

ВВЕДЕНИЕ

Данная работа посвящена разработке прототипа системы автоматизации учета заказов для ресторана.

Актуальность работы обусловлена реалиями настоящего времени, которые требуют использования инноваций в производственном процессе ресторанов. Делается это с целью повышения конкурентоспособности ресторана в бизнесе. Автоматизация процесса учета заказов сокращает время на обслуживание клиентов, позволяет выполнять заказы точно и не допускать неточностей.

Основной идеей данного проекта является автоматизация производственного процесса ресторана путем введения системы оформления и расчёта заказов, которая, в свою очередь, позволяет решить такие задачи, как повышения качества и скорости обслуживания гостей, контроль действий персонала, сокращение ошибок при работе с клиентами «человеческий фактор».

ГЛАВА 1. РАЗРАБОТКА ПРОЕКТА ПРОТОТИПА ИНФОРМАЦИОННОЙ СИСТЕМЫ

1. Анализ предметной области

В качестве объекта исследования был взят некий ресторан, в котором необходимо автоматизировать оформление и расчёт заказов. Рассмотренный ресторан предоставляет клиентам блюда определенных разделов: холодные закуски, супы, горячие блюда, гарниры, салаты, десерты, напитки. В каждом заказе может быть указано несколько наименований блюд разного количества. Руководству ресторана необходимо автоматизировать оформление заказов, подсчет итоговой суммы каждого заказа, предоставляет возможность автоматически делать скидки в “счастливые часы” «скидка действует в определенные часы». В том случае если время заказа попадает на несколько скидок сразу, то система рассчитывает заказ по минимальной из скидок.

База данных содержит всю информацию о пользователях, блюдах и заказах.

Система позволяет официантам оформлять заказы, а также безошибочно рассчитывать итоговую сумму заказов с учетом скидки. Во-втором случае система помечает данный заказ как выданный, и, далее, система более не выводит этот заказ в список заказов для расчета.

2. Составление глоссария проекта

Глоссарий позволяет описать терминологию предметной области. Ниже приведен словарь данных системы «Таблица 1»:

Таблица 1

3. Функциональные и нефункциональные требования к системе.

К функциональным требованиям данной системы можно отнести следующее:

- оформление заказов «содержание заказа и прочая необходимая информация о нем»
- подсчет итоговой суммы заказа.
- предоставление скидки в определенные часы

К нефункциональным требованиям данной системы можно отнести следующее:

- присутствие интуитивно-понятного для пользователя интерфейса
- хранение паролей пользователей в зашифрованном виде
- исключение дублирования логинов пользователей
- содержание нормализованной базы данных
- реализация запросов к базе данных должна непосредственно производиться в классах сущностей
- реализация системы должна быть выполнена с использованием ООП

4. Сравнение с другими альтернативными системами

На рынке существует множество подобных систем, которые упрощают жизни владельцам ресторанов. Такие приложения, как правило, позволяют решать целый ряд проблем. Например, наиболее важные цели владельцев – ускорение обслуживания гостей, контроль сотрудников и снижение вероятности обмана с их стороны.

Существует три типа систем автоматизации ресторана - это автономные, облачные и гибридные, где каждый имеет свои минусы. Автономные системы не защищают от обмана со стороны сотрудников, а облачные целиком и полностью зависят от наличия доступа к интернету.

И, наконец, гибридный вариант системы автоматизации, ярким примером которого является, например, Jowi. Где в каждом ресторане «если речь идет о целой сети ресторанов» устанавливается локальный модуль, а все данные сохраняются в удаленный модуль. Это приложение позволяет не беспокоиться о скорости интернета.

Другим довольно ярким примером такого же рода системы является приложение iiko «айко» — самая современная и мощная система на рынке автоматизации предприятий общественного питания.

iiko позволяет осуществлять:

- Ввод заказов через настраиваемые меню и характеристики;
- Простое разделение, объединение и перенос блюд и столов;
- Управление столами, информация об их состоянии в реальном времени;
- Мониторинг скорости и качества выполнения заказов официантами;
- Экранная клавиатура для передачи спец. указаний на кухню;
- Экранные сообщения о специальных сезонных и ежедневных предложениях;
- Автоматическое разделение чека по гостям;
- Отчеты о продажах: по официантам, блюдам, дням, времени суток, клубным картам;

Если сравнивать данную систему с системой, разрабатываемой в данной работе, то можно непосредственно выделить достаточно большое количество различий. Так, например, преимущества системы iiko относительно разрабатываемого в данной работе приложения выглядят следующим образом:

- Система поддерживает просмотр забронированных/незабронированных столов
- Система предусматривает мониторинг скорости и качества выполнения заказов официантами;

- Система предоставляет отчеты о продажах.

