COLLEGE OF ENGINEERING

Electrical Engineering and Computer Science

MENUFY: A NEW WAY TO ORDER FOOD ONLINE

A business-friendly web application to digitalize the ordering process and inventory management for restaurants

Serena Tay | Matthew Egan | Alexander Drath

BACKGROUND

- The food ordering business has skyrocketed over the last couple years. The current projections suggest that number of users that use a platform-to-consumer delivery segment will amount to 69.9 million users in the US alone by 2024.
- With the recent COVID-19 pandemic, more and more users are choosing to have their food delivered to them instead of going out and risking illnesses.
- With a revenue of $26.5 billion dollars a year for the Online Food Delivery market in the US, building an application that can capture even a small portion of it will net a large profit for investors.
- The online ordering business also benefits the restaurants as according to a recent study, about 60% of operators surveyed by Technomic say that offering deliver has generated incremental sales.
- According to consumers, 1 in 4 say they spend more on off-premise orders than dining in at a restaurant.
- The application in development allows the consumer to order off-site and either carry out or dine-in with options to expand to delivery services with future updates.

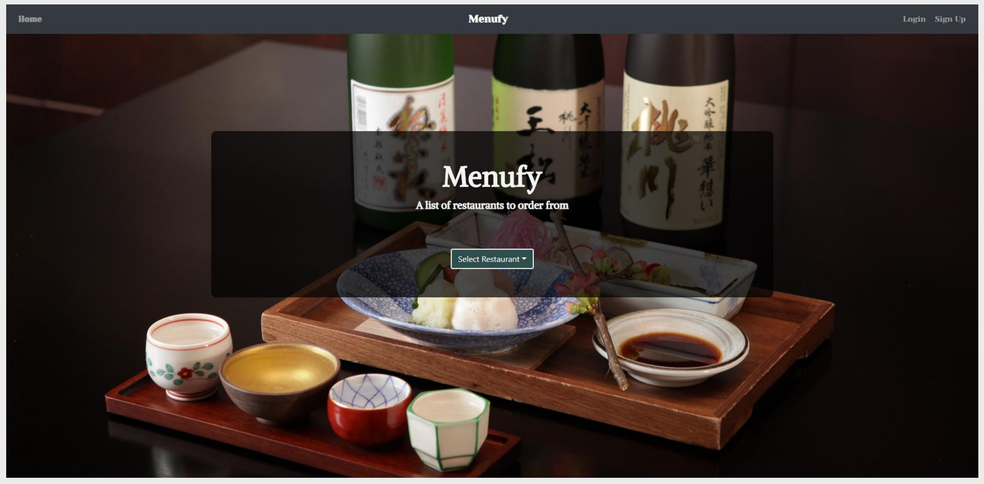
Sources:

https://upserve.com/restaurant-insider/online-ordering-statistics/#:~:text=60%25%20of%20restaurant%20operators%20say,phone%20orders.

https://www.restaurantbusinessonline.com/consumer-trends/stats-are-consumers-are-upping-restaurant-delivery

https://www.statista.com/outlook/374/109/online-food-delivery/united-states#market-marketDriver

Oregon State University



DESCRIPTION:

A web application that provides an online user platform that integrates food ordering and serving process as well as the inventory system for restaurant businesses. The application also enables data storage on a consolidated cloud-based system to enable restaurant management to utilize the data to help serve customers better and improve resource planning. This application aims to achieve the following objectives:

- **Improve customer outreach and accessibility -** easy-to-access online menu available on web app
- **Increase efficiency in the food ordering process -** provide customers with the ability to order food and beverages via the web application
- **Achieve sustainability -** eliminate paper-based menus
- **Improve inventory planning -** provide ready-to-use and consolidated information on customer preferences to better plan for future demands
- **Improve resource planning -** utilize information to identify bottlenecks in rush hours to better manage human resource for future needs

FEATURES:

Customer user interface

- **Personalized customer account –** Customers can create accounts to log into the application securely and store personal information needed to utilize the ordering system. Customers can view their orders from their customized dashboard which also feature filtering capabilities.
- **Integrated restaurants database –** Customers can select from a database of restaurants that they wish to order from. The restaurant menu is integrated in the database, which allows customers to view and perform their ordering from directly.
- **User-friendly ordering interface –** A flexible ordering flow allows customer to review and edit their order as well as updating any necessary information prior to confirming an order. Users can choose to mark their orders as dine-ins or take-outs.
- **Automated confirmation email and status update page –** An email containing order information will be sent to the customer once their order is processed. A link to an order status page is also provided where the order status can be tracked from page in real-time as the restaurant works on the order.

Business user interface

- **Personalized business account –** Restaurant owners will use a business account which allows them access and control from end-to-end within the ordering process. This includes menu creation, customization and updating information (unit price, inventory levels, description) when needed.
- **Order Tracking System –** The order dashboard allows restaurant owners to view and update order status as orders are being fulfilled in the kitchen. This dashboard also contains filtering capabilities to view only orders in progress or all orders.
- **Sales Tax Integration –** Sales taxes are calculated and applied to the customer bills based on the zip code of the restaurant to comply with state sales tax legislations.
- **Inventory tracking –** Inventory levels for all menu items are tracked and updated in real-time after each order is successfully processed from the customer's end. It features a reordering capability that pushes an email to vendors, informing them to restock specific menu items.

```python
@bp.route('/<rid>/<order_id>', methods=['GET'])
@business_check_user_login
def get_order_details(restaurant_username, restaurant_id, rid, order_id):

    order_table=dynamodb.Table('order')

    order_data = order_table.query(
        KeyConditionExpression=Key('order_id').eq(order_id)
    )

    try:
        #dynamodb tables
        menu_table=dynamodb.Table('menu_item')
        order_item_table=dynamodb.Table('order_item')

        #declare variables
        oi_id = order_data['Items'][0]['oi_id']
        food_list = []

        for each in oi_id:
            #query order_items to get the item_id, quantity and unit_price
            oi_data = order_item_table.query(
                KeyConditionExpression=Key('order_item_id').eq(each)
            )
            #query menu table for the item_name and item_unit price
            menu_data = menu_table.query(
                KeyConditionExpression=Key('menu_item_id').eq(oi_data['Items'][0]['item_id'])
            )

            #To display in order review
            item_name = menu_data['Items'][0]['item_name']
            oi_quantity = oi_data['Items'][0]['oi_quantity']
            item_unit_price = menu_data['Items'][0]['item_unit_price']
            oi_unit_price = oi_data['Items'][0]['oi_unit_price']

            order_details = {
                'item_name': item_name,
                'oi_quantity': oi_quantity,
                'item_unit_price': item_unit_price,
                'oi_unit_price': oi_unit_price
            }
            food_list.append(order_details)

        order_total = order_data['Items'][0]['order_total']
        food_list.append({'order_total': order_total})
        food_list = json.dumps(food_list, cls=DecimalEncoder)

        return (food_list)
```