SRH Hochschule Heidelberg

Advanced Computer Science

HIGH ON FLAVOURS

April-June 2019

Author:

Nithin Bhardwaj Sridhar

Supervisor:

Prof. Dr. Mohammed Yass

Table of Contents

Introduction

The application “High on flavours” is an end to end restaurant management application to manage the orders as per the customer need, process the data to the kitchen and provide a bill at the end of the transaction in form of a PDF. The application has two interfaces namely the “Waiter” and the “Kitchen” all the orders placed by the customer are initially processed by the waiter interface, the kitchen interface is only used to provide an update to the waiter interface when the food cooked is ready to be served.

Initial wireframes and concept

The below images are the initial wireframe and concept design for the application,

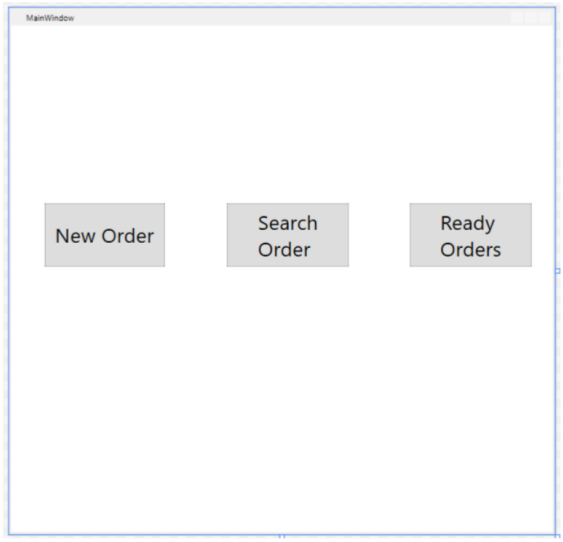

Figure 1: Landing page concept

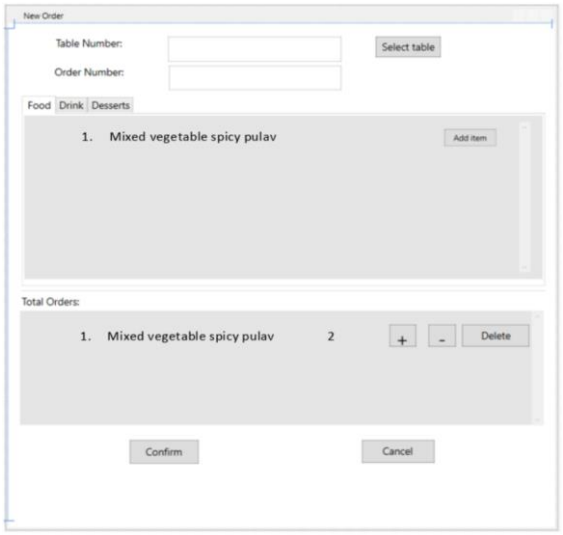

Figure 2: New orders page concept

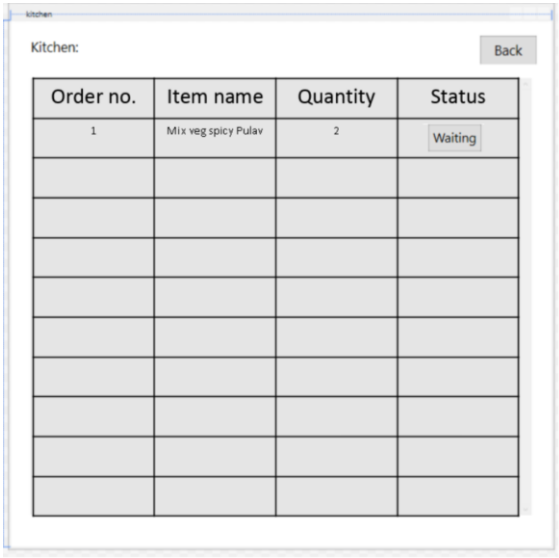

Figure 3: Kitchen window concept

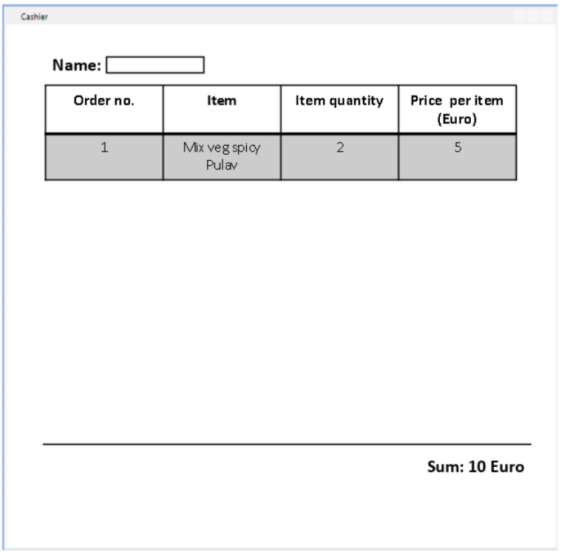

Figure 4: Billing page concept

Upon multiple feedbacks and considering the Gestalt and Dialog principles multiple changes were made to meet the requirements.

Project methodology:

An iterative and incremental software engineering methodology was used to build the application. Each functionality was built separately in a 5-day sprint with unit testing and was integrated with the initial build of the application followed by an integration testing.

Sprint 1: Initial Skeleton of the application, UI conceptual design, and switch between different pages from the home page.

Sprint 2: New order page element functionality i.e. increment and decrement, push orders to kitchen and change the status with help of a converter class and bind the content to a button.

Sprint 3: Orders ready to serve are populated in Ready orders screen on click of the status button the status of the order changes from ready to serve to delivered and the content is added to the bill collection, which is further used to find the total bill amount of all the orders for a table.

Sprint 4: Billing screen implementation to close the order for a table and this screen also provides an option to generate the bill in the form of a PDF.

Complete details about application and all the builds can be found at;