Также можно выделить и один немаловажный недостаток системы iiko:

- Платное использование системы

5. Описание действующих лиц системы

Данное приложение включает в себя следующих действующих лиц:

Система – получает информацию о заказе, рассчитывает итоговую сумму заказа с учетом скидок.

Официант – оформляет заказ «выбирает стол, заказанные блюда, выбирает количество одного блюда», вносит исправления в промежуточный список заказа в случае ошибки оформления заказа, отправляет информацию заказа системе для его расчета.

Администратор – ведет информацию об официантах, ведет информацию о блюдах ресторана, ведет информацию об акции «счастливые часы», ведет информацию о блюдах.

6. Описание вариантов использования

Исходя из вышеперечисленных действующих лиц системы можно составить следующий список вариантов использования:

1. Войти в систему
2. Ведение информации о блюдах
3. Ведение информации о сотрудниках
4. Ведение условий акции
5. Ведение заказа

Подробное описание данных вариантов использования представлены ниже.

1. Вариант использования "Войти в систему"

Краткое описание:

Данный вариант использования описывает вход пользователя систему для манипуляции данными.

Основной поток событий:

Данный вариант использования начинается выполняться, когда пользователь хочет получить доступ к данным системы.

1. Система генерирует стартовое окно, где запрашивает логин и пароль.
2. Пользователь вводит логин и пароль.
3. Система подтверждает логин, пароль и определяет пользователя и его должность
4. После чего открывается доступ к допустимым для должности данным.

Альтернативные потоки:

1. Пользователь ввел неверные логин или пароль. При неправильных логине и пароле система выведет сообщение о том, что пользователь ввел неверные логин или пароль. В этом случае пользователь вернется к пункту 2 основного потока.
2. Отсутствует соединение с базой данных. В этом случае система генерирует новое окно с предупреждением о том, что системе не удастся связаться с базой данных.

Предусловие:

Необходимо знать логин и пароль пользователя

2. Вариант использования " Вести информацию о сотрудниках"

Краткое описание:

Данный вариант использования позволяет администратору вносить, обновлять, удалять информацию о сотрудниках.

Основной поток событий:

1. Система генерирует новое окно с выбором раздела:
 - Сотрудники
 - Акции

-Блюда

2. Администратор выбирает раздел «Сотрудники»
3. Система генерирует новое окно со списком сотрудников и запрашивает действие.

Внести данные:

1. Администратор выбирает действие «Добавить»
2. Система генерирует новое окно для ввода данных.
3. Администратор вводит следующую информацию о сотрудниках: Фамилия, Имя, Отчество «при наличии», Логин, Пароль, Должность

4. Администратор подтверждает свое действие.
5. Система проверяет введенную информацию.
6. Система сохраняет введенные данные.

Изменить данные:

1. Администратор выбирает строку с сотрудником, информацию о котором необходимо изменить и выбирает действие «Изменить»

2. Система генерирует окно для изменения информации

Администратор изменяет следующую информацию: Фамилия, Имя, Отчество «при наличии», Логин, Пароль, Должность «администратор или официант»

3. Администратор подтверждает свое действие.
4. Система проверяет введенные данные.
5. Система сохраняет введенные данные.

Удалить данные:

1. Администратор выбирает необходимую строку с сотрудником и выбирает действие «Удалить».
2. Система удаляет данные, выводит сообщение об успешном удалении и запрашивает действие «Обновить»

Альтернативные потоки:

Внести данные:

1. Введены ФИО неверного формата.
2. Не присвоена должность
3. Логин занят

В этих случаях система выдаст сообщение об ошибке.

После чего система перейдет на 2 пункт данного подчиненного потока.

Изменить данные:

Введены ФИО неверного формата. В этом случае выдается сообщение об ошибке.

После чего система перейдет на 2 пункт данного подчиненного потока.

Удалить данные:

Удаляемый сотрудник прикреплен к какому-либо заказу или удаляемый сотрудник в данный момент находится в системе. В этом случае система не удалит данного сотрудника.

Предусловия:

Администратору необходимо войти в систему.

3. Вариант использования "Вести информацию о блюдах"

Краткое описание:

Данный вариант использования позволяет администратору вносить, обновлять, удалять информацию о блюдах.

Основной поток событий:

1. Система генерирует новое окно с выбором раздела:
 - Сотрудники
 - Акции
 - Блюда
2. Администратор выбирает раздел «Блюда»
3. Система генерирует новое окно со списком блюд и запрашивает действие.

Внести данные:

- a. Администратор выбирает действие «Внести»
- b. Система генерирует новое окно для ввода данных.
- c. Администратор вводит следующую информацию о блюде:

Название блюда, раздел, доступность, стоимость.

- d. Администратор подтверждает свое действие.
- e. Система проверяет введенную информацию о блюде.
- f. Система сохраняет введенные данные.

Изменить данные:

- a. Администратор выбирает строку с блюдом, информацию о котором необходимо изменить и выбирает действие «Изменить».

- b. Система генерирует окно для изменения информации

- c. Администратор изменяет следующую информацию:

Название блюда, Доступность, Стоимость.

- d. Администратор подтверждает свое действие.
- e. Система проверяет введенные данные.
- f. Система сохраняет введенные данные.

Удалить данные:

- a. Администратор выбирает необходимую строку с блюдом и подтверждает свое действие.

- b. Система удаляет данные, выводит сообщение об успешном удалении и запрашивает действие «Обновить»

Альтернативные потоки:

Внести данные:

1. Введен неверный формат одного из полей. В этом случае выдается сообщение о том, что пользователь ввел неверный формат.

После чего администратор перейдет на 3 пункт данного подчиненного потока.

2.Администратор не ввел информацию. В этом случая выдается сообщение о том, что пользователь ввел неверный формат.

Изменить данные:

1.Введенные поля имеют неверный формат. В этом случая выдается сообщение о том, что пользователь ввел неверный формат. После чего администратор перейдет на 3 пункт данного подчиненного потока.

Предусловия:

Администратору необходимо войти в систему.

4. Вариант использования "Вести информацию об акции «счастливые часы»"

Краткое описание:

Данный вариант использования позволяет администратору вносить, обновлять, удалять информацию об акции «Время начала, Время окончания, Скидка»

Основной поток событий:

1. Система генерирует новое окно с выбором раздела:
 - Сотрудники
 - Акции
 - Блюда
2. Администратор выбирает раздел «Акции»
3. Система генерирует новое окно со списком блюд и запрашивает действие.

Внести данные:

- a. Администратор выбирает действие «Добавить».
- b. Система генерирует новое окно для ввода данных.
- c. Администратор вводит следующую информацию об акции:

Время начала, Время окончания, Размер скидки.

- d. Администратор подтверждает свое действие.
- e. Система проверяет введенную информацию об акциях.

f. Система сохраняет введенные данные.

Изменить данные:

a. Администратор выбирает строку с акцией, информацию о котором необходимо изменить и выбирает действие «Изменить».

b. Система генерирует окно для изменения информации

c. Администратор изменяет следующую информацию: Время начала, Время окончания, Размер скидки.

d. Администратор подтверждает свое действие.

e. Система проверяет введенные данные.

f. Система сохраняет введенные данные.

Удалить данные:

a. Администратор выбирает необходимую строку с блюдом и подтверждает свое действие.

b. Система удаляет данные, выводит сообщение об успешном удалении и запрашивает действие «Обновить»

Альтернативные потоки:

Внести данные:

1. Введен неверный формат одного из полей. В этом случае выдается сообщение об ошибке.

После чего система перейдет на 2 пункт данного подчиненного потока.

Изменить данные:

Введенные поля имеют неверный формат. В этом случае выдается сообщение об ошибке. После чего система перейдет на 2 пункт данного подчиненного потока.

Предусловия:

Администратору необходимо войти в систему.

5. Вариант использования “Ведение заказа”

Краткое описание:

Данный вариант использования позволяет официанту вести информацию о заказе.

Основной поток событий:

1. Система генерирует новое окно с выбором раздела:
 - Оформить заказ
 - Рассчитать заказ
 - «Оформить заказ»:
1. Официант выбирает «Оформить заказ»
2. Система генерирует новое окно и выводит список столов
3. Официант выбирает необходимый стол.
4. Система выводит список блюд
5. Официант выбирает блюдо и указывает его количество «до тех пор, пока все необходимые блюда не будут выбраны»
6. Выбирает действие посмотреть состав заказа
7. Система генерирует новое окно со списком блюд в заказе
8. Официант удостоверяется, что заказ верный и подтверждает действие.
9. Система возвращается к меню со списком блюд.
10. При необходимости официант добавляет блюда в заказ и подтверждает окончание формирования заказа.
11. Система формирует информацию о заказе и сохраняет ее.

Альтернативные потоки:

Невозможно установить связь с базой данных или системе не удалось сохранить заказ, тогда система отправит сообщение об ошибке.

- «Рассчитать заказ»
 1. Официант выбирает «Рассчитать заказ»
 2. Система предоставляет генерирует окно со списком заказов
 3. Официант выбирает необходимый заказ

4. Система выводит информацию о заказе с его итоговой суммой.

Официанту необходимо изменить заказ:

- Официант сообщает системе о том, что необходимо изменить список заказа
- Система генерирует новое окно со списком блюд в заказе.
- Официант удаляет необходимые блюда и подтверждает окончание изменения заказа.
- Система сохраняет изменения и возвращается к предыдущему окну

5. Официант сообщает, что заказ верный

6. Система формирует и выводит чек.

7. Система помечает заказ как выданный.

Альтернативные потоки:

Невозможно установить связь с базой данных или системе не удалось выдать чек, тогда система отправит сообщение об ошибке.

Также ниже приведена диаграмма использования:

Рисунок 1

7. ER-модель базы данных системы

Рисунок 2

Описание сущностей ER-модели:

Меню - содержит информацию о разделах в меню

Список блюд - содержит информацию о всех блюдах ресторана

Создержание заказов - содержит состав каждого заказа «так как отношение между «Заказы» и «Список блюд» М:М – была создана таблица содержание заказов»

Заказы – содержит необходимую информацию о заказах

Акции – содержит информацию о вариантах акции «счастливые часы»

Список сотрудников- содержит информацию об официантах

Список должностей -содержит список должностей

Список столов-содержит список столов

Данная ER- модель является нормализованной и находится в третьей нормальной форме.

В конечном итоге составляется следующая диаграмма классов:

Рисунок 3

8. Макеты разрабатываемой системы

Макет 1 -Форма входа

При неправильном пароле или логине выходит сообщение «Макет2»

Макет 2-Форма входа при неправильных пароле или логине

Если не удалось связаться с базой данных генерируется диалоговое окно с сообщением «Макет 3»

Макет 3 - Форма для сообщений пользователю

Макет 4 – Меню Администратора

Макет 5- Форма с необходимым списком «Администратор»

Макет 6-Изменение информации о блюде «Администратор»

- При нажатии «Удалить» при предварительном выделении блюда система удалят данную строку.

- При нажатии «Обновить» система обновляет список в поле.

- При нажатии «Сохранить» система сохраняет список и возвращается к окну Макета 5.

2. В том случае, если на Макете 4 была нажата кнопка «Сотрудники» система на Макет 5 и выводит список всех сотрудников.

- При нажатии «Изменить» при предварительном выделении строки система генерирует окно Макета 7.

- При нажатии «Добавить» система генерирует окно Макета 7

- Кнопка «Сохранить» позволяет «Сохранить» новую введенную информацию.

- Кнопка «Отмена» возвращает к окну Макета 5

Макет 7- Изменение информации о сотруднике «Администратор»

3. В том случае, если на Макет 4 была нажата кнопка «Акции» система на Макет 5 выводит список всех акций.

- При нажатии «Изменить» при предварительном выделении строки система генерирует окно Макета 6.

- При нажатии «Добавить» система генерирует окно Макета

- Кнопка «Сохранить» позволяет «Сохранить» новую введенную информацию.

- Кнопка «Отмена» возвращает к окну Макета 3

- При нажатии УДАЛИТЬ при предварительном выделении строки система удалят данную строку.

- При нажатии «Обновить» система обновляет список в поле.

- При нажатии ««Сохранить»» система сохраняет список и возвращается к окну Макета 2

Макет 8 - Изменение информации о скидке «Администратор»

В том случае, если вошел официант появляется следующее окно: «Макет 9»

Макет 9 – Меню официанта

1. При нажатии на кнопку «Завершить сеанс» система переходит на стартовую окно «Макет 1»

2. При нажатии «Создать заказ» система генерирует новое окно «Макет 10» со списком столов «Необходимо обязательно выбрать номер стола». Далее генерируется новое окно «Макет 11». Состав заказа можно посмотреть, нажав на «Заказ» система генерирует следующее окно «Макет 12»:

Макет 10 – Оформление заказа. Выбор стола. «Официант»

Макет 11- Оформление заказа. Формирование заказа. «Официант»

Макет 12- Оформление заказа. Просмотр содержания предварительного списка заказа. «Официант»

1. При нажатии «Итоговая сумма заказа» «Макет 9» система генерирует окно Макета 13

Макет 13- Расчет итоговой суммы заказа. «Официант»

Необходимо выбрать номер заказа

При нажатии «Выдать» система помечает заказ как выданный и переходит на окно Макета 9

При нажатии «Изменить» система сгенерирует окно Макета 12

При нажатии НАЗАД система вернется к окну Макета 9

ГЛАВА 2. РАЗРАБОТКА ПРОТОТИПА ИНФОРМАЦИОННОЙ СИСТЕМЫ

Программа разработана на основе Windows Forms с помощью технологии ADO.NET Entity Framework на языке C#. Ниже представлено описание программной части системы:

При запуске системы генерируется стартовое окно «Макет 1».

На стартовой странице система запрашивает логин и пароль пользователя.»Макет 1». После заполнения полей логина необходимо нажать на кнопку «Войти». Ниже представлен обработчик нажатия кнопки «Войти»«Рисунок 4».

Рисунок 4-Обработчик кнопки «Войти»

В данном обработчике создается экземпляр класса employee, по которому идет поиск пользователя по логину и паролю. И, в зависимости от должности, система генерирует следующие окна, либо выводит сообщение о неправильном пароле или логине. Введённый пароль шифруется для поиска его в базе данных. «Рисунок 5»

Рисунок 5 – Конструктор класса сущности employee

В том случае, если вошел администратор система генерирует следующее окно «Макет 4»

1. При нажатии кнопки «Сотрудники» система генерирует новое окно «Макет 5» с информацией обо всех сотрудниках.
2. При нажатии кнопки «Блюда» система генерирует новое окно «Макет 5» с информацией обо всех блюдах.
3. При нажатии кнопки «Акции» система генерирует новое окно «Макет 5» с информацией обо всех акциях.
4. При нажатии кнопки «Выйти», система переходит на стартовое окно «Макет 1»

В том случае, если на Макете 4 было нажато «Сотрудники» система перейдет на Макет 5 и выведет список всех сотрудников.

Рисунок 6 -Инициализация формы с Макета 5

Рисунок 7 - Заполнение списка Макета 5

· При нажатии «Изменить» при предварительном выделении сотрудника система генерирует следующее окно «Макет 7». Ниже приведен обработчик события кнопки «Изменить»

Рисунок 8- Обработчик кнопки "Изменить" Макета 5

Рисунок 9 – Поиск сотрудника из выделенной строки со списком сотрудников

Рисунок 10 - Инициализация формы для изменения информации о сотруднике

○ Кнопка «Сохранить» позволяет «Сохранить» новую введенную информацию.

Рисунок 11- Обработчик кнопки "Сохранить"

Рисунок 12

Рисунок 13- Сохранение информации

○ Кнопка «Отмена» возвращает к предыдущему окну «Макет 5»

Рисунок 14 -Обработчик кнопки "Отмена"

- При нажатии «Удалить» при предварительном выделении строки система удалят данную строку. «Макет 5» «Рисунок 15»

Рисунок 15 - Обработчик кнопки "Удалить"

Рисунок 16

- При нажатии «Обновить» система обновляет список в поле. «Макет 5»
«Рисунок 17»

Рисунок 17 - Обработчик кнопки "Обновить"

-При нажатии «Сохранить» система сохраняет список и возвращается к окну Макета 4 «Макет 4»

ЗАКЛЮЧЕНИЕ

В данной курсовой работе были реализованы все поставленные задачи.

Приложение работает на основе разработанной базы данных на языке SQL.

Разработанная база является нормализованной.

Разработанная система имеет интуитивно понятный интерфейс, что способствует быстрому овладению программы пользователем.

Системы автоматизации ресторанного бизнеса в будущем станут еще более функциональными, а число использующих их заведений обязательно будет расти. Ведь в данных системах реализована и функциональность для решения разного рода задач для этой сферы— например, бронирование столов, или сбор отзывов посетителей. Кроме всего прочего, в обязательном порядке появятся и решения по автоматизации и внедрение дополнительных мер для контроля новых аспектов функционирования ресторана — например, скорости реагирования на нажатие кнопки вызова официанта.