 構築の参考にすること。

・**DB 定義書_サンプル.docx** テーブル構造・カラム定義・FK 関係のサンプル。DB 構築の参考にすること。

（予備の競技資料）

※以下は Mockoon が停止した場合にのみ利用する

・**toyama_shiroebi_maru.json** テイクアウト注文サーバー（Mockoon）の環境定義ファイル。
競技時間中にサーバーが停止した場合に再起動する場合にのみ使用する。手順は「Mockoon 起動手順書」を参照すること。

・**Mockoon 起動手順書.docx** テイクアウト注文サーバー（Mockoon）が競技時間中に停止した場合の再起動・動作確認手順書。

※ その他、当日配布される資料がある場合は別途説明する。

課題 1（要件定義と設計）

ヒアリング資料をもとに、地域特産飲食店チェーン仕入業務支援デスクトップアプリケーションを構築するための、要件定義と設計を行い、以下を提出しなさい。なお必須提出物は必ず提出すること。

＜必須提出物＞

- ・ 要件定義書
- ・ DFD（データフロー図）
- ・ ユースケース図
- ・ 画面遷移図

＜その他の提出物＞

基本設計段階で有益と思われるドキュメントがあれば可能な範囲で作成して提出すること（例：画面デザインモックアップ、ER 図、データベーススキーマ定義書、受注処理・在庫管理のデータフロー設計書、テイクアウト注文サーバーとのデータ連携仕様書、簡易集計機能設計書、システム設計書、セキュリティ設計書など）。

※ 在庫管理サーバー（SQL Server）、テイクアウト注文サーバー（Mockoon）についてはコーディング・実装する必要はないが、すでに存在するという前提で資料を作成すること。

※ テイクアウト注文サーバーは注文の状態を保持しない（同じ注文が繰り返し取得される）。受信した注文情報をどこに・どのように保持するかは、要件（受取時間順の表示、キャンセル履歴の保持等）を踏まえて設計すること。

課題 2（アプリケーションの実装）

課題 1 で作成したドキュメントに基づいて「富山湾直送！とやま白えび丸」仕入業務支援デスクトップアプリケーションを作成しなさい。

＜必須提出物＞

- ・ ソリューションフォルダ（ソースファイル一式）
- ・ アプリケーション実行ファイル（EXE、DLL 等）
- ・ データベース作成スクリプト（自スキーマ内にテーブル、インデックス、制約などを作成するための SQL スクリプト）、またはダンプファイル
- ・ API 仕様書（サンプルを参照）
- ・ アプリケーション仕様書
- ・ アプリケーション操作手順書（ログイン ID や PASS、起動の仕方を必ず記載する）

＜その他の提出物＞

本アプリケーションの特徴を理解するために必要と考えられるドキュメント（設計意図、工夫点、技術的な挑戦点など）

※ いずれも自由フォーマットで可だがサンプルを参考にしてもよい。

※ 在庫管理サーバー、テイクアウト注文サーバーについてはコーディング・実装する必要はない。

課題 3: プレゼンテーションとデモンストレーション

課題 1 と 2 の成果をもとに、「富山湾直送！とやま白えび丸」仕入業務支援デスクトップアプリケーションによるプレゼンテーション資料を作成し、約 5-7 分程度（会場で指定する）で、**システムの採用を検討している顧客に向けた効果的なプレゼンテーションとデモンストレーション**を行いなさい。

プレゼンテーションにあたっては、競合プレゼンテーションであると仮定して資料を準備し、顧客に対してあなたの設計・作成したアプリケーションが採用されるために、日々の業務課題をどのように解決し運営に貢献できるか、また他社製品に比べて優れていると思われるアプリケーションの利点や特徴などを、強調すること。

※ あなたの同僚や上司、あるいは競技委員に対する報告会ではなく、あくまで導入を検討する顧客へのプレゼンテーションであることに注意してください。

<必須提出物>

- ・ プレゼンテーション資料（作成したアプリケーションによる簡潔なデモ画面などを含む）

以上