https://github.com/Nithinbs18/ACS-WPF-application

Users:

Two users involved in the application:

1. **Waiter** – receives orders, closes the order and prints the bill.
2. **Chef** - the status of the orders places is maintained by the chef as he changes the state in the process of cooking the ordered item.

User stories:

1. As a waiter, I would like to receive a new order, comment for order e.g. less spicy, change the quantity of the order, delete or decrement the ordered item on behalf of the customer.
2. As a waiter, I would like to know when the ordered item is cooked and ready to be served.
3. As a waiter, I would like to update an open order when the customer places a new food item.
4. As a waiter, I would like to close the order when the customer is done with his/her dining and would like to print a bill.
5. As a chef, I would like to view all the orders that are placed by the customer using the waiter interface.
6. As a chef, I would like to change the status of the ordered item when I start preparing and complete cooking of the food, i.e. from waiting to in progress and ready to serve.

Technology description

The application uses two XML files to persist the data, one containing the information about the food items and the other is to save the transaction information of all the orders in the restaurant. Both files are read when the application is launched and only the transaction XML is updated each time an order is closed and is persisted when the application is closed. Syncfusion PDF libraries are used to generate the bill in form of a pdf.

Gestalt and Dialog principles references

The UI of the application is built considering the Gestalt principles, Dialog principles and the guidelines of visual presentation of information. All the elements that are related are placed closely which follows the Proximity principle (e.g. buttons to alter the food item). The tables, buttons follow the Similarity principle as they all have a similar look. In the complete application information is communicated quickly (e.g. alert when there is a ready order, invalid selections made, etc.), content is distinct from each other and compact at the same time providing only the necessary information. Legibility and Comprehensibility are given high importance and suitable for the end users meeting the expectations.

Please find the below screenshots of the final application;

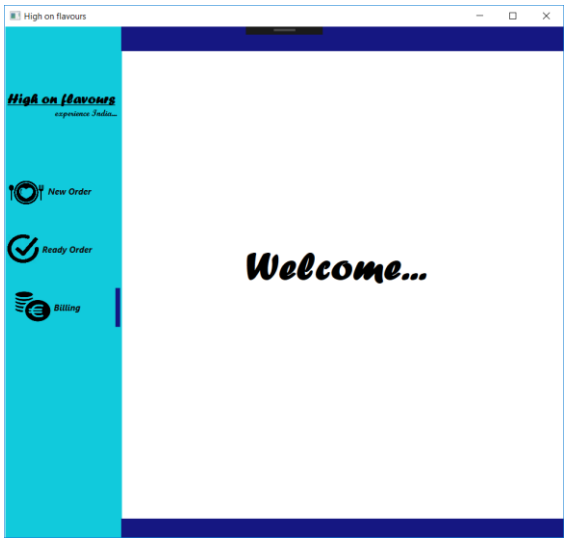

Figure 1: Landing page

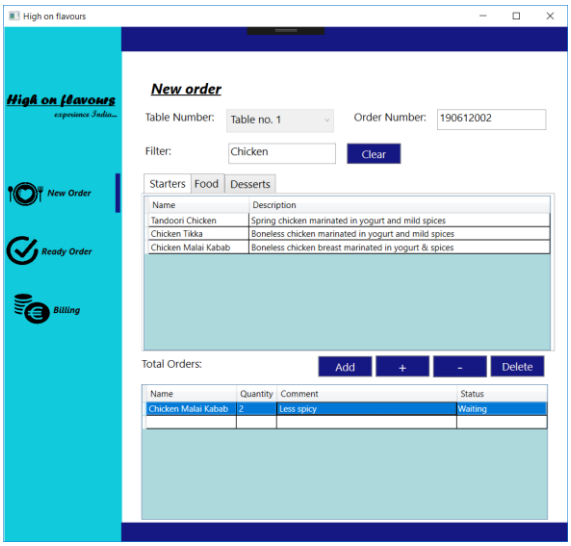

Figure 2: New orders

Figure 3: Kitchen Window

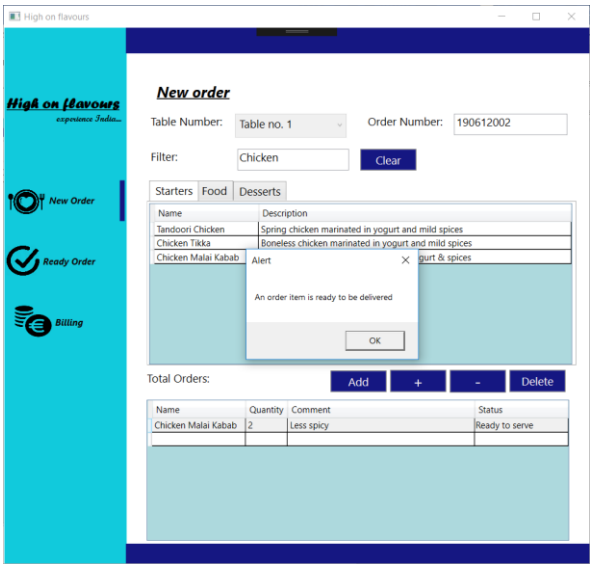

Figure 4: Alert on food is ready to serve

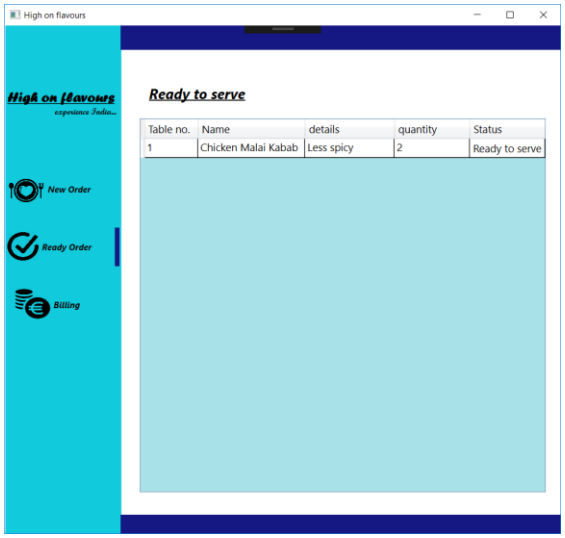

Figure 5: Ready products page

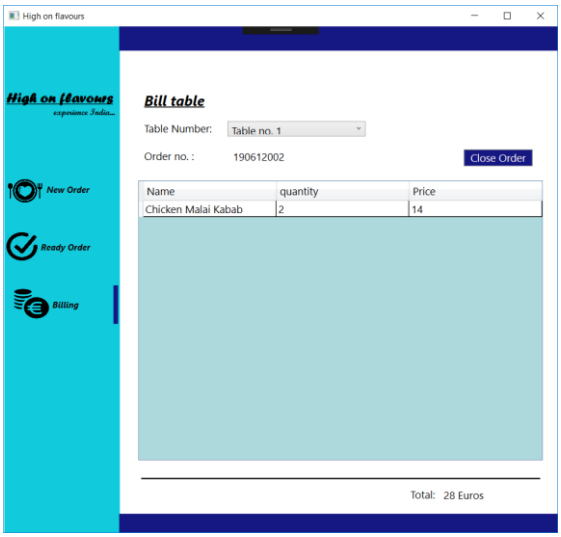

Figure 6: Billing page

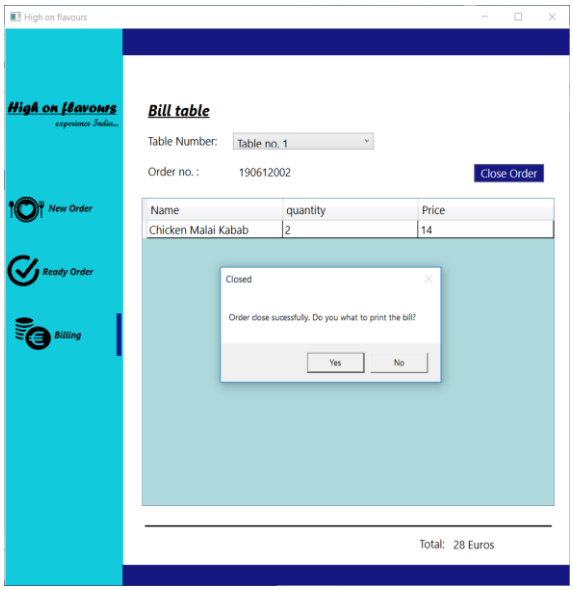

Figure 7: Print bill confirmation

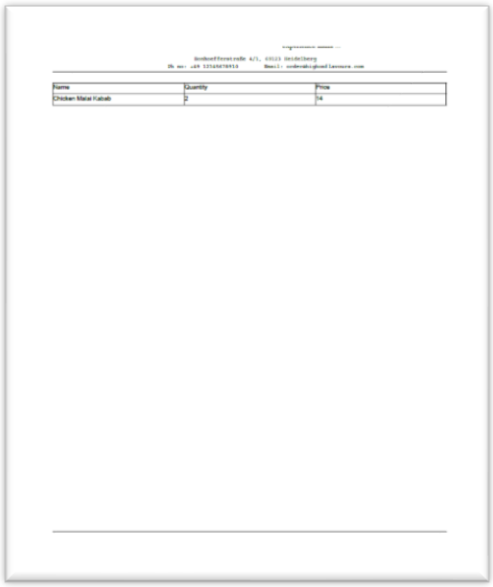

Figure 8: Sample bill

Conclusion and Future scope

The application provides a user-friendly and error tolerant experience to the management team to carry out end to end transaction in a restaurant. This application is designed to adapt to future changes and modifications i.e. it is scalable.

As a future scope, an admin page can be integrated to find out details about all the transaction history of the restaurant